# Development of New ATLAS Trigger Algorithms in Search for New Physics at the LHC

Dissertation
zur Erlangung des Grades

Doktor der Naturwissenschaften

am Fachbereich Physik, Mathematik und Informatik
der Johannes Gutenberg-Universität
in Mainz

**Marcel Weirich**
geboren in Weilburg

Mainz, den 28.04.2021

# Development of New ATLAS Trigger Algorithms in Search for New Physics at the LHC

**Marcel Weirich**

Dissertation
zur Erlangung des Grades

Doktor der Naturwissenschaften

am Fachbereich Physik, Mathematik und Informatik
der Johannes Gutenberg-Universität
in Mainz

1. Berichterstatter: [in der elektronischen Version aus Datenschutzgründen entfernt]
2. Berichterstatter: [in der elektronischen Version aus Datenschutzgründen entfernt]

Datum der mündlichen Prüfung: 06.10.2021

Ich versichere, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Mainz, den 28.04.2021

Marcel Weirich

*No amount of money*
*ever bought a second of time.*

( Tony Stark | Avengers: Endgame )

# Kurzfassung

Um die Natur der Physik weiter zu erforschen, werden vom Large Hadron Collider (LHC) immer höhere Luminositäten erzielt. Mit dem Neustart des LHC im Jahr 2022 muss das ATLAS-Experiment und insbesondere das dedizierte Triggersystem mit erhöhten Ereignisraten zurechtkommen. Die erste Triggerstufe wird unter Verwendung der neuesten FPGA-Technologie (Field Programmable Gate Array) neu gestaltet, um u. a. Daten aus den elektromagnetischen und hadronischen Kalorimetern mit höherer Granularität auszuwerten. Als Teil davon wird ein neues Subsystem mit dem Namen jet Feature EXtractor (jFEX) hinzugefügt. Dessen Aufgabe ist es, Teilchenschauer (jets) zu identifizieren und globale Energiesummen zu berechnen.
Im Rahmen dieser Arbeit wurde eine neue Algorithmus-Firmware für das jFEX-System entwickelt. Die neu entwickelten Trigger-Algorithmen wurden auf dem Ziel-FPGA implementiert, wobei eine maximale Verarbeitungszeit von 125 ns benötigt wird, was deutlich innerhalb der verfügbaren Latenz von 150 ns liegt. Begleitend wird die Leistungsfähigkeit der neuen Triggeralgorithmen vorgestellt, die verschiedene Verbesserungen gegenüber dem alten System aufzeigt. Darüber hinaus wurden Studien zur Triggereffizienz durchgeführt, welche sich auf Ereignisse mit unsichtbar zerfallenden Higgs-Bosonen beziehen, die in Verbindung mit hadronisch zerfallenden $W$- oder $Z$-Bosonen erzeugt werden. Mit dem neu entwickelten Algorithmus zur Berechnung von fehlender Transversalenergie ($E_{\mathrm{T}}^{\mathrm{miss}}$) kann bei gleichbleibender Rate die zugehörige Triggerschwelle um mehr als 30 GeV reduziert werden, was wiederum eine Grundlage für eine mögliche Sensitivitätserhöhung für Ereignisse mit niedrigem $E_{\mathrm{T}}^{\mathrm{miss}}$ bildet. Nach dem Standardmodell der Elementarteilchenphysik ist der Anteil an Higgs-Bosonen, die in einen unsichtbaren Endzustand zerfallen, eher gering, so dass eine Beobachtung solcher Ereignisse ein direkter Hinweis auf neue Physik wäre.

# Abstract

To further explore the physics nature, ever higher luminosities will be achieved by the Large Hadron Collider (LHC). With the restart of the LHC in 2022, the ATLAS experiment, and in particular the dedicated trigger system, has to cope with increased event rates. The first level trigger is redesigned taking advantage of the latest Field Programmable Gate Array (FPGA) technology to i.a. exploit higher granularity data from the electromagnetic and hadronic calorimeters. As part of this, a new subsystem called jet Feature EXtractor (jFEX) is added. It is intended to identify particle showers (jets) and to process global energy sums.
In the context of this thesis, new algorithm firmware was developed for the jFEX system. The newly developed trigger algorithms were implemented appropriately on the target FPGA by requiring a maximum processing time of 125 ns, which is well within the available latency budget of 150 ns. Accompanying, the performance of the new trigger algorithms is presented, showing several improvements over the legacy system. Moreover, trigger efficiency studies were carried out targeting events with invisibly decaying Higgs bosons, which are produced in association with hadronically decaying $W$ or $Z$ bosons. With the newly developed algorithm for computing missing transverse energy ($E_{\mathrm{T}}^{\mathrm{miss}}$), the corresponding trigger threshold can be reduced by more than 30 GeV without changing the rate, which in turn forms a basis for a potential gain in sensitivity for events with low $E_{\mathrm{T}}^{\mathrm{miss}}$. According to the Standard Model of particle physics, the fraction of Higgs bosons decaying to an invisible final state is rather small, so that an observation of such events would be a direct indication of new physics.

# Contents

# I

# General Introduction and Explanation

CHAPTER 1

# Introduction and Motivation

*I believe (...)*
*that one of the strongest motives*
*that leads men to art and science*
*is to escape from everyday life*
*with its painful crudity*
*and hopeless dreariness,*
*from the fetters*
*of one's own ever shifting desires.*

( Albert Einstein )

The Standard Model of particle physics was developed in the 1960s and is one of the most successful and well tested theories. Over the course of time, various experiments have confirmed the Standard Model and sought answers to open questions. Nevertheless, it is incomplete as e.g. it does not incorporate the full theory of gravitation, nor does it contain any viable dark matter particle. The latter is a postulated form of weakly interacting matter that is not directly visible, but interacts via gravity. Its presence is implied in a variety of astrophysical observations. Therefore, the nature of dark matter and its composition are important open questions.

One of the latest achievements came with the discovery of the Higgs boson by the Large Hadron Collider (LHC) experiments ATLAS and CMS in 2012 [1,2]. Since then, numerous Higgs boson measurements have been carried out, the results of which are compatible with the Standard Model expectations. However, the current measurement uncertainties can accommodate theories beyond the Standard Model (BSM). Higgs boson decays to invisible[1] final states are predicted by many BSM theories, where final state particles can be dark matter candidates.

According to the Standard Model, the fraction of Higgs bosons decaying to an invisible final state is rather small. An observation of such events would therefore be a direct indication of new physics and potential evidence of Higgs bosons decaying to dark matter or other unknown weakly interacting particles.

The higher the luminosity of a particle accelerator, the more data can be gathered in order to be able to observe such rare processes. In the future, ever higher luminosities will be achieved by the LHC, exploring further the physics nature, but also placing ever greater challenges on the trigger system of the ATLAS experiment. In addition to the increasing

[1] *Invisible* in this context means no or negligible interaction of the decay products with the detector.

event rates, the data from the electromagnetic and hadronic calorimeters are transmitted with increased granularity. To still efficiently trigger and record data, major upgrades of the trigger and data acquisition system are required. As part of this, the first level trigger will be equipped with a new subsystem, namely the jet Feature EXtractor (jFEX). It is intended to identify particle showers (jets) and to process global energy sums. jFEX will take advantage of the latest Field Programmable Gate Array (FPGA) technology to improve the physics object selection using more sophisticated algorithms.

This thesis describes the development of new algorithm firmware for the jFEX system, from the design concept of individual algorithms through associated trigger efficiency studies to their realization in hardware.
The primary goal is to improve the trigger efficiency of individual physics objects. The focus is particularly on the signatures in events with invisibly decaying Higgs bosons, which are produced in association with hadronically decaying $W$ or $Z$ bosons. An improvement in the underlying trigger efficiency forms a basis for a potential gain in sensitivity in the search for new physics.

The structure of this thesis is as follows:
The theoretical principles of the Standard Model are presented in Chapter 2, including a brief discussion of its limitations and open questions. Chapter 3 introduces the experimental background, starting with an overview of the LHC physics, followed by a detailed description of the ATLAS detector and the dedicated trigger system. The structure and functionality of FPGAs are explained in Chapter 4, while Chapter 5 is about the steps required to develop firmware for these hardware devices.
The jFEX processing functionality and hardware design are discussed in Chapter 6. It also gives an overview of the overall firmware architecture. A detailed description of the algorithm design concepts follows in Chapter 7, while general trigger efficiencies are then presented in Chapter 8. Chapter 9 is dedicated to the strategies with which the newly developed algorithms were implemented in firmware, including logic simulation and hardware tests. Results from the Higgs-to-invisble trigger efficiency studies are then presented in Chapter 10. Finally, Chapter 11 closes with a summary and a brief outlook on possible improvements in the future.

# CHAPTER 2

# Theoretical Background

*Explanations exist;*
*they have existed for all time;*
*there is always a well-known solution*
*to every human problem - neat, plausible,*
*and wrong.*

( Henry L. Mencken )

This chapter provides an overview of the rudimentary theoretical knowledge needed for this thesis. The very first section introduces the Standard Model of particle physics. Subsequently, the limits of the Standard Model are given at a glance.
Unless otherwise stated, the following content is based on References [3–5].

## 2.1 The Standard Model of Particle Physics

One can break down each and every known physical phenomenon using one or more of the four fundamental interactions of nature: strong force, electromagnetism, weak force and gravity. Table 2.1 provides an overview of their relative strength and range.

| **Interaction** | **Relative strength** | **Range [m]** | **Mediators** |
|---|---|---|---|
| Strong | 1 | $\approx 10^{-15}$ | Gluons |
| Electromagnetic | $10^{-2}$ | $\infty$ | Photons |
| Weak | $10^{-15}$ | $< 10^{-15}$ | $W$ and $Z$ bosons |
| Gravitational | $10^{-41}$ | $\infty$ | Gravitons (hypothetical) |

**Table 2.1:** Fundamental interactions of nature.

The Standard Model of particle physics contains all known elementary (fundamental) particles and describes the strong, electromagnetic and weak interactions. Generally, there are two classes of particles: fermions (particles with half-integer spin) and bosons (particles with integer spin). The Standard Model contains 12 flavors of elementary fermions plus their corresponding antiparticles, as well as 12 elementary vector bosons (gauge bosons) and the scalar Higgs boson.

These 12 vector bosons (eight gluons, photon, $W^+$, $W^-$, $Z$) are the force carriers that mediate the strong, electromagnetic and weak interactions (also listed in Tab. 2.1). The group of fermions is made of six quarks (up, down, charm, strange, top, bottom) and six leptons (electron, electron neutrino, muon, muon neutrino, tau, tau neutrino). In Figure 2.1 all these particles are grouped with respect to their various classficiations. The first generation forms the everyday matter: gluons, the carrier of the strong force, allow up and down quarks to cluster and form protons and neutrons; the electromagnetic force, carried by photons, lets electrons orbit these clusters to build atoms that combine into molecules and therefore into all known visible matter in the universe.

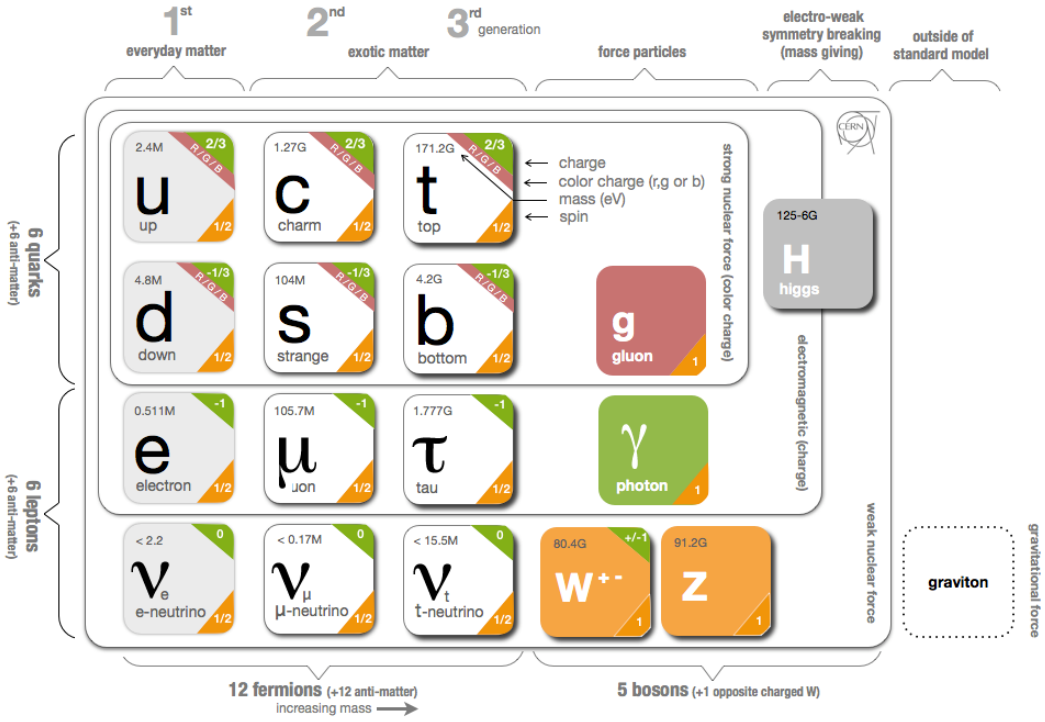



**Figure 2.1:** Elementary particles of the Standard Model [6].

The following subsections provide more detailed information on the elementary particles, fundamental forces and underlying theories within the Standard Model.

### 2.1.1 Quarks, Gluons and the Strong Interaction

Each (anti-) quark is a spin-1/2 particle (fermion) and carries an electromagnetic charge of either $\pm 1/3\, e$ or $\pm 2/3\, e$. In addition, quarks carry three types of color charge, while antiquarks carry three types of anticolor. A common choice for these color states is $\{r, g, b\} = \{\text{red, green, blue}\}$ and $\{\bar{r}, \bar{g}, \bar{b}\} = \{\text{antired, antigreen, antiblue}\}$, respectively. The mass[1] of quarks of different flavor ranges from $\mathcal{O}\left(10^0\right)$ MeV to $\mathcal{O}\left(10^5\right)$ MeV.
Gluons are spin-1 particles (bosons), do not carry an electromagnetic charge and are massless. As they are mediators of the strong interaction, gluons carry both color and anticolor. This results in $3 \times 3 = 9$ possible combinations of color and anticolor. According

[1] General Remark: In this thesis, natural units are used ($c = 1$ and $\hbar = 1$).

to group theory, they are splitted into two multiplets. The octet forms a system of eight ground states, from which all color states can be created. The choice of these states is a matter of convention. One variant is:

$$r\bar{g},\ r\bar{b},\ g\bar{b},\ g\bar{r},\ b\bar{r},\ b\bar{g},\ \frac{1}{\sqrt{2}}\cdot(r\bar{r}-g\bar{g}),\ \frac{1}{\sqrt{6}}\cdot\left(r\bar{r}+g\bar{g}-2\cdot b\bar{b}\right). \tag{2.1}$$

The color singlet

$$\frac{1}{\sqrt{3}}\cdot\left(r\bar{r}+g\bar{g}+b\bar{b}\right) \tag{2.2}$$

is symmetric with respect to the three colors and three anticolors, not color-specific, and therefore cannot be exchanged between color charges. As a result, only 8 different gluons exist.

In addtion to the quark-gluon interaction, the gluons can also couple to each other. Figure 2.2 illustrates all four fundamental couplings of the strong interaction with the help of Feynman diagrams[2].

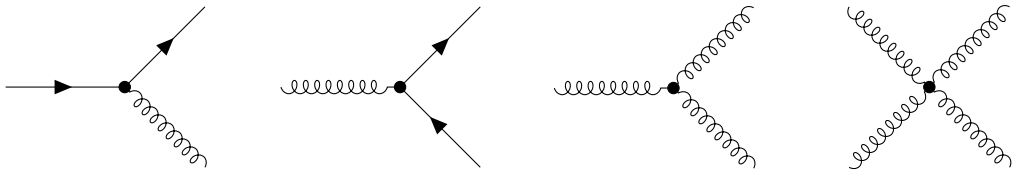



**Figure 2.2:** Fundamental couplings of the strong interaction: (a) gluon radiation; (b) gluon splitting; (c) and (d) gluon self-coupling.

In theoretical physics, the strong interaction between quarks and gluons is described by Quantum Chromodynamics (QCD). The so-called *color confinement* is an important property of QCD saying that free particles must have a color charge of zero. Therefore, isolated quarks and gluons cannot be observed. They can only exist in bound states, called hadrons. According to their spin number, hadrons are divided into two classes: baryons (fermions) formed of three quarks and mesons (bosons) formed of one quark and one antiquark.
As part of our everyday matter, protons (quark composition: $uud$) and neutrons ($udd$) are just two out of many baryons. Depending on their flavor content, baryons are typically named using greek letters $\Lambda$, $\Sigma$, $\Xi$, $\Omega$ and $\Delta$. A complete list of all known and partly still hypothetical particles can be found in Reference [7]. Typical representatives of the mesons are pions (e.g. $\pi^+$ with $u\bar{d}$), kaons (e.g. $K^+$ with $u\bar{s}$), D-mesons (e.g. $D^+$ with $\bar{d}c$) and B-mesons (e.g. $B^+$ with $u\bar{b}$). With $\tau_t \approx 5\cdot 10^{-25}\,\mathrm{s}$ the lifetime of a top quark is too short for the process of clustering into hadrons, called hadronisation, which starts approximately after $10^{-23}\,\mathrm{s}$.
Due to the color confinement all hadrons have to have a color charge of zero. Therefore, quarks inside baryons have to carry the color charges $r$, $g$ and $b$, while for mesons there are three different configurations ($r\bar{r}$, $g\bar{g}$, $b\bar{b}$).

[2] General Remark: In all Feynman diagrams in this thesis, the positive time axis always runs from left to right. The used symbolism complies with the usual standard (see Sec. 4.4 in Ref. [3]).

According to the theory of strong interaction, quarks forming hadrons (called *valence quarks*) are also interacting with virtual quark-antiquark pairs (called *sea quarks*) and virtual gluons. These virtual particles contribute to the mass, momentum and spin of the hadrons.

### 2.1.2 Leptons and the Unification of Electromagnetic and Weak Interaction

Next to quarks, there is another group of six particles plus their corresponding antiparticles, called leptons: three electrically charged leptons ($e^-$, $\mu^-$, $\tau^-$) and three uncharged leptons ($\nu_e$, $\nu_\mu$, $\nu_\tau$). Quarks and charged leptons interact via both electromagnetic and weak forces, while neutrinos can only interact weakly. Since none of the leptons carry a color charge, they cannot interact strongly.

#### Electromagnetic Interaction

Photons, typically represented by the greek letter $\gamma$, are the mediators of the electromagnetic interaction. Like gluons, photons are spin-1 particles (bosons), do not carry an electromagnetic charge and are massless. The electromagnetic interaction can be illustrated by the exchange of a single photon between two electrically charged particles. Figure 2.3 gives two examples. A theoretical description of the corresponding phenomena is provided by Quantum Electrodynamics (QED).

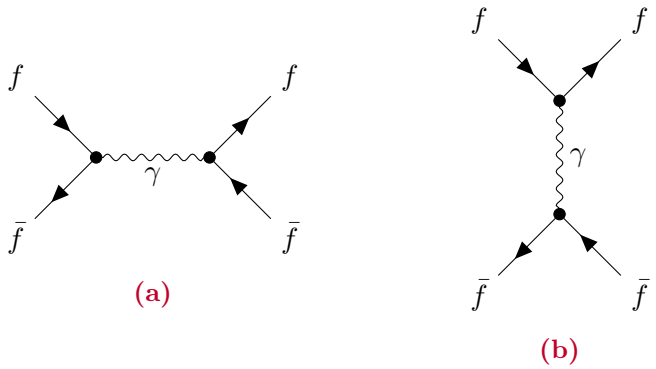



**Figure 2.3:** Feynman diagrams of electromagnetic interactions: (a) fermion-antifermion annihilation; (b) fermion-antifermion scattering.

The resulting force acting on a particle with charge $Q$ and velocity $\vec{v}$ in an electromagnetic field is called *Lorentz force*:

$$\vec{F} = \vec{F}_L = \vec{F}_E + \vec{F}_B = Q \cdot \left(\vec{E} + \vec{v} \times \vec{B}\right). \tag{2.3}$$

By using electric fields $\vec{E}$, charged particles can be accelerated to high speeds. With the help of magnetic fields $\vec{B}$ they can be directed onto a circular path. A combination of both type of fields is required to operate particle accelerators.

#### Weak Interaction

$W$ and $Z$ bosons are the mediators of the weak interaction. Both are spin-1 particles. The $W$ boson is either positively or negatively charged ($W^\pm$), while the Z boson has a charge

of zero ($Z^0$). At around 80 GeV ($W^\pm$) and 91 GeV ($Z^0$), the masses of these bosons are far greater than the masses of leptons and most quarks. Basically, one differentiates between two categories of weak interaction: Processes in which $W^\pm$ bosons are involved (*charged current*) and processes in which a $Z^0$ boson is involved (*neutral current*).
Within neutral current interactions, the charge and flavor of the interacting particles remain unaffected. Exchanging a $Z^0$ boson only transfers momentum, spin and energy. Two examples of neutral current interactions can be constructed by simply replacing each photon in Figure 2.3 with a $Z^0$ boson.
In case of charged current interactions, the flavor of quarks and leptons involved changes as well as the charge by $\pm 1\,e$. One differentiates again into three categories:

- **Leptonic processes** in which $W^\pm$ bosons couple only to leptons. The elementary equation is:
$$l + \bar{\nu}_l \longleftrightarrow l' + \bar{\nu}_{l'} \quad \text{with} \quad l = \{e,\, \mu,\, \tau\}\,. \tag{2.4}$$
As an example, Figure 2.4 (a) illustrates the leptonic decay of a tau lepton.

- **Semi-leptonic processes** in which $W^\pm$ bosons couple to leptons and quarks. The elementary equation is:
$$q_1 + \bar{q}_2 \longleftrightarrow l + \bar{\nu}_l \quad \text{with} \quad q = \{u,\, d,\, c,\, s,\, t,\, b\}\,. \tag{2.5}$$
A famous example is the beta decay shown in Figure 2.4 (b). Mechanisms such as the inverse beta decay also belong to the semi-leptonic processes.

- **Non-leptonic processes** take place without the participation of leptons. The elementary equation is:
$$q_1 + \bar{q}_2 \longleftrightarrow q_3 + \bar{q}_4. \tag{2.6}$$
Due to the charge conservation, only those quarks can be combined whose total charge is $\pm 1\,e$. Examples are hadronic decays of various baryons and mesons. One is given in Figure 2.4 (c) drawing the $K^+$ decaying into pions.

Like the leptons, the quarks are grouped into three generations according to their charge and mass (see also Fig. 2.1):

$$\begin{pmatrix} u \\ d \end{pmatrix},\ \begin{pmatrix} c \\ s \end{pmatrix} \text{ and } \begin{pmatrix} t \\ b \end{pmatrix}. \tag{2.7}$$

In addition to the transitions within a generation, one also sees transitions from one generation to another. Thus, the "partner" of the flavor eigenstate $|u\rangle$ is not $|d\rangle$, but a linear combination of $|d\rangle$, $|s\rangle$ and $|b\rangle$, hereinafter referred to as $|d'\rangle$. The flavor eigenstates $|s'\rangle$ and $|b'\rangle$ can be represented accordingly. In summary, these relations can be brought together using a $3 \times 3$ matrix, the so-called Cabibbo-Kobayashi-Maskawa matrix (CKM matrix):

$$\begin{pmatrix} |d'\rangle \\ |s'\rangle \\ |b'\rangle \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \cdot \begin{pmatrix} |d\rangle \\ |s\rangle \\ |b\rangle \end{pmatrix}. \tag{2.8}$$

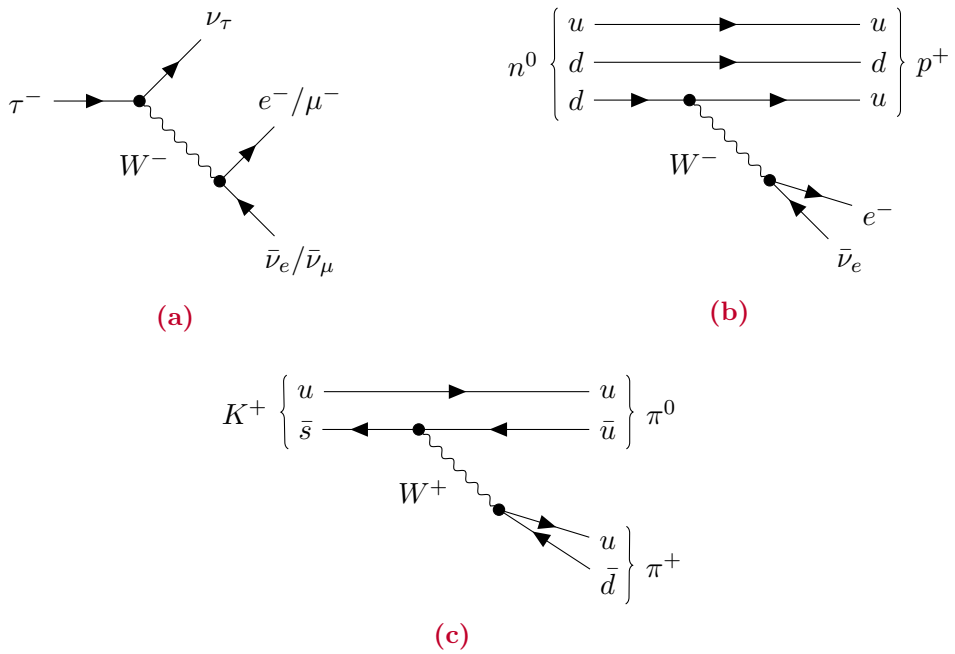



**Figure 2.4:** Feynman diagrams to illustrate different categories of charged current interactions: (a) $\tau$ lepton decay (leptonic process); (b) radioactive $\beta$ decay (semi-leptonic process); (c) $K^+$ decaying into pions (non-leptonic process).

The transition probability from one quark $q_i$ to another quark $q_j$ is proportional to $|V_{ij}|^2$. The corresponding values of the CKM matrix can be determined experimentally from various decays. According to Reference [7], it is approximately:

$$(\,|V_{ij}|\,) = \begin{pmatrix} 0.974 & 0.227 & 0.004 \\ 0.226 & 0.973 & 0.041 \\ 0.009 & 0.040 & 0.999 \end{pmatrix}. \tag{2.9}$$

The values on the diagonal correspond to quark transitions within a generation and are close to 1. Transitions from one generation to another are suppressed by one to two orders of magnitude.

### Electroweak Unification

In 1967, Sheldon L. Glashow, Abdus Salam and Steven Weinberg succeeded in combining QED with the weak interaction into a unified theory of electroweak interaction. In the formalism developed by the three physicists, a new quantum number, the weak isospin T, is introduced (similar to the strong isospin $I$ in QCD)[3]. Each generation of left-handed[4] quarks and leptons thus forms an isospin doublet as it is shown in Table 2.2. The fermions within such a doublet can be interconverted by emission or absorption of $W$ bosons.

[3] In QCD, the strong isospin $I$ is a flavor quantum number that describes an internal symmetry and is used to classify hadrons.

[4] The chirality of a particle is used to define a handedness. In case of massless particles, chirality and helicity are the same. The helicity of a particle is right-handed [left-handed] if spin and momentum are pointing in the same [opposite] direction.

| Fermions | Multiplets | | | $T$ | $T_3$ | Q |
|---|---|---|---|---|---|---|
| **Leptons** | $\begin{pmatrix} \nu_e \\ e \end{pmatrix}_L$ | $\begin{pmatrix} \nu_\mu \\ \mu \end{pmatrix}_L$ | $\begin{pmatrix} \nu_\tau \\ \tau \end{pmatrix}_L$ | $1/2$<br>$1/2$ | $+1/2$<br>$-1/2$ | $0$<br>$-1$ |
| | $e_R$ | $\mu_R$ | $\tau_R$ | $0$ | $0$ | $-1$ |
| **Quarks** | $\begin{pmatrix} u \\ d' \end{pmatrix}_L$ | $\begin{pmatrix} c \\ s' \end{pmatrix}_L$ | $\begin{pmatrix} t \\ b' \end{pmatrix}_L$ | $1/2$<br>$1/2$ | $+1/2$<br>$-1/2$ | $+2/3$<br>$-1/3$ |
| | $u_R$<br>$d_R$ | $c_R$<br>$s_R$ | $t_R$<br>$b_R$ | $0$<br>$0$ | $0$<br>$0$ | $+2/3$<br>$-1/3$ |

**Table 2.2:** Multiplets of the electroweak interaction; $T$: weak isospin; $T_3$: third component of $T$; $Q$: charge. The sign of $T_3$ is defined so that the difference $Q - T_3$ within a doublet is constant. For right-handed antifermions, the sign of $T_3$ and $Q$ changes.

Right-handed fermions and left-handed antifermions do not couple to $W$ bosons. They are described as isospin singlets ($T = 0$ and $T_3 = 0$). Right-handed neutrinos and left-handed antineutrinos ($T_3 = 0$ and $Q = 0$) do not participate in the electroweak interaction.

If one demands the quantum number $T_3$ to be conserved within charged current processes, then $T_3(W^+) = +1$ and $T_3(W^-) = -1$ must apply. In addition, a third state $W^0$ with $T = 1$ and $T_3 = 0$ should exist. $W^\pm$ and $W^0$ form a triplet of the weak isospin. The $W^0$ cannot be identified with the $Z^0$, since the coupling strength of the $Z^0$ also depends on the electrical charge. Therefore, one postulates another state $B^0$ with $T = 0$ and $T_3 = 0$ (singlet of the weak isospin). $B^0$ and $W^0$ couple to fermions without changing their weak isospin and thus their flavor.
In the course of electroweak unification, the bosons $\gamma$ and $Z^0$ are represented as mutually orthogonal linear combinations of $B^0$ and $W^0$:

$$\begin{pmatrix} |\gamma\rangle \\ |Z^0\rangle \end{pmatrix} = \begin{pmatrix} \cos\theta_W & \sin\theta_W \\ -\sin\theta_W & \cos\theta_W \end{pmatrix} \cdot \begin{pmatrix} |B^0\rangle \\ |W^0\rangle \end{pmatrix}. \tag{2.10}$$

The angle $\theta_W$ is known as Weinberg angle or weak mixing angle. Its value can only be determined experimentally.

### Higgs Mechanism

The theory of electroweak unification is based on four massless gauge bosons. However, only the photon has a rest mass of zero. Due to the introduction of the massive bosons $W^\pm$ and $Z^0$, the gauge invariance[5] was violated. A solution to this problem provides the so-called Higgs mechanism, which allows *spontaneous symmetry breaking* in order to save the gauge invariance.
According to this theory, the mass of elementary particles arises only through the interaction with a scalar background field, namely the Higgs field. Mathematically speaking, the Higgs field is a doublet of complex scalar fields. Thus, there are four components,

[5]Gauge invariance: The invariance to a (gauge) transformation (e.g. rotation) of fields involved. From the resulting symmetries, conserved quantities can be formulated.

three of which are "absorbed" by the electroweak gauge bosons to give them mass. The remaining component either manifests as a Higgs particle, or may couple separately to fermions, causing these to acquire mass as well. The existence of the Higgs particle is a confirmation of the theory of the Higgs mechanism and all theories based on it.

The Standard Model Higgs boson is a massive spin-0 particle (scalar boson) and carries neither electric nor color charge. The coupling strength between the Higgs boson and fermions is proportional to their mass, while for gauge bosons it is proportional to their square mass.

### Higgs Production and Decay

The Standard Model predicts that the Higgs bosons could be produced in a number of ways. At hadron colliders, the most dominant process for Higgs boson production is the gluon fusion, where two gluons combine to form a loop of virtual quarks (Fig. 2.5 (a)).
If a fermion collides with an antifermion, the two can merge to form a virtual $W$ or $Z$ boson, which can then emit a Higgs boson (Fig. 2.5 (b)). This process, also known as associated production, is the most dominant at lepton colliders.

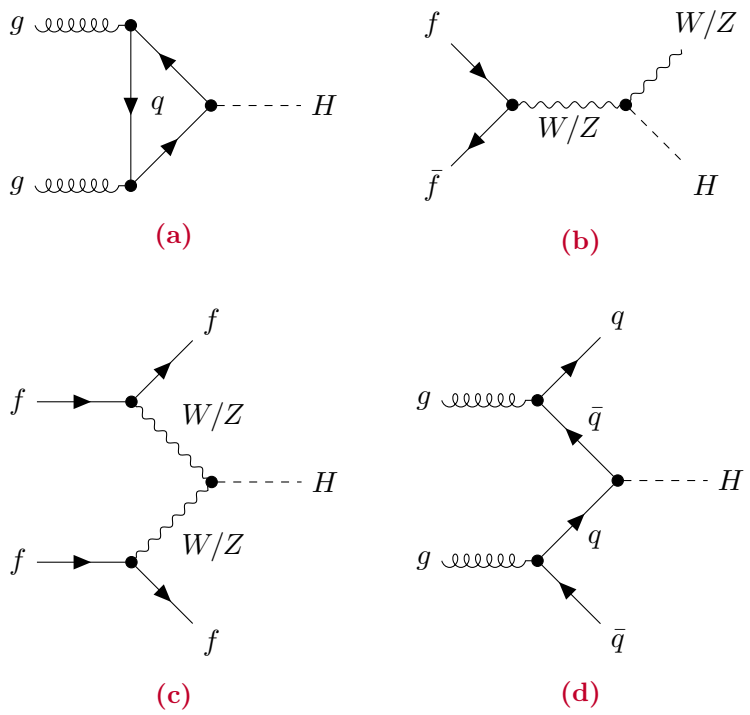



**Figure 2.5:** Feynman diagrams for Higgs boson production: (a) gluon fusion; (b) Higgs Strahlung (associated production); (c) vector boson fusion; (d) quark fusion.

Another important process is the vector boson fusion, where two (anti-) fermions collide and exchange a virtual $W$ or $Z$ boson, which form a Higgs boson. Note that the incoming particles do not have to have the same flavor.
The quark fusion is the last commonly considered process and by far the least likely. It requires two colliding gluons, which each decay into a heavy quark-antiquark pair. A quark and an antiquark from each pair can then fuse to form a Higgs particle.

For a Higgs boson with a mass of 125 GeV the Standard Model predicts a mean life time of about $1.6 \cdot 10^{-22}$ s [8], which makes it impossible to detect it directly. Because it interacts with all the massive (elementary) particles of the Standard Model, the Higgs boson has many different decay processes. As illustrated in Figure 2.6, the Higgs boson either decays into a fermion-antifermion pair or splits into a pair of gauge bosons.

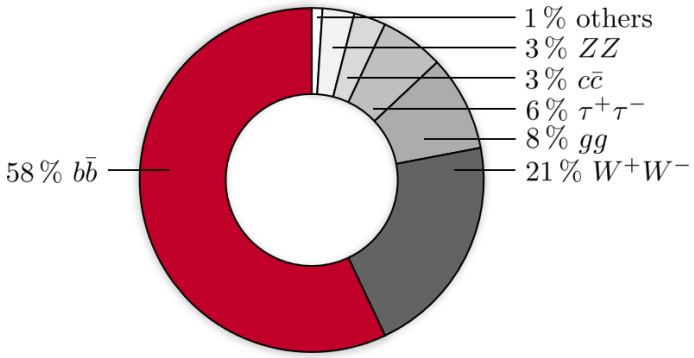



**Figure 2.6:** Branching ratios for a Standard Model Higgs boson with a mass of 125 GeV. Numbers are taken from Reference [8].

The most common decay of a Higgs boson with a mass of 125 GeV is into a bottom-antibottom quark pair. The decay into massless gauge bosons is also possible, but requires an intermediate loop of virtual heavy quarks or massive gauge bosons. The Standard Model Higgs boson decays invisibly only through the $H \to ZZ \to \nu\bar{\nu}\nu\bar{\nu}$ process, which has a branching fraction of about $10^{-3}$ [7, 8].

## 2.2 Limits of the Standard Model

Overall, the Standard Model and its assumptions have been confirmed by a large number of experiments. However, it is incomplete and there are still many unresolved questions:

Is it possible to unify the strong and electroweak interaction, as it was done with the electromagnetic and weak interaction? The gravitational force and its hypothesized force carrier, the graviton, are not even described by the Standard Model. Is it possible to include gravitation? Compared to the strong, electromagnetic and weak interaction, fortunately, the gravitational interaction can be neglected in many experiments of high-energy physics to a good approximation (as shown in Tab. 2.1).
The Standard Model contains at least 18 free parameters, which so far have to be determined experimentally. Is it possible to derive these parameters from other theories? Why are there just three generations of fundamental fermions? Why do the fundamental interactions have such different coupling strengths? Is the found Higgs boson actually the Standard Model Higgs boson? Are there any more Higgs bosons?
Furthermore, there are effects and observations that cannot be explained by the Standard Model, such as the experimentally confirmed, nonzero mass of neutrinos. What is the origin of dark matter and dark energy, which dominate the evolution and structure of the cosmos? Why is there an asymmetry between matter and antimatter in the observable universe?

There are a variety of alternative models, simply extending the established Standard Model without changing its base. Best known approaches are attempts to unify the strong, electromagnetic and weak interaction into one so-called *Grand Unified Theory* (GUT). Such models often include *supersymmetry* (SUSY), a symmetry between fermions and bosons. These theories postulate for each Standard Model particle an accociated particle, called superpartner, that differs from the original particle in its spin. So far, there is no experimental evidence for the existence of supersymmetric particles.

CHAPTER 3

# Accelerator and Detector

*Now what is science? ...*
*it is before all a classification,*
*a manner of bringing together facts*
*which appearances separate,*
*though they are bound together*
*by some natural and hidden kinship.*
*Science, in other words,*
*is a system of relations.*

( Jules H. Poincaré )

This chapter is dedicated to the experimental background of this thesis. After a short presentation of the CERN accelerator complex [9], a more detailed description of the ATLAS experiment [10] follows. The focus is on the calorimeter system and the associated trigger. The last two sections are about future upgrades and ATLAS simulations.
Unless otherwise stated, the information on the ATLAS experiment is based on the ATLAS detector paper [11].

## 3.1 CERN and its Large Hadron Collider

The European Organization for Nuclear Research, known as CERN, is based at Meyrin in the canton of Geneva in Switzerland. Essentially, CERN provides particle accelerators and a large computing facility needed for high-energy physics research.
At CERN, the structure of matter and the fundamental interactions between elementary particles are explored. Thus, it is about the fundamental question of what the universe is made of and how it works.
Using large particle accelerators, particles are accelerated to nearly the speed of light and collided at certain interaction points. Particle detectors are constructed around these interaction points to measure various properties of the collided and newly formed particles.

### Accelerator Complex

Figure 3.1 gives an overview of the accelerator complex at CERN. By this time, the Large Hadron Collider (LHC) is the largest and most powerful machine accelerating and colliding protons and heavy ions up to a center of mass energy of 13 TeV.

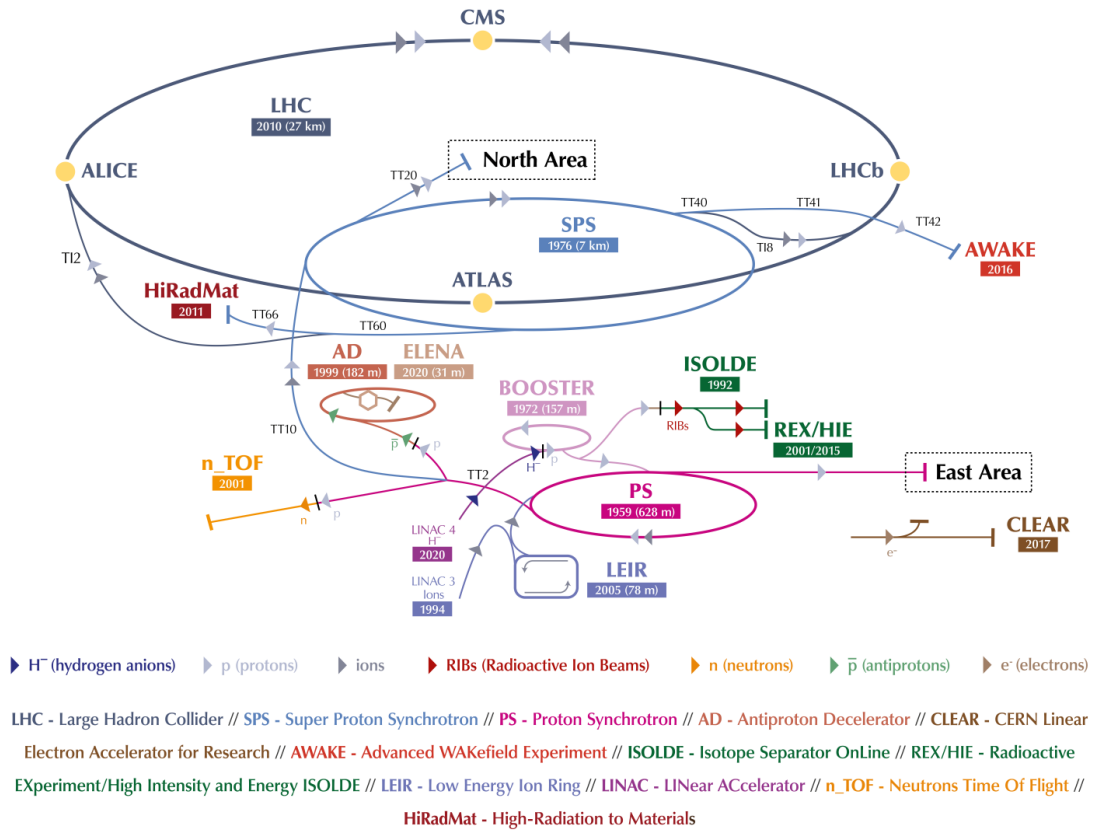



**Figure 3.1:** CERN accelerator complex [12].

Following the path of the protons, there are four accelarators in the accelerator chain prior to injection into the LHC. They keep boosting the particles and divide them into bunches. The proton source is a bottle of hydrogen gas. The hydrogen is passed through an electric field to remove its electrons, leaving only protons to enter the Linear Accelerator 2 (LINAC 2)[1]. When leaving, the protons have reached an energy of 50 MeV. They are fed into the Proton Synchrotron Booster (PSB), which is made up of four superimposed synchrotron rings and accelerates the protons to 1.4 GeV. The Proton Synchrotron (PS) accelerates either protons delivered by the PSB or heavy ions from the Low Energy Ion Ring (LEIR). The PS has a circumference of 628 m and operates at up to 25 GeV. The Super Proton Synchrotron (SPS) takes particles from the PS and accelerates them to provide beams for the LHC and other experiments. The SPS measures nearly 7 km in circumference and boosts the protons further to an energy of 450 GeV.

Most of the accelerators in the chain have their own experimental halls hosting a variety of experiments driven at lower energies. For example, different kinds of radioactive nuclei are produced at the ISOLDE (Isotope Separator On Line DEvice) facility by shooting protons delivered by the PSB at a fixed target. These nuclei are ionized, accelerated, separated by magnets in terms of their masses, and provided to several experiments in atomic and nuclear physics as well as materials and life science.

[1]LINAC 2 was replaced by the Linear Accelerator 4 (LINAC 4) in 2020.

In the following, only the LHC and its experiments are considered further. Information on the other accelerators, facilities and experiments can be found in Reference [9].

### Large Hadron Collider

The LHC has a circumference of almost 27 km and consists of eight circular arcs and eight straight sections. It is made up of two storage rings, such that the protons can travel in opposite directions. Both tubes are kept at ultrahigh vacuum.
The particles are guided on their circular path by a strong magnetic field maintained by superconducting dipole magnets made of niobium and titanium. By means of currents of almost 12 kA, a magnetic field of about 8 T is generated. To keep the particles focused and thus to increase the collision rate, superconducting quadrupole magnets are employed. Liquid helium is used to cool down the magnets to a temperature of about $-271\,^{\circ}\mathrm{C}$. Cavity Resonators are placed along the straight sections accelerating the particles using an electric field of roughly $5\,\mathrm{MV/m}$ at a frequency of about 400 MHz. The strength of the magnetic field in the dipoles and the frequency of the electric field in the resonators are constantly adapted to the increasing energy of the particles.
There are at maximum 2808 bunches of protons per circulation direction. A single bunch contains over 100 billion protons, each with an energy of up to 6.5 TeV.

