Test beam studies of pixel detector prototypes for the ATLAS-Experiment at the High Luminosity Large Hadron Collider

Dissertation

zur Erlangung des mathematisch-naturwissenschaftlichen Doktorgrades
„Doctor rerum naturalium"
der Georg-August-Universität Göttingen

im Promotionsprogramm ProPhys
der Georg-August University School of Science (GAUSS)

vorgelegt von

Tobias Bisanz

aus Linz, Österreich

Göttingen, 2018

Betreuungsausschuss

Prof. Dr. Arnulf Quadt
PD. Dr. Jörn Große-Knetter
Prof. Dr. Ariane Frey

Mitglieder der Prüfungskommission:

Referent: Prof. Dr. Arnulf Quadt
II. Physikalisches Institut, Georg-August-Universität Göttingen

Korreferent: Prof. Dr. Gerald Eigen
Department of Physics and Technology, University of Bergen

Weitere Mitglieder der Prüfungskommission:

Prof. Dr. Steffen Schumann
II. Physikalisches Institut, Georg-August-Universität Göttingen

Prof. Dr. Stan Lai
II. Physikalisches Institut, Georg-August-Universität Göttingen

Prof. Dr. Hans-Christian Hofsäss
II. Physikalisches Institut, Georg-August-Universität Göttingen

Prof. Dr. Michael Seibt
IV. Physikalisches Institut, Georg-August-Universität Göttingen

Tag der mündlichen Prüfung: 18.12.2018

Referenz: II.Physik-UniGö-Diss-2018/03

Test beam studies of pixel detector prototypes for the ATLAS-Experiment at the High Luminosity Large Hadron Collider

Abstract

The upgrade of the Large Hadron Collider (LHC) in the mid-2020's to the High Luminosity Large Hadron Collider will provide large amounts of data, enabling precision measurements of Standard Model processes and searches for new physics. This will also maximise the physics potential of the experiments located at the LHC. In order to record the desired integrated luminosity, the ATLAS detector will face challenges regarding the radiation damage, pile-up and amount of data. To cope with those challenges, the entire tracking detector is replaced by the new *Inner Tracker*, ITk.

In order to develop novel detector modules, test beam measurements are a crucial tool to study and understand new read-out chips, novel sensor technologies, as well as the effect of radiation on the sensor and read-out chip. Not only in the R&D-phase but also in the phase of module production, test beams need to play a necessary role in the continuous quality assurance procedure.

This thesis is focused on the test beam measurements of pixel modules for the ITk upgrade. Several aspects of test beams are covered and have been improved. Modifications to one of the used DAQ systems at test beams (USBpix) have been made. A new measurement technique using the USBpix system to make so-called in-time measurements at test beams has been developed. The measurement method as well as results from such a measurement are discussed in this thesis. Furthermore, many parts of the reconstruction framework have been modernised and improved. The track fit based on the General-Broken-Line algorithm now allows track reconstruction together with ATLAS pixel modules. Also, test cases were implemented to ensure long-term stability, and guarantee a consistent reconstruction, even in future versions. Additionally, a Monte Carlo framework was expanded to enable validation of the reconstruction algorithms.

Test beam studies of pixel detector prototypes for the ATLAS-Experiment at the High Luminosity Large Hadron Collider

Zusammenfassung

Mit dem Upgrade des Large Hadron Colliders (LHC) zum High Luminosity Large Hadron Collider Mitte der 2020er Jahre werden durch größere Datenmengen neue und präzisere Messungen des Standardmodells sowie weitere Suchen nach neuer Physik ermöglicht. Das LHC Upgrade wird das Potenzial aller Experimente am LHC erweitern. Um die erwünschte integrierte Luminosität aufzunehmen, muss der ATLAS-Detektor mit erhöhten Strahlenschäden, einem größeren Pile-Up und großen Datenmengen umgehen können. Infolgedessen wird der innere Spurdetektor vollständig durch den neuen *Inner Tracker* (ITk) ersetzt.

Teststrahlmessungen sind ein wichtiges Werkzeug, um neue Detektormodule zu entwickeln. Sie helfen neue Auslesechips, neue Sensoreigenschaften, sowie die Auswirkung von Strahlenschäden auf Auslesechip und Sensor zu verstehen. Sie sind nicht nur ein notwendiges Werkzeug in der Entwicklungsphase, sondern auch ein essentieller Teil der fortlaufenden Qualitätssicherung während der Modulproduktion.

Der Fokus dieser Arbeit liegt auf Verbesserungen von Teststrahlmessungen für Pixelmodule für das ITk Upgrade. Es werden Modifikationen der Datennahme mittels eines der Datenerfassungssysteme (USBpix) sowie eine neue Messmethode, die mittels dieses Systems zeitaufgelöste Messungen an Teststrahleinrichtungen ermöglicht, vorgestellt. Diese Messmethode wird zusammen mit Ergebnissen einer Beispielmessung diskutiert. Des Weiteren wurden etliche Verbesserungen und Modifikationen an der Rekonstruktionssoftware selbst vorgenommen. Der Funktionsumfang der Spurrekonstruktion mittels des General-Broken-Line Algorithmus wurde erweitert, welcher nun auch zusammen mit ATLAS Pixelmodulen funktioniert. Darüber hinaus wurden Testfälle implementiert, welche auch in Anbetracht zukünftigen Entwicklungen eine konsistente Rekonstruktion gewährleisten. Außerdem wurde ein Monte Carlo Framework als Validierungswerkzeug erweitert, um die Rekonstruktionsalgorithmen zu validieren.

Δεν ελπίζω τίποτα,
δε φοβούμαι τίποτα,
είμαι λεύτερος

I hope for nothing.
I fear nothing.
I am free.

Nikos Kazantzakis' epitaph

In memory of my father.

Contents

Contents

Introduction

1.1. Atomism

The modern word **atom** derives from old Greek *atomos* (ἄτομος), i.e. *uncuttable* or *indivisible*. It was coined by the philosophers Leucippus (Λεύκιππος) and Democritus (Δημόκριτος) who are closely linked to the natural philosophy of atomism, which arose in the 5th century before the Common. The main assumption of atomism is that any domain can be broken down into smaller composition pieces, which at some point reach the level of fundamental, non-further dividable pieces, the elementary constituents of that domain. This idea not only pervaded ancient Greek philosophy but can also be found in Indian antiquity.

While these ideas were not driven by the modern scientific method, they reveal humankind's desire to understand the most basic and most fundamental building blocks. Thus, it is not surprising that when the *building blocks* of chemistry were discovered and believed to be the fundamental constituents, they were named atoms. Yet, scientific curiosity challenged this idea and revealed that they are indeed not the most basic blocks. The name, however, was coined and remains, reminding us not only of our desire to understand the fundamental building blocks, but also how science is a trail of try-and-fail and the need to challenge persisting ideas.

The story continues similarly. The substructure of the atom, composed of a nucleus and an electron shell, could be subdivided even further. Nucleons are the building blocks of the nucleus, and even they exhibit a substructure. History well motivates our urge to probe the structure of our universe at smaller and smaller scales. It has taught us that we need to keep searching, exploring and challenging the known, not only to soothe our curiosity but to progress our knowledge of how the universe works on its most basic level.

1.2. The Evolution of Particle Physics

Given the historical evolution, pinpointing the birth of particle physics to a specific event seems ungrateful and difficult, if not impossible. However, the discovery of the electron by J. J. Thomson and his colleagues in 1897 [1] is indisputably an important milestone in particle physics. As it still is one of the elementary particles we know today, this event will serve as the starting point to give a brief history of its evolution.

J. J. Thomson investigated cathode rays within a magnetic field. Technical advancements enabled him to reduce the remaining air pressure within his experimental set-up. Therefore, he could prove that these rays were deflected by a magnetic field. Not only could he estimate the charge to mass ratio of the particles involved, showing that they were much lighter than hydrogen ions, but he also discovered that they behaved the same, regardless of what electrode material he used. This was in contrast to observations made with anode rays and led him to the conclusion that the observed rays were composed of a universal constituent, and indeed as we know today, this was the first discovery of an elementary particle. Thomson's research also resulted in the plum pudding model of the atom. For "[...] his theoretical and experimental investigations on the conduction of electricity by gases"[1] he was awarded the Nobel prize in 1906.

The observations of random motions of small particles suspended in a fluid by R. Brown in 1827 (nowadays known as Brownian motion) were explained by A. Einstein theoretically in 1905 [2]. This manifested the idea of small constituents of matter. Together with the experimental work of J. Perrin on Brownian motion, and the plum pudding model proposed by J. J. Thomson, this was the beginning of modern atomic theory.

It did not take long until the plum pudding model was challenged. The scattering experiments carried out by E. Rutherford together with H. Geiger and E. Marsden from 1909 onwards [3] lead to the development of the Rutherford model. Since then, scattering experiments have become a vital tool to probe atomic and subatomic structures. In that sense, the Rutherford experiment is the precursor of many modern particle collider experiments. However, Rutherford only investigated elastic scattering. In later experiments, mostly inelastic scattering processes were of interest.

It was in the mid 20's, that modern quantum mechanics was formulated. The foundation was laid by M. Planck in 1900 when he introduced his infamous *Hilfsgröße*, today known as Planck's constant h, which he used to quantise energy to derive the spectral emission of a black body at a given temperature, Planck's law. Einstein used the concept of energy quanta to explain the photoelectric effect in 1905. While its name was coined later, this was in some way the birth of the photon.

Quantum mechanics revolutionised physics, and it would go beyond the scope to give a detailed history. To summarise, in 1927 the understanding had reached the stage we would consider to be modern quantum mechanics. Heisenberg's uncertainty principle was published, the Schrödinger equation established and key concepts like spin and quantisation were widely accepted. This also led to a rapid understanding of the atomic structure. The Bohr model which had introduced the concept of quantised electron

---

[1]Rationale according to the official website of the Nobel prize committee.

energy levels to the atomic model was superseded by the concepts of atomic orbitals which are based on solutions to the Schrödinger equation for the electrons located around the nucleus.

The year 1932 came to be an *annus mirabilis* in particle physics, with three notable discoveries and inventions: the Cockcroft–Walton generator which enabled novel acceleration of charged particles, the discovery of the neutron, and that of the positron. The neutron discovery by J. Chadwick was driven by experiments carried out with α-rays [4,5]. Unknown electrically neutral radiation was observed, when certain, light materials were irradiated by α-sources. This new radiation produced protons if absorbed by paraffin wax. Kinematic studies led Chadwick to the conclusion that this unknown radiation had to have a very similar mass to the proton, but carried no electric charge. The neutron was discovered.

A couple of years earlier, in 1928, P. Dirac formulated a relativistic wave equation, embracing both quantum mechanics as well as special relativity [6]. The Dirac-Equation describes massive spin-½ particles and has the particularity of *negative energy* solutions, which describe the anti-particle states. In 1932, C. D. Anderson discovered the positron, the anti-particle of the electron [7,8].

Overall, it was in the 30's where several interactions, which are today well described in the context of modern field theories, were initially introduced. E. Fermi described β-radiation via Fermi's interaction theory, an effective theory of the weak interaction [9,10]. H. Yukawa built upon that idea and introduced the exchange of a heavy particle (today we know it is a meson, more specifically a pion) to mediate the binding force within a nucleus [11]. The discovery of the muon in 1936 by C. D. Anderson and S. Neddermeyer [12] was initially thought to be the one of the Yukawa particle, but it turned out to be *merely a heavier* electron. Thus, the physics-wise wrong term *mu meson* for the muon was quickly discarded. Being the odd one out, Nobel laureate I. I. Rabi famously commented on it with *"Who ordered that?"*.

The Yukawa particle, the pion, was finally discovered in 1947 in Bristol when C. Powell, C. Lattes, and G. Occhialini studied cosmic rays via photographic emulsions [13]. The pion, being the lightest hadron, was the start of a series of discoveries of various hadronic states. Developments in accelerator physics led to numerous discoveries of new particles in the 50's and 60's.

It was also during that time that the theory of Quantum Electrodynamics (QED) was continuously developed. Major contributions were made by by S. Tomonaga [14–17], J. Schwinger [18, 19], R. Feynman [20–22] and F. Dyson [23]. A series of difficulties had to be overcome, divergences in the perturbative calculation lead to non-physical observables, and it was the concept of renormalisation which tamed QED. Today, it can be said that QED paved the road for all the subsequent field theories and in particular of the embraced methods.

In 1956, the Cowan-Reines neutrino experiment confirmed the existence of the neutrino, specifically the electron antineutrino from fission in a nuclear reactor [24]. Its existence was already predicted by W. Pauli in 1930 due to the observation of continuous energy spectra of the decay products in β-decays, which are inconsistent with the

observed two-body decay. In the same year as the neutrino discovery, i.e. 1956, C.-S. Wu observed parity violation (P-violation) in weak interactions [25]. In fact, later it was confirmed that the parity violation is maximal, a distinct feature of the weak force. At this time, while P-symmetry violation was established, charge conjugation parity (CP) symmetry was still believed to hold. In 1964, J. Cronin and V. Fitch observed (indirect) CP-violation in kaon systems [26].

These observations advanced the theoretical description of the weak interaction, namely quantum flavourdynamics (QFD). A significant milestone was the combination of QFD and QED, the electroweak (EW) unification by S. Glashow [27], S. Weinberg [28] and A. Salam [29] in the 60's. The electroweak theory exhibits a spontaneous symmetry breaking which results in observable, force mediating, particles, the gauge bosons. These are: a massless particle which behaves as predicted by QED, the photon, and three massive gauge bosons related to the weak interaction, the electrically neutral Z-boson and the two electrically charged W-bosons.

The theoretical description in the context of gauge theory requires local gauge invariance. One difficulty which had to be overcome is that the gauge bosons in the EW unification are massless. While, when formulating the theory, no experimental evidence for the massive gauge bosons existed, the short range of the weak force requires the gauge bosons to carry mass. A mass term in the Lagrange density would violate gauge invariance. To overcome this problem, R. Brout, F. Englert [30], and P. Higgs [31] introduced what is today known as the Higgs mechanism. Via EW symmetry breaking, the Higgs mechanism couples the gauge bosons to the Higgs field, giving mass to the weak gauge bosons. Another consequence is the existence of a scalar boson, the Higgs boson.

In the 60's not only our understanding of the electroweak sector drastically changed, but also the understanding of the hadronic sector improved. In 1961 M. Gell-Mann proposed the *Eightfold Way* [32, 33], a scheme to classify the zoo of hadronic states observed at that time. In 1964 the quarks were postulated by M. Gell-Mann [34] and G. Zweig [35, 36], which gave rise to the quark model. Concepts like the strong force and colour charge introduced in the early 70's ultimately led to the formulation of quantum chromodynamics (QCD). An important clue which paved the road for QCD was, for example, the observation of the Δ^{++} baryon, a hadronic state of three up-quarks with parallel spin. This, seemingly Pauli exclusion principle violating state, hinted that something was missing and motivated a new type of charge, the colour charge as the charge for the strong interaction, as well as their force mediating gauge bosons, the gluons.

Experimental data supporting the quark hypothesis were provided by deep inelastic scattering (DIS) experiments carried out at the Stanford Linear Accelerator Center (SLAC) in the late 60's [37,38]. DIS experiments are performed to probe the substructure of a hadron by inelastically scattering a highly energetic (e.g., an electron) elementary particle off a hadron. They revealed the substructure of the nucleus. Unsure of whether they were the quarks, the term *parton* was coined, which up to today remains as an umbrella term for any constituent of a hadron, i.e., quark, anti-quark and gluon.

The proposed quark model consisted of three differently flavoured quarks, which explained the plethora of observed particles at that time. While a fourth quark flavour was

suggested earlier [39], only the lack of so-called flavour changing neutral currents (FCNC) motivated the prediction of a further quark flavour experimentally. This happened in 1970, in the context of the GIM-mechanism, proposed by S. L. Glashow, J. Iliopoulos and L. Maiani [40], which required a fourth quark to suppress FCNCs in loop diagrams. Naturally, this fuelled the experimental search for it.

It was the 11th November 1974 when the discovery of the J/Ψ was announced, often referred to as the *November Revolution.* The fourth quark, the charm, was simultaneously discovered by the groups of B. Richter [41] at SLAC and S. Ting [42] at the Brookhaven National Laboratory. In consequence of this discovery, the quark model was widely accepted and its significance is emphasised by the Nobel prize being rewarded to Richter and Ting merely two years later in 1976.

But why stop here? It was before the discovery of the charm quark that M. Kobayashi and T. Maskawa proposed even more quarks [43], motivated by the experimental observation of CP-violation which could be explained with further quarks. It was in 1977 when L. Lederman observed a new quark, the bottom quark, at Fermilab [44].

Another crucial experimental observation in the hadronic sector took place in the late 70's. The TASSO experiment at the PETRA accelerator complex, located at the Deutsche Elektronen Synchrotron (DESY), observed three-jet events [45], a direct evidence for gluons.

In the 80's, the UA1 and UA2 experiments at the European Organization for Nuclear Research (CERN) discovered the weak gauge bosons. Earlier experimental observations in 1973, performed with the Gargamelle bubble chamber, observed scattering of neutrinos off electrons via the exchange of a neutral Z-boson, so-called neutral current interactions. The measurements of neutral and charged current interaction with neutrinos allowed a determination of the weak mixing angle, which in turn gave an estimate of the weak gauge bosons masses. These motivated the direct search for those particles and ultimately the W- and Z-boson discoveries were achieved in 1983. First, the slightly lighter W-boson was observed by both, the UA1 and UA2 collaboration [46, 47]. Only a couple of months later, they announced the observation of the Z-boson [48, 49].

The UA2 is not only notable for these discoveries, but also for being the first experiment to use novel silicon detectors. During a detector upgrade, they introduced silicon-pad detectors, the first use of silicon as a tracker in a high energy physics experiment.

While this summary is far from complete, two further discoveries must not be left out. The last quark still missing in the picture, the top, was discovered in 1995 at the Tevatron by the DØ and CDF collaboration [50, 51]. The last significant discovery, as of today, was the observation of the Higgs boson in 2012 by the ATLAS [52] and CMS [53] experiments at the Large Hadron Collider at CERN.

The Standard Model

2.1. Overview and Tests

The Standard Model (SM) provides a theoretical description of three fundamental forces in the Universe: the weak force, the electromagnetic force, and the strong force. The fundamental particles within the SM are the quarks and leptons in the fermionic (half-integer spin) sector and the force mediating gauge bosons in the bosonic (integer spin) sector.

The leptons and quarks are grouped into three generations. While there is no natural reasoning for three generations and there is no intrinsic need to have the same number of generations for the leptonic and quark sector, experimental results suggest that in both cases no fourth generation is missing. In the leptonic sector the measurement of the Z-decay width supports three light neutrino generations, and the unitarity conditions of the CKM-matrix, the matrix relating the weak eigenstates (flavour states) to the propagating mass states, agrees with three generations of quarks.

The constituents of the Standard Model together with their mass, electric charge, and colour charge are depicted in Figure 2.1[1]. The six quarks (the antiparticles are implicitly meant as well) are the only fermions that carry a colour charge and thus interact via the strong force, mediated by the gluons. As quarks also carry electric charge they also participate in the electromagnetic interaction, together with the charged leptons (the electron, muon and tau) and the charged W^+- and W^--bosons. This force is mediated by the photon. All fermions, as well as the electroweak gauge bosons, interact via the weak interaction.

The Higgs-boson couples to all particles which have mass in the Standard Model. This includes all the quarks, the charged leptons, the massive Z- and W-bosons as well as the Higgs-boson's self-coupling.

---

[1]In the Figure the neutrinos are shown having a non-zero mass, this is experimentally established, but strictly speaking not incorporated in the SM.
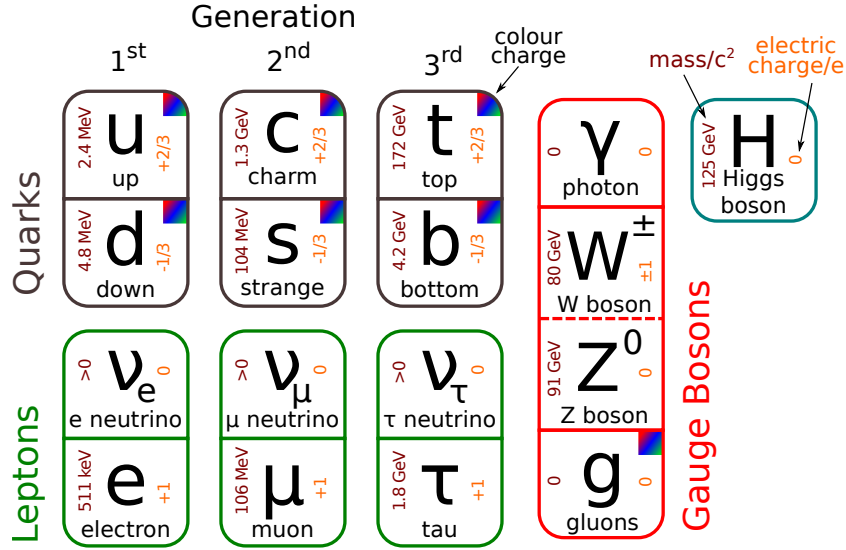
Figure 2.1.: An overview of the constituents of the Standard Model.

Measurement of these interactions has been and still is a very important tool to test the Standard Model. For example, the existence of triple-gauge-boson couplings (TGCs) is an essential feature of the gauge group structure, and hence of the SM.

For the WW cross-section to not violate unitarity, these couplings must exist and interfere with the t-channel $\bar{\nu}_e$ exchange diagram. Thus, the experimental investigation of this process at LEP was a critical test of the Standard Model. In particular, allowed TGCs in the electroweak sector are WWZ and WWγ, while ZZZ, ZZγ, Z$\gamma\gamma$ and $\gamma\gamma\gamma$ are not possible as neither the Z-boson nor the photon carries either electric charge or weak isospin. All of them are shown in Figure 2.2. Obviously, to probe the SM one has also to investigate the non-observation of forbidden processes and set exclusion limits on their production rates.

(a) Allowed WWZ/WWγ. (b) Forbidden ZZZ/ZZγ. (c) Also forbidden $Z\gamma\gamma/\gamma\gamma\gamma$.

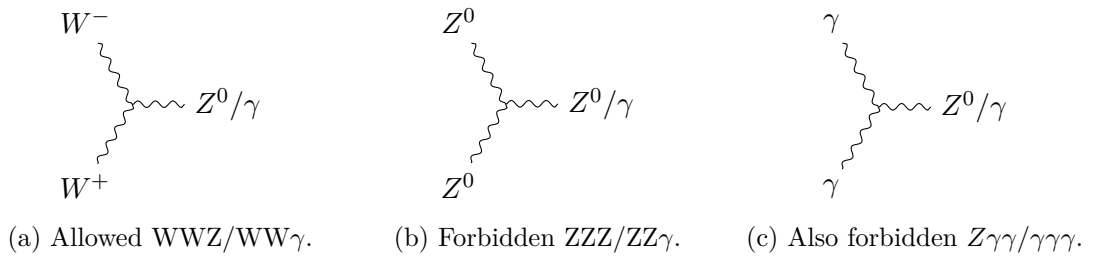Figure 2.2.: All possible TGC vertices of the EW gauge bosons' self coupling.

Other tests of the Standard Model are for example precision measurements of cross-sections. One of the significant achievements of the Standard Model is its predictive power. Cross-sections over several orders of magnitude have been predicted as well as measured. In Figure 2.3 a summary of the ATLAS experiment's measurements is given.
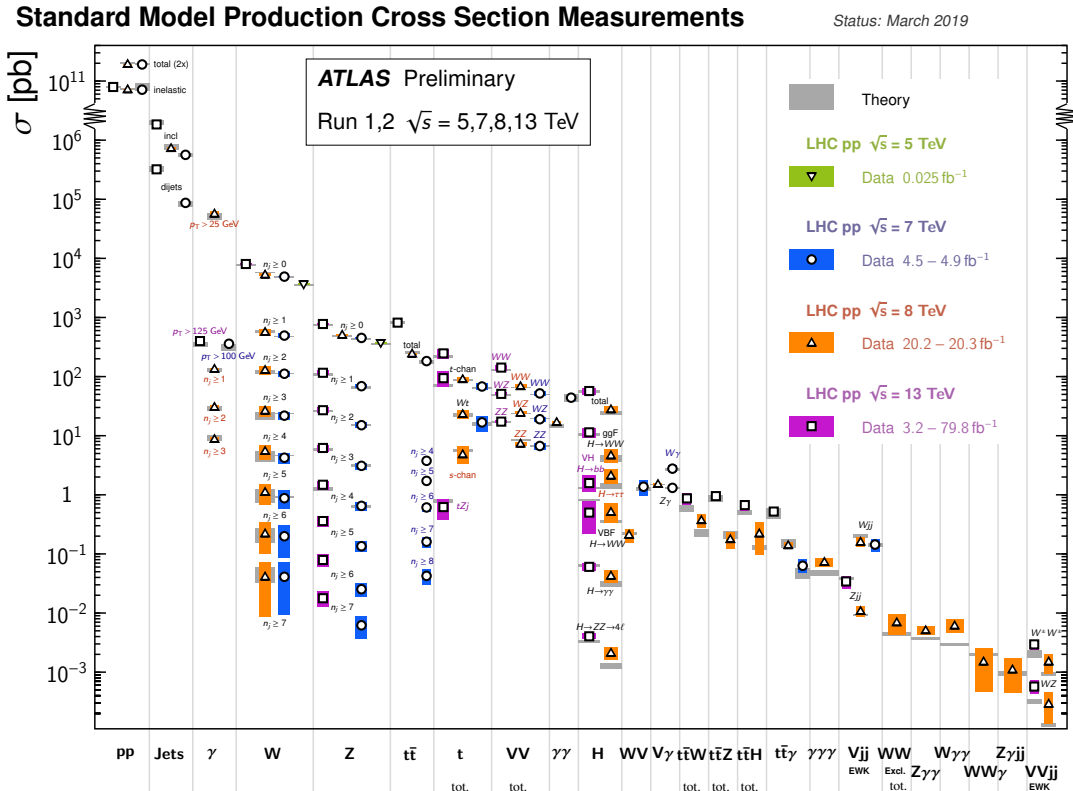
Figure 2.3.: Standard Model cross-section measurements by the ATLAS Collaboration. Published by the ATLAS Standard Model physics group [54].

They range from 10^{11} nb for the total proton-proton (pp) interaction cross-section[2] to 10^{-3} nb for the EW-only scattering process of two massive vector-bosons, i.e. Z- or W-boson, in association with two jets (VVjj)[3].

While the predictive power of the Standard Model is astonishing, we know that it cannot be the final theory. Hence, today's particle physicists need to put the SM through its paces. Even the smallest deviations from the predictions might hint at new physics. To succeed, excellent theoretical predictions, enough data to not be limited by statistics, novel and refined analysis techniques, and precise detector measurements are needed.

As this thesis focusses on the upgrade of the ATLAS pixel detector, the emphasis is put on two physics processes. Namely, the top-quark and Higgs-boson physics prospects, as these are the most prominent fields where so-called b-tagging is of importance. B-tagging heavily relies on the pixel detector's performance.

---

[2] nb: nanobarn, 1 barn corresponds to 10^{-24} cm^2 and is the unit to express cross-sections in HEP

[3] It is necessary to point out, that the observed signal significance for this process is above 3 σ but below 5 σ. Hence, in particle physics we say that there is *evidence* for this process.

2.1.1. The Top-Quark

The top-quark has a mass of $m_{\text{t}} \approx 173.3$ GeV and is thus the heaviest known quark. As it is heavier than the W-boson, it can decay weakly into a real W-boson and a bottom-quark which gives rise to its very short lifetime of $\mathcal{O}\left(10^{-25} \text{ s}\right)$. Its almost exclusive decay channel, the t→Wb decay, is shown in Figure 2.4. The very short lifetime of the top-quark results in its decay before any possibility of forming hadronic states, i.e. before hadronisation.
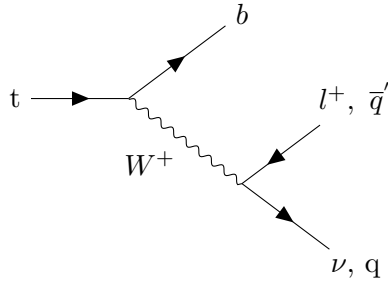

Figure 2.4.: Decay of a top quark into a W-boson and a b-quark.

This is one of the motivations for studying the top-quark, as it is a good approach to try and study something like a *bare* quark. A further reason for the pronounced research interest is its mass. The strength of the Yukawa-coupling to the Higgs-boson is directly proportional to the mass. Hence, many loop level diagrams that contain top-quarks are of importance when it comes to more precise measurements.

The decay into b-quarks has a very important consequence. As the b-quarks form hadrons containing b-quarks, b-hadrons, they can be used to identify events with probable t-quark decays. B-hadrons have a relatively long lifetime. This is because the CKM mixing angle highly suppresses the b-quark's decay into quarks of the second generation (or even first). Lifetimes in the order of $\mathcal{O}\left(\text{ps}\right)$ of the B-hadrons lead to a displacement of $\mathcal{O}\left(\text{mm}\right)$ from the primary vertex, for hadrons of at least a few GeV transverse momentum (p_T).

B-Tagging in ATLAS

The long lifetime of the b-hadrons is exploited in so-called lifetime-tagging approaches of b-tagging, which aim at identifying jets which contain a b-quark. The ATLAS Collaboration uses two different groups of algorithms [55]. One approach uses the impact parameter (IP), which takes the track from the decay and computes the closest point of approach to the primary vertex. The IP is also shown in Figure 2.5, where the decay of a long-lived particle in a secondary vertex is shown. The second lifetime-tagging approach fully reconstructs the secondary vertex.
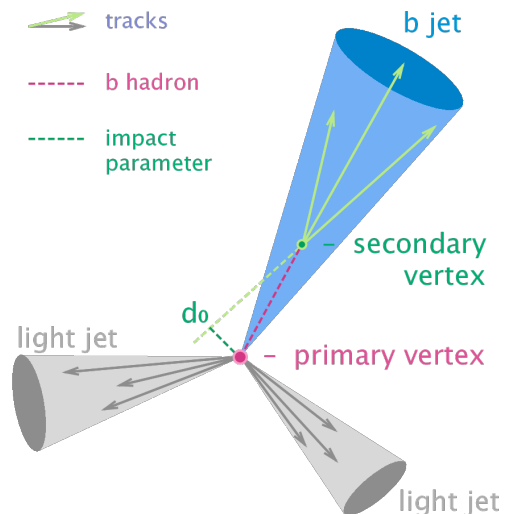

Figure 2.5.: Secondary vertex from a long-lived particle. Credit: Nazar Bartosik (CC BY 4.0).

For actual analyses, the ATLAS experiment uses these different algorithms and combines them via a multivariate technique[4], the output is then used to tag jets as jets with a b-quark.

In 2014, the tracking detector of the ATLAS experiment has been upgraded. In particular, an additional layer of pixel detector has been installed, the Insertable B-Layer (IBL). This results in an additional measurement point and improves the tracking resolution, the increase is shown in Figure 2.6.
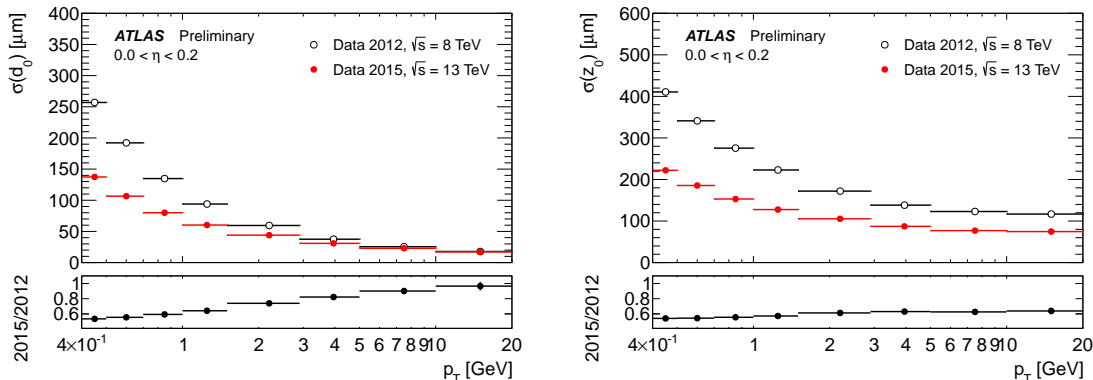
(a) Transverse impact parameter resolution.   (b) Longitudinal impact parameter resolution.

Figure 2.6.: Comparison of the tracking resolution prior to the upgrade (Data 2012) and after (Data 2015). Shown is the transverse (d_0) and longitudinal (z_0) impact parameter resolution σ, binned in transverse momentum p_T.

To quantify the b-tagging performance, one needs to investigate a two-parameter space. One is the fraction of identified b-jets, which is the b-tagging efficiency ε_b. The other relevant parameter is the non-b-jet rejection rate[5] which quantifies how many non-b-jets are tagged as b-jets. A high-purity sample, i.e. a high rejection rate, is desired.

In Figure 2.7 the algorithm's performance is profiled. In Figure 2.7a the overall increase in light-jet rejection is shown. The improved tracking resolution at low p_T provided by the IBL has a direct impact on the tagger's performance; this can be seen when investigating the light-jet rejection at a fixed working point, versus jet p_T (Fig. 2.7b). At low p_T values the rejection rate increases by about a factor of 4. The overall improvements are not only due to better tracking, but also because of algorithmic improvements. However, the enhanced tracking performance has a significant contribution [56].

2.1.2. The Higgs-Boson

Another very important field where b-tagging is of importance is the physics related to the Higgs-boson. Not only because the ttH final state is very interesting, as it allows to probe the top-quark's Yukawa-coupling directly, but also because the predominant decay

---

[4]In Run-1 this was a neural network (MV1), for Run-2 a boosted decision tree is used (MV2).

[5]Typically this is split up into c-jet and light-jet rejection.
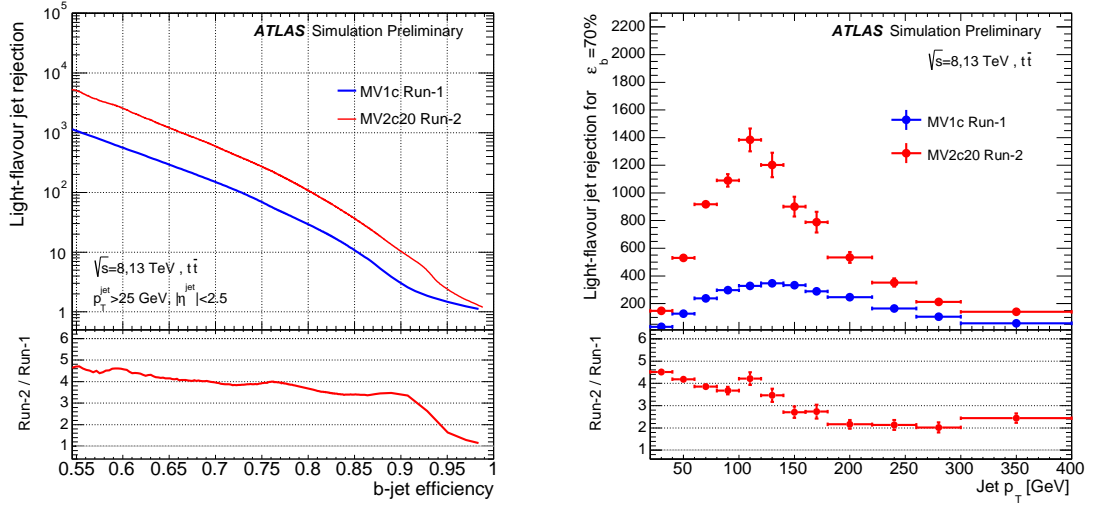
(a) Light-jet rejection versus b-tagging effi-
    ciency ε_{b}.

(b) Light-jet rejection for a fixed $\varepsilon_{\mathrm{b}} = 70\%$
    versus jet p_T.

Figure 2.7.: Predicted performance of the b-tagging algorithm prior to the upgrade
(MV1c Run-1) and after the upgrade (MV2c20 Run-2) in comparison [56].

mode of the Higgs-boson is into pairs of b-quarks. Measurements, in particular precision
measurements, of the newly discovered Higgs-boson, are a possible path towards new
physics.

An exemplary analysis which heavily relies on b-tagging is given in [57] where the ttH
process, with a subsequent H→bb decay, is investigated, i.e. an analysis where the final
state has four b-jets.

The branching ratios for the Higgs-boson are given in Figure 2.8. It seems somewhat
counter-intuitive that the discovery of it was done in the H→$\gamma\gamma$, H→ZZ and H→WW
channels, which are by one or two orders of magnitude more unlikely than the decay
H→bb. However, the clear signature of these processes compared to two b-jets, made
these the discovery channels, despite merely a small fraction of the events.

Today, the ATLAS experiment has observed a variety of Higgs-boson decays and has
set limits on others. A summary of the measured and expected signal strength is given
in Figure 2.9. The Zγ and $\mu\mu$ channels have a very low branching ratio, as indicated in
Figure 2.8. However, a precise measurement of the $\mu\mu$ channel would directly probe the
coupling to the second generation of fermions (leptons).

This is one of the many measurements which motivate the upcoming accelerator and
detector upgrades, as such measurements will be possible with a signal significance of
$7\,\sigma$, compared to $2.3\,\sigma$ without the upgrade, as discussed in [58].

Another exciting measurement in the future is the Higgs-boson pair production. The
two leading order diagrams contributing are given in Figure 2.10, for a collider where
gluon fusion is the dominant production process. There is one diagram where the Higgs-
boson's trilinear self-coupling contributes (Figure 2.10a) whereas in the second it does
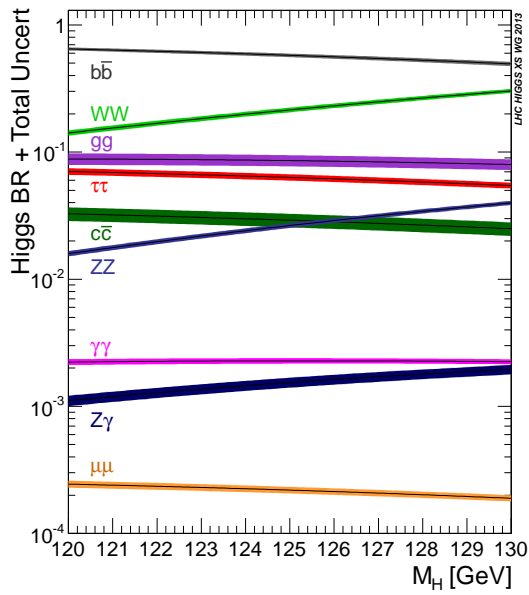
Figure 2.8.: The branching ratios of the Higgs-boson's decay for various Higgs-boson masses.
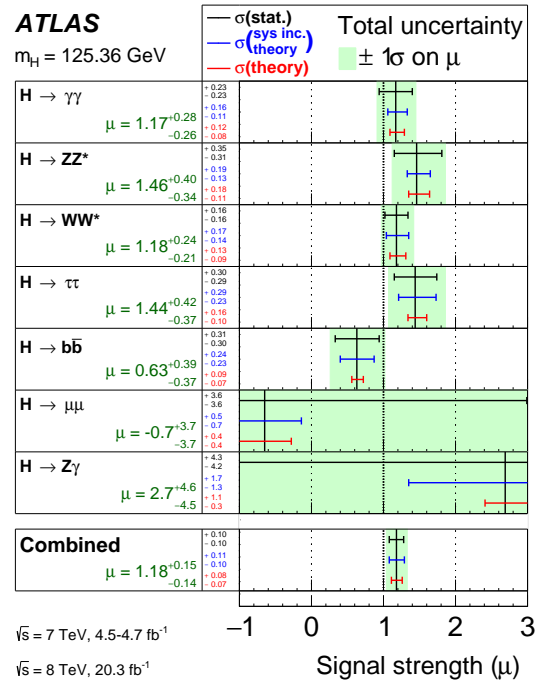
Figure 2.9.: Measurements of the Higgs-boson decays with $\sqrt{s} = (7\ \&\ 8)$ TeV data.

not (Figure 2.10b). These diagrams interfere destructively. Hence, if no trilinear self-coupling is realised in nature, only the right diagram contributes. This would double the pair-production cross-section [59]. The sensitivity in two final channels has been studied, H($\rightarrow$bb)H($\rightarrow\gamma\gamma$) and H($\rightarrow$bb)H($\rightarrow\tau\tau$) in [60] and [61], with an estimated signal significance of $1.3\,\sigma$ and $0.6\,\sigma$, respectively. While this is far from observing Higgs-boson pair production, a combination of various channels and experiments might provide evidence for it.

(a) Higgs-boson pair production involving a trilinear self-coupling.

(b) Second diagram contributing to the Higgs-boson pair production.

Figure 2.10.: The two diagrams which contribute to Higgs-boson pair production at the LHC.

This overview of the physics prospect is far from complete and based on a personal, biased selection. Regardless of that, it motivat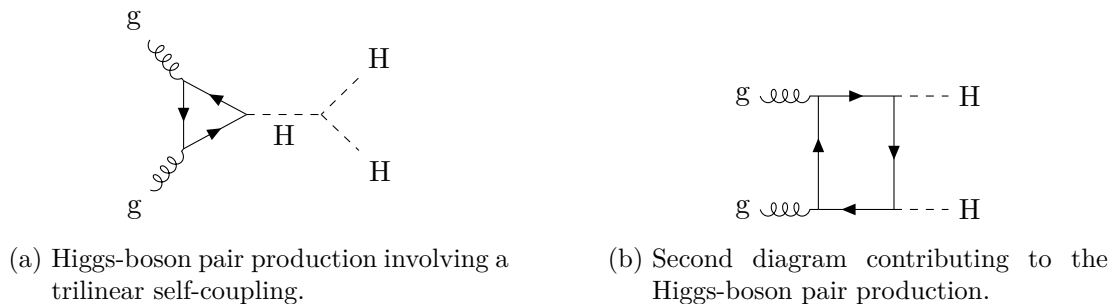es the upcoming upgrade of the accelerator and detectors. Aside from the discussed measurements directly related to the Standard Model, there is the need to further probe Physics Beyond the Standard Model, or short BSM physics. There are several experimental observations and open questions in the Standard Model which have to be addressed.

2.2. Limitations of the Standard Model

Despite the success of the Standard Model, there exist several reasons why it cannot be the final theory. Also, gravity as one of the fundamental forces is not incorporated in the SM as a quantum field theory.

2.2.1. The Hierarchy Problem

The so-called hierarchy problem is not a limitation of the SM, but rather a property of it, which many physicists consider *unnatural*. It can be related to the Higgs-boson's mass, which has been measured to $m_\mathrm{H} \approx 125$ GeV. Quantum loop corrections modify the observed mass of the scalar Higgs-boson (cf. Figure 2.11). Any massive particle (and not only the fermions as indicated by their propagator in the Figure) which couples to the Higgs-boson contribute. One would naively expect the corrections to be in the order of the scale up to which the theory is valid, i.e. the Planck scale. However, the observed mass is in the order of 10^2 GeV, while the Planck mass it at 10^{19} GeV.
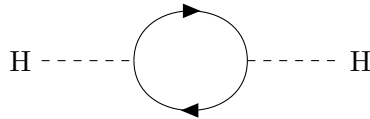
Figure 2.11.: Corrections to the Higgs-bosons mass via quantum-loop corrections.

To bring these corrections back to the Higgs-boson's mass, a fine-tuning is needed, which is considered *unnatural*. One of the most famous family of extensions to the Standard Models are the so-called super-symmetric extensions. They introduce super-partners to all the particles, which would lead to a natural cancellation of these corrective terms. Unfortunately, no evidence for them has been observed so far.

2.2.2. Dark Matter

In 1933, swiss astronomer F. Zwicky studied the Coma Cluster. Observing the motion of galaxies at the edge of the cluster and estimating the cluster's mass from visible radiation, he concluded that there must be *dunkle Materie*, dark matter, in the cluster [62, 63]. Later, many other studies came to the same conclusion and nowadays the astronomical observations pointing to the existence of dark matter are overwhelming.

Up to today, there is no viable candidate that could explain dark matter. Particle physics, describing the fundamental building blocks of our surrounding as well as exotic matter, should provide a solution to this problem. And indeed, some stable, not yet observed particle could be responsible for dark matter. Hence, one of the more exotic fields of searches with the ATLAS detector is the one for dark matter particles. As the name suggests, dark matter candidates will not interact with the detector. Thus, searches for it typically look for missing transverse energy in the detector or look for decay products of short-lived partners of the stable dark-matter particles.

2.2.3. Neutrino Oscillation and Neutrino Mass

The Homestake experiment was an experiment which measured the neutrino flux of neutrinos originating from nuclear fusion in the sun [64]. It established that the measured flux was by a factor of about three lower than expected based on the current model of the sun. This observation was later confirmed by other experiments and came to be known as the *solar neutrino problem.*

It was the Super-Kamiokande experiment which found evidence for so-called neutrino oscillation more than two decades later [65]. Neutrino oscillation happens when the neutrino mass is not zero and their flavour eigenstates are not their mass eigenstates. The Pontecorvo-Maki-Nakagawa-Sakata matrix (PMNS matrix) is the analogue to the CKM matrix in the quark sector, relating these states to each other. Today, the phenomenon of neutrino oscillation is well established. The absolute masses of the neutrinos however remain unknown, with upper limits set in the range of 120 meV for the sum of the three generations [66].

Proposed solutions to incorporating neutrino mass into the SM exist, however the nature of the neutrino is still to be discovered, and depending on the outcome different mass mechanism can be incorporated. To determine the properties of the neutrinos, other experiments try to investigate neutrinoless double β-decay, a lepton number violating process which would show that the neutrino is its own anti-particle, i.e. a Majorana particle.

2.2.4. CP-Violation and Baryogenesis

The PMNS matrix and CKM mixing are two sources of CP-violation in the Standard Model. No CP-violation has been observed in the strong interaction. This is somewhat puzzling, as there is no natural reason why this should be the case. This is known as the strong CP-problem - *Why does nature conserve CP-symmetry in the strong sector?*

Somewhat related is the problem of matter-antimatter symmetry, as CP-violation could explain why there is matter surrounding us. In the phase of baryogenesis, the baryonic asymmetry between matter and antimatter was produced. However, the experimentally observed amount of CP-violation is not enough to explain this asymmetry and the reason for it remains unknown.

Measurements at the LHC by the experimental collaborations can provide novel observations which could push our understanding of the fundamental forces and particles

they act upon. While theoretically, many extensions and modifications can be proposed, they ultimately need to be fuelled by experimental observations, hopefully made by the ATLAS Collaboration and other experiments at the LHC.

The Large Hadron Collider and ATLAS

To carry out future precision measurements and searches for new physics beyond the Standard Model, the Large Hadron Collider (LHC), as well as the experimental detectors, will be upgraded. A short introduction to the current set-up and performance is given, followed by a concise summary of the upcoming upgrade plans.

3.1. CERN's Accelerator Complex and the LHC

The Large Hadron Collider is a proton-proton and heavy ion collider with a designed centre of mass energy of $\sqrt{s} = 14$ TeV. It is located near the city of Geneva, Switzerland at the CERN research site.

The LHC is located in the tunnels of the former Large-Electron-Positron collider and provides particle collisions to four large experiments, namely ATLAS, CMS, ALICE and LHCb. The LHC uses radio-frequency cavities to accelerate particles, resulting in a bunched structure of the beam. Superconducting dipole magnets, cooled down to 1.9 K, generate a magnetic field of 8.33 T which is used to bend the beam onto a circular track of approximately 27 km circumference. A single bending dipole consists of two apertures
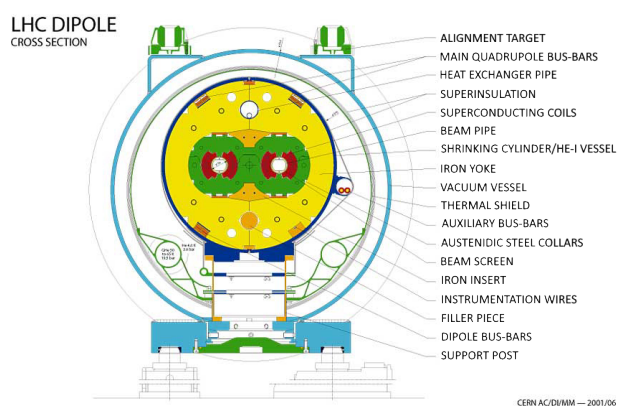

Figure 3.1.: Cross-section of an LHC dipole used to bend the charged beam. Copyright and image credit: CERN.

17

with flipped field direction, enabling both beams, the clockwise and counter-clockwise travelling one, to use the same magnets. A cross-section of an LHC dipole is shown in Figure 3.1, where the two apertures with the beam pipes are shown. Further magnets are used to focus or steer the beam.
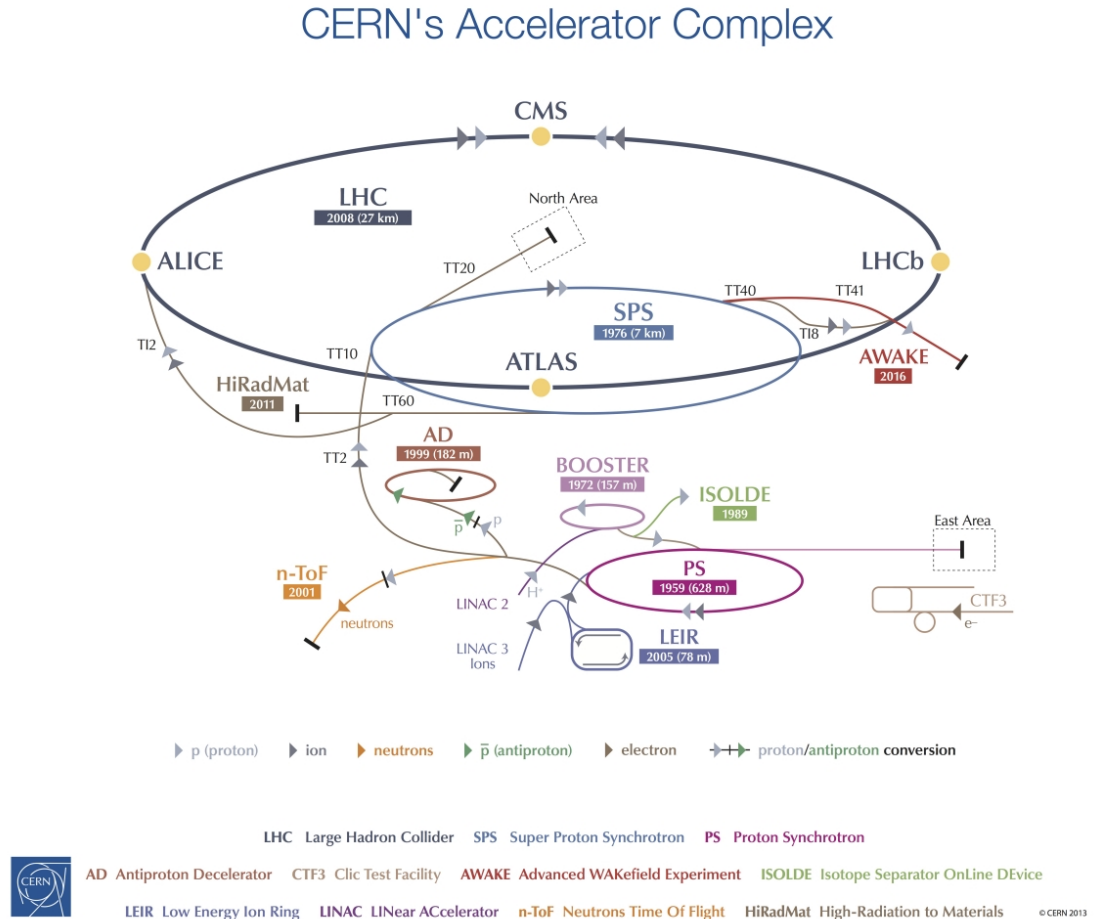
Figure 3.2.: An overview of CERN's current accelerator complex. The pre-accelerator chain leading up to the LHC as well as the experimental interaction points are shown. Copyright and image credit: CERN.

Before particle bunches are accelerated in the LHC, they have to pass through the entire pre-accelerator chain. The accelerator chain, as well as some further accelerators at the CERN research site, are schematically shown in Figure 3.2. For proton beams, this starts at the LINAC 2 accelerator, which accelerates protons to 50 MeV. The Proton Synchrotron Booster (PSB, or BOOSTER) is the next accelerator in the chain. It accelerates the protons to 1.4 GeV which are then injected into the Proton Synchrotron (PS). The Super Proton Synchrotron (SPS) is the final pre-accelerator. It is the same

accelerator which was operated as the Sp$\overline{\text{p}}$S and discovered the electroweak gauge bosons. It accelerates the protons up to 450 GeV which are then injected into one of the LHC's rings.

3.1.1. Luminosity

The interaction point of experimental particle physicists and accelerator physicists is the luminosity. The instantaneous luminosity $\mathcal{L}_{\text{inst.}}$ relates the cross-section of a process σ to the event rate dN/dt via:

$$\frac{dN}{dt} = \mathcal{L}_{\text{inst.}} \times \sigma$$

For experimental particle physicists, the integrated luminosity $\mathcal{L}_{\text{int.}} = \int \mathcal{L}_{\text{inst.}} dt$ gives the amount of observed events for a given cross-section:

$$N = \mathcal{L}_{\text{int.}} \times \sigma$$

One can relate the instantaneous luminosity to accelerator parameters:

$$\mathcal{L}_{\text{inst.}} = \frac{n_{\text{b}} N^2 f_{\text{rev}}}{4\pi \beta^* \varepsilon_{\text{n}}} R$$

Where N is the number of protons per bunch, n_{b} the number of bunches in the machine, f_{rev} the revolution frequency, β^* the beam beta function, ε_{n} the transverse normalised emittance and $R = 1/\sqrt{1 + \frac{\theta_{\text{c}}\sigma_{\text{z}}}{\sigma^*}}$ a geometrical reduction factor, where the beam crossing angle θ_{c}, RMS bunch length σ_{z}, and transverse RMS beam size σ^*, contribute[1].

As the revolution frequency is fixed and the number of bunches limited, only the bunch intensity (given by the number of protons in each bunch), the beam beta function and transverse normalised emittance, as well as the geometrical reduction factor, can be optimised to increase the luminosity. The goal of the *High Luminosity-LHC* upgrade project (HL-LHC) [67] is an increase of the luminosity. It aims at providing at least 250 fb^{-1} of integrated luminosity to the ATLAS and CMS experiments each year.

One problem in the LHC accelerator is electrons produced by the synchrotron radiation inside the beam pipe. The proton bunches accelerate these photoelectrons towards the beam, which then impact the other side of the beam pipe and release further secondary electrons. This effect is known as the build-up of electron-clouds (or e-clouds) and has a negative impact on the beam stability and quality. Initially, the fall-back option was a 50 ns bunch spacing[2] which mitigates this. However, if less but further apart spaced bunches are used, the average interactions per bunch crossing must be larger to provide the same instantaneous luminosity. This quantity is known as pile-up. Higher pile-up poses a problem to the experiments. Consequently, the 50 ns bunch spacing scheme was

---

[1]This assumes a Gaussian beam profile and accounts for the beam crossing angle.

[2]When putting the LHC into operation, the 50 ns bunch spacing was intensively used to learn the operational aspects of the LHC.

	LHC (design)	HL-LHC (baseline)	HL-LHC (8b+4e)
Beam collision energy [TeV]	7	7	7
$N \left[10^{11}\right]$	1.15	2.2	2.3
n_{b}	2808	2748	1968
Beam current [A]	0.58	1.09	0.82
β^* at collision [m]	0.55	0.2	0.2
ε_{n} [um]	3.75	2.50	2.2
R without crab cavities	0.836	0.369	0.369
R with crab cavities	n.a.	0.715	0.715
Peak luminosity $\left[10^{34} \text{ cm}^{-2}\text{s}^{-1}\right]$	1.18	12.6	11.6
Levelled luminosity $\left[10^{34} \text{ cm}^{-2}\text{s}^{-1}\right]$	n.a.	5.32	5.03
Pile-up $\langle\mu\rangle$ (levelled)	27	140	140

Table 3.1.: Design goals for the HL-LHC, comparing the original LHC design to the HL-LHC baseline as well as the 8b+4e filling scheme.

discarded. It has been replaced by a fall-back filling scheme where eight filled bunches are followed by four empty ones (8b+4e). This also mitigates the effects of e-clouds.

An overview of the design parameters to achieve the luminosity increase with the HL-LHC [67] is given in Table 3.1, where the original LHC design[3] is compared to the HL-LHC.

3.1.2. Accelerator Upgrade

For the accelerator upgrade the number of protons per bunch is roughly doubled, resulting in a beam current of over 1 A. At the same time, the transverse normalised emittance is reduced. A reduction of β^* is possible with stronger and larger aperture quadrupole magnets used in the focussing of the beam at the interaction points. This, however, goes hand-in-hand with an increased crossing angle to prevent more than one head-on collision in the interaction region, reducing the geometrical reduction factor, R, as the crossing angle, θ_{c}, increases from 280 µrad to 510 µrad. The usage of crab cavities allows mitigating this effect by turning the bunches in a way so that they geometrically overlap again.

A beam current of 1 A exceeds the *ultimate beam* conditions the LHC systems were originally designed for, which is $I_{\mathrm{beam}} = 0.86$ A, by about 30%. This is a challenge not only to systems maintaining a stable beam (RF-systems, and similar) but also a possible risk as it puts more stress onto the cryogenic systems and leaves less room for errors.

A beam with high brightness is needed. The beam brightness is proportional to the ratio of beam intensity to its transverse emittance, i.e. an intense beam with a small transverse emittance will have a high brightness. As for any Hamiltonian system, Liouville's theorem states that the phase space will be conserved if only conservative

---

[3]Some of these values have been outperformed, shown here are the ones the LHC was designed to meet.

forces act upon the system. As a result, it is necessary to generate a beam with a high brightness right at the beginning of the accelerator chain. The LHC Injectors Upgrade Project (LIU) aims at this [68, 69], with the new LINAC4 probably being the most significant change. The LINAC4 is intended to replace the LINAC2 and provide a beam with a doubled brightness.
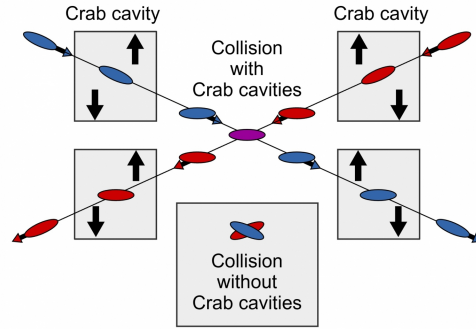

Figure 3.3.: Crab cavities. Copyright and image credit: CERN.

The reduction of β^* can be achieved with improved beam optics, squeezing the bunches even closer together at the interaction points. The use of so-called crab cavities mitigates the decrease of the geometric reduction factor. Their working principle is shown in Figure 3.3. Crab cavities are superconducting RF-cavities which rotate the bunches in a specific manner to maximise their geometrical overlap: A first cavity will *kick* the bunch prior to the collision, a second cavity will undo this rotation after the collision. This way the geometrical reduction factor, R, which deteriorated due to the increased crossing angle down to about 0.37, is restored to about 0.72.
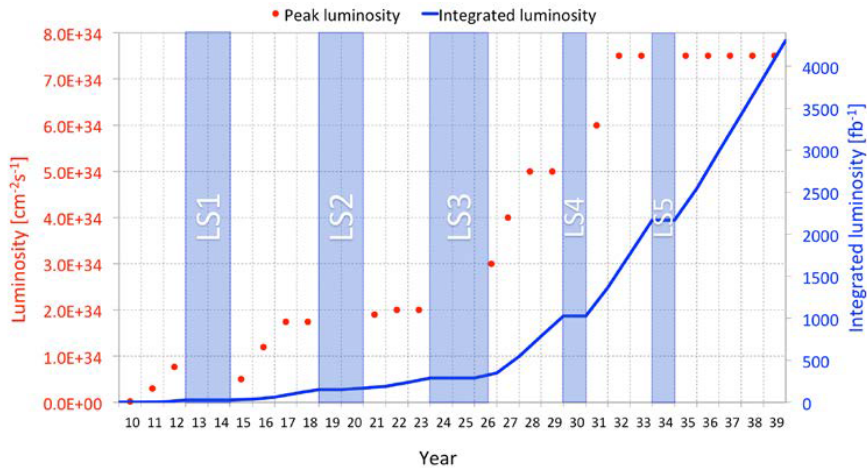

Figure 3.4.: Time-line for the present and future LHC, assuming ultimate HL-LHC conditions from 2010 to 2039.

In Figure 3.4, a proposed timeline for the LHC is given. It is the most optimistic scenario where the design parameters are outperformed, and a total integrated luminosity of 4000 fb^{-1} is collected by 2040. The past, as well as future, long shutdowns (LS) are indicated. LS3 is the shutdown where the detector and machine upgrades are carried out. A more conservative timeline which aims at 3000 fb^{-1} also exists and can be reviewed in [67].

3.2. The ATLAS Experiment

The ATLAS experiment is operating one of the two general purpose detectors installed at the LHC. The detector features an onion-layered layout which is standard to all modern HEP general-purpose detectors.

The innermost part aims at the spatial detection of charged particles, i.e. all charged particles which are created in the interaction or are a subsequent decay product, but also conversion electron-positron pairs from converted photons. As discussed in Chapter 2, b-tagging is an essential technique for many existing and upcoming measurements. As we have also seen, the tracking detector and its performance play a vital role in the b-tagging algorithm. It further provides important measurements of charged-particles momenta.

The next layer is the electromagnetic calorimeter, followed by the hadronic calorimeter. The purpose of the calorimeters is to absorb the produced particles and measure the total deposited energy. In the process, the particles interact with an absorber material and start to *shower*, i.e. produce secondary particles which undergo the same process until the entire energy is released. This energy is measured by an active material, usually a scintillator.

The outermost parts of the detector are the muon systems. Muons interact only very weakly, thus are not stopped by the calorimeter. They exit the detector, but their trajectory is measured in the outermost layers. Magnetic fields will bend them, allowing to derive their momentum from their trajectory. A picture of the ATLAS detector is given in Figure 3.5, where also the large dimensions of the detector are indicated. The ATLAS detector is the largest HEP detector ever built as of today.

3.2.1. The Inner Detector

To describe acceptance regions, it is important to introduce the coordinate system used in the ATLAS experiment. It has its origin at the nominal interaction point in the beam pipe. The x-axis points towards the centre the LHC ring, the y-axis upwards and the z-axis along the beam axis. The azimuthal angle ϕ is measured around the beam axis and the polar angle θ is the angle from the beam axis. The pseudorapidity is defined as $\eta = -\ln \tan{(\theta/2)}$.

The Inner Detector (ID) is the tracking detector at the core of the ATLAS detector, located right around the interaction point. In Figure 3.6 a quarter section of this part of
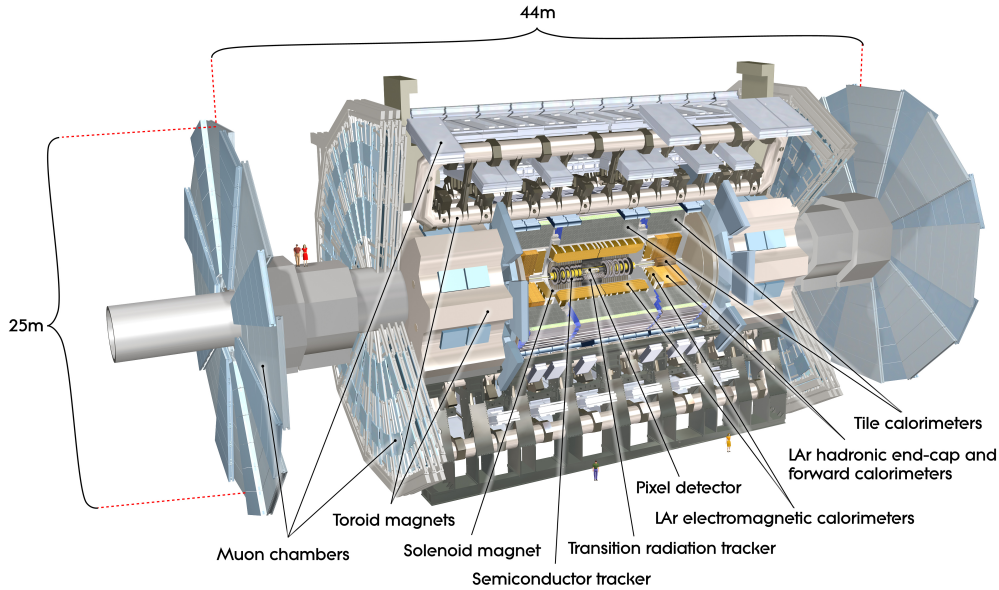
Figure 3.5.: Cut-away view of the ATLAS detector. ATLAS Experiment © 2018 CERN.

the detector is shown. The entire section is immersed in an axial magnetic field of 2 T, created by a solenoid right outside of the ID aligned with the beam axis.

The different parts of the detector are the insertable B-layer (IBL) (not shown in Fig. 3.6), the pixel detector (Pixel), the silicon microstrip layers (SCT) and the transition radiation tracker (TRT).

The Pixel and SCT provide coverage up to $|\eta| = 2.5$, whereas the TRT only provides coverage up to $|\eta| = 2$. The nominal p_T threshold for several of the algorithms designed for pattern recognition is at 0.5 GeV, however, as was also shown in the tracking performance plots in the b-tagging section in Chapter 2, some algorithms and measurements are feasible down to 0.1 GeV. With the TRT, electron identification is possible in a p_T range from 0.5 GeV to 150 GeV. The amount of emitted transition radiation is proportional to the γ-factor, which is why mostly electrons pass a certain high threshold in the TRT and are in this way identified as such. Other particles TRT signal only passes a low threshold.

The Pixel Detector

The Pixel detector consists of three barrel layers, located around the beam pipe. They are located at a radial distance of 50.5 mm, 88.5 mm and 122.5 mm away from the nominal interaction point (IP). Furthermore, three disk layers on each side provide additional angular coverage, starting at about $|\eta| \approx 2$ (cf. the zoom view in Fig. 3.6).

The barrel layers and disks together with their mechanical support structure are shown in Figure 3.7.

Figure 3.6.: Plan view of a quarter section of the ATLAS ID.
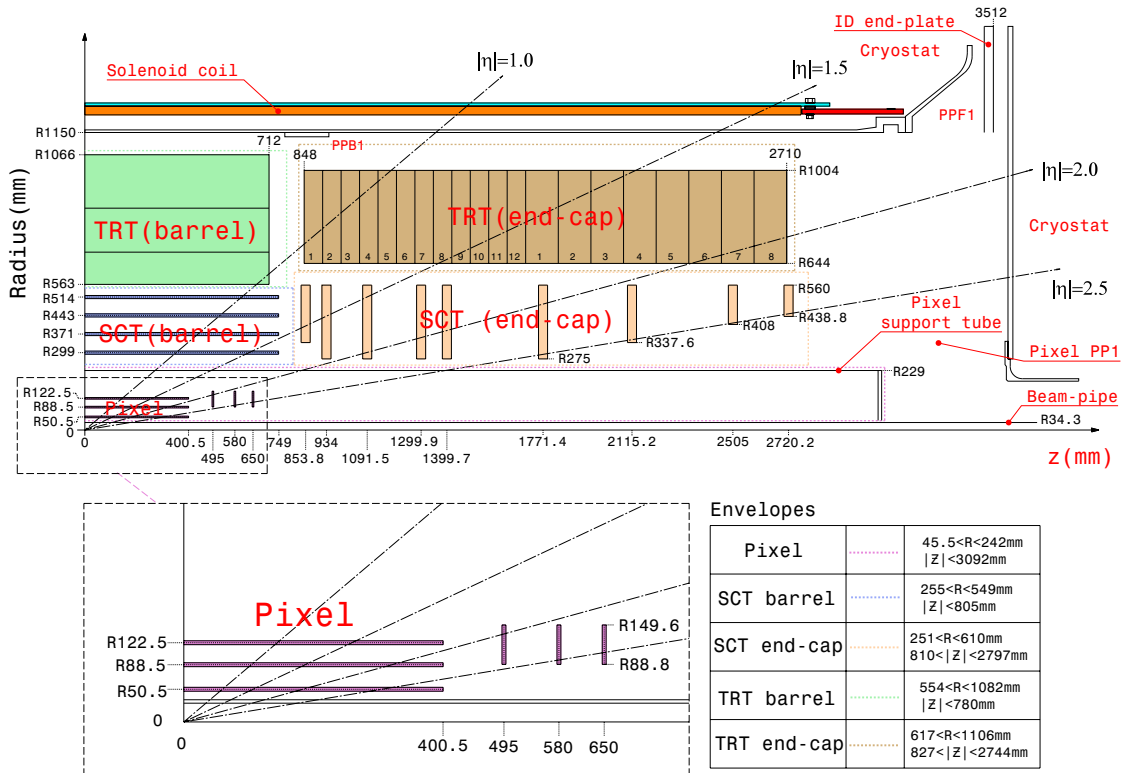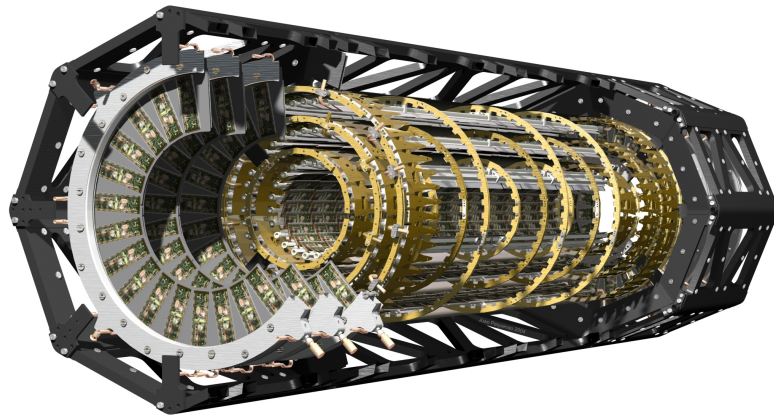

Figure 3.7.: The ATLAS Pixel detector, not included in this Figure is the IBL. ATLAS Experiment © 2018 CERN.

The ATLAS experiment uses hybrid pixel detectors. In hybrid pixel detectors, the pixellated sensor bulk and the read-out parts are on separate chips. These chips are connected via small solder-like bumps, an interconnection technology called *bump bond-*

ing. This decouples the sensor chip from the read-out chip. In monolithic pixel detectors, contrary to hybrid pixel detectors, the sensor and read-out parts are in a single silicon chip.

For the ATLAS detector, oxygenated n-type silicon wafers are used for the pixel sensors with n^+-implants on the pixellated side and a p^+-doped backside. The n-bulk will undergo type inversion after accumulating enough dose. An applied bias voltage is used to deplete the sensitive volume of thermally created charge carriers in order to detect ionisation signals from charged particles (this topic is covered in more detail in the next Chapter). Initially, the depletion region grows from the p-n junction at the backside. The pixels are isolated once the entire bulk becomes depleted. After type inversion, the junction moves to the pixel's side and the depletion region grows from there. This is shown in Figure 3.8. The pixel implants on the frontside are put on ground potential via the connection to the read-out electronics, and a more negative voltage is applied to the backside as the bias voltage.
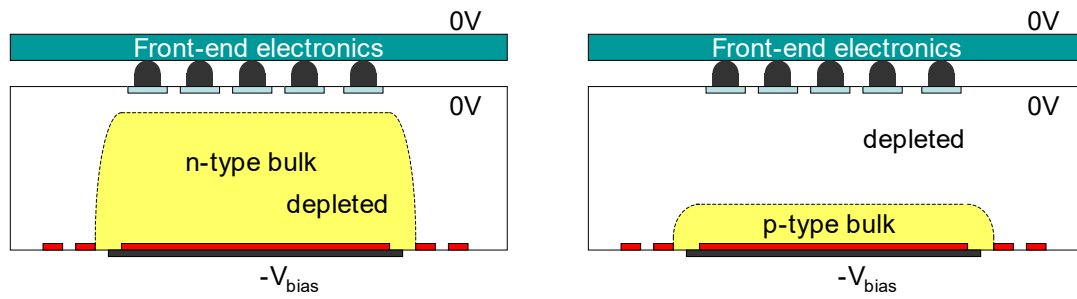
(a) Depletion zone forming from the backside before type inversion.

(b) After type inversion the bulk gets depleted from the pixel side.

Figure 3.8.: Behaviour of the n^+-in-n sensors before and after type inversion.

As the pixels become isolated by default, full depletion is not necessary to operate the sensor. However, maximal depletion provides the largest signal as the entire depleted region contributes to the signal generation. This is desired, given it is technically possible.

Oxygenated silicon is used, as it has been shown, that oxygen impurities reduce the bulk damage by irradiation with charged hadrons. The evolution of the effective doping concentration (N_{eff}) for normal and oxygenated float-zone silicon versus fluence is given in Figure 3.9. The type inversion can be seen as the effective doping concentration initially approaches zero before increasing with
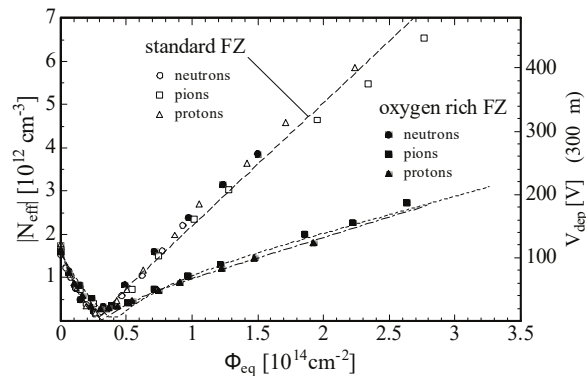

Figure 3.9.: Absolute effective doping concentration ($|N_{eff}|$) versus fluence (Φ) [70].

additional irradiation. A larger N_{eff} requires a higher voltage for full depletion, i.e. for a larger signal. If not irradiated further, the effective doping concentration will change with time. This is known as annealing and for the bulk damage it can be divided into two stages, beneficial annealing and reverse annealing. During the beneficial annealing, N_{eff} will decrease. However, reverse annealing has the opposite effect, an overall increase in N_{eff}. Oxygenated float-zone silicon also exhibits increased beneficial annealing, while the amount of reverse annealing remains the same. Thermal diffusion drives the annealing processes. Thus, they are greatly reduced at lower temperatures, which is one reason why the detector is operated at low temperatures.