### LHC Experiments

Research objectives at the LHC are the generation and study of known and still unknown elementary particles and states of matter. The starting point is the verification of the Standard Model of particle physics. Moreover, the LHC is intended to give answers to unresolved questions by searching for physics beyond the Standard Model.
The beams inside the LHC are made to collide at four locations around the accelerator ring (yellow dots in Fig. 3.1). The two largest particle detectors, build around these interaction points, are A Toroidal LHC ApparatuS (ATLAS) and Compact Muon Solenoid (CMS). Both are general-purpose detectors, having the same scientific goals and investigating a wide range of physics, from studying the Higgs boson to searching for extra dimensions and particles that could make up dark matter. However, ATLAS and CMS use different technical solutions and a different magnet-system design. In this way, experimental results can be checked vice versa.
The other two particle detectors build around the interaction points are A Large Ion Collider Experiment (ALICE) and Large Hadron Collider beauty (LHCb). ALICE is a heavy-ion detector, designed to study the physics of strongly interacting matter at extreme energy densities (quark-gluon plasma). The LHCb experiment was started to study the nature of bottom quarks (also known as beauty quarks) and to investigate the difference between matter and antimatter.

## 3.2 The ATLAS Detector

Particles collide at the center of the ATLAS detector. The resulting particles from a collision fly out from the collision point in all directions. To record properties such as momentum and energy, six different detecting subsystems are stacked in layers around the collision point (see Fig. 3.2). Hence, ATLAS has the shape of a cylinder with 46 m in

length, 25 m in diameter and 7 kt in weight. It is located 100 m below ground, close to the main CERN site.

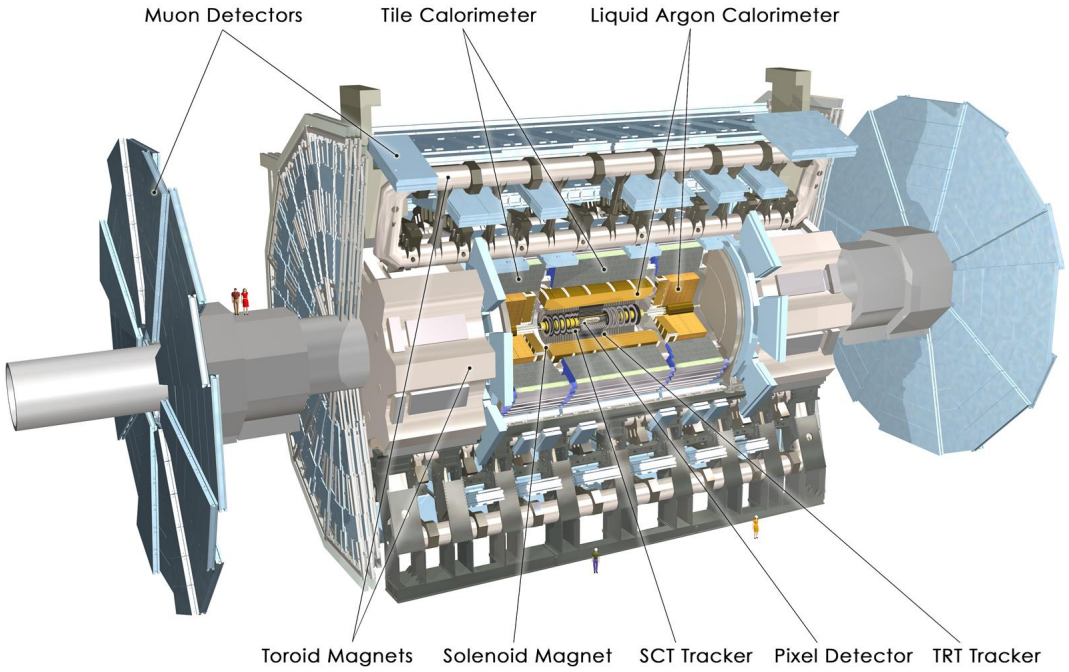



**Figure 3.2:** Cut-away view of the ATLAS detector [10].

The Inner Detector is the first element through which particles pass. It is composed of three independent but complementary sub-detectors:

- Pixel Detector [first layer],
- Semi-Conductor Tracker (SCT) [second layer],
- Transition Radiation Tracker (TRT) [third layer].

A solenoid magnet is arranged around the entire Inner Detector to generate a magnetic field forcing charged particles to bend. This way, the basic function of the Inner Detector is to track charged particles, revealing detailed information about their type and momentum.
The calorimeter system is built around the solenoid magnet. It includes the Liquid Argon (LAr) Calorimeter and the Tile Calorimeter. The LAr Calorimeter houses the entire electromagnetic (EM) calorimeter and part of the hadronic (HAD) calorimeter, while the Tile Calorimeter houses the remaining part. Particles which interact electromagnetically (charged particles and photons) and via the strong force (primarily hadrons) are stopped inside the calorimeters by absorbing their energy.
The last and largest element of ATLAS are the Muon Detectors used to accurately measure the momentum of muons. Because of their large mass and due to the fact that they do not interact strongly, muons can pass the calorimeters almost undisturbed. The magnetic field is generated by several toroid magnets distributed over the entire muon spectrometer.

### 3.2.1 Coordinate System

The coordinate system used by the ATLAS experiment is a three-dimensional, right-handed coordinate system originating at the interaction point. The three axes are labeled $x$, $y$ and $z$. The beam axis runs in the $z$ (longitudinal) direction, while the $x$-$y$ (transverse) plane is oriented perpendicular to the beam axis. The positive $x$ axis points from the interaction point to the center of the LHC, while the positive $y$ axis points vertically upwards. The A-side is defined as the part of ATLAS built around the positive $z$ axis, while the C-side represents the other part.
For the purpose of orientation, spherical coordinates are used by default:

$$\vec{r} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} r \cdot \sin\theta \cdot \cos\varphi \\ r \cdot \sin\theta \cdot \sin\varphi \\ r \cdot \cos\theta \end{pmatrix}. \tag{3.1}$$

The azimuth angle $\varphi$ defines the angle between the positive $x$ axis and the projection of the direction vector $\vec{r}$ onto the transverse plane, denoted as $\vec{r}_{xy}$ or $\vec{r}_{\mathrm{T}}$. The polar angle $\theta$ defines the angle between the positive $z$ axis and the direction vector $\vec{r}$. Illustrations are given in Figure 3.3.

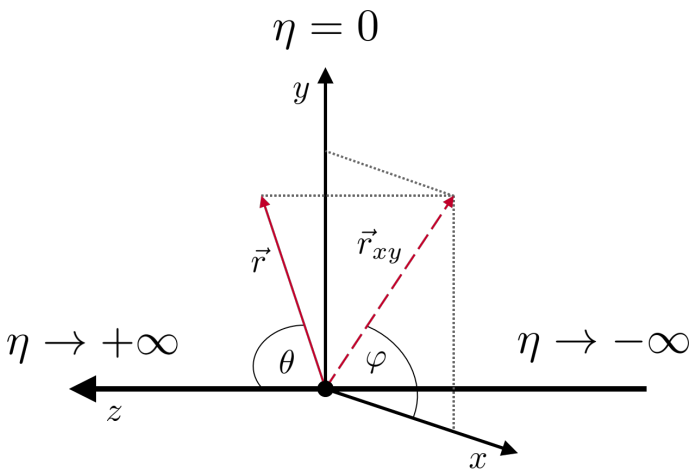



**Figure 3.3:** ATLAS coordinate system.

The inverse transformation takes place according to the following parameterization:

$$r = \sqrt{x^2 + y^2 + z^2}, \tag{3.2}$$

$$\theta = \arccos\left(\frac{z}{r}\right) = \arcsin\left(\frac{\sqrt{x^2 + y^2}}{r}\right), \tag{3.3}$$

$$\varphi = \begin{cases} \arctan\left(\frac{y}{x}\right) & \text{if } x > 0 \\ +\frac{\pi}{2} & \text{if } x = 0 \wedge y > 0 \\ -\frac{\pi}{2} & \text{if } x = 0 \wedge y < 0 \\ \arctan\left(\frac{y}{x}\right) + \pi & \text{if } x < 0 \wedge y \geq 0 \\ \arctan\left(\frac{y}{x}\right) - \pi & \text{if } x < 0 \wedge y < 0 \end{cases}. \tag{3.4}$$

In hadron collider physics, so as in the ATLAS experiment, the rapidity $Y$ is preferred over the polar angle $\theta$. It is defined as:

$$Y = \operatorname{artanh} |\vec{v}| = \operatorname{artanh} \left( \frac{|\vec{p}|}{E} \right) = \frac{1}{2} \cdot \ln \left( \frac{E + |\vec{p}|}{E - |\vec{p}|} \right). \tag{3.5}$$

Thus, the rapidity can be calculated by measuring the momentum $|\vec{p}|$ and the energy $E$. It is also a common choice to measure the relativistic velocity $|\vec{v}|$.
Particle production is constant in $Y$ and differences in $Y$ are Lorentz invariant under boosts along the beam axis. The latter is an important fact for hadron collider physics, where the colliding quarks and gluons carry different momentum fractions along the beam axis, i.e. there is no fixed rest frame.
By replacing $|\vec{p}|$ with the component of the momentum along the beam axis, Equation 3.5 turns into a modified definition of the rapidity relative to the beam axis:

$$Y^* = \frac{1}{2} \cdot \ln \left( \frac{E + |p_z|}{E - |p_z|} \right). \tag{3.6}$$

The mass of highly boosted particles can be neglected to a good approximation, i.e. momentum and energy become equivalent. In this case, one no longer speaks of the rapidity, but of the pseudorapidity $\eta$:

$$\eta = \frac{1}{2} \cdot \ln \left( \frac{|\vec{p}| + |p_z|}{|\vec{p}| - |p_z|} \right) = -\ln \left[ \tan \left( \frac{\theta}{2} \right) \right]. \tag{3.7}$$

Inversely,

$$\theta = 2 \cdot \arctan \left( \exp^{-\eta} \right). \tag{3.8}$$

Since the pseudorapidity depends only on the momentum, it is experimentally easier to determine and therefore more common.

The initial momentum of the colliding protons along the beam axis is not known, as their constituents carry an unknown momentum fraction. However, the initial momentum of the protons in the transverse plane is zero. Particles (most notably neutrinos) that do not interact with the detector material cause a net momentum in the transverse direction, referred to as missing transverse momentum:

$$\vec{p}_{\mathrm{T}}^{\,\mathrm{miss}} = \sum_{i \in \text{invisible}} (\vec{p}_{\mathrm{T}})_i = - \sum_{i \in \text{observable}} (\vec{p}_{\mathrm{T}})_i \,. \tag{3.9}$$

The presence of such non-detectable particles is also a signature in many theories of physics beyond the Standard Model.
Projections onto the transverse plane are also defined for scalar quantities, e.g. transverse energy $E_{\mathrm{T}}$:

$$\begin{pmatrix} E_x \\ E_y \\ E_z \end{pmatrix} = \begin{pmatrix} E \cdot \sin\theta \cdot \cos\varphi \\ E \cdot \sin\theta \cdot \sin\varphi \\ E \cdot \cos\theta \end{pmatrix} \rightarrow \; E_{\mathrm{T}} = \sqrt{E_x^2 + E_y^2} = E \cdot \sin\theta. \tag{3.10}$$

By combining Equations 3.8 and 3.10, it is possible to obtain Cartesian energies using the measured quantities $E_{\mathrm{T}}$, $\varphi$, $\eta$:

$$\begin{pmatrix} E_x \\ E_y \\ E_z \end{pmatrix} = \begin{pmatrix} E_{\mathrm{T}} \cdot \cos\varphi \\ E_{\mathrm{T}} \cdot \sin\varphi \\ E_{\mathrm{T}} \cdot \sinh\eta \end{pmatrix} \rightarrow \; E = E_{\mathrm{T}} \cdot \cosh\eta. \tag{3.11}$$

In addition to the absolute position of individual particles, it is often useful to define a measure of angular separation between particle $i$ and particle $j$:

$$\Delta R_{ij} = \sqrt{(\Delta\eta_{ij})^2 + (\Delta\varphi_{ij})^2} \tag{3.12}$$

with

$$\Delta\eta_{ij} = |\eta_i - \eta_j| \tag{3.13}$$

and

$$\Delta\varphi_{ij} \;=\; \begin{cases} |\varphi_i - \varphi_j| & \text{if } |\varphi_i - \varphi_j| \leq \pi \\ 2 \cdot \pi - |\varphi_i - \varphi_j| & \text{if } |\varphi_i - \varphi_j| > \pi \end{cases} . \tag{3.14}$$

In any case, differences in $\varphi$ are invariant under Lorentz boosts along the beam axis, while for $\Delta\eta$, and thus for $\Delta R$, this is only true if the involved particles are considered to be massless.

### 3.2.2 Magnet System

ATLAS is forward-backward symmetric with respect to the interaction point. Its individual subsystems, already introduced at the beginning of this section, are described in more detail here and in the subsections below. It starts with the magnet configuration, whose planning has driven the design concept of the rest of the detector.
The whole magnetic system is 22 m in diameter and 26 m in length. The configuration is made up of a thin superconducting solenoid surrounding the Inner Detector and three large superconducting air-core toroids (one barrel and two end-caps) arranged with an eight-fold azimuthal symmetry around the calorimeters and within the muon system. Figure 3.4 shows the arrangement of the individual magnet components relative to each other.
The solenoid magnet generates a 2 T axial magnetic field for the Inner Detector. Its layout has been optimized with respect to the material thickness (only 5 cm) in order to achieve the desired calorimeter performance. It is 5 t in weight and contains 9 km of superconducting cable, which is kept at operating temperature using liquid helium. The nominal current used to generate the magnetic field reaches almost 8 kA.
The air-core toroids weigh about 1.3 kt and do not contain a magnetic core, leaving enough space for hosting superconducting cables with a total length of 100 km. A nominal current of about 21 kA generates a magnetic field of 4 T. The end-cap toroid system is rotated by 22.5° with respect to the barrel toroid system to optimize the bending power in the resulting radial overlap.

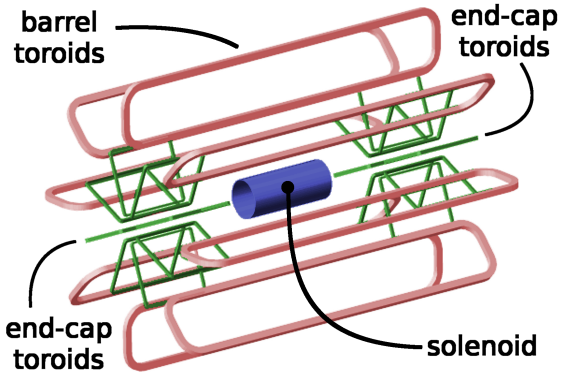



**Figure 3.4:** ATLAS magnet system [13].

### 3.2.3 Inner Detector

The Inner Detector is about 2 m in diameter and 6 m in length. The distance between the beam axis and the beginning of the Inner Detector is only 33 mm. Track and vertex reconstruction are its main tasks. Charged particles are forced to curve in the presence of the magnetic field generated by the solenoid magnet surrounding the Inner Detector. The momentum of such a particle can be derived from the curvature of its trajectory, while the direction of the curvature gives the charge.
The Inner Detector consists of a layer-wise system of precision tracking detectors (pixels and SCT) and straw tubes (TRT). A cut-away view is shown in Figure 3.5.

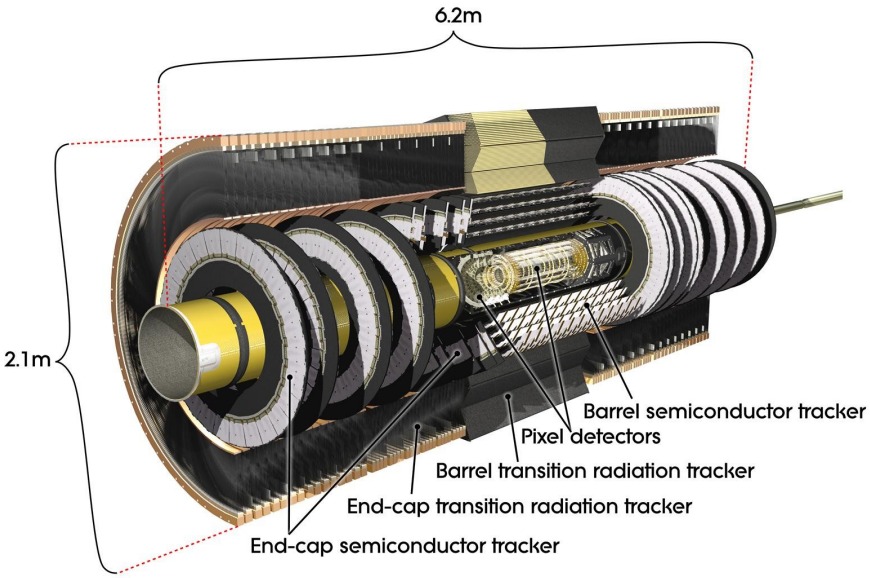
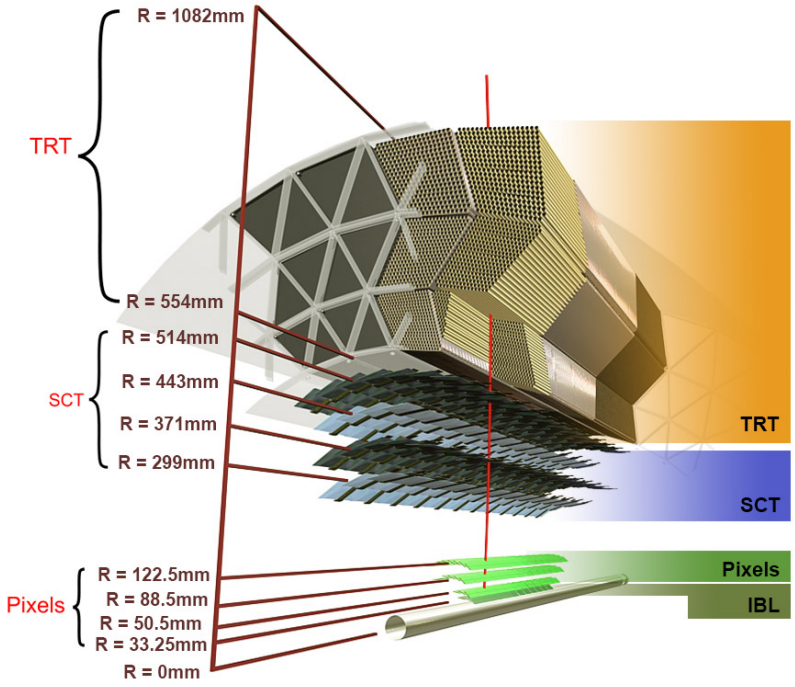
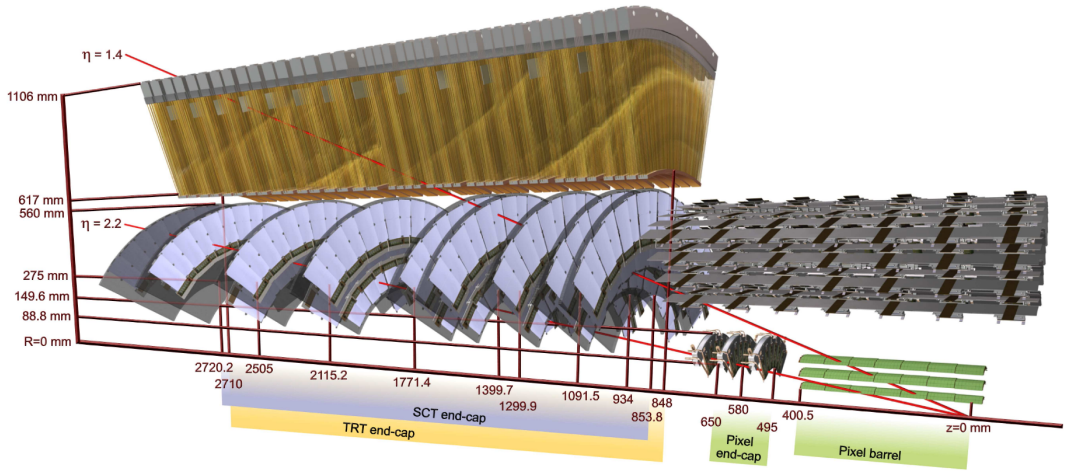



**Figure 3.5:** Cut-away view of the ATLAS Inner Detector [10].

End-cap detectors are placed at each end of the barrel-shaped detectors to provide the most complete $\eta$ coverage. Pixel Detector and SCT cover the region $|\eta| < 2.5$, while the TRT enables track-following up to $|\eta| = 2.0$. Figure 3.6 provides a deeper insight into the texture and spacing of the individual layers in the barrel and end-cap regions.




(a)




(b)

**Figure 3.6:** ATLAS Inner Detector: (a) barrel region (traversed by a charged track of $p_{\mathrm{T}} = 10\,\mathrm{GeV}$ at $\eta = 0.3$) [14]; (b) end-cap region (traversed by two charged tracks of $p_{\mathrm{T}} = 10\,\mathrm{GeV}$ at $\eta = 1.4$ and 2.2) [11].

There are four layers of silicon pixel detectors. In the barrel region, they are arranged on concentric cylinders around the beam axis (see Fig. 3.6 (a)), while in the end-cap regions they are located on disks perpendicular to the beam axis (see Fig. 3.6 (b)). The innermost layer, called Insertable B-Layer (IBL) [14], contains 12 million pixels with a typical size of $50 \times 250\,\mu\mathrm{m}^2$ in $r - \varphi \times z$. In the other three layers, about 90% of the pixels have a nominal size of $50 \times 400\,\mu\mathrm{m}^2$ in $r - \varphi \times z$ with an accuracy of $10\,\mu\mathrm{m}$ in $r - \varphi$ and $115\,\mu\mathrm{m}$

in $z$ (115 $\mu$m in $R$ in the disks), while the remaining ones are slightly larger. Despite the high track density, a total of around 92 million pixels (92 million readout channels) allow high-precision measurements close to the interaction point.
The SCT is made up of silicon micro-strip sensors distributed over four cylindrical barrel layers and 18 planar end-cap disks (nine per side). Both layers and disks use small-angle (40 mrad) stereo strips to measure both coordinates. With a total number of approximately six million readout channels, the intrinsic accuracy is 17 $\mu$m in $r - \varphi$ and 580 $\mu$m in $z$ (580 $\mu$m in $R$ in the disks).
The TRT uses polyimide drift (straw) tubes of 4 mm diameter filled with a $Xe/CO_2/O_2$ gas mixture and a 0.03 mm diameter gold-plated tungsten wire in its center. In the barrel region, the tubes run parallel to the beam axis and thus provide only $r - \varphi$ information with an intrinsic accuracy of 130 $\mu$m per straw. In the end-cap regions, the tubes are arranged radially in wheels. In total, there are about 350,000 readout channels.
Due to the large number of measurements and longer measured track length, the TRT hits at large radii contribute significantly to the momentum measurement. In combination with the pixels and SCT at small radii, a very robust pattern recognition and high precision measurements in $r - \varphi \times z$ are achieved.

### 3.2.4 Calorimeter System

The sampling calorimeters cover the range $|\eta| < 4.9$ with full $\varphi$-symmetry and coverage around the beam axis. A finer granularity of the EM calorimeter over the $\eta$ range of the Inner Detector allows precision measurements of electrons and photons, while the coarser granularity of the rest of the calorimeter is sufficient to also capture large particle showers and to measure missing energy.
A cut-away view of the Calorimeter System is shown in Figure 3.7. Its components are the LAr Calorimeter, which is divided into an EM barrel part and two end-caps, and the Tile Hadronic Calorimeter, which is divided into a central barrel and two extended barrels. Each LAr end-cap contains an EM End-Cap (EMEC), a HAD End-Cap (HEC), which is located behind the EMEC, and a Forward Calorimeter (FCal).
The calorimeter depth had to be chosen carefully in order to provide good containment for EM and HAD showers, thus to minimize energy losses and to limit punch-through into the muon system. For the same reasons, the $\eta$ ranges of the individual calorimeter subsystems overlap slightly.

#### Electromagnetic Barrel and End-Cap Calorimeters

EM barrel ($|\eta| < 1.475$) and EMEC ($1.375 < |\eta| < 3.2$) are lead-LAr detectors with accordion-shaped kapton electrodes and lead absorber plates (about 1 to 2 mm thick; varying in $\eta$). They are segmented in three layers in depth over the region covered by the Inner Detector ($|\eta| < 2.5$). The first layer has the finest segmentation in $\eta$ to achieve an accurate position measurement (see Fig. 3.8). The second layer collects the largest fraction of the energy of an electromagnetic shower, while the third layer collects only its tail. There are only two different layers at higher $\eta$ and in the overlap regions between barrel and EMEC. Furthermore, in the region of $|\eta| < 1.8$, a separate thin LAr layer (Pre-Sampler, 11 mm in depth) made of interleaved cathode and anode electrodes glued between glass-fiber composite plates is used to measure the energy lost upstream of the calorimeters (not shown in Fig. 3.8).

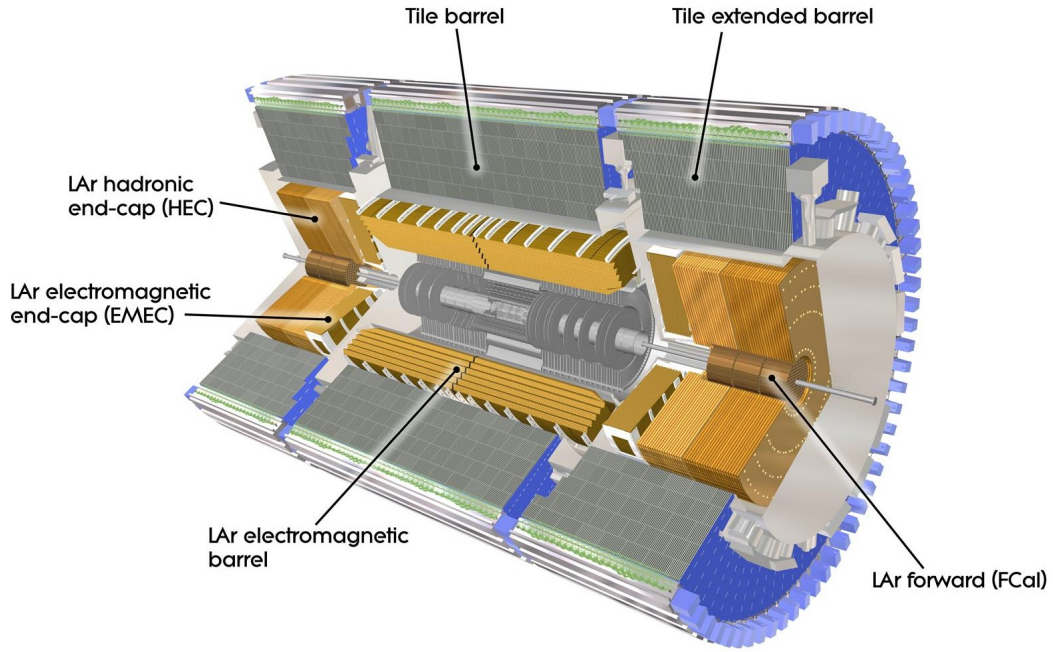



**Figure 3.7:** Cut-away view of the ATLAS Calorimeter System [10].

### Tile and Hadronic End-Cap Calorimeters

The barrel of the Tile Calorimeter covers the region $|\eta| < 1.0$, while the two extended barrels cover $0.8 < |\eta| < 1.7$. Steel is used as the absorber and scintillating tiles as the active material. Each Tile module forms an almost-periodic steel-scintillator structure. The front-end electronics, which also provide analogue sums forming trigger towers for the L1 trigger (discussed in Sec. 3.3), and the readout photomultiplier tubes are highly integrated with the mechanical structure. Wavelength-shifting fibers collect the light produced in the tiles. These fibers are grouped and coupled to the photomultipliers forming three-dimensional readout cells in three layers in depth. The dimensions of these cells are $\Delta\eta \times \Delta\varphi = 0.1 \times 0.1$ in the first two layers and $0.2 \times 0.1$ in the last layer.

The HEC extends the Tile Calorimeter to larger $\eta$ covering the region $1.5 < |\eta| < 3.2$. Close to the interaction point it is built from 25 mm parallel copper plates (50 mm further away), which are interleaved with 8.5 mm LAr gaps used as the active medium. Each LAr gap is divided into four separate drift zones (1.8 mm width), individually supplied with high voltage. The dimensions of the readout cells are $\Delta\eta \times \Delta\varphi = 0.1 \times 0.1$ in the region $|\eta| < 2.5$ and $0.2 \times 0.2$ beyond.

### Forward Calorimeter

The FCal consists of three layers with a total coverage of $3.1 < |\eta| < 4.9$. The first layer (FCal-1) is made of copper, optimized for electromagnetic measurements, while the other two layers (FCal-2 and FCal-3) are made of tungsten, measuring the energy of hadronic interactions. To further reduce backgrounds in the end-cap muon system, a shielding plug made of copper alloy has been mounted behind FCal-3 (see Fig. 3.9).

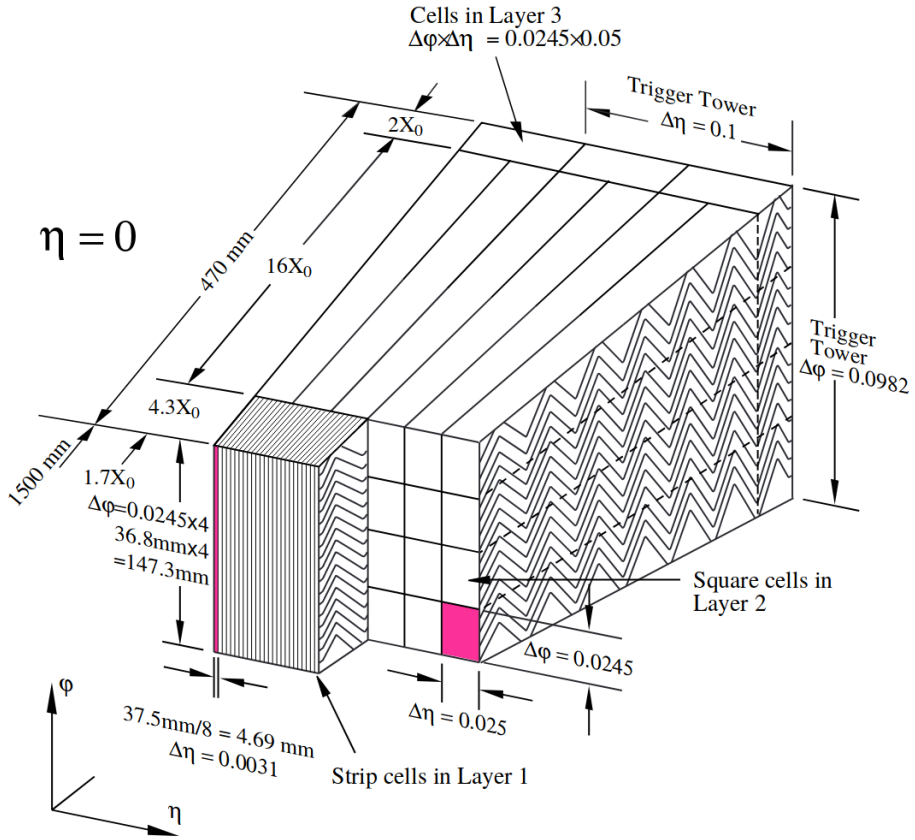



**Figure 3.8:** Sketch of an EM barrel module showing the granularity in $\eta \times \varphi$ of the cells in each of the three layers [11].

Each FCal layer consists of a metal matrix with regularly spaced electrodes. The electrodes consist of concentric rods and tubes parallel to the beam axis. The gap between the rod and the tube is filled with LAr as the sensitive medium. The readout electrodes are hard-wired together in groups of four, six and nine for FCal-1, FCal-2 and FCal-3, respectively, and routed to summing boards. Due to geometric constraints and higher particle fluxes at the inner radius, the signal summings are in general different at the inner and outer radii.

### Pile-up and Pulse Shaping

Next to the hard interaction, soft interactions in inelastic proton-proton collisions happening at the same bunch crossing lead to additional signatures in the detector, usually referred to as *pile-up*. The LAr calorimeter is likely to be affected by the pile-up from other bunch crossings:

The signal in the LAr calorimeter is a triangular pulse with a very fast rise and, due to the relatively long drift time, a long tail. Thus, multiple pulses from subsequent bunch crossings may be overlaid. To minimize the effects of such out-of-time pile-up, *bipolar pulse shaping* is performed. The amplitude, which is related to the energy deposited, and the peak time can then be extracted from multiple sampling points, which also allows negative energy in the cell for the out-of-time background. For more details, see Reference [15].

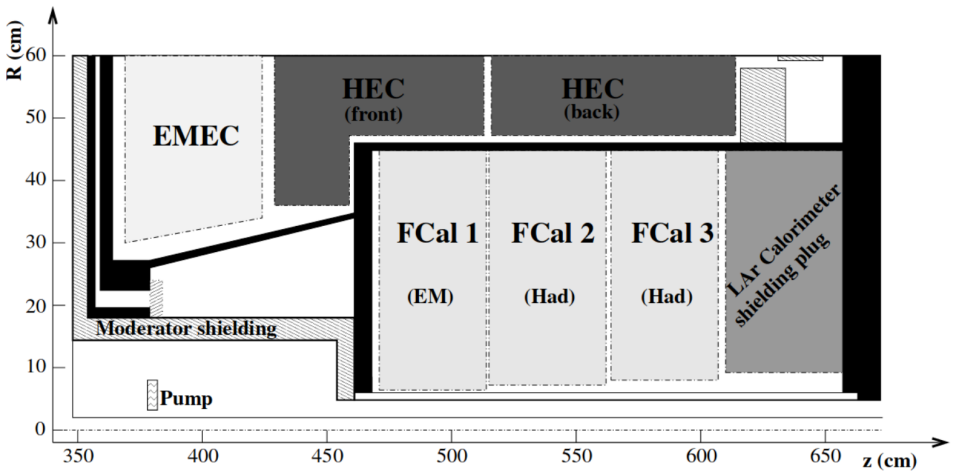



**Figure 3.9:** FCal layers located in the LAr end-cap [11]. The deep black areas represent structural parts.

### Energy Resolution

The relative energy resolution of a calorimeter is usually parameterized as

$$\frac{\sigma_E}{E} = \frac{a}{\sqrt{E}} \oplus \frac{b}{E} \oplus c, \tag{3.15}$$

where $a$ is the stochastic term, which represents statistical shower, sampling and signal quantum fluctuations; $b$ is the noise term, which includes electronic noise and pile-up fluctuations; and $c$ is the constant term, which originates from the calibration of the calorimeter and dominates at high energy. Due to mass resolution requirements for the Higgs discovery, the design values for the stochastic (constant) term were required to be 10% (0.7%) for the EM calorimter, 50% (3%) for the HAD calorimeter and 100% (10%) for the forward calorimeters. The energy resolution has been measured during test beams showing that all these requirements have been met [11].

### Electromagnetic and Hadronic Shower Development

Electrons, positrons and photons induce electromagnetic showers when passing through dense matter, essentially caused by bremsstrahlung (in case of electrons and positrons) and pair production (in case of photons). The shower profile is determined by the ratio

$$\frac{R_M}{X_0} \propto \frac{1}{E_c} \propto Z, \tag{3.16}$$

where the Molière radius $R_M$ defines the lateral shower profile. The longitudinal spread scales with the radiation length $X_0$, which is the mean distance over which an high-energy electron loses $1 - 1/e = 63\%$ of its energy due to bremsstrahlung. $E_c$ represents the critical energy, at which the loss of energy through bremsstrahlung and ionisation becomes equal. About 90% of the energy is deposited within a cylinder with radius $R_M$ around the shower axis [16]. Thus, the choice of the detector material (atomic number $Z$) ultimately determines the required size and granularity of the calorimeter.

As described above, lead and copper have been chosen for the EM layers of the ATLAS calorimeter. It is $R_M = 16.0\,\text{mm}$ (15.7 mm) and $X_0 = 5.6\,\text{mm}$ (14.4 mm) for lead (copper) with $Z = 82$ ($Z = 29$) [7].

Hadrons also form particle showers, commonly known as *jets*, when they pass through dense matter. However, the processes involved are far more diverse than in the electromagnetic case, and therefore much more difficult to describe theoretically. Hadronic cascades are caused by inelastic scattering with nuclei producing secondary particles, emission and evaporation of nucleons and (light) fragments from excited nuclei (spallation; intranuclear cascade; fission) plus electromagnetic decays of short-lived particles like neutral pions or $\eta$-mesons into photons. Such a photon creates an electromagnetic shower that breaks out of the hadronic cascade taking away a portion of its energy. Furthermore, energy is carried away by neutrinos originating from weak decays of e.g. charged pions and kaons. This combination of different processes (hadronic, electromagnetic and weak) lead to strong fluctuations in the shower development and poorer energy resolution compared to a pure electromagnetic shower. The shower profile can be estimated as

$$\frac{\lambda_a}{X_0} \approx 0.37\,Z, \tag{3.17}$$

where the nuclear absorption length $\lambda_a$ is an important scale for the development of a hadronic shower, as it indicates the depth of penetration into a material at which the probability that a particle was not absorbed has dropped to $1/e = 37\%$. Neglecting material dependencies, approximately 95% of the energy can be expected to be in a cylinder with radius $\lambda_a$ [16].
The HAD layers of the ATLAS calorimeter are made of steel and tungsten. It is $\lambda_a = 16.8\,\text{cm}$ ($\lambda_a = 9.9\,\text{cm}$) and $X_0 = 17.6\,\text{mm}$ ($X_0 = 3.5\,\text{mm}$) for iron (tungsten) with $Z = 26$ ($Z = 74$) [7].

### 3.2.5 Muon Spectrometer

The muon spectrometer defines the overall dimensions of the ATLAS detector. Over the range $|\eta| < 2.7$, it is designed to detect and to measure the momenta of charged particles exiting the calorimeters. The magnetic bending is provided by the large barrel toroids for $|\eta| < 1.4$ and by the smaller end-cap toroids for $1.6 < |\eta| < 2.7$, while in the transition region it is a combination of both.
As illustrated in Figure 3.10, the spectrometer is made up of high-precision tracking and separate trigger chambers using four different technologies:

- Monitored Drift Tubes (MDTs),
- Cathode Strip Chambers (CSCs),
- Resistive Plate Chambers (RPCs),
- Thin Gap Chambers (TGCs).

Precision measurement of each track coordinate in the bending ($\eta$) plane is provided by the MDTs covering the range $|\eta| < 2.7$. In the barrel region, the chambers are arranged in three concentric cylindrical layers around the beam axis, located between and on the eight

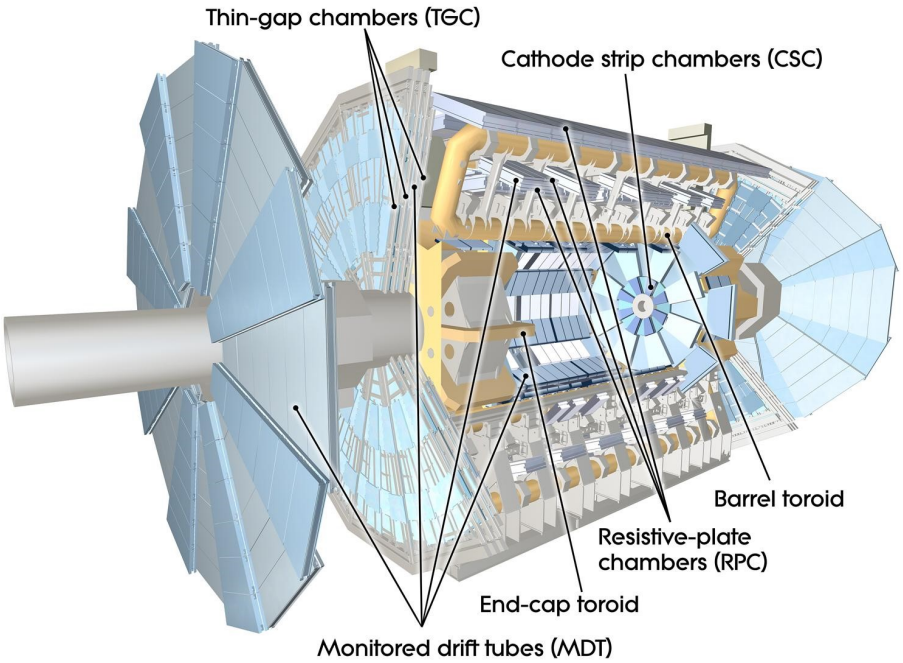



**Figure 3.10:** Cut-away view of the ATLAS muon system [11].

coils of the toroids, while in the end-cap regions they form large wheels perpendicular to the $z$ axis, located in front and behind the end-cap toroids (see Fig. 3.11). These chambers consist of three to eight layers of 30 mm aluminium drift tubes filled with an $Ar/CO_2$ gas mixture. There is a gold-plated tungsten-rhenium wire with a diameter of 50 $\mu$m in the centre of each tube collecting the electrons resulting from ionisation. The achieved average resolution is about 80 $\mu$m per tube (35 $\mu$m per chamber with only three layers).
To deal with the demanding rate and background conditions at large $\eta$, CSCs with higher granularity are used instead of MDTs in the innermost layer over $2.0 < |\eta| < 2.7$. The CSCs, which measure both track coordinates simultaneously, are multiwire proportional chambers with cathode plates segmented into strips. The resolution achieved is 60 $\mu$m per chamber in the bending plane and, due to a coarser cathode segmentation, only 5 mm in the transverse plane.

The trigger chambers, RPCs in the barrel ($|\eta| < 1.05$) and TGCs in the end-cap regions ($1.05 < |\eta| < 2.4$), are arranged in three to four layers covering the range $|\eta| < 2.4$: Two RPC layers sandwich the MDTs of the middle layer, while the third one is located behind the outer MDT layer; three TGC layers are in front and behind the second MDT wheel, while the fourth layer is located in front of the innermost MDT wheel (see Fig. 3.11). Both chamber types provide fast and coarse tracking information, allowing the ATLAS trigger stages to select events based on multiplicity and approximate energy range.
The basic element of an RPC chamber is a set of two parallel electrode-plates (2 mm thick plastic laminate) at a distance of 2 mm. The gap is filled with a gas mixture of $C_2H_2F_4$/Iso-$C_4H_{10}$/$SF_6$, which provides a comfortable plateau for safe avalanche operation.
TGCs are multiwire proportional chambers with a smaller wire-to-wire distance compared to the wire-to-cathode distance. This in combination with a high electric field around the wires lead to a very good time resolution for most of the tracks.

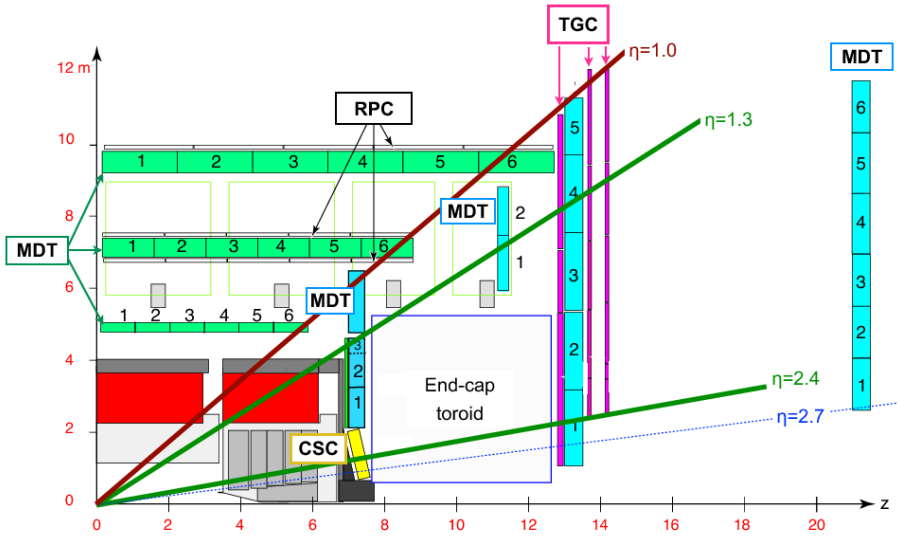



**Figure 3.11:** Schematic picture showing a quarter-section of the muon system in a plane containing the beam axis [17].