The sensors are about 256 µm thick and have a pixel pitch of 400 µm × 50 µm for most of their pixels[4]. Each sensor pixel is attached to a read-out pixel on the read-out chip, the front-end (FE) chip. They are interconnected via bump bonding, using In or SnPb as a bump bonding material.

The front-end is the FE-I3. Each pixel cell in the FE features an analogue part, consisting of a charge sensitive amplifier whose output is processed by an adjustable comparator. The leading and trailing edge timestamps are processed and sent to the chip periphery by the shared digital part of a pixel cell. As the charge sensitive amplifier is DC coupled to the sensor pixel, it has a leakage current compensation to remove any effect of the leakage current on the reset circuit.

The Insertable B-Layer

The IBL upgrade was a modification of the existing ID in the LS1 (cf. Fig. 3.4) which introduced an additional layer of pixelated detector. This improvement could be done by replacing the existing beam-pipe with a smaller one, providing space between the new beam-pipe and the first layer of the existing Pixel detector. A technical sketch of the IBL is shown in Figure 3.10 where the beam pipe, as well as three staves, are indicated. The modules are located within the IBL envelope, ranging between 31 mm and 40 mm away from the nominal IP.

The IBL project aimed at several tasks. It is intended to make the

Figure 3.10.: Drawing of the Insertable B-Layer.

tracking detector more robust to failures and inefficiencies of the innermost Pixel layer, which are inevitable due to radiation damage. Moreover, the LHC performed exception-
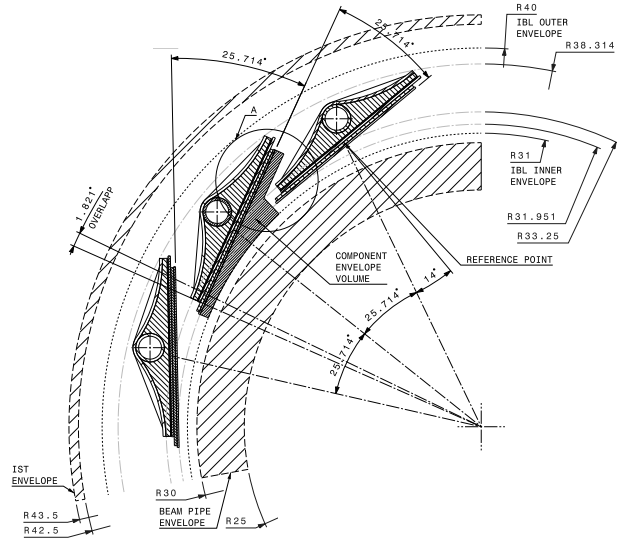
---

[4]Some pixels are prolonged to cover additional area.

ally well, and the instantaneous luminosity exceeded the nominal value. The increased bandwidth needed for more pile-up is a limitation for the original Pixel detector, inefficiencies due to high occupancy would deteriorate tracking performance. Ultimately, the additional measurement point close to the interaction point increases tracking precision.

As stated in the technical design report of the IBL [71]: "*In addition to serving AT-LAS until the HL-LHC upgrade in 2020, the IBL project will develop technologies and valuable experience for the subsequent high luminosity era.*" Aside from the technological developments discussed below, the production and assembly of IBL have led to an experience gain, valuable to the upcoming upgrade projects.

A peculiarity of IBL is that shingling of the modules in the z-direction is not possible due to spatial constraints. As a consequence, sensors with little inactive area, placed as close as possible to each other, were desired to minimise efficiency losses. This pushed the development of novel edge techniques. Two different sensor technologies are used in the IBL, planar pixel sensors and 3D pixel sensors.

The planar pixels sensors are similar to the ones used in the Pixel detector. They are also n^+-in-n sensors, but with a reduced thickness of 200 µm. The pixel pitches have been modified to fit the new read-out chip, the FE-I4, and the inactive edge has been reduced from 1100 µm to 200 µm. This could be done by moving the guard rings, which are located on the p-side of the sensor, beneath the prolonged edge pixel as shown in Figure 3.11. As this impacts the electric field in the region of the edge pixels, the charge collection properties in this region could deteriorate. However, measurements have proven this design to be satisfactory regarding efficiency.
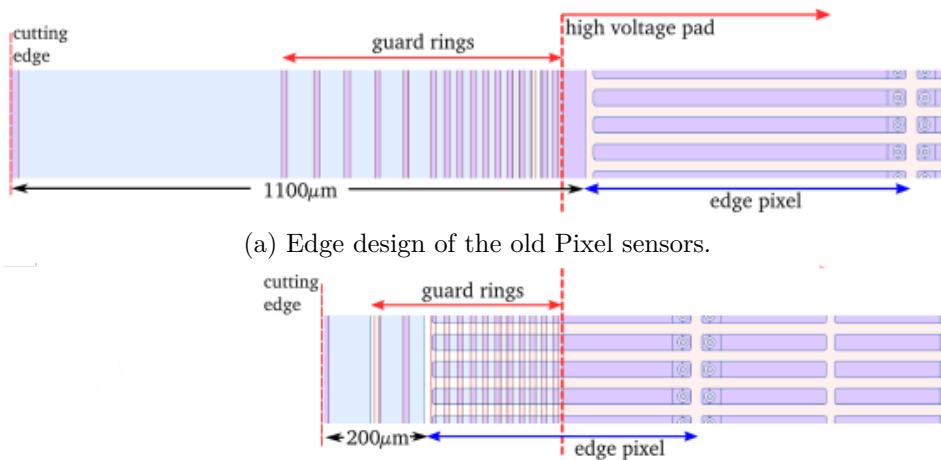
(a) Edge design of the old Pixel sensors.

(b) New design aiming at the reduction of the inefficient edge.

Figure 3.11.: Reduction of the edge in the new pixel design. The guard rings are pushed beneath the edge pixel, the number of guard rings was reduced and the cutting edge was moved closer towards the first guard ring.

The 3D sensors feature three-dimensional electrodes, penetrating into the sensor's bulk. This decouples the sensor thickness, which in turn determines the signal ampli-

27

tude, from the depletion voltage, responsible for the electric field and charge collection distance. As the leakage current is proportional to the depletion voltage and further effects of charge trapping are mitigated by the shorter charge collection distance, 3D sensors are a novel development in the field of radiation hard sensors.
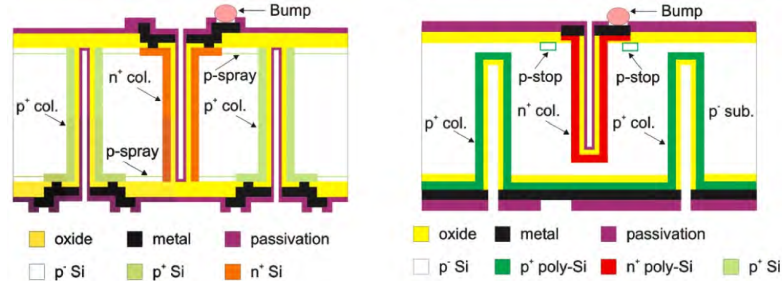

Figure 3.12.: 3D sensors as provided by two vendors for IBL, FBK (left) and CNM (right).

For the 3D in ATLAS, a p-type bulk is used. Columns are etched into the sensor from both sides and then are processed via dopant diffusion. This leads to n^+-columns penetrating into the sensor from the read-out side and p^+-columns from the backside as can be seen in Figure 3.12. The columns have a diameter of approximately 10 µm and are spaced roughly 67 µm apart. The sensors are 230 µm thick, which makes them slightly thicker than the planar sensors. Both, the planar as well as 3D sensors, are read out by the same FE.

The FE-I4[5] features 80×336 pixels of 250 µm $\times$ 50 µm pitch. This correlates with the nominal pitch of the sensor pixels. It is similar to the FE-I3 in the sense that it features similar functionality. An analogue amplification stage is followed by a discriminator and logic which processes the output.

The analogue pixel cell is shown in Figure 3.13. The two amplification stages (green) are followed by the discriminator stage (orange). The discriminator output is connected to a logic AND-gate which allows enabling and disabling of individual pixel cells (purple). To test and calibrate each pixel, each FE has a charge injection circuit (red). Two capacitors, a large one and a small one, can be used to inject charge in the amplifier. This charge injection is an essential tool to *tune* the module.

Two important parameters of the chip can be tuned: the threshold and the charge response. The length of the discriminator output is used to determine the amplitude of the charge signal. This is the so-called time-over-threshold, or short ToT. In the FE-I4 this is a 4 bit value. The principle of ToT is depicted in Figure 3.14, where on the left, the effect of a small and large input charge is shown. A smaller charge will result in a lower signal. As the discharge slope is independent of the charge, the overall time the signal stays over the threshold is shorter. In consequence, the physical quantity of amplified charge is proportional to ToT. Furthermore, a smaller charge will have a

---

[5]The prototype of the final front-end was the FE-I4A, the newest one the next revision, the FE-I4B. FE-I4 will be used as a synonym for the FE-I4B, i.e. the one in use in the ATLAS detector.
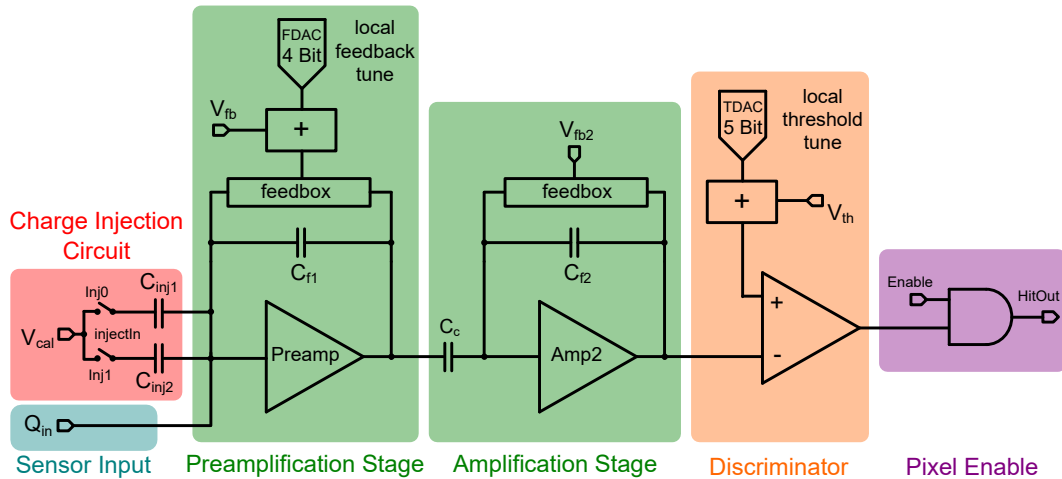
Figure 3.13.: The analogue part of an FE-I4 pixel. The different parts are highlighted.

slightly less steep rising edge. This effect is known as time-walk and also depicted in Figure 3.14.

A higher discharge current (Fig. 3.14, centre) will reduce the time the signal needs to return to baseline by increasing the steepness of the trailing slope. This is exploited in the ToT tuning, where the module is tuned in a way, that a certain injected charge results in the desired ToT. In Figure 3.13 this is done in the section labelled *local feedback tune*. The FDAC is the digital to analogue converter (DAC) which provides the reference voltage for the feedback current. In addition, a global[6] feedback voltage (V_{fb}) is added to all FDACs.

The second stage of the amplifier provides an additional gain by a fixed factor of $C_c/C_{f2} \approx 6$, providing freedom in the choice of C_{f1} which can be optimised for the expected signal charges and desired discharge times.

In Figure 3.14 the rightmost sketch shows the effect of the discriminator threshold. Small signals below the threshold will not cause the discriminator to go into the high state. The threshold tuning will set the threshold to a value that signals below the desired threshold charge will not be registered, while signals above threshold will. The threshold tuning has a substantial impact on the ToT tuning. In the actual FE, also the ToT tuning will impact the threshold tuning. This is not depicted in Figure 3.14, as these sketches show a simplified model. In the actual FE, the discharge current will additionally impact the signal shape, though much less than vice versa.

In Figure 3.13 the threshold tuning is done by providing the adequate reference voltage to the comparator via the *local threshold tune* section, in particular with the TDAC, the threshold DAC. Again, a global threshold voltage (V_{th}) is added to all TDACs.

A more detailed explanation of the signal creation in Silicon detectors will be given in Chapter 4.

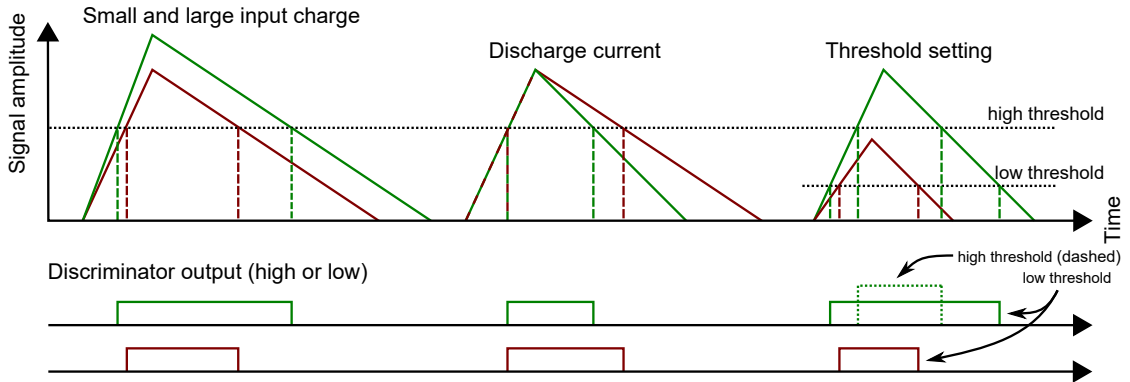[6]This voltage is common for all pixels of the FE.

Figure 3.14.: Behaviour of the signal and discriminator output for different input charges (left), for different discharge currents (centre) and a high and low threshold (right).

The Silicon Strip Detector and Transition Radiation Tracker

The SCT consists of four barrel layers and nine disks at each end. It uses conventional p-in-n sensors with a thickness of 285 μm. The strips are AC coupled to their respective binary read-out ASICs[7]. The mean strip pitch is 80 μm, the disk modules feature radial strips. Two layers of strips are used per module with ±20 mrad stereo rotation to provide a full two-dimensional hit point.

The TRT uses polyimide drift tubes with a diameter of 4 mm with a 31 μm thick tungsten wire in the centre acting as the anode which is directly connected to the read-out ASICs. They are kept at ground potential, whereas the cathodes are put to a more negative potential of approximately -1.5 kV. The straw tubes are filled with a 70 % Xe, 27 % CO_2 and 3 % O_2 mixture at a slight overpressure of 5 mbar to 10 mbar.

In the barrel region, the TRT can be divided into three rings, each with 32 triangular modules. Each module is composed of several $\mathcal{O}(100)$ straws. Up to $|\eta| = 2$, a particle is expected to traverse at least 36 straws, with this number decreasing to 22 at the transition from mere barrel hits to the barrel-end-cap-transition at $0.8 < |\eta| < 1$.

The mechanical layout of the SCT and TRT can be seen in Figure 3.6.

3.2.2. The Calorimeters

The different parts of the calorimeters use different absorber and active materials. The parts closest to the beam pipe use liquid argon (LAr) as an active material and are hence enclosed in a cryostat. Three cryostats exist for this purpose, one in the barrel region and one for each end cap. The calorimeter is shown in Figure 3.16, where the different systems are depicted.

The LAr electromagnetic barrel and LAr electromagnetic end cap (EMEC) form the EM calorimeter and are entirely in the cryostats. The Tile barrel and Tile extended

---

[7]ASIC: Application-specific integrated circuit.
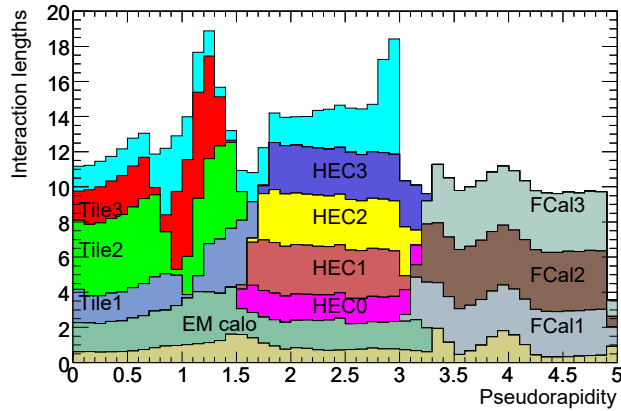
Figure 3.15.: Distribution of the material's interaction length for various $|\eta|$ [72].

barrel are the parts of the hadronic calorimeter which are outside of it. Figure 3.15 gives the interaction length for a given pseudorapidity which clearly shows the different areas where each calorimeter operates.
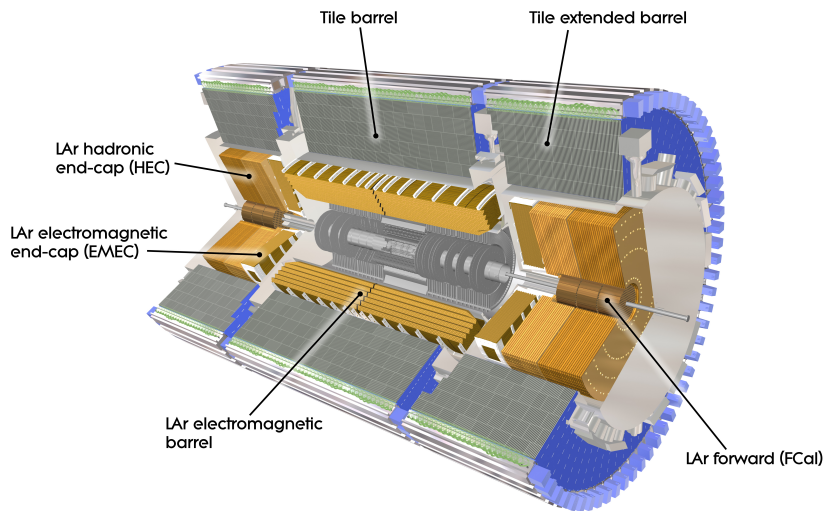
Figure 3.16.: Overview of the different calorimeter subsystems in the ATLAS detector. ATLAS Experiment © 2018 CERN.

The Electromagnetic Calorimeters

Both the barrel and end cap sections of the EM calorimeter use lead as an absorber material and LAr as the active material. They are arranged in an accordion pattern to

provide coverage without any cracks. In total, the EM calorimeter provides coverage up to $|\eta| < 3.2$ (cf. Fig. 3.15). A total of at least 22 radiation length worth of material is present in the EM calorimeter. Together with the fine segmentation, compared to the hadronic sections, it provides precision measurements for photons and electrons with an angular coverage slightly larger than the ID.

The Hadronic Calorimeters

The tile calorimeter is situated right outside the barrel section of the EM calorimeter. It uses steel as an absorber and scintillating tiles as an active material. At the centre ($\eta = 0$) the total material penetrated after the tile calorimeter is 9.7 interaction lengths, which is enough to prevent any spill-over of hadronic decays into the muon systems. The scintillating tiles are attached via wavelength shifting fibres to photomultiplier tubes (PMT).

The tile extended barrel region sits outside of the end cap cryostats, uses the same modules as the tile barrel calorimeter and extends angular coverage up to $|\eta| = 1.7$

The HECs, using LAr, are located right behind the EMECs inside the end cap cryostats. To reduce the drop in material density and ensure full containment of the energy deposition, there is a slight overlap of the HEC and tile calorimeter as well as of the HEC and FCal; this can be seen in Figure 3.15. The HECs uses Copper plates as an absorber material.

The forward calorimeters provide coverage in the very forward region between $3.1 < |\eta| < 4.9$. As shown in Figure 3.16, the FCal is displaced from the interaction point slightly back. This is to reduce neutron background but requires a more compact and dense design. Situated also inside the end cap cryostats, the FCals use LAr as the active material. There are three different sections. The first one uses copper to predominantly measure electromagnetic processes, whereas the subsequent ones use tungsten. This is to enhance the electromagnetic interactions in the EM region as well as the hadronic interactions in the hadronic region, respectively, due to the material dependent interaction probability. The FCal has to deal with high particle fluxes, due to its very forward location.

The hadronic calorimeters are optimised to satisfy the requirements for jet reconstruction and $E_\mathrm{T}^\mathrm{miss}$ measurements, which require at least a certain granularity, i.e. spatial resolution, as well as energy resolution.

3.2.3. The Muon System

The muon system uses magnetic fields to bend muons and measure their momenta up to a pseudorapidity of $|\eta| = 2.7$, the key component to create these magnetic fields are the large superconducting air-core toroid magnets, which also gave ATLAS its name. They are responsible for the deflecting magnetic field in the barrel region. Additional, smaller end cap magnets complement the field in those regions. The field configuration is that the field is perpendicular to the muon tracks in most cases. The various muon chambers and assisting magnets are shown in Figure 3.17.
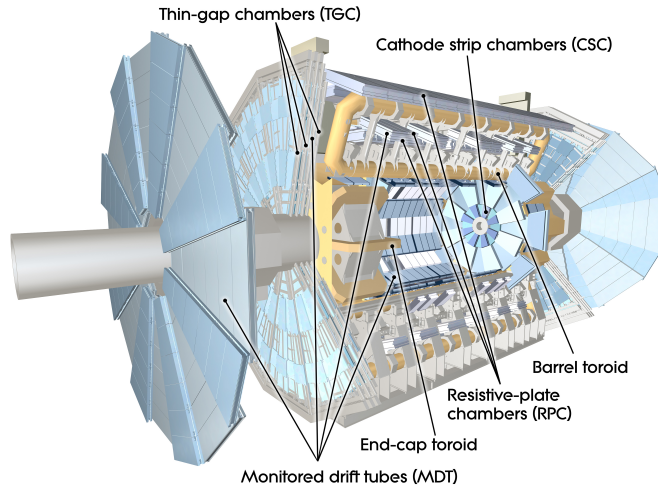
Figure 3.17.: The muon chambers inside of the ATLAS detector, also the magnets are indicated. ATLAS Experiment © 2018 CERN.

Two types of precision tracking chambers are used: Monitored Drift Tubes (MDTs) and Cathode Strip Chambers (CSCs). The MDTs are used in the central region over the range $|\eta| < 2.7$, however only up to $|\eta| = 2.0$ in the inner regions. To cover the region $2.0 < |\eta| < 2.7$, CSCs are used in the inner layers. They feature a finer granularity and can cope with the higher flux and more challenging conditions in this very forward region, close to the interaction point.

Tracking chambers are also used for triggering. They provide measurements of the coordinate orthogonal to the direction of the precision chambers. Resistive Plate Chambers (RPCs) are used in the barrel region $|\eta| < 1.05$, followed by Thin Gap Chambers (TGCs) for $1.05 < |\eta| < 2.7$.

MDTs use 29.97 mm tubes filled with an Ar (97 %) and CO_2 (3 %) mixture at 3 bar. Electrons from ionisation are collected by the 50 µm thick wire running through each tube, held at a potential of approximately 3 kV. A limitation of the MDTs used in ATLAS is the maximal rate of 150 Hz/cm^2, which is why multiwire proportional chambers with cathode strips are used in the high-η regions with higher particle rates. The anode wires are kept at a potential of 1.9 kV, and both cathodes are segmented, one in the direction orthogonal to the wires (the precision coordinate) and one parallel to them. Similar to the MDTs, an Ar (80 %) and CO_2 (20 %) gas mixture is used. The lack of hydrogen in the system makes it more robust against neutron background, which poses a problem in the forward region.

The RPCs used in the triggering chambers use segmented parallel electrode plates. Two of such chambers, with strips oriented orthogonal, form a unit. The resistive plates in one RPC are kept at a distance of 2 mm and are filled with a gas mixture to optimise avalanche operation. If a charged particle passes through the 4.9 kV/mm electrical field, avalanches of electrons will form at the ionised tracks (Townsend discharge). The

electrical signal picked up from the RPC strips has a jitter below 1.5 ns, due to this working mechanism and high field. This is lower than that of the previously mentioned wire chambers and makes it a viable detector for triggering.

The last type of detector used, the TGCs, are again multiwire proportional chambers. As the wires are all at the same positive voltage, tracks which pass exactly between two wires cause a problem. The lack of a (large) drift field makes signal formation very imprecise regarding time, as initially the released charges will diffuse. This is not a problem for the TGCs as they are in the wheels in the end caps, a nominal track emerging from the interaction point will have sufficient inclination. Together with the small wire-to-wire distance in the TGC, a fast signal formation is guaranteed, making the TGCs useful triggering detectors. As mentioned, the triggering detectors also complement the MDTs in the measurement of the second coordinate.

3.3. The ATLAS HL-LHC Upgrade

As discussed, the accelerator will be upgraded to provide a levelled[8] luminosity of $\mathcal{L} \approx 5 \times 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$ and an average pile-up of $\langle \mu \rangle = 140$.[9] This poses a challenging environment for the ATLAS detector. To cope with the high pile-up, the ATLAS collaboration agreed upon replacing the entire tracking detector, i.e. IBL, Pixel, SCT and TRT with the new *Inner Tracker*, or short, ITk. The ITk upgrade aims at building a detector to cope and withstand the HL-LHC challenges. As stated in the ITk Strip Technical-Design-Report (ITk Strip TDR) [73], the performance of the current tracker would significantly comprise the physics reach with the upcoming upgrade. This is due to the inabilities of the various subsystems to cope with the high track rates.

The ITk detector has to deal with a higher track rate, i.e. a higher occupancy which also results in higher data throughput to the off-detector components. Despite this, the tracking performance needs to remain as high as possible in order to separate the various tracks. At the same time, the subsystems will need to withstand the harsh radiation environment, the innermost layers of the planned pixel detector will be exposed to an 1 MeV neutron equivalent fluence of $\phi_{\mathrm{neq}} \approx \left(2 \times 10^{16}\right)\,\mathrm{cm}^{-2}$. Due to the quadratic distance law, the outermost layers are exposed to a significantly lower dose, as can be seen in Figure 3.18[10].

The conceptual layout of the ITk is shown in Figure 3.19. The ITk detector will be composed of five barrel layers of pixelated detector (shown in red) near the beam pipe, and four barrel layers of strip detector (blue) further outside. Furthermore, disks will provide additional angular coverage. The pixel detector will reach up to $|\eta| = 4$. To guarantee good tracking performance, a replacement of the innermost two pixel layers is

---

[8]At the beginning of a run the instantaneous luminosity is artificially reduced (*levelled*) to operate at a tolerable amount of pile-up.

[9]Also the option of $\langle \mu \rangle = 200$ is considered. Hence, some predictions and plots assume this higher pile-up.

[10]The exact layout is not finalised, however as the distance to the interaction point will not decrease, the fluences the innermost layers need to be able to withstand, will not change significantly.
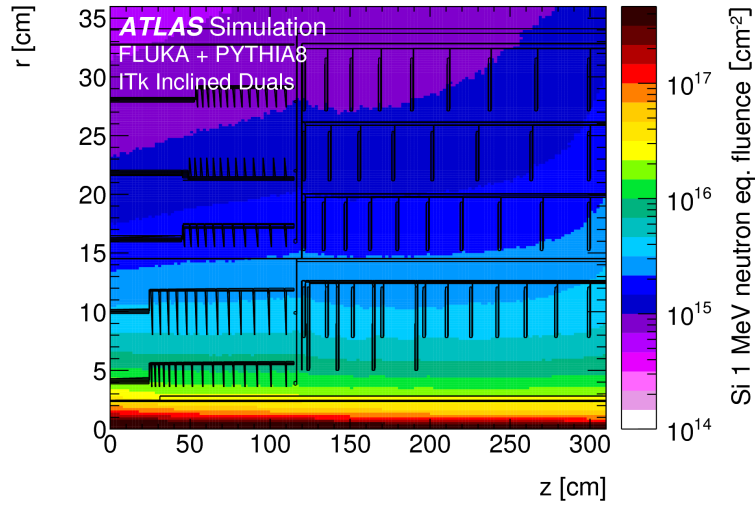
Figure 3.18.: 1 MeV neutron equivalent fluence for the planned new pixel detector. Shown are the five barrel layers and disks for the ITk upgrade after 4000 fb^{-1} of collisions [74].
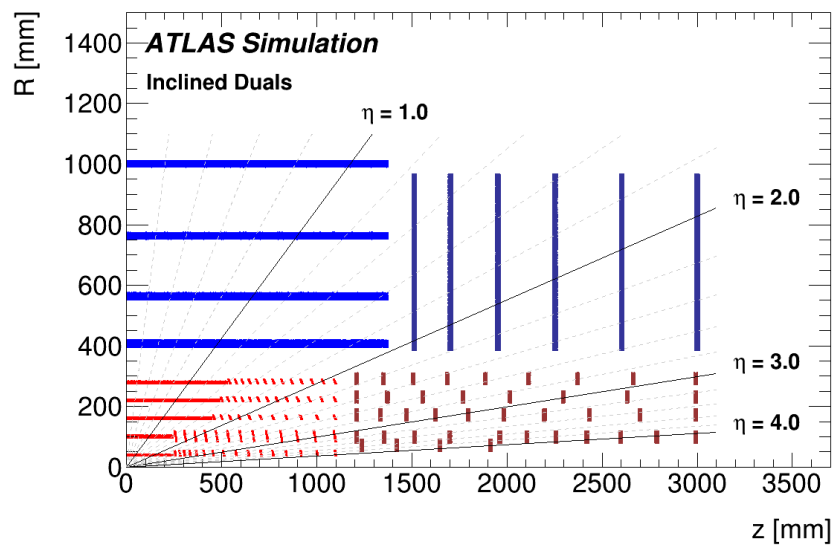
Figure 3.19.: The approximate layout for the ITk upgrade. The pixel detector in red and the strip layers in blue [74].

foreseen. This puts requirements on the mechanics, as these layers need to be extractable from the detector. In order to maintain good tracking performance, the radiation damage is a crucial factor to be considered when deciding upon the technology used.

35

3.3.1. The ITk Pixel Read-Out Chip

A new read-out chip will be used in the ITk pixel detector. The RD-53A is a prototype of this read-out chip. It features $50\,\mu m \times 50\,\mu m$ pixels, [11] the smaller pixel pitch compared to the FE-I4 increases the resolution and reduces the chance of one pixel being hit by two separate tracks.

The FE is designed to have a lower threshold than the FE-I4 to compensate signal loss due to radiation damage in the sensor bulk. At the same time, the read-out sections have been improved to deal with the higher hit rates. Accordingly, the output bandwidth has been increased to 5.12 Gbit/s. The FE has a radiation tolerance of at least 500 Mrad. The entire chip is supposed to consume less power per area, to reduce the need for cooling infrastructure. To further decrease cables and thus material, a serial powering scheme will be used in the final detector.

In contrast to the FE-I4, which used IBM 130 nm technology, the RD-53A is manufactured in 65 nm CMOS technology. There are several reasons which motivated this decision. Firstly, the 65 nm radiation tolerance has been investigated by other R&D projects and proven to be feasible, and secondly, the smaller feature size is necessary to incorporate all the mentioned functionality. Moreover, fewer and fewer vendors still operate the outdated 130 nm production lines, making the 65 nm choice more future-proof.

Another important upgrade of the FE is the possibility to reload the configuration during data taking. While some of the important registers are redundant, it is not possible to incorporate this for all registers. Ionising radiation can induce a bit-flip, i.e. change the internal state of a bit in the chip, if this bit is not redundant. This is a so-called single event upset (SEU). For example, in Figure 3.20 flips of the enable bit (cf. Fig. 3.13) versus run duration in the current ATLAS detector are shown. This poses a problem, as the reconfiguration of the FE during data taking was not foreseen and would impose a

Figure 3.20.: SEUs of the current pixel FE during one run. Shown is the number of disabled pixels which were activated by an SEU on different modules [75].

large dead time. Instead, a workaround aims at a (partial) reconfiguration during short pauses in data taking to mitigate SEUs in the current detector. For the upgrade, this functionality should be intended a priori.
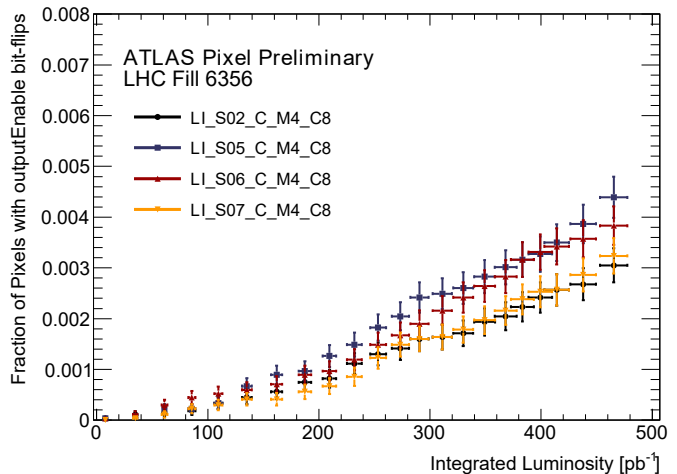
---

[11]The sensor pixels will either have the same dimension of $50\,\mu m \times 50\,\mu m$ or $25\,\mu m \times 100\,\mu m$.

3.3.2. ITk Pixel Sensors

With the standard high-resistivity silicon sensors, e.g. the planar sensors in the Pixel and IBL detector right now, radiation damage will progressively reduce the amount of collected charge. As a countermeasure, increasing the bias voltage will mitigate this effect by increasing the charge collection speed and reducing the probability of trapping charge carriers, as well as ensuring full depletion of the bulk. An increase in the bias voltage also increases the leakage current, which has to be compensated by the FE. The leakage current is proportional to the temperature. This may cause a positive feedback loop if the current gets large enough to heat up the sensor. This is called *thermal runaway*. This limits the parameter space for operation and needs to be considered when making the sensor choice, with the boundary conditions of the required tracking performance, and cost as well as production yield. The exact technology that will be used is not decided yet. However, an overview of most probable candidates will be given.

The innermost layer is likely to use 3D sensors, due to their superior radiation hardness and the R&D progress in the last couple of years. This has increased the production yield, which was one limiting factor of this technology. A remaining challenge is the smaller pixel pitch, moving the electrodes closer together increases the pixel cell's capacitance which increases the electronic noise of the attached amplification circuit. Recent studies, where $50\,\mu m \times 50\,\mu m$ cells were irradiated up to $\phi_{neq} = \left(3 \times 10^{16}\right)$ cm^{-2} show a hit reconstruction efficiency of 98 %. At $\phi_{neq} = \left(2.8 \times 10^{16}\right)$ cm^{-2} the sensors showed a hit efficiency of the nominal 97 % at a depletion voltage of 150 V [76], which makes them the most promising candidate as of today for the innermost layers.

Planar sensors are foreseen for the remaining layers. Contrary to the sensors in the current Pixel and IBL detector, n-in-p sensors are the most viable candidates. N-in-p sensors have a simpler and thus cheaper manufacturing process than n-in-n sensors.

There are three notable developments in this field, relevant to the ITk project. Firstly, sensors with a thickness of 100 µm or 150 µm will reduce the leakage currents after irradiation. However, the thin sensors are more difficult to handle. Secondly, to take CV- or IV-curves on the wafer at the testing procedure during production, made to assess the quality of the delivered wafers, so-called *bias grids* are used to connect the floating pixel implants. The traces of the bias grid impact the electric field in the sensor and can reduce the detection efficiency, especially in irradiated sensors. This effect becomes more relevant as the pixel pitches get smaller and smaller, and the width of the bias traces reach the dimension of the pixels. Thirdly, novel thin edge techniques, already introduced for IBL, are investigated to reduce inactive area.

The radiation hardness of such a sensor has been studied with thin IBL style sensors, irradiated to $\phi_{neq} = \left(1 \times 10^{16}\right)$ cm^{-2}, where efficiencies of about 96 % to 97 % were reached at bias voltages of 400 V to 600 V. Smaller pitches have been studied as well. The results indicate that several vendors will be capable of providing sensors which can withstand the desired fluences.

It should be noted, that many of the measurements made with those sensors use the old FE-I4. In particular, for thin planar sensors which suffer from signal loss, the FE-I4 has a higher low threshold limit, i.e. it cannot reach as low thresholds as desired, as well

as a higher noise level than the RD-53A and final read-out chip. Hit detection efficiencies depend on the entire module, i.e. the FE itself can limit the efficiency of an irradiated sensor. That is why, future validation and study of the feasible sensor candidates are crucial to qualify sensors. However, current measurements with the FE-I4 provide a lower limit for the efficiency of the module, which might rise due to lower threshold tuning with a newer FE.[12]

The 3D, as well as planar sensors, are hybrid sensors, which means that the read-out chip is bump bonded to the sensor chip. As the bump bonding process with such low pitches and thin sensors is custom to HEP detector development, it is not widely available and costly. Therefore, it would be of advantage to make use of industry standard processes in HEP detector production. This would guarantee a good yield and lower manufacturing costs. A possibility for the outermost layer is monolithic CMOS sensors. These would use a standard CMOS process and feature the sensor and front-end on the same chip, i.e. a monolithic design, which would reduce the cost even further. CMOS sensors are a possible candidate for the outermost layer of the pixel detector, where the hit rates, as well as fluences, are not as demanding as in the innermost layers.