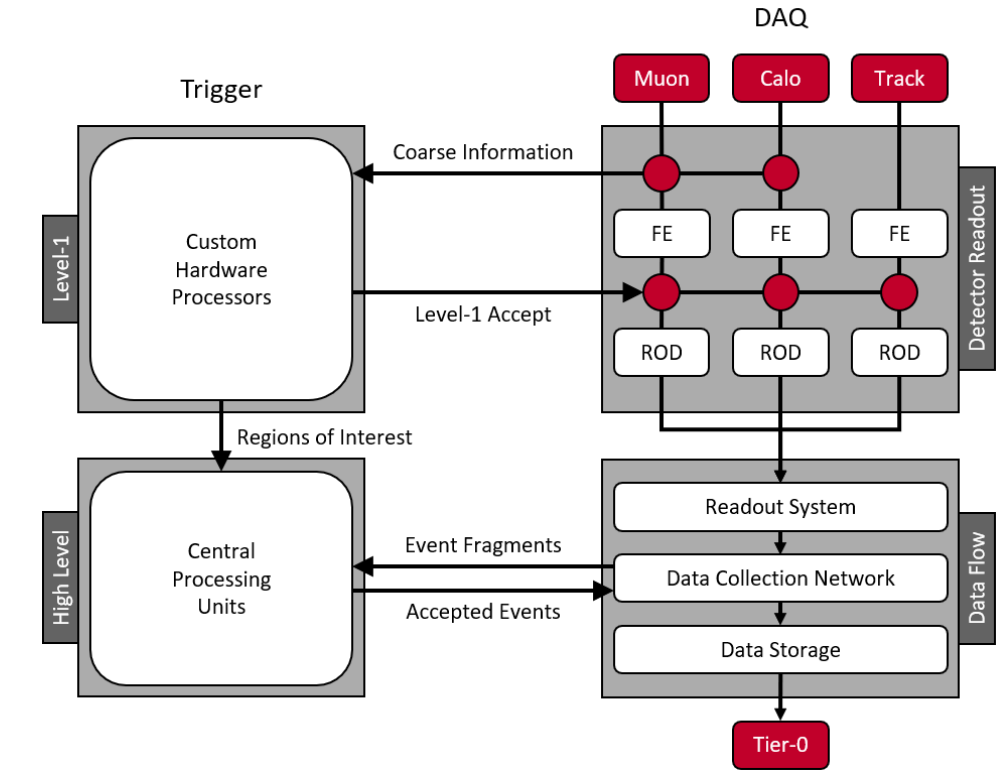
To complement the MDT measurement, RPC and TGC chambers measure each track coordinate in the direction orthogonal to that determined by the tracking chambers, i.e. in the non-bending ($\varphi$) plane. In addition, due to a fast signal transmission (only 15 to 25 ns), both chamber types provide bunch-crossing identification.

## 3.3 ATLAS Trigger and Data Acquisition

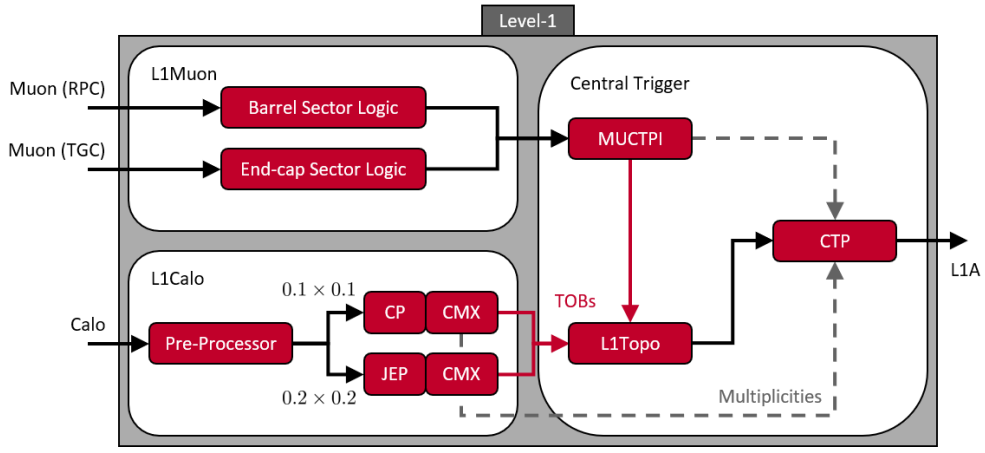
At a bunch crossing rate of 40 MHz and a given (design) luminosity of about $10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, corresponding to an average number of interactions per bunch crossing of about $\langle\mu\rangle = 40$, ATLAS has to deal with around 1.6 billion proton-proton collisions per second. The resulting amount of data can neither be completely processed nor permanently stored. Therefore, ATLAS uses a specialized two-level online event selection system. By distinguishing characteristics, this trigger system selects such events that are interesting for physics analyses.
A data acquisition (DAQ) system is running in parallel to the trigger system, channelling the full data from all detectors to permanent storage. An overview of the ATLAS Trigger and DAQ (TDAQ) system is given in Figure 3.12 (a). A more detailed description follows with a strong focus on the calorimeter trigger including its object identification algorithms.

### 3.3.1 Detector Readout and Level-1 Trigger

Each sub-detector uses specific front-end (FE) electronics, which receive, digitize and buffer the event data for a time long enough to accommodate the trigger latency. In the first trigger stage, namely the Level-1 Trigger, custom hardware processors based on Field Programmable Gate Array (FPGA) technology are used to reduce the rate from 40 MHz to 100 kHz within 2.5 $\mu$s. FPGAs are (re-) programmable (digital) integrated circuits that enable fast signal processing (further described in Chap. 4).




(a)




(b)

**Figure 3.12:** ATLAS TDAQ system: (a) overview; (b) Level-1 Trigger.

The trigger decision is based on a subset of information from the muon and calorimeter detectors. Figure 3.12 (b) shows the individual components of the Level-1 Trigger. After an event is accepted by the Level-1 Trigger, data from all the subsystems are transferred to the Readout Drivers (RODs) to allow full monitoring and verification of the functionalities, including input data, intermediate calculations and trigger results.

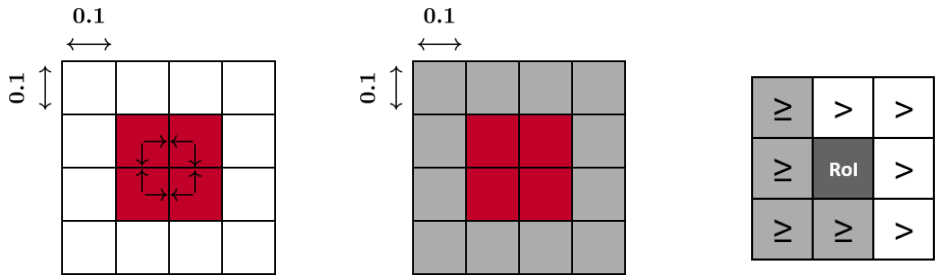
### Level-1 Muon Trigger

The Level-1 Muon (L1Muon) system is based on the trigger chambers, RPCs in the barrel and TGCs in the end-cap regions, which are part of the muon system (described in Sec. 3.2.5). For a muon candidate a coincidence of hits in the different trigger layers is required. Different ranges for $p_{\mathrm{T}}$ thresholds are applied by limiting the accepted width of the road, which tracks the path of the muon.
Results from the barrel and end-cap triggers are combined in the MUon to Central Trigger Processor Interface (MUCTPI), which calculates total multiplicity values for each $p_{\mathrm{T}}$ threshold. For a given muon candidate the MUCTPI also produces a set of object quantities (e.g. $\eta$, $\varphi$ and momentum), collectively referred to as Trigger Object (TOB).

### Level-1 Calorimeter Trigger

The Level-1 Calorimeter (L1Calo) system [18] receives trigger towers of $\Delta\eta\times\Delta\varphi = 0.1\times0.1$ (in most parts, but coarser at larger $|\eta|$), formed by analogue summation of up to 60 calorimeter cells. The Pre-Processor samples these signals, converts them into digital data and sends them to the Cluster Processor (CP) and Jet Energy Processor (JEP). These processors search for features in overlapping sliding windows and output their results via the Common Merger module eXtended (CMX). The CMX provides TOBs and multiplicities for each TOB type by counting the number of TOBs above different thresholds.
The CP subsystem hosts 56 Cluster Processor Modules (CPMs). It identifies and counts $e/\gamma$ and $\tau$ candidates over $|\eta| < 2.5$, which is the limit of high-precision data from the EM calorimeter. Both algorithms use very similar logic based on $\Delta\eta \times \Delta\varphi = 0.1 \times 0.1$ trigger towers. Narrow, high-$E_{\mathrm{T}}$ showers in the EM calorimeter are associated with $e/\gamma$ candidates, while the $\tau$ algorithm looks for $\tau$ decays into collimated clusters of hadrons.
The JEP subsystem contains 32 Jet/Energy Modules (JEMs) and covers $|\eta| < 4.9$. It receives sums of $2 \times 2$ trigger towers, giving a basic granularity of $\Delta\eta \times \Delta\varphi = 0.2 \times 0.2$. Since there is no need to keep EM and HAD calorimeters separate, each JEM sums the $2\times2$ EM and HAD regions to form "jet elements". Based on these jet elements, each JEM processes an area of calorimeter data to identify and count jet candidates, as well as to compute missing transverse energy ($E_{\mathrm{T}}^{\mathrm{miss}}$) and total transverse energy ($\Sigma E_{\mathrm{T}}$).
The CPM and JEM algorithms are described in more detail below.

**CPM Algorithms** The $e/\gamma$ and $\tau$/hadron algorithms are run for all possible sets of overlapping $4 \times 4$ trigger tower windows. The $2 \times 2$ trigger tower region at each center is tested for a local maximum, also called Region of Interest (RoI).
In case of the $e/\gamma$ algorithm, at least one of the four possible EM $1\times2$ and $2\times1$ pairs within an RoI is required to pass a threshold in $E_{\mathrm{T}}$ (Fig. 3.13 (a)). To differentiate between the signal and the main background (hadronic jets), the 12 EM towers surrounding the RoI are summed and required to be below a threshold (EM isolation). HAD isolation is required in the same way (Fig. 3.13 (b)). In addition, the sum of the central $2 \times 2$ HAD towers must be below a threshold (hadronic veto).
For the $\tau$/hadron algorithm, each of the four EM $1\times2$ and $2\times1$ pairs is added to the sum of the $2 \times 2$ towers in the HAD calorimeter, and at least one of the four sums is required to pass a threshold. For $\tau$/hadron candidates some level of isolation and penetration into the HAD calorimeters is allowed.



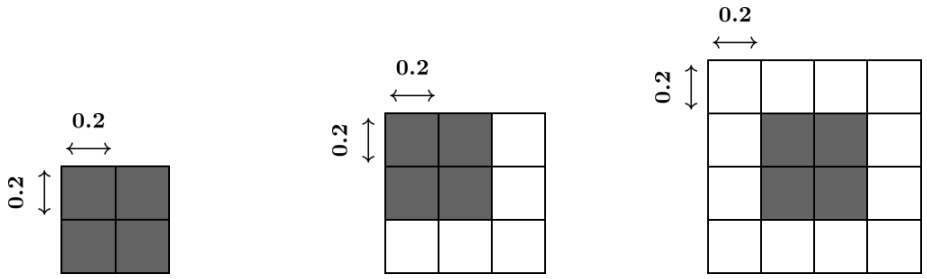

**(a)** $0.4 \times 0.4$ window **(b)** $0.4 \times 0.4$ window **(c)** overlapping RoIs

**Figure 3.13:** Elements used for the $e/\gamma$ and $\tau$/hadron algorithm in the (a) EM layer and (b) HAD layer; (c) local maximum test ($+\eta \rightarrow$, $+\varphi \uparrow$): four 'greater than' conditions (in the $+\eta$ and $+\varphi$ directions) and four 'greater than or equal' conditions (in the $-\eta$ and $-\varphi$ directions).

It is possible for an object to satisfy the algorithm in more than one overlapping trigger tower window. This multiple counting is avoided by requiring that the RoI being tested must be a local maximum compared to its eight overlapping nearest neighbors. In addition, the possibility of comparing equal digital values must be taken into account. Therefore, as illustrated in Figure 3.13 (c), a mixture of 'greater than' and 'greater than or equal' conditions is used.

**JEM Algorithms** For the jet algorithm, sums of $E_{\mathrm{T}}$ in windows consisting of $2 \times 2$, $3 \times 3$ and $4 \times 4$ jet elements, i.e. window sizes of 0.4, 0.6 and 0.8 in $\Delta\eta$ and $\Delta\varphi$ (Fig. 3.14), are compared to jet thresholds. These windows overlap and slide by one element, thus by 0.2 in $\Delta\eta$ and $\Delta\varphi$. The smaller window sizes are better for resolving multiple jets, while the largest window size includes more of the jet energy.

As with the CPM algorithms, it is possible for a jet to pass the threshold in more than one window. So again it is required that the (jet) RoI ($0.4 \times 0.4$ in $\Delta\eta \times \Delta\varphi$) must be a local maximum (Fig. 3.13 (c)). Note that for the $0.6 \times 0.6$ window there are four possible RoI candidates, where the one with the highest $E_{\mathrm{T}}$ sum is used.




**(a)** $0.4 \times 0.4$ window **(b)** $0.6 \times 0.6$ window **(c)** $0.8 \times 0.8$ window

**Figure 3.14:** Jet trigger algorithm windows based on jet elements of $\Delta\eta \times \Delta\varphi = 0.2 \times 0.2$; RoIs are shown in gray. For the $0.6 \times 0.6$ window there are four possible RoI candidates, only one of which is marked.

The JEMs also serve as the first stage of the $E_{\mathrm{T}}^{\mathrm{miss}}$ and $\Sigma E_{\mathrm{T}}$ triggers. For $E_{\mathrm{T}}^{\mathrm{miss}}$, Equation 3.9 turns into

$$\begin{pmatrix} E_x^{\mathrm{miss}} \\ E_y^{\mathrm{miss}} \end{pmatrix} = - \sum_{i \in \text{jet elements}} \begin{pmatrix} E_{\mathrm{T}} \cdot \cos\varphi \\ E_{\mathrm{T}} \cdot \sin\varphi \end{pmatrix}_i , \tag{3.18}$$

assuming that the invisible particles are massless. Each jet element is multiplied by the appropriate geometrical constants to obtain its transverse energy components $E_x$ and $E_y$. The final summation and comparison with thresholds are done on the CMXs.

#### Level-1 Topological Processor

The Level-1 Topological Processor (L1Topo) receives muon, $e/\gamma$, $\tau$ and jet TOBs plus missing and total transverse energy values from the MUCTPI and CMXs. Interesting event topologies, e.g. opening angles between TOBs and effective masses of TOB pairs, are calculated.

#### Central Trigger Processor

In addition to the multiplicities provided by the MUCTPI and CMXs, the Central Trigger Processor (CTP) receives various L1Topo output bits, indicating whether individual topological algorithms have been satisfied or if topological cut values have been met. Finally, the overall L1 accept decision is taken by combining the collected information.

### 3.3.2 High Level Trigger and Data Flow

The Readout System (ROS) temporarily stores the data (event fragments) received from the RODs. Upon request, information is provided to the subsequent stages.
The High Level Trigger (HLT) is a software based trigger, accessing more detector information than the Level-1 Trigger. CPUs are used to perform more precise analyses in the RoIs that are spotted by and received from the Level-1 Trigger. Event fragments from the muon, calorimeter and tracking detectors are used to further reduce the event rate to 1 kHz. The HLT result, accept or reject, is returned to the DAQ system, where the event fragments are collected from the ROS to build a single event data structure, the event. The full event is then prepared for permanent storage, and thus transferred to CERNs central data recording facility, Tier-0.

## 3.4 ATLAS Upgrades

The first stage of the ATLAS upgrade program, Phase-0, took place during the first LHC long shutdown (LS1) from 2013 to 2015. Core activity was the installation of the IBL in the Inner Detector (described in Sec. 3.2.3) to improve the tracking by providing an additional measurement point. From 2015 to 2018, referred to as Run-2, ATLAS took data from proton-proton collisions at a centre-of-mass energy of 13 TeV and a nominal luminosity of about $10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$ ($\langle\mu\rangle = 40$).

The next ATLAS upgrade period, Phase-1, already started with the beginning of LS2 in 2019, preparing ATLAS for the next round of data taking, Run-3. By this time, the third LHC run is planned to start early 2022 and to continue until the end of 2024.

After the Phase-I upgrade, ATLAS has to deal with an increased instantanenous luminosity of about 2.5 times the nominal one, and thus an enhanced $\langle\mu\rangle$ of 60. To still efficiently trigger and record data, major upgrades to the TDAQ system are required in particular. The Phase-I TDAQ upgrade will also benefit from the construction of a new wheel in the innermost layer of the muon system, the New Small Wheel (NSW). New information from the NSW will be included in the L1Muon end-cap trigger, improving the rejection of fake triggers.

### 3.4.1 Level-1 Calorimeter Trigger Upgrade

One of the physics goals of the ATLAS experiment is to maintain sensitivity to electroweak processes despite the increased event rate. Therefore, as part of the Phase-I upgrade, the L1Calo trigger will be equipped with a new subsystem of three Feature EXtractors (FEXs), each optimized to trigger on different physics objects. The FEXs take advantage of the latest FPGA technology available by exploiting higher granularity data from the calorimeters, and thus improving the physics object selection using more sophisticated algorithms. The finer granularity data will be optically transmitted from new dedicated LAr calorimeter readout electronics.
For the beginning of Run-3 it is important to maintain the legacy trigger, providing a stable and well understood trigger for ATLAS physics and a baseline for measuring the efficiency of the new trigger. Figure 3.15 shows the system architecture during Run-3.

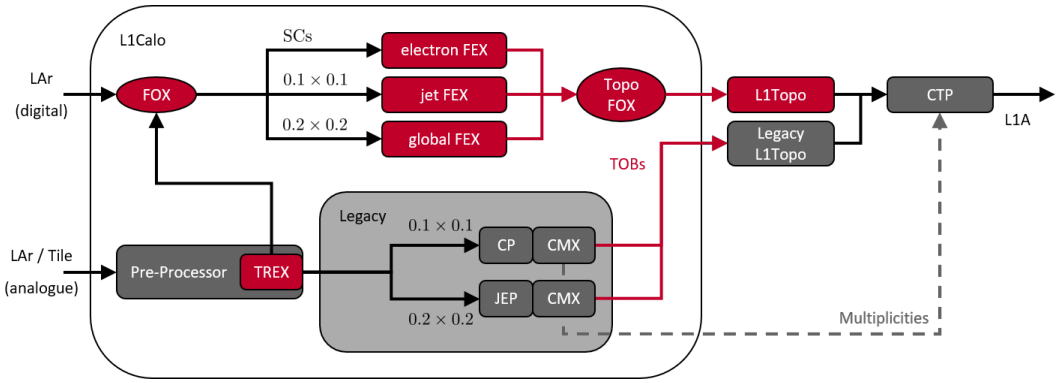



**Figure 3.15:** Level-1 Calorimeter Trigger during Run-3. New components are shown in red. Connections to the L1Muon system are not shown.

The Tile Rear Extension (TREX) module is a subcomponent of the Pre-Processor system. It provides digitized values of Tile $E_{\mathrm{T}}$ to the FEXs and to the legacy processors. Better resolved data from the LAr system are provided by the LAr Trigger prOcessing MEzzanine (LATOME). The new trigger granularity is no longer based on $\Delta\eta\times\Delta\varphi = 0.1\times0.1$ trigger towers, but on so-called super cells (SCs). For $|\eta| < 2.5$ a single trigger tower contains 10 SCs, spreaded over four layers in depth with a 1-4-4-1 segmentation in $\eta$. Thus, the dimensions of a SC in the first and last layer are $\Delta\eta \times \Delta\varphi = 0.1 \times 0.1$, while it is $\Delta\eta \times \Delta\varphi = 0.025 \times 0.1$ for each SC in the inner two layers (the full granularity of an EM barrel module is shown in Fig. 3.8).
In the Fibre Optical Exchange (FOX) box signals from LATOME/LAr and TREX/Tile are re-mapped and distributed to the individual FEX systems:

- The electromagnetic Feature EXtractor (eFEX) identifies $e/\gamma$ and $\tau$ candidates using the newly available SC granularity. The eFEX system consists of 24 modules, each equipped with four processor FPGAs, covering the region $|\eta| < 2.5$.

- The jet Feature EXtractor (jFEX) identifies jets and computes $\Sigma E_{\mathrm{T}}$ and $E_{\mathrm{T}}^{\mathrm{miss}}$. In total, six jFEX modules with four processor FPGAs each receive data from the entire calorimeter (up to $|\eta| = 4.9$), exploiting a granularity of $\Delta\eta \times \Delta\varphi = 0.1 \times 0.1$. Thanks to the large overlap between neighboring processors, jFEX is also able to provide additional information on $\tau$ isolation. Furthermore, a search for electrons in the very forward regions is carried out to extend eFEX.

- The global Feature EXtractor (gFEX) identifies large-radius jets and computes global event observables. Just like jFEX, gFEX processes information from the entire calorimeter. Its input granularity is $\Delta\eta \times \Delta\varphi = 0.2 \times 0.2$, which allows to receive and compute everything on a single module with three processor FPGAs plus one merger FPGA, used to combine global TOB fragments into global TOBs.

A new L1Topo system will be able to process all the (TOB) information provided by the FEX systems and received through the L1Topo Fibre Optical Exchange (TopoFOX) box. Just like the FOX on the input side of the FEXs, the TopoFOX re-maps signals and distributes them to the individual L1Topo modules. There is no connection between FEXs and CTP, since multiplicities are computed in one out of three L1Topo modules with two processor FPGAs each.

## 3.5 ATLAS Simulations

The simulation software for the ATLAS experiment [19] includes various generators that produce complete collision events starting from e.g. a proton-proton initial state. To describe the hard scattering processes, immediate particle decays, electromagnetic and hadronic shower developments, etc., matrix elements are calculated that describe the transition amplitudes between the initial and final states. The history of the interactions, including incoming and outgoing particles, is recorded for each event. This kind of information is known as *truth*, because it represents a complete list of particles that were actually in a particular event, regardless of whether or not they are ultimately seen by the detector. Truth $E_{\mathrm{T}}^{\mathrm{miss}}$, for example, is then calculated according to Equation 3.9 by summing over the list of corresponding truth particles.
A full detector description is required to properly simulate the response of the individual sub-detectors, including physics interactions and digitization of energy values into voltages and currents. The latter is particularly important so that simulated and real data can run through the same trigger and reconstruction stages. The simulation of ATLAS relies on the GEANT4 particle simulation toolkit [20]. It offers the ability of creating a geometrical model of the detector, especially taking into account the different materials used, to then simulate the passage of particles.

### 3.5.1 Jet Reconstruction

In the ATLAS experiment, the anti-$k_{\mathrm{T}}$ jet clustering algorithm [21] is used as standard to determine jet candidates. A general distinction is made between truth and reconstructed

("offline") jets. The latter are based on clusters of calorimeter cells, while truth jets are groups of truth particles.
The anti-$k_{\mathrm{T}}$ algorithm introduces the distances $d_{ij}$ between the entities $i$ and $j$ (truth particles or calorimeter cells) and $d_{iB}$ between the entity $i$ and the beam ($B$):

$$d_{ij} = \min\left(\frac{1}{(k_{\mathrm{T}})_i^2}, \frac{1}{(k_{\mathrm{T}})_j^2}\right) \cdot \frac{\Delta R_{ij}}{R^2}, \tag{3.19}$$

$$d_{iB} = \frac{1}{(k_{\mathrm{T}})_i^2}, \tag{3.20}$$

where $(k_{\mathrm{T}})_{i/j}$ refers to the transverse momentum of the entity $i/j$ and $R$ is a user-selectable free radius parameter. The smallest of the distances is identified and if it is a $d_{ij}$ the entities $i$ and $j$ are combined (energy-weighted mean), while if it is $d_{iB}$ then $i$ is called a jet and is removed from the list of entities. The distances are then recalculated and the entire procedure is repeated until there are no more entities left.

### 3.5.2 Simulation Samples

The simulation samples that were used for the trigger efficiency studies in the context of this thesis are listed in Table 3.1.

| **Sample** | **Generator** |
|---|---|
| Dijet | Pythia 8 + EvtGen |
| $ZH$ with $Z \to 2\nu$ and $H \to 2b$ | Powheg + Pythia 8 + EvtGen |
| $2H \to 4b$ | Herwig 7 + EvtGen |
| $VH$ with $V \to 2q$ and $H \to$ inv | Powheg + Pythia 8 + EvtGen |

**Table 3.1:** ATLAS simulation samples at a center of mass energy of 13 TeV. The $VH$ sample has a lower $E_{\mathrm{T}}^{\mathrm{miss}}$ cut of 75 GeV; the vector boson $V$ can be either a $W$ or a $Z$ boson. For further information on the various generators, see Reference [19].

The Dijet sample is also known as *minimum bias* sample. Minimum bias events are inelastic proton-proton collisions, which are dominated by soft (low-$p_{\mathrm{T}}$, low-multiplicity) QCD interactions. The modeling of such events is important for a better understanding of pile-up. Dijet samples produced for different leading truth jet $p_{\mathrm{T}}$ slices are named JZ0W (0 to 20 GeV), JZ1W (20 to 60 GeV), JZ2W (60 to 160 GeV), etc.
The performance of L1 $E_{\mathrm{T}}^{\mathrm{miss}}$ triggers was studied using the $ZH$ with $Z \to 2\nu$ and $H \to 2b$ sample, while the $2H \to 4b$ sample was used for corresponding studies related to jets.
For direct searches for invisibly decaying Higgs bosons, the Higgs boson has to recoil against a visible system, while a certain amount of $E_{\mathrm{T}}^{\mathrm{miss}}$ has to be present in the final state. For the corresponding trigger efficiency studies, the $VH$ with $V \to 2q$ and $H \to$ inv sample was used, where two jets are expected to be in the final state. In case of highly boosted $W/Z$ bosons, the jets can be close-by or even merge into one large jet.

CHAPTER 4

# Technology and Technological Knowledge

*Any sufficiently advanced technology*
*is indistinguishable from magic.*
( Sir Arthur C. Clarke )

In the previous chapter it has already been mentioned, that for the purpose of fast signal processing, FPGAs are used in the first stage of the ATLAS TDAQ system. This chapter is dedicated to the technology of these hardware components [22].
At the very beginning, a brief overview of digital logic is given. The structure and functions of FPGAs are presented afterwards. Finally, special features of large FPGA devices are discussed.

## 4.1 Binary Information and Digitial Logic

As the name suggests, binary information is based on two possible states: Positive logic (bit value '1' in presence of a voltage) and negative logic (bit value '0' in absence of a voltage). Thus, analogue signals can be converted into digital data by applying high and low voltage limits. Safe operations at very high speeds can be ensured by leaving a sufficiently large gap between these two limits.
Multiple bits are used to represent more information. A collection of $n$ bits has $2^n$ states.

Metal Oxide Semiconductor (MOS) transistors are voltage controlled switches. A (p-type/n-type) MOS transistor acts like a switch between its source (high/low) and drain. The switch of a p-type transistor will be closed (transistor active), if its gate is low. This allows current to flow from source to drain. On the other hand, if its gate is high, the transistor will be off (open switch) and no current will flow. The n-type gate works complementary. Hence, a combination of both transistor types is called Complementary MOS (CMOS). CMOS transistor-level circuits are used to implement logical operations. In CMOS technology, for example, a 2-input AND gate is constructed from a NAND followed by an inverter, which requires a total of $4 + 2 = 6$ transistors.

## 4.2 Field Programmable Gate Arrays

In short, FPGAs are (re-) programmable (digital) integrated circuits. An FPGA contains an array of programmable logic blocks plus an also programmable routing network.

Different to the programming of computers (CPUs), the term "programming" refers not only to the specification of time sequences, but to the definition of the desired circuit. This structure is formulated using a hardware description language (discussed in Chap. 5) and translated into a configuration file, which specifies how the physical elements in the FPGA are to be configured and interconnected. Therefore, one speaks more of the "configuration" of an FPGA.
The content of the configuration file is loaded into Static Random-Access Memory (SRAM) elements, manufactured in CMOS technology and distributed all over the FPGA. These memory cells determine the functions of individual logic blocks and control which signal lines in the routing network are connected to each other.
The basic structure of an FPGA is shown in Figure 4.1. The package contains all logic blocks embedded in the routing network. It provides connections to the external environment via input/output (I/O) pins for the purpose of power supply and data transfer.

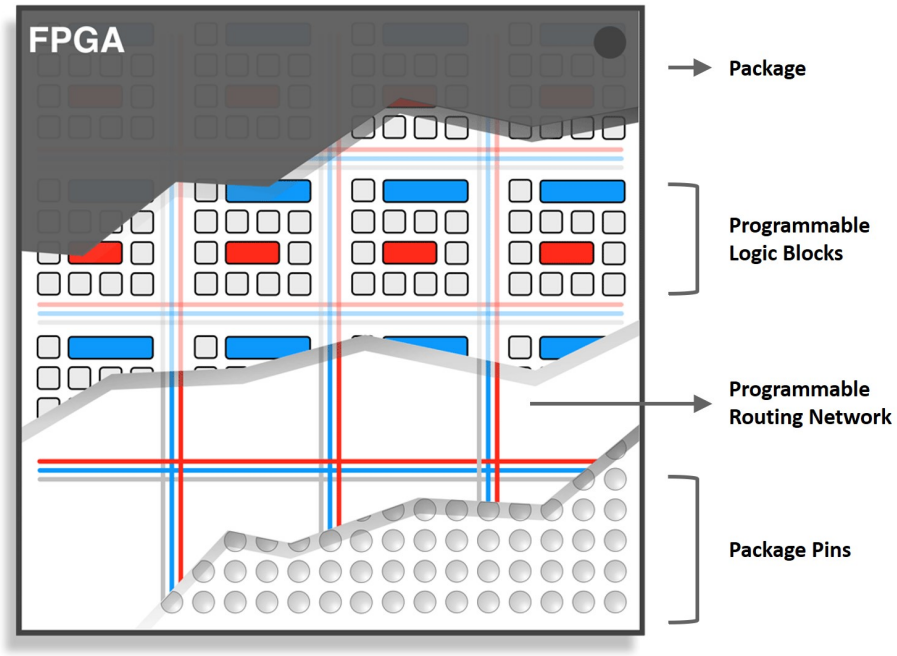
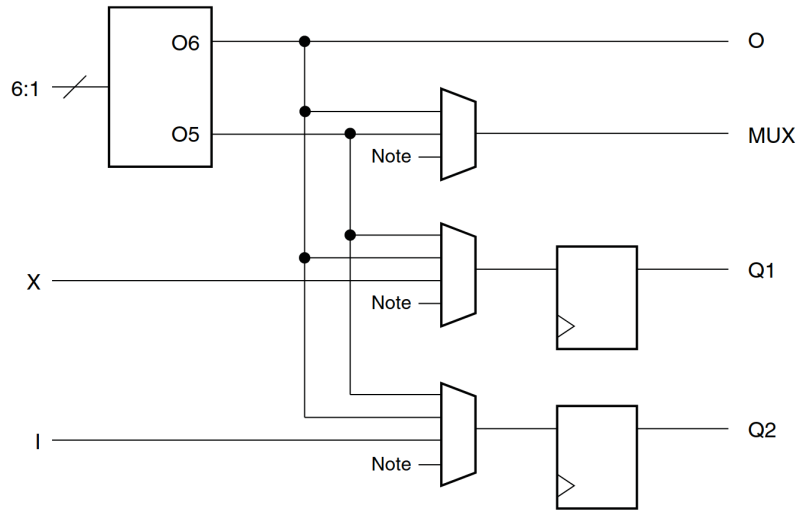



**Figure 4.1:** Schematic picture showing the basic structure of an FPGA [23].

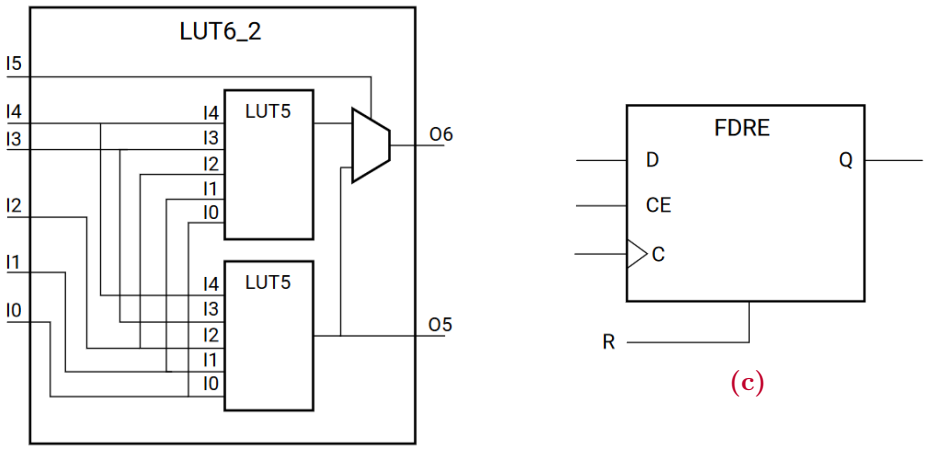
From today's perspective, the first FPGAs were rather small. Nowadays, FPGAs are made up of billions of transistors corresponding to millions of logic blocks. The Xilinx [24] Virtex UltraScale+ (US+) devices [25] are currently among the latest and most advanced FPGAs available on the market. They are the target devices for this thesis, as they are used in the jFEX system. In the following, the most relevant components and features of US+ type FPGAs are highlighted.

### 4.2.1 Configurable Logic Blocks

Each Configurable Logic Block (CLB) of an US+ type FPGA device consists of eight Lookup Tables (LUTs) and 16 Flip-Flops (FFs), as well as several Multiplexers (MUXs) and arithmetic carry logic [26]. CLB slices can be easily connected to each other to build larger functions. A schematic picture of a CLB sub-unit/octant is shown in Figure 4.2 (a).




**(a)** Note: MUX inputs include carry and wide multiplexers (not shown).




**(b)** **(c)**

**Figure 4.2:** Schematic picture of a (a) CLB sub-unit/octant [26], (b) 6-input, 2-output LUT [27] and (c) D-type FF with clock enable and synchronous reset [27].

LUTs are used for implementing small binary functions, FFs for storing data and enabling clocked data propagation. The output of a LUT can be connected to the output of the underlying CLB slice or optionally be registered in a FF. MUXs can be used to combine LUTs to create even wider functions. Direct inputs to the slice (X and I in Fig. 4.2 (a)) as well as results of the internal MUXs and carry logic can also be registered. Carry logic uses the concept of generating and propagating carry bits by switching from a chain of logic gates to a tree of logic gates (from a linear latency to a logarithmic latency). Dedicated carry logic thus provides higher density and improves the performance of arithmetic functions, e.g. adders, counters and multipliers.

There are two types of CLB slices: The one described above is referred to as SLICEL, where the 'L' is for logic. The LUT in a SLICEM, where the 'M' is for memory, can also be configured as distributed memory or a shift register (described further down).

#### Lookup Tables

In digital circuits (and especially in FPGAs) simple functions such as logical operations (AND, OR, XOR, etc.) are implemented using LUTs. Compared to a transistor-level circuit, these elements can be configured more easily.
LUTs within SLICEL type CLBs have six independent inputs and two independent outputs. Each 6-input LUT actually consists of two 5-input LUTs with shared inputs (see Fig. 4.2 (b)). Furthermore, a MUX extends its functionality.
LUTs within SLICEM type CLBs have a separate write address, write enable and clock signal. Therefore, they can be used as 64-bit memory with data coming from the direct inputs to the slice. Multiple LUTs in a SLICEM, and also multiple SLICEMs, can be combined to store larger amounts of data (up to 512 bits per slice). Alternatively, these LUTs can be configured as a 32-bit shift register in order to save FF resources. Each LUT can delay data from one to 32 clock cycles. LUTs and SLICEMs can be cascaded to achieve even larger delays.

#### Flip-Flops

Storage elements in both CLB slice types can be configured as edge-triggered D-type FFs, where the 'D' is for data or delay. As already mentioned above, data from various sources can be registered by a single FF, which enhances design flexibility.
Figure 4.2 (c) shows the schematic of a D-type FF. Next to the data input (D) and output (Q), it has three control signals: clock enable (CE), clock (C) and set/reset (R). Each CLB slice has four clock enables, two clock inputs and two set/reset inputs. While the LUT and two FFs of a given CLB octant share the same clock and set/reset signals, different clock enables are provided for the upper and lower halves of the CLB and for the two FFs per octant. Each clock enable specifies whether the FFs it is connected to should store new data on the next clock edge or keep the old one.
Each storage element can also be configured as a level-sensitive latch. In contrast to edge-triggered FFs, a latch is transparent during the entire active clock phase, i.e. input changes can affect the output immediately.

### 4.2.2 Specialized Building Blocks

US+ type FPGAs also contain specialized building blocks, which even further extend the functionality, e.g. by enabling the configuration of larger, more complex (arithmetic) functions:

#### Digital Signal Processor

Digital signal processing (DSP) applications require many binary multipliers and accumulators, which are implemented in dedicated DSP resources. The basic functionality of these DSP slices is shown in Figure 4.3. For a more detailed view, see Reference [28].
Each DSP slice consists of a 27-bit pre-adder/subtracter (D input), a $27 \times 18$ multiplier and a flexible 48-bit arithmetic logic unit (ALU), that serves as a post-adder/subtracter (C input), accumulator or bitwise AND, OR, NOT, NAND, NOR, XOR and XNOR. Internal MUXs can be configured to perform special operations, e.g. squaring, while internal pipeline registers are required for some operations to run at full speed.
Multiple DSP slices can be cascaded without the use of external logic.

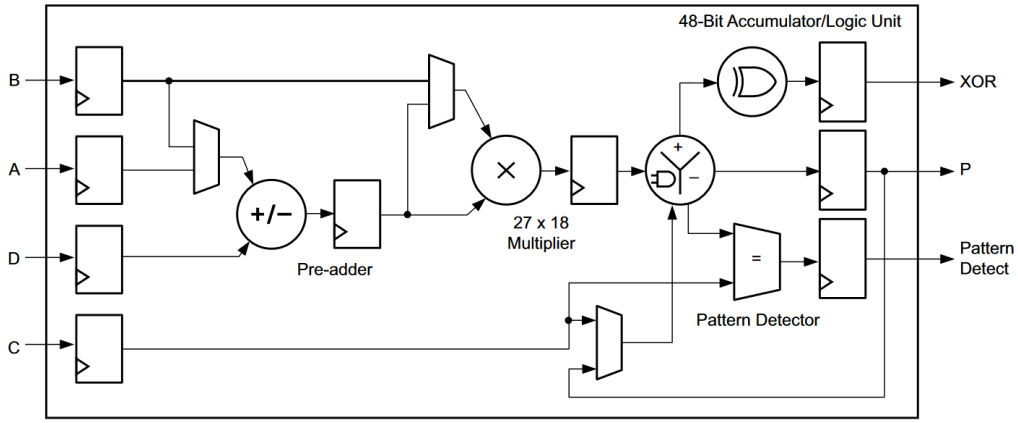



**Figure 4.3:** Basic DSP (DSP48E2) functionality [28].

#### Random-Access Memory

The US+ architecture features two kinds of Random-Access Memory (RAM) [29], which provide an increased density and capacity compared to SLICEM LUTs:
The Block RAM (BRAM) consists of two 18-kib sub-units, where each can be configured and used as a 1-bit wide and 16,384 deep to an 36-bit wide and 512 deep memory. Read and write operations are fully synchronous to the input clock, while their ports can operate fully independent and asynchronous to each other.
The second type of dense and deep RAM is the Ultra RAM (URAM). It features 288 kib of storage with a fixed width of 72 bits, resulting in a depth of 4,096. Different to BRAMs, URAMs can only support read or write operation per port per cycle. Thus, an URAM is less configurable than a BRAM, but has eight times its capacity.
Similar to other logic, BRAMs and URAMs can be cascaded to form even deeper memory configurations.

### 4.2.3 Clock and Routing Resources

Inputs and outputs of the functional elements described above are connected via a (re-) programmable routing network. CLB slices, DSP slices and RAM resources are arranged in a regular array. Various connections are provided by switch matrices, which run vertically and horizontally through the entire device (shown in Fig. 4.1: CLB slices/network in gray, DSP slices/network in blue, RAM resources/network in red).
US+ devices are subdivided into columns and rows of segmented clock regions. Each clock region contains CLB slices, DSP slices, RAM resources, various interconnects and the root of the clock network at its center. A sophisticated clock distribution network is then provided by vertical and horizontal connectivity through separate clock routing and clock distribution resources. More details on clocking resources and clock management can be found in Reference [30].

### 4.2.4 Input and Output

I/O pins are grouped in so-called banks. Most I/O banks consist of 52 pins that can be used for input, output or bidirectional operations. Such pins, which are used to transfer data,

are connected to configurable I/O Blocks (IOBs) that are located on the perimeter of an FPGA (not shown in Fig. 4.1). Each IOB consists of an input buffer and an output buffer. The line rates (serial bit rates) vary from general purpose (low-speed) I/O interfaces to fast I/Os with up to many Gbit/s. The latter are described in more detail below. For more information on packaging, pinouts and I/O resources, see References [31, 32].

#### Multi-Gigabit Transceivers

Multi-Gigabit Transceivers (MGTs) are used in high-speed communications to minimize the number of I/O pins and interconnects. As a serializer/deserializer, an MGT transmits parallel data as a stream of serial bits and converts serial bits it receives into parallel data. Line rates from 500 Mb/s to 32.75 Gb/s are supported (GTY transceivers in the US+ architecture [33]). Thus, MGTs can cope with a wide range of optical rates. Each transceiver is highly configurable and tightly integrated with the programmable logic resources, well fitted for parallel data processing algorithms.

## 4.3 Large FPGA Devices

The term *large FPGA device* is a time-changing expression, since especially the device capacity increases significantly with each new FPGA device family available on the market. Table 4.1 shows the number of CLB LUTs, CLB FFs and DSP slices for different devices. The CP and JEP FPGAs were already operating during Run-1 and Run-2 as part of the Level-1 Calorimeter Trigger in the ATLAS TDAQ system, while the FEX FPGAs will start operating with the beginning of Run-3.

| Device Type | CLB LUTs | CLB FFs | Multiplier Blocks/ DSP Slices |
|---|---|---|---|
| XCV1000E (CP) | 27,648 | 27,648 | - |
| XC2V3000 (JEP) | 28,672 | 28,672 | 96 |
| XC7VX550T (eFEX) | 346,400 | 692,800 | 2,880 |
| XCVU9P (jFEX, gFEX) | 1,182,240 | 2,364,480 | 6,840 |

**Table 4.1:** Number of logic resources for different FPGA devices [34–37].

When comparing devices, it is not just about the number of logic elements, but also their functionality. As an example, 4-input LUTs are available in JEP FPGAs, while for jFEX FPGAs there are 6-input LUTs.
Some of the US+ architecture-based devices (e.g. XCVU9P) are created using a manufacturing process known as Stacked Silicon Interconnect (SSI) technology. Additional considerations are required when targeting such devices.

### 4.3.1 Stacked Silicon Interconnect Technology

There are many challenges when connecting two or more FPGAs to create a larger, so-called *virtual FPGA*, e.g. limited connectivity, excessive latency and power penalty [38]. With the SSI technology, Xilinx has developed a new approach, which enables high-bandwidth connectivity between multiple dies within a single package. Both latency and

power consumption are lower than in the case of combined FPGAs. By combining several dies in a single device, it is feasible to match or exceed the capacity and bandwidth of large monolithic devices. In addition, smaller dies can be manufactured much faster.
Xilinx SSI technology combines multiple FPGA die slices, called Super Logic Regions (SLRs), via a passive interposer. Data flow between neighboring SLRs is provided by high-bandwidth, low-latency connections, referred to as Super Logic Lines (SLLs). An illustration of such a die stack-up with four SLRs is given in Figure 4.4. The jFEX FPGA devices are made up of 3 SLRs.

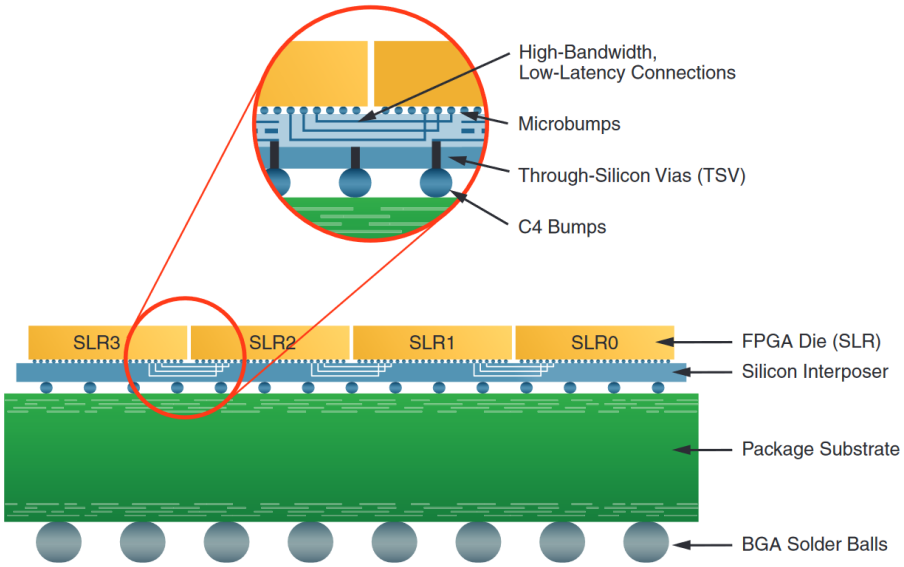



**Figure 4.4:** Side view of a die stack-up with four FPGA SLRs based on the Xilinx SSI technology [39].

Note that there are neither SLR0-to-SLR2, SLR0-to-SLR3 nor SLR1-to-SLR3 connections. Each SLR has its own clocking and configuration circuitry. Cascading of e.g. carry logic, DSP slices or memory resources does not propagate across SLR components at all.
The silicon interposer offers far finer interconnect geometries compared to organic or ceramic substrates. The interposer stack-up is mounted on a package substrate, providing connections to the package pins.

CHAPTER 5

# Firmware Development Methodologies

*In fact,*
*the opportunities for progress*
*lie in just those areas of physics*
*that seem most messy.*

(Steven Weinberg)

The structure and characteristics of FPGAs have been discussed in the previous chapter. This chapter is about the path from the desired function or algorithm to the integrated circuit and thus the configuration of an FPGA [40].
A brief introduction to hardware description languages is followed by a more detailed description of the software suite used to generate FPGA configuration files, highlighting the overall design flow.

## 5.1 Hardware Description Languages

A hardware description language (HDL) is a specialized computer language used to describe the structure and behavior of digital logic circuits at a high level. In contrast to most (software) programming languages, HDLs explicitly include the notation of time.
VHDL[1] [22] and Verilog are the most commonly used HDLs. VHDL is more verbose than Verilog, which can improve readability and may be a better choice for beginners. In this thesis, VHDL is used to write the source code. A brief overview of its features is given below.

### 5.1.1 VHDL

In VHDL, a design begins with the declaration of an `entity`. It describes the interface, containing a port list with incoming and outgoing signals. Each entity has at least one `architecture`, which contains the actual implementation, i.e. combinatorial logic, registers and how they are interconnected. In other words, it specifies the internal behavior and how this relates to the ports of the entity.
A simple AND gate in VHDL is shown in Listing 5.1 as an example.

[1] Very High Speed Integrated Circuit Hardware Description Language (also VHSIC Hardware Description Language), or VHDL for short.

```
1  library IEEE;
2  use IEEE.std_logic_1164.all;
3
4  -- interface
5  entity AND_GATE is
6    port(
7      CLK : in  std_logic := '0';
8      A   : in  std_logic := '0';
9      B   : in  std_logic := '0';
10     C   : out std_logic := '0'
11   );
12 end entity AND_GATE;
13
14 -- actual implementation
15 architecture IMPLEMENTATION of AND_GATE is
16 begin
17   C <= A and B when rising_edge(CLK);
18 end architecture IMPLEMENTATION;
```

**Listing 5.1:** Registered AND gate in VHDL.

VHDL is not case sensitive. However, the use of upper and lower case letters increases readability. A comment begins with a '`--`'.
The AND gate takes two inputs (`A` and `B`) and produces one output (`C`). Within the port list, the type of each signal needs to be specified, while its value can optionally be initialized. The `std_logic` type is a 9-valued logic (`U, X, 0, 1, Z, L, H, W, -`) and has to be imported from an external library. It is an extension of the 2-valued bit data type, since in reality it can happen that signals are neither `0` nor `1`, but e.g. uninitialized (`U`) or driven by more than one active signal (in conflict; `X`). These additional values are particularly important for logic simulations, which are described further down.
The logical conjunction is defined in the architecture body, which starts right after the keyword `begin`. The signal assignment is represented by a '`<=`' (equivalent to the mathematical equation $C = A \wedge B$). The result is captured at the rising edge of the incoming clock signal `CLK`. The `rising_edge` function detects the rising edge of a `std_logic` value and returns 'true' when the signal changes from a low value to a high value.

The header of an architecture may contain local signal declarations. These are for internal use only and therefore not visible outside of the entity. Code that is written directly in an architecture body is executed concurrently. Sequential statements only occur in special environments, e.g. processes or functions. Global design parameters, such as constants, type and function definitions, can be collected in and imported from separate files, so-called packages.
Multiple architectures can be assigned to the same entity and other entities can be instantiated within an architecture, allowing the construction of complex designs. The top entity of an FPGA design defines how the ports are mapped to the physical pins and therefore to the outside world. The actual mapping is defined in a separate constraints file.

## 5.2 Vivado Design Suite

With increasing design capacity and complexity, it is hardly possible to manually create FPGA configuration files. The Vivado Design Suite [41] is a software suite developed by Xilinx for the synthesis and analysis of HDL designs. It includes an built-in logic simulator and supports both VHDL and Verilog.
This section is dedicated to the firmware development with VHDL and Vivado.

### 5.2.1 Design Flow

Vivado offers a complete design flow, from the specification via source code to the actual configuration of an FPGA. A schematic picture of the Vivado register transfer level (RTL)-to-bitstream FPGA design flow is shown in Figure 5.1.

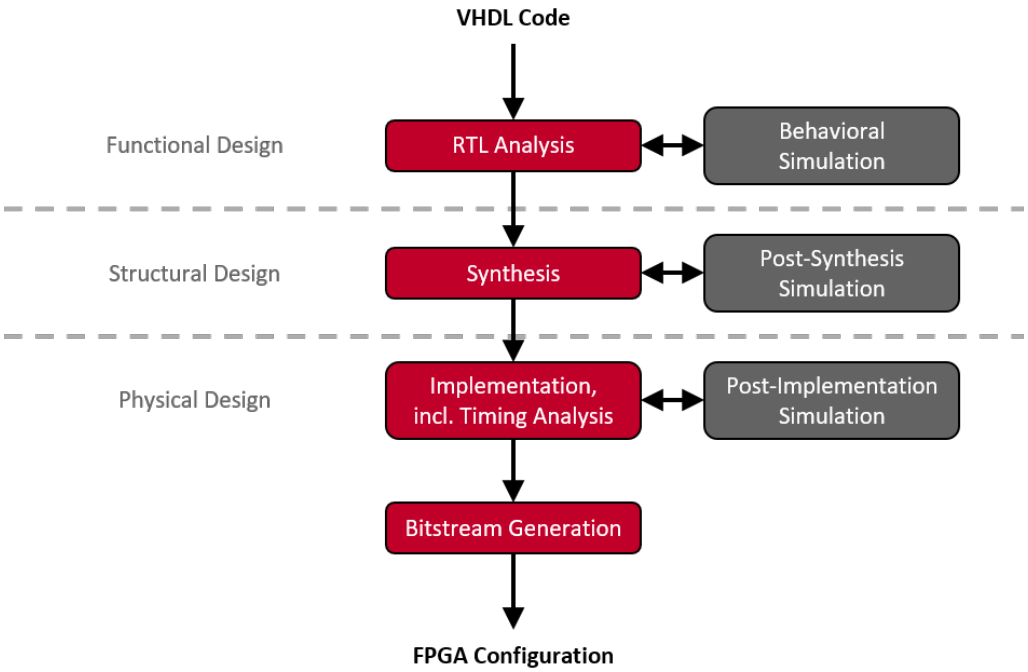



**Figure 5.1:** Vivado RTL-to-bitstream FPGA design flow.

Design analysis and verification is enabled at each stage, including e.g. logic simulation, power analysis, design logic visualization or debugging. A complete list of all the features can be found in Reference [42].
The individual stages are described in more detail below. The registered AND gate in Listing 5.1 serves as an example.

#### RTL Analysis and Behavioral Simulation

The design specified in the VHDL code is referred to as RTL design, as it describes how data flow and are processed between registers. As part of the RTL analysis, the design is transformed into a description of behavioral components and their relation to each other. This so-called *elaborated design* enables various analysis views to debug and optimize the design. Figure 5.2 shows the corresponding schematics for the registered AND gate.

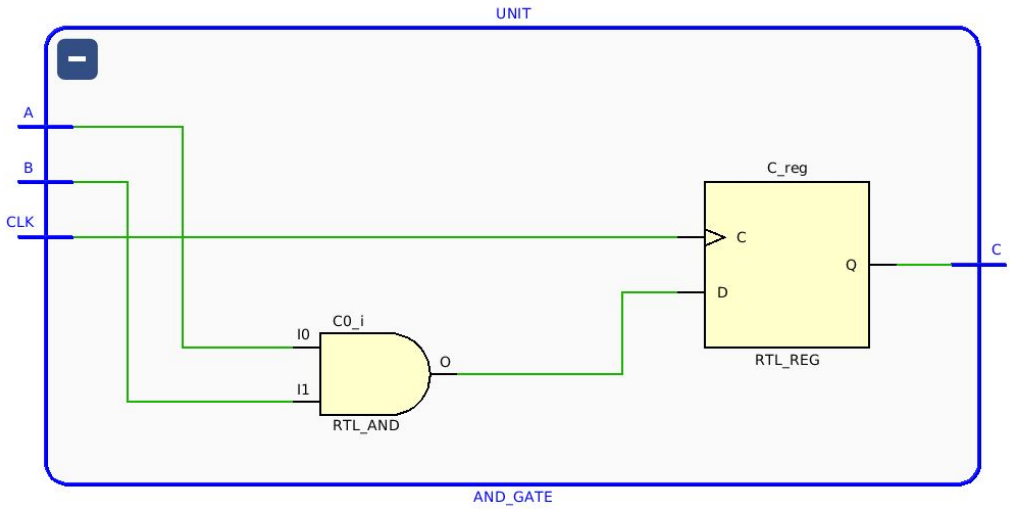



**Figure 5.2:** Elaborated design schematics generated from the VHDL code in Listing 5.1.

As specified in the source code, signal A is ANDed with signal B and the result is registered using the incoming clock signal CLK. Note that the AND gate is instantiated in a top entity, which takes care of the connections to the physical pins of the FPGA.

In addition to the RTL analysis, a low-level (behavioral) simulation can be performed using the HDL description of the design. It allows to verify syntax and functionality without proper timing information such as routing delays. Before being able to run such a simulation, a so-called *test bench* needs to be created. Listing 5.2 shows the lines of code required to verify the functionality of the VHDL code in Listing 5.1.

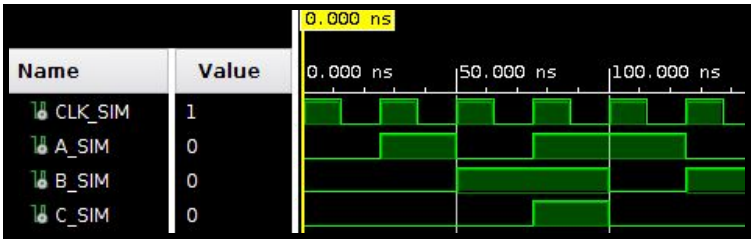
The entity of the test bench remains empty, while its architecture contains the entire simulation cycle. The AND gate, referred to as *unit under test*, is instantiated at the end of the architecture body. Its ports are connected to locally declared signals. The clock signal is generated inside a process, which is executed sequentially and starts again after each run. It is alternating between 1 and 0 every 12.5 ns, thus building a 40 MHz input clock. Within a second process, test inputs for signal A and signal B are generated. The input configuration of both signals is supposed to vary with each clock cycle.
The simulation results are displayed in a *static simulation waveform viewer*. Figure 5.3 shows four snapshots at different time. At first, A and B adopt the values with which they were initialized. All four snapshots combined show the desired functionality of the AND gate: C only assumes the value 1, if A and B are set to 1, otherwise 0.

### Synthesis and Post-Synthesis Simulation

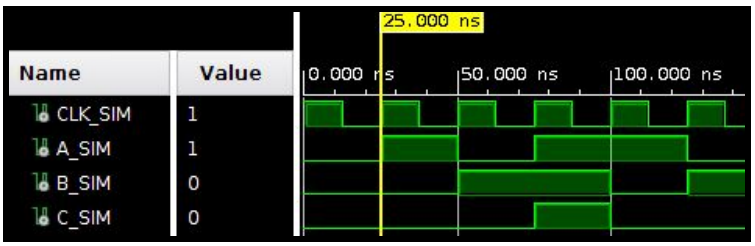
After the successful behavioral simulation of the design, the elaborated design is ready to be mapped to hardware components and their interconnections. The synthesis tool also performs logic optimization, where e.g. unrequired (not properly connected) design parts are removed. It creates a list of *nets* (signal lines) and *leaf cells* (hardware components), which can be displayed in schematics.

```
1  library IEEE;
2  use IEEE.std_logic_1164.all;
3
4  entity AND_GATE_SIM is
5  end entity AND_GATE_SIM;
6
7  architecture SIMULATION of AND_GATE_SIM is
8
9    -- local signal declarations
10   signal CLK_SIM : std_logic := '0';
11   signal A_SIM   : std_logic := '0';
12   signal B_SIM   : std_logic := '0';
13   signal C_SIM   : std_logic := '0';
14
15 begin
16
17   SIMULATION_CLOCK: process -- sequential
18   begin
19     CLK_SIM <= '1';
20     wait for 12.5ns;
21     CLK_SIM <= '0';
22     wait for 12.5ns;
23   end process SIMULATION_CLOCK;
24
25   TEST_INPUT: process -- sequential
26   begin
27     A_SIM <= '0'; B_SIM <= '0';
28     wait for 25 ns;
29     A_SIM <= '1'; B_SIM <= '0';
30     wait for 25 ns;
31     A_SIM <= '0'; B_SIM <= '1';
32     wait for 25 ns;
33     A_SIM <= '1'; B_SIM <= '1';
34     wait for 25 ns;
35   end process TEST_INPUT;
36
37   UNIT_UNDER_TEST: entity work.AND_GATE
38   port map(
39     CLK => CLK_SIM,
40     A   => A_SIM,
41     B   => B_SIM,
42     C   => C_SIM
43   );
44
45 end architecture SIMULATION;
```

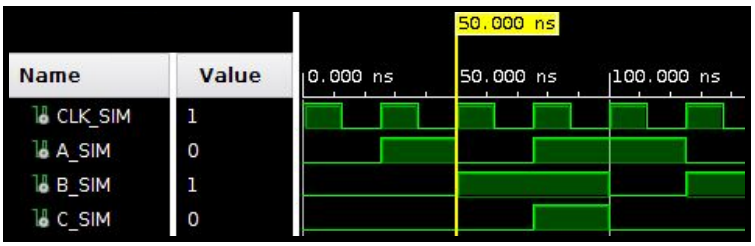
**Listing 5.2:** Test bench to simulate the VHDL code in Listing 5.1.



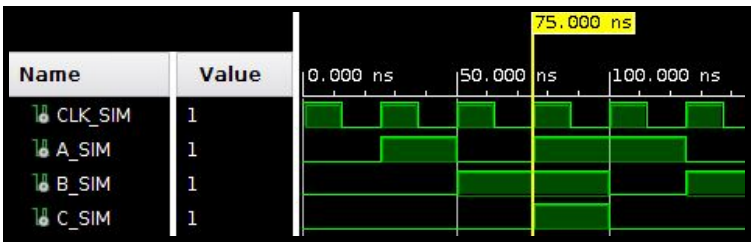

(a)




(b)




(c)




(d)

**Figure 5.3:** Behavioral simulation results of the test bench in Listing 5.2; waveform snapshots after (a) 0 ns, (b) 25 ns, (c) 50 ns and (d) 75 ns, showing each signal name, its current value and propagation over time.

Figure 5.4 shows the synthesized design schematics of the example given above. In comparison to the elaborated design schematics (Figure 5.2), the behavioral (RTL) components have been replaced by physical electronic components, more specifically: one 2-input, 1-output LUT and one D-type FF with constant clock enable and grounded synchronous reset (described in Sec. 4.2.1).

Post-synthesis (structural) simulation can be run to obtain more detailed verification data. It is based on the synthesized design to verify its translation from the behavioral design. Note that there is still no proper timing information at this level.

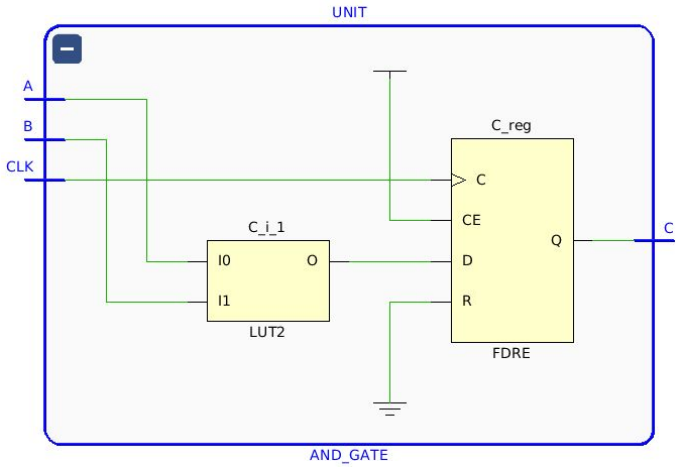



**Figure 5.4:** Synthesized design schematics generated from the elaborated design of the VHDL code in Listing 5.1.

### Implementation, Timing Analysis and Post-Implementation Simulation

Basically, the implementation process performs further design optimization, placement and routing. There are several ways to optimize a given design. For example, to reduce the utilization of components, two small LUTs from the synthesized design could be combined into one large LUT, e.g. a 6-input, 2-output LUT (shown in Fig. 4.2 (b)). Placement and routing are highly complex tasks. During placement, all the synthesized components are allocated to physical hardware components in the target FPGA. The routing tool determines how the interconnections between these components can be routed through the routing network. Ideally, all signal paths should be as short as possible to meet all timing constraints and keep the power consumption low. In reality, local network congestion and path lengths are competing aspects. It is possible to modify the implementation run, e.g. to extend it by further optimization steps.

Figure 5.5 shows a screenshot of the implemented design generated from the synthesized design of the AND gate in Listing 5.1. The target device is a Xilinx US+ type FPGA. The zoom displays a single CLB slice (compare with Fig. 4.2 (a)). Active hardware components and signal lines are highlighted in red and magenta, respectively. The 2-input, 1-output LUT from the synthesized design was mapped to a 6-input, 2-output LUT, with some ports remaining free. Clock enable and synchronous reset of the active FF are driven by constants. It can also be seen that other CLB FFs share the same clock enable, regardless of the fact that they are inactive.

Timing analysis is important to verify the operation of a given design circuit based on the worst-case place and route (logic and net) delays. To make sure that data can propagate reliably, setup and hold time requirements must be satisfied. The setup time is the minimum time during which the data bit should be stable before it is captured in a register. It ensures that the signal propagation from one point to another is not taking more than the required time. The hold time is the minimum time during which the data bit should be stable after it has been captured in a register. It guarantees that the data bit does not change while the register is capturing it.

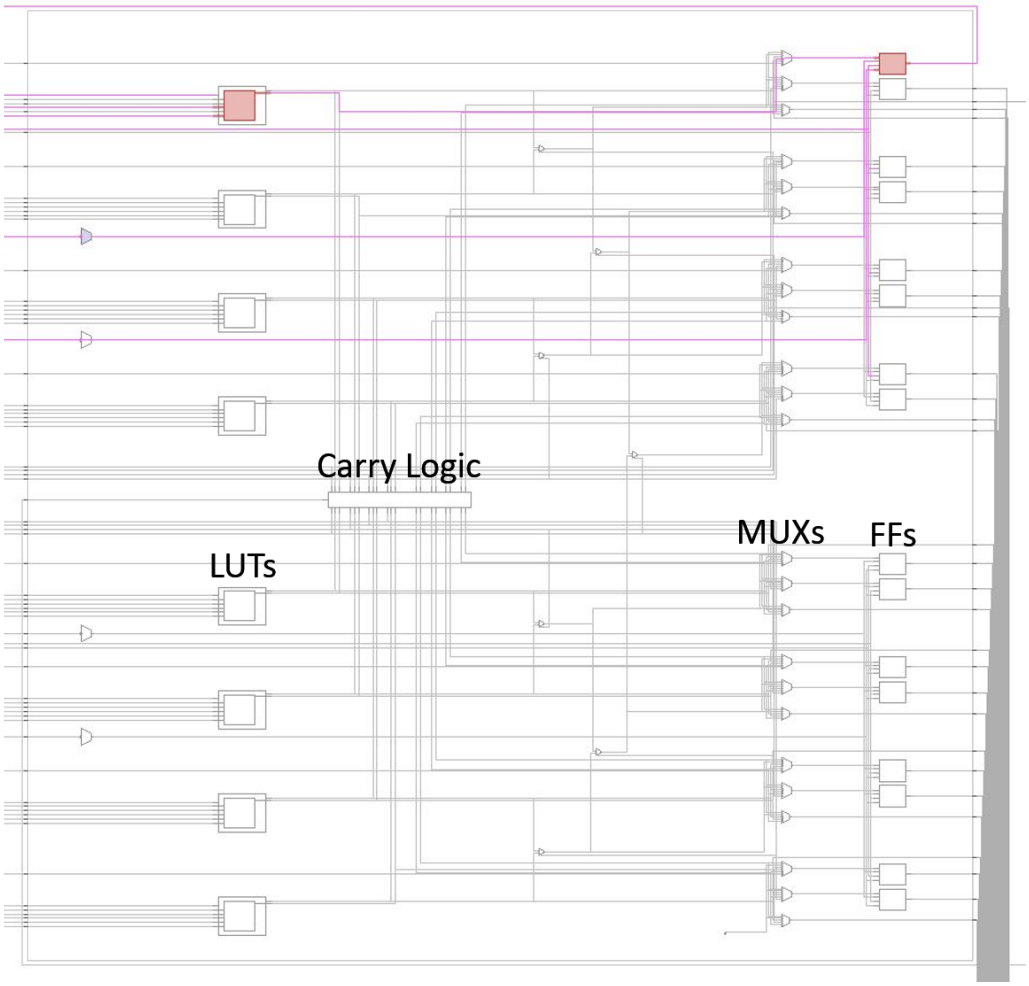



**Figure 5.5:** Implemented design generated from the synthesized design of the VHDL code in Listing 5.1. The screenshot was taken after zooming in on the CLB. The colors have been inverted, especially to improve the visibility of the routing network. The target device is a Xilinx US+ type FPGA (XCVU9P [37]).

Setup and hold time violations can cause errors during the implementation run. All timing violations need to be fixed before moving ahead with the bitstream generation. The resulting configuration file would be of no use unless all timing constraints are met.

Post-implementation (timing) simulation includes proper timing information. It takes longer to run, but provides more details than either the behavioral or structural simulation methods.

### Bitstream Generation

Following the placement and routing, a configuration file is created which specifies how the components of the target FPGA need to be configured and interconnected. After loading the bitstream into the FPGA, the firmware can be debugged and validated.

### 5.2.2 Xilinx IP Cores

Xilinx Intellectual Property (IP) refers to preconfigured logic functions that can be instantiated like VHDL entities. They only need to be configured w.r.t. their design parameters. The IP library is filled with functions (IP cores) ranging in complexity from simple arithmetic operators and delay elements to complex system-level building blocks such as DSP slices and block memory resources. The great advantage of IP cores is the availability of tested and ready-to-use functions that have been optimized for Xilinx FPGAs.
Integrated Logic Analyzer (ILA) IP cores [43], for example, are used to monitor the internal signals of a design. All design clock constraints are also applied to the components inside the ILA core, since it needs to be synchronous to the design being monitored.

### 5.2.3 FPGA Design Characteristics

Utilization, timing and power are important FPGA design characteristics. Reports on timing and power are only available after the implementation step.
The estimated FPGA resource utilization is reported after synthesis, while the final numbers are listed after implementation. Due to potential design optimizations, post-implementation and post-synthesis utilization may differ.
The timing summary report can be used to analyze and debug timing issues. It includes all checks related to setup and hold time requirements. The worst negative slack (WNS) and the worst hold slack (WHS) are important timing characteristics. The *slack* is the margin by which a timing requirement is met (positive slack) or not met (negative slack). In other words, it is the time difference between the expected arrival of a signal and the actual arrival of a signal. The WNS (WHS) corresponds to the worst slack of all timing paths checked during setup (hold) analysis. In addition, the sum of all WNS (WHS) violations and the number of failing endpoints are reported.
The real power consumption and junction temperature[2] can actually be measured during hardware operations only. Both strongly depend on the toggle rate of the bits propagating through the circuit. Therefore, the on-chip power and temperature values estimated by Vivado should be used with caution.

[2]The junction temperature is the highest operating temperature of the actual semiconductor in an electronic device. In operation, it is higher than the case temperature.

# II

# Project Specifications, Algorithm Firmware Development and Trigger Efficiency Studies

# CHAPTER 6

# Jet Feature EXtractor

*Science is simply common sense at its best,*
*that is, rigidly accurate in observation,*
*and merciless to fallacy in logic.*

( Thomas H. Huxley )

As part of the Phase-I upgrade of the Level-1 Calorimeter Trigger in the ATLAS TDAQ system, jFEX has already been introduced in Chapter 3. Together with eFEX and gFEX, it is installed during LS2 and put into operation with the start of Run-3.
This chapter is dedicated to the processing functionality, hardware design and firmware architecture of the jFEX system. The very first section is about the calorimeter trigger input and general data handling.

## 6.1 Processing Functionality

All four processor FPGAs on each of the six jFEX modules are interfaced with incoming and outgoing data fibers via electro-optical transceivers. Data are received from the LAr and Tile calorimeters via LATOME and TREX.
Data handling on the real-time data path includes input data handling, algorithmic data processing and output data formatting. The output is then transmitted to the L1Topo processors. In addition, extended information is sent to the HLT and DAQ system on a separate readout path.

### 6.1.1 Calorimeter Trigger Input

As illustrated in Figure 6.1, each jFEX module receives data from a full $\varphi$ ring, while it has limited access in $\eta$. Furthermore, each processor FPGA on a given module covers a quarter in $\varphi$, denoted as *core region*. To properly identify TOBs at the boundaries, data from neighboring modules/FPGAs are required, resulting in a large amount of data duplication. At both ends of the detector, a larger area in $\Delta\eta \times \Delta\varphi$ can be covered by a single module, since the granularity is reduced.
In the following, the modules 2C, 3C, 3A and 2A are referred to as *central modules*, 1C and 1A as *forward modules* (see Fig. 6.1 for their $\eta$ limits). The term *central* (*forward*) FPGA is used analogously.

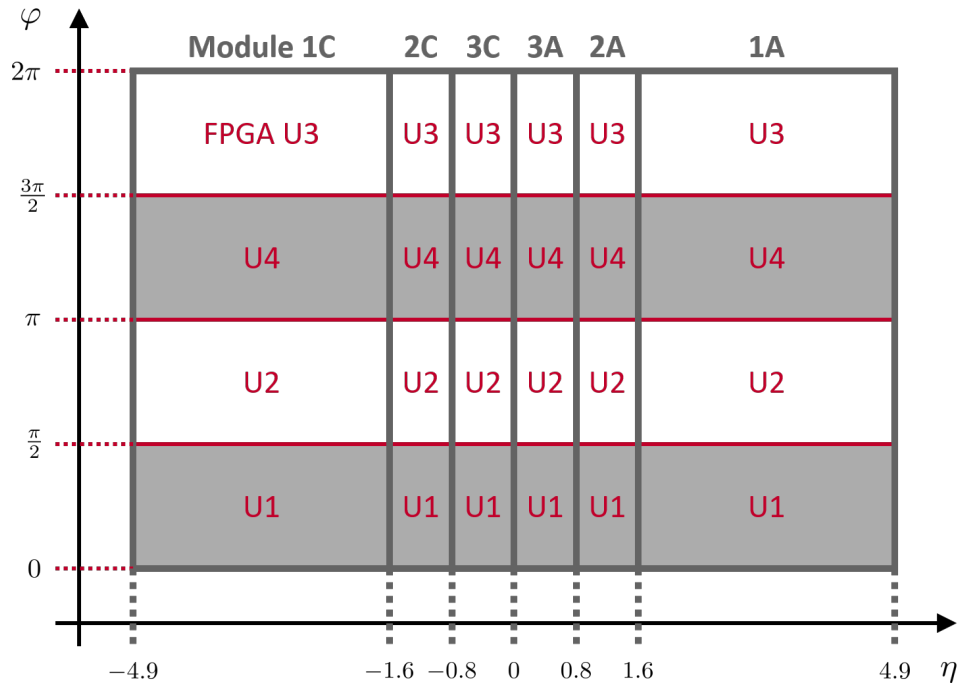
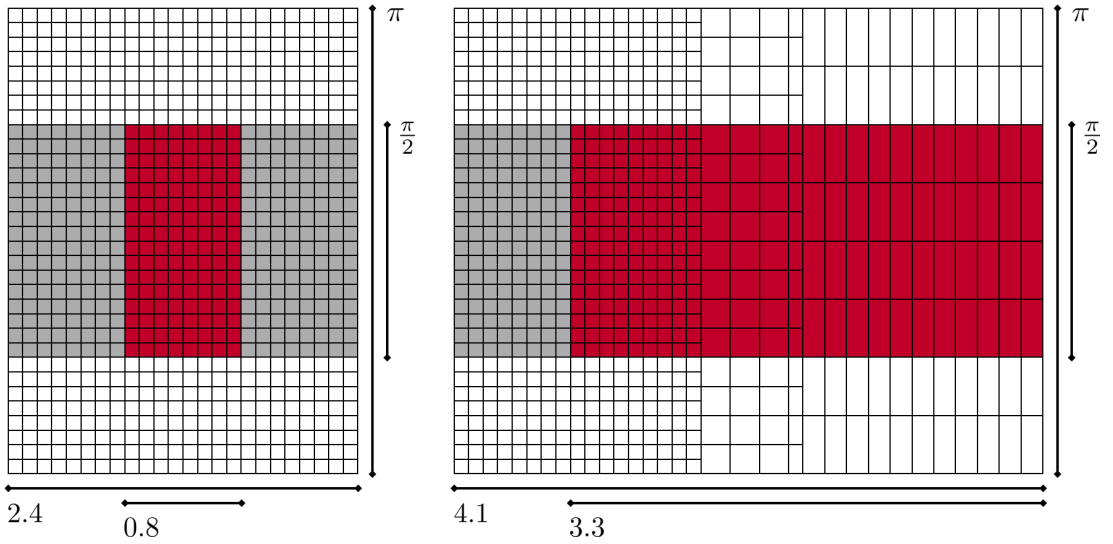



**Figure 6.1:** Calorimeter coverage of the jFEX modules and their processor FPGAs. Note that the boundaries shown here correspond to the core region of each module/FPGA.

Data from the barrel and end-cap calorimeters are based on trigger towers. For $|\eta| < 2.5$, each trigger tower has a size of $\Delta\eta \times \Delta\varphi = 0.1 \times 0.1$. In the region of $2.5 < |\eta| < 3.1$, the size becomes $\Delta\eta \times \Delta\varphi = 0.2 \times 0.2$, while it is $\Delta\eta \times \Delta\varphi = 0.1 \times 0.2$ for $3.1 < |\eta| < 3.2$.
In addition, jFEX receives data based on super cells from each of the three FCal layers separately. While the binning width is about 0.4 in $\Delta\varphi$, it becomes pretty irregular in $\Delta\eta$. The first FCal layer is divided into 12 $\eta$ bins and covers the region $3.1 < |\eta| < 4.9$. The second (third) layer is beyond $|\eta| = 3.2$ and is divided into eight (four) $\eta$ bins.
Figure 6.2 shows the total environment (core + overlap) of a single processor FPGA on a central and a forward module. Since there is no inter-module communication, overlap in $\eta$ and not in $\varphi$ (shown in gray) has to be duplicated before entering the jFEX system. In order to reduce the total input bandwidth, overlap in $\varphi$ (shown in white) is shared on-board.
Since the calorimeter ends at $|\eta| = 4.9$, each forward module/FPGA has to deal with a border region in $\eta$.

### Input from LATOME

LATOME sends 12-bit $E_T$ values with a saturation point at 800 GeV. These energy values are encoded using a multi-linear encoding scheme (see Tab. 6.1). In total, there are 5 linear regions of different resolution (25, 50, 100, 200 and 400 MeV). The first linear region also contains negative values, primarily caused by the bipolar pulse shaping performed to discriminate in-time signals from the out-of-time background in the LAr calorimeter (discussed in Section 3.2.4).




**(a)** central FPGA **(b)** forward FPGA

**Figure 6.2:** Environment of a single processor FPGA on a (a) central and a (b) forward module ($|\eta| \leftrightarrow$, $\varphi \updownarrow$). The core regions are shown in red, overlap to neighboring modules in gray and overlap to neighboring FPGAs on the same module in white. In a simplified version, only the first FCal layer is shown here. Its first $\eta$ bin is located behind the last end-cap bin.

#### Input from TREX

As long as the legacy system is running in parallel, TREX sends 8-bit $E_{\mathrm{T}}$ values with a resolution of 500 MeV and a saturation point at 127.5 GeV. To stick to the LATOME format, TREX actually sends 12-bit data words, where only the least significant bits (LSBs) represent the energy. The most significant bit (MSB) is used to signal invalid data, leaving 2 bits as spare. After the legacy system has been decommissioned and removed, TREX is able to provide 9-bit or 10-bit $E_{\mathrm{T}}$ values with a resolution of 250 MeV, if this will be needed to improve the trigger performance.

### 6.1.2 Real-Time Data Path

Data from LATOME and TREX are received at a line rate of 11.2 Gbit/s, using an 8b/10b encoding[1]. Thus, at a bunch crossing frequency of 40 MHz, each fiber contains $7 \times 32$-bit data words. Up to 16 energy values are distributed over the first six data words. The last data word, referred to as *trailer*, contains an alignment character and a checksum for protection purposes, as well as a part of the bunch crossing ID and a saturation flag per energy value. The latter is only sent by LATOME, indicating if at least one of the underlying super cells was saturated.
Upon reception through the electro-optical devices, the input data are routed and replicated through the physical layer of each MGT. When leaving the MGTs, the data are processed at $7 \times 40\,\mathrm{MHz} = 280\,\mathrm{MHz}$. Data alignment is followed by error handling, including zeroing of potentially corrupted data. Duplicated data from neighboring FPGAs

[1] In telecommunications, 8b/10b is a line code that maps 8-bit words to 10-bit symbols. This encoding supports continuous transmission with a balanced number of zeros and ones (DC balancing).

| Code | Code meaning/ Energy range [MeV] | Resolution [MeV] |
|---|---|---|
| 0 | no data available | - |
| 1 | $< -3{,}150$ | - |
| 2 | $[-3{,}150 \ldots -3{,}125)$ | 25 |
| ... | ... | ... |
| 383 | [6,375 ... 6,400) | 25 |
| 384 | [6,400 ... 6,450) | 50 |
| ... | ... | ... |
| 767 | [25,550 ... 25,600) | 50 |
| 768 | [25,600 ... 25,700) | 100 |
| ... | ... | ... |
| 1535 | [102,300 ... 102,400) | 100 |
| 1536 | [102,400 ... 102,600) | 200 |
| ... | ... | ... |
| 3071 | [409,400 ... 409,600) | 200 |
| 3072 | [409,600 ... 410,000) | 400 |
| ... | ... | ... |
| 4047 | [799,600 ... 800,000) | 400 |
| 4048 | $\geq 800{,}000$ | - |
| 4049 | reserved code | - |
| ... | ... | |
| 4094 | reserved code | |
| 4095 | invalid data | - |

**Table 6.1:** LATOME-to-jFEX energy encoding scheme based on Reference [44].

are received and processed in the same way. Once the locally received and duplicated data are re-timed, they are deserialized and presented to the algorithm block, synchronous to the LHC bunch clock (40 MHz) and within a deterministic latency.

The algorithms have to be implemented as pipelined processors by making use of a maximum parallelization to fit the available latency budget. The algorithm block calculates global variables and identifies TOBs. The results are then serialized for transmission to the L1Topo processors. At a line rate of 12.8 Gbit/s and with an 8b/10b encoding, each fiber contains $8 \times 32$-bit data words, where the first seven data words are filled with TOBs and global information. The higher line rate provides an additional data word for the algorithm results on each of the outgoing fibers. Just like the incoming data stream, the outgoing data stream is protected with an alignment character and a checksum. The trailer also contains a part of the bunch crossing ID and an overflow bit, which indicates whether further TOBs of a type have been found, for which there is no more space on the fiber. A total of four copies of the output are sent to the downstream modules (forward duplication).

### 6.1.3 Readout Data Path

In the case of an L1 accept, RoI and event data are sent to the HLT and DAQ system. This is done on a separate readout path, on which data from three different sources are acquired. RoI data consist of TOBs and extended TOBs (xTOBs), while event data are a full copy of the jFEX input. In contrast to TOBs, the number of xTOBs is not limited. Although TOBs are a subset of xTOBs, both object lists are transmitted in order to monitor what was generally found and what was actually sent to L1Topo.

The readout itself consists of several subsequent stages: buffering, merging and formatting. In the buffering stage, data are stored in BRAMs. Their size is sufficient to wait for an L1 accept delayed by up to 12.8 $\mu$s [45]. Upon L1 accept, the data are retrieved from memories and merged into data streams. Finally, the data streams are protected with checksums and various IDs. Readout communication is enabled via direct links from the processor FPGAs.

Further details on the readout path and in particular on the format of the data streams can be found in Reference [45].

## 6.2 Hardware Design

All the FEXs are based on Advanced Telecommunications Computing Architecture (ATCA) form factor boards [46]. In high-speed interconnect technologies, ATCA has become a standard with a series of specifications, e.g. to limit electromagnetic interference and to limit the spread of fire. An ATCA board is 280 mm deep and 322 mm high.

Each jFEX module consists of an ATCA mainboard and several mezzanine cards[2]. To fully exploit the information from the calorimeters, the choice of hardware components, and especially the FPGA devices, was primarily based on the requirements for input bandwidth and processing power.

### 6.2.1 ATCA Mainboard

The mainboard has a 24-layer stack-up and is composed by MEGTRON6[3], commonly used for signal transmissions above 10 Gbit/s. It is equipped with four processor FPGAs (Xilinx Virtex UltraScale+, XCVU9P [37]) and 24 electro-optical devices (Avago MiniPODs [48]). These MiniPODs are embedded 12-channel optical transmitters/receivers, that operate at up to 14 Gbit/s and enable a dense optical interconnection with low power per Gb.

As illustrated in Figure 6.3, the jFEX design is modular and consists of four identical blocks, each containing one processor FPGA and six MiniPODs (five receivers and one transmitter). The resulting symmetry ensures that the connections between processors and MiniPODs have a constant length. The same applies to the connections between the processors.

[2] A mezzanine card (also mezzanine board or daugtherboard) is an expansion card that usually fits on top of and parallel to a mainboard.

[3] The main attributes of MEGTRON6 are: low dielectric constant and dielectric dissipation factors, low transmission loss and high heat resistance [47].

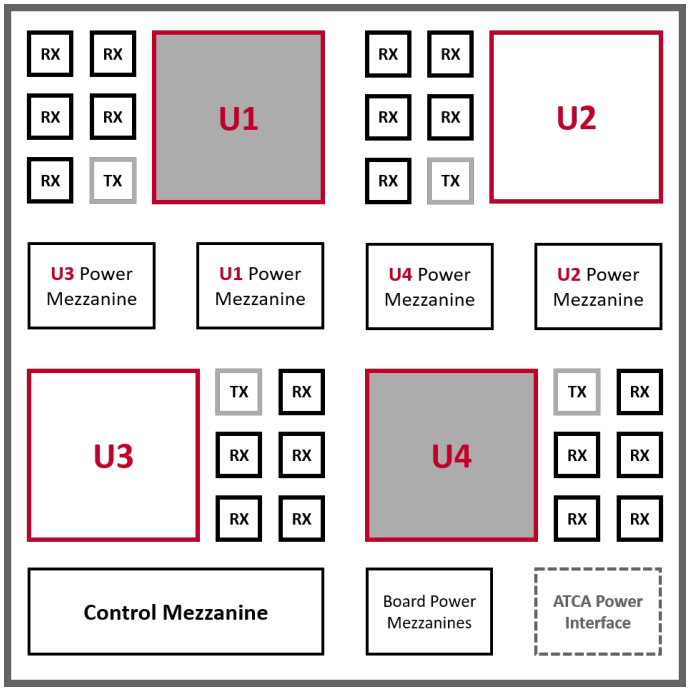



**Figure 6.3:** Schematic design of a jFEX module.

### 6.2.2 Control Mezzanine

To allow a smooth and reliable way of upgrading control functions and components without affecting the mainboard, module control is implemented on a mezzanine card. The control mezzanine is responsible for non-real-time services: board control, clock and configuration circuitry. It also provides initialization circuitry for the processor FPGAs and acts as an interface to environmental monitoring devices. The controller is equipped with a single FPGA (Xilinx Zynq UltraScale+, XCZU7EV [37]). High-level functional control is enabled via IPbus [49], which is a protocol that runs over Ethernet and provides register-level access to the hardware. IPbus requests are forwarded to the processor FPGAs via MGT links.