---

[12]Some measurements were also carried out with an RD-53A prototype.

Semiconductor Sensors

Semiconductor sensors work by charged particles passing through them and ionising the bulk material. This ionisation track leaves an electronic signal which can be detected and processed.

4.1. Energy Deposition in Matter

The average energy loss per distance is given by the Bethe-Bloch-equation [77] (Equation 4.1, confer Table 4.1 for an explanation of the used symbols, β and γ are the usual relativistic kinematic factors).

$$-\left\langle \frac{dE}{dx} \right\rangle = Kz^2 \frac{Z}{A} \frac{1}{\beta^2} \left[\frac{1}{2} \ln \left(\frac{2m_e c^2 \beta^2 \gamma^2 T_{\max}}{I^2} \right) - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right] \tag{4.1}$$

With:

$$K = 4\pi N_A r_e^2 m_e c^2 \tag{4.2}$$

$$T_{\max} = \frac{2m_e c^2 \beta^2 \gamma^2}{1 + 2\gamma m_e/M + (m_e/M)^2} \tag{4.3}$$

The Bethe-equation provides a good approximation in the range of $0.1 \leq \beta\gamma \leq 10^3$. Often, the energy loss per distance and unit density (called stopping power) is used, as is the case in Figure 4.1 where the stopping power of copper on μ^+ is shown. The region labelled *Bethe* is the one where the Bethe-equation yields a good approximation.

The assumptions made in Eq. 4.1 suppose a first-order (Born) cross-section. This fails at lower energies, and higher-order shell corrections need to be introduced. Furthermore,

the electric charge becomes relevant. Negatively charged particles feature a reduced stopping power (μ^- indicated in Fig. 4.1) compared to positively charges ones.

In the other case of very high energies, radiative effects become much more important than ionising ones. In this region, the energy loss due to e^+e^- pair production, bremsstrahlung and photonuclear effects scales linearly with energy while the ionisation contribution remains approximately constant and becomes negligible at higher energies.

Symbol	Definition
M	Incident particle mass
E	Incident particle energy
T	Kinetic energy
m_e	Electron mass
r_e	Classical electron radius
N_A	Avogadro's number
ze	Charge of incident particle
Z	Atomic number of absorber
A	Atomic mass of absorber
I	Mean excitation energy
$\delta(\beta\gamma)$	Density effect correction

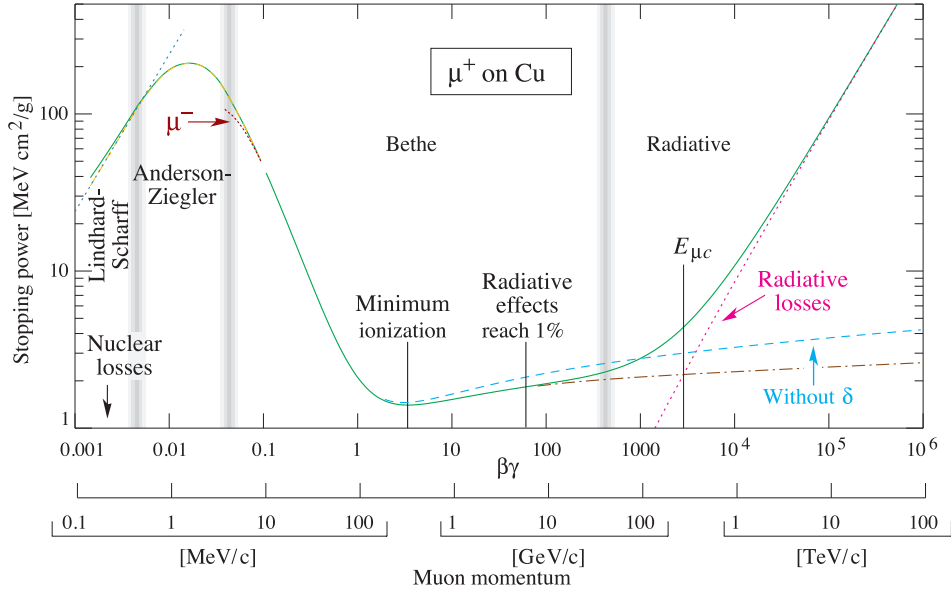Table 4.1.: Symbols appearing in the Bethe-equation.

Figure 4.1.: Stopping power for anti-muons in copper normalised by density. Particle Data Group © 2018.

Despite these limitations, the Bethe-equation provides an important description to estimate the energy loss in the region which is of interest for semiconductor trackers. It can be seen in Figure 4.1 that the stopping power is nearly constant in the range of $1 \lesssim \beta\gamma \lesssim 10$, labelled *minimum ionisation*. A hypothetical particle which behaves in the way predicted in this plateau region is a so-called minimum-ionising particle, or short MIP.

The concept of MIPs is useful for two reasons. Firstly, most particles will behave MIP-like in the field of application for semiconductor trackers. Secondly, the MIP provides the minimal ionisation, i.e. the smallest possible signal. While in other applications of semiconductor imaging, too large a signal might cause a problem as the dynamic range of the read-out chip might be too small and internal counters could saturate, this is not a problem in the detector for the ATLAS experiment. Here, the limiting factor is the deterioration of the collected signal due to radiation damage, a MIP is always the *worst case* regarding signal amplitude. As a rule of thumb, a MIP will experience a stopping power of $1.5 \text{ MeV cm}^2/\text{g}$. There is a slight Z dependence for this value, decreasing with increasing Z. However, as a first approximation, it holds for all materials with the exception of hydrogen.

The Bethe equation (Eq. 4.1) only provides the mean energy deposition, not the energy loss distribution for a fixed $\beta\gamma$. Rare high energy collisions lead to pronounced tails at high energy depositions, if looking at the distribution of energy loss for individual particles. For very thick sensors, the distribution will turn Gaussian again, but in the regime of typical sensor thicknesses, $\mathcal{O}(100)$ µm, they are well described by a Landau distribution. For even thinner sensors, the Landau description fails as described by H. Bichsel [78]. A better description are the so-called straggling functions. This has been experimentally verified by S. Meroli et al. [79]. In both descriptions, the mean energy loss and the most probable value differ as the distributions are skewed, the most probable value being lower than the mean energy loss.
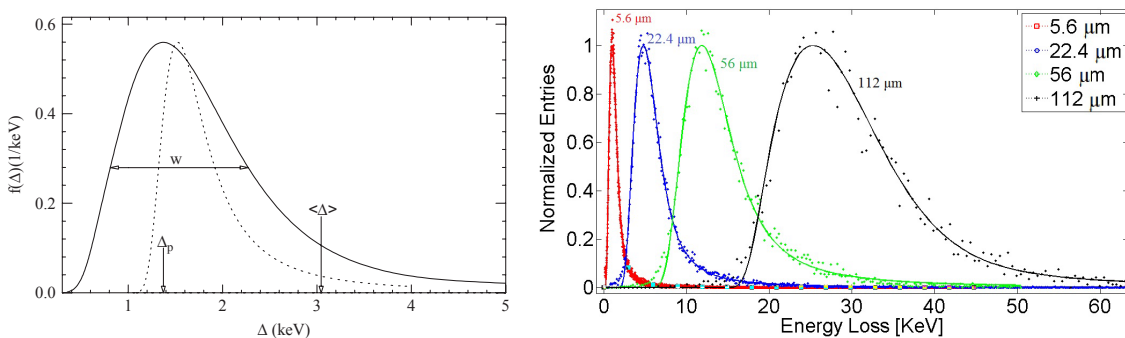
(a) Straggling functions (solid) compared to the Landau description (dotted) [78].

(b) Measurements (dots) of the energy loss of 12 GeV protons in silicon [79].

Figure 4.2.: The Landau distribution and straggling functions for the energy loss of individual particles.

In Figure 4.2a the predicted energy loss for particles at $\beta\gamma = 3.6$ passing through

1.2 cm of Ar gas is shown. The solid distribution shows the prediction via the straggling functions, for comparison the Landau distribution is also indicated by the dotted distribution. Furthermore, the mean energy loss is indicated ($\langle\Delta\rangle$) as well as the most probable value (Δ_p). In Figure 4.2b the actual measurements for various thin silicon sensors and the fit of straggling functions to data are shown.

4.1.1. Charge Carriers in Silicon

The process of ionisation will create free charge carriers in a semiconductor material. The band gap of silicon is 1.12 eV at 300 K, and the number of released charge carriers is proportional to that energy. However, not all the deposited energy will release charge carriers, and instead, lattice vibrations are excited as well, due to momentum conservation. This increases the mean energy E_i required to produce an electron-hole-pair (e-h-pair) to 3.67 eV at room temperature. A back-of-the-envelope estimation yields a signal of $\mathcal{O}(2{\times}10^4)$ released e-h-pairs in a 200 µm thick sensor.

The variance in the number of released charge carriers does not merely follow Poissonian statistics. The square root of the number of released charge carriers, $N = E/E_i$, is modified by the Fano-factor F, hence $\sigma_N = \sqrt{FN}$. As the electronic and lattice excitations are coupled, and there is no continuous way of exciting them, the actual variance is decreased. This results in $F \approx 0.1$ for Si. As a consequence, the energy resolution $\sigma_E/E = \sqrt{FE_i/E}$ is better than one would naively expect from a mere Poisson-driven charge carrier release process.

Thermal excitation, as dictated by the Fermi-Dirac distribution, will release charge carriers into the conduction band. At room temperature, typical intrinsic charge carrier densities n_i in silicon are of the order $\mathcal{O}(10^{10}\ \mathrm{cm}^{-3})$. The same 200 µm thick sensor, assuming an area of 1 cm^2, would have $\approx 2{\times}10^8$ thermally created e-h-pairs in its bulk. Hence, to use silicon as a sensor material, it needs to be depleted, which is achieved via a pn-junction and applying a reverse bias voltage.

4.2. The pn-Junction

Introducing tri- or pentavalent elements into a Si bulk dopes the semiconductor. These impurities add additional electrons or holes into the silicon lattice. Pentavalent elements introduce additional electrons, are hence known as *donors* and act as an n-type dopant. Respectively, trivalent elements add holes, are known as *acceptors* and are p-type dopants.

If an n-doped and p-doped bulk are brought together, a so-called pn-junction is formed. In the region of the pn-junction, diffusion will move electrons from the n-doped region into the p-type region and vice versa for holes. The introduced impurities in the crystal lattice will not move and cause the bulk to have a net charge in this region. The n-type bulk will have positively charged ions in the lattice, the p-type one negatively charged ones. This results in a space charge region which creates an electric field, limiting the

diffusion so that an equilibrium is reached. As the free charge carriers diffuse into the other region they recombine and are so removed, this forms the so-called depletion region.

Applying a more negative voltage to the p-doped region will create an electric field which aligns with the intrinsic field of the junction, causing the depletion region to increase. Applying a voltage in this manner is called reverse biasing the junction. This is the mode semiconductor trackers are operated since, as discussed, the thermally created intrinsic charge carriers would dominate any signal in a non-depleted bulk.

In detectors, typically an asymmetric structure with a high and a low doped region of opposite doping types is used. As discussed in Section 3.2.1, the existing ATLAS detector used n^+-in-n pixels with a p^+-backside. Before type inversion, the junction is at the backside, the p^+-backside forms the junction with the n-bulk. After type inversion, the n^+-pixel implants form the junction with the (now) p-bulk. The asymmetric doping has the consequence that the depletion region grows predominantly into the lightly doped region, depleting the major sensor bulk in the case of the ATLAS pixel sensors.

4.3. Signal Formation

Most pixel detectors are built to collect electrons. Hence, the electric field will be such, that the electrons drift towards and are collected by the electrodes connected to the read-out pixels, whereas the holes drift towards the backside. This causes a signal that is processed by the read-out electronics. However, it is important to note that already the drift of the charge carriers in the bulk will induce a current on the electrodes. This has been described by W. Shockley [80] and S. Ramo [81] and is known as the Shockley-Ramo theorem. The theorem relates the induced current on an electrode i to the charge carrier's velocity $\vec{v}$ and the weighting field $\vec{E_w}$ for this electrode:

$$i = q\vec{E_w}\vec{v} \tag{4.4}$$

The weighting field is the electrical field which is obtained if the electrode of interest is put to unit potential, the remaining electrodes to ground potential and any free charges are removed from the bulk.

4.4. Multiple Scattering

Mostly Coulomb scattering is responsible for the deflection of a particle at small angles, described by the Rutherford cross-section. In the case of many, subsequent small angle scatterings, the resulting net angular distribution will be Gaussian, as stated by the central limit theorem. The width of this Gaussian distribution, projected into a plane, is approximated by:

$$\theta_{\text{plane}} = \frac{13.6 \text{ MeV}}{\beta c p} z \sqrt{x/X_0} \left[1 + 0.038 \ln\left(x/X_0\right)\right] \tag{4.5}$$

The width of the spatial distribution is given by $\theta_{\text{space}}^2 = \left(\theta_{\text{plane,x}}^2 + \theta_{\text{plane,y}}^2 \right)$ where x and y are the directions orthogonal to the one of movement. The $\theta_{\text{plane,i}}$ distributions in both directions are identical.

There is a limitation to this approximation. Rare hard scattering events cause non-Gaussian tails. As they are less frequent, they are not smeared out by the central limit theorem and the scattering distribution also features pronounced tails. The given Equation 4.5 is a good approximation for the central 98% of the distribution, which is sufficient for many applications.

4.5. Radiation Damage

There are two types of damage which need to be considered when asserting the impact of radiation on the detector modules, bulk damage and surface damage. Bulk damage is damage to the crystal lattice, due to the displacement of atoms in the bulk. This damage is proportional to the nonionising energy loss (NIEL). Hence, it is important to look at the total non-ionising energy deposition to that material. As the incident particle type and energy is important, the NIEL model [82] is used to quantify radiation damage to the bulk. Bulk damage leads to an increased leakage current and formation of space charges which impact the electric field in the sensor. Certain types of bulk damage will trap released charge carriers, which reduces the amount of collected charge. The higher mobility of electrons compared to holes (electrons in crystalline silicon have a two- to threefold mobility) is a reason why after irradiation they are more likely not to be trapped. This is a reason why typically the pixel implants collect electrons and not holes.

Charge buildup in surface layers is the other type of damage which needs to be considered. This damage is proportional to the ionising energy. Hence, not the NIEL model is used, but the total amount of absorbed ionising energy, the ionising dose. This damage modifies the behaviour of semiconductor integrated circuits and hence is the dominating damage to the FE electronics.

A prominent effect due to surface damage is the Total Ionising Dose (TID) effect in the FE-I4 during operation [83]. Charge buildup at the transistor gate leads to leakage current in the transistor, posing a problem as the supply current of the FE is increased to a critical point due to this damage.

Test Beam Measurements

5.1. Test Beam Infrastructure

Full system tests are needed to qualify and quantify properties of novel sensors, read-out chips, or even entire DAQ systems. While several sensor and module properties are measured in a lab environment, for example, IV- or CV-curves or measurements with a radioactive source, ultimately measurements with a collider-like particle source need to be performed. Using track information enables precise measurements of spatially resolved sensor properties, for example inefficiency measurements and charge distribution properties, even on a sub-pixel scale. An example of such an analysis is schematically depicted in Figure 5.1.
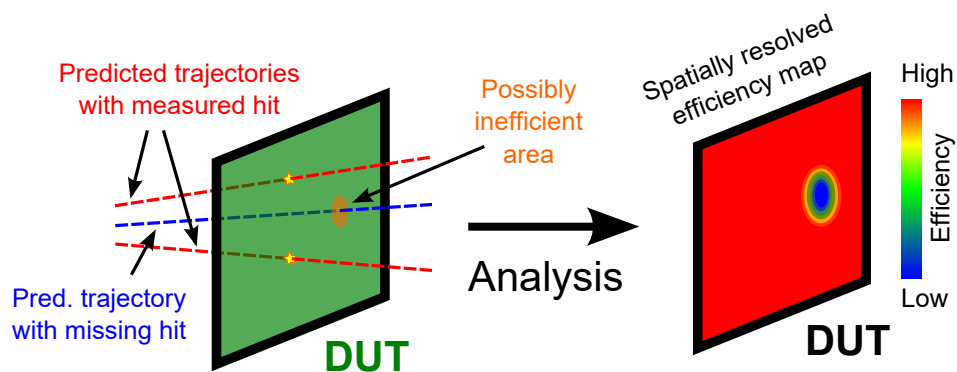

Figure 5.1.: Principle of a spatially resolved efficiency measurement carried out with test beam data. Trajectories are extrapolated onto the device under test (DUT). This information is used to investigate properties of the sensor at this point.

To perform such measurements, a source providing a beam of charged particles is necessary. Furthermore, a reference pixel detector set-up to obtain the trajectories of the individual particles is needed. Additional hardware to synchronise the different DAQ systems, the one for the modules to be tested and the one for the reference modules, is a central part of a test beam experiment.

5.1.1. The Telescope

A beam telescope is the reference system which is used to obtain particle trajectories. Several telescopes have been built and are still in use. Some are designed to provide very high spatial or temporal resolution, others to cope with a very high beam rate or to measure trajectories with as little material as possible to be also feasible in low energy applications. Examples include the Timepix telescope [84], the FE-I4 telescope [85] or the MuPix telescope [86]. While many techniques and results in this thesis apply to any telescope, the focus is set on the EUDET-type telescopes which use six Mimosa26 sensors [87].

The Mimosa26 is a monolithic active pixel sensor (MAPS), i.e. a CMOS sensor which features 576×1152 quadratic pixels with a pitch of 18.4 µm. Each pixel features correlated-double-sampling, every column has its own discriminator followed by zero-suppression circuitry. Hits are read out by a rolling-shutter architecture, providing binary hit information. The read-out time for a single frame on a Mimosa26 sensor is 115.2 µs.
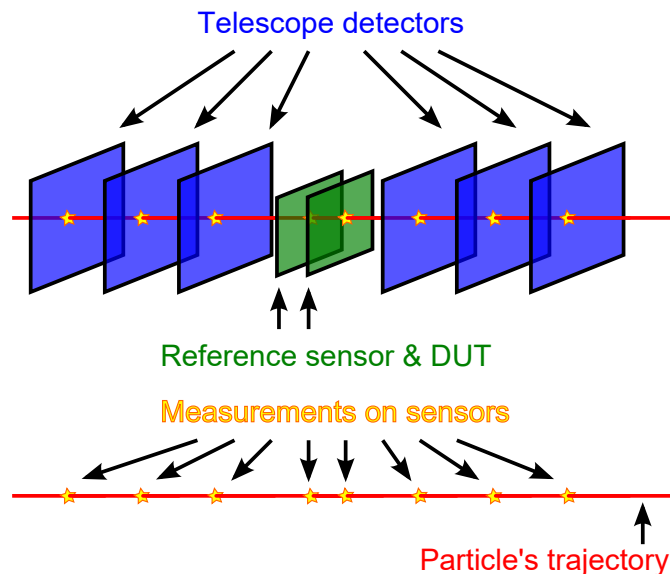

Figure 5.2.: A test beam set-up with six telescope sensors and two LHC-type devices (reference and DUT).

The sensor was designed to be operated at beam conditions with approximately 5 hits per image. This is, however, no limiting factor, and also beam rates with up to $\mathcal{O}(100)$ hits per image pose no problem. This is especially important as these sensors can suffer

from noise, especially at low thresholds and due to radiation effects, which can increase hit rates to these magnitudes.

The slow readout speed compared to the LHC devices, which operate at 40 MHz, is a limiting factor in data taking. Several approaches exist to speed up data acquisition, however all suffer from ambiguous synchronisation issues. Hence, a reference LHC-type device is used to determine whether the tested module was active, as the slow Mimosa26 modules will veto any further trigger in that time window and thus render the LHC-type devices inactive.

This second module is the so-called *reference sample*. The tested sample (or samples if multiple samples are used, given that the beam energy is sufficient), is placed within the telescope as well. This is the so-called *device under test*, short DUT. The reference sensor and DUT are typically placed in the centre of the telescope, in between the upstream- and downstream-arm of it. A schematic of a telescope with reference sensor and DUT is shown in Figure 5.2.

5.1.2. The Beam

The ATLAS ITk pixel upgrade studies are mostly carried out at two test beam facilities, namely, the CERN and the DESY research site. At CERN, mostly the SPS accelerator is used and a secondary beam is created by extracting the 400 GeV SPS beam[1] and guiding it onto a target.

In the H6 areas, the areas used for the ATLAS ITk pixel upgrade studies, only a secondary beam is available at energies up to 205 GeV. The particle type can be chosen, options are electrons, muons, and mixed hadrons (pions, protons, and kaons). In the last three years, in nearly all ITk related test beams, mixed hadrons at 120 GeV were used. An overview of the beam generation for the test beam areas at the CERN SPS is given in Figure 5.3.
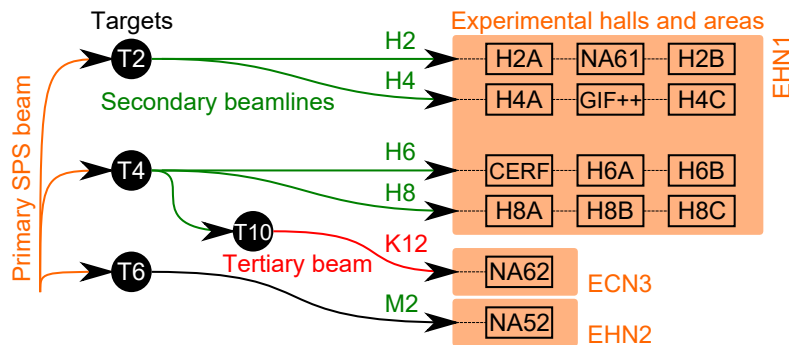

Figure 5.3.: Beam generation and beamlines at CERN's North Area experimental site.

---

[1]The beam energy for the extractions to the North Area of the SPS is reduced compared to LHC injection, which takes place at 450 GeV.

At the DESY site, the DESY II synchrotron is used to provide beam to the test beam areas. Again, not the primary beam, but a secondary beam is used. The difference to the SPS beam generation, however, is that not the DESY II beam is extracted and used for secondary beam generation, but a thin carbon fibre (25 µm diameter) is introduced into the beam, such that bremsstrahlung photons are created when bunches in the synchrotron hit the fibre. The bremsstrahlungs photons hit a conversion target after which polarity and momentum selection of the created electron-positron pairs takes place. A combination of magnets and collimators allows the users to select the desired beam properties. This *parasitic* usage mode results in a quasi-continuous[2] secondary beam, not only when the primary beam is extracted and shot onto target.
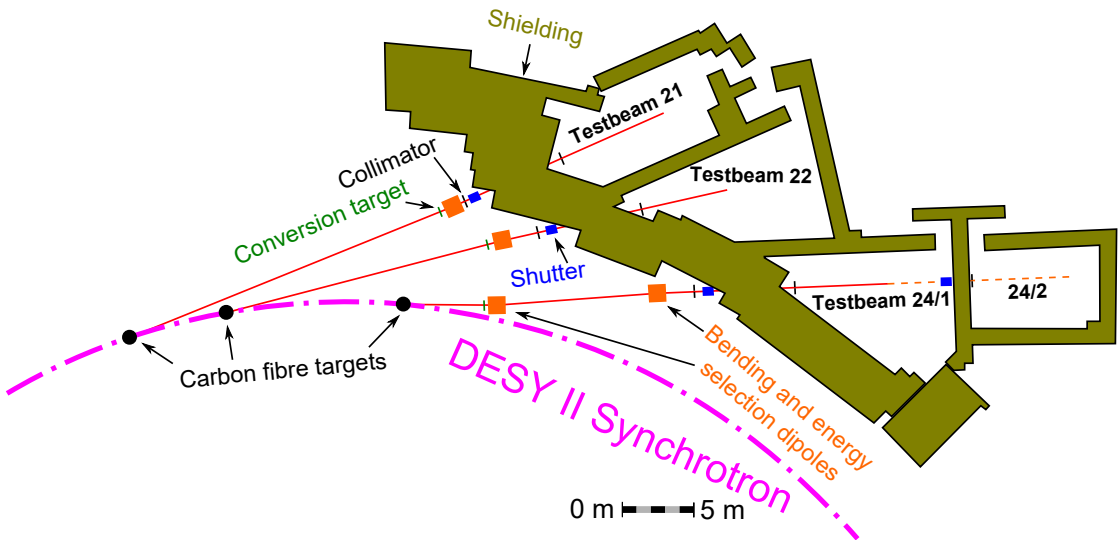

Figure 5.4.: Schematic of the DESY beam generation: the three test beam areas (21,22,24/1) which directly use the generated secondary beam as well as the *parasitic* area 24/2, which uses the remnant beam from 24/1, are shown.

The DESY II operates as the pre-accelerator to the PETRA III synchrotron light source at a beam energy of 6 GeV. Hence, this is the upper limit of the electron or positron beam in the test beam area. However, at these energies the particle rate is very low and typical beam energies at the DESY beamlines which are used for test beams range from 3 GeV to 5 GeV.

5.1.3. Triggering and Synchronisation

Contrary to the ATLAS detector at the LHC, there is no dedicated synchronisation of the accelerator and the telescope's or DUT's DAQ system. Hence, the particle passing through the sensor and the DAQ system's clock is uncorrelated, or correlated in an unknown way. To give an example, at the DESY II accelerator the beam is resynchronised

---

[2]A substructure due to the bunch structure and the revolution frequency is present.

with the power grid all the time, resulting in an unpredictable correlation. Therefore, additional synchronisation is needed in a test beam set-up.

The *trigger logic unit* (TLU) uses signals from photomultiplier tubes (PMTs) attached to scintillators to trigger and synchronise the attached DAQ systems [88]. For this purpose, scintillators are mounted on both sides of the telescope. The TLU allows to set a coincidence requirement to trigger a read-out.

Once the coincidence criterion is met, the TLU issues a trigger to all connected DAQ systems. In addition, the TLU can provide a trigger number to each trigger. The triggering scheme can be configured that DAQ systems can veto subsequent triggers for a given period of time. The introduced dead time is undesired, however necessary, if the DAQ system cannot process triggers while processing a prior event.

However, if a trigger is suppressed for the active time of a DAQ system, this also introduces an artefact due to a second system becoming inactive if no new triggers are issued. This was already mentioned in the context of the *reference sample* and is schematically shown in the Figure below:
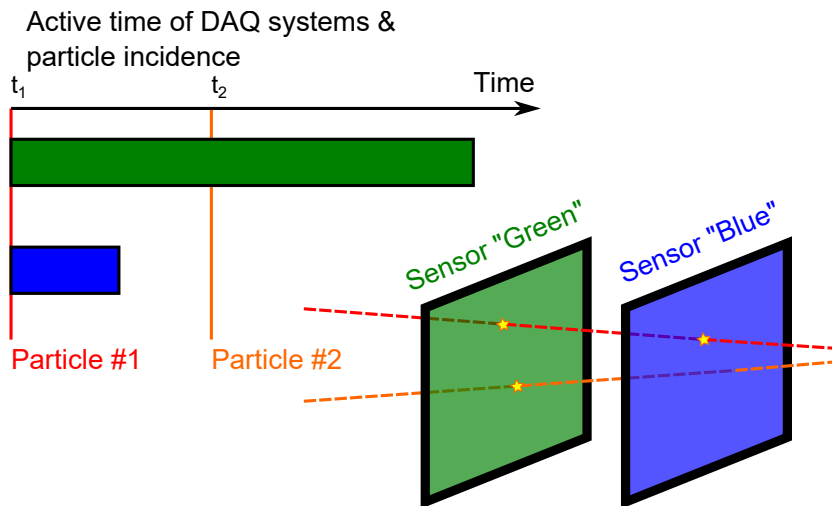

Figure 5.5.: Two DAQ systems with different active time. The green system will detect both hits, the blue one only the first particle.

In Figure 5.5 the first particle arrives at t_1. This triggers both the *green* and *blue* DAQ system via the TLU. Assuming they are fully efficient, both sensors will detect this hit. After some time, at t_2 a second particle passes the telescope. The *green* system is still active, while the *blue* one is not. The *green* system vetoes any subsequent triggers and hence no new trigger is issued. The *blue* sensor will therefore not detect the second particle[3].

---

[3]This is a simplified model, depending on the read-out mode this may be even more complicated. To give an example, in a rolling shutter read-out mode, the second hit might or might not be detected, depending on the hit and rolling shutter position.

5.2. Test Beam Software

Multiple software frameworks are utilised in a test beam experiment. The data from the telescope and custom DAQ systems need to be read out and stored. For the triggering, read-out, and synchronised storage of test beam data a framework exists, namely EU-DAQ. EUDAQ can be interfaced by the individual DAQ software. Furthermore, EUDAQ provides online monitoring of the data, allowing supervision of data taking.

After data collection, particle trajectories need to be reconstructed to provide tracks for analyses. This involves steps for data conversion, the clustering of hits and derivation of a hit position from clusters, alignment as well as the ultimate track finding and track fit.

5.2.1. Data Acquisition Software

The EUDAQ framework is used in the ATLAS test beam campaigns. It exploits Ethernet[4] for the communication of the different DAQ systems. As these are mostly, for example in case of the ATLAS Pixel read-out systems, running on a standard desktop PC, integration into an Ethernet network is easily possible. This disentangles the various components, as not the entirety of DAQ systems need to run on the same PC.

The main part of EUDAQ is the *RunControl* which coordinates the different parts of EUDAQ. With *RunControl*, the user can send configurations as well as start and stop commands to the other parts of EUDAQ. The *DataCollector* is the part which receives and collects the data from the different DAQ systems. The data are stored in EUDAQ packages, containing all raw data associated to the given trigger.

The triggers are issued by the TLU as discussed in the previous section. However, also a software executable, the *TLUProducer*, is running which allows to configure the TLU via *RunControl*. The *TLUProducer* also writes a package into the data stream, containing a precise timestamp and the current trigger number issued by the TLU for this event.

Every DAQ system has to implement a *Producer*, it is the part of EU-DAQ which allows any custom DAQ system to add packages into the data stream. The EUDAQ packages allow to store any binary data which is the recommended way of implementing the *Producer*. No interpretation
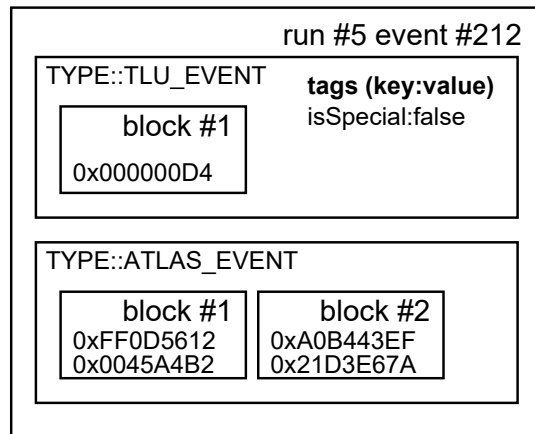

Figure 5.6.: Anatomy of an entire EUDAQ event.

---

[4]To be specific, the IEEE 802.3 protocol family.

of the data is intended as this poses a risk of data corruption and instead the raw data from the module should be sent to the *DataCollector*.

The structure of an EUDAQ event, as stored by the *DataCollector*, is shown in Figure 5.6. Every event has the run number and a corresponding event number attached to it. The actual data from the DAQ systems are packed into containers. In Figure 5.6 this is schematically shown for the case of a TLU and ATLAS container. Each of them carries a unique identifier (*TYPE::EVENT_TYPE*). The data are subdivided into blocks, i.e. further sub-containers. This allows to attach multiple blocks of data from a single DAQ system into the same container. The blocks are able to store an arbitrary amount of unsigned 32 bit numbers, the raw data.

An example use-case is when a single DAQ system controls and processes multiple modules. Depending on the implementation, there might be multiple, separate data-streams. Merging them into a single sub-container would on the one hand entangle this information and on the other hand pose a risk of data corruption. Hence, the mechanism of sub-containers circumvents this problem.

In addition to the raw data blocks, each container can be equipped with an arbitrary amount of tag-value pairs. This is especially important to flag special events or attach important information on the configuration of the system.

At the beginning and end of each run, a special event is attached. These are the beginning-of-run-event (BORE) and end-of-run-event (EORE), respectively. The BORE are used to store important information on the data taking configuration, i.e. they can be utilised to store parameters related to settings which are later needed to correctly interpret the data. This is important as such crucial information needs to be directly linked to the data file. For the same reason, the EUDAQ configuration files are also attached to the EUDAQ output files, allowing the user to validate the used settings in retrospect.

Two monitoring tools exist within EUDAQ. The *LogCollector* is used to display, save, and log messages issued by the other parts of the framework. Each part can issue notification with a given severity, e.g. log messages range from informal notifications of the start and stop of a run, up to severe warnings or error messages of producers which can warn the user that a module is malfunctioning.

The other important monitoring tool is the *OnlineMonitor*. As the name suggests, it is a tool which monitors the acquired data on-the-fly and displays plots of the collected data. Important examples are plots of the hitmap of a sensor. A hitmap is a plot of hit pixels in the space of the two-dimensional pixel matrix. Using the hitmap, noisy pixels, possibly due to mistuning, as well as the alignment of the sensor in the beam-line, can be validated. Furthermore, the *OnlineMonitor* can correlate pixel hits on the different sensors to each other. This yields the so-called correlation plots. Monitoring the correlations is crucial, as any de-synchronisation of the different DAQ systems can be immediately detected.

Examples of monitoring plots are given in Figure 5.7 where the two mentioned types of plots are shown. In Figure 5.7a a hitmap is shown. In this hitmap, there is a sudden increase of recorded hits between column 8 and 9. This is due to the scintillator used for

(a) Hitmap as provided by the *OnlineMonitor* to monitor data aquisition.

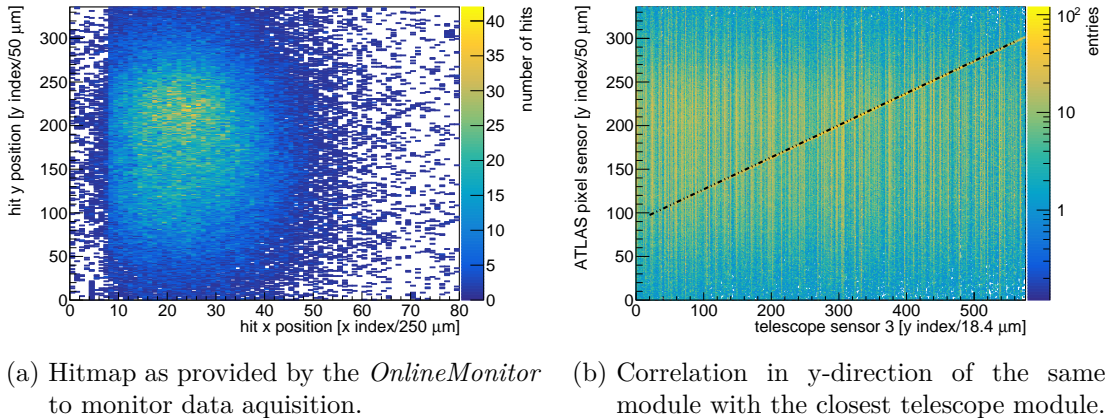(b) Correlation in y-direction of the same module with the closest telescope module.

Figure 5.7.: Two monitoring plots produced by EUDAQ during data taking. The line in the right plot has been added by hand to correlate the different pixel pitches of the Mimosa26 sensor and FE-I4 module.

triggering and the hits left to this line are due to multiple events for a single trigger[5]. The data displayed in Figure 5.7a can be correlated to the other modules in the telescope.

The correlation of this ATLAS FE-I4 module and the last upstream telescope module is shown in Figure 5.7b for the y-direction. No track information is used when correlating events on the different modules. Hence, noisy pixels will appear as lines in the correlation plots as a noisy pixel will provide a hit in multiple events and will be wrongly correlated to all hits on the other sensors. Noisy pixels on the ATLAS FE-I4 module would appear as horizontal lines, noisy pixels on the telescope as vertical ones. No noisy hits are expected on the ATLAS FE-I4, this can be seen in the hitmap shown in Figure 5.7a. However, as mentioned in Section 5.1.1, the Mimosa26 modules tend to be more noisy. The noise hits are uncorrelated with the actual pixel hit and appear as lines in the correlation histograms. Hence, the correlation plots are smeared by lines, as is shown in Figure 5.7b. Despite this, the correlation is clearly visible beneath the dotted line. This line with the slope of 50 µm/18.4 µm ≈ 2.72, corresponding to the ratio of the pixel's pitches, is indicated to confirm this claim[6].

In order to provide online monitoring of the data they must be interpreted. This is the *ConverterPlugin*s responsibility. The *ConverterPlugin* is the second part a user needs to implement for a custom DAQ system. The converter plugin takes the EUDAQ events and extracts raw hit information from them. In addition to the hit pixel indices, this may include charge as well as timestamp information, depending on the capabilities of the DAQ system and modules.