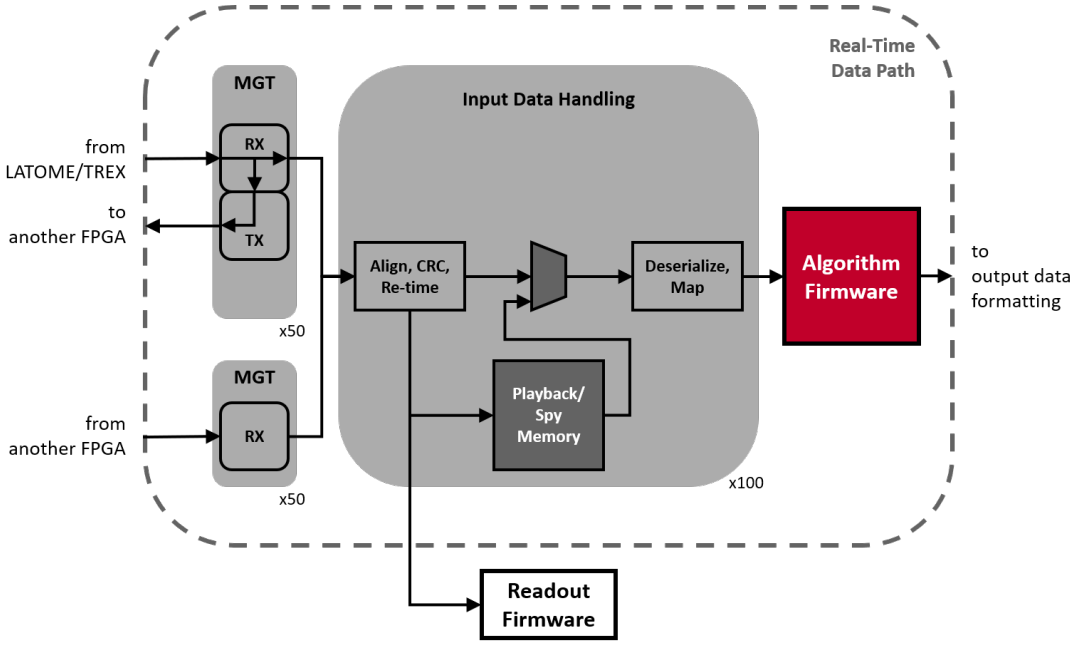
### 6.2.3 Power Mezzanines

For similar reasons, the power supplies are placed on mezzanine cards, too. This also enables a careful assessment of the stability of each mezzanine card before it is connected to the mainboard and powered up.
Each processor FPGA has an independent power mezzanine. In total, each jFEX module hosts four FPGA power mezzanines and three board power mezzanines. For safety reasons, each power mezzanine includes an over-voltage protection circuit. For more details, see Reference [50].

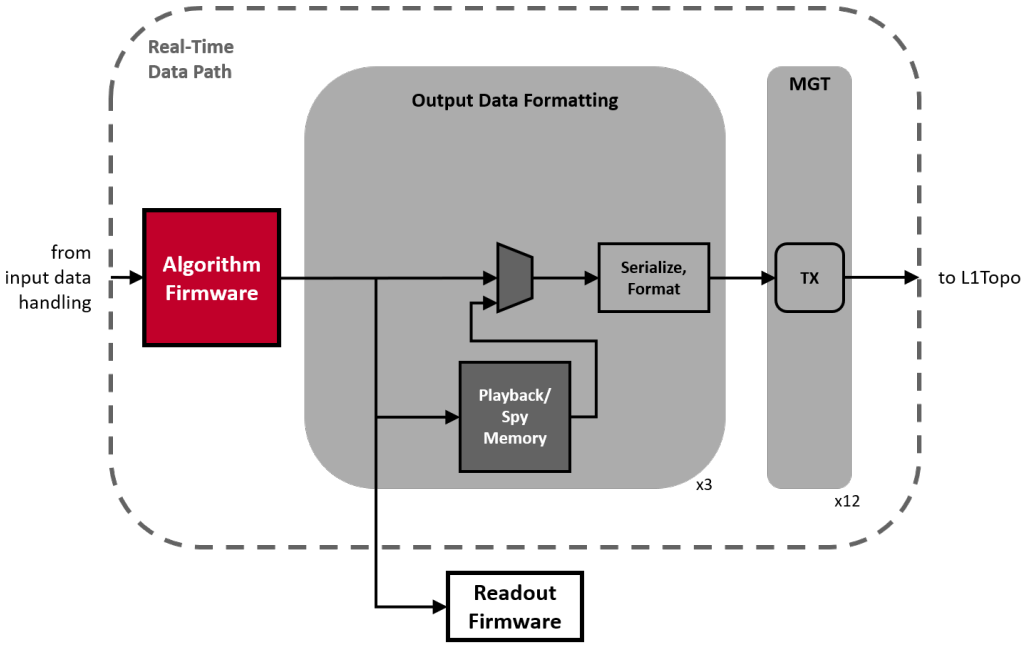
## 6.3 Firmware Architecture

The jFEX firmware has three main components: infrastructure, readout and algorithm firmware. The infrastructure firmware takes care of the overall MGT configurations, input data handling, output data formatting and the entire control circuitry. The latter

includes communication and monitoring via IPbus. A simplified block diagram of the jFEX firmware architecture is shown in Figure 6.4.




(a)




(b)

**Figure 6.4:** Simplified block diagram of the jFEX firmware architecture, showing all the components and the main data paths (a) before and (b) after algorithmic data processing.

### 6.3.1 MGT Firmware

Each processor FPGA needs to share the data it receives directly with its two neighbors. For example, FPGA U2 on module 3A receives data for $-0.8 < \eta < 1.6$ and $\frac{\pi}{2} < \varphi < \pi$ directly from LATOME and TREX, while for the same eta range it receives data for $\frac{\pi}{4} < \varphi < \frac{\pi}{2}$ from U1 and for $\pi < \varphi < \frac{5\pi}{4}$ from U4 (see also Fig. 6.1).
Since passive splitting would affect the signal quality, a feature of the MGTs is used instead: The incoming serial data stream is digitized in the analogue front-end of the receiver and forwarded to the FPGA fabric. In parallel, a connection to the transmitter of the MGT channel allows the received data to be retransmitted to the neighboring processor before it is decoded (Fig. 6.4 (a)).
All MGTs are configured using Xilinx IP cores (introduced in Sec. 5.2.2).

### 6.3.2 Input Data Handling and Output Data Formatting

The incoming data streams are not perfectly aligned in time. Streams from different transmitters may appear to be delayed by a few 280 MHz clock cycles. To compensate for this, individually programmable delays are added for each stream. Another problem is that individual bits can be corrupted by interference during transmission. In order to detect such errors, cyclic redundancy checks (CRCs)[4] are performed.
Once the data streams are aligned, error checked and re-timed, they are fed into a MUX (see Fig. 6.4 (a)). The MUX can be configured via IPbus and forwards either the data streams on the real-time data path or data retrieved from playback memories. The playback memories can also be used in a spy mode, in which input data can be read. Technically, playback/spy memories are implemented in IPbus controlled URAMs.
Before the data are presented to the algorithm block, they are deserialized and mapped into an $\eta/\varphi$ grid. A detailed description of the algorithms and their implementation in firmware follows in the subsequent chapters.

Another MUX is placed right behind the output of the algorithm block to again forward either the data streams on the real-time data path or data retrieved from playback memories (see Fig. 6.4 (b)). A spy mode enables error checking of the outputs to L1Topo.
Finally, the data are serialized, formatted as described above and forwarded to the MGTs.

### 6.3.3 Control Path

The IPbus *master* is located on the control FPGA, while IPbus *slaves* are instantiated on the processors. Each slave is an interface between IPbus and the application logic, i.e. a simple control or status register, or an area of RAM. Connections between IPbus master and slaves are enabled via MGT based IPbus *bridges*. Amongst other things, the IPbus interface allows algorithmic parameters to be set, modes of operation to be controlled, playback memories to be loaded and spy memories to be read via software applications. Coherence between software and firmware view of registers and RAM resources is guaranteed by a common XML[5] description.

[4] A CRC is an error-detecting code. It is simple to implement in binary hardware and easy to analyze mathematically. The CRC procedure is described in Reference [51].

[5] XML (Extensible Markup Language) [52] is a markup language used for the representation of arbitrary data structures in a format that can be read by humans and machines. It is platform independent and language independent.

In addition, each jFEX module monitors its voltages and currents of all power rails, and temperatures of all FPGAs and optical transmitters/receivers. If any of the critical monitoring data exceeds its safe threshold, the board is powered off immediately.

CHAPTER 7

# Feature Extraction Algorithms

*The human imagination,*
*including the creative scientific imagination,*
*can ultimately function only*
*by evoking potential*
*or imagined sense impressions.*

( M. Deutsch )

This chapter describes the jFEX algorithms developed in the context of this thesis. They are designed to cope with the increased event rates delivered by the LHC during Run-3 and beyond by exploiting higher granularity data from the calorimeters. Corresponding trigger efficiencies are presented in Chapter 8, while the implementation strategies for their realization in hardware are then discussed in Chapter 9.
In the following, the hierarchy of the algorithms is presented first. Data that are fed into the algorithm block are then specified in more detail, before the individual algorithms are described.

## 7.1 Hierarchy

The sequence of algorithms from calorimeter data reception via LATOME/TREX to output data transmission to L1Topo is illustrated in Figure 7.1. Initially, trigger tower information from the EM and HAD layers are processed separately from one another:
Individual trigger towers are disabled/masked in case of errors of any kind. The average pile-up is then calculated and subtracted event by event. Afterwards, two different sets of noise cuts are applied to the tower energies used as inputs for particle-like objects (jets, taus, electrons) and global variables ($\Sigma E_{\mathrm{T}}$, $E_{\mathrm{T}}^{\mathrm{miss}}$). Depending on the following algorithm, the data are then either combined (EM + HAD) or forwarded separately (EM, HAD).
For the identification of small-radius (small-R) jets, large-R jets and hadronically decaying taus there is no need to keep EM and HAD towers separate. The same applies to the $\Sigma E_{\mathrm{T}}$ and $E_{\mathrm{T}}^{\mathrm{miss}}$ calculations. For electrons, on the other hand, it is important to consider the EM and HAD layers separately.
TOBs are formed for each object type by collecting calculated quantities, in particular $\eta/\varphi$ coordinates and transverse energy sum(s). For the sake of simplicity, results from $\Sigma E_{\mathrm{T}}$ and $E_{\mathrm{T}}^{\mathrm{miss}}$ calculations are also collected in TOB-like structures. Finally, each list of

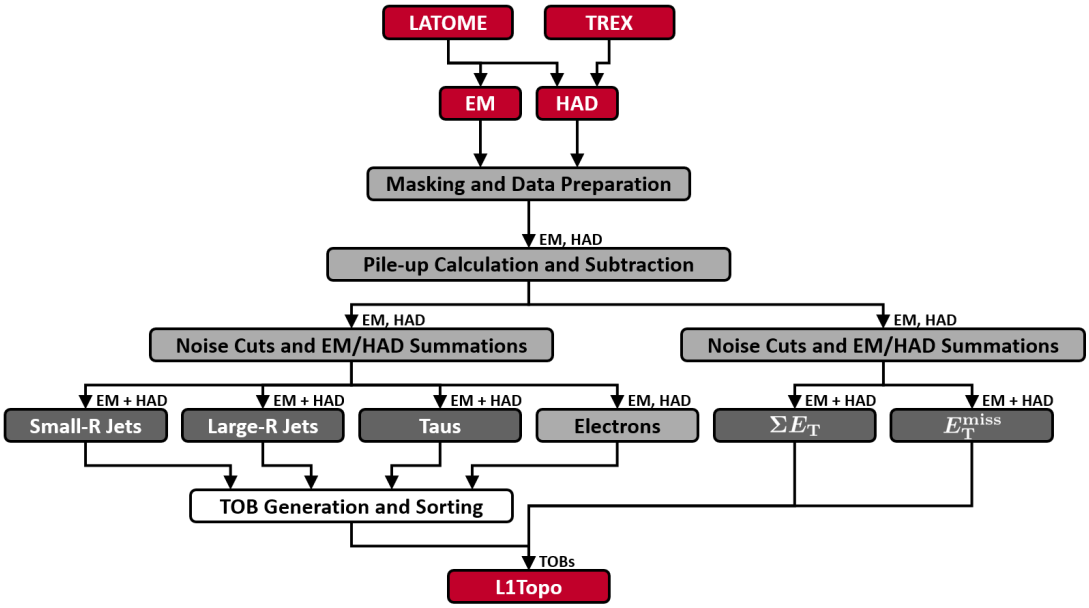



**Figure 7.1:** Simplified block diagram of the jFEX algorithms.

particle-like TOBs is sorted by energy and, together with the global variables, only the most energetic TOBs are then transmitted to L1Topo.

## 7.2 Algorithm Input

EM data are only received by LATOME, while HAD data are provided by TREX for the region $|\eta| < 1.5$ and by LATOME for $|\eta| > 1.5$.
To a good approximation, trigger towers up to $|\eta| = 2.5$ are of the same size in $\Delta\eta \times \Delta\varphi$ and therefore equally spaced. With $|\eta| > 2.5$, the spacing varies and becomes quite irregular in the FCal regions. Depending on the algorithm approach, it is necessary to change from the representation in Figure 6.2 (b) to that in Figure 7.2, where the geometric centers of the individual trigger towers are displayed. The coordinates of the FCal super cells (hereinafter referred to as towers, each consisting of a single super cell) are derived by averaging over the coordinates of the underlying electrodes[1]. These $x$ and $y$ coordinates, which can be found in Reference [53], are converted into $\eta$ and $\varphi$ using Equations 3.3, 3.4 and 3.7. The distance from the center of FCal layer 1/2/3 to the interaction point is 4931.95/5389.15/5871.25 mm (see Tab. 3 in Ref. [54]).
Fortunately, the FCal is symmetric under a rotation of 180° with respect to the beam line and under reflection with respect to the transverse plane. All other coordinates can thus be obtained from Figure 7.2 by a translation in $\varphi$ and/or mirroring in $\eta$.

## 7.3 Functional Descriptions

The order in which the trigger algorithms are described below is based on the data flow shown in Figure 7.1.

[1] For the actual electrode mapping in each of the three FCal layers, see Appendix G.

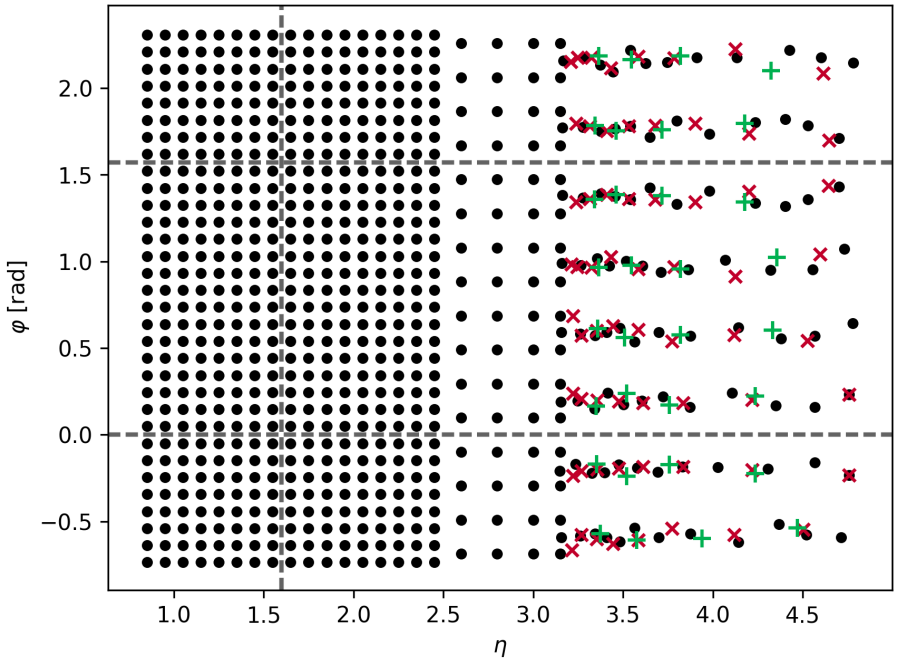



**Figure 7.2:** Environment of FPGA U1 on module 1A. Dashed lines are used to separate core and overlap in $\eta$ and $\varphi$. Each black point represents the geometric center of an EM trigger tower, while each red cross (green plus) corresponds to a tower from FCal layer 2 (layer 3). Outside of the FCal, the EM and HAD tower coordinates are identical.

### 7.3.1 Masking and Data Preparation

Before a trigger tower can actually be used for arithmetic operations, several checks are required: The energy value of a trigger tower is set to zero, if it does not correspond to a physical energy value (see Tab. 6.1) or if an error flag is set (TREX). In such a case, the corresponding saturation flag is also disabled (only affects input from LATOME).

In addition, each trigger tower is checked for saturation, and if this is the case, the resulting flag is ORed with the one received from upstream. Thus, the combined saturation flag indicates whether there is an (upstream) saturation at either tower or super cell level.

Finally, individual towers and corresponding flags are optionally masked. Energy values received from LATOME are then linearized before they are passed on to the pile-up algorithm together with the TREX data.

### 7.3.2 Pile-up Calculation and Subtraction

Numerous soft interactions take place at each bunch crossing. The resulting pile-up leads to additional energy depositions in the calorimeter cells. The aim of an event-by-event pile-up correction on jFEX is a more precise identification of the physically interesting objects and thus a better discrimination from the background.

To get an estimate of the event pile-up, the average energy $\rho$ is calculated per FPGA and in each FPGA separate for the regions EM ($|\eta| < 3.2$), HAD1 ($|\eta| < 1.5$), HAD2

($1.5 < |\eta| < 1.6$), HAD3 ($1.6 < |\eta| < 3.2$) and FCal according to

$$\rho = \frac{\sum_i^N \frac{\left(\tilde{E}_\mathrm{T}\right)_i}{w_i}}{N} \tag{7.1}$$

with

$$\left(\tilde{E}_\mathrm{T}\right)_i = \begin{cases} (E_\mathrm{T})_i & \text{if} \quad E_\mathrm{T}^\mathrm{min} < \frac{(E_\mathrm{T})_i}{w_i} < E_\mathrm{T}^\mathrm{max} \\ 0 & \text{else} \end{cases}, \tag{7.2}$$

where the $(E_\mathrm{T})_i$ correspond to the trigger tower energies and the $w_i$ are tower-specific weights. These weights correspond to the product of tower area and detector response. Only terms that lie within a certain energy range, which is defined by the configurable threshold values $E_\mathrm{T}^\mathrm{min}$ and $E_\mathrm{T}^\mathrm{max}$, are used for the sum.
Due to the irregular shapes of the FCal towers, their areas cannot simply be computed by multiplying $\Delta\eta$ with $\Delta\varphi$. Instead, they were calculated by scanning a very fine grid in the $\eta$-$\varphi$ space, where the center of each grid element was transformed into the $x$-$y$ space and checked which electrode it is closest to. Each grid element was then assigned to the trigger tower associated with this electrode. The results are shown in Figure 7.3.

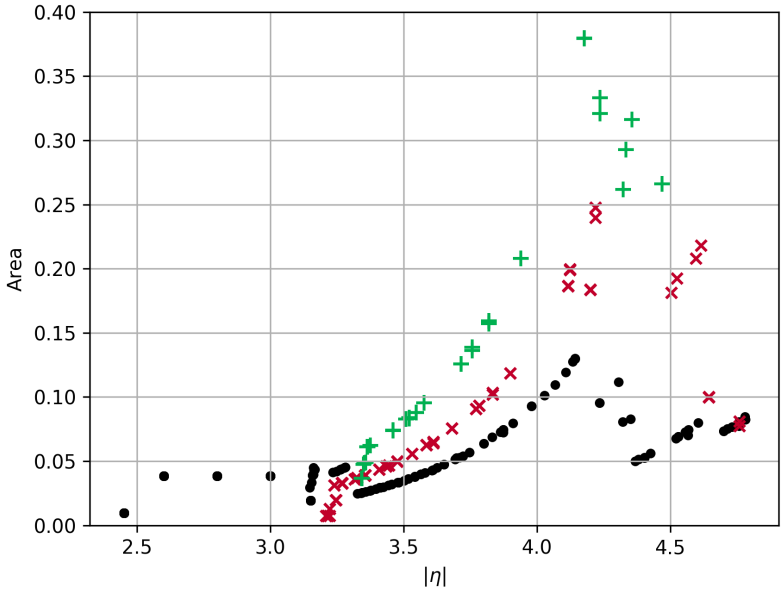



**Figure 7.3:** Trigger tower area in the $\eta$-$\varphi$ space versus $|\eta|$. Each black point represents an EM trigger tower, while each red cross (green plus) corresponds to a tower from FCal layer 2 (layer 3). Outside of the FCal, the EM and HAD tower areas are identical.

The detector response is determined from the average energy deposition in minimum bias events. Initial values were obtained from the dijet simulation sample (JZ0W), but must be refined with minimum bias data. The same applies to the threshold values $E_\mathrm{T}^\mathrm{min}$ and $E_\mathrm{T}^\mathrm{max}$ in Equation 7.2, which have been tuned w.r.t. the $E_\mathrm{T}^\mathrm{miss}$ trigger performance (presented in Chap. 8). Currently, the response of EM towers is not taken into account as it has not

improved the overall performance. All the weights are normalized to a central-like EM tower with $\Delta\eta = 0.1$ and $\Delta\varphi = 0.1$.
$\rho$ is then rescaled by the same weight before it is subtracted from each trigger tower so that the corrected $E_{\mathrm{T}}$ is given by

$$\left(E_{\mathrm{T}}^{\mathrm{corrected}}\right)_i = (E_{\mathrm{T}})_i - \rho \cdot w_i. \tag{7.3}$$

To gain flexibility, it is possible to transfer either the pile-up subtracted or non-subtracted data to the noise cut stages.

### 7.3.3 Noise Cuts and EM/HAD Summations

As already mentioned, there are two different sets of noise cuts. They are applied individually to each EM and HAD trigger tower energy. Fixed noise cuts derived from Run-2 performance studies are currently in use, but must be retuned according to the Run-3 conditions and later refined with real data. To stay flexible, the noise cuts can also be turned off by simply setting each threshold to its minimum value.

Before EM and HAD towers can be combined, data from TREX are brought into the LATOME format (25 MeV per LSB). Since there is no alignment among the FCal layers in $\eta$ (and only approximately $\Delta\varphi = 0.4$), their EM and HAD towers are kept separate and processed differently depending on the following algorithm.

### 7.3.4 Global Variables

Including the overlap to neighboring FPGAs, U1 and U4 together have full access in $\varphi$ (see Figs. 6.1 and 6.2). Since this applies to each of the modules, global variables are only processed and transmitted by half of the jFEX FPGAs in order to reduce the total output bandwidth. The overlap in $\eta$ is not required for either $\Sigma E_{\mathrm{T}}$ or $E_{\mathrm{T}}^{\mathrm{miss}}$ calculations.

#### Total Transverse Energy Sum

The $\Sigma E_{\mathrm{T}}$ algorithm computes the scalar sum of the transverse energy in all calorimeter towers. However, since there is no inter-FPGA communication, only partial sums are formed, which requires further processing on L1Topo.
jFEX provides two values of $\Sigma E_{\mathrm{T}}$ for configurable consecutive $\eta$ regions to allow more flexibility for later trigger decisions. The $\eta$ boundary can be changed according to the available granularity, with the exception of the FCal, which is treated as a whole.

#### Missing Transverse Energy

For jFEX, Equation 3.18 turns into

$$\begin{pmatrix} E_x^{\mathrm{miss}} \\ E_y^{\mathrm{miss}} \end{pmatrix} = -\sum_{i \in \mathrm{tower}} \begin{pmatrix} E_{\mathrm{T}} \cdot \cos\varphi \\ E_{\mathrm{T}} \cdot \sin\varphi \end{pmatrix}_i, \tag{7.4}$$

where the sum is no longer over jet elements, but over a four times finer segmentation in $\eta \times \varphi$. Like for $\Sigma E_{\mathrm{T}}$, jFEX can only produce partial sums for $E_x^{\mathrm{miss}}$ and $E_y^{\mathrm{miss}}$. In the FCal, constant values for $\varphi$ are assumed in steps of 0.4.

#### Global TOBs

Results from $\Sigma E_{\mathrm{T}}$ and $E_{\mathrm{T}}^{\mathrm{miss}}$ calculations are collected and stored in separate TOBs, the compositions of which are listed in Table 7.1. To stick to the format of the outgoing data stream (discussed in Sec. 6.1.2), each TOB has a fixed length of 32 bits.

| | Label | Bits | Description |
|---|---|---|---|
| $\boldsymbol{\Sigma E_{\mathrm{T}}}$**:** | $E_{\mathrm{T}}$ 1 | 15 | Transverse energy (lower $\lvert\eta\rvert$ range) |
| | Sat. 1 | 1 | Saturation (jFEX input) |
| | $E_{\mathrm{T}}$ 2 | 15 | Transverse energy (higher $\lvert\eta\rvert$ range) |
| | Sat. 2 | 1 | Saturation (jFEX input) |
| $\boldsymbol{E_{\mathrm{T}}^{\mathrm{miss}}}$**:** | $E_x^{\mathrm{miss}}$ | 15 | Energy (x-component) |
| | Sat. | 1 | Saturation (jFEX input) |
| | $E_y^{\mathrm{miss}}$ | 15 | Energy (y-component) |
| | Res. | 1 | Reserved for future use |

**Table 7.1:** jFEX global TOB formats. All energies have to be interpreted as signed values with a resolution of 200 MeV per LSB.

Each energy value is based on the sum of multiple trigger towers and is therefore provided with its own saturation flag, which is determined by ORing all the underlying flags. Since $E_x^{\mathrm{miss}}$ and $E_y^{\mathrm{miss}}$ are composed of the same towers, the overflow does not have to be signaled for each component. As a result, one bit is left for future use.

### 7.3.5 Small- and Large-R Jets

In contrast to the processing of global variables, the jet algorithms also require the overlap in $\eta$. In Reference [55] it was shown that in terms of resources and latency, an implementation of the anti-$k_{\mathrm{T}}$ algorithm[2] on jFEX is impossible. Instead, the jFEX jet algorithms are based on the sliding window approach, which can be highly parallelized and has already been used in the JEMs. However, since jFEX has access to a finer granularity, there are some differences to the JEM algorithm (described in Sec. 3.3.1):

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

$$\text{cond.} = \begin{cases} \geq & \text{if} \quad (\Delta\eta + \Delta\varphi < 0) \ \vee \ [\,(\Delta\eta + \Delta\varphi = 0) \ \wedge \ (\Delta\eta < 0)\,] \\ > & \text{else} \end{cases}, \tag{7.5}$$

where $\Delta\varphi$ and $\Delta\eta$ are calculated w.r.t. $\varphi_{\mathrm{RoI}}$ and $\eta_{\mathrm{RoI}}$, respectively.

[2] Briefly described in Section 3.5.1.

Illustrations are given in Figures 7.4 (a) and (b): Only the geometric center of a given trigger tower defines whether it is part of a sum. On central FPGAs, each seed (search window) has a fixed size of $0.3 \times 0.3$ ($0.5 \times 0.5$) in $\Delta\eta \times \Delta\varphi$.
Figure 7.5 shows the situation in the high $|\eta|$ regions, where the shapes of the seed and search windows become irregular. Using radii to define each window enables a smooth transition across regions with different granularity. Note that for the values of the radii a multiple of $\pi/32$ is used to be consistent with the shapes in Figure 7.4.
In the FCal, only the geometric centers of the trigger towers in the first layer are considered as RoI candidates, since it has the finest granularity. Towers from all layers are taken into account when calculating energy sums.

The mixture of 'greater than' and 'greater than or equal' conditions, in combination with the fact that neighboring seeds overlap, leads to a shift of the local maximum in the case of a well-isolated energy deposition in a single trigger tower: Assuming an energy greater than zero at $(\eta, \varphi) = (i, j)$ and zero energy around it (e.g. due to noise cuts), the seeding procedure identifies a local maximum at $(i + 1, j + 1)$, and thus at a shifted position compared to the expected one (+1 in $\eta$ and +1 in $\varphi$ for $|\eta| < 2.4$; in different directions beyond). This is a constant false identification that is corrected by performing an additional comparison of the two trigger towers at the expected and shifted positions.

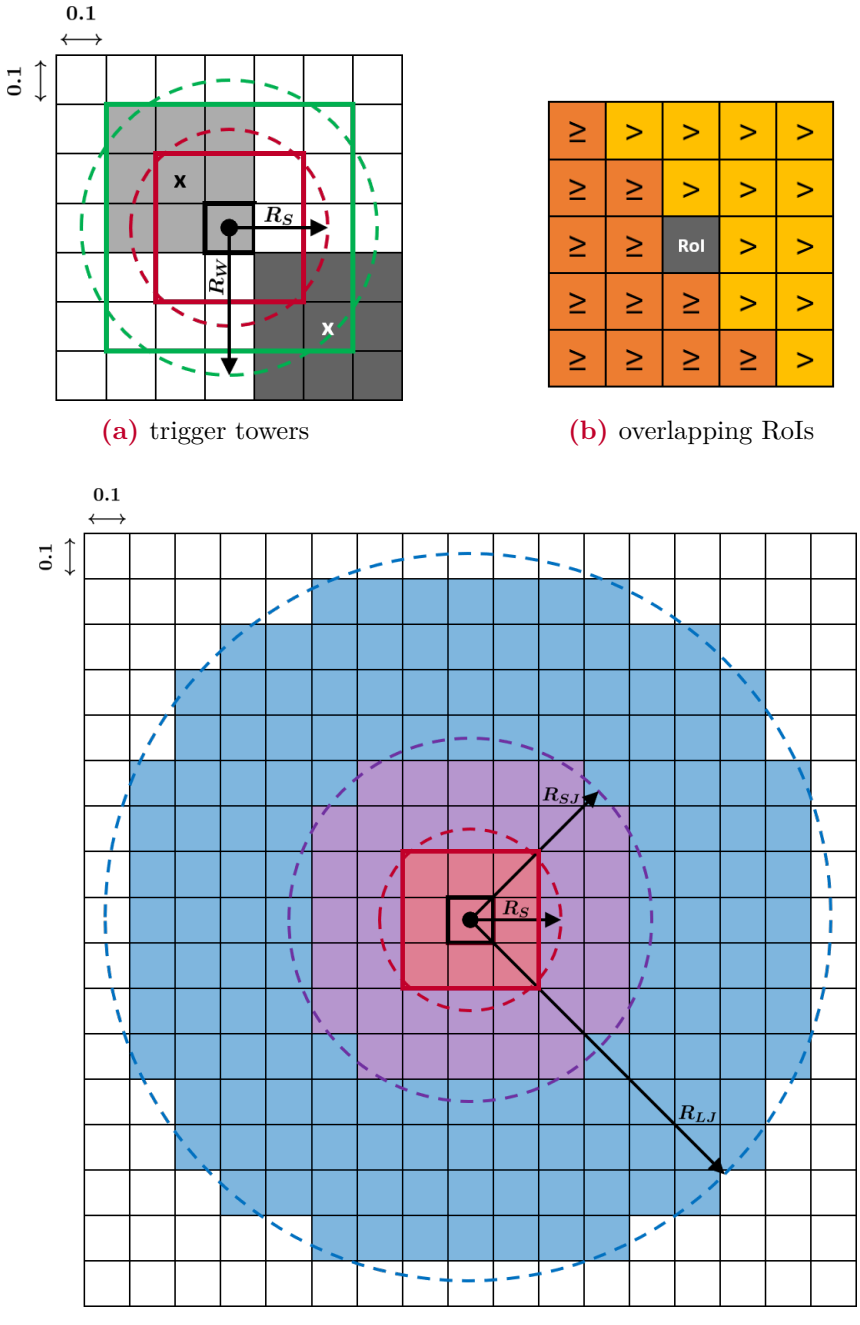
**Jet Clustering** jFEX calculates jet cluster energies at small and large radii: For small-R jet candidates this is the sum of all trigger towers within $\Delta R < 0.4$, while it is $\Delta R < 0.8$ for large-R jets. The resulting jet windows are shown in Figures 7.4 (c) and 7.5 (b).

### 7.3.6 Taus

The overlap between eFEX processor FPGAs is only 0.1 in $\eta$ as well as in $\varphi$ [56], while for jFEX it is 0.8 in both directions. Thus, jFEX has access to a larger window around the tau candidates. For this reason, additional information about the isolation is calculated and sent to L1Topo, where it can be combined with eFEX taus. Appropriate studies are currently ongoing to determine a possible reduction in the trigger rate.
Like eFEX, jFEX identifies taus up to $|\eta| = 2.5$, since beyond that there is no tracking information available for further processing in the HLT.

**Tau Seeding** To achieve a proper matching of eFEX and jFEX taus on L1Topo, both algorithms must be based on the same seeding procedure. eFEX is using a seeding based on $\Delta\eta \times \Delta\varphi = 0.1 \times 0.1$ trigger towers (EM + HAD) [56], which can also be implemented in jFEX:
A single trigger tower is tested for a local maximum by comparing its energy with the energies of all eight adjacent towers using a mixture of 'greater than' and 'greater than or equal' conditions according to Equation 7.5 (see also Fig. 7.4 (b)). For $|\eta| < 2.4$, the resulting search window has a fixed size of $0.3 \times 0.3$ in $\Delta\eta \times \Delta\varphi$ ($R_W = 0.2$), while it becomes slightly different for $2.4 < |\eta| < 2.5$.

**Tau Clustering and Isolation** The tau cluster energy is formed by summing all trigger towers within $\Delta R < 0.2$, while the sum of all towers within $0.2 \leq \Delta R < 0.4$ is defined as the isolation energy (see Fig. 7.4 (c)). Both $E_{\mathrm{T}}$ values are sent to L1Topo.

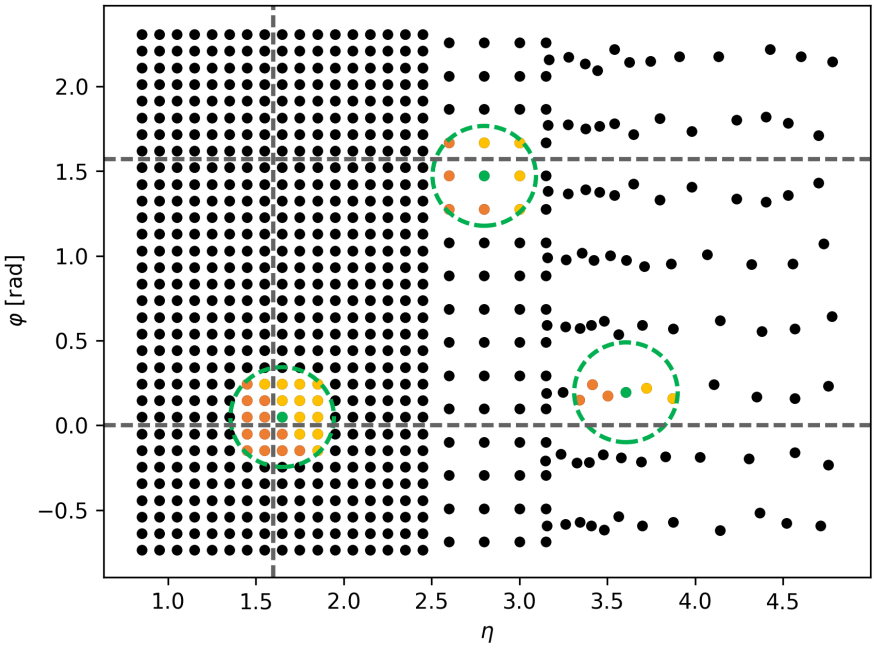



**(a)** trigger towers

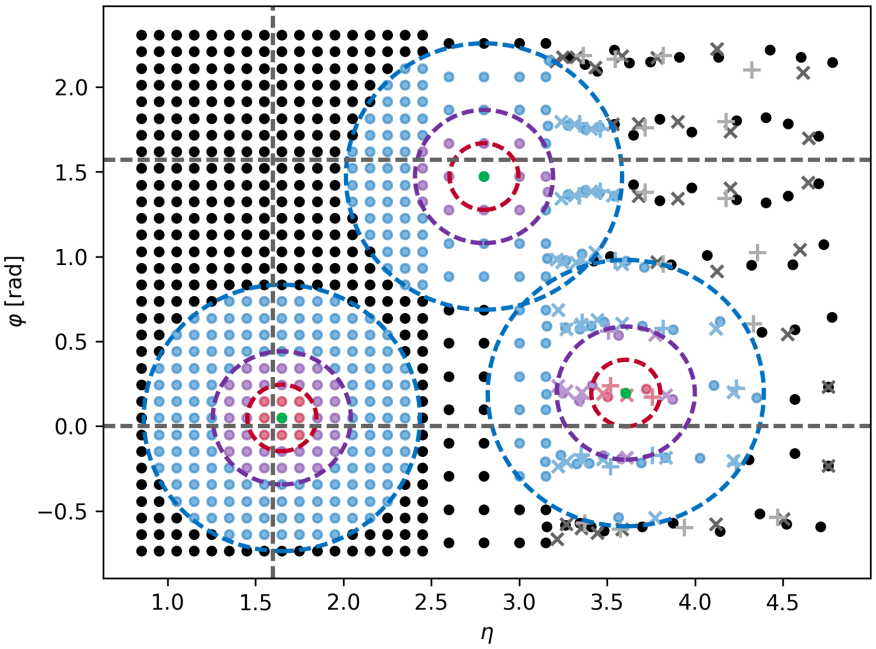
**(b)** overlapping RoIs

**(c)** trigger towers

**Figure 7.4:** Seeding and clustering for the jFEX jet algorithms on central FPGAs: (a) central seed colored in red ($R_S = 0.2$) with associated search window in green ($R_W = 0.3$) and two out of 24 neighboring (and overlapping) seeds in different shades of gray; (b) local maximum test ($+\eta \rightarrow$, $+\varphi \uparrow$): 12 'greater than' conditions and 12 'greater than or equal' conditions according to Equation 7.5; (c) seed window colored in red, small-R jet window in violet ($R_{SJ} = 0.4$) and large-R jet window in blue ($R_{LJ} = 0.8$).




**(a)** overlapping RoIs




**(b)** trigger towers

**Figure 7.5:** Seeding and clustering for the jFEX jet algorithms on FPGA U1 of module 1A: (a) search windows at $\eta = 1.65$ (central-like), $\eta = 2.8$ (transition region) and roughly $\eta = 3.6$ (FCal) with highlighted condition operators according to Equation 7.5; (b) for the same $\eta$ coordinates: seed windows colored in red, small-R jet windows in violet and large-R jet windows in blue. Each point represents the geometric center of an EM trigger tower, while each cross (plus) corresponds to a tower from FCal layer 2 (layer 3).

### 7.3.7 Electrons

eFEX does not search for electrons beyond $|\eta| = 2.5$ [56]. jFEX is able to extend electron identification up to $|\eta| = 4.9$. To avoid inefficiencies in the transition regions (e.g. for a central-forward dielectron trigger), jFEX searches for electrons beyond $|\eta| = 2.3$ instead of $|\eta| = 2.5$. Double counting on L1Topo is avoided by setting $\eta$ limits depending on the target trigger. Corresponding studies are currently ongoing.
Since the first $\eta$ bin of FCal layer 1 (EM layer) is located behind EMEC/HEC (see Fig. 3.9), it is neither considered in the seeding nor in the clustering process.

**Electron Seeding** Analogous to taus, a single trigger tower is tested for a local maximum using a mixture of 'greater than' and 'greater than or equal' conditions according to Equation 7.5 (see also Fig. 7.4 (b)). Though, only towers from the EM layer are taken into account. In the regions $2.3 < |\eta| < 2.5$, a search window of $\Delta R < 0.2$ is used, while it is $\Delta R < 0.3$ for $|\eta| > 2.5$. The larger window in the high $|\eta|$ regions ensures that there are always neighbors to compare with. To give an example, there would be no neighbor within $\Delta R < 0.2$ around an RoI candidate at $|\eta| = 2.8$ (see Fig. 7.5).

**Electron Clustering** The electron cluster energy is the sum of the EM trigger tower at the position of the RoI candidate and its most energetic direct $\eta/\varphi$ neighbor, which prevents a large part of the cluster energy from being excluded or background energy from being added. The trigger towers considered to be direct $\eta/\varphi$ neighbors are illustrated in Figure 7.6. Due to the large $\varphi$ binning in the FCal regions, only $\eta$ neighbors are taken into account for $|\eta| > 3.2$.

**Electron Identification Variables** The eFEX electron identification variables (EM isolation, HAD veto, cluster width) [56] are based on a finer granularity than jFEX receives. Therefore, a modified set of variables was developed for jFEX:

**EM Isolation** The electron EM isolation is defined as the sum of all EM trigger towers within $\Delta R < 0.4$, but without the cluster. In order to be able to process the electrons on L1Topo in a similar way, jFEX uses an eFEX-like thresholding according to

$$1 - \frac{E_{\mathrm{T}}(\mathrm{Cluster})}{E_{\mathrm{T}}(\mathrm{Cluster}) + E_{\mathrm{T}}(\mathrm{EM\ isolation})} < C_i, \tag{7.6}$$

where the $C_i$ are a set of three constants, referring to a "loose", "medium" and "tight" isolation.

**EM Fraction** There are two different versions of the EM fraction variable: Version A ignores data from the last FCal layer, while data from the second layer are excluded for version B. The latter is only defined for $|\eta| > 3.3$.
The definition of EM fraction is similar to that of EM isolation. Generally, it is

$$\frac{E_{\mathrm{T}}(\mathrm{HAD})}{E_{\mathrm{T}}(\mathrm{EM}) + E_{\mathrm{T}}(\mathrm{HAD})} < C_f, \tag{7.7}$$

where $E_{\mathrm{T}}(\mathrm{EM})$ is associated with the energy of the EM trigger tower at the position of the RoI candidate, while for $E_{\mathrm{T}}(\mathrm{HAD})$ the closest HAD tower in $\Delta R$ is taken. As with the EM isolation, a set of three constants is provided for each version.

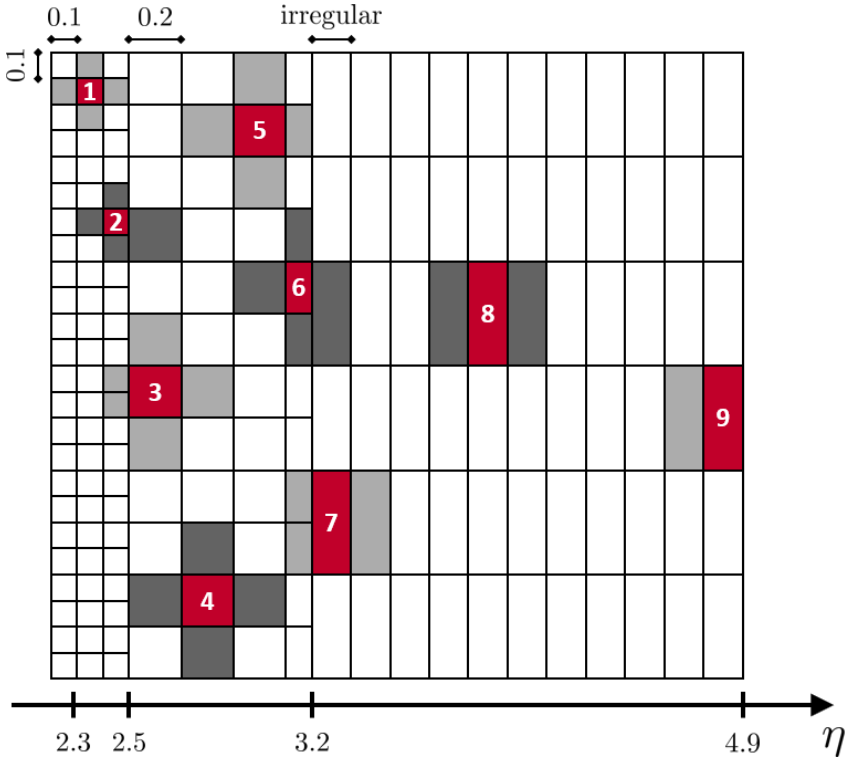



**Figure 7.6:** Different sets of direct $\eta/\varphi$ neighbors along $\eta$ ($\varphi$ ↕): 9 different cases are shown, where the EM trigger tower at the position of the RoI candidate is colored in red, while the appropriate $\eta/\varphi$ neighbors are highlighted in gray. The different shades of gray are only used for a better differentiation.

The trigger tower areas in the first $\eta$ bin of FCal layer 2 are rather small (see Fig. 7.3)[3]. As a consequence, the ratio between the tower areas of $E_{\mathrm{T}}$(EM) and $E_{\mathrm{T}}$(HAD) is greater than 1 for each RoI candidate in the second $\eta$ bin of FCal layer 1. To keep this ratio less than or equal to 1, the second next HAD tower in $\Delta R$ is added to the closest one. For an RoI candidate at $\varphi \approx 1.0$ (4.2), even the third next tower has to be added.