---

[5]This is a good example of how triggering impacts the observation. The beam itself has a Gaussian profile, however the selection (or sampling) of the beam imposes a structure. This is somewhat like the actual detector, where only a sub-sample of events are recorded. If one is not careful enough, this can have an undesired bias on the observation.

[6]This is no fit to data, but merely an overlaid line with manually adjusted offset.

(a) Timestamp versus charge correlation histograms for FE-I4 type devices.

(b) Average charge response of a FE-I4 device in units of ToT code.
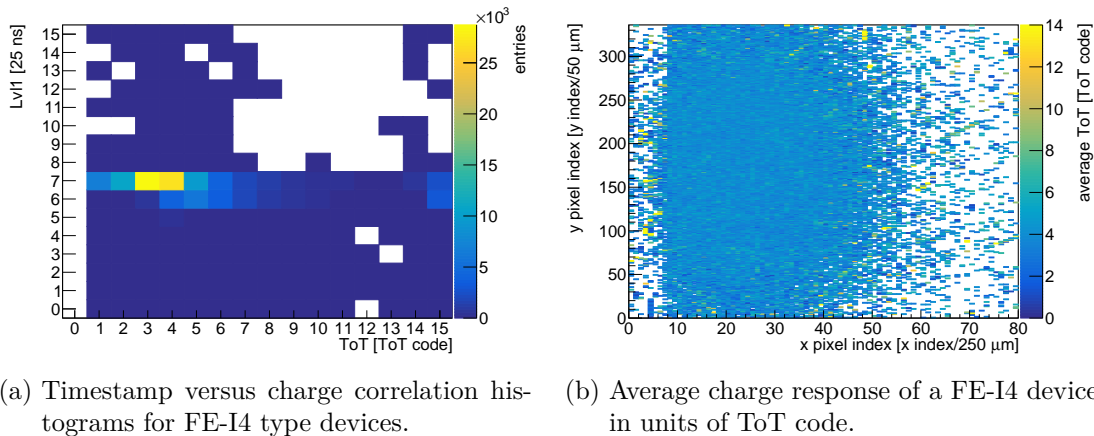
Figure 5.8.: Two further monitoring plots produced by EUDAQ during data acquisition, aimed at supervising the timing and charge tuning of a module.

As the FE-I4-type modules provide more information than the Mimosa26 modules, namely also charge and timing information, additional functionality has been introduced in the context of this thesis, allowing EUDAQ's *OnlineMonitor* to provide monitoring of these data. Two novel plots are now created and filled for FE-I4-type devices, namely, the correlation of the timing (the timing of the FE-I4 is the so-called LV1) and charge information (ToT) as well as a two-dimensional histogram showing the average ToT response. The first plot is useful to detect and monitor any time-walk effects of the FE whereas the latter can provide crucial monitoring of the tuning. Detecting a wrongly tuned sensor already at the stage of data acquisition allows to take corrective measures at an early stage. Furthermore, noisy pixels in most cases will show a deviating ToT response as well. Hence, the average ToT map will help to identify such pixels. An example of a Lvl1 and ToT correlation histogram is given in Figure 5.8a and an average ToT map in Figure 5.8b.

The *ConverterPlugin* is also responsible for conversion of the data into the so-called *LCIO* format, the Linear-Collider-Input-Output framework's file type. For the ATLAS ITk, this framework is used in the subsequent test beam reconstruction steps.

5.2.2. Reconstruction Software

After collecting the data, they must be processed and reconstructed. As mentioned in the previous Section, EUDAQ provides an interface to LCIO data. These data can be processed by *Marlin*, a modular framework for reconstruction and analysis of LCIO data.

Marlin

The concept behind Marlin is that a reconstruction or analysis can be disentangled into small and simple tasks. Each task is accomplished by a *Processor*, and each *Processor* operates on the event format governed by LCIO. The user can define which *Processor*s

are executed in a given order and with which parameters. To pass parameters and configure the execution order, a steering file is used. It is an XML-like file where single values (integer and floating point values or strings) or arrays of them are parsed.

To implement a *Processor*, certain callback functions need to be implemented, e.g. functions which are executed at the beginning or end of a run, or the function which is called when processing an event.
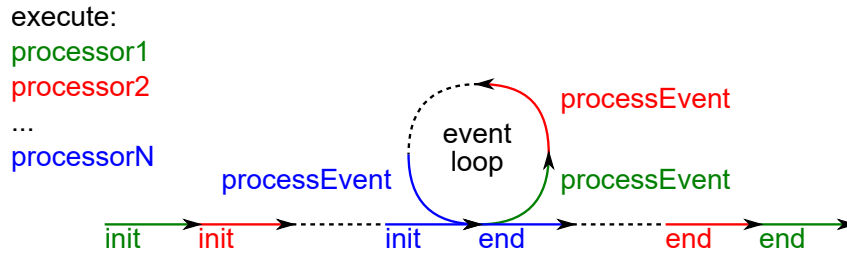

Figure 5.9.: Execution order of Marlin *Processors* and the event loop, processing the data.

A schematic working principle of a Marlin execution step is given in Figure 5.9. It is worth noting, that the finalising routine is called in reverse order. This makes sense in terms of resource allocation and deallocation. For example, if two processors interface an external framework, e.g. an alignment framework, the first processor will initialise and provide the interface. If this resource is deallocated before the second processor, which makes use of it, finalises its results, this may cause an exception or segmentation violation.

LCIO

The LCIO input/output file format's design impacts Marlin's working mechanism. The philosophy of Marlin is that it hides most of the I/O from the users which only need to implement processors. However, some knowledge of LCIO is needed to implement processors, as the data structures provided by LCIO are the ones the user will use in a Marlin processor.

LCIO is an event based format. This means that all information belonging together is put into the same event. The triggering and event definition of EUDAQ, as described in Section 5.2.1, enable to take a single EUDAQ event and to transform it into an LCIO event. However, the event definition is not always as simple and easy as that. Other DAQ systems provide triggerless read-out or allow multiple triggers while a different DAQ system is still busy. In these cases, LCIO is not suited to cope with these ambiguities, and either a proper event definition has to be achieved prior to Marlin or a different reconstruction framework has to be used.

An LCIO file stores collections. A collection is a container for a dedicated type for a given collection. The dedicated types are native LCIO objects which were designed to store data from a HEP detector or a Monte-Carlo framework. The objects used in test beam reconstruction are given in Table 5.1 with a short description what they are used

Type	Description
TrackerData	Raw data describing pixel hits (possibly multiple hits)
TrackerPulse	Used to interface raw data for a single cluster
TrackerHit	A spatial hit on a detector. Either from Monte-Carlo or from cluster
Track	A track through the telescope from a particle or from Monte-Carlo

Table 5.1.: LCIO types used in collections exploited during test beam reconstruction and analysis.

for. Certain objects can link to other objects, also in different collections, in the same event. The structure of a possible LCIO event is depicted in Figure 5.10.
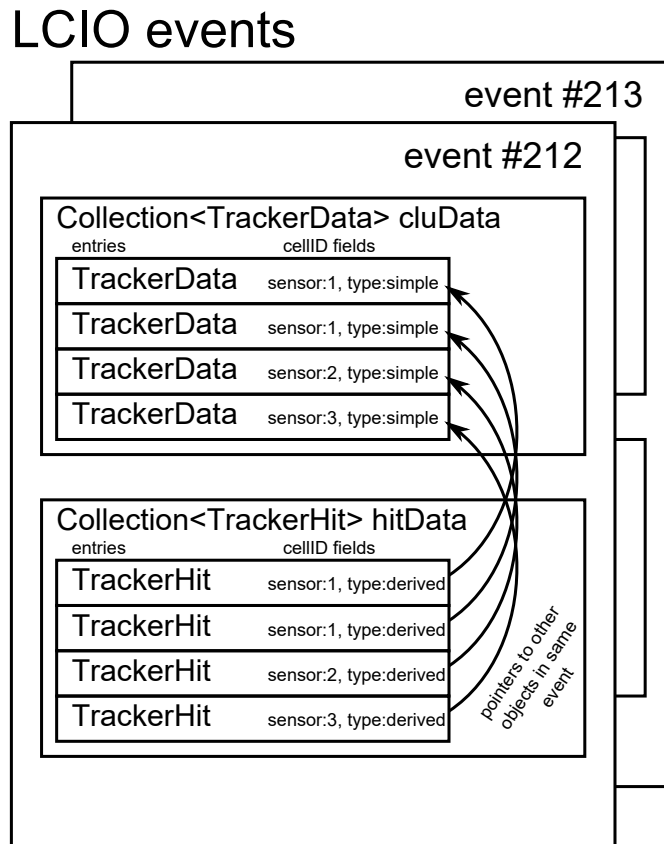

Figure 5.10.: An LCIO event with two collections, one for storing cluster data and one holding the derived hit positions. The entries in the collection of hits point back to their original raw data they were derived from.

The *TrackerData* type is used to store raw data of pixel sensors. Contrary to one's intuition, a single TrackerData instance can (and in general will) hold multiple detected hits in a detector. For this purpose a *TrackerData* object has a container to hold an arbitrary amount of floating point numbers. These can be pixel indices, but also charge

information or timestamps. It is up to the framework atop of Marlin to make sense of the data stored.

A *TrackerPulse* is used to mask a *TrackerData* entry as a cluster. This illustrates how a single *TrackerData* entry can be used to describe different things, it can hold all hits on a sensor for an event, or merely the ones belonging to a cluster. In addition to pointing to the raw data, a *TrackerPulse* can also store information on the cluster's quality and charge.

Spatial hit positions are stored in *TrackerHit* objects. In addition to a three dimensional point, again a link to the original raw data is present. No link to the cluster, i.e. the *TrackerPulse*, is kept, however. Furthermore, the covariance matrix for this measurement is stored in a *TrackerHit*. Again, the quality of the hit can be stored as well.

Finally a *Track* is the object used to store a particle's trajectory. A track in LCIO can be parametrised by five track parameters. The used track model is summarised in [89]. This is one drawback of using LCIO for test beam reconstruction, as the track model imposed by the framework is more complex than necessary. Fortunately however, the *Track* allows to attach a link to an arbitrary amount of *TrackerHit* objects. Hence, a Marlin processor can exploit its own track parametrisation and store the resulting track by passing the predicted impact positions into a *Track* object. Furthermore, this also allows to describe trajectories which are *broken*, a term used for trajectories which take multiple scattering into account, and not only straight lines or helices.

Every object in an LCIO collection carries a 64 bit field. In this field, additional, user-defined, information can be stored. This makes the LCIO objects flexible, however, also imposes additional overhead.

The LCIO framework also allows to store relations between objects. For this purpose an object to store such relations exists, the *LCRelation*. These relations themselves are stored in a collection, hence mimic a relational-database structure. However, exploiting these relations is a tedious task if only accomplished with the type introspection of C++ and without any reflection and thus not extensively used in the test beam software frameworks.

GEAR

The GEAR framework is used by Marlin to specify the experiment's set-up. In the case of a test beam experiment, this is the arrangement of the telescope and user's samples. The position and their rotation has to be specified. Furthermore, GEAR is also used to describe the pixel matrix of a module, i.e. the dimensions and pitches of the pixels.

In the actual test beam software package used by the ATLAS collaboration's ITk pixel upgrade community, an extended framework atop of the mere GEAR description exists. This is necessary, as in the R&D process, several new pixel layouts have been implemented and used, deviating from a simple chequered pixel matrix.

Furthermore, a new syntax for the GEAR file has been introduced in the context of this thesis. The new design aims at being easier to use, reducing the amount of ambiguous values and providing more functionality at the same time. For example, it is possible to

introduce layers of *dead* (passive) material. Furthermore, it allows to position passive structures relative to a given sensor, i.e. allows simple modelling of a read-out chip or support structure.

A comparison of the two types of GEAR files is given in Appendix B where the features and changes are outlined in more detail.

EUTelescope

EUTelescope is a collection of Marlin processors for the reconstruction of test beam data. Aside from the packages provided by Marlin, i.e. LCIO and GEAR, further external libraries are used, namely, Eigen3 is used as a linear algebra library, the MillepedeII framework for alignment and the GBL framework for alignment (together with MillepedeII) and for track fitting.

The interface between EUTelescope and EUDAQ is via the so-called *NativeReader* which is a Marlin processor exploiting EUDAQ's *ConverterPlugins* to write LCIO files. Hence, this processor is executed first, providing the desired LCIO collections for the subsequent processors.

Reconstruction can be divided into several steps: data conversion, noise processing, clustering, hit position derivation, alignment, track finding and ultimately track fitting. While several steps can be merged into a single reconstruction step, it is good to split up the tasks and process them consecutively, as this allows to monitor and validate the results between steps.

The data conversion and noise determination is typically done in a single step. In order to determine the occupancy fraction of a pixel, it is necessary to process the entire (or at least a significant subset) of a run, hence it is a natural choice to end the first reconstruction step after this processor.

The next steps involve the determination of clusters and noise treatment. Noise can be treated on the pixel or cluster level, i.e. either noisy hits are removed prior to clustering or clusters containing a noisy hit are removed after clustering. Once this is accomplished, a hit position is derived from the clusters. Different algorithms for this are implemented in EUTelescope. Once a spatial position is obtained, the hit is transformed into the telescope's global coordinate system using the GEAR position values, and correlation histograms as well as pre-alignment values are derived. Pre-alignment values are stored by updating the GEAR file.

The correlation histograms and pre-alignment value determination work without any tracking. They simply assume no beam divergence and no scattering, i.e. a straight line from a hit on the first plane to any subsequent one. For every hit, the straight trajectory is extrapolated to all downstream planes and any hit in proximity to the extrapolated position on that plane is correlated to the one on the initial plane. This way, shifts in both x- and y-direction can be corrected for.

While in principle these tasks can be processed in a single step, often the reconstruction is split into the clustering and hit position derivation step. This has the advantage that the cluster histograms can be inspected prior to the next step.
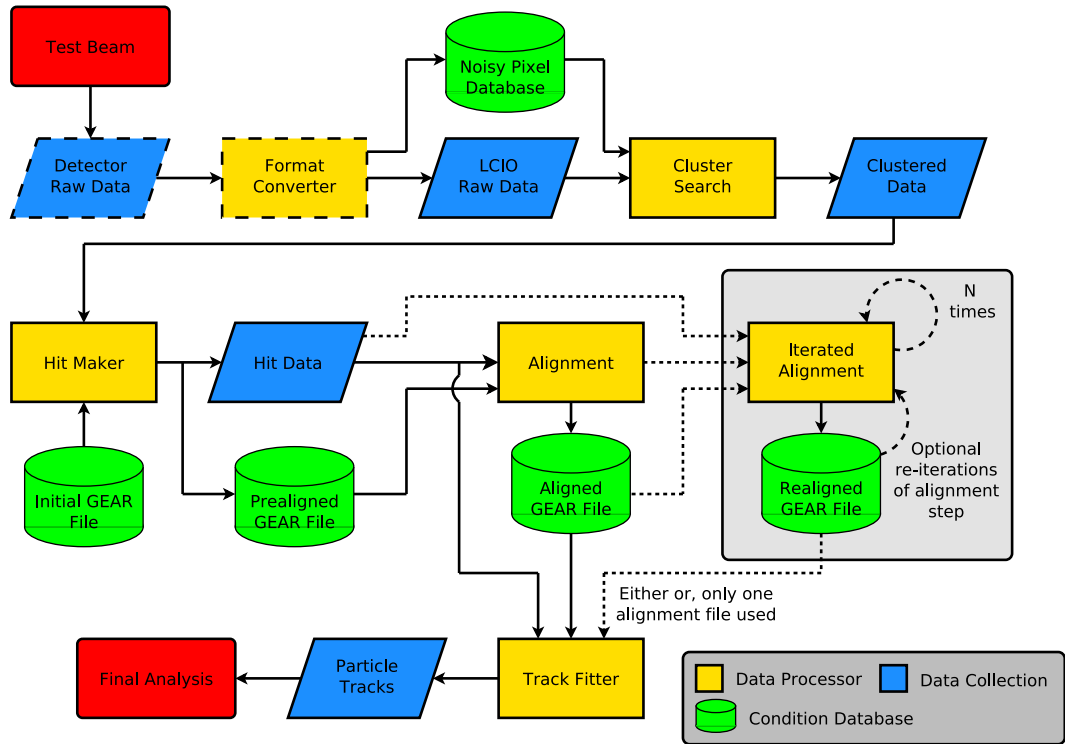
Figure 5.11.: A default test beam reconstruction chain with EUTelescope. Indicated
is the possible iterative alignment procedure.

The next step is the alignment step. The previously obtained hit positions and pre-alignment values are used to perform a track fit with the coarsely aligned data. No new LCIO data file is created in this step, as only the alignment values are updated. Again, these values are written into an updated GEAR file. The alignment step can be repeated for iterative alignment.

Ultimately the final track fit is performed. The most recent aligned GEAR file is used for track finding and fitting. The tracks are extrapolated to all detector planes and this data is stored in the final LCIO file. Furthermore, processors to export the data into custom analysis files (e.g. into ROOT tuples) can be called.

The entire reconstruction chain is shown in Figure 5.11. As discussed, certain steps may be split up or even merged. However, Fig 5.11 shows a baseline for the most common reconstructions in today's test beam community.

USBpix

6.1. Introduction

The USBpix DAQ system is a small scale read-out system for FE-I3, FE-I4 and future RD-53(A) modules [90]. Compared to the ATLAS detector's DAQ hardware, the USBpix boards can only process data from a limited number of FE's, depending on the actual FE, in some cases even only at reduced data throughputs. For test beams, this holds no limitation, as other parts of the experimental set-up and analysis frameworks pose stronger bottlenecks, making the cheap, small, and versatile USBpix systems ideal DAQ systems in test beam experiments.

6.1.1. The DAQ Boards

Two generations of USBpix boards are in circulation, the second and third generation boards. The most significant difference between them is the used field-programmable gate array (FPGA). The third generation boards come in two flavours. One of the USBpix boards of the third generation, namely the MultiIO, is very similar to the second generation board, as both boards need attachment cards which feature regulators required to communicate with any attached FE or FEs. Multiple adapter cards exist, featuring up to four channels to communicate with attached FEs. MultiIO (MIO) boards exist for the second and third generation, i.e. the MIO2 and MIO3 boards.

The third generation also features the MMC3 board. This is a single board which already houses the adapters and regulators needed to communicate with up to eight attached FEs via RJ45 connectors. As an RJ45 connector only features eight connections, only four differential lines are available. This is enough to transmit the clock and receive

as well as transmit data[1]. Most FEs allow for more connections to investigate and probe properties of the chip, e.g. injection lines or monitoring capabilities of test pixels. If this functionality is desired, MIO boards with adapter boards for more connections need to be used. For test beam the MMC3 boards provide all functionality needed.

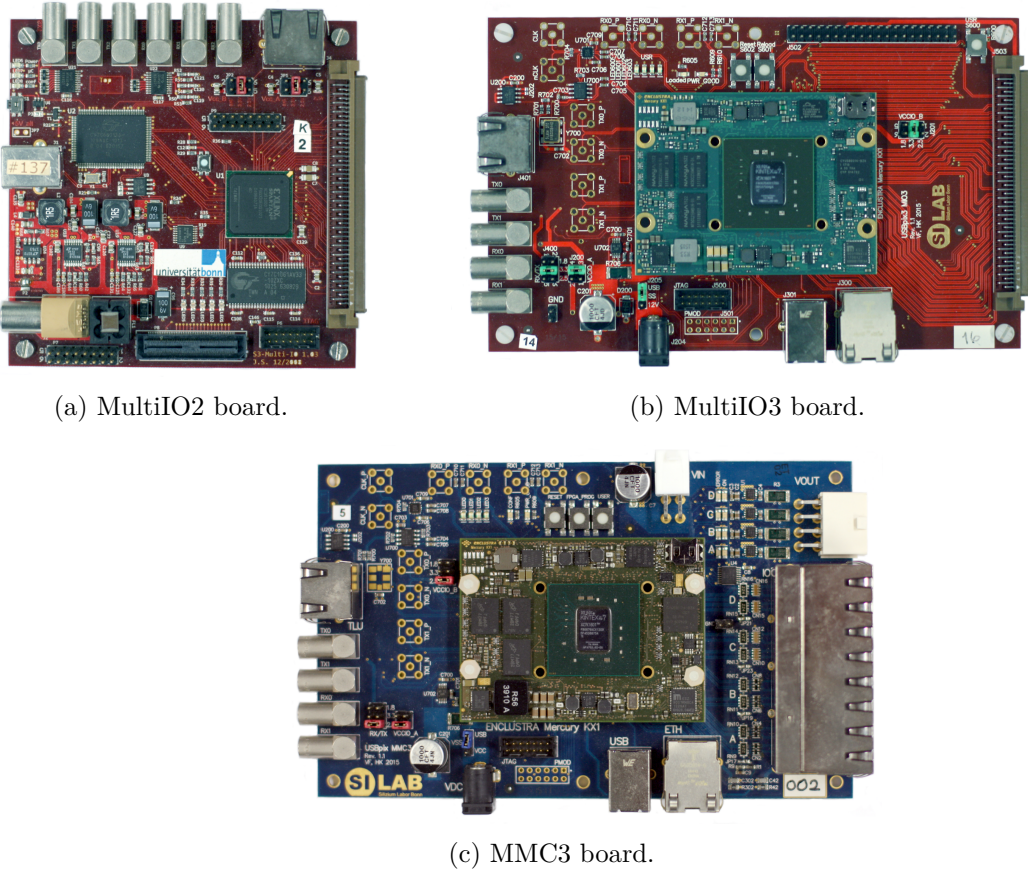(a) MultiIO2 board.

(b) MultiIO3 board.

(c) MMC3 board.

Figure 6.1.: Overview of the currently used USBpix hardware boards of the second and third generation.

An overview of the USBpix boards is shown in Figure 6.1. Each board has an additional RJ45 connector for communication with the TLU. Communication with the DAQ PC is done via USB. The second generation boards comply to the USB 2.0 standard, whereas the third generation supports USB 3.0. As USB 3.0 is limited to 5 Gbit/s, this would just be slightly below the bandwidth of a single FE for the ITk upgrade. The third generation boards also feature a gigabit ethernet interface. This interface is currently being implemented in the DAQ software. As mentioned, this is no limiting factor for test beam experiments.

---

[1]The newest firmware will allow to use one RJ45 connector to transmit one clock and command line each to two FEs, each with their own dedicated data line, increasing the total number of possible attached FEs to 16.

6.1.2. The STcontrol Software

Data acquisition with the USBpix board is possible with the STcontrol software, which is a possible software front-end to the USBpix hardware. STcontrol offers full EUDAQ integration, which has been completely reimplemented in the context of this thesis. The new implementation has been used extensively during the last one and a half years at several test beams by the ATLAS community. The integration and interplay of the STcontrol software, the read-out chips and the USBpix board, as well as the EUDAQ framework is depicted in Figure 6.2.
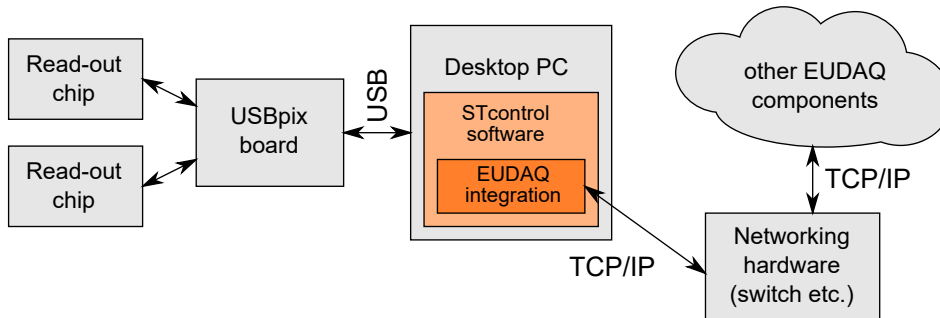

Figure 6.2.: Schematic of STcontrol with the EUDAQ integration, communicating with the read-out chips via the USBpix board and the EUDAQ framework via TCP/IP.

STcontrol makes use of the abstract interface provided by PixLib, an abstraction layer similar to that used in the current ATLAS Pixel DAQ software. This at least holds true for all the FEs up to, and including, the FE-I4[2]. The actual hardware abstraction is done via implementation of a *PixController*. A PixController can be a piece of hardware in the actual detector's DAQ system or a USBpix board for example. To the software both systems will look and act the same. The version of PixLib used in STcontrol has slight changes from the one used in the ATLAS Pixel DAQ software. This also enabled modification of PixLib components for the novel EUDAQ integration. As most parts of PixLib are compatible, this allows to implement and test new scans for both: the STcontrol software and attached modules to the USBpix board, as well as modules in the ATLAS detector.

6.2. Novel EUDAQ Integration

With the new generation of boards, there was the need to implement them into EUDAQ. In the process of this, the old implementation which heavily relied on the signal and slot mechanism of Qt[3], was replaced as well. This allowed to easily implement an observer pattern design for the producer. However, previous experience showed that depending

---

[2]For the ITk upgrade a common, new DAQ software is foreseen.
[3]A free and open-source cross-platform software framework for graphical user applications

on system load, this communication can be slow and in order to guarantee no data corruption, the USBpix board vetoed data acquisition, i.e. it halted TLU triggers, when processing data. This introduced unnecessary dead time.

In order to circumvent this, a multi-threaded approach was chosen to separate data acquisition and data processing. Contrary to executing a callback when data are ready, a buffer is monitored and processed as soon as it is filled. This reduces the required thread-safe communication and is therefore less error prone. For this, the functionality of the PixController has been updated so that each PixController in STcontrol partially owns a collection of buffers. The buffers are implemented as circular queues (FIFO). This is an important fact to keep in mind, as such data structures cannot grow dynamically, however they have a defined memory location which can be allocated on the stack and are hence faster than dynamically managed data structures. The lack of dynamic growth can lead to buffer overflows which results in data corruption.

It is vital that these buffers are thread-safe. This is required as the buffers are populated by the thread of the PixController, but processed by the thread the EUDAQ Producer executes in. An implementation which guarantees wait-freedom is used. A wait-free algorithm has guaranteed system-wide throughput with additional guarantee that every thread makes progress (i.e. starvation-freedom). Per definition, wait-free algorithms are lock-free. Locks (e.g. realised by mutexes, semaphores or similar) are in general undesired as they impose synchronisation at the cost of halting individual threads. This not only reduces the throughput, i.e. is slower, but also poses the risk of a deadlock.

Contrary to the prior implementation, the new one is asynchronous, meaning that there is no sequential *copying data from the USBpix board* and once this is finished *processing and sending data to EUDAQ.* Instead, these tasks are executed simultaneously. There is no mechanism to synchronise this. The thread processing and sending data to EUDAQ might execute faster, reaching the state where all available data are processed. In this case, the assumption that no more data are present is erroneous. A context switch can easily halt copying data into the buffer, even mid single value (given that the values are of non-atomic types[4]). The buffer's implementation ensures that such values are not considered to be valid (yet), and in the same fashion, that only halfway processed entries are not considered deleted (yet).

A further requirement for the new EUDAQ implementation was that multiple boards, also of different generations, could be used conjoined. This ensures that the old boards are still fully supported and used. Initially the third generation boards had only limited availability. In order to make users operate with the updated software, it was necessary to provide full support of all available hardware.

From the data processing point of view, the categorisation into second and third generation is sufficient. The attached adapter card (or integrated controllers) is hidden for the processing software.

---

[4]An atomic type is a type free of data races. This is achieved with atomic operation, i.e. operations which are guaranteed by the system to be uninterruptible.

Each EUDAQ Producer (cf. Section 5.2.1) inherits functionality from two base classes, the *CommandReceiver* and the *DataSender*. The initial class is responsible for communication with RunControl in EUDAQ via callbacks for certain events in RunControl, e.g. for initialising or starting and stopping runs. The latter part is responsible for transferring data to a DataCollector. The limitation of this model is, that there is only a single instance of STcontrol running for multiple USBpix boards, which are standalone DAQ systems without the need for software assisted synchronisation. Synchronisation is obtained via the TLU which directly triggers the USBpix board. However, configuration of the USBpix boards is done by the single STcontrol instance.

To circumvent this design limitation, EUDAQ's concept of a Producer is relaxed. Each instance of STcontrol will implement a class which merely inherits from the *CommandReceiver*, this is the implementation of the STcontrol's Producer. Upon loading a configuration in STcontrol, all the attached boards are known to STcontrol. Each physical board will be reflected by a PixController and each PixController will have a collection of buffers. The buffers are initialised at creation of the PixControllers. Loading a configuration in STcontrol is performed in the initialisation step in EUDAQ. Note that merely the configuration is loaded, the modules are not configured in this process.

In the next step, the configuration step in EUDAQ, two tasks are performed. The so-called DataSenders are created by the STcontrol Producer and the modules are configured. The DataSenders are EUDAQ Producers. One DataSender per USBpix board is created and they connect to RunControl as any other EUDAQ Producer. In addition, they also get a pointer to the PixController's buffers. This way, each DataSender is linked to a different PixController.

When a run is started or stopped, both the DataSenders and STcontrol Producer are informed by EUDAQ. The STcontrol Producer will put all boards into data acquisition mode by executing the corresponding scan[5]. The DataSenders will start monitoring their associated buffers and send data to the DataCollector. When a run is stopped, the scan is stopped by the STcontrol Producer and the DataSenders will stop monitoring the buffers after some delay, ensuring that any remaining data are correctly processed.

This schematic layout is depicted in Figure 6.3, where an exemplary use-case with two attached boards, one second and one third generation. The STcontrol Producer is responsible creating the DataSenders, however no backwards communication between these entities is present. All communication is done by EUDAQ. The FIFOs in the PixControllers work entirely independent of EUDAQ.

In Figure 6.3 the major difference between the second and third generation boards is depicted as well. The second generation has multiple FIFOs, whereas the third generation merely has a single one for each board. To be more precise, the second generation has a FIFO for every attached read-out channel. This may range from a single one, in the case of the most simple adapter cards attached to the MultiIO2 board, up to four for the so-called *BurnIn-Card* (BIC) which provides four read-out channels. The reason for this is the way the firmware arranges and provides data to STcontrol. In the second

---

[5]STControl will not *scan* over parameters, however it uses the implemented scan mechanism to acquire data by putting the FE into the proper mode.
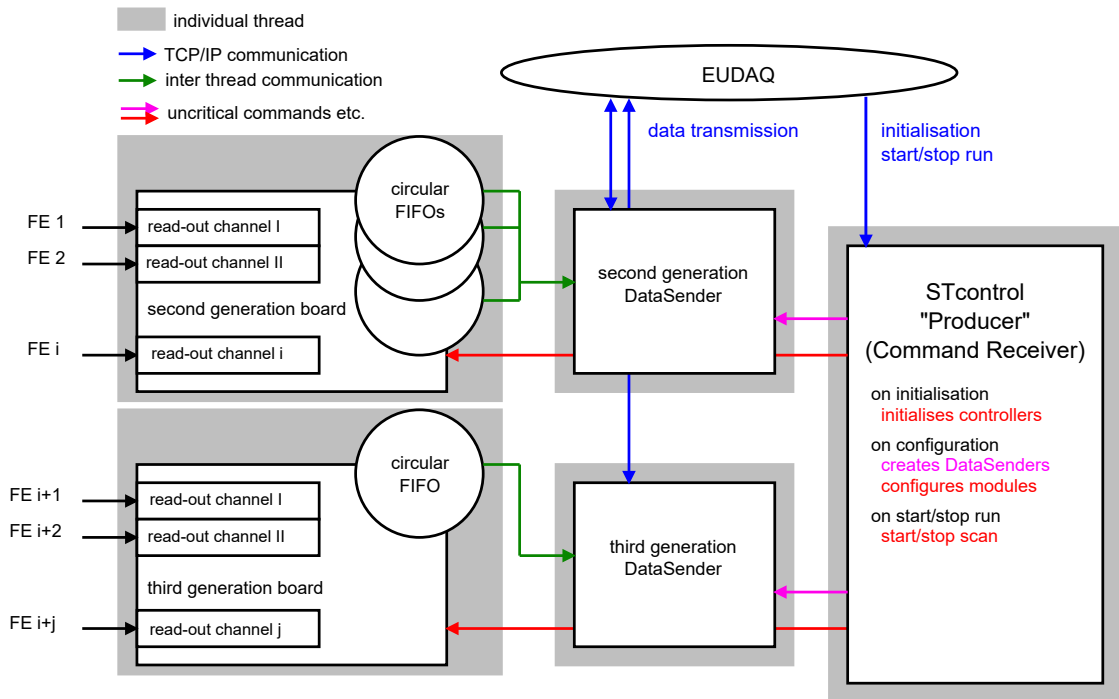
Figure 6.3.: The case of two USBpix boards attached to a single instance of STcontrol. Communication between the different parts of the Producer and DataSenders is shown. Highlighted are the sections which are executed in individual threads.

generation boards, memory is divided up into equally sized cells, one for each read-out channel. In each memory cell, the FPGA's firmware writes all the data from the corresponding channel, as well as trigger words, which are merged into the data stream by the FPGA. This is shown in Figure 6.4 on the left. Each block can be considered a standalone data stream with all relevant information.

Contrary to this layout, the third generation data stream is different. Even for multiple read-out channels, there is only a single data stream, depicted in Figure 6.4 on the right. Trigger words are only inserted once into the entire data stream.

Each data stream is written into a different FIFO by the PixController, giving the second generation boards feature multiple buffers. As discussed in Section 5.2.1, all information belonging to the same event is packed into an event before being sent by the DataSender. In the case of the second generation, this is slightly more complicated, as the data from all buffers, i.e. all read-out channels, need to be processed and packed into the same event. Additional synchronisation is implemented to ensure that all data belonging to the event are processed. In the case of the third generation, once the next trigger word in the data stream appears, there is the guarantee that all data have arrived and the EUDAQ event is sent to the DataCollector.

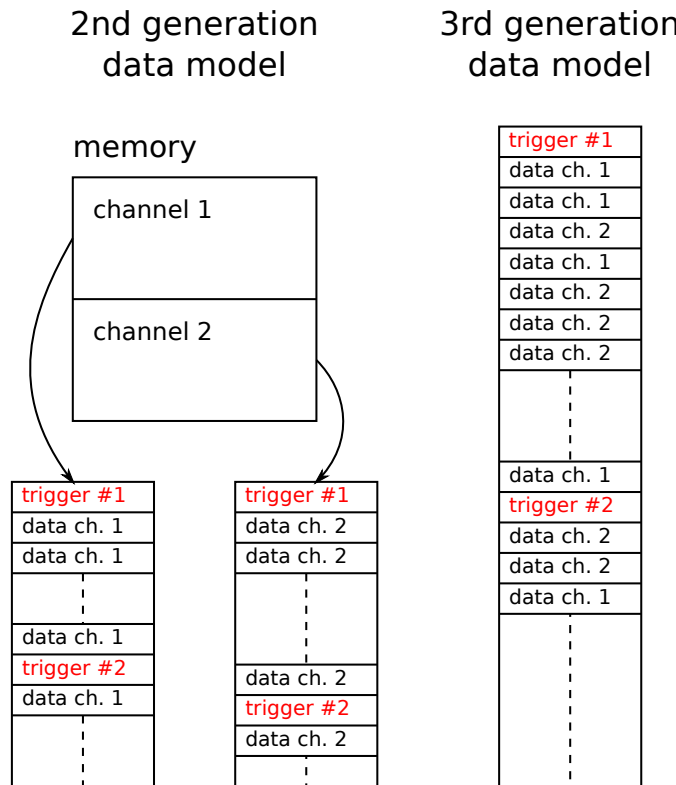On the EUDAQ side, the ConverterPlugin for both generations was implemented. A

Figure 6.4.: Physical memory layout and data-stream management of the second and third generation boards.

type safe decoding class to interpret the data was implemented.

For the third generation implementation, the data words from the FE-I4 are 24 bit wide, while in the software 32 bit data types are used. The remaining 8 bit are used by the FPGA to encode the read-out channel. This way, the association between channel and data word is established. In the case of the second generation, the data are stored in different containers in the EUDAQ event, this way no additional association between read-out channel and data word is necessary.
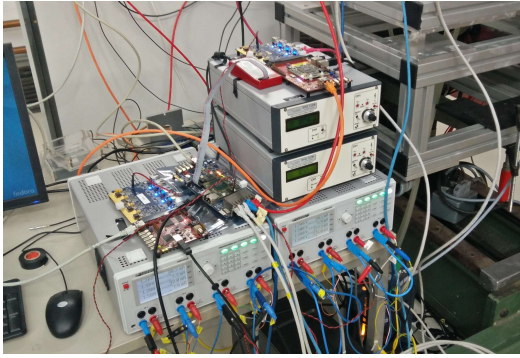
6.2.1. Validation of the EUDAQ Producer

In order to test the functionality of the new implementations, in particular the new EUDAQ scheme with a relaxed Producer design, a mock-up DataSender for the third generation boards has been implemented in STcontrol. This enables execution of scans without an actual operational sensor to be attached to the USBpix board. The actual physical board still needs to be present, as it will process the TLU's triggers and provide a data stream.