### 7.3.8 TOB Generation and Sorting

The TOB formats for small-R jets, large-R jets, taus and electrons are listed in Table 7.2. As with the global TOBs (Tab. 7.1), each particle-like TOB has a fixed length of 32 bits.

For each local maximum the corresponding TOB is filled with all the calculated quantities, generally local $\eta/\varphi$ coordinates and transverse energy of the object cluster. A saturation flag indicates whether there is an upstream saturation on at least one trigger tower (or one super cell) that make up the cluster. Some bits are unassigned and therefore remain for future use.

Local TOB sorting is performed w.r.t. the cluster energy and separate for each object type. xTOBs[4] are not sorted in order to save FPGA resources and latency.

[3] Also visible in Figure G.2 in the appendix.
[4] Already introduced in Section 6.1.3.

| | Label | Bits | Description |
|---|---|---|---|
| **Small-R Jet:** | $\eta$ | 5 | Local $\eta$ coordinate (FPGA core region) |
| | $\varphi$ | 4 | Local $\varphi$ coordinate (FPGA core region) |
| | $E_{\mathrm{T}}$ | 11 | Transverse energy |
| | Res. | 11 | Reserved for future use |
| | Sat. | 1 | Saturation (jFEX input) |
| **Large-R Jet:** | $\eta$ | 5 | Local $\eta$ coordinate (FPGA core region) |
| | $\varphi$ | 4 | Local $\varphi$ coordinate (FPGA core region) |
| | $E_{\mathrm{T}}$ | 13 | Transverse energy |
| | Res. | 9 | Reserved for future use |
| | Sat. | 1 | Saturation (jFEX input) |
| **Tau:** | $\eta$ | 5 | Local $\eta$ coordinate (FPGA core region) |
| | $\varphi$ | 4 | Local $\varphi$ coordinate (FPGA core region) |
| | $E_{\mathrm{T}}$ | 11 | Transverse energy |
| | Iso. | 11 | Transverse isolation |
| | Sat. | 1 | Saturation (jFEX input) |
| **Electron:** | $\eta$ | 5 | Local $\eta$ coordinate (FPGA core region) |
| | $\varphi$ | 4 | Local $\varphi$ coordinate (FPGA core region) |
| | $E_{\mathrm{T}}$ | 11 | Transverse energy |
| | Iso. | 2 | EM isolation |
| | Frac. A | 2 | EM fraction (version A) |
| | Frac. B | 2 | EM fraction (version B) |
| | Res. | 5 | Reserved for future use |
| | Sat. | 1 | Saturation (jFEX input) |

**Table 7.2:** jFEX particle-like TOB formats. All energies have to be interpreted as unsigned values with a resolution of 200 MeV per LSB.

In contrast to a sorting algorithm that is executed on a CPU, the data on an FPGA must be sorted in a fixed time. The implementation of such an algorithm therefore differs greatly from common sorting algorithms such as *quick sort* or *merge sort*, where the time required for sorting varies depending on how many inputs are already in the correct order. Chapter 9 contains a detailed description of the TOB sorting algorithm and how it is implemented on the various jFEX processor FPGAs.

CHAPTER 8

# Performance of Feature Extraction Algorithms

*It is fundamentally the confusion*
*between effectiveness and efficiency*
*that stands between doing the right things*
*and doing things right.*

( Peter F. Drucker )

With the jFEX algorithms described in the previous Chapter, the design concept of new L1 trigger algorithms was introduced. In this chapter, the trigger efficiencies for the newly developed $E_{\mathrm{T}}^{\mathrm{miss}}$ and jet algorithms are presented in direct comparison to the legacy system. These are the relevant signatures for an invisibly decaying Higgs boson that is produced in association with a hadronically decaying $W$ or $Z$ boson. In the case of a highly boosted $W$ or $Z$ boson, the resulting jets can be close-by or even merge into one large jet.
First, there is a general introduction to trigger rates and rate-matched efficiencies. Results of the Higgs-to-invisible trigger efficiency studies are then summarized and discussed in Chapter 10.

## 8.1 Rates and Rate-Matched Efficiencies

The maximum allowed trigger rate is determined by the DAQ bandwidth divided by the event size. Taking into account all correlations, the sum of all L1 trigger rates must therefore not exceed the overall L1 trigger rate. For a correct comparison of different L1 trigger algorithms, it is necessary to adjust the underlying trigger thresholds to an equal rate. To a good approximation, the individual trigger rates/thresholds can be derived from a low-$p_{\mathrm{T}}$ minimum bias sample (JZ0W). For all the rate histograms in this thesis, the vertical axis is transformed into the L1 trigger rate using the scaling factor

$$s_f = \frac{40\,\mathrm{MHz}}{N_{\mathrm{MB}}} \cdot \frac{B_{\mathrm{filled}}}{B_{\mathrm{max}}}, \tag{8.1}$$

where $N_{\mathrm{MB}}$ is the total number of minimum bias events, $B_{\mathrm{filled}}$ represents the number of filled bunches during Run-2 (2556 [57]) and $B_{\mathrm{max}}$ defines the maximum number of bunch positions (3564).

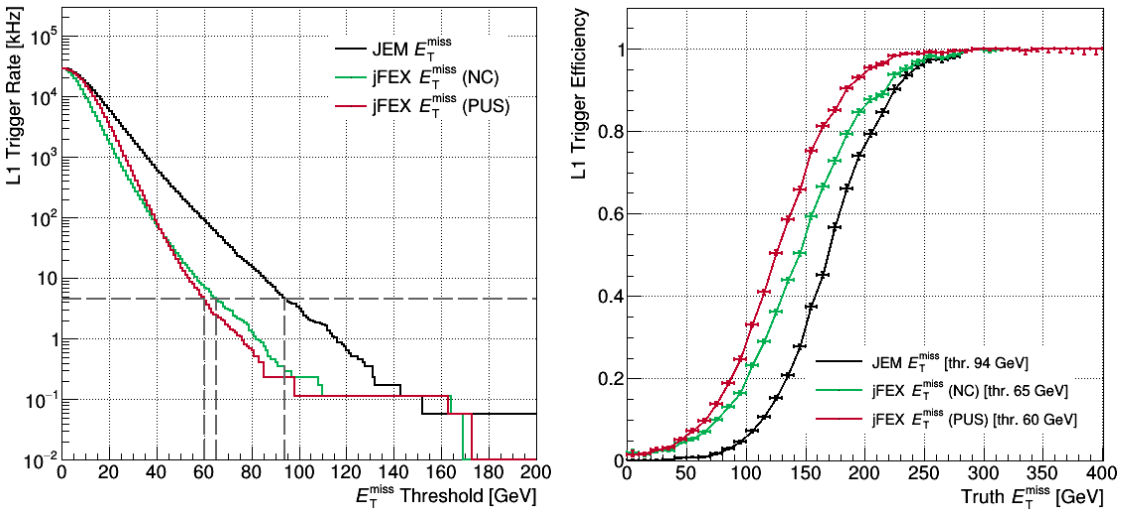
Trigger efficiencies are computed with respect to a reference value that can be e.g. an offline reconstructed observable or based on truth information. The efficiency is defined as the ratio between the number of events accepted (triggered) and the total number of events produced. The resulting curves, commonly referred to as turn-on curves, are more like an error function than a step function, which is mainly due to the detector resolution and differences in the reconstruction algorithms (trigger vs. offline/truth).

In general, the goal is to maximize trigger acceptance by keeping trigger thresholds lower than any conceivable (offline) analysis cut, while keeping rates at a manageable level.

## 8.2 Performance of Level-1 $E_{\mathrm{T}}^{\mathrm{miss}}$ Algorithms

Figure 8.1 shows the performance of the jFEX $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm, once with upstream noise cuts (NC) and once with upstream pile-up subtraction (PUS), in direct comparison to the version that was running on the JEM processor FPGAs during Run-2:

The L1 trigger rates at different $E_{\mathrm{T}}^{\mathrm{miss}}$ thresholds are shown in Figure 8.1 (a), while Figure 8.1 (b) reveals the rate-matched efficiencies with respect to truth $E_{\mathrm{T}}^{\mathrm{miss}}$.



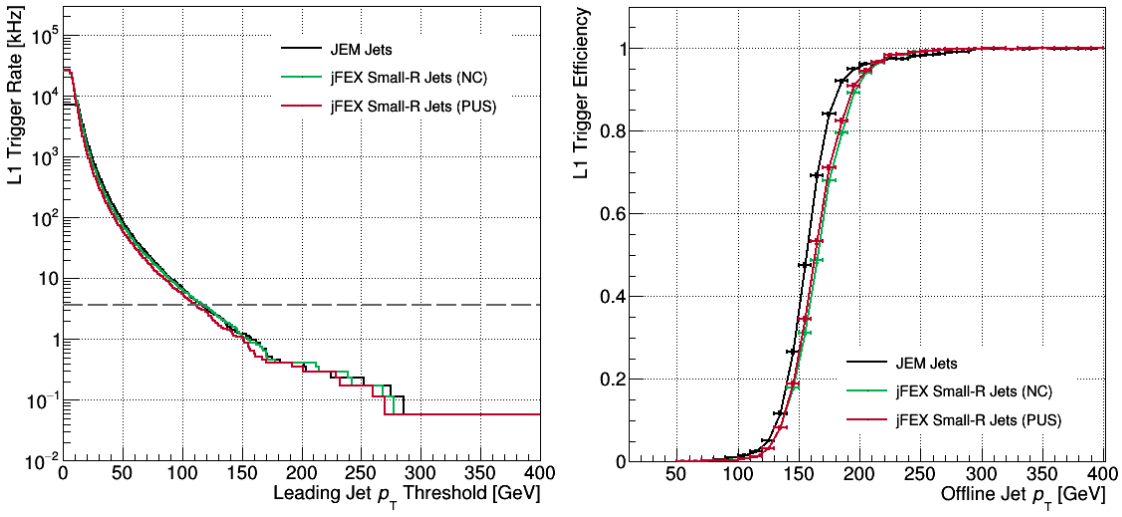

**(a)** rates **(b)** per-event efficiencies

**Figure 8.1:** Performance of $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms: (a) rates at different $E_{\mathrm{T}}^{\mathrm{miss}}$ thresholds, derived from the minimum bias sample (JZ0W); (b) per-event efficiencies with respect to truth $E_{\mathrm{T}}^{\mathrm{miss}}$, derived from the $ZH$ with $Z \to 2\nu$ and $H \to 2b$ sample and rate-matched to 4.6 kHz (according to the current Run-3 rate estimate in Ref. [58]). To illustrate the relationship between the rate and efficiency histograms, the derived threshold values are drawn in and listed here once.

The jFEX $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm with upstream pile-up subtraction shows the best efficiency as it reaches the plateau much smoother and allows the trigger threshold to be lowered significantly without changing the rate. Thus, the latter has great potential to increase the sensitivity to physically interesting events with lower $E_{\mathrm{T}}^{\mathrm{miss}}$.

## 8.3 Performance of Level-1 Jet Algorithms

Figure 8.2 shows the performance of the jFEX small-R jet algorithm, once with upstream noise cuts (NC) and once with upstream pile-up subtraction (PUS), in direct comparison to the Run-2 JEM jets:
The L1 trigger rates at different leading jet $p_{\mathrm{T}}$ thresholds are given in Figure 8.2 (a), while Figure 8.2 (b) shows the rate-matched efficiencies with respect to the offline reconstructed jet $p_{\mathrm{T}}$. As there can be more than one jet in a single event, the efficiency histogram allows multiple entries per event: First, a $p_{\mathrm{T}}$-sorted list is generated for the offline jets. Starting with the leading offline jet, the corresponding online jet is identified by applying a $\Delta R$ matching, where at the end only pairs with $\Delta R < 0.4$ are accepted, from which the pair with the smallest distance is then selected. Successfully matched online jets are removed from their list to avoid matching the same jet more than once.

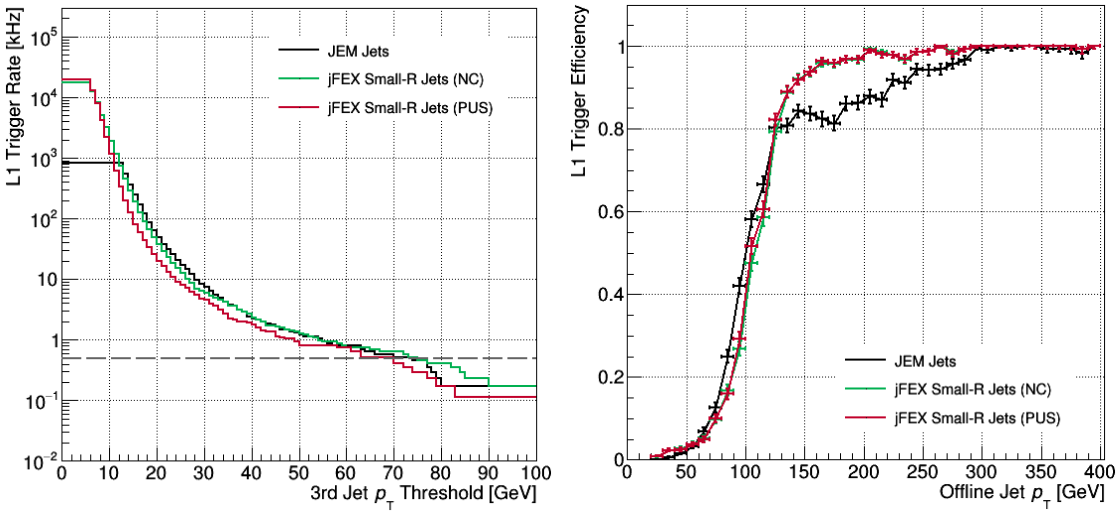



**(a)** rates **(b)** per-jet efficiencies

**Figure 8.2:** Overall performance of small-R jet algorithms: (a) leading jet rates at different $p_{\mathrm{T}}$ thresholds, derived from the minimum bias sample (JZ0W); (b) per-jet efficiencies with respect to the offline reconstructed jet $p_{\mathrm{T}}$ ($|\eta| < 2.5$ and $p_{\mathrm{T}} > 20\,\mathrm{GeV}$), derived from the $2H \to 4b$ sample and rate-matched to 3.7 kHz (according to the current Run-3 rate estimate in Ref. [58]).

While there are no major differences between the two jFEX versions, the JEM curve is struggling to reach the efficiency plateau. This becomes even more evident in Figure 8.3, where the performance in events with at least two jets with $\Delta R < 0.6$ is shown. The inefficiency of the Run-2 jet algorithm is mainly due to the fact that there is no suitable JEM jet for some offline jets. The JEM algorithm can obviously no longer differentiate between jets at very small distances. However, jFEX can go further here as it enables jet identification on the basis of a granularity that is four times finer.

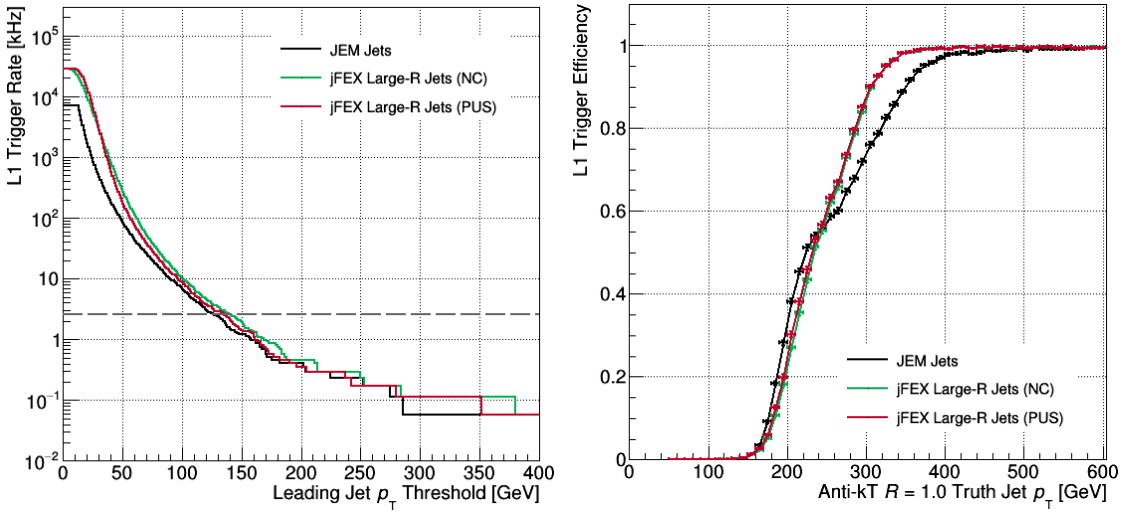
Figure 8.4 shows the performance of the jFEX large-R jet algorithm, once with upstream noise cuts (NC) and once with upstream pile-up subtraction (PUS). Both are compared to the standard Run-2 jet algorithm, as there is no extra large-R jet algorithm implemented




**(a)** rates **(b)** per-jet efficiencies

**Figure 8.3:** Close-by performance of small-R jet algorithms: (a) 3-jet rates at different $p_{\mathrm{T}}$ thresholds, derived from the minimum bias sample (JZ0W); (b) per-jet efficiencies with respect to the offline reconstructed jet $p_{\mathrm{T}}$ ($|\eta| < 2.5$ and $p_{\mathrm{T}} > 20\,\mathrm{GeV}$, where at least one jet above 20 GeV is required to be within $\Delta R = 0.6$ of the probe jet), derived from the $2H \to 4b$ sample and rate-matched to 0.5 kHz (according to the current Run-3 rate estimate in Ref. [58]), while requiring at least three triggered jets per event.

in the JEMs. The underlying trigger threshold was adjusted so that the rate is the same for all three curves.
The per-jet efficiencies in Figure 8.4 (b) are shown with respect to the anti-$k_{\mathrm{T}}$ $R = 1.0$ truth jet $p_{\mathrm{T}}$. Both jFEX algorithms show a clear improvement in the identification of large-R jets, mainly due to the enlarged window that can capture more of their energy.




**(a)** rates **(b)** per-jet efficiencies

**Figure 8.4:** Performance of large-R jet algorithms: (a) leading jet rates at different $p_{\mathrm{T}}$ thresholds, derived from the minimum bias sample (JZ0W); (b) per-jet efficiencies with respect to the anti-$k_{\mathrm{T}}$ $R = 1.0$ truth jet $p_{\mathrm{T}}$ ($|\eta| < 2.5$ and $p_{\mathrm{T}} > 50\,\mathrm{GeV}$), derived from the $2H \to 4b$ sample and rate-matched to 2.6 kHz (according to the current Run-3 rate estimate in Ref. [58]).

CHAPTER 9

# Implementation and Tests of Feature Extraction Algorithms

*The programmer, like the poet,*
*works only slightly removed*
*from pure thought-stuff.*
*He builds his castles in the air, from air,*
*creating by exertion of the imagination.*

( Frederick P. Brooks, Jr. )

This chapter is dedicated to the firmware development of the jFEX algorithms with VHDL and Vivado. The focus is on the strategies for implementing the algorithms on the target FPGA, including hardware tests.
After a brief overview of internal saturation and overflow handling, a description of the most important implementation strategies follows. A summary on latency and FPGA resource utilization is then given for a stand-alone implementation of the jFEX algorithms. Finally, the current status of the integration with infrastructure and readout firmware is briefly discussed.

## 9.1 Internal Saturation and Overflow Handling

Within jFEX, all calculations are based on the best available $E_{\mathrm{T}}$ resolution, namely 25 MeV (500 MeV) per LSB for input data from LATOME (TREX). The bit lengths are extended in order to avoid uncontrolled arithmetic overflows. However, when generating TOBs, energy values have to be truncated. If an overflow occurs while truncating, the corresponding energy value is set to full-scale.
There is a limited number of TOBs per type that can be transmitted to the L1Topo processors. If more TOBs of a kind are found, an overflow bit is set and stored in the trailer of the corresponding output data stream.

## 9.2 Repeating Implementation Strategies

There are individual and repeating implementation strategies. The latter are discussed together in this section. They primarily deal with saving FPGA resources and/or latency.

### 9.2.1 Parallel Processing

The jFEX algorithms are full of structures that occur repeatedly and independent of each other, e.g. masking of individual trigger towers in $\eta \times \varphi$. Such structures are ideally suited to be executed in parallel.
In VHDL, the `generate` statement is used to describe regular design structures. It allows a large number of identical components to be created by specifying the component once and then repeating it with the generate mechanism. An example is given in Listing 9.1, which shows an extension of the registered AND gate in Listing 5.1.

```
1  G_EXAMPLE: for i in 9 downto 0 generate
2    C(i) <= A(i) and B(i) when rising_edge(CLK);
3  end generate G_EXAMPLE;
```

**Listing 9.1:** Example of a generate statement in VHDL, which shows the instantiation of 10 registered AND gates.

A total of 10 AND gates are instantiated to be executed in parallel. The signals `A`, `B` and `C` are one-dimensional arrays of the `std_logic` type. The generation parameter `i` and its scope of values (9 downto 0) need to be specified before the keyword `generate`.
Generate statements can also be nested to create multidimensional structures, e.g. for addressing the $\eta \times \varphi$ phase space.

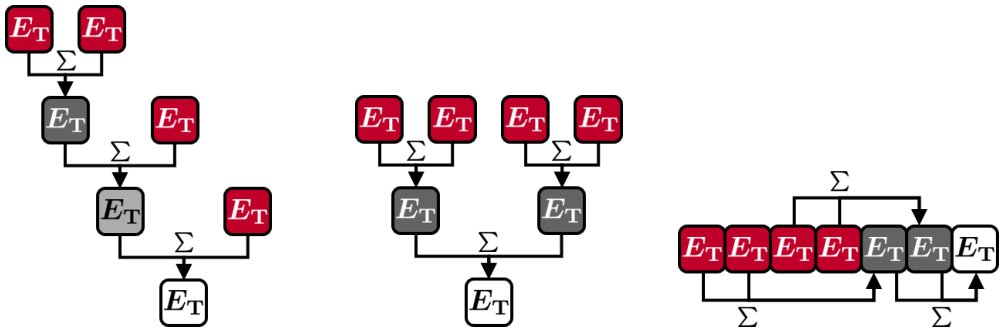
### 9.2.2 Cost Savings

The math primitives addition, subtraction and multiplication are available in the VHDL library provided by Xilinx/Vivado. Adding and subtracting works the same way. The synthesis tool translates both operations using a certain number of LUTs depending on the number of bits involved. A multiplication can also be implemented in LUTs, but requires a large amount of resources. Instead, it is rather pushed into a DSP block, which also enables fast signal processing.
The division requires a more complex logic circuit. Xilinx provides an optimized IP core that can be used for such operations. However, it requires a certain number of FPGA resources and/or clock cycles: A fast division algorithm requires a lot of resources. Conversely, serial algorithms require a moderate amount of resources, but many clock cycles. Therefore, divisions in hardware should be avoided whenever possible.
If a division cannot be avoided: Dividing (multiplying) by $2^N$ is equivalent to a shift right (left) by $N$ bits. To divide by constants that are not powers of two, a multiplication with the inverse can be performed. For numbers that are not constant and not powers of two, the division values can be mapped into a LUT using the divisor as the address to get the right output value.

### 9.2.3 Efficient Summation

With almost all jFEX algorithms, sums of different sizes have to be formed. In software programming, the standard method is to add two energies step by step, as it is illustrated in Figure 9.1 (a). For $N$ inputs there are $N-1$ stages. To make use of the large amount

of resources available on an FPGA, the scheme in Figure 9.1 (b) is used instead. It shows a so-called *balanced tree*, where as many sums as possible are processed in parallel. The number of stages is reduced to $\lceil \log_2 N \rceil$.




**(a)** non-balanced **(b)** balanced **(c)** balanced

**Figure 9.1:** Summation methods for four $E_T$ inputs (colored in red): (a) non-balanced, (b) balanced and (c) balanced w.r.t. how it is implemented in VHDL.

The generate statement described above is used to implement such a balanced tree wherever a sum is built. Listing 9.2 shows a code snippet for a balanced sum of $N$ inputs.

```
1   G_SUM: for s in ( (2*N - 1) - 1 ) downto N generate
2     ET(s) <= ET( 2*(s - N) + 1 ) + ET( 2*(s - N) );
3   end generate G_SUM;
```

**Listing 9.2:** VHDL code snippet for a balanced sum of N inputs.

The signal `ET` is a one-dimensional array and contains all the $N$ inputs in the range 0 to $N-1$. All connections are set with the generation parameter `s`, as it is shown in Figure 9.1 (c). Compared to a non-balanced sum, a balanced sum does not reduce the number of FPGA resources required, but rather reduces latency.
Analogously, chains of logical conjunctions and disjunctions are implemented as balanced trees by simply replacing the operator in Listing 9.2.

## 9.3 Individual Implementation Strategies

In this section, algorithm-specific implementation strategies are discussed in the order in which the algorithms were described in Chapter 7.
Since there are no special implementation strategies for the initial data/saturation checks and the tower masking (see Sec. 7.3.1), they are omitted below. The same applies to the noise cut stages and EM/HAD summations (see Sec. 7.3.3).

### 9.3.1 Energy Linearization

The energy linearization of the LATOME input is done by a function that converts 12-bit words (multi-linear encoded) into 16-bit words (25 MeV per LSB), including a sign bit to

also account for the negative energies. According to Table 6.1 it is

$$(E_{\mathrm{T}})_{\texttt{linear}} = \begin{cases} (\texttt{Code} - 128) & \text{if } \texttt{Code} < 384 \\ (\texttt{Code} - 384) \cdot 2 + 2^8 & \text{if } \texttt{Code} < 768 \\ (\texttt{Code} - 768) \cdot 2^2 + 2^{10} & \text{if } \texttt{Code} < 1536 \\ (\texttt{Code} - 1536) \cdot 2^3 + 2^{12} & \text{if } \texttt{Code} < 3072 \\ (\texttt{Code} - 3072) \cdot 2^4 + 2^{14} & \text{if } \texttt{Code} < 4048 \\ 32000 & \text{if } \texttt{Code} = 4048 \end{cases} . \tag{9.1}$$

Technically, only constants are added and bit shifts are carried out. The latter are free of cost in terms of FPGA logic resources.

### 9.3.2 Pile-up Calculation and Subtraction

In order to avoid the divisions in Equation 7.1, multiplications are performed with the inverse. Thus, Equation 7.1 becomes

$$\rho = \left( \sum_{i}^{N} (E_{\mathrm{T}})_i^w \right) \cdot N^{-1} \tag{9.2}$$

with

$$(E_{\mathrm{T}})_i^w = (E_{\mathrm{T}})_i \cdot w_i^{-1}. \tag{9.3}$$

The weights are loaded into the firmware once normally and once inverted. Since they are actually floating point values, they have to be transformed into integers beforehand: The (non-) inverted weights are multiplied by $2^6$ ($2^{11}$) and cut off after the decimal point, resulting in 17-bit integer values. This way, the smallest weight becomes 1. After the multiplications in Equation 9.3 (7.3), the results are divided by $2^6$ ($2^{11}$) to correct for the transformation. Each division is implemented as a simple bit shift.
For the multiplication in Equation 9.2 the factor $N^{-1}$ might change from event to event. $N$ ranges from 0 to $24 \times 32$, which results in floating point values for $N^{-1}$. A transformation into integers is also required here: The values for $N^{-1}$ are multiplied by $2^{20}$ and cut off after the decimal point. This way, there are no duplicate values. Technically, these values are stored in a LUT that is quite large, but is only called once per region/$\rho$. After the multiplication in Equation 9.2, the results are divided (bit shifted) by $2^{20}$ to again correct for the transformation.

### 9.3.3 Energy Re-formatting

The energy re-formatting of the TREX input is done by a function that converts 8-bit words (500 MeV per LSB) into 13-bit words (25 MeV per LSB). To save some DSP slices, the multiplication is avoided by applying two different bit shifts to the original value and then adding up the results:

$$(E_{\mathrm{T}})_{25\,\texttt{MeV}} = (E_{\mathrm{T}})_{500\,\texttt{MeV}} \cdot 20 = (E_{\mathrm{T}})_{500\,\texttt{MeV}} \cdot 2^4 + (E_{\mathrm{T}})_{500\,\texttt{MeV}} \cdot 2^2. \tag{9.4}$$

### 9.3.4 Global Variables

For the $\Sigma E_{\mathrm{T}}$ algorithm, $\eta$ slices are formed by adding up energies at constant $\eta$. To provide pairs of $\Sigma E_{\mathrm{T}}$ values for all possible $\eta$ boundaries, various combinations of $\eta$ slices are grouped and summed in parallel. The desired pair of values is then selected via an IPbus register.
Analogously, $\varphi$ slices are formed for the $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm. According to Equation 7.4, the sums are then multiplied with the appropriate $\cos\varphi$ and $\sin\varphi$ values. These are floating point values multiplied by $2^9$ and cut off after the decimal point before being implemented as 11-bit integers. The underlying deviation from the exact values is less than 0.5%, which is negligible compared to the energy resolution. After adding up the individual (weighted) $\varphi$ slices, the results are divided (bit shifted) by $2^9$ to correct for the previous transformation.

### 9.3.5 Small-R Jets, Large-R Jets and Taus

Some of the quantities calculated for small-R jets, large-R jets and taus are identical. To save FPGA resources, they are only computed once and then shared among each other. Figure 9.2 shows the corresponding VHDL entities and all of the shared quantities.

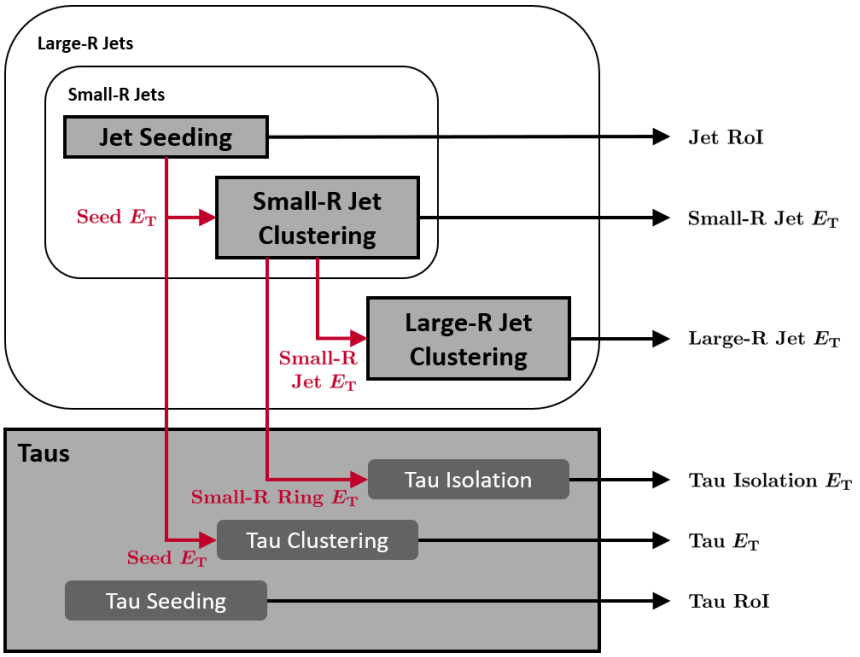



**Figure 9.2:** VHDL entities (light gray) with shared quantities (red): seed $E_{\mathrm{T}}$ ($\Delta R < 0.2$), small-R ring $E_{\mathrm{T}}$ ($0.2 \leq \Delta R < 0.4$) and small-R jet $E_{\mathrm{T}}$ ($\Delta R < 0.4$). The tau entity has three sub-units (dark gray). The output of each entity/sub-unit is shown on the right.

Seed $E_{\mathrm{T}}$ sums are formed and tested for local maxima in order to identify jet RoIs. In addition, each seed serves as a full tau cluster and as a partial sum for a small-R jet. To get the full small-R jet $E_{\mathrm{T}}$, the sum of all trigger towers within $0.2 \leq \Delta R < 0.4$, referred to as small-R ring $E_{\mathrm{T}}$, is calculated and then added to the seed. In parallel, the small-R

ring serves as the tau isolation. The large-R jet clustering algorithm computes the sum of all trigger towers within $0.4 \leq \Delta R < 0.8$ and adds it to the small-R jet.
All quantities are shared with the best available $E_{\mathrm{T}}$ resolution (25 MeV per LSB), while the outputs are truncated to fit the different TOB formats (listed in Tab. 7.2).

As already shown in Figure 7.5, the patterns of seeds, jet and search windows vary in the high $|\eta|$ regions. To avoid a large number of lines of code covering all the different cases, a list-based implementation is used instead: These lists were generated with a Python script that takes care of all the distance calculations to check which trigger tower is part of which energy sum. They provide information about the number of sums to be calculated as well as the length of each sum itself. Thus, each list represents a kind of construction manual for the synthesis tool. Changes to a list only requires a new synthesis run, but no change to the firmware code.
For the sake of simplicity, the trigger towers are numbered consecutively according to their $\eta/\varphi$ coordinates, so that they can be clearly assigned in the software and firmware code. Figure 9.3 shows the corresponding indexing of the towers on forward FPGAs (compare with Fig. 7.2). To avoid gaps in the numbering, some of the towers are shifted towards smaller $\eta/\varphi$ values, while the three FCal layers are stacked in the vertical direction.

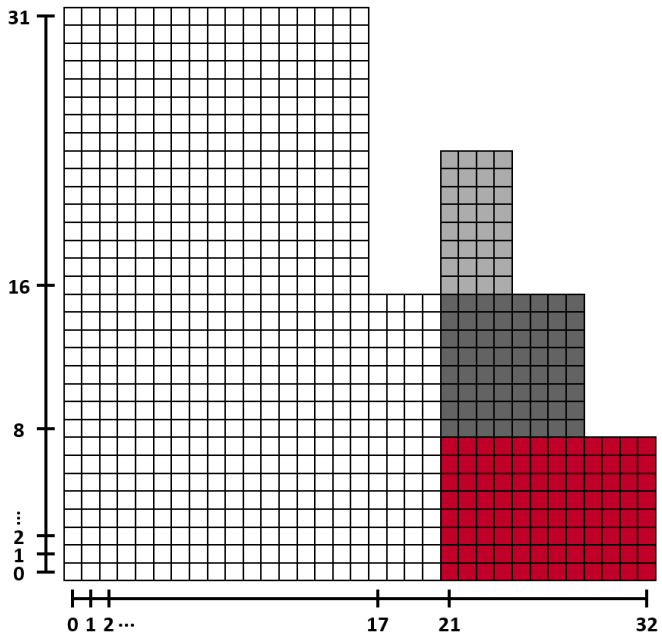



**Figure 9.3:** Trigger tower indexing on forward FPGAs: For the FPGA U1 on module 1A the index pair (0|0) corresponds to the tower at $1.6 < \eta < 1.7$ and $0 < \varphi < \pi/32$. Towers from the FCal layer 1/2/3 are colored in red/dark gray/light gray.

### 9.3.6 Electrons

The electron entity processes EM and HAD data separately. Since the seeding and clustering algorithms only run in the EM layer, there are no shared quantities with jets/taus.

However, the lists generated for their search windows are reused for the electron seeding. To identify the most energetic $\eta/\varphi$ neighbor, potential candidates are sorted by energy. The underlying sorting algorithm is based on the TOB sorting algorithm described in the following section.
Separate lists are generated for the EM isolation and both versions of EM fraction. In order to avoid divisions, each condition is redefined: Equation 7.6 turns into

$$E_{\mathrm{T}}(\mathrm{Cluster}) > C_i^* \cdot E_{\mathrm{T}}(\mathrm{EM\ isolation}) \tag{9.5}$$

with $C_i^* = (1 - C_i) \cdot C_i^{-1}$, while Equation 7.7 turns into

$$E_{\mathrm{T}}(\mathrm{EM}) > C_f^* \cdot E_{\mathrm{T}}(\mathrm{HAD}) \tag{9.6}$$

with $C_f^* = C_f - 1$. Since the parameters $C_i$ and $C_f$ are meant to be configurable, they can easily be adapted to the modified equations.

### 9.3.7 TOB Sorting

The algorithm for sorting $N_{\mathrm{in}}$ TOBs to get a sorted list of $N_{\mathrm{out}}$ TOBs ($N_{\mathrm{out}} \leq N_{\mathrm{in}}$) has been ported from the Phase-0 L1Topo firmware repository [59] and modified to meet the various jFEX TOB requirements.

The basic algorithm works as follows [60]: In an FPGA it is possible to carry out a large number of comparisons in parallel and thus to build up a comparison matrix, as it is illustrated in Figure 9.4.

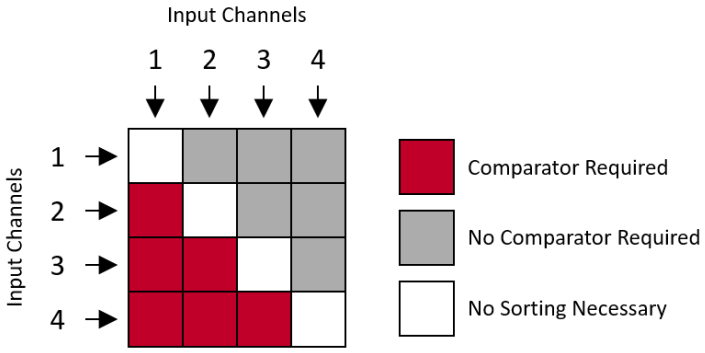



**Figure 9.4:** Comparison matrix for four input channels based on Figure 6.8 in Reference [60]. No comparators are required for elements above the diagonal, while no sorting is required on the diagonal.

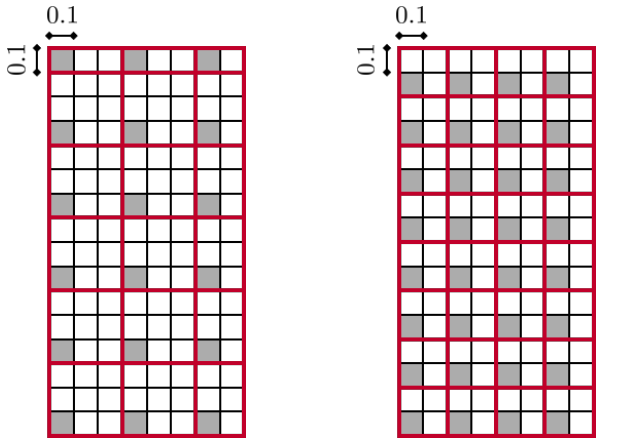
For each entry in the half below the diagonal, the $i^{\mathrm{th}}$ input channel is compared with the $j^{\mathrm{th}}$ input channel, where $i$ ($j$) is the index of the row (column) of the matrix. If the $i^{\mathrm{th}}$ input channel is greater or equal to the $j^{\mathrm{th}}$ input channel, a '1' is written in the corresponding field, '0' otherwise. The half above the diagonal can be derived from the lower one and therefore does not require any additional comparisons. The diagonal itself is not required as it is not necessary to compare the input channels with themselves.
The number of ones in the $i^{\mathrm{th}}$ row determines the position of the $i^{\mathrm{th}}$ input channel in the sorted output list. In case of $k$ input channels, the $i^{\mathrm{th}}$ input channel is assigned to the $l^{\mathrm{th}}$

position of the output list, if the $i^{\text{th}}$ row contains $k-l$ ones. However, adding up all fields of a row requires too many resources. As an alternative, a row can always be compared with the binary number '11...1' ($k$ digits). This way, only the largest output channel can be found with the original comparison matrix. For each additional position in the output list, a new comparison matrix must be created from the previous one. Using XOR gates, the fields of the matrix are changed in such a way that the row originally filled with $k-l$ ones in the $l^{\text{th}}$ comparison matrix is now filled with $k$ ones. The number of necessary comparison matrices is therefore determined by the number of output channels.

For the implementation of the TOB sorting algorithms on jFEX, the conditions on the central and forward FPGAs must be considered individually:

### Sorting on Central FPGAs

The amount of FPGA resources required increases significantly with the size of $N_{\text{in}}$ and $N_{\text{out}}$. The same applies to the latency. In order to make the TOB sorting more efficient in terms of both resources and latency, the sorting takes place in two sequential steps:
Due to the definition of a local maximum, there can only be one small-R/large-R jet (tau) TOB within an area of $0.3 \times 0.3$ ($0.2 \times 0.2$) in $\eta \times \varphi$. As a consequence, there cannot be more than 18 (32) TOBs per FPGA core region. Possible scenarios for a maximum number of TOBs per type are given in Figure 9.5.




**(a)** $\Delta\eta \times \Delta\varphi = 0.8 \times 1.6$ **(b)** $\Delta\eta \times \Delta\varphi = 0.8 \times 1.6$

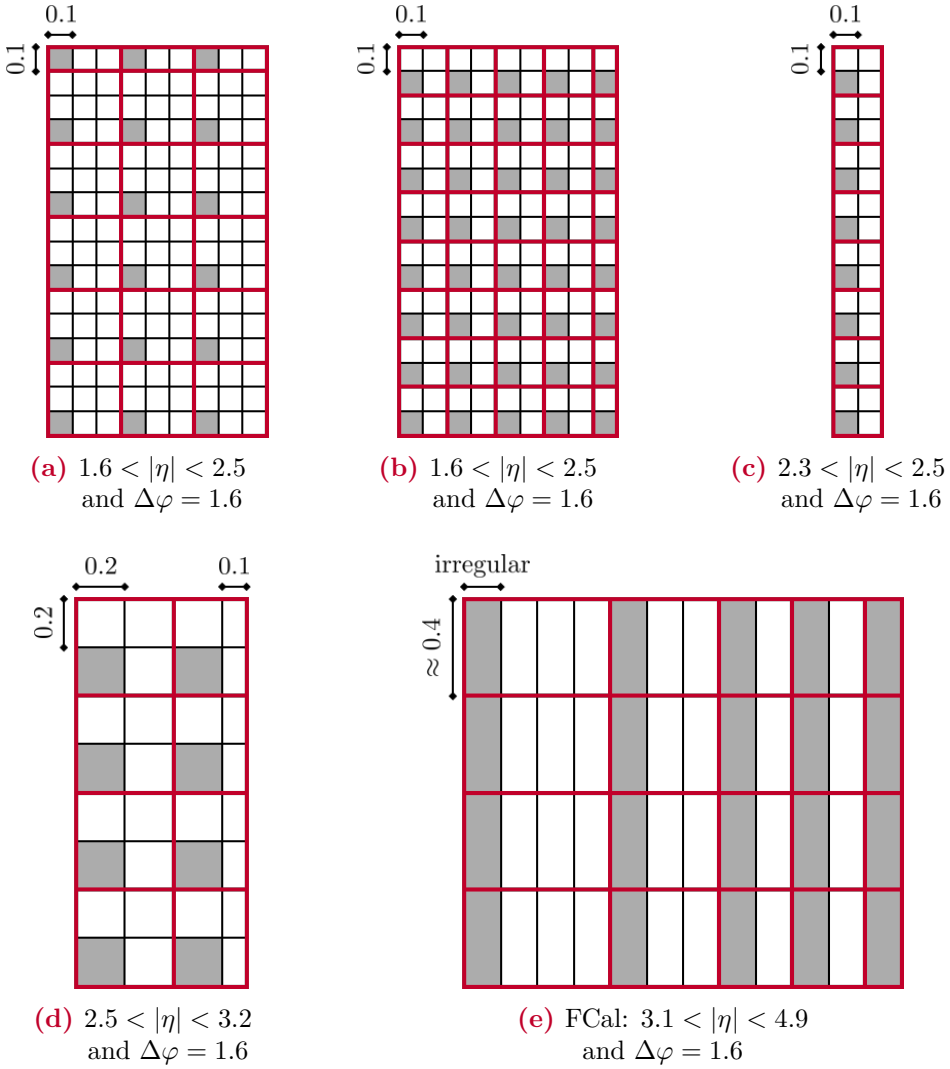
**Figure 9.5:** TOB sorting on central FPGAs: Possible scenario for a maximum number of TOBs for (a) small-R/large-R jets and (b) taus. Local maxima are highlighted in gray, while the red boxes indicate areas in which there cannot be more than one maximum. The positive $\eta$ ($\varphi$) axis runs from left (bottom) to right (top).