The mock-up data sender mimics data and sends events to the DataCollector. This way, the communication between EUDAQ and STcontrol, as well as the basic functioning of the ConverterPlugin, can be validated.

While these are tests where the basic functionality can be investigated, synchronisation between multiple DAQ systems needs to be ensured. For this purpose, a dedicated test beam to validate and test the new STcontrol EUDAQ integration was carried out.

(a) DAQ boards used in the validation of the new Producers.
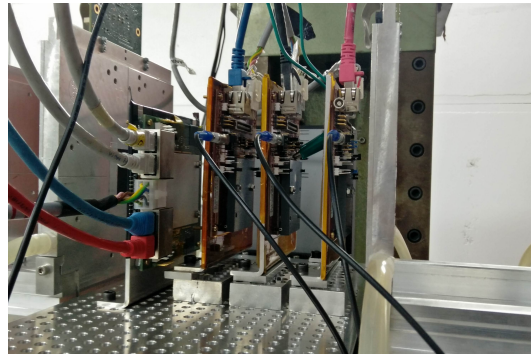
(b) FEs (FE-I4) used as a detector in the test set-up.

Figure 6.5.: Overview of the currently used USBpix hardware boards of the second and third generation.

In Figure 6.5 two photos of the set-up are shown. The three different DAQ boards in Figure 6.5a, namely the MIO2 board with a BIC adapter card (leftmost), the MMC3 (centre), and the MIO3 with a BIC adapter card (right) read out a total of 7 FEs simultaneously and with the same instance of STcontrol. A four-chip module (leftmost DUT in Fig. 6.5b) was attached to the MMC3 board, two single chip modules to the MIO2 board and a single chip module to the MIO3.

The configuration mechanism as well as test beam related parameters, used by the STControl software to correctly trigger the FEs, were validated. For example, a *delay* parameter is used in test beam mode to tell the DAQ system with what delay it has to trigger. This is necessary as the triggering of the FE has to be correctly timed with respect to the TLU issued trigger. Shifting this parameter will move the LV1 distributions accordingly, and hence this functionality was tested with real data and a reference telescope.

Finally, the long term stability in the sense of synchronisation, of the USBpix DAQ systems with respect to the telescope was tested. All of the investigated functionality worked as expected and the updated version of STcontrol was shipped and successfully used by the ATLAS pixel community at an ITk test beam just a few weeks later.

EUTelescope

7.1. Reconstruction Chain

7.1.1. Converter and Noise Determination Step

The initial step in data reconstruction is data interpretation. Strictly speaking, the executed code, i.e. the Marlin *NativeReader* processor, is located in the EUDAQ software repository, as it uses EUDAQ's ConverterPlugin. In that sense, this section is not part of EUTelescope. However, in order to write proper, i.e. correctly encoded LCIO TrackerData collections, EUTelescope routines are used, in particular the newly implemented interface between user code and raw pixel data, which has been entirely rewritten in the context of this thesis.
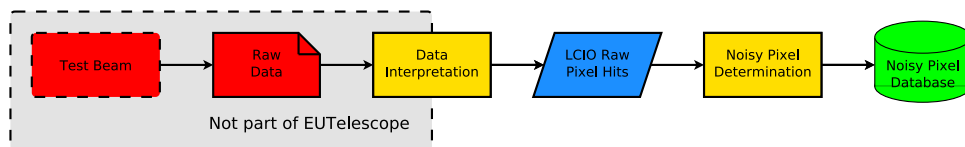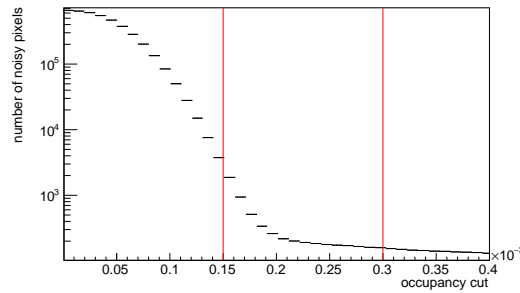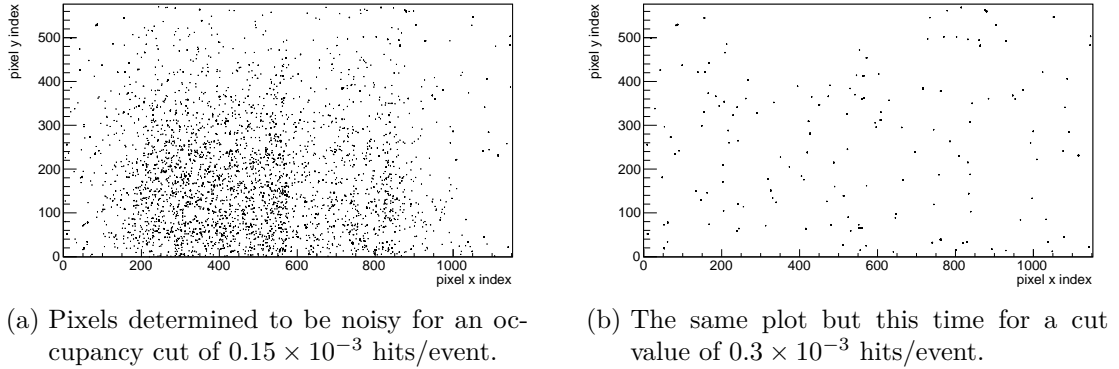

Figure 7.1.: The data conversion step in test beam reconstruction.

This ambiguity is also reflected in Figure 7.1 where the data interpretation step is only partly considered to belong to EUTelescope. The output of the *NativeReader* is an LCIO file with a collection of TrackerData entries. In the EUTelescope framework there is the convention that the *NativeReader* has to write one TrackerData entry per sensor module, i.e. all the hits detected by this sensor in this event need to be written into one TrackerData entry. Any sensor which detected no hits has to write a TrackerData entry without any hits as well. This is necessary as other parts of the framework rely on the existence of an entry in this initial collection. Violating this convention can

lead to undefined behaviour, and more important, to random seeming runtime faults only appearing when a sensor does not detect a hit in the initial event. The name of the collection created and populated by the *NativeReader* is hard coded by most implementations of a ConverterPlugin.

As the created collections contain raw pixel hits, processable by EUTelescope, occupancy maps can be filled. Occupancy is the number of events in which a hit was detected divided by the total number of events[1].

(a) Pixels determined to be noisy for an occupancy cut of 0.15×10^{-3} hits/event.

(b) The same plot but this time for a cut value of 0.3×10^{-3} hits/event.

(c) The amount of pixels considered to be noisy versus the occupancy cut value. At very low values, pixels with a non-zero occupancy due to beam hits are wrongly considered noisy. The cut values used in (a) and (b) are indicated by vertical lines.

Figure 7.2.: Determination of a proper occupancy threshold for noise treatment at a test beam experiment.

The *NoisyPixelFinder* Processor uses these occupancy values to determine a database containing pixels considered as noise, referred to as *noisy pixels* henceforth. Pixels above a certain occupancy threshold are considered noisy. The occupancy threshold needs to be adjusted to the test beam environment, making it a user-adjustable parameter. Beam properties like rate, intensity but also shape require different occupancy cut values.

---

[1] The occupancy may be larger than one in certain cases. For example the Mimosa26's rolling shutter read-out process more than one frame in a single event, i.e. they can 'detect' more than a single hit in a pixel per event.

Furthermore, the pixel size and active time of the used modules play an important role to determine a proper cut value.

As a rule-of-thumb, a low occupancy cut value should be chosen initially. The *Noisy-PixelFinder* produces two-dimensional histograms indicating the pixels considered to be noisy. Running on a small sub-sample of the events, the occupancy cut is increased until the beam spot, which is falsely considered to be *noisy* if the cut value is too low, vanishes in this map. Furthermore, a one-dimensional scatter plot showing the number of pixels considered to be noisy versus the occupancy cut value has been added. This helps to correctly identify a suited occupancy cut value. The impact of a too low occupancy cut versus a properly chosen one is shown in Figure 7.2. Also in Figure 7.2c the newly introduced scatter plot aiding users to select a proper occupancy threshold is shown.

The noisy pixel database is stored as an LCIO file, with a single TrackerData collection. Again, each detector module is reflected by a single entry which contains the entirety of noise pixels on that sensor. As these data are stored in the first event, this explains why some processors are only initialised during the first event.

7.1.2. Clustering and Noise Treatment Step

(a) Approach to clustering with noisy cluster masking and removing.

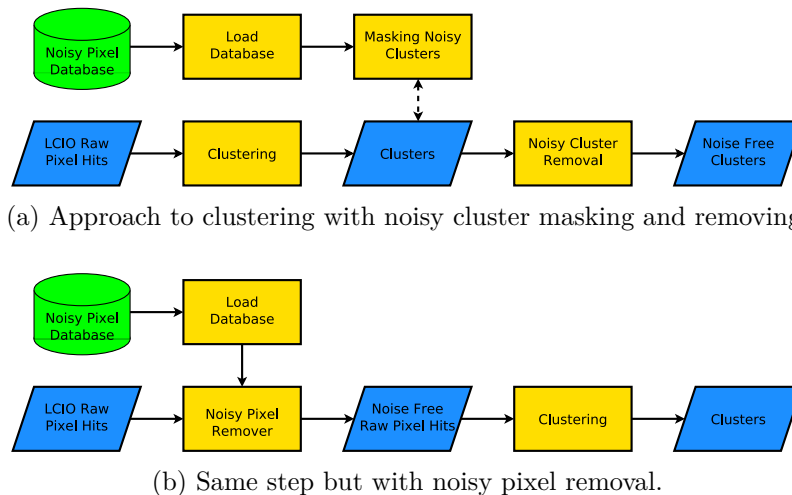(b) Same step but with noisy pixel removal.

Figure 7.3.: Schematic flows of the clustering step with noisy pixel treatment in EUTelescope.

The next step involves clustering raw pixel hits together to yield clusters. Different clustering routines are available and can be configured to meet one's need. Moreover, the noisy pixels are removed to yield a sub-sample of clusters which can be used for alignment and tracking. It must be noted, that clusters with noise can be kept as well for later analysis. Especially, analyses investigating either noisy pixels directly, or assessing a systematic uncertainty due to removing noisy pixels need these data.

The noisy pixel masking and removal can take place at two stages, either prior to clustering or after clustering. In Figure 7.3 the two options are shown. In Figure 7.3a the clustering is performed before masking noisy cluster, in Figure 7.3b the noisy pixels are removed prior to execution of the clustering algorithm.

In both approaches a Marlin processor to load the external noisy pixel database is executed. This makes the LCIO collection with the TrackerData entries in the first event available to all subsequent processors in the execution chain.

A novel processor was implemented to remove noisy pixels prior to clustering. As no assumption can be made on the ordering of the raw data from the DAQ systems, in the clustering routines, each pixel which seeds a cluster has to be checked against all remaining other pixels.

Figure 7.4.: Possible differences obtained with the two approaches to clustering.

Hence, the check to determine a single cluster scales with n^2 where n is the number of pixel hits which could belong to the cluster. It is therefore desirable to keep n low, in particular if the sensor is known to be very noisy.

The *NoisyPixelRemover* loads the noisy pixel database. It will then iterate through all the pixel hits in a TrackerData collection and create a new collection without any of the noisy pixel hits. This collection is then passed as the input collection to the clustering processor, as shown in Figure 7.3b.

It is noteworthy, that all the noisy pixel encoding has been modified to use Cantor's pairing function. In the first event, all the noisy pixels are read in, encoded, and stored in a sequential container. This container is then sorted which enables subsequent searches to exploit a binary search, operating with $\log(n)$ instead of n.

In the case of noise removal after clustering, two processors are executed. The *NoisyClusterMasker* iterates over all clusters and masks the ones containing a noisy pixel. In order to do that, it also reads the noisy pixel database in the first event. The cluster quality field houses a bit which is set to mask a cluster as noisy. The second processor, the *NoisyClusterRemover*, operates very similar to the *NoisyPixelRemover*. It takes the cluster collection with the masked noisy clusters and writes out a new collection which only has clusters which have not been flagged as noisy.

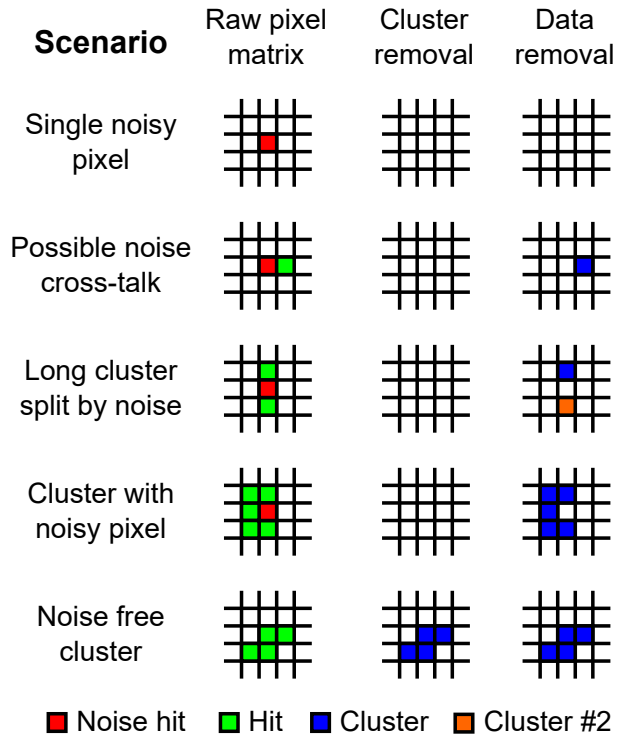The possible differences of these approaches are depicted in Figure 7.4: in the middle,

labelled *Cluster removal*, the case of clustering the noisy data and removing clusters flagged as noisy is compared to the case where noisy data are removed prior to clustering, shown on the right, labelled *Data removal*.

While this schematic raises awareness of the cases which could lead to different results, it is important to keep in mind that for most modules, the noise level should not be at a level where these differences play a crucial role. In most analyses there is a selection of a fiducial volume excluding regions with noisy pixels, rendering effects of the chosen noise treatment on the results negligible.

The central task of this reconstruction step is the actual clustering. Clustering in EUTelescope terminology is the grouping together of hits which are considered to belong to the same cluster. There are several clustering algorithms in EUTelescope. Some of them are legacy implementations which are still maintained for backwards compatibility. Two algorithms are actively maintained and are the recommendation for any current test beam reconstruction. Both algorithms aim at zero-suppressed data.

Zero-suppression is a type of detector response in which only information from read-out channels which detected a hit are propagated. Contrary to zero-suppressed data are data-streams where signal values for each read-out channel are logged when triggered. For example, this could be a voltage over a defined resistivity to digitise a transistor current. Every read-out channel will have an electric current reading, where most will be at their baseline value. The FE-I4 as well as the Mimosa26 sensors perform some type of zero-suppression on their chip. This is also true for many other current pixel sensors.

To cluster non-zero-suppressed data, either the legacy algorithms may be used, or, the recommended way, one could implement a processor to perform zero-suppression on the given data-stream prior to clustering.

The definition which pixels belong to the same cluster can be defined by the user. The *Sparse-Clustering* algorithm uses pixel indices to group them together. Furthermore, temporal requirements can be applied. The metric used to compute the distance between two pixels i, j is defined as: $d_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2$ where x_i and y_i are the pixel indices in x- and y-direction.

Figure 7.5.: Different clustering values and the resulting clusters.

The default parameter used for clustering requires $d_{ij} \leq 1$, corresponding to pixels being required to touch at least at an edge, which however can be changed by the user. Relaxing this to $d_{ij} \leq 2$ would also group together pixels which merely touch diagonally, i.e. at a corner. Increasing this value even more, can group together split clusters. In Figure 7.5 the cases of different d_{ij} requirements are shown, resulting in a different number of clusters.

Aside from *SparseClustering* there is also the *GeometricClustering* implementation. It uses a special pixel type in EUTelescope which not only stores the pixel's indices, charge and time information, but also their geometrical position reflected by a rectangle. The algorithm does not rely on pixel indices as its metric, but exploits the actual spatial
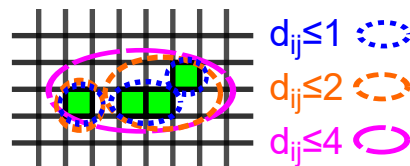
area covered by the pixels. In order to circumvent any floating point precision effects, the rectangles are enlarged by 10% during the cluster finding and any then overlapping pixels are grouped together. In the case of a chequered pixel matrix, this results in the same clusters as *SparseClustering* with $d_{ij} \leq 2$.
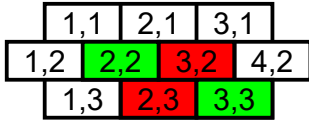

1,1	2,1	3,1	
1,2	2,2	3,2	4,2
	1,3	2,3	3,3

Figure 7.6.: Clustering of bricked pixel matrices.

Advantages of the *GeometricClustering* are that it can also correctly process pixel matrices arranged in a bricked layout as indicated in Figure 7.6. For the *SparseClustering* the two green pixels would both touch at one corner, being clustered together if $d_{ij} \leq 2$ is required. At the same time, the two red pixels would also merely touch at the corner, requiring at least a cut value of $d_{ij} \leq 2$ for them to be grouped together. Hence, there is no cut value for d_{ij} which would correctly cluster a bricked pixel matrix like the one shown in Figure 7.6.

This is not just a theoretical use-case, but such pixel matrices have been used in prototypes to investigate effects of charge sharing by the ATLAS ITk pixel community. Furthermore, the geometric pixel implementation decoupled the pixel's indices from its spatial position. This is important as many recent studies for the ATLAS ITk pixel upgrade used the old FE-I4 read-out chip with altered pixel layouts to mimic the reduced pixel size aimed at for the upgrade. Hence, the default FE layout can be mapped to an arbitrary sensor layout.

7.1.3. Position Derivation and Pre-Alignment Step

The *Hitmaker* is used to derive hit positions from clusters and a schematic flow diagram is shown in Figure 7.7. It is the first step which uses the alignment information from the GEAR file. A change which has been introduced in the context of this thesis is the migration to local coordinates for pixel hits. Local coordinates refer to positions expressed in each sensor's local frame of reference, having its origin in the centre of the pixel matrix. Contrary to that, the global coordinate system is the lab coordinate system. Typically, the most upstream sensor will be placed at its origin and all the other sensors relative to this origin.
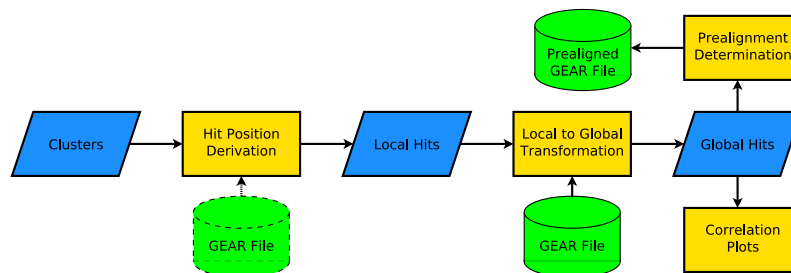

Figure 7.7.: The step of deriving hit positions from clusters as well determining the pre-alignment from correlation plots.

The *Hitmaker* has two different approaches to the derivation of a pixel's position. The first approach uses the GEAR values specified for the pixel pitch, sensor size and pixel index and computes $x_{\text{pos}} = (x_{\text{index}} + 0.5) \times x_{\text{pitch}} - x_{\text{size}}/2$ and analogously the y-direction, respectively. The effect of an inconsistent GEAR file is obvious, a wrong sensor size leads to a linear shift, while wrong pitches will stretch or compress the derived hit position with increasing pixel index.

The second approach uses the geometric pixel and derives the proper hit position using the values stored in the pixel field. Hence, this approach is also feasible with pixel matrices which feature prolonged edge pixels, like the IBL style modules used by the ATLAS ITk pixel community.

The position of a cluster is computed as:

$$X = \frac{1}{Q}\Sigma_i x_i q_i \tag{7.1}$$

Where x_i are the individual positions in a cluster and q_i their recorded charge. Q is the total recorded charge, i.e. Σq_i. This motivates the EUTelescope wide convention that detectors with merely binary, i.e. hit or no-hit, information need to set the hit's charge to 1 (in principle, any non-zero value).

There is one rather counter-intuitive peculiarity in EUTelescope's definition of the local coordinate system. Before storing hits in the local hit LCIO collection, the axis can be flipped and even mirrored. For this purpose, the hits are transformed:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{7.2}$$

The transformation matrix t_{ij} is only allowed to flip and/or mirror axes, i.e.:

$$\det(t_{ij}) = \pm 1 \tag{7.3}$$

Internally, it is ensured that a right-handed coordinate system is maintained.

This is an easy mechanism which allows users to swap or flip axes. This mechanism is useful, as it is common that different sensor modules will be mounted in a way that their axes are swapped or pointing in the mirrored direction compared to the reference telescope sensors.

Once the local hits are stored in an LCIO database, a transformation Processor will convert them into global hits and store them in a further LCIO collection. This collection is discarded after the subsequent processors in this step, i.e. it is not intended to be stored in the final LCIO file after clustering. This follows the new alignment philosophy which has been implemented in EUTelescope within this thesis. A more detailed description of the old versus novel alignment implementation is given in Section 7.2.2.

Similar to the EUDAQ OnlineMonitor tool, two processors will investigate correlation between hits of different modules, namely the *CorrelationProcessor* and the *PrealignmentProcessor*. Both processors require hits in the global detector frame, hence the temporary collection is used as input.

The *CorrelationProcessor* will fill two-dimensional histograms of the position of a hit on one module versus the position of hits on any subsequent module, split up in x- and y-direction. Therefore, it only provides visual feedback to the user. However, it is the most simple way to determine if the geometry is initialised correctly and the data are synchronised properly. An example of a correlation plot is given in Figure 7.8a where the correlation of an IBL type module (*detector 21*) and the first Mimosa26 telescope detector (*detector 0*) is shown.

(a) Correlation plot for a DUT and the first telescope detector in x-direction.

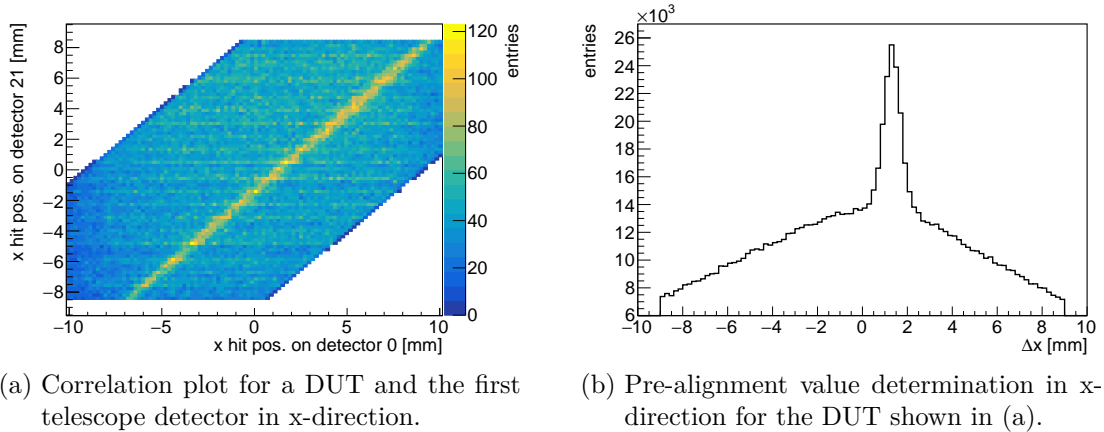(b) Pre-alignment value determination in x-direction for the DUT shown in (a).

Figure 7.8.: Plots created by the *CorrelationProcessor* and *PrealignmentProcessor* in the hit position derivation step in an EUTelescope reconstruction. A cut of 9 mm is applied in both processors.

The *PrealignmentProcessor* processes the hits in a very similar way, it uses the hit positions on the first, i.e. most upstream, detector and uses all hits on subsequent detectors to compute the residual $\Delta x_i^j = x_0 - x_i^j$ where x_0 is the hit position on the initial detector and x_i^j is the j-th hit on the i-th detector. This is done for all hits on the most upstream detector versus all other hits and analogously in y-direction.

These values are binned in a custom histogram object and the bin with the most entries is used to derive a pre-alignment value, i.e. a rough estimate of the shift in x- and y-direction. No x-y correlations are investigated, i.e. there is no pre-alignment of the beam-angle. In Figure 7.8b the pre-alignment plot for the correlation plot in Figure 7.8a is given.

In order to reduce permutation background, for both the processors, a cut value on the residual value can be set.

7.1.4. Alignment Step

After the pre-alignment is performed in the previous step, the alignment step aims at the determination of the precise telescope and DUT alignment. In order to apply the pre-alignment, the same processor as the one used in the previous step is executed to

transform the local hits into global ones. However, the new GEAR file which stores the pre-aligned values is used instead of the original GEAR file.

This is schematically shown in Figure 7.9. The global hits are then used in the alignment process which results in a newly updated GEAR file. This process can be carried out iteratively, as indicated by the dotted arrow. In case of iterative alignment, the new GEAR file is used as input in the subsequent alignment step.
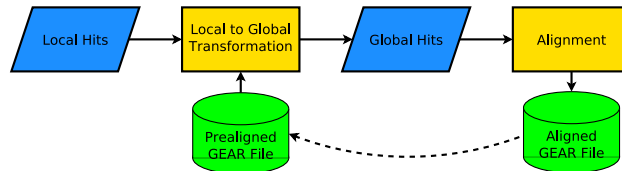

Figure 7.9.: The process flow of the alignment in EUTelescope. The dotted arrow indicates the possibility of iterative alignment.

The actual alignment is performed with the MillepedeII framework. MillepedeII performs an overall least squares fit, fitting the global and local parameters simultaneously in a single fit. The global parameters are the parameters that are the same for every track, i.e. the positions and rotations which need to be aligned. For example, these could be the x- and y-shifts as well as rotations along the beam axis. Then every detector module being aligned would contribute with three global parameters. If a linear track model is used, i.e. a straight line, the track itself is parametrised with a two-dimensional point and a slope, hence four local parameters per track. Every further track introduces four additional local parameters into the fit.

MillepedeII exploits the mathematical structure of the problem, performing a simultaneous fit and solving the global χ^2-minimisation for the global, i.e. alignment, parameters. This makes MillepedeII more powerful compared to other alignment approaches where alignment parameters are often assumed to be fixed and iteratively corrected in hope of them converging. The downside of keeping alignment parameters fixed in the alignment process is that the track fit is biased by the misaligned data-point and there is no guarantee that the alignment will converge.

There are two ways to interface the MillepedeII framework. The first one is the *Mille* processor which is a generic interface to it, allowing different input modes for obtaining tracks. Either an external track fit processor is used or an internal simple track finding is carried out by the processor itself. The second way uses the General-Broken-Line framework, short GBL. GBL is a library dedicated to a broken line track refit, it has a native interface to MillepedeII built in which makes it useful to be used in conjunction with MillepedeII. For this type of alignment a standalone processor, exploiting the general GBL functionality in EUTelescope, is available.

7.1.5. Track Finding and Track Fitting Step

The ultimate step is the final track fit. Two approaches are available to the user: an implementation of a Kálmán filter (KF), the Deterministic Annealing Filter (DAF) fitter,

as well as the previously mentioned GBL track re-fit.

Again, the hits are aligned by using the local to global hit position transformation of the EUTelescope framework. Using the most recent GEAR file, aligned hits are obtained which are then used as input into the track finding and fitting. This is the first section in the schematic shown in Figure 7.10.
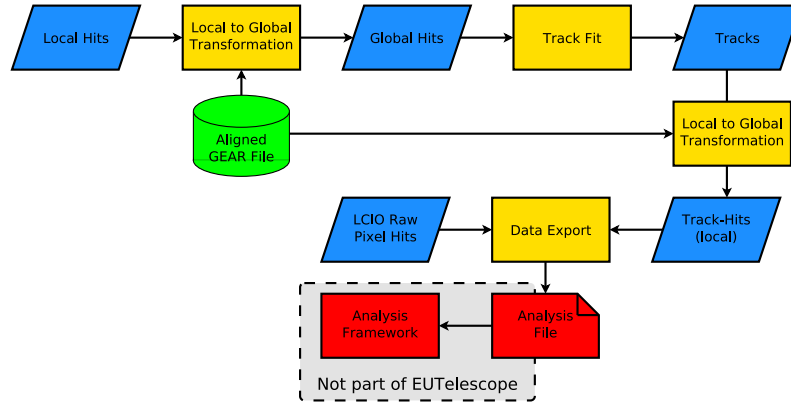

Figure 7.10.: The ultimate step in a test beam reconstruction, the track fit.

After the track fit is performed, it is typically desired to export the tracks into an external analysis framework, at least this is often the case for the ATLAS ITk pixel community. In order to make the analysis independent of the alignment constants used during the reconstruction, the alignment is undone and the exported track is parametrised in local coordinates. Furthermore, the raw data are exported as well.

The Kálmán filter approach requires a track candidate, i.e. a collection of hits which are considered to originate from the same particle. It then starts at the initial detector module and recursively updates its state as it propagates through the telescope detectors and processes the hit information. The state estimate improves further as more information is provided to the filter, i.e. it will have the best state estimate on the last detector module.

In order to also obtain a good state estimate on the initial plane, a KF running in the opposite direction is executed as well, hence, this filter will provide the best estimate on the initial plane. As precise track information all along the particle's trajectory is needed, a smoother combines the forward and backward running KF in order to obtain the best estimate on every detector.

In order to introduce the effect of multiple scattering into the KF, there is a propagation step of the state *through* the sensor. The running KF has a prediction of the state on the k-th detector module, the measurement on this module will be combined with the prediction to obtain the updated state on detector k. This state is then propagated through detector k, in this step the material and multiple scattering is introduced as process noise. This final state is then used for the prediction onto the next detector $k+1$. The amount of noise, i.e. scattering, is estimated using the scattering model described

in Section 4.4 using Equation 4.5. The scattering will increase the uncertainty on the angular spread, but not on the position.

For track finding, the DAF implementation has a combinatorial cluster finder, which projects all hits onto the initial plane and merely looks geometrically for spatially near clusters. The second option is to use a KF for track finding as well, seeding possible trajectories from the initial plane and propagating through the set-up.

The second track fit implementation uses the GBL framework. It was initially implemented for the telescope only, i.e. for a set-up of six telescope detectors, divided into the upstream and downstream arm of three detectors each. The track finding is tailored to this set-up, using a so-called triplet finding and triplet matching approach.

A triplet is a candidate track segment composed out of three hits. It is seeded by two hits, one on the initial and one on the ultimate detector of the corresponding telescope arm. A straight line is projected to the middle detector and if a hit is found in proximity on that detector, a triplet is formed. As the implementation of the GBL algorithm has been modified, a more detailed description is given in Section 7.2.3.

7.2. Major Modifications To EUTelescope

Aside from the implementation of several new processors which were mentioned already, some very fundamental changes in EUTelescope have been made.

7.2.1. Data Interfacing

As discussed in Section 5.2.2 there is no guideline how the values stored in an LCIO::TrackerData object are to be interpreted. It is up to EUTelescope to ensure that these values are correctly read back and provided to the user's processor. The way the data are accessed and stored in the backend framework has been modified to make LCIO more user friendly and up-to-date to modern languages. The concept of range based loops is the modern way of using container structures in most languages.

It is obvious that the LCIO data interfacing object has to know the pixel's type, hence it is templated in this `PixelType`. However, in many cases users might simply need the basic pixel information, i.e. their x- and y-indices. This is the base information all pixels inherit from. An example is the noisy pixel treatment, in order to check if a pixel is noisy we merely need its indices, no more further information.

The excerpt below sketches the way this had been implemented:

```
PixelType* getSparsePixelAt(size_t index, PixelType* pixel) const;
\* Templated implementation
 * Process the LCIO stuff in here
 */

\\ Generic access
EUTelBaseSparsePixel* getSparsePixelAt(
                        size_t index,
                        EUTelBaseSparsePixel* pixel) const {
```

```
    if(!pixel) pixel = new PixelType();
    EUTelBaseSparsePixel* result = static_cast<EUTelBaseSparsePixel*>(
                                   getSparsePixelAt(index,
                                   dynamic_cast<PixelType*>(pixel))
                              );
    pixel = result;
    return result;
}
```

The generic access is potentially leaking memory and the ownership semantics are not clear at all. Moreover, the call of `getSparsePixelAt` will bloat up code and an additional index is required.

```
EUTelBaseSparsePixel* pixel = nullptr;
for(size_t iPixel = 0; iPixel < sparseDataInterface->size(); iPixel++) {
        pixel = sparseDataInterface->getSparsePixelAt(iPixel, pixel);
        .. do stuff ..
}
delete pixel;
```

The old implementation will on request load the information from the LCIO object into the allocated space of the pixel pointer or allocate that space and then load this information. However, in all processors in EUTelescope it is the case that one wants to loop over all the pixels. Hence, the new implementation will keep a container of the pixels in sequential memory upon construction of the interfacing class. This allows to mimic a container-like data structure.

If the pixel type is known one can easily directly access the pixel data:

```
auto sparseData = EUTelTrackerDataInterfacerImpl<PixelType>(trackerData);
for (auto& pixel: sparseData) {
    .. do stuff ..
}
```

In the more generic context, the syntax is not that simple, mainly because the data interface is kept as polymorphic pointer and needs dereferencing and the underlying vector is accessed via a `std::reference_wrapper`. Despite this, there is no chance of leaking memory and it is possible to use range based loops.

```
for(auto& pixelRef: *sparseData) {
        auto& pixel = pixelRef.get();
        .. do stuff ..
}
```

Moreover, the references to the pixel objects kept in the containers of the interfacing class are ensured to be `const`. Modifying them would not update the LCIO container, this is a possible pitfall as users might expect this behaviour. In the old implementation there was no protection against this.

In addition, access operators, iterators and appending functions are newly available. A user creates an interface to `LCIO::TrackerData` and can simply call container-like functions like `emplace_back`, `at`, etc.

These modifications were applied to every processor interfacing LCIO raw data, especially also all ConverterPlugins in EUDAQ which export LCIO data. Hence, all these parts have been refactored.

7.2.2. Alignment

The alignment comes into play at various stages in EUTelescope. The pre-alignment is derived during the hit position derivation step, the *precise* alignment in the subsequent alignment step, possibly iteratively. Before the alignment modifications, the alignment corrections were always applied atop of each other. Every step resulted in an individual alignment database which were applied successively. A difficulty which arose was that rotations were parametrised in the local reference frame.

In order to correctly rotate hits, it was necessary to shift the hits back into the origin, apply the rotation and shift them back. For this purpose a so-called *reference hit* was kept which stored the information on the total shifts. This is depicted in Figure 7.11 where alignments are applied atop of each other. Each alignment corresponds to a linear shift, indicated by the large arrow and rotations at the new position, indicated by the smaller arrows. Assume the case that the first two transformations in Figure 7.11 are applied and a further, third, transformation is applied atop. Initially, the hit would need to be transformed back into the origin and the new rotations applied. For this, the hit is transformed back along the negative reference hit, the rotations applied, and shifted by the reference hit again. Then, the new offset would be applied and the reference hit would be updated.
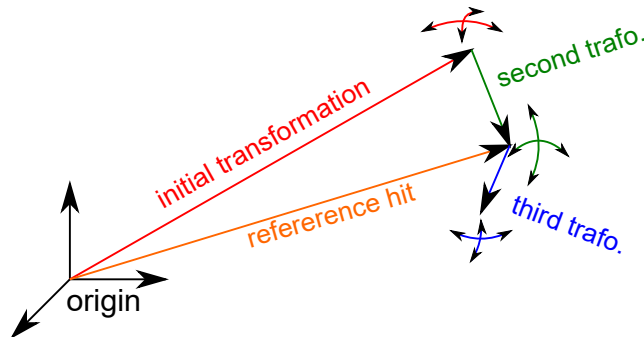

Figure 7.11.: Old alignment scheme in EUTelescope.

As there is no internal mechanism to track which alignments were already applied, this scheme is error prone as the users need to precisely track their alignment. In the novel alignment scheme, hits are always stored in the local frame of reference and the global alignment is updated.

The rotations for the angles are parametrised as Euler angles, in particular within EUTelescope a Tait–Bryan angle convention is used. Internally, transformations are expressed via rotation matrices. The idea of using quaternions was discarded, mainly because matrices are more widely used in test beam reconstruction. Quaternions are

commonly used in computer visualisation, as they can be implemented slightly more efficient. Furthermore, interpolating rotations with them is easier than with rotation matrices which is needed when rendering smooth rotations in video animations. As only a single rotation, which is not time nor memory critical, is performed when reconstructing test beam data, rotation matrices are perfectly well suited for this context.

Angle Scheme

Using rotation matrices in three dimension:

$$r_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix} \quad r_y(\theta_y) = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix}$$

$$r_z(\theta_z) = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

EUTelescope follows a yxz convention, i.e. an initial rotation along the beam axis (z) is followed by the x- and then the y-rotation. This yields:

$$r_{ij} = r_y(\theta_y)r_x(\theta_x)r_z(\theta_z) =$$
$$\begin{bmatrix} \cos\theta_y\cos\theta_z + \sin\theta_x\sin\theta_y\sin\theta_z & \cos\theta_z\sin\theta_x\sin\theta_y - \cos\theta_y\sin\theta_z & \cos\theta_x\sin\theta_y \\ \cos\theta_x\sin\theta_z & \cos\theta_x\cos\theta_z & -\sin\theta_x \\ \cos\theta_y\sin\theta_x\sin\theta_z - \cos\theta_z\sin\theta_y & \cos\theta_y\cos\theta_z\sin\theta_x + \sin\theta_y\sin\theta_z & \cos\theta_x\cos\theta_y \end{bmatrix}$$

When deriving the angles from the matrix, via the $r_{12} = -\sin\theta_x$ entry, θ_x can be computed. If $\theta_x = \pm\pi/2$ then there will be one degree of freedom which can be chosen freely, this is the case of gimbal lock.

If $r_{12} \neq \pm\pi/2$ then $\cos\theta_x \neq 0$ and the ratio of r_{02}/r_{22} yields θ_y as well as r_{10}/r_{11} yields θ_z. The function arctan2(y,x) is used to compute θ_y and θ_z, it corresponds to atan(y/x) but takes the signs of the input arguments and provides the angle in the correct quadrant. It is a generalisation of the atan(x) function which maps onto its principle values with an image of $(-\pi/2; \pi/2)$. The arctan2(y,x) function maps onto an image of $(-\pi; \pi]$.

Summarising, in the first case:

$$\theta_x \neq \pm\pi/2 \quad \theta_x = -\mathrm{asin}(r_{12}) \quad \theta_y = \mathrm{atan2}(r_{02}, r_{22}) \quad \theta_z = \mathrm{atan2}(r_{10}, r_{11}) \tag{7.4}$$

Given that $\theta_x = \pi/2$ then $\sin\theta_x = 1$ and $\cos\theta_x = 0$ and the above solution (Eq. 7.4) will be undefined. However, one obtains:

$$r_{00} = \cos\theta_y\cos\theta_z + \sin\theta_y\sin\theta_z = \cos(\theta_z - \theta_y)$$
$$r_{01} = \cos\theta_z\sin\theta_y - \cos\theta_y\sin\theta_z = -\sin(\theta_z - \theta_y)$$

Which can be exploited to give:

$$\theta_x = \pi/2 \quad \theta_z - \theta_y = \mathrm{atan2}(-r_{01}, r_{00}) \tag{7.5}$$

In the final case, $\theta_x = -\pi/2$ which again gives $\cos\theta_x = 0$ but $\sin\theta_x = -1$. In the same fashion:

$$r_{00} = \cos\theta_y\cos\theta_z - \sin\theta_y\sin\theta_z = \cos(\theta_z + \theta_y)$$
$$r_{01} = -\cos\theta_z\sin\theta_y - \cos\theta_y\sin\theta_z = -\sin(\theta_z + \theta_y)$$

Hence:

$$\theta_x = -\pi/2 \quad \theta_z + \theta_y = \mathrm{atan2}(-r_{01}, r_{00}) \tag{7.6}$$

In the decomposition from the matrix to angles implemented in EUTelescope, in case of a non-unique solution, θ_z is set to 0. Hence, the algorithmic decomposition looks like:

```
if(r12 < 1) {
  if(r12 > −1) {
    x = asin(−r12)
    y = atan2(r02, r22)
    z = atan2(r10, r11)
  } else {
    x = pi/2
    y = −atan2(−r01, r00)
    z = 0
  }
} else {
  x = −pi/2
  y = atan2(−r01, r00)
  z = 0
}
```

This exploits Equations 7.4, 7.5 and 7.6. The evaluation of asin() will only be executed in the case of $\theta_x \neq \pm\pi/2$. The above decomposition scheme shows that the problem of gimbal lock is not a feature of the matrix representation, but a feature of the angle parametrisation. However, using angles is the easiest way for a user to define rotations and the way it is implemented in the GEAR file.

Alignment frameworks like the MillepedeII toolkit will provide corrections to the rotations of the form: $r_{ij} \rightarrow \Delta r_{ik} r_{kj}$ where Δr_{ik} are the corrections to the rotations. The implemented decomposition of the new rotation matrix allows to derive the angles which are then stored in the updated GEAR file, regardless of how the corrections were initially parametrised.

The entire EUTelescope framework has been modified to consistently use this convention. In particular, functionality to internally fill and cache the correct rotation matrices from a set of angles, as well as functionality to derive updated angles from a corrected rotation matrix and store them in a GEAR file, was implemented. A new processor has been written to process the output of alignment procedures, correct the current GEAR

files and write a new GEAR file. Hence, the new alignment scheme is also called the *GEAR file based alignment.* This alignment scheme has been implemented in multiple examples and is the recommendation for any reconstruction of the ATLAS pixel community. Several validations, in particular with iterative alignment where the convergence of the alignment was verified, but also with Monte Carlo data, were carried out. The results from the Monte Carlo validation are given in the next Chapter.

7.2.3. General Broken Line Algorithm

As already mentioned in Section 7.1.5, to seed the GBL track refit, so-called *triplets* are used. The concept of triplets is a track finding technique, and in the way it was implemented into EUTelescope, it is tailored to a test beam set-up with six telescope modules which are separated into an upstream and downstream arm of three modules each.

Track Finding

On each arm of the telescope the triplets are constructed. The initial and final detectors are used to seed the candidates. Every combination of a pair of hits, one on the first and one on the last detector, is used to construct a straight line through those points. A slope cut on this straight line is used to reduce combinatorial background, this is shown in Figure 7.12a where the lower triplet candidate is rejected for a too steep slope. For the passing candidates, the trajectory is extrapolated to the sensor in the centre. If a hit in close proximity is found at this location on the sensor in the centre, the triplet is accepted, as depicted in Figure 7.12b. The maximal search radius is again a parameter of this track finding algorithm.

Once triplets are formed on both arms of the telescope, the straight lines are extrapolated to the virtual plane half-way between the two arms of the telescope. Triplets from the upstream and downstream arm are matched by proximity, again governed by a distance cut. This is shown in Figure 7.13 where on the one arm, five triplets are matched to two triplets on the other arm. In addition to the proximity requirement, an isolation can be enforced which rejects ambiguous matches. This is the upper most case shown in the figure. The other two cases show an accepted triplet-triplet pair and a single triplet on the left arm, not matched to any other triplet.

This approach can also be used with DUTs, however, some peculiarities need to be considered. The matching of upstream and downstream triplets will be in the region where the DUTs are placed. Therefore, the material of the DUTs will impact the trajectory and the matching distance needs to be scaled with the amount of introduced scatterer. On a more technical note, as the concept of triplets is geared towards sets of three points, no way of attaching DUT hits to a trajectory was available.

In order to support an arbitrary set-up, an algorithm which links DUT hits to triplets was implemented, as well as the possible link between additional hits and the DUT. In the GBL processors, the user can specify which detectors form the upstream and downstream arm of the telescope. The DUTs between those arms will then be automatically matched

(a) Triplet candidates, formed by hits on the outer modules, are initially rejected if their slope is too steep.

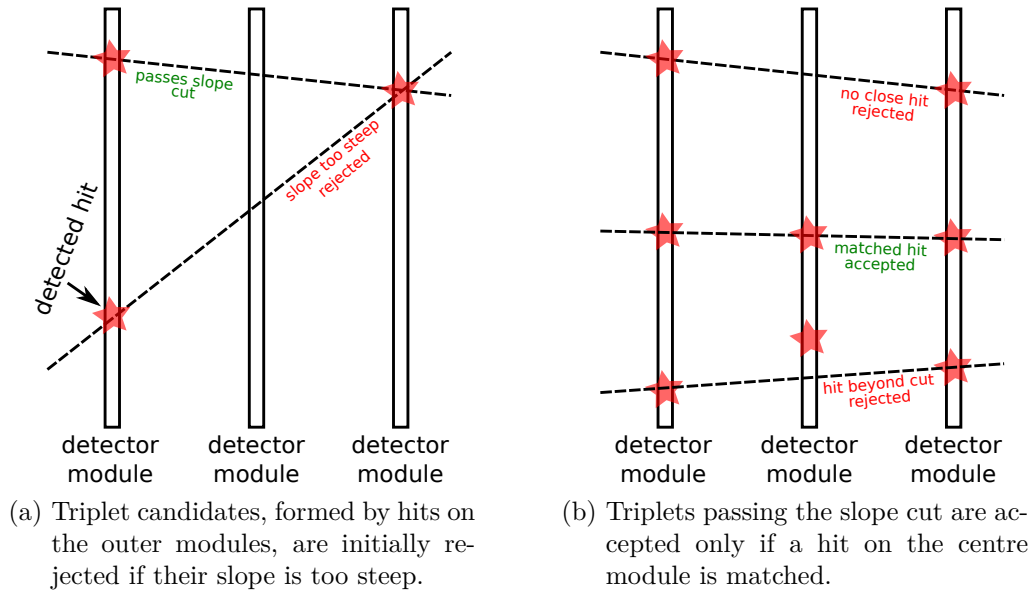(b) Triplets passing the slope cut are accepted only if a hit on the centre module is matched.

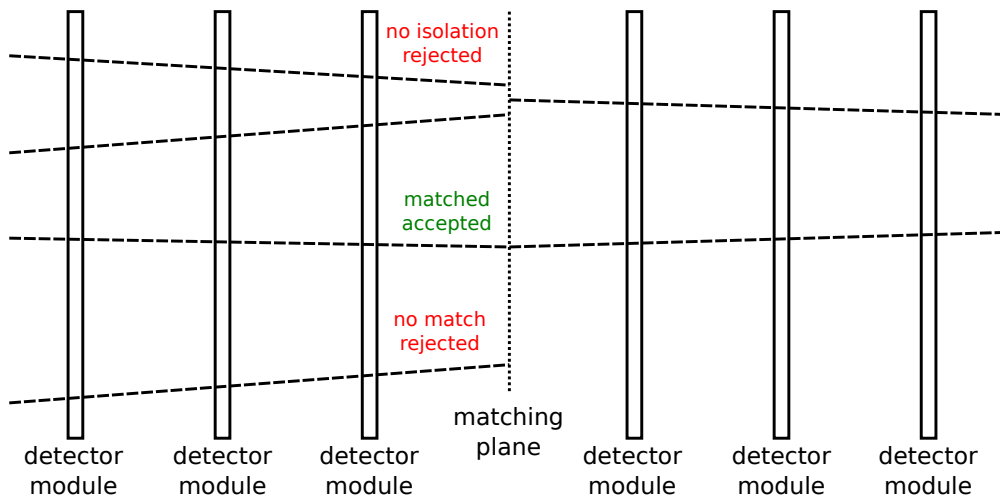Figure 7.12.: The two-step process of finding triplets on the upstream and downstream arm of the telescope.

Figure 7.13.: Old alignment scheme in EUTelescope.

to the closer triplet. To give an example, in the case of two DUTs in the centre of the telescope, the more upstream DUT will be matched with the upstream triplet in a found upstream-downstream triplet pair. The same is done for the downstream DUT and more downstream triplet.

One could argue that attaching a DUT to the trajectory is not needed, as DUT measurements are usually excluded in the track fit. However, attaching DUTs to the trajectory is necessary for three reasons. First, the existence of the DUT needs to be

known in order to incorporate the material of the DUT into the track fit. Second, in the alignment step, the measurement on the DUT is very well needed to compute the alignment corrections. And third, in order to fill residual and similar histograms, even if the measurement is excluded from the track fit, it needs to be associated to the track.

GBL Track Construction

The trajectory composed of two matched triplets is already a track, with a possible kink in the centre of the telescope. In that sense, the GBL framework provides a track refit of that preliminary track. It takes this track and performs the final track fit. In order to execute the track fit, the trajectory within the GBL framework needs to be constructed.

Construction of the trajectory is done by providing points to GBL in their correct sequence. However, a point may not only be a measurement in one of the detector modules, but can also be merely a scattering point. This is useful in two applications, if an additional layer of material is introduced in the beam, or to model scattering by extended volumes of material. The latter is important because it allows to incorporate scattering in air into the GBL trajectory. Every detector plane specified in the GEAR file will be added as a point into the GBL trajectory. The air between two detector planes is modelled via two scattering planes after the first detector and before the second one. To illustrate this, in Figure 7.14 the simplified case of three detector planes is shown. The detectors themselves will contribute as scattering planes, however, in addition, two scattering planes to account for air are placed between all detectors.
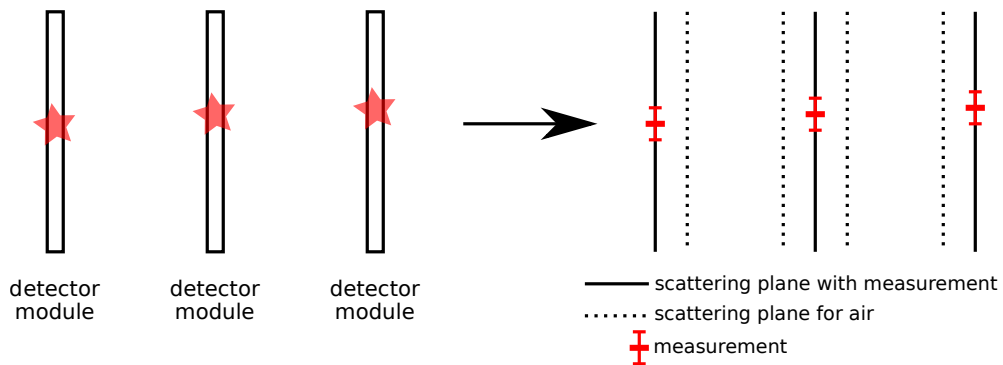

Figure 7.14.: Construction of a GBL trajectory with measurements and scattering planes.

The implementation of the GBL trajectory within EUTelescope was adapted to take any plane specified in the GEAR file as a scattering plane. This way, also other materials like cooling boxes or any other structure, can be included in the trajectory description.

Track Fit

Once the complete trajectory is built, either the final track fit can be performed, or the interface of GBL to MillepedeII exploited, to pass the trajectory to MillepedeII for

alignment. The obtained results are filled into histogramms by the appropriate processors. For example, studies exploiting the functionality of the new alignment scheme as discussed in Section 7.2.2 are given in Section 8.1. In the next section, the aspects and results of GBL are discussed. Initially, in Section 8.1.1, results of the GBL algorithm without a DUT are discussed, in particular looking at cluster properties of the telescope. Then, in Section 8.1.2, results with the novel DUT implementation will be given.

EUTelescope Studies and Validation

8.1. GBL Studies and Results

8.1.1. Cluster Studies

While the mere GBL implementation is not novel, the entire GBL alignment processor was refactored and modified to work with an arbitrary number of detectors. Following changes have also been made: Firstly, the new alignment scheme is used in the updated code which makes transformations from the global to the local detector frame easily possible. Secondly, the GBL processor now correctly processes the cluster sizes in x- and y-direction. Now, plots for the various cluster sizes can be separated by their corresponding size in that direction.

Initially, the rotations were not automatically undone with the old implementation. This posed a problem for any of the detectors after the first one, which defines the coordinate systems origin, and has accordingly no rotation.

In the subsequently shown plots, the terms biased and unbiased are used. A biased result, for example a biased residual distribution, has the detector of interest as an active measurement plane in the track fit. That is, the measurement point will bias the shown result. In the unbiased case, the measurement point is excluded from the fit and the shown results are not biased towards the measurement.

The samples shown were dedicated GBL runs by the DESY groups at the DESY test beam facility at a beam energy of 6 GeV. They are available as reference data for any EUTelescope user, which has the advantage that any user can redo the reconstruction and obtain these results. They are dedicated runs with the EUDET-style telescope only.

In Figure 8.1 the in-pixel hit maps for the track impact position for clusters of size one are shown. The maps are the biased and unbiased results for the second sensor in the beam. Obviously, no distinction between x- and y-direction is necessary for a total

cluster size of one. In the biased case the measurement will pull the track-fit towards the pixels centre, which is why a more pronounced peak is visible.
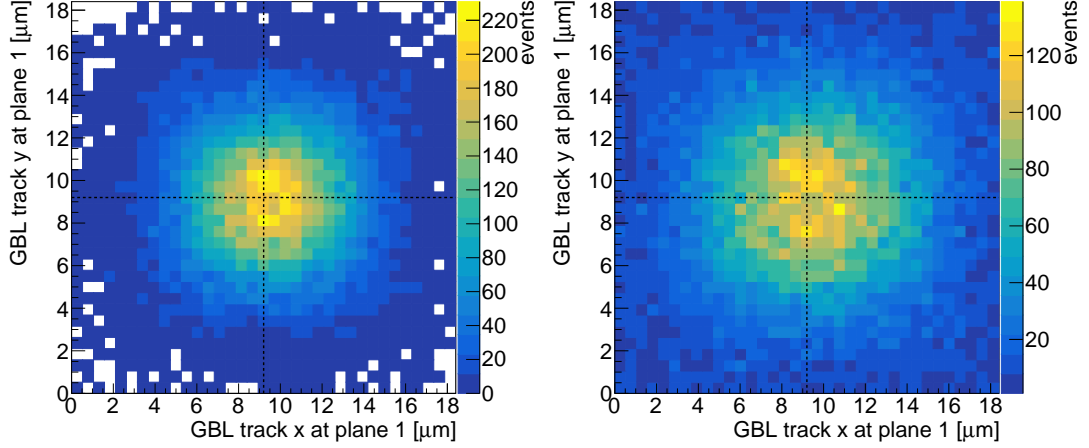
(a) Biased result for the second detector in the beam.

(b) The same distribution, but with the measurement point not biasing the track fit.

Figure 8.1.: Track position in the biased and unbiased case for a total cluster size of one. The bias is clearly visible. The unbiased distribution can be used to validate the alignment.

Instead of showing these result for each individual cluster size, in Figure 8.2 the spatially resolved average cluster size is shown. To obtain this result, the individual cluster size plots (like the ones shown in Figure 8.1) are added weighted by their corresponding cluster size and are normalised by the total number of entries made.

Furthermore, an arbitrary total cluster size can be studied by only looking at clusters of size one in x- and y-direction. Assuming isotropic charge sharing, one would expect central hits in the direction where the cluster size is one. Such in-pixel plots can give an estimation of the charge sharing properties of a sensor. The unbiased results for clusters of size one in x- and y-direction are shown in Figure 8.3. The results obtained are the same in x- and y-direction. It is possible to perform fits to the projections for various cluster sizes, this is shown next.

In order to use these plots to study the alignment, as it was re-implemented, the plots for all clusters with a cluster size of two in x-direction with any total cluster size, were used. This is the inverted case of the above discussed cluster size of one in a specified direction, as in the case of a size of two, hits are expected mostly between two pixels, i.e. at the edge of the pixel. This is shown in Figure 8.4 where the actual track map and the projection onto the x-axis are shown.

The projections of the unbiased track maps, like the one depicted, are summarised in Figure 8.5 for all the detectors in the telescope. A distinct and clearly visible feature
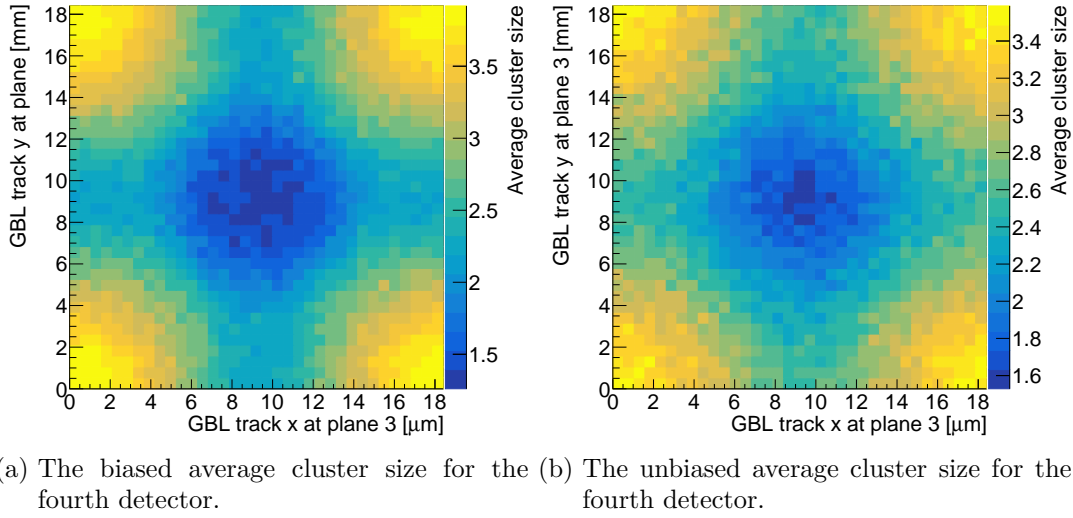
(a) The biased average cluster size for the fourth detector.

(b) The unbiased average cluster size for the fourth detector.

Figure 8.2.: Spatially resolved average cluster size, combining the information from the individual cluster size plots.

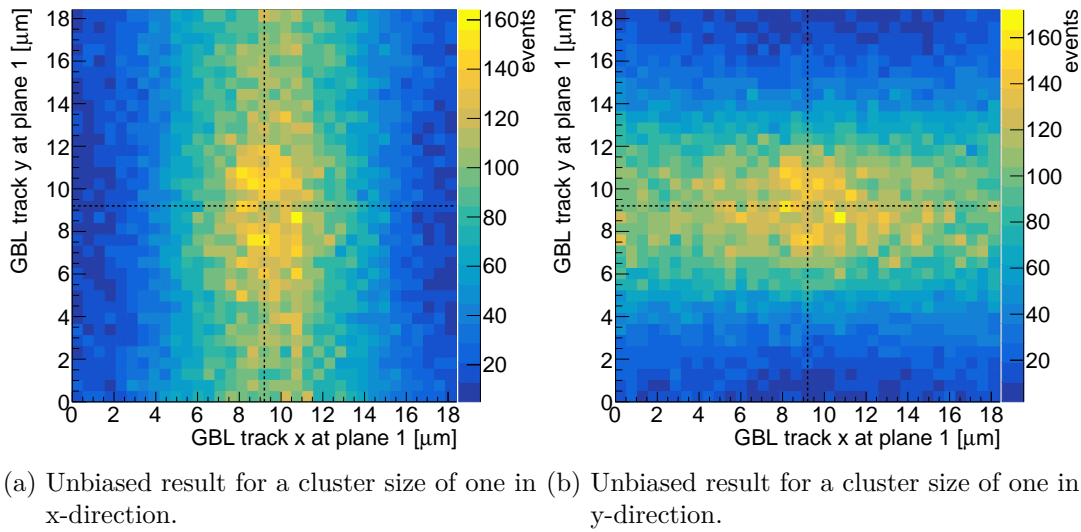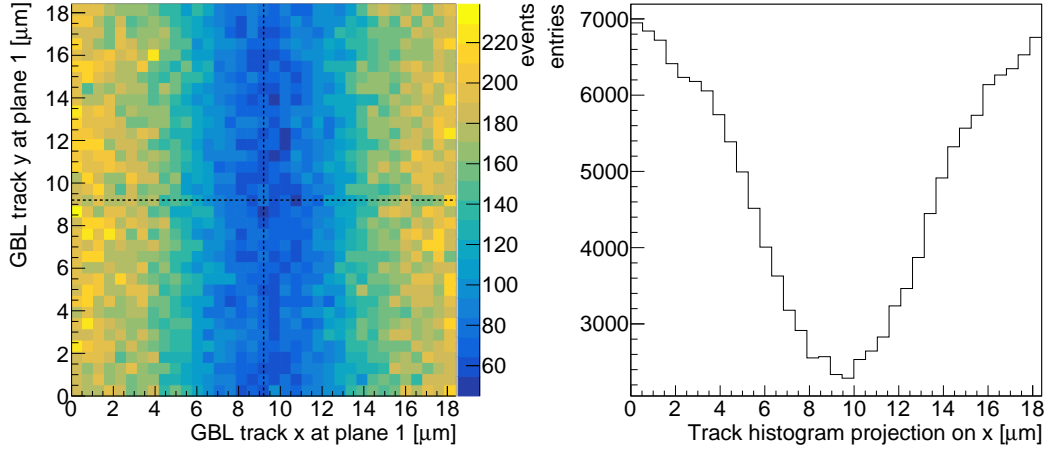(a) Unbiased result for a cluster size of one in x-direction.

(b) Unbiased result for a cluster size of one in y-direction.

Figure 8.3.: Track positions for cluster sizes partitioned in x- and y-direction cluster size.

is that the first and last detector plane show a less pronounced dip. This is due to the fact that there is no measurement point prior to or after these detector planes. Hence, the predicted trajectory is not confined by both-sided measurement points, but merely an extrapolation from the one-sided measurement.

In order to assess the spatial alignment, a fit to the obtained profiles is made. In

(a) Unbiased result for a cluster size of two in x-direction.

(b) The projection onto the x-axis of the track map shown in (a).

Figure 8.4.: Results using the data with a cluster size of two in x-direction.

particular, a constant value minus a Gaussian distribution is fitted:

$$f(const, \sigma, x_0, N; x) = const - \mathrm{N} \times \mathrm{Gaussian}(\sigma; x - x_0) \qquad (8.1)$$

The derived centre values are all slightly biased towards a higher value than 9.2 µm, which is the expected centre of the pixel, ranging from 9.3 µm to 9.5 µm. These deviations are a magnitude lower than the reported tracking resolution of the telescope and might be due to a slight misalignment, yet, this claim is not validated.

In the same fashion the projections of the biased track maps can be studied. It would be wrong to validate the alignment from those, as the measurement of the detector under investigation is included and biases the result. Nevertheless, they show the features of the track fit itself. In Figure 8.6 the same projections of the hit maps, as shown in Figure 8.5, are shown, but for the biased case. In the biased results the initial and last plane feature the exact opposite behaviour. They are the most pronounced. This can be explained as in the case of the biased measurement they seed the trajectory. One must take precaution to not incorrectly assume that this is due to better tracking resolution. If one thinks of a filter application, the filtered state on this detector is the measured state in the case of seeding the trajectory, i.e. maximally biased. This information is then combined with a filter running in opposite direction to obtain the actual predicted state.

Furthermore, the χ^2-distributions of the obtained track fits can be investigated. Each measurement provides a two-dimensional point. A track is parametrised by a spatial point and a direction. Accordingly, the degrees of freedom are $n \times 2 - 4$ where $n \times 2$
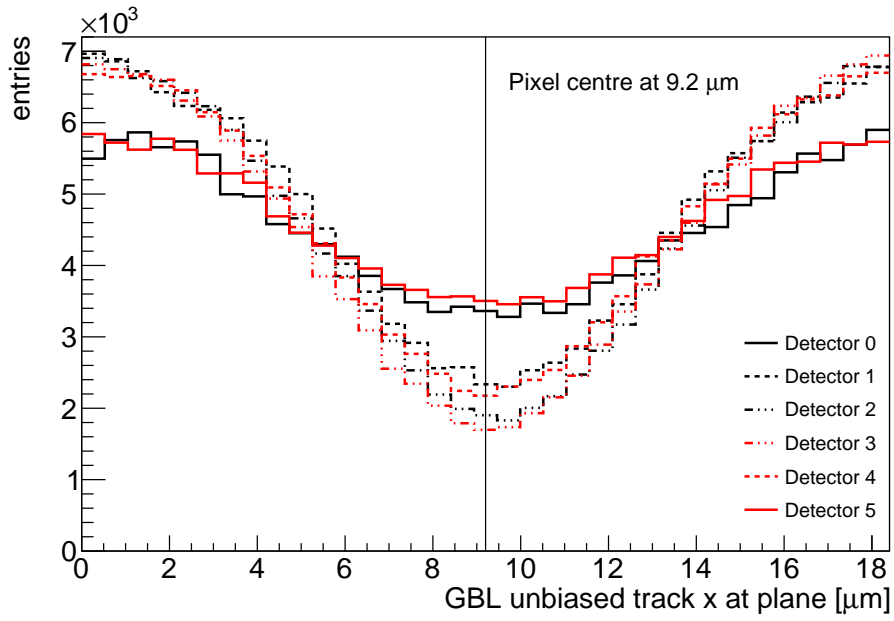
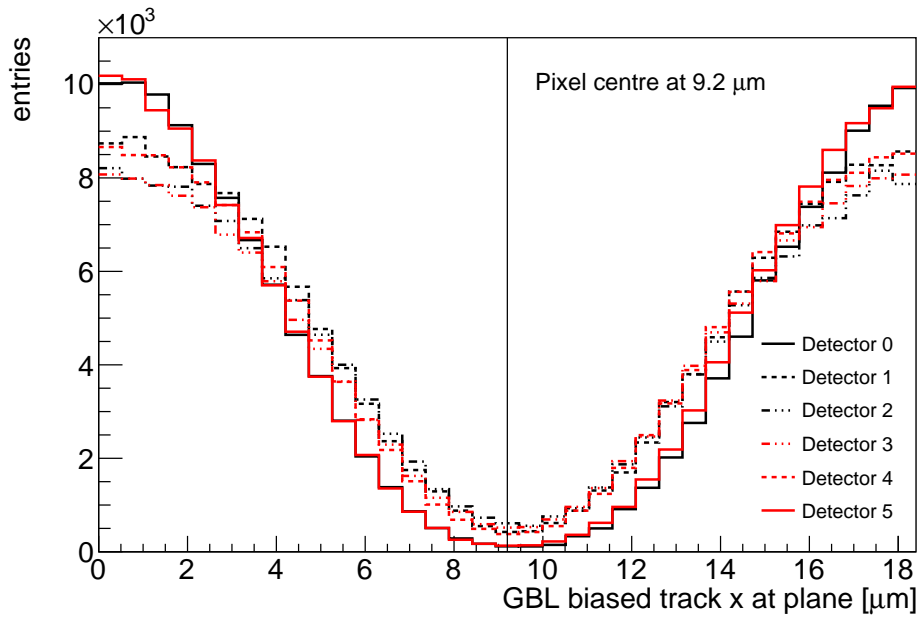Figure 8.5.: The results from the biased profile analysis.

Figure 8.6.: The biased x-axis projections for the track maps.

is the number of measurements (n) in two directions, minus the number of parameters consumed by the track parametrisation (4).

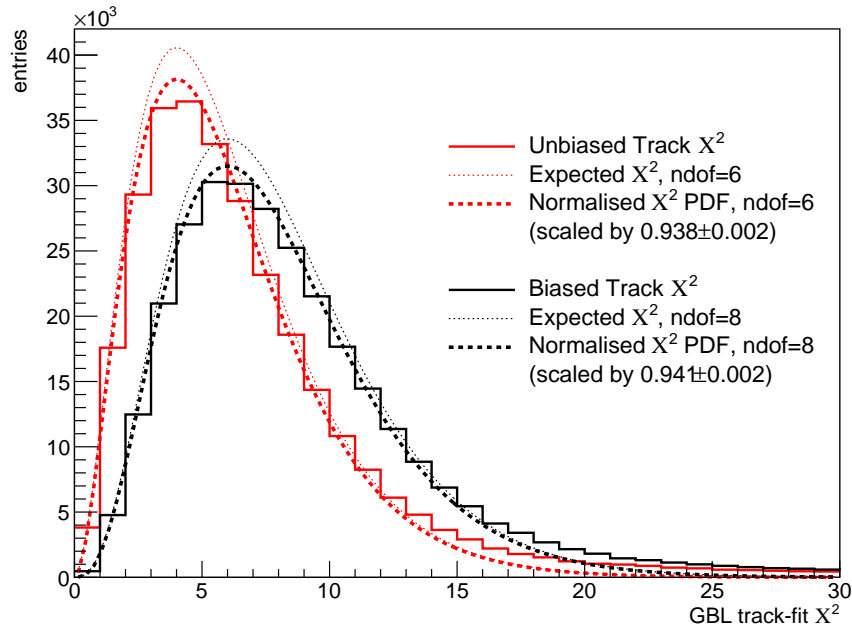In Figure 8.7 the obtained distributions for the biased and unbiased case are shown.

Figure 8.7.: The χ^2 distribution for the biased and unbiased track fits.

In the unbiased case, the second detector module was not included in the measurement. In the biased measurement $6 \times 2 - 4 = 8$ degrees of freedom are present, whereas in the unbiased case there is one detector plane less, i.e. $5 \times 2 - 4 = 6$. The full lines show the results from data. The dotted distribution is the theoretical expectation, that is a χ^2 distribution for the given degrees of freedom and scaled by the number of events. If the theoretical distribution is compared to the one in data, the distribution from data shows more pronounced tails towards high χ^2 values. This can be explained by the fact that, as discussed in Section 4.4, there is a non-Gaussian tail in the angular distribution due to multiple scattering. Events which experience more scattering than expected also push the χ^2 distribution to larger values. As these events are correlated in the biased and unbiased sample, the effect should be the same in both samples. This is validated by deriving a scaling factor by performing a fit which fits the scaled theoretical distribution to the experimentally obtained one. Within uncertainty, both fits reveal a consistent scaling by approximately 0.94. The fit results are shown with dashed lines.

8.1.2. Results with DUT

In order for GBL to be used with a DUT, it needs an improved performance compared to previous track fit implementations. There are several technical reasons this is the case, which are difficult to assess in track fit performance, mainly on the implementation details. The GBL library can be easily included into the project and is in the implementation sense a very lightweight approach, compared to the entire implementation of a Deterministic Annealing Filter (DAF). The direct interface between GBL and Mille-

pedeII is also desired as it again keeps the implementation simple and clean. There are less errors to be made as a lot of functionality is implemented by GBL and does not have to be rewritten in EUTelescope. In the light of maintainability, this is a big advantage as it keeps implementations separated and easy to understand.

In order to justify the implementation of GBL, there need to be advantages over the existing track fits aside from maintainability. The GBL implementation is more powerful in terms of the material description. This means, that both scattering in air, but also due to other material in the beam are correctly described. Furthermore, the GBL implementation allows to set the measurement uncertainty on the measurement for each hit independently depending on cluster size in both directions in contrast to the DAF fitter where an average resolution is used.

In order to compare the two track fits, data of a run taken at the DESY test beam facility at 4 GeV are reconstructed and compared. The set-up was a standard ATLAS pixel measurement with the Mimosa26 telescope and two LHC-type devices, in particular two FE-I4 modules, between the upstream and downstream arm of the telescope.

Shown in Figure 8.8 are the obtained residuals of the track fits on the second telescope module, i.e. plane 1. A fit of a Gaussian is used to quantify the width of the residuals. The assumption is, that the GBL track fit will provide a better residual due to the more precise description of the scattering system and as the results will shown, this indeed is the case.

The Kálmán filter in the DAF implementation uses state predictions in both directions, combining them via a smoother. As a consequence, the predicted state on detector is not biased by the measurement on this detector. Therefore, the DAF predictions are compared to the unbiased residuals of the GBL track fit.

In detail, the GBL fit yields an unbiased residual width of (7.7 ± 0.1) µm whereas the DAF implementation yields a biased residual width of (8.9 ± 0.1) µm as shown in Figure 8.8. The other direction yields consistent results.

(a) Residual distribution on the second telescope module with the GBL track fit.

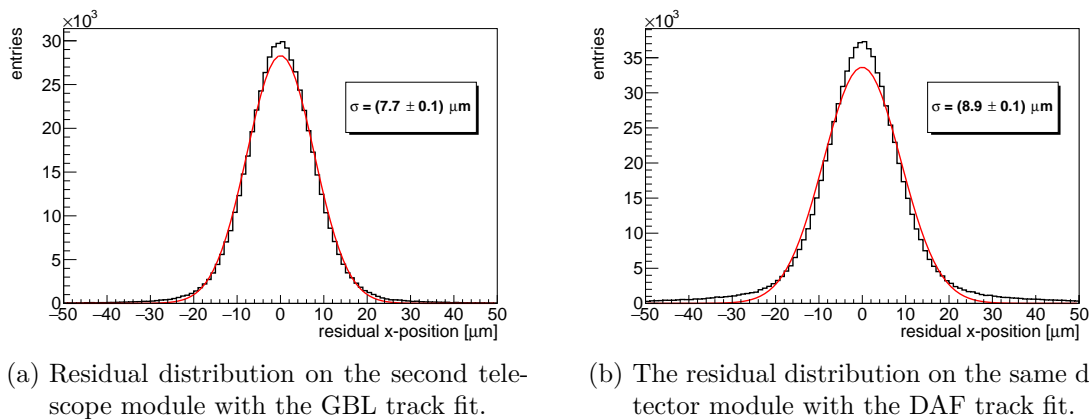(b) The residual distribution on the same detector module with the DAF track fit.

Figure 8.8.: Comparison of the residuals of the two track fits.

In order to compare the tracking resolution at the location of the DUT, the residual

distributions at one of the DUTs is investigated. The long pixel edge of the FE-I4 measuring 250 µm is used to fit a box function smeared with a Gaussian. The results are shown in Figure 8.9, where again the GBL and DAF algorithm are compared. The rather simple attaching of the DUT to the triplet pair, as discussed in the previous Section, does not reject all background. Still, the derived σ determining the rising and falling behaviour of the box was fitted to (7.7 ± 0.4) µm for the GBL track fit and (11.5 ± 0.3) µm for the DAF track fit.