Therefore, 18 small-R/large-R jet (32 tau) TOB sorting algorithms are executed in parallel. Each algorithm takes up to nine (four) inputs and produces one output. The resulting list of 18 (32) TOBs is then sorted in a second step to get a sorted list of eight/one (seven) TOBs. A type-specific, configurable $E_{\text{T}}$ cut is applied and only the seven/one (six) most energetic TOBs are sent to L1Topo. For small-R jets and taus, the least energetic TOB is only required to check for a TOB overflow. No overflow is determined for large-R jets.

If all 128 TOBs were sorted in one step and there were eight TOBs in the sorted output list, the number of LUTs required would be almost 13 times higher.

### Sorting on Forward FPGAs

As with sorting on central FPGAs, sorting on forward FPGAs also takes place in two sequential steps. However, the size and number of areas in which there cannot be more than one maximum are partly different, as shown in Figure 9.6.




**(a)** $1.6 < |\eta| < 2.5$ and $\Delta\varphi = 1.6$

**(b)** $1.6 < |\eta| < 2.5$ and $\Delta\varphi = 1.6$

**(c)** $2.3 < |\eta| < 2.5$ and $\Delta\varphi = 1.6$

**(d)** $2.5 < |\eta| < 3.2$ and $\Delta\varphi = 1.6$

**(e)** FCal: $3.1 < |\eta| < 4.9$ and $\Delta\varphi = 1.6$

**Figure 9.6:** TOB sorting on forward FPGAs: Possible scenario for a maximum number of TOBs for (a) jets, (b) taus, (c) electrons, (d) and (e) jets/electrons. Local maxima are highlighted in gray, while the red boxes indicate areas in which there cannot be more than one maximum. The positive $\eta$ ($\varphi$) axis runs from left (bottom) to right (top).

According to Figures 9.6 (a), (d) and (e), there are 46 jet TOB sorting algorithms that are executed in parallel, while there are 40 for taus (Fig. 9.6 (b)) and 36 for electrons (Figs. 9.6 (c), (d) and (e)). After the second sorting step, sorted TOB lists of eight small-R jets,

one large-R jet, seven taus and six electrons remain. With the exception of large-R jets, the least energetic TOBs are again only required to check for TOB overflows.

Generally, the TOB sorting is based on the FIFO[1] principle. If the energy is the same, prioritization initially takes place according to the $\eta$ regions $|\eta| < 2.5$, $2.5 < |\eta| < 3.2$ and $3.1 < |\eta| < 4.9$ (FCal). Within these regions, priority is assigned in ascending order in $\varphi$ and at constant $\varphi$ in ascending order in $|\eta|$ (always w.r.t. the areas in which there cannot be more than one maximum).

## 9.4 Stand-Alone Implementation

Before being integrated with the infrastructure and readout firmware, the jFEX algorithms were thoroughly tested in stand-alone implementations. For this purpose, all I/O signals were connected to BRAMs, while ILAs were used for verification and debugging.

### 9.4.1 Latency and Behavioral Simulation

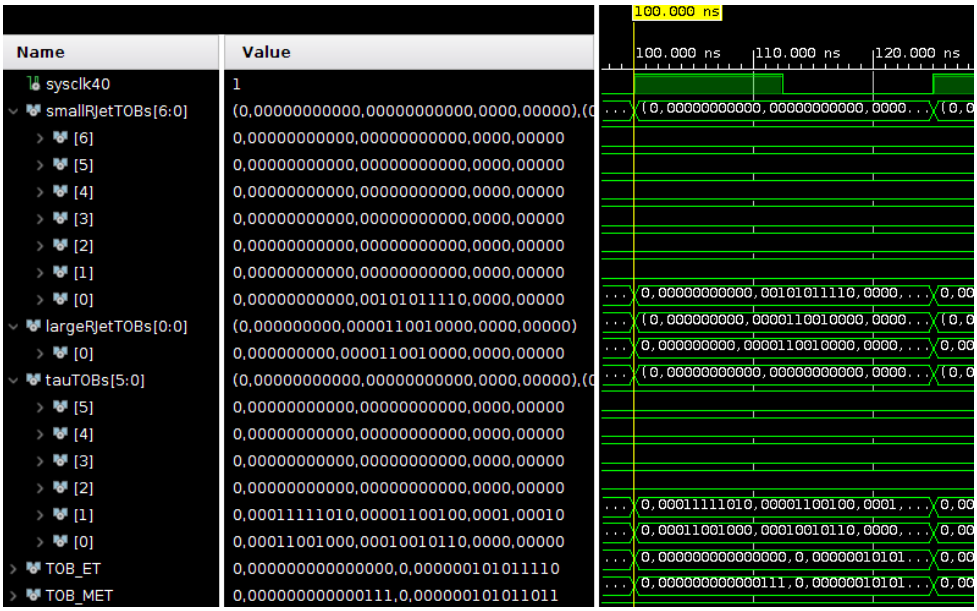
New data must be processed every 25 ns, which requires the jFEX algorithms to be implemented as pipelined processors. In order to achieve timing closure, 40 MHz registers are placed

- before the multiplication in Equation 9.2,
- before applying noise cuts,
- before generating TOBs and
- after sorting TOBs.

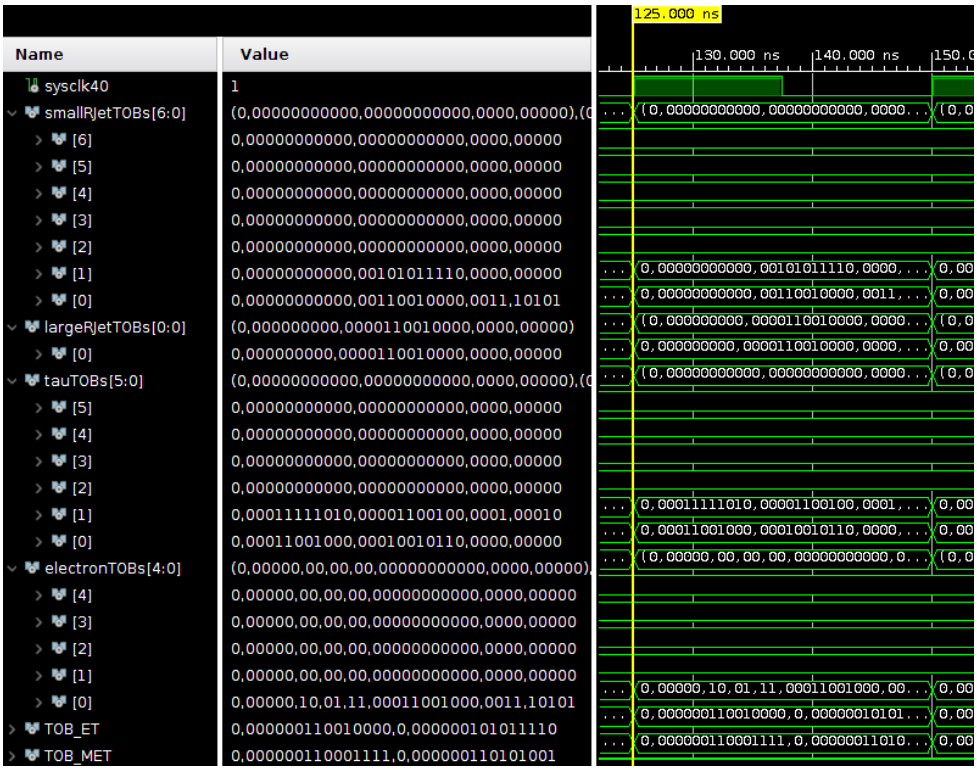
Due to the increased number of TOBs to be sorted on forward FPGAs, additional registers are placed between the two sorting steps. Since $\Sigma E_{\mathrm{T}}$ and $E_{\mathrm{T}}^{\mathrm{miss}}$ do not need to be sorted, their results are delayed to be in sync with the jet/tau/electron TOBs.
In summary, $4 \times 25$ ns (100 ns) are required for processing the algorithms on central FPGAs, while $5 \times 25$ ns (125 ns) are required on forward FPGAs. Taking the TOB sorting into account, both values are well within the available latency budget of 150 ns (according to the current Phase-I latency envelope in Ref. [61]).

Vivado simulations were carried out to prove the functionality of each individual algorithm. As an example, Figure 9.7 shows the results of a combined simulation run for a complete central/forward FPGA configuration using the inputs from Table 9.1:
All inputs are large enough to exceed the pile-up and noise cut limits. Each TOB energy threshold is set to zero, while the $\eta$ boundary for splitting the $\Sigma E_{\mathrm{T}}$ value is set to 4, forming the lower $|\eta|$ range of 4 bins.
As expected, a 70 GeV small-R (80 GeV large-R) jet at $\eta/\varphi = 0/0$ and a 30 GeV (20 GeV) tau at 0/0 (2/1) with an isolation $E_{\mathrm{T}}$ of 40 GeV (50 GeV) were identified by the central processor (Fig. 9.7 (a)). The two energy depositions in the HAD layer at $\eta = 8$ result in two taus due to the smaller search window compared to jets. In addition, $\Sigma E_{\mathrm{T}}$ was

[1] FIFO is a common acronym for *first in, first out.*




**(a)** FPGA U1 on module 2A




**(b)** FPGA U1 on module 1A

**Figure 9.7:** Behavioral simulation results for a (a) central and a (b) forward FPGA configuration: Each waveform snapshot shows the list of TOBs to be transmitted to L1Topo after a processing time of 100 ns and 125 ns, respectively. The test inputs are listed in Table 9.1, while the binary output can be decoded using the various TOB formats in Tables 7.1 and 7.2.

| $E_{\mathrm{T}}$ [GeV] | $\eta$ index global (local) | $\varphi$ index global (local) | Calorimeter layer |
|---|---|---|---|
| 10 | 8 (0) | 8 (0) | EM |
| 10 | 4 (-) | 12 (4) | EM |
| 20 | 8 (0) | 8 (0) | HAD |
| 20 | 10 (2) | 9 (1) | HAD |
| 20 | 8 (0) | 6 (-) | HAD |
| 25 | 29 (21) | 5 (3) | FCal-1 |
| 15 | 28 (20) | 5 (3) | FCal-1 |
| 10 | 30 (22) | 5 (3) | FCal-1 |
| 20 | 27 | 13 | FCal-2 |
| 10 | 24 | 21 | FCal-3 |

**Table 9.1:** Inputs for testing the jFEX algorithm firmware: The $\eta/\varphi$ index is given w.r.t. the total FPGA environment (global) and its core region (local). The latter are only used to en-/decode the coordinates of the TOBs. In case of central FPGAs, the trigger towers are indexed in ascending order in $\eta/\varphi$ (see Fig. 6.2 (a)), while for forward FPGAs see Figure 9.3. Inputs from the FCal layers are only used in tests with forward FPGA configurations.

calculated to 70 GeV (0 GeV) for the lower (higher) $|\eta|$ range as well as $E_x^{\mathrm{miss}}$ to 69.4 GeV and $E_y^{\mathrm{miss}}$ to 1.4 GeV. Since the energy deposition at $\eta = 4$ is in the overlap region, it contributes neither to the $\Sigma E_{\mathrm{T}}$ nor to the $E_{\mathrm{T}}^{\mathrm{miss}}$ values.
Similar results can be obtained from the simulation of the forward processor (Fig. 9.7 (b)). An additional 80 GeV small-R (large-R) jet was identified at 21/3. While the second small-R jet appears at the first position in the sorted output list, the large-R jet was sorted out in the course of prioritizing TOBs with the same energy (described in Sec. 9.3.7). The list of taus remains unchanged, as there is no tau identification beyond $|\eta| = 2.5$. However, since electron identification takes place beyond $|\eta| = 2.3$, a single 40 GeV electron was found at 21/3, for which the most energetic $\eta$ neighbor was correctly determined as the one with $E_{\mathrm{T}} = 15$ GeV. The EM isolation $E_{\mathrm{T}}$ is supposed to be 10 GeV, while it is $E_{\mathrm{T}}(\mathrm{EM}) = 25$ GeV and $E_{\mathrm{T}}(\mathrm{HAD}) = 20$ GeV (10 GeV) for version A (B) of EM fraction. Given that each set of $C_i^*/C_f^*$ constants is set to $(3, 2, 1)$, the electron was identified to be "tight" in terms of EM isolation and "loose" ("medium") in terms of EM fraction version A (B). Furthermore, $\Sigma E_{\mathrm{T}}$ was calculated to 70 GeV (80 GeV) for the lower (higher) $|\eta|$ range and $E_x^{\mathrm{miss}}$ ($E_y^{\mathrm{miss}}$) to 85 GeV (79.8 GeV).

### 9.4.2 Post-Synthesis Utilization

Each entity/algorithm was synthesized individually to get an overview of the distribution of the FPGA logic resources required. The post-synthesis utilization of the individual algorithms is shown in Table 9.2.
Compared to the modules 3A and 3C, the modules 2A and 2C receive a larger fraction of data from LATOME. Since their values consist of more bits than the TREX values, more

| Algorithm | Central FPGAs | | | Forward FPGAs | | |
|---|---|---|---|---|---|---|
| | **LUTs [%]** | **FFs [%]** | **DSP Slices [%]** | **LUTs [%]** | **FFs [%]** | **DSP Slices [%]** |
| Masking & Data Preparation | 5.3 (4.2) | - | - | 5.8 | - | - |
| Pile-up Calculation & Subtraction | 13.5 (12.6) | 3.1 (3.0) | 45.0 | 13.2 | 4.1 | 41.3 |
| Noise Cuts & EM/HAD Summations | 4.9 (5.3) | - | - | 4.2 | - | - |
| $\Sigma E_{\mathrm{T}}$ | 0.5 | $< 0.1$ | - | 1.1 | $< 0.1$ | - |
| $E_{\mathrm{T}}^{\mathrm{miss}}$ | 0.4 | $< 0.1$ | 1.4 | 0.8 | $< 0.1$ | 2.4 |
| Jet Seeding | 6.4 | $< 0.1$ | - | 6.9 | $< 0.1$ | - |
| Small-R Jet Clustering | 4.2 | $< 0.1$ | - | 6.4 | 0.1 | - |
| Large-R Jet Clustering ($\lvert\eta\rvert < 2.5$) | 13.8 | $< 0.1$ | - | 15.4 (15.3) | $< 0.1$ | - |
| Taus ($\lvert\eta\rvert < 2.5$) | 0.9 | 0.1 | - | 1.0 | 0.2 | - |
| Electrons ($2.3 < \lvert\eta\rvert < 4.9$) | - | - | - | 5.1 | $< 0.1$ | 9.5 (9.4) |
| Small-R Jet TOB Generation & Sorting | 1.0 | $< 0.1$ | - | 2.9 | 0.1 | - |
| Large-R Jet TOB Generation & Sorting | 0.8 | $< 0.1$ | - | 1.3 | $< 0.1$ | - |
| Tau TOB Generation & Sorting | 1.5 | $< 0.1$ | - | 2.1 | 0.1 | - |
| Electron TOB Generation & Sorting | - | - | - | 1.5 | 0.1 | - |
| $\Sigma$ | **53.2 (51.6)** | **3.2 (3.1)** | **46.4** | **67.7 (67.6)** | **4.7** | **53.2 (53.1)** |

**Table 9.2:** Post-synthesis/estimated utilization of individual entities/algorithms: Resource usage of LUTs, FFs and DSP slices for the central FPGAs U1/U2/U3/U4 on the modules 2A/2C (3A/3C) and for the forward FPGAs U1/U4 (U2/U3) on the modules 1A/1C. The percentages refer to the XCVU9P FPGA [37].

bits must be processed and stored, which results in a slightly increased resource usage in terms of LUTs and FFs. However, this is no longer the case after the data have been summed up in depth (EM/HAD summations).
The ratio between the input data from LATOME and TREX is constant for all FPGAs on the modules 1A and 1C. But since the FCal geometry is different on the FPGAs U1/U4 and U2/U3, the utilization also differs in some places.
In general, the pile-up algorithm and the large-R jet clustering require the most resources. Note that in case of large-R jets, the identification ends at $|\eta| = 2.5$ (according to the current Run-3 trigger menu [58]).

### 9.4.3 Post-Implementation Utilization

In the integrated firmware, the MGTs ensure a connection between the FPGA and the outside world. Without such a connection, the entire design would be optimized away during the implementation run. For the stand-alone implementation discussed here, I/O lines were connected to BRAMs in order to mimic the functionality of the MGTs. On the input side, the BRAMs are configured as read-only, while on the output side they are bidirectional. The latter enables the results to be checked via ILAs.
The post-implementation utilization of the combined algorithms is shown in Table 9.3. In direct comparison to the sum of the individual post-synthesis results given in Table 9.2, there is a reduction in resources in the course of the optimization steps during the implementation run. Note that a tiny fraction of the LUT and FF resources are allocated to the ILA cores.

| **Logic Resource** | **Central FPGAs** | **Forward FPGAs** |
|---|---|---|
| LUTs [%] | 49.7 (47.7) | 64.6 (64.8) |
| FFs [%] | 3.4 (3.2) | 3.9 |
| DSP Slices [%] | 42.8 (42.3) | 51.4 |
| BRAMs [%] | 17.9 (17.6) | 18.1 |

**Table 9.3:** Post-implementation utilization of the combined entities/algorithms listed in Table 9.2: Resource usage of LUTs, FFs and DSP slices for the central FPGAs U1/U2/U3/U4 on the modules 2A/2C (3A/3C) and for the forward FPGAs U1/U4 (U2/U3) on the modules 1A/1C. The BRAMs are only required for the stand-alone implementation to mimic the functionality of the MGTs. The percentages refer to the XCVU9P FPGA [37].

## 9.5 Integration with Infrastructure and Readout Firmware

Infrastructure, readout and algorithm firmware are stored in the official ATLAS L1Calo GitLab repository [62]. While the readout simply acquires the jFEX input and output from the real-time data path, the algorithms have to be tightly integrated with the infrastructure firmware. In addition to the connections to the MGTs, numerous IPbus lines ensure the configuration of the algorithm parameters.

### 9.5.1 Parameter Control via IPbus

A complete list of the jFEX algorithm parameters is shown in Table 9.4. For parameters in the upper half, values are obtained from ATLAS databases, while the trigger menu decides on the values in the lower half.

| **Parameters** | **Number of bits** |
|---|---|
| Tower masks | 1 per EM/HAD trigger tower |
| Pile-up weights (inverted once) | $2 \times 17$ per EM/HAD trigger tower |
| Pile-up $E_{\mathrm{T}}^{\mathrm{min}}/E_{\mathrm{T}}^{\mathrm{max}}$ thresholds | $2 \times 16$ ($2 \times 8$) per region |
| Pile-up on/off switches | 1 per noise cut stage |
| Noise cuts | $2 \times 16$ ($2 \times 8$) per EM/HAD trigger tower |
| $\eta$ boundaries | 4 per module |
| $C_i^*$ (EM isolation) | $3 \times 8$ |
| $C_f^*$ (EM fraction version A) | $3 \times 8$ |
| $C_f^*$ (EM fraction version B) | $3 \times 8$ |
| TOB/xTOB $E_{\mathrm{T}}$ thresholds | $2 \times 11$ ($2 \times 13$) per object type |

**Table 9.4:** Parameters for the jFEX algorithms, revealing the number of bits to be configured via the IPbus interface.

The number of bits for the pile-up thresholds and noise cuts depends on the data source (LATOME: 16 bits, TREX: 8 bits). The TOB and xTOB thresholds are 13 bits in case of large-R jets and 11 bits for small-R jets, taus and electrons.

Each parameter has its own control register that can be accessed via a software interface. By far the majority of bits are assigned to the pile-up algorithm and noise cut stages.

### 9.5.2 Post-Implementation Utilization

At this point in time, the overall firmware integration is not yet complete, as some parts of the infrastructure and readout firmware are still under development. Table 9.5 shows the post-implementation utilization of the firmware components that have already been successfully integrated.

The MGT firmware, the playback/spy memories and the readout of the jFEX input/output are in place, but some of their features are not yet fully functional. The latter uses URAM resources exclusively, while playback and spy functionalities are implemented in BRAMs. Most of the FFs are required for the IPbus control registers.

In terms of algorithms, the following components are integrated (compare with Tab. 9.2): noise cuts and EM/HAD summations, $\Sigma E_{\mathrm{T}}$, $E_{\mathrm{T}}^{\mathrm{miss}}$, jet seeding, small-R jet clustering, small-R jet TOB generation and sorting. Thus, the firmware design in Table 9.5 allows to properly identify small-R jets and to calculate global variables.

| Logic Resource | Utilization |
|---|---|
| LUTs [%] | 24.4 |
| FFs [%] | 16.5 |
| DSP Slices [%] | 1.2 |
| BRAMs [%] | 44.2 |
| URAMs [%] | 49.9 |
| MGTs [%] | 95.8 |

**Table 9.5:** Post-implementation utilization of the infrastructure, readout and algorithm firmware components that are already integrated at this point in time (for the central FPGAs U1/U2/U3/U4 on the modules 2A/2C) [63]. The percentages refer to the XCVU9P FPGA [37].

## 9.6 Hardware Tests

The jFEX algorithm firmware was fully tested in stand-alone implementations on a VCU118 evaluation board for the Xilinx Virtex UltraScale+ FPGA (XCVU9P) [64], while initial tests with partially integrated firmware were carried out on the jFEX first production module.

### 9.6.1 VCU118 Evaluation Platform

A picture of the VCU118 evaluation board is shown in Figure 9.8. It is about 18 cm high and 24 cm long, while its operating voltage is 12 V (DC). An external device (Platform Cable USB II [65]) was used to set up a connection between the board/FPGA and a computer for configuration, monitoring and debugging purposes.

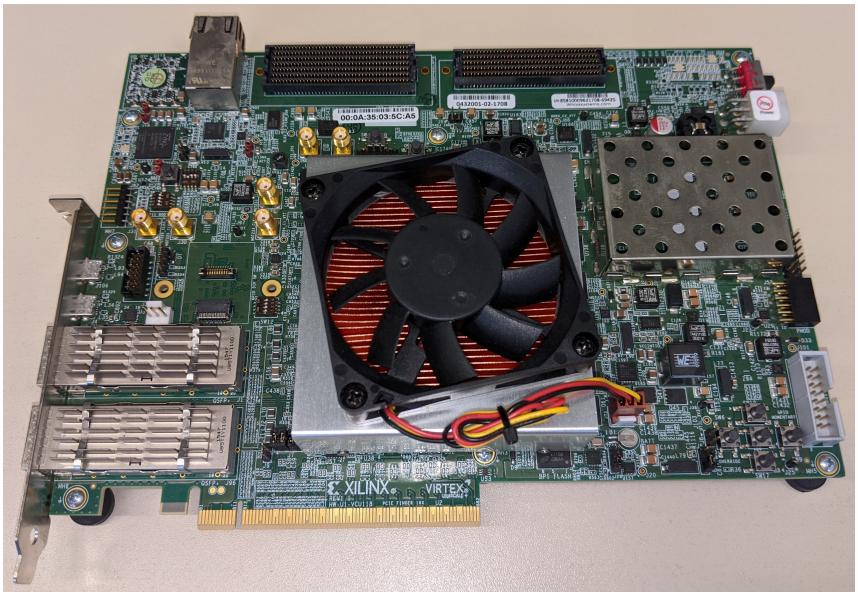
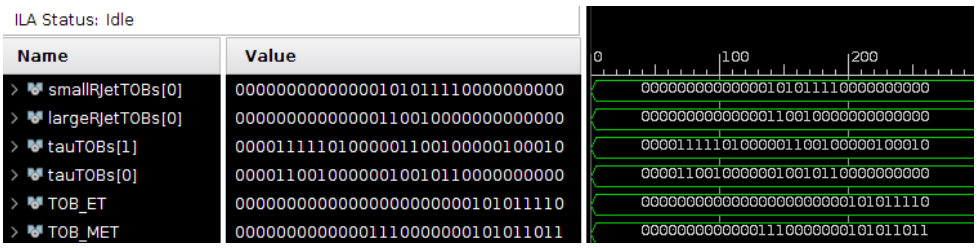


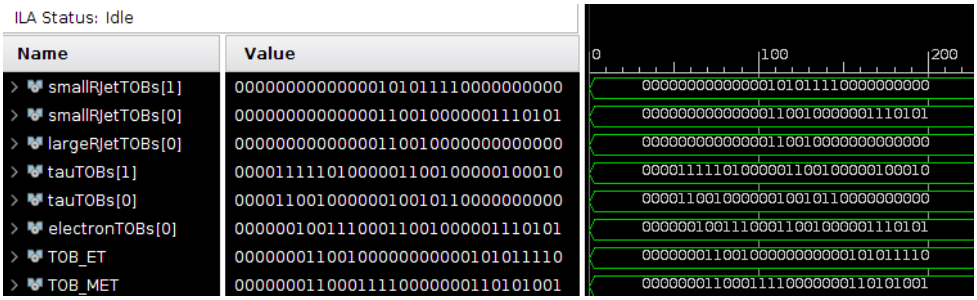
**Figure 9.8:** VCU118 evaluation board for the Xilinx Virtex UltraScale+ FPGA (XCVU9P). The FPGA is covered by a heat sink with an attached fan.

The VCU118 has an integrated clock generator that produces an LVDS[2] 300 MHz clock. The differential clock pair is connected to the top-level entity of the design in Table 9.3. A clock buffer is used to convert the differential clock signal into a single-ended clock signal, from which the required 40 MHz clock is derived via a *clocking wizard* IP core [67].

The BRAMs used as the data source and sink were instantiated with the help of *block memory generator* IP cores [68], which also enable memory initialization. After loading the bitstream generated from the implemented design in Table 9.3, hardware test results were obtained from spying on the BRAMs via ILAs. As an example, Figure 9.9 shows the results when initializing the BRAMs with the test inputs in Table 9.1. A comparison with Figure 9.7 shows complete agreement with the expected results (discussed in detail in Section 9.4.1).




**(a)** FPGA U1 on module 2A




**(b)** FPGA U1 on module 1A

**Figure 9.9:** Hardware test results on the VCU118 evaluation board when loading a (a) central and a (b) forward FPGA configuration: Each snapshot shows the nonzero TOBs produced from the test inputs in Table 9.1.

### 9.6.2 jFEX Production Module

Figure 9.10 shows a picture of the jFEX first production module (compare with the schematic design in Fig. 6.3). It has been at CERN since September 2020 and is consecutively validated in integrated tests with other modules such as LATOME, TREX and L1Topo.

[2]Low-voltage differential signaling (LVDS) is a technical standard in the areas of high-speed data transmission. Information is transmitted as the difference between the voltages on a pair of wires to reduce the generation of electromagnetic noise. See Reference [66] for more details.

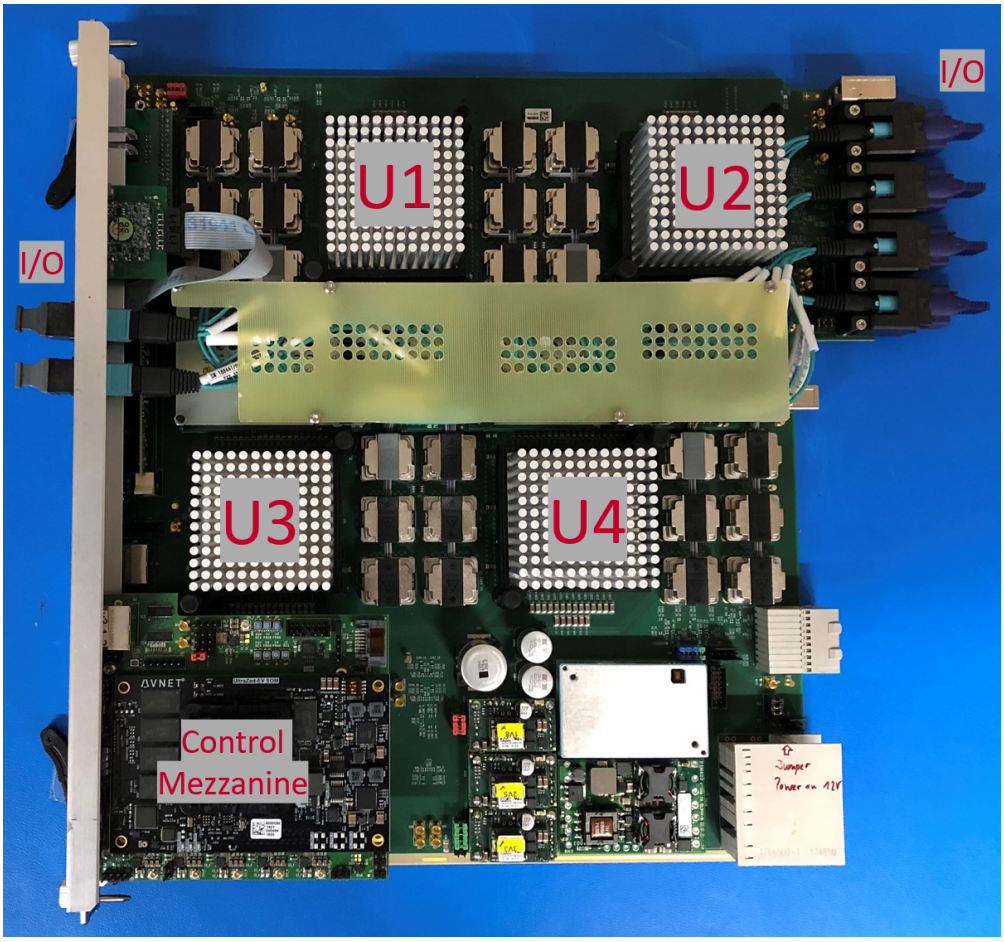



**Figure 9.10:** jFEX first production module [69]. Each FPGA is covered by a heat sink.

Hardware tests with partially integrated firmware have also been carried out [70]. So far, however, no physical inputs have been received, only test patterns that gave the expected results on the readout path.

# CHAPTER 10

# Higgs-to-Invisible Trigger Efficiency

*Doubt is useful,*
*it keeps faith a living thing.*
*After all,*
*you cannot know the strength of your faith*
*until it has been tested.*

(David Magee)

In the previous chapter it was demonstrated that the newly developed jFEX algorithms can be executed on the target FPGA. Moreover, it was shown in Chapter 8 that these algorithms lead to an improvement in the trigger efficiency of the signatures relevant for targeting events with invisibly decaying Higgs bosons, which are produced in association with hadronically decaying $W$ or $Z$ bosons.

The results already achieved are brought together in this chapter in order to determine the best Higgs-to-invisible trigger efficiency. Furthermore, possible cooperations with L1Topo are discussed.

## 10.1 Gathering Trigger Efficiencies

Figure 10.1 shows the $E_{\mathrm{T}}^{\mathrm{miss}}$ and jet trigger efficiencies for the various jFEX and JEM algorithms. These are per-event efficiencies that are computed with respect to the truth Higgs $p_{\mathrm{T}}$, reconstructed from the four-vector sum of the neutrinos that originate from the decay $H \to ZZ \to 4\nu$.

By far the best results are provided by the $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms. As with the efficiencies derived from the $ZH$ with $Z \to 2\nu$ and $H \to 2b$ sample (see Fig. 8.1 (b)), both jFEX variants show a clear improvement over the legacy system.

A look at the different jet algorithms shows that the efficiency improves with increasing jet window size, while the two jFEX variants hardly differ at fixed radii. The latter was already recognizable in the per-jet efficiencies derived from the $2H \to 4b$ sample (see Figs. 8.2 (b), 8.3 (b) and 8.4 (b)). An even larger radius for the purpose of jet energy clustering can only be achieved with the help of further processing on the L1Topo system, which will be discussed in more detail in the following section.

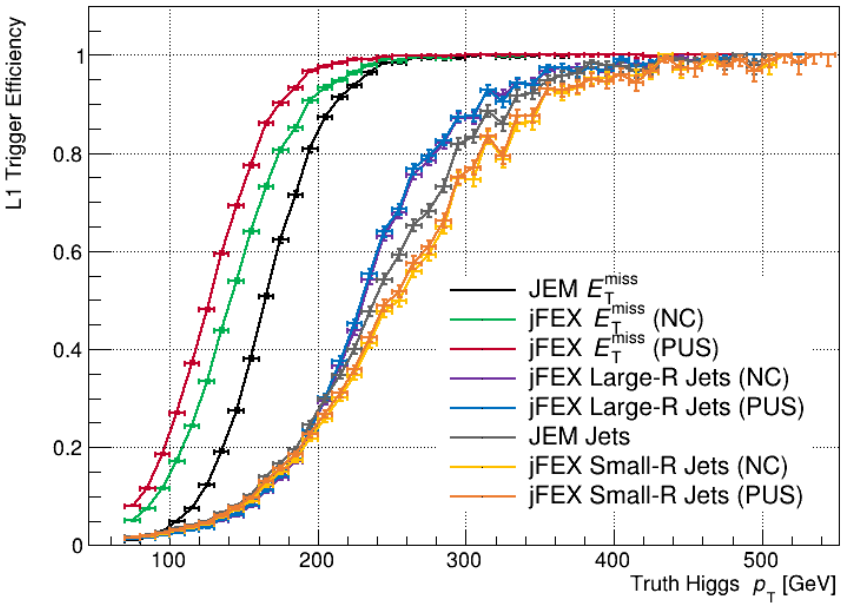


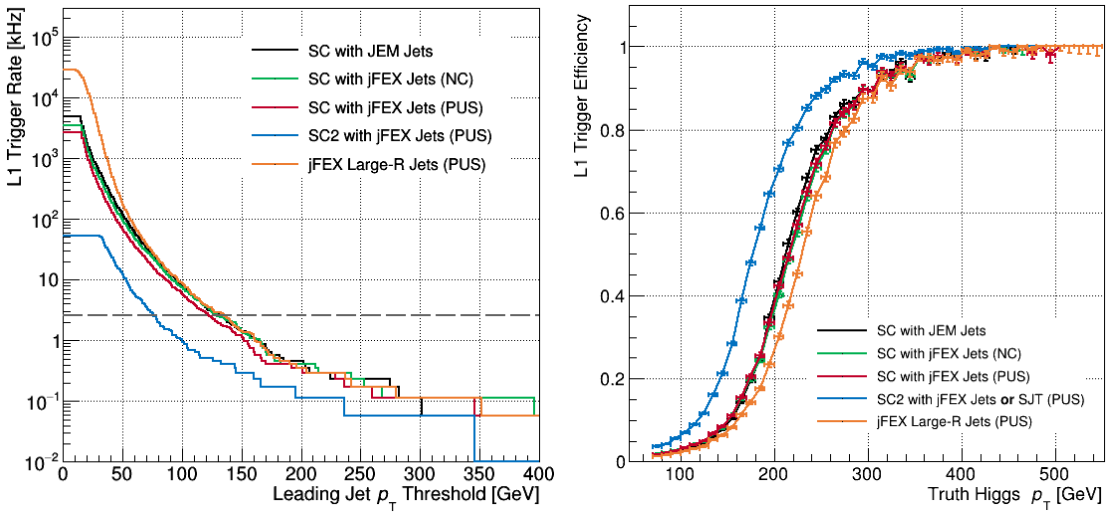

**Figure 10.1:** Per-event efficiencies with respect to the truth Higgs $p_{\mathrm{T}}$, derived from the $VH$ with $V \to 2q$ and $H \to$ inv sample and rate-matched to 4.6 kHz [$E_{\mathrm{T}}^{\mathrm{miss}}$], 2.6 kHz [large-R jets] and 3.7 kHz [JEM and small-R jets] (according to the current Run-3 rate estimates in Ref. [58]). The underlying thresholds for the $E_{\mathrm{T}}^{\mathrm{miss}}$ triggers are obtained from Figure 8.1 (a), while the jet trigger thresholds are taken from Figures 8.2 (a) and 8.4 (a).

## 10.2 Large-R Jet Clustering in Cooperation with L1Topo

Reference [71] shows that the performance of the Run-2 JEM jet trigger worsens as the number of subjets within a jet grows, as a significant fraction of the jet energy becomes more likely to fall outside of the selected window. In order to improve the trigger efficiency for such large-R jets, the so-called *simple cone algorithm* was implemented on the L1Topo processors and already in operation during Run-2. It was designed to sum the transverse energy of all JEM jet TOBs with $E_{\mathrm{T}} > 15\,\mathrm{GeV}$ and center within a cone of $\Delta R = 1$.

Figure 10.2 shows the performance of the simple cone algorithm using different TOBs as input: The rate histogram in Figure 10.2 (a) contains different curves for simple cone jets, for which at least one subjet (SC) and at least two subjets (SC2) are required, while 10.2 (b) reveals the rate-matched efficiencies with respect to the truth Higgs $p_{\mathrm{T}}$.
In reference to the jFEX large-R jet algorithm, all three SC algorithms slightly improve the trigger efficiency. The SC2 algorithm was developed within the scope of this thesis to specifically address the two-jet substructure in the signal process. The resulting efficiency of the pure SC2 algorithm drops in the high $p_{\mathrm{T}}$ regions, which is due to the fact that for some jets only a single subjet could be identified[1]. To compensate for this, the SC2 trigger is ORed with the single small-R jet trigger (SJT). As a result, the SC2 trigger in combination with the SJT becomes significantly more efficient than the SC trigger.

[1] Possible reasons for this are the given limits due to the available granularity and the minimum distance between neighboring objects as a result of the algorithm definition.




**(a)** rates **(b)** per-event efficiencies

**Figure 10.2:** Performance of simple cone algorithms: (a) leading jet rates at different $p_{\mathrm{T}}$ thresholds, derived from the minimum bias sample (JZ0W); (b) per-event efficiencies with respect to the truth Higgs $p_{\mathrm{T}}$, derived from the $VH$ with $V \to 2q$ and $H \to$ inv sample and rate-matched to 2.6 kHz [large-R and simple cone jets] and 3.7 kHz [small-R jets] (according to the current Run-3 rate estimates in Ref. [58]). SC (SC2) denotes simple cone jets with at least one (two) subjet(s) [$p_{\mathrm{T}} > 15$ GeV and $|\eta| < 2.5$ each], while SJT refers to the single small-R jet trigger, for which the underyling threshold is obtained from Figure 8.2 (a).

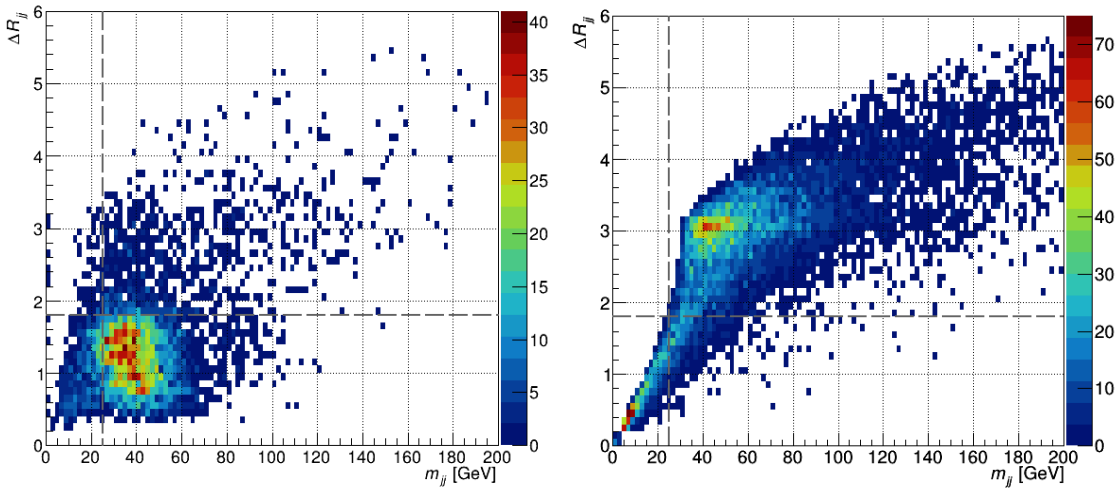
## 10.3 Dijet Requirements in Cooperation with L1Topo

A comparison of Figures 10.1 and 10.2 (b) shows that all three $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms are still more efficient than any of the legacy and newly developed jet algorithms. However, the previous section has shown that addressing the two jets originating from the hadronically decaying $W$ or $Z$ boson leads to an improved efficiency. Possible improvements in the efficiency of the currently best $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm are now examined with the addition of topological requirements on the dijet system:
The invariant mass of the dijet system ($m_{jj}$), as well as the angular separation ($\Delta R_{jj}$), can be computed by the L1Topo processor FPGAs. Since L1Topo does not receive any information about the mass of a jet, it must be set to zero. As a consequence, the invariant mass of jet $i$ and jet $j$ becomes

$$m_{ij}^2 = 2 \cdot (E_{\mathrm{T}})_i \cdot (E_{\mathrm{T}})_j \cdot [\cosh(\Delta\eta_{ij}) - \cos(\Delta\varphi_{ij})] . \tag{10.1}$$

Figure 10.3 (a) shows the distributions and correlation of both quantities for the signal sample ($VH$ with $V \to 2q$ and $H \to$ inv). Since the L1 energy scale does not coincide with the true one, the correct pair of jets was first identified with the help of the anti-$k_{\mathrm{T}}$ $R = 0.4$ truth jets by requiring an invariant mass close to the mass of the vector boson and then applying a $\Delta R$ matching with the online jets, whereby only matches with $\Delta R < 0.4$

were accepted. To keep mismatches with pile-up jets low, only jets with $p_{\mathrm{T}} > 15\,\mathrm{GeV}$ and $|\eta| < 2.5$ were taken into account.
Figure 10.3 (b) shows the situation for the background, where all possible combinations of jet pairs with the same $p_{\mathrm{T}}$ and $\eta$ restrictions are displayed. By setting a lower limit of 25 GeV for $m_{jj}$ and an upper limit of 1.8 for $\Delta R_{jj}$, it is possible to exclude a large fraction of background events, while keeping the signal acceptance high.



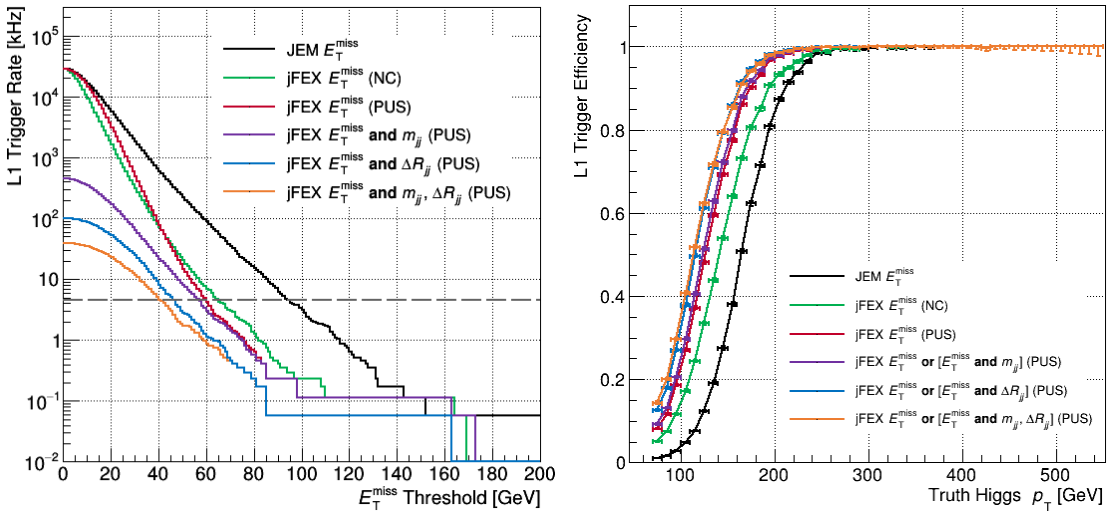

**(a)** signal **(b)** background

**Figure 10.3:** Distributions and correlation of $m_{jj}$ and $\Delta R_{jj}$ for (a) signal and (b) background events, derived from the $VH$ with $V \to 2q$ and $H \to$ inv and the minimum bias (JZ0W) sample, respectively. For the signal events, the dijet system targets the two jets originating from the hadronically decaying vector boson, while all possible dijet combinations are shown for the background. Generally, only jets with $p_{\mathrm{T}} > 15\,\mathrm{GeV}$ and $|\eta| < 2.5$ were taken into account.

Figure 10.4 (a) now shows the L1 trigger rate at different $E_{\mathrm{T}}^{\mathrm{miss}}$ thresholds for the pure $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms and in combination with the dijet requirements defined above. Especially the $\Delta R$ requirement significantly reduces the rate and improves the efficiency in Figure 10.4 (b). To also include events for which the dijet system could not be resolved, the combined trigger turn-on is ORed with the pure $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm. Adding the $m_{jj}$ requirement on top does not seem to have any significant effect.

## 10.4 Final Results

Omitting the jFEX small-R jet algorithms, as they show the least efficient turn-ons for triggering events with invisibly decaying Higgs bosons, which are produced in association with hadronically decaying $W$ or $Z$ bosons, Figure 10.5 shows the histogram, in which the previously discussed trigger efficiencies are collected to be finally compared with one another. Since none of the jet algorithms showed any major differences between the version




**(a)** rates **(b)** per-event efficiencies

**Figure 10.4:** Performance of $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms in combination with dijet requirements ($m_{jj} > 25\,\mathrm{GeV}$, $\Delta R_{jj} < 1.8$): (a) rates at different $E_{\mathrm{T}}^{\mathrm{miss}}$ thresholds, derived from the minimum bias sample (JZ0W); (b) per-event efficiencies with respect to the truth Higgs $p_{\mathrm{T}}$, derived from the $VH$ with $V \to 2q$ and $H \to$ inv sample and rate-matched to 4.6 kHz (according to the current Run-3 rate estimate in Ref. [58]).

with upstream noise cuts and the version with upstream pile-up subtraction, only the latter is displayed for each algorithm type.
The L1 trigger algorithms that were in operation during Run-2 are shown in black ($E_{\mathrm{T}}^{\mathrm{miss}}$) and in different shades of gray (SC and JEM jets). By just comparing the JEM and jFEX jet turn-ons, the jFEX large-R jet algorithm is slightly more efficient towards higher truth Higgs $p_{\mathrm{T}}$. Allowing cooperation with the legacy/new L1Topo system, the SC2 algorithm, which is based on jFEX small-R jets, significantly improves the overall efficiency, and thus even beats the Run-2 SC algorithm.
The best results are still provided by all three $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms, with jFEX $E_{\mathrm{T}}^{\mathrm{miss}}$ clearly dominating. With respect to the legacy trigger, the jFEX $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm with an upstream pile-up subtraction allows to lower the threshold by more than 30 GeV without changing the rate.
Even though the pure jFEX $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm in combination with requirements on the dijet system further improves the efficiency, it must be considered with caution, as there is no corresponding trigger in the current Run-3 rate estimates [58]. First, it must be clarified how this trigger correlates with the $E_{\mathrm{T}}^{\mathrm{miss}}$ related triggers that are already part of the menu. Fortunately, the required topological algorithms (even a combination of both) are already in place and, at this point in time, planned to be part of the new L1Topo algorithm firmware.

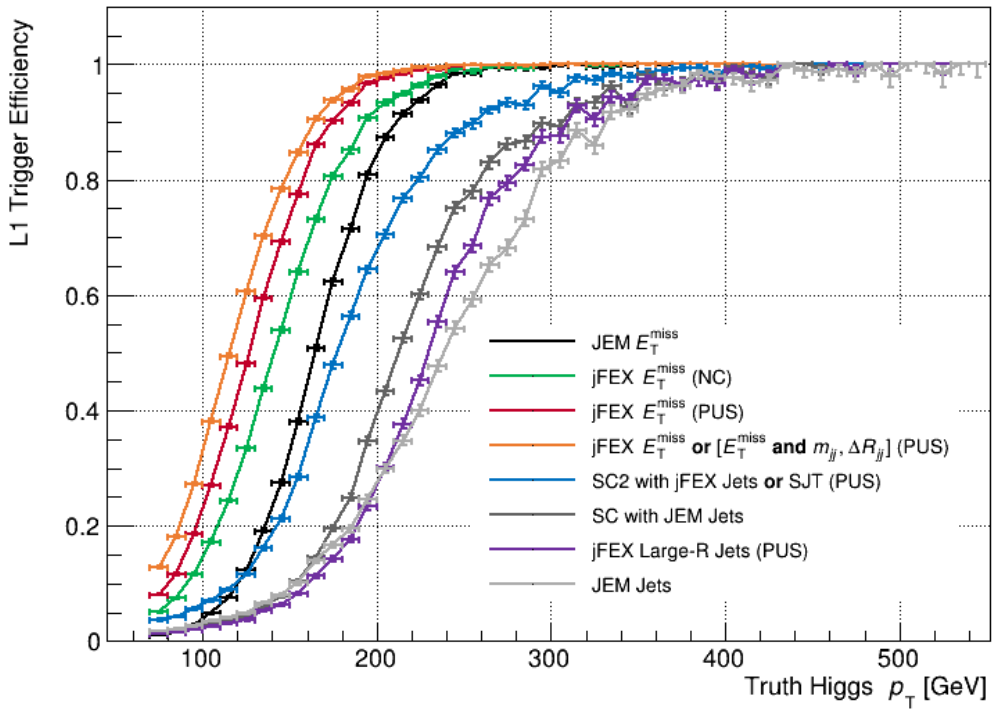



**Figure 10.5:** Per-event efficiencies with respect to the truth Higgs $p_T$, derived from the $VH$ with $V \to 2q$ and $H \to$ inv sample and rate-matched to 4.6 kHz [$E_T^{miss}$ and $E_T^{miss}$ in combination with dijet requirements ($m_{jj} > 25$ GeV, $\Delta R_{jj} < 1.8$)], 2.6 kHz [simple cone and large-R jets] and 3.7 kHz [JEM and small-R jets] (according to the current Run-3 rate estimates in Ref. [58]). The underlying thresholds for the $E_T^{miss}$ related triggers are obtained from Figure 10.4 (a), while the jet trigger thresholds are taken from Figures 10.2 (a) and 8.2 (a). SC (SC2) denotes simple cone jets with at least one (two) subjet(s) [$p_T > 15$ GeV and $|\eta| < 2.5$ each], while SJT refers to the single small-R jet trigger. The same restrictions on $p_T$ and $\eta$ are applied for each jet in the dijet system.

# CHAPTER 11

# Summary and Outlook

*No matter how many instances*
*of white swans we may have observed,*
*this does not justify the conclusion*
*that all swans are white.*

( Sir Karl R. Popper )

To cope with the increased luminosity delivered by the LHC in 2022, the ATLAS experiment has planned major upgrades, quantified as Phase-I upgrades. As part of this, the Level-1 Calorimeter (L1Calo) Trigger system is redesigned taking advantage of the latest FPGA technology in order to exploit higher granularity data from the calorimeters. It is equipped with three new subsystems, namely the FEXs, each optimized to trigger on different physics objects.

The jet FEX (jFEX) is mainly intended to identify jets and to calculate the total transverse energy sum and missing transverse energy ($E_{\mathrm{T}}^{\mathrm{miss}}$). For a full calorimeter coverage ($|\eta| < 4.9$), six jFEX modules are required, where each hosts four processor FPGAs. In particular, the FPGAs on different modules have to cope with different conditions in terms of energy resolution/range and granularity/irregularity of the incoming data.

In the context of this thesis, new algorithm firmware was developed for the entire jFEX system: It was shown that the newly developed trigger algorithms were implemented appropriately on the target FPGA by requiring a maximum processing time of 125 ns, which is well within the available latency budget of 150 ns. Their correct behavior was verified in hardware tests and validated in logic simulations.

Moreover, results from corresponding trigger efficiency studies were presented on object-level, showing several improvements over the legacy system: The newly developed and implemented jFEX jet algorithms greatly improve the overall identification of large jets and the differentiation of close-by jets ($\Delta R_{jj} < 0.6$) over a wide energy range. Furthermore, an implementation based on lists has been developed for the outer modules (at large $|\eta|$), to also properly identify jets across regions with different/irregular granularity. An event-by-event pile-up calculation and subtraction on jFEX primarily improves the $E_{\mathrm{T}}^{\mathrm{miss}}$ efficiency, as it allows to significantly lower the underlying trigger threshold without changing the rate.

Finally, the individually achieved results were brought together to determine the best trigger efficiency targeting events with invisibly decaying Higgs bosons, which are produced in association with hadronically decaying $W$ or $Z$ bosons. By keeping the rate at a constant

level, the trigger threshold can be reduced by more than 30 GeV when switching from the legacy $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm to the best available version developed for and implemented in jFEX. It has also been shown that additional requirements on the jets in the final state can further improve the overall efficiency. As a consequence, the lower trigger threshold for $E_{\mathrm{T}}^{\mathrm{miss}}$ and the improved identification/differentiation of jets have great potential to increase the sensitivity to physically interesting events in all ATLAS analyses that require $E_{\mathrm{T}}^{\mathrm{miss}}$ and/or jets.

The first jFEX production module has been at CERN since September 2020. The assembly of the remaining boards is currently in progress. They will be installed along with all other new subsystems over the next few months.
In parallel, a bitwise accurate simulation of the new L1Calo system is being developed. Among other things, it is used for commissioning, monitoring and validation of the new trigger algorithms. As soon as the jFEX infrastructure firmware is ready and fully integrated with both readout and algorithm firmware, the latter can be officially validated using the bit-level simulation.

All three FEX systems will continue to operate after the next long shutdown of the LHC, which begins at the end of 2024 and will last for more than two years. During this time and as part of the Phase-II upgrade, a fourth FEX system called forward Feature EXtractor (fFEX) [72] will be added. In parallel to the other FEXs, fFEX is supposed to identify electrons and jets in the very forward regions ($|\eta| > 2.5$) based on the full detector granularity, which in particular results in a much finer segmentation in $\varphi$.
At this point in time, it is planned to port the jFEX jet algorithms, but to use the data with higher granularity as input. Fortunately, the jFEX jet algorithms in the forward regions rely on the list-based implementation, which is easily customizable. From a technical point of view, essentially only the content of the lists and the interface to the entity have to be modified.

# Acronyms

**ALICE** A Large Ion Collider Experiment 17

**ATCA** Advanced Telecommunications Computing Architecture 63

**ATLAS** A Toroidal LHC ApparatuS 17–23, 25, 28–31, 34–37, 39, 44, 59, 100, 101, 111, 112, 123, 125

**BRAM** Block RAM 43, 63, 96, 100–103, 123

**CKM matrix** Cabibbo-Kobayashi-Maskawa matrix 9, 10

**CLB** Configurable Logic Block 40–44, 53, 54, 126

**CMOS** Complementary MOS 39, 40

**CMS** Compact Muon Solenoid 17

**CMX** Common Merger module eXtended 32, 34

**CP** Cluster Processor 32, 44

**CPM** Cluster Processor Module 32, 33

**CRC** cyclic redundancy check 66

**CSC** Cathode Strip Chamber 28, 29

**CTP** Central Trigger Processor 34, 36

**DAQ** data acquisition 30, 34, 59, 63, 81

**DSP** digital signal processing 42–45, 55, 88, 90, 99, 100, 102, 123, 126

**eFEX** electromagnetic Feature EXtractor 36, 59, 75, 78

**EMEC** EM End-Cap 24, 78

**FCal** Forward Calorimeter 24–27, 60, 61, 70–73, 75, 77–79, 92, 95, 96, 98, 100, 123, 125–128

**FEX** Feature EXtractor 35, 36, 44, 63, 111, 112

**FF** Flip-Flop 40–42, 44, 52, 53, 99–102, 123, 126

**fFEX** forward Feature EXtractor 112

**FOX** Fibre Optical Exchange 35, 36

**FPGA** Field Programmable Gate Array 30, 35, 36, 39, 40, 42, 44, 45, 47–50, 53–55, 59–61, 63, 64, 66, 67, 71, 73–77, 79, 80, 82, 87–100, 102–105, 107, 111, 123, 124, 126–128

**gFEX** global Feature EXtractor 36, 59

**HDL** hardware description language 47, 49, 50

**HEC** HAD End-Cap 24, 25, 78

**HLT** High Level Trigger 34, 59, 63, 75

**IBL** Insertable B-Layer 23, 34

**ILA** Integrated Logic Analyzer 55, 96, 100, 103

**IOB** I/O Block 44

**JEM** Jet/Energy Module 32, 34, 74, 82–84, 105, 106, 109, 110, 128, 129

**JEP** Jet Energy Processor 32, 44

**jFEX** jet Feature EXtractor 36, 40, 59, 60, 62–65, 67, 69–71, 73–78, 80–84, 87, 88, 93, 94, 96, 98, 100–106, 108, 109, 111, 112, 123, 124, 126–128

**L1Topo** Level-1 Topological Processor 34, 36, 59, 62, 63, 66, 69, 70, 73, 75, 78, 87, 93, 94, 97, 103, 105–107, 109, 128

**LATOME** LAr Trigger prOcessing MEzzanine 35, 59–62, 66, 69–71, 73, 87, 89, 98, 100, 101, 103, 123

**LEIR** Low Energy Ion Ring 16

**LHC** Large Hadron Collider 15–17, 19, 34, 62, 69, 111, 112

**LHCb** Large Hadron Collider beauty 17

**LINAC 2** Linear Accelerator 2 16

**LINAC 4** Linear Accelerator 4 16

**LSB** least significant bit 61, 73, 74, 80, 87, 89, 90, 92, 123

**LUT** Lookup Table 40–44, 52, 53, 88, 90, 95, 99, 100, 102, 123, 126

**MDT** Monitored Drift Tube 28–30

**MGT** Multi-Gigabit Transceiver 44, 61, 64, 66, 100–102, 123

**MOS** Metal Oxide Semiconductor 39

**MSB** most significant bit 61

**MUCTPI** MUon to Central Trigger Processor Interface 32, 34

**MUX** Multiplexer 40–42, 66

**NSW** New Small Wheel 35

**PS** Proton Synchrotron 16

**PSB** Proton Synchrotron Booster 16

**QCD** Quantum Chromodynamics 7, 10, 37

**QED** Quantum Electrodynamics 8, 10

**RAM** Random-Access Memory 43, 66

**ROD** Readout Driver 31, 34

**RoI** Region of Interest 32–34, 63, 74–79, 91, 126, 127

**ROS** Readout System 34

**RPC** Resistive Plate Chamber 28–30, 32

**RTL** register transfer level 49, 50, 52, 126

**SCT** Semi-Conductor Tracker 18, 22, 24

**SLL** Super Logic Line 45

**SLR** Super Logic Region 45, 126

**SPS** Super Proton Synchrotron 16

**SRAM** Static Random-Access Memory 40

**SSI** Stacked Silicon Interconnect 44, 45, 126

**TDAQ** Trigger and DAQ 30, 31, 35, 39, 44, 59, 125

**TGC** Thin Gap Chamber 28–30, 32

**TOB** Trigger Object 32, 34, 36, 59, 62, 63, 69, 70, 74, 79, 80, 87, 92–99, 101, 103, 106, 123, 128

**TopoFOX** L1Topo Fibre Optical Exchange 36

**TREX** Tile Rear Extension 35, 59, 61, 66, 69–71, 73, 87, 90, 98, 100, 101, 103

**TRT** Transition Radiation Tracker 18, 22, 24

**URAM** Ultra RAM 43, 66, 101, 102

**WHS** worst hold slack 55

**WNS** worst negative slack 55

**xTOB** extended TOB 63, 79, 101

# A

# Bibliography


[1] ATLAS Collaboration. *Observation of an Excess of Events in the Search for the Standard Model Higgs boson with the ATLAS detector at the LHC.* ATLAS Note, ATLAS-CONF-2012-093, July 2012; URL: `http://cdsweb.cern.ch/record/1460439/`.

[2] CMS Collaboration. *Observation of a new boson with a mass near 125 GeV.* CMS Physics Analysis Summaries, CMS-PAS-HIG-12-020, July 2012; URL: `https://cds.cern.ch/record/1460438/`.

[3] B. Povh et al. *Teilchen und Kerne - Eine Einführung in die physikalischen Konzepte.* 9. Auflage, Springer Spektrum, 2014; ISBN: 978-3-642-37821-8.

[4] C. Berger. *Elementarteilchenphysik - Von den Grundlagen zu den modernen Experimenten.* 3., aktualisierte und überarbeitete Auflage, Springer Spektrum, 2014; ISBN: 978-3-642-41752-8.

[5] J. Bleck-Neuhaus. *Elementare Teilchen - Von den Atomen über das Standard-Modell bis zum Higgs-Boson.* 2. Auflage, Springer Spektrum, 2013; ISBN: 978-3-642-32578-6.

[6] A. Purcell. *Go on a particle quest at the first CERN webfest.* Bulletin Issue, BUL-NA-2012-269, August 2012; URL: `http://cds.cern.ch/record/1473657/`; Image file: `http://cds.cern.ch/record/1473657/files/SMinfographic_image.png`.

[7] P. A. Zyla et al. (Particle Data Group). *The Review of Particle Physics (2020).* Prog. Theor. Exp. Phys. 2020, 083C01 (2020); URL: `https://pdg.lbl.gov/`.

[8] LHC Higgs Working Group (LHCHWG). *SM Higgs Branching Ratios and Total Decay Widths.* URL: `https://twiki.cern.ch/twiki/bin/view/LHCPhysics/CERNYellowReportPageBR#Branching_Ratios`.

[9] CERN. *Accelerating science.* URL: `https://home.cern/`.

[10] ATLAS Experiment at CERN. URL: `https://atlas.cern/`; Image files: `https://atlas.cern/resources/multimedia/`.

[11] ATLAS Collaboration et al. *The ATLAS Experiment at the CERN Large Hadron Collider.* 2008 JINST 3 S08003; DOI: 10.1088/1748-0221/3/08/S08003.

[12] E. Mobs. *The CERN accelerator complex.* CERN-GRAPHICS-2019-002, 2019; URL: https://cds.cern.ch/record/2684277/; Image file: https://cds.cern.ch/record/2684277/files/CCC-v2019-final-white.png.

[13] J. J. Goodson. *Search for Supersymmetry in States with Large Missing Transverse Momentum and Three Leptons including a Z-Boson.* Thesis, CERN-THESIS-2012-053, April 2012; URL: http://cds.cern.ch/record/1449722/.

[14] K. Potamianos. *The upgraded Pixel detector and the commissioning of the Inner Detector tracking of the ATLAS experiment for Run-2 at the Large Hadron Collider.* ATLAS Note, ATL-PHYS-PROC-2016-104; arXiv:1608.07850 [physics.ins-det].

[15] ATLAS Collaboration. *ATLAS liquid-argon calorimeter: Technical Design Report.* Scientific Committee Paper, CERN-LHCC-96-041, ATLAS-TDR-2, December 1996; URL: https://cds.cern.ch/record/331061/.

[16] H. Kolanoski et al. *Teilchendetektoren - Grundlagen und Anwendungen.* Springer Spektrum, 2016; ISBN: ISBN 978-3-662-45349-0.

[17] ATLAS Collaboration et al. *Performance of the ATLAS muon trigger in pp collisions at $\sqrt{s} = 8$ TeV.* Eur. Phys. J. C 75, 120 (2015); DOI: 10.1140/epjc/s10052-015-3325-9; Image file: https://cds.cern.ch/record/1749694/files/muon_schematic.png.

[18] R. Achenbach et al. *The ATLAS Level-1 Calorimeter Trigger.* 2008 JINST 3 P03001; DOI: 10.1088/1748-0221/3/03/P03001.

[19] ATLAS Collaboration et al. *The ATLAS Simulation Infrastructure.* Eur. Phys. J. C 70, 823–874 (2010); DOI: 10.1140/epjc/s10052-010-1429-9.

[20] S. Agostinelli et al. *GEANT4 - a simulation toolkit.* Nucl.Instrum.Meth.A 506 (2003) 250-303; DOI: 10.1016/S0168-9002(03)01368-8.

[21] M. Cacciari et al. *The anti-$k_t$ jet clustering algorithm.* LPTHE-07-03, JHEP 0804:063, 2008; arXiv:0802.1189 [hep-ph].

[22] K. Fricke. *Digitaltechnik - Lehr- und Übungsbuch für Elektrotechniker und Informatiker.* 8., überarbeitete und aktualisierte Auflage, Springer Vieweg, 2018; ISBN: 978-3-658-21065-6.

[23] German Research Center for Artificial Intelligence GmbH. *Robotics Innovation Center.* URL: https://robotik.dfki-bremen.de/en/research/softwaretools/respace.html; Original image file: https://robotik.dfki-bremen.de/uploads/tx_dfkiprojects/fpga_explained.gif.

[24] Xilinx Inc. URL: https://www.xilinx.com/.

[25] Xilinx Inc. *Virtex UltraScale+ devices.* URL: https://www.xilinx.com/products/silicon-devices/fpga/virtex-ultrascale-plus.html.

[26] Xilinx Inc. *UltraScale Architecture Configurable Logic Block (User Guide).* UG574 (v1.5) February 28, 2017; URL: https://www.xilinx.com/support/documentation/user_guides/ug574-ultrascale-clb.pdf.

[27] Xilinx Inc. *UltraScale Architecture Libraries Guide.* UG974 (v2018.3) December 5, 2018; URL: https://www.xilinx.com/support/documentation/sw_manuals/xilinx2018_3/ug974-vivado-ultrascale-libraries.pdf.

[28] Xilinx Inc. *UltraScale Architecture DSP Slice (User Guide).* UG579 (v1.9) September 20, 2019; URL: https://www.xilinx.com/support/documentation/user_guides/ug579-ultrascale-dsp.pdf.

[29] Xilinx Inc. *UltraScale Architecture Memory Resources (User Guide).* UG573 (v1.11) August 18, 2020; URL: https://www.xilinx.com/support/documentation/user_guides/ug573-ultrascale-memory-resources.pdf.

[30] Xilinx Inc. *UltraScale Architecture Clocking Resources (User Guide).* UG572 (v1.9) October 31, 2019; URL: https://www.xilinx.com/support/documentation/user_guides/ug572-ultrascale-clocking.pdf.

[31] Xilinx Inc. *UltraScale and UltraScale+ FPGAs Packaging and Pinouts (Product Specification).* UG575 (v1.14) March 18, 2020; URL: https://www.xilinx.com/support/documentation/user_guides/ug575-ultrascale-pkg-pinout.pdf.

[32] Xilinx Inc. *UltraScale Architecture SelectIO Resources (User Guide).* UG571 (v1.12) August 28, 2019; URL: https://www.xilinx.com/support/documentation/user_guides/ug571-ultrascale-selectio.pdf.

[33] Xilinx Inc. *UltraScale Architecture GTY Transceivers (User Guide).* UG578 (v1.3) September 20, 2017; URL: https://www.xilinx.com/support/documentation/user_guides/ug578-ultrascale-gty-transceivers.pdf.

[34] Xilinx Inc. *Virtex™-E 1.8 V Field Programmable Gate Arrays (Preliminary Product Specification).* DS022-1 (v2.2) November 9, 2001; URL: https://datasheet.octopart.com/XCV1000E-6BG560I-Xilinx-datasheet-9523.pdf.

[35] Xilinx Inc. *Virtex-II Platform FPGAs: Complete Data Sheet (Product Specification).* DS031 (v3.5) November 5, 2007; URL: https://datasheet.octopart.com/XC2V3000-6FG676C-Xilinx-datasheet-13696474.pdf.

[36] Xilinx Inc. *7 Series FPGAs Data Sheet: Overview (Product Specification).* DS180 (v2.6) February 27, 2018; URL: https://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf.

[37] Xilinx Inc. *UltraScale Architecture and Product Data Sheet: Overview (Product Specification).* DS890 (v3.13) July 21, 2020; URL: https://www.xilinx.com/support/documentation/data_sheets/ds890-ultrascale-overview.pdf.

[38] K. Saban. *Xilinx Stacked Silicon Interconnect Technology Delivers Breakthrough FPGA Capacity, Bandwidth, and Power Efficiency.* WP380 (v1.2) December 11, 2012; URL: https://www.xilinx.com/support/documentation/white_papers/wp380_Stacked_Silicon_Interconnect_Technology.pdf.

[39] Xilinx Inc. *Large FPGA Methodology Guide; Including Stacked Silicon Interconnect (SSI) Technology.* UG872 (v14.3) October 16, 2012; URL: https://www.xilinx.com/support/documentation/sw_manuals/xilinx14_7/ug872_largefpga.pdf.

[40] R. Gessler. *Entwicklung Eingebetteter Systeme - Vergleich von Entwicklungsprozessen für FPGA- und Mikroprozessor-Systeme Entwurf auf Systemebene.* 2., aktualisierte und erweiterte Auflage, Springer Vieweg, 2020; ISBN: 978-3-658-30548-2.

[41] Xilinx Inc. *Vivado Design Suite.* URL: `https://www.xilinx.com/products/design-tools/vivado.html`.

[42] Xilinx Inc. *Vivado Design Suite User Guide - Design Flow Overview.* UG892 (v2017.2) June 7, 2017; URL: `https://www.xilinx.com/support/documentation/sw_manuals/xilinx2017_2/ug892-vivado-design-flows-overview.pdf`.

[43] Xilinx Inc. *Integrated Logic Analyzer v6.2 - LogiCORE IP Product Guide.* Vivado Design Suite; PG172 October 5, 2016; URL: `https://www.xilinx.com/support/documentation/ip_documentation/ila/v6_2/pg172-ila.pdf`.

[44] G. Aad et al. *ATLAS LAr Calorimeter trigger electronics phase I upgrade: LATOME Firmware Specifications.* Internal Note, ATL-COM-LARG-2017-052, November 2017; URL: `https://cds.cern.ch/record/2294187/`.

[45] M. Palka. *L1Calo jFEX readout specification.* Internal Note, ATL-COM-DAQ-2018-057, May 2018; URL: `https://cds.cern.ch/record/2319207/`.

[46] PCI Industrial Computer Manufacturers Group (PICMG). *Open Modular Computing Standards - AdvancedTCA.* URL: `https://www.picmg.org/openstandards/advancedtca/`.

[47] Panasonic Corp. *Ultra-low transmission loss Highly heat resistant Multilayer circuit board materials MEGTRON6 | R-5775(N), R-5775.* URL: `https://industrial.panasonic.com/ww/products/electronic-materials/circuit-board-materials/megtron/megtron6`.

[48] Avago Technologies. *MiniPOD™ (Product Brief).* AV02-4039EN - March 8, 2013; URL: `http://www.fit-foxconn.com/Images/Products/Spec/AFBR-824VxyZ_20160510175057487.pdf`.

[49] C. G. Larrea et al. *IPbus: a flexible Ethernet-based control system for xTCA hardware.* 2015 JINST 10 C02019; URL: `https://iopscience.iop.org/article/10.1088/1748-0221/10/02/C02019`.

[50] B. Bauss et al. *Project Specification and Hardware Tests of the jet Feature Extractor (jFEX) in ATLAS Level-1 Calorimeter Trigger for TDAQ Phase-I Upgrade (Report for Production Readiness Review).* Internal Note, ATL-COM-DAQ-2019-137, August 2019; URL: `https://cds.cern.ch/record/2687132/`.

[51] H. W. Lang. *Codierungstheorie: CRC-Verfahren.* URL: `https://www.inf.fh-flensburg.de/lang/algorithmen/code/crc/crc.htm`.

[52] World Wide Web Consortium (W3C). *XML Technology.* URL: `https://www.w3.org/standards/xml/`.

[53] ATLAS Geometry Database. *LAr FCal Electrodes*. URL: https://atlas-geometry-db.web.cern.ch/atlas-geometry-db/tag_hierarchy_browser.php; Tag: ATLAS-R2-2016-01-01-00/LAr-Revised-18/LArFCalElectrodes-06.

[54] A. Artamonov et al. *The ATLAS Forward Calorimeter*. 2008 JINST 3 P02010; URL: https://iopscience.iop.org/article/10.1088/1748-0221/3/02/P02010.

[55] N. Nottbeck. *Implementierung und Optimierung des Anti-$k_T$-Jet-Clustering-Algorithmus' auf FPGAs*. Bachelorarbeit, Johannes Gutenberg-Universität Mainz, 2016.

[56] ATLAS L1Calo Collaboration. *Documentation of the L1Calo algorithm specifications*. ATLAS Note, Draft version 0.1, September 2020; URL: https://gitlab.cern.ch/l1calo-run3-simulation/documentation/Run3L1CaloOfflineSWReqs/.

[57] B. Salvachua. *Overview of Proton-Proton Physics During Run II*. Proceeding, 9th LHC Operations Evian Workshop, 2019; URL: https://indico.cern.ch/event/751857/contributions/3259373/attachments/1783143/2977752/2019_01_28_Evian_pp_run-2.pdf.

[58] A. Ruiz-Martinez et al. *Run 3 trigger menu design*. Internal Note, ATL-COM-DAQ-2019-116, July 2019; URL: https://cds.cern.ch/record/2683881.

[59] ATLAS L1Calo Collaboration. *GitLab Repository for the Firmware of the Phase-0 L1Topo*. URL: https://gitlab.cern.ch/atlas-l1calo/l1topo/ph0topo/.

[60] S. Maldaner. *Algorithmenentwicklung für den topologischen Trigger am ATLAS Detektor*. Diplomarbeit, Johannes Gutenberg-Universität Mainz, 2013.

[61] N. Gee. *Phase-I Latency Envelopes for the Level-1 Trigger*. URL: https://edms.cern.ch/document/1256858/1.

[62] ATLAS L1Calo Collaboration. *GitLab Repository for the Firmware of the jFEX*. URL: https://gitlab.cern.ch/atlas-l1calo/jfex/jfex_firmware/.

[63] R. Wang. Private Communication, 2021.

[64] Xilinx Inc. *VCU118 Evaluation Board (User Guide)*. UG1224 (v1.4) October 17, 2018; URL: https://www.xilinx.com/support/documentation/boards_and_kits/vcu118/ug1224-vcu118-eval-bd.pdf.

[65] Xilinx Inc. *Platform Cable USB II*. DS593 (v1.5.1) August 6, 2018; URL: https://www.xilinx.com/support/documentation/data_sheets/ds593.pdf.

[66] Texas Instruments. *LVDS Application and Data Handbook*. SLLD009, November 2002; URL: https://www.ti.com/lit/ug/slld009/slld009.pdf.

[67] Xilinx Inc. *Clocking Wizard v6.0 - LogiCORE IP Product Guide*. Vivado Design Suite; PG065 February 25, 2019; URL: https://www.xilinx.com/support/documentation/ip_documentation/clk_wiz/v6_0/pg065-clk-wiz.pdf.

[68] Xilinx Inc. *Block Memory Generator v8.4 - LogiCORE IP Product Guide*. Vivado Design Suite; PG058 December 9, 2019; URL: https://www.xilinx.com/support/documentation/ip_documentation/blk_mem_gen/v8_4/pg058-blk-mem-gen.pdf.

[69] J. Vieira De Souza. *Picture of the jFEX first production module*. Original image file: https://twiki.cern.ch/twiki/pub/Atlas/STFjFEXL1Topo/jfex_first_production.jpeg.

[70] R. Wang. In Joint Effort, 2021.

[71] J. H. Stelzer et al. *Performance of the ATLAS Level-1 topological trigger in Run 2*. Internal Note, ATL-COM-DAQ-2020-026, April 2020; URL: https://cds.cern.ch/record/2714861.

[72] J. Fischer et al. *fFEX Overview*. October 14, 2020; URL: https://indico.cern.ch/event/958563/contributions/4029597/attachments/2122954/3573567/ffex-uli.pdf.

# B

# List of Tables

# C

# List of Figures

# D

# List of Listings

# E

# Danksagung

[in der elektronischen Version aus Datenschutzgründen entfernt]

F

# Curriculum Vitae

[in der elektronischen Version aus Datenschutzgründen entfernt]

# G

# Appendix

*The FCal is fairly pathological*
*when it comes to L1 triggering*
*and in the past we just ignored the reality,*
*made simplifications and hoped for the best.*
*And 10 years on,*
*we still don't really understand it properly.*
*Our experience in the past was*
*that FCal represented 1/50 of the towers*
*and 50% of the problems.*

( Murrough P. J. Landon )

## Readout Electrodes in the ATLAS Forward Calorimeters

The coordinates of the readout electrodes in the three layers of the ATLAS Forward Calorimeters are shown in Figures G.1, G.2 and G.3. The density in the positioning of the electrodes decreases from the first to the last layer.

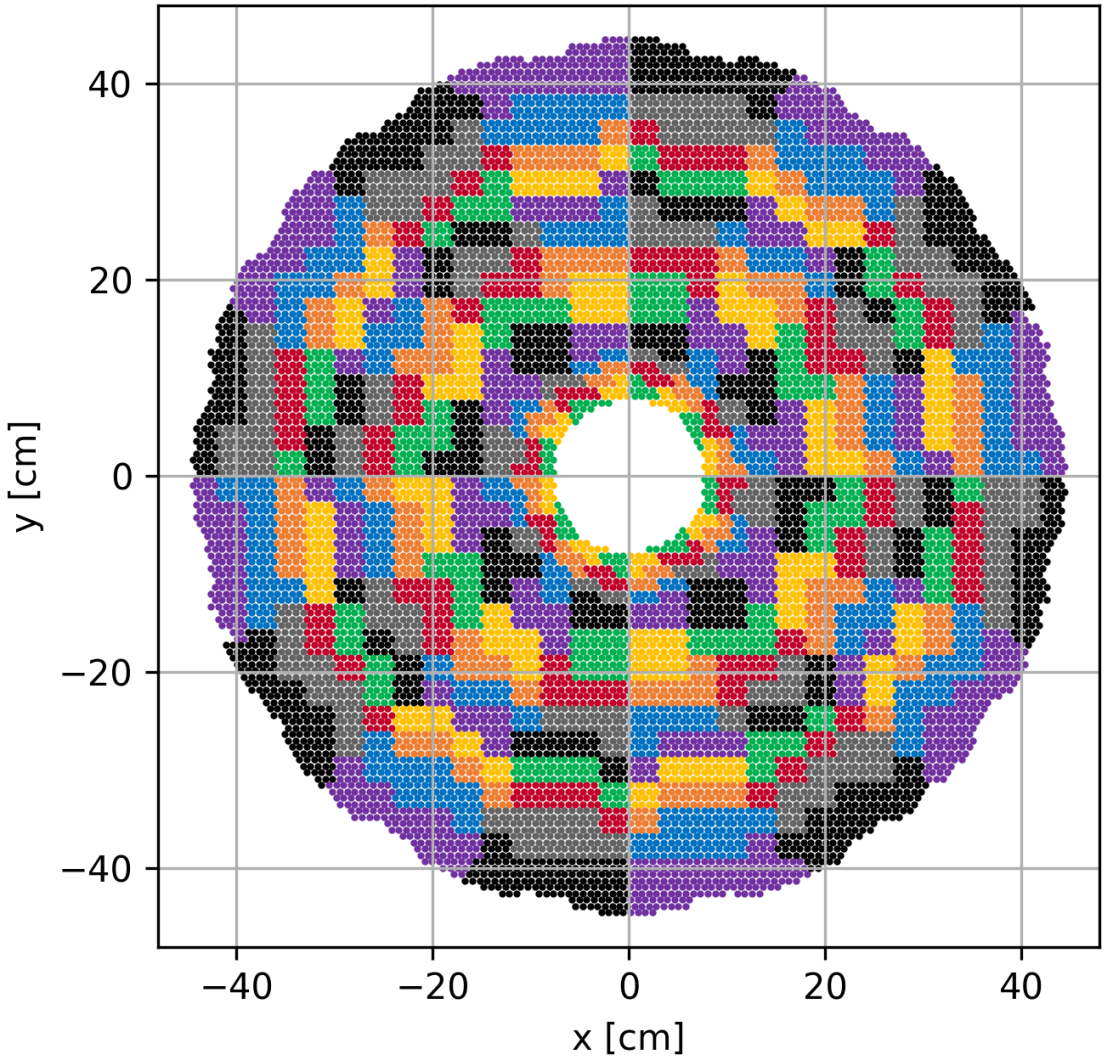



**Figure G.1:** Readout electrodes in the 1st layer of the ATLAS Forward Calorimeters. Each point represents a single electrode, while the alternating color scheme separates each group of electrodes that make up a super cell.

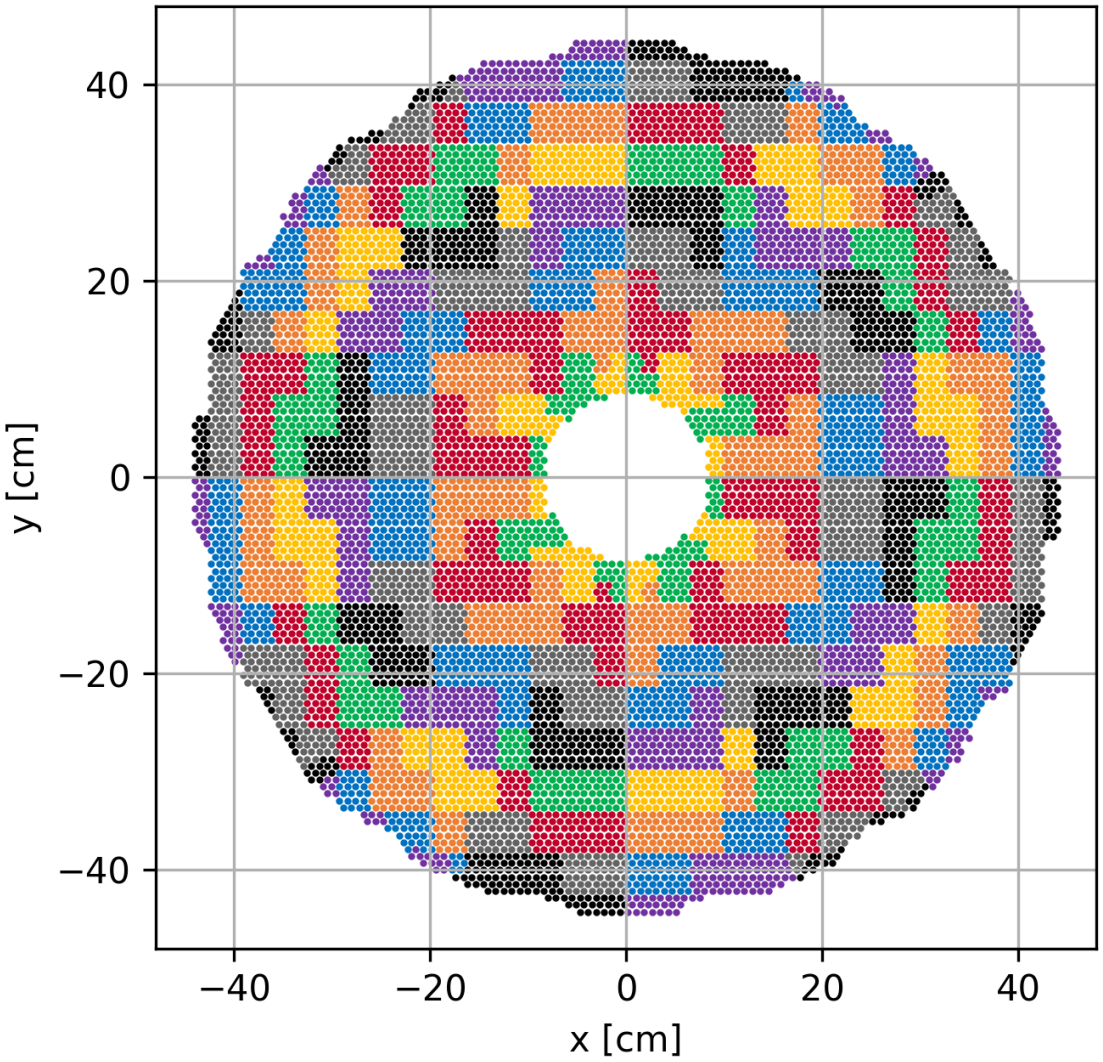



**Figure G.2:** Readout electrodes in the 2nd layer of the ATLAS Forward Calorimeters. Each point represents a single electrode, while the alternating color scheme separates each group of electrodes that make up a super cell.

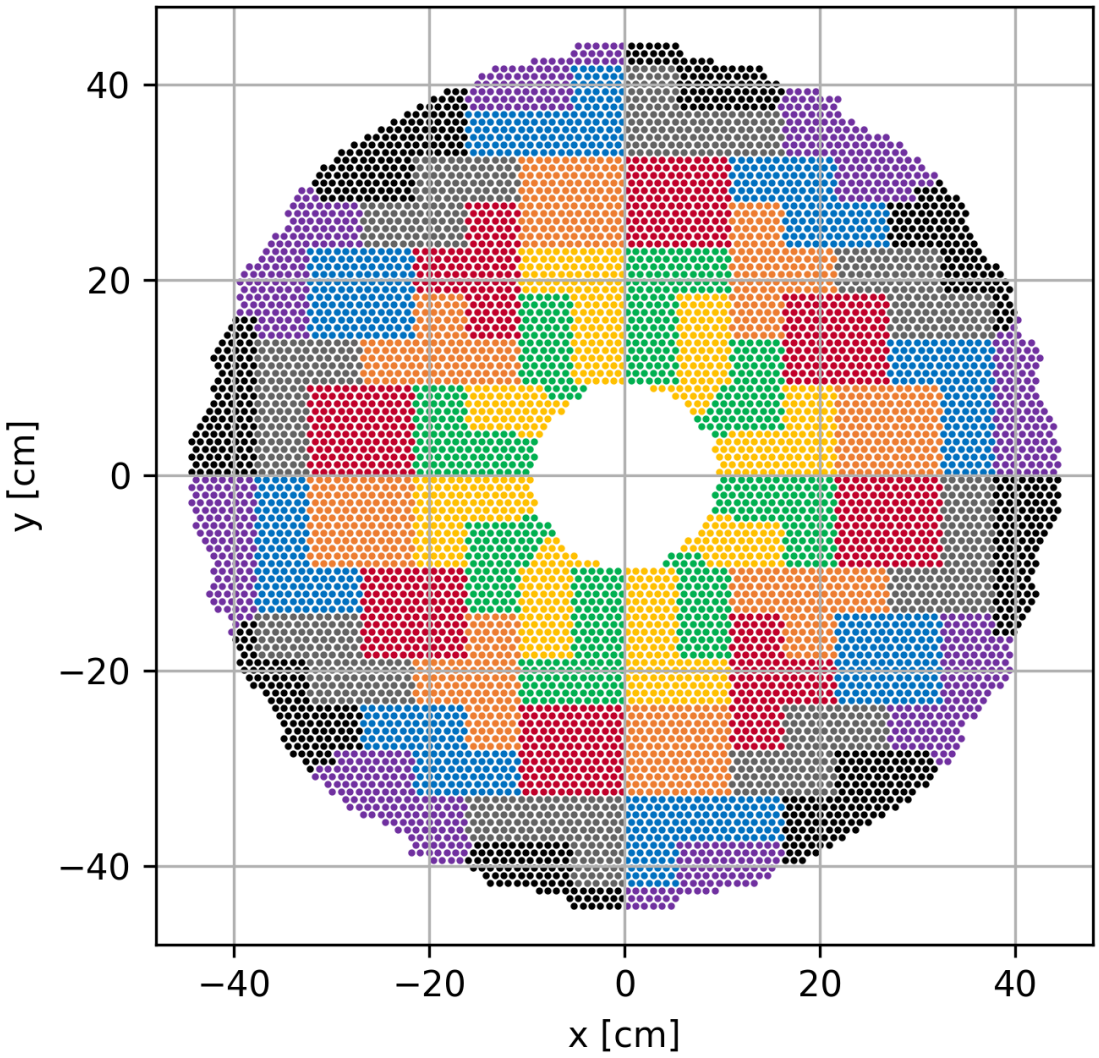



**Figure G.3:** Readout electrodes in the 3rd layer of the ATLAS Forward Calorimeters. Each point represents a single electrode, while the alternating color scheme separates each group of electrodes that make up a super cell.

*Part of the journey is the end.*
( Tony Stark | Avengers: Endgame )