(a) DUT residual distribution in x-direction with the GBL track fit.

(b) The same distribution but with the DAF fitter used.
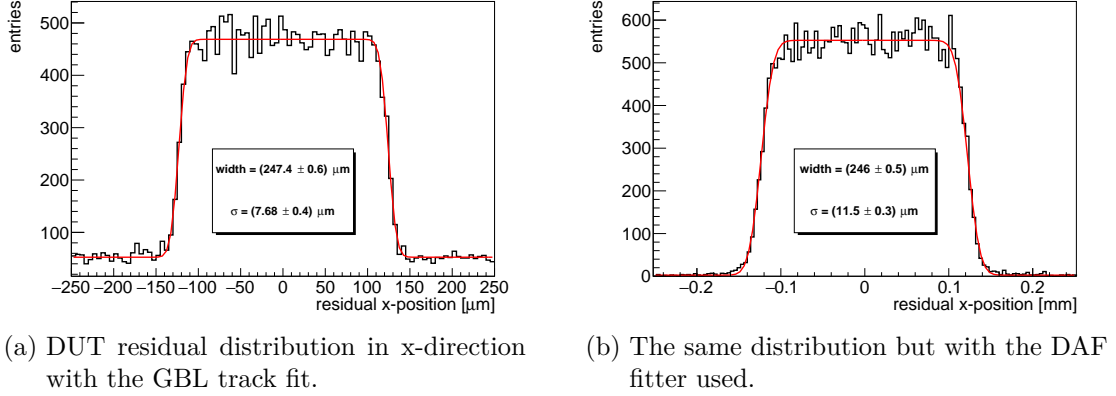
Figure 8.9.: The same comparison as in Figure 8.8 but here for the long pixel direction of a FE-I4 module.

Another important result of the GBL track fit is the obtained χ^2-distribution. To show results of a DUT included in the track fit, the χ^2-distribution obtained during alignment when the DUTs are included in the track fit, is shown. A cut on the degrees of freedom is imposed so that only χ^2-values for ten degrees of freedom are shown. This reflects the case where there is one measurement from the DUTs contributing to the track fit during alignment. The results are shown in Figure 8.10.

Indicated by the red fit is again a χ^2-distribution with ten degrees of freedom, i.e. the theoretical expectation. There is an even more pronounced tail towards high χ^2-values. As discussed, this is due to more extreme scattering events which are present as non-Gaussian tails in the angular scattering distribution. However, in addition, the FE-I4 is a hybrid pixel sensor. Tracks propagating through the interconnecting bumps of at least one of the DUTs in the beam will experience significantly more material than described by the scattering algorithm.

As a comparison, the run was also reconstructed with the DAF fitter and entries with ten degrees of freedom selected. Results are shown in Figure 8.11, where a large deviation between the expected tail in the χ^2 distribution and the obtained results from data are visible.

It must be noted, that the DAF fitter is typically not used to incorporate DUT measurements and also lacks the functionality to input precise measurement uncertainty, hence the deviations in the high-χ^2 can also be attributed to an incorrect DUT de-
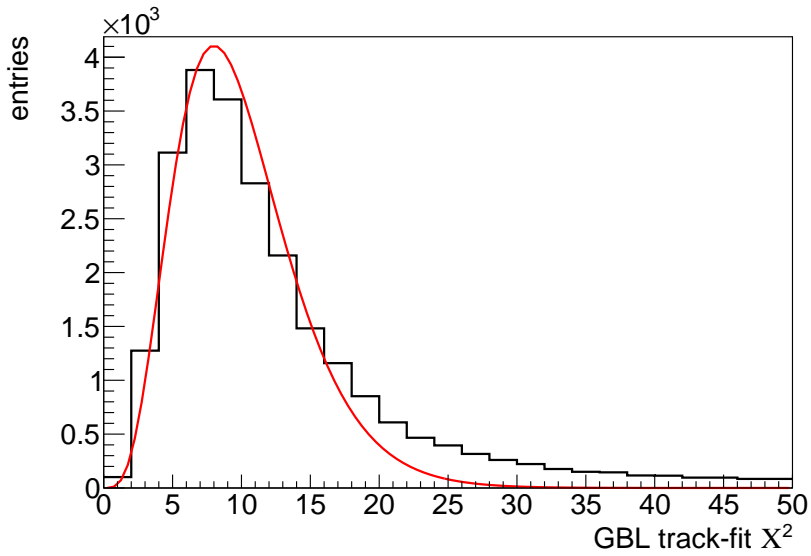
Figure 8.10.: The measured (black) and expected (red) χ^2-distribution for events with ten degrees of freedom for the GBL track fit.
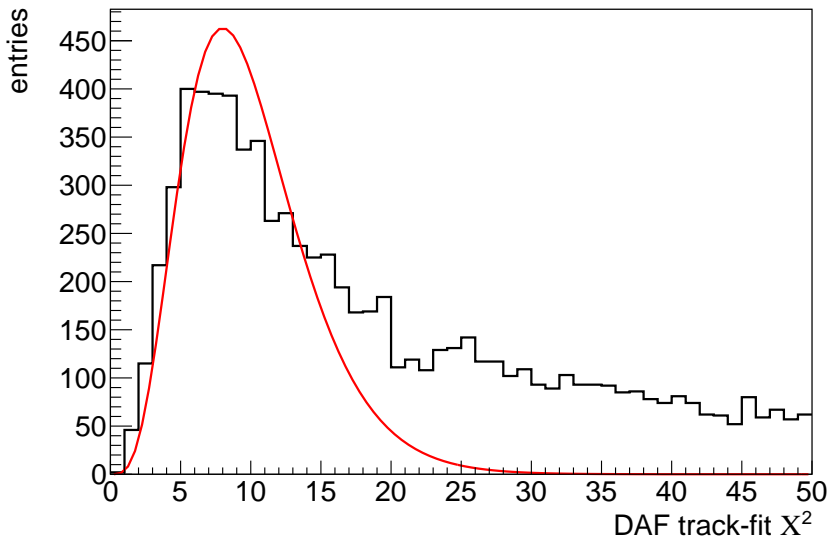

Figure 8.11.: The measured (black) and expected (red) χ^2-distribution for events with ten degrees of freedom as obtained by the DAF track fit.

scription. Summarising, the GBL track-fit given in Figure 8.10, yields a reasonable χ^2-distribution with expected deviations due to non-Gaussian scattering, whereas, implementation details limit the accuracy of the track fit when using the DAF fitter, as seen in Figure 8.11.

It is important to keep in mind that the obtained estimates for the tracking resolution, i.e. the σ from the box-fit, and χ^2-distributions, are for this sample and set-up. Different locations of the sensors, different tunings and other beam energies will impact the actual performance. It is, therefore, necessary that these parameters are validated at each test beam campaign.

In order to validate the tracking algorithms even more precisely, a Monte-Carlo framework is used. For the purpose of further studies, Allpix^2 has been modified to incorporate needed functionality, as described in the next Section.

8.2. Allpix^2

According to the project's website, the Allpix^2 (or *Allpix-Squared*) framework [91] is a *"Generic Pixel Detector Simulation Framework"*. It uses the Geant4 simulation software [92, 93] for the simulation of particle interactions with matter and has individual modules to simulate various properties, e.g. charge diffusion and drift, signal creation and detection, digitisation, and many other processes.

One of the aims of Allpix^2 is simulation of set-ups used in test beam measurements. This makes it a useful tool to study the impact of the set-up on a simulation level as well as to validate the reconstruction and analysis techniques. In particular, Allpix^2 is a very powerful tool, if the Monte-Carlo truth information, i.e. the true particle's trajectory within the telescope, is exported and used to compare to reconstructed data. Prior to the modifications and extensions to Allpix^2, this was possible to a certain extent. The true spatial positions of deposited charge were available in the framework. However, neither the interface to EUTelescope, i.e. the export into the LCIO format, nor the full track information, i.e. which track a charge deposition in the material belongs to, was correctly implemented.

8.2.1. Objects and Terminology

An Allpix^2 *MCParticle* (for: Monte-Carlo particle) is the representation of a charged particle crossing a detector. In particular, the information it holds are the entry and exit points into the detector volume.

The Allpix^2 *MCTrack* is the supplementing class which has been introduced in the context of this thesis when expanding the functionality of Allpix^2. It stores information on the production process of a particle's trajectory, i.e. the physical process which created it, the volume and position where this happened, the initial and final kinetic as well as total energy.

Both these classes, the *MCParticle* and *MCTrack*, carry links to other objects. Most important are the links between *MCParticles* and their associated *MCTrack*. Furthermore, *MCTracks* can have a parent *MCTrack* from which they originate. Combining this information, the full track hierarchy can be obtained and reconstructed. This is depicted in Figure 8.12. It can be seen that Allpix *PixelHit* objects are also in the shown hierarchy. They are the result of the digitisation step and correspond to detected charge

by a pixel inside a detector. Both the *MCParticles* and the *PixelHits* are associated with a given detector, unlike the *MCTracks* which are not linked to any detector.
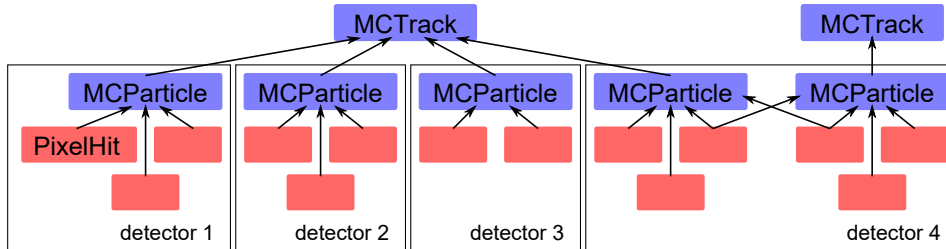

Figure 8.12.: The Monte-Carlo objects in Allpix[2] reflecting the truth trajectory of a particle passing through the detectors. The red boxes indicate *PixelHit* objects.

The picture shows a case of a four detector set-up, where a single particle passes through all the detectors and in the fourth detector a secondary particle is created. This explains why a single PixelHit can be associated with multiple MCParticles. A delta ray can, and probably will, deposit energy in the same pixel cell as the originally passing particle[1].

Disentangling the PixelHits which are linked to multiple MCParticles is difficult. For example there could be the hypothetical case where two particles each induce four arbitrary units of charge. The detector is tuned to only detect charges higher than six units. In this case, the detector will detect a hit only due to the interplay of both particles. As a result, there is no definition of what fraction of a PixelHit contributed to a given MC-Particle, just the information that there was a contribution. This leads to ambiguities in the truth level description.

In addition to the implementation of the MCTracks and mechanism to correctly extract and track the information from the Geant4 simulation, test cases have been implemented to validate the MCTrack within Allpix[2] also in the future.

8.2.2. LCIO Data Export

In addition to implementing the mentioned features, the information needs to be extracted into EUTelescope. The example sketched in Figure 8.12 is again shown in Figure 8.13 focussed on how it is reflected in LCIO. Aside from the collection for the raw data (shown at the bottom), there are three collections which are exported if the option to fully export the Monte-Carlo truth data is enabled.

The MCTrack objects are exported as LCIO::Track objects and stored within a collection in the output file. Contrary to the situation in Allpix[2], in LCIO the tracks have a connection to the LCIO::TrackerHit objects, and not vice versa. The LCIO::TrackerHits

---

[1]Technically, the production of a delta electron by definition is already a deposition of energy and the key issue is to what MCParticle this is associated. The net physical effect of an induced signal is unaffected by this.
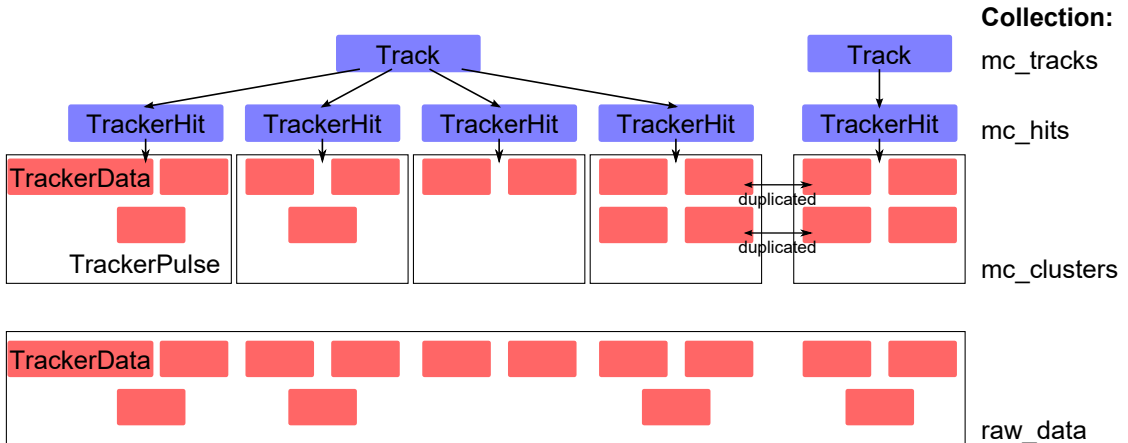
Figure 8.13.: Reflection of the Allpix^2 objects within LCIO. See Table 5.1 for an overview of the used LCIO objects.

correspond to the MCParticles and are stored in a collection corresponding to truth hits. All PixelHits which link to the same MCParticle are considered to be a cluster. If a PixelHit links to multiple MCParticles, the PixelHit is assumed to belong to both clusters. The PixelHit objects are represented by LCIO::TrackerData objects. In order to represent truth clusters, PixelHits with multiple MCParticles are duplicated in this process and are stored together with an LCIO::TrackerPulse collection in the LCIO output file, representing the cluster.

Ultimately, the unmodified and unsorted PixelHits are also exported as LCIO::TracerData in an LCIO collection, corresponding to the detector's simulated response. This last collection is also exported if the option to export Monte-Carlo truth data is disabled, as it is the simulation output.

Additionally, the GEAR file with the precise alignment is exported.

8.2.3. Alignment Validation

In order to validate the new alignment schemes, discussed in Section 7.2.2, Allpix^2 was employed. A single Mimosa26-like detector, displaced 10 mm along the beam axis away from the coordinate systems origin, was used. No further displacements or rotations were set, i.e. all were at 0. Allpix^2 allows to specify an uncertainty in the alignment. In particular, a σ of a Gaussian distribution can be provided and a smearing with a randomly drawn perturbation value from that distribution is simulated[2]. Rotations were smeared with $\sigma_{\mathrm{rot}} = 1°$ and displacements along the x- and y-axis with $\sigma_{\mathrm{x/y}} = 50$ µm. For the z-axis (beam axis) a $\sigma_{\mathrm{z}} = 1$ mm was used. This resulted in a rotation of $\approx -1.2°$ around the x-axis, $\approx 1.1°$ around the y-axis and $\approx 0.6°$ around the z-axis.

---

[2]This is specifically useful as it not only reflects an actual set-up more realistically, but also as the case of perfectly aligned sensors will (even more) correlate measurements on the different sensors.

To validate the alignment, the Monte-Carlo truth hits in the exported LCIO file (labelled *mc_hits* in Figure 8.13) were compared with hits derived from the simulated detector response (*raw_data*). In order to do this, the simulated pixel hits were clustered, the hit positions derivation performed and ultimately they were exported into the global coordinate frame using the exported GEAR file. The Monte-Carlo truth hits are already in the global reference frame, hence the comparison of the reconstructed and transformed hits with the Monte-Carlo hits will validate the alignment.

In order to verify the impact, i.e. to show that the smearing is sufficient to detect a deviation if the alignment is carried out wrongly, a GEAR file with the rotations set to half their nominal value as well as one where they were set to $0°$, was used. A sample with a single hit in each event is used and a pre-selection to discard events in which a secondary particle is produced is applied. A narrow beam, centred in the middle of the sensor, with no divergence, is simulated. The differences between the reconstructed and derived hits and the Monte-Carlo hits are shown as residuals in Figure 8.14 for the x-direction.

In Subfigure 8.14a the alignment with the angles set to $0°$ are shown, in 8.14b the angles are set to half their nominal values and ultimately the correct alignment is used in 8.14c. The nominal pixel pitch of 18.4 µm is indicated in the plots as well. In first approximation, a boxed distribution with the pixels pitch is expected, in particular if there is no beam divergence and there are mostly single-hit clusters.

As expected from the isotropy in x- and y-direction, the distribution for the y-direction is similar and shows consisting results. The z-direction distributions are shown for the case of all angles set to $0°$ as well as the correct alignment. Subfigure 8.15b shows that the z-coordinate is consistently set by the simulation and reconstruction.

Aside from validations of the alignment, the newly implemented functionality has been used in the context of a bachelor's thesis to start validating track finding routines within the GBL implementation [94]. Also, Allpix[2] simulated data have also been used to verify processors like the noisy pixel determination.

8.3. Self Testing Software Repository

In order to ensure that the EUTelescope software correctly builds and produces the expected output, several mechanisms have been implemented. An important tool to help verifying the build process is the Docker[3] framework. Docker offers virtualisation within the Linux operating system. This allows to run independent so-called containers within the user's operating system. It is especially useful as it allows to execute the installation routine on different Linux distributions, which vary in the libraries installed by default.

This not only allows to test the installation routine on various distributions, but also the self tests can simultaneously run on multiple distributions, each with different compiler versions. As a consequence, it is ensured that modifications implemented by

---

[3]https://www.docker.com/

(a) Residuals in x-direction for all rotations set to 0°. Note that the y-axis is offset and starts at 60.

(b) The same distribution but with the angles set to half their true value.
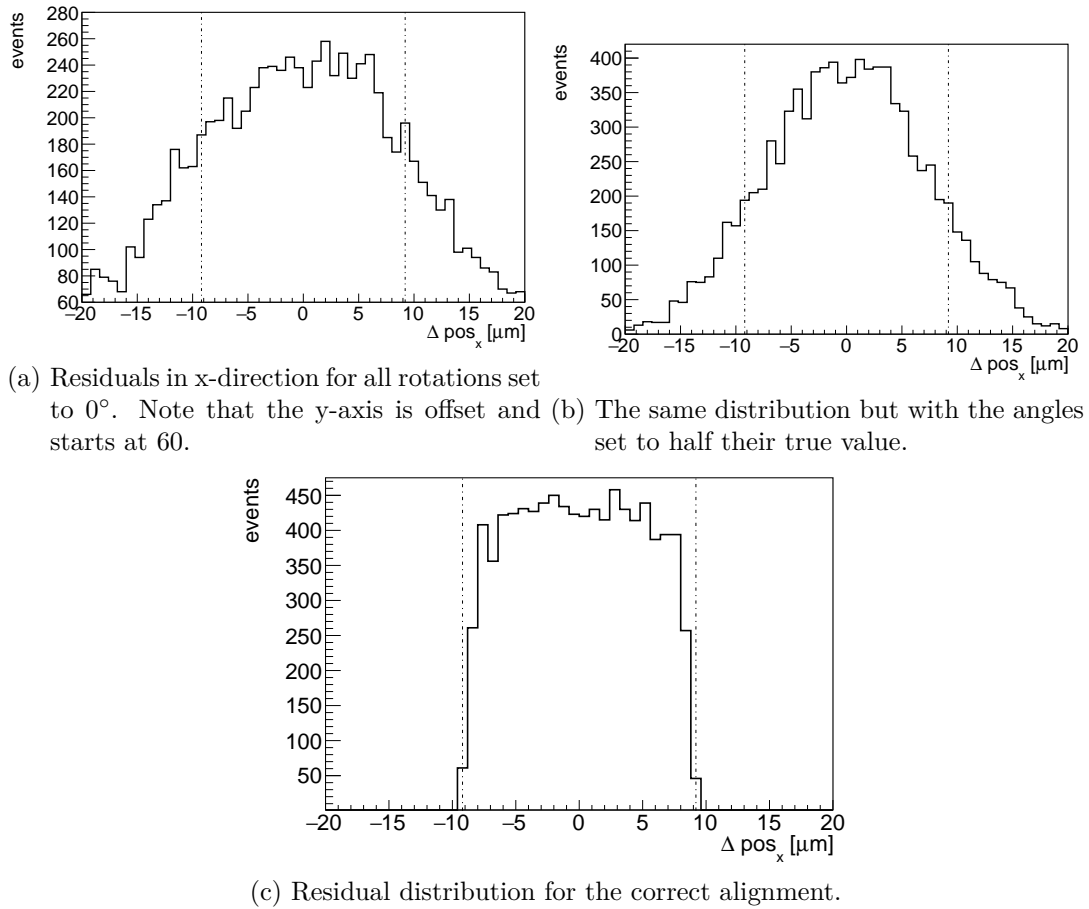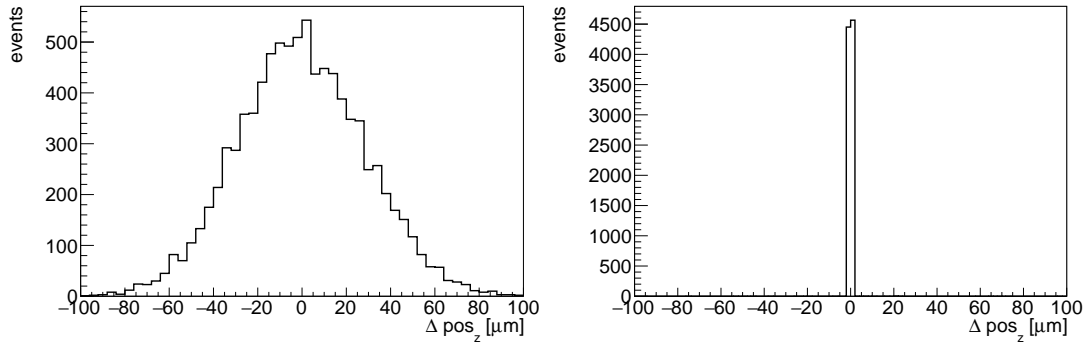

(c) Residual distribution for the correct alignment.

Figure 8.14.: Residual distributions showing the difference in x-position of the reconstructed and simulated Monte-Carlo hit. Indicated by a dashed line at $\pm 9.2\,\mu m$ is the pixel dimension.

a developer on their system also feature compatibility with older, but still supported, compiler versions as well as comply to the more modern standards which are enforced by novel and up-to-date compilers, available within rolling release[4] distributions.

Within the list of available Docker builds, the most common package managers are represented. This verifies and ensures that external libraries are available and functioning with these package managers. Different versions of Ubuntu, Fedora, ScientificLinux, CernCentOS, Debian, openSUSE, and ArchLinux are available and used to validate that changes to the EUTelescope code base do not break functionality.

Aside from the simple check if the project compiles, the cmake framework provides tests which are executed after a build. These tests will start an EUTelescope reconstruc-

---

[4]A rolling release model is a distribution with a relaxed version scheme and instead of releasing tags or versions, small but frequent updates are issued.

(a) Residuals in z-direction for all rotations set to 0°.

(b) Residual distribution for the correct alignment.

Figure 8.15.: The residual distribution for the same data sample, but in z-direction.

tion job and validate parameters like the obtained number of clusters or the derived alignment constants. If a deviation is observed, the test reports a failure. These tests are particularly interesting, as changes are expected to impact certain parameters but leave others unchanged. For example, if a fix in the hit position derivation suddenly modifies the clustering routine, the changes have modified more than expected and probably have an undesired impact.

In order to automate these tests, a feature of continuous integration, the previously mentioned Docker images are used. When a user requests to merge code back into the hosted repository, an automated series of tests is started. EUTelescope is rebuilt on all available Docker images and the mentioned cmake tests are executed.

If compilation or a single test fails, this is reported to the repository maintainers. As only features which are actively monitored are tracked by this mechanism, it is important that new features are implemented along with a test case to validate this new feature. In Figure 8.16 a screenshot of the web-interface of the CI server is shown with succeeding and failing builds. *Pull-requests*[5] (PR) #445 and #446 passed, whereas #444 is continuously failing. The multiple entries for PR #444 are explained by modifications made to a PR that trigger the build again.

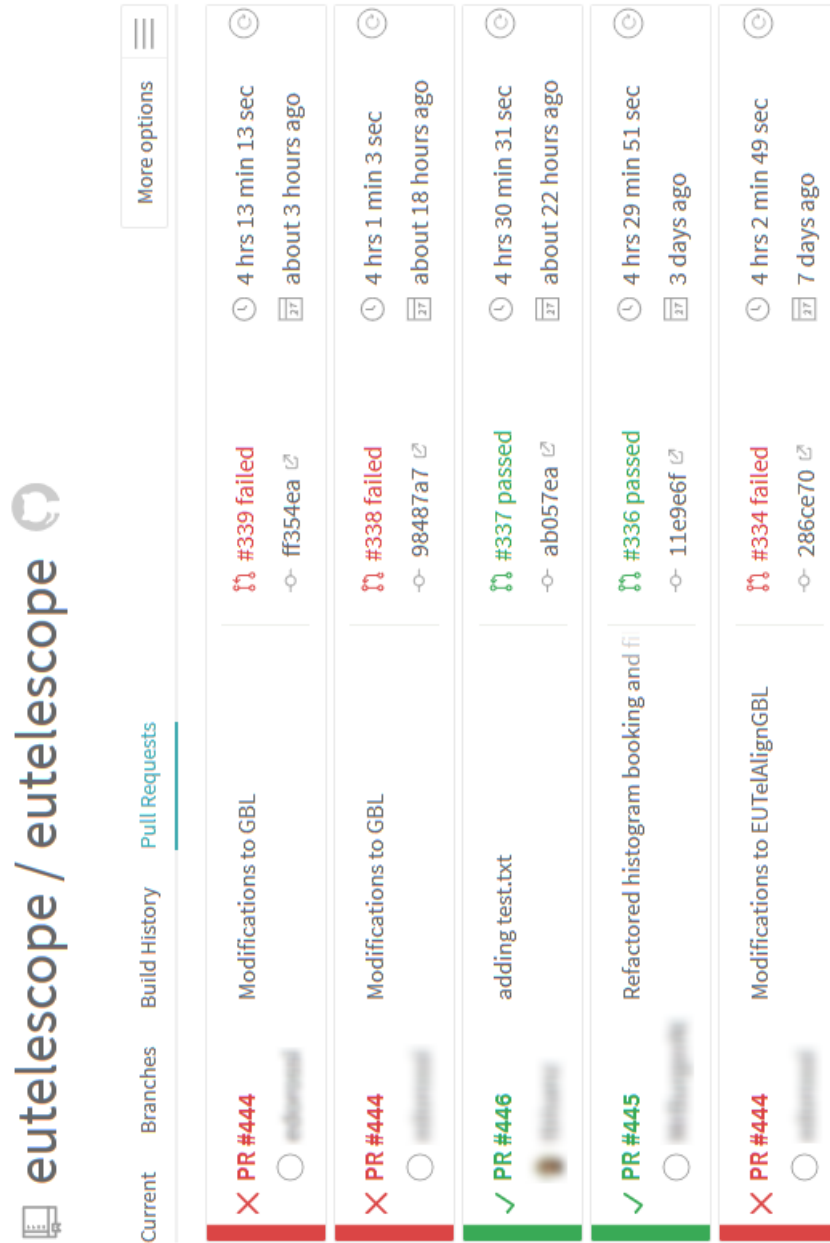[5]Those are the requests of a user to merge back code.

Figure 8.16.: Web-based feedback for repository maintainers of the build status. (Usernames are blurred out)

In-Time Studies

The particles at test beam facilities arrive with a (pseudo) random timing. While beam clocks (for the DESY II accelerator) or spill signals (from the SPS accelerator) are available, a generic solution to perform in-time measurements is preferred. As discussed in Section 5.1.2, the DESY II beam-clock is continuously resynchronised with the power grid. Therefore, a measurement set-up without using these signals has been established.

The FE-I4 as well as the upcoming RD-53(A) read-out chips will continuously record data and internally store them. This is necessary as the high-level trigger decision, i.e. the decision to read-out an entire event, is only made in retrospect in the ATLAS detector. As the data need to be associated with a single bunch-crossing, these acquisition windows are 25 ns long. They are the so-called *LV1 accept windows* or *LV1 accept bins*. Once the trigger decision is made, the detector is triggered to send the recorded data from the past to the off-detector DAQ systems.

In test beam measurements, typically 16 consecutive LV1 bins are processed. This gives rise to the different LV1 windows for a single TLU trigger. In normal operation, the ATLAS detector will only send out data from a single LV1 bin.

9.1. Motivation and Aim

In order to test the functionality of the FE in the same operation condition as in the detector, only a single LV1 bin must be processed. The measurement of the efficiency within a single 25 ns window is important as this is the mode the FE is operated within the actual ATLAS detector, also with the ITk upgrade. At the test beam facility, the particles will not arrive at a precise time, but randomly. This is a problem, if only a single LV1 bin is processed, as the particles might arrive too late to be correctly associated with the LV1 window. Consequently, a new measurement technique is necessary.

The basic concept of a measurement technique to circumvent this problem is that only triggers, which arrive in a limited time window within a precise location relative to the FE clock, are accepted. This time window will be referred to as the gate, its duration as the gate length t_{gate} and the relative position to the FE clock as the delay time t_{delay}. The FE clock runs at 40 MHz, corresponding to the LHC beam clock, i.e. a 25 ns period.

It is desired to use a relatively short gate, though, this also reduces the fraction of triggers by $t_{\mathrm{gate}}/25$ ns. A gate of $\mathcal{O}(\mathrm{few\ ns})$ is wanted, which is about a magnitude lower than the 25 ns period but will still provide enough triggers for runs to complete within a reasonable time scale.

In Figure 9.1 this idea as well as the temporal quantities are indicated. It must be noted, that the t_{delay} in the schematic is relative to the rising FE clock. In the actual measurement, the reference point is arbitrary and only differences in t_{delay} are a precisely defined quantity.
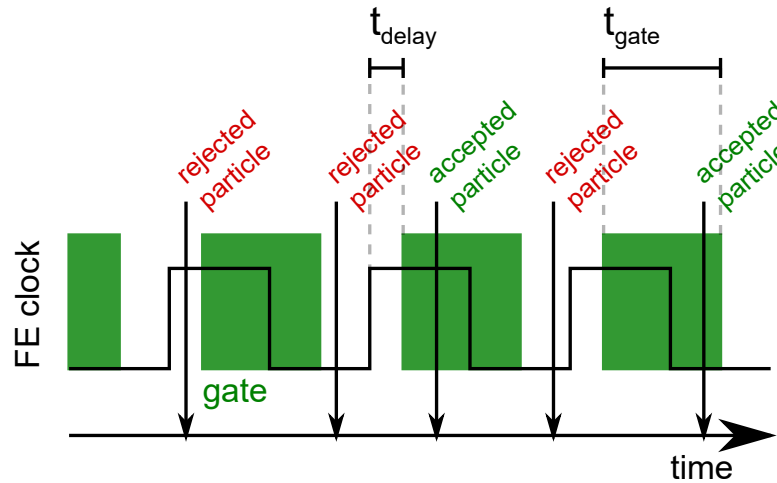
Figure 9.1.: Principle of an acceptance window to only select particles which arrive within the gate. The vertical arrows indicate incident particles.

9.2. Experimental Set-Up

In order to achieve such a set-up, the gate signal has to be derived from the FE clock. The initial idea, to directly use this signal as input into the TLU, does not work, as the TLU's discriminator will latch in a high state much longer than the gate period, i.e. it will continuously stay at a high level[1]. For this reason, the gate and the signal of a PMT need to be processed externally, before triggering the TLU with a coincidence on the two signals.

---

[1]This can be seen in Figure 9.7 and 9.8 where on channel 3 (magenta) the approximately 0.5 μs time window the TLU latches onto a single trigger is shown.

The initial idea was to use high-speed NECL[2] components. This approach failed for several reasons. A circuit using high speed NECL components might be improved for further future measurements. This approach is discussed in Appendix D.

A NIM[3] set-up was developed to perform the measurement, as schematically depicted in Figure 9.2. The clock signal from the MultiIO3 board is converted into NIM logic levels and a coincidence unit is used to shape the signal into a single-shot pulse of approximately 5 ns duration. Afterwards, the signal is delayed by a variable delay unit and routed to another coincidence unit. The PMT signal is discriminated by a CFD, the TTL output of the constant fraction discriminator (CFD) is converted into a NIM signal which is then shaped in the same fashion into a 5 ns pulse by a coincidence unit before being fed to the final coincidence unit which combines the processed clock and PMT signals. The output of the coincidence is led to another delay line and is then used as the TLU input signal. As the TLU input has an input resistance of 50 Ω, a NIM high state results in a negative voltage pulse at the TLU input. Hence, the TLU's discriminator for this channel can be configured to trigger on this pulse.

In this particular set-up, the following hardware was used: a CAEN Model N 89 NIM-TTL-NIM Adapter, a CAEN Model N 405 3 Fold Logic Unit, a CAEN Model N 108 Dual Delay, and a CANBERRA Model 2128 Constant Fraction Discriminator.
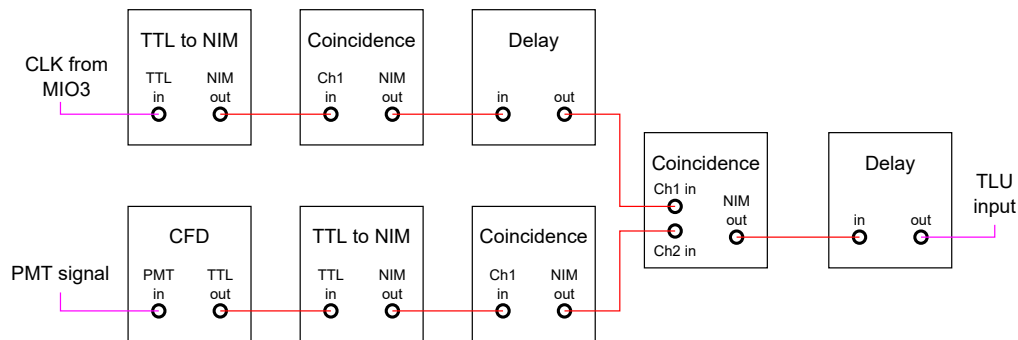
Figure 9.2.: Schematic of the NIM set-up used for deriving a trigger signal for the in-time measurements.

In Figure 9.3 the ideally derived signals are shown. The top part is the compressed sketch shown in Figure 9.1. The signal labeled *gate* is the derived gate signal. In the circuit in Figure 9.2 the coincidence unit in the upper signal path, i.e. the one processing the CLK from the MIO3 board, is responsible for the shape, i.e. duration, of the gate signal. The delay unit following the coincidence unit is responsible for the shift. If the cables and NIM units are correctly terminated and no noise or other effects distort the signal, the ideal delay unit will only shift, but not shape the signal. As the signal is already shaped but also delayed, the *gate* shows the case where we probe the signal right in front of the coincidence unit on the right, i.e. the one where the two signals are combined.

---

[2]Negative emitter-coupled logic.

[3]Nuclear Instrumentation Module.

The signal with the label *pmt* shows the lower signal path, also right in front of the just mentioned coincidence unit. Ideally, the original pulse from the photomultiplier tube is very short and the discrimination and shaping circuitry will not change this. Therefore, the signal arriving at the coincidence unit is depicted as a delta peak.

Assuming a perfect response of the coincidence unit, the output of the coincidence unit processing both input signals will be a perfect logical-AND, i.e. if the *gate* signal is in the high state (indicated by the highlighted duration in green) and a delta peak is at the second channel, the output will go into a high state. It will remain in the low state in all other cases. This is shown as the signal labeled *coincidence.*
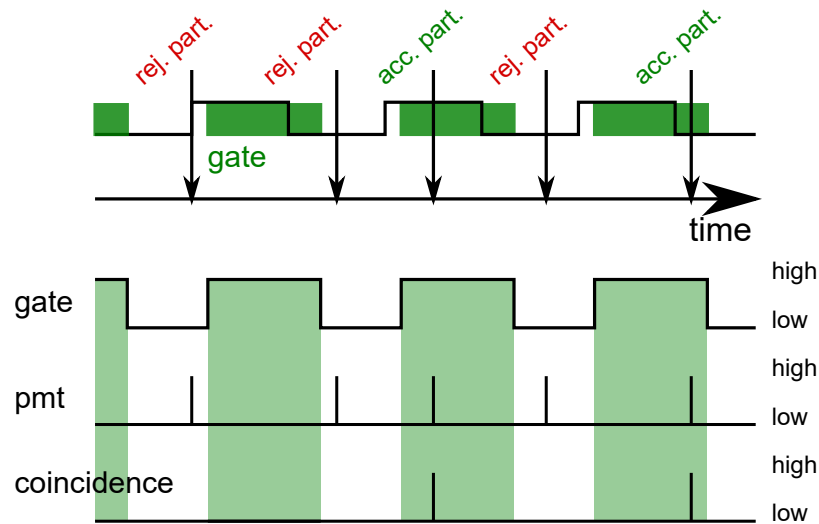
Figure 9.3.: Ideal signals after shaping and delaying, in front and after the final coincidence unit.

However, this does not reflect the actual set-up. The signal shaping is done with the CAEN Model N 405 3 Fold Logic Unit for both signal paths. The logic unit has a minimal duration output of 6 ns with a rise- and fall-time of 2 ns according to the specifications. To trigger the coincidence unit, a 3 ns minimum overlap of the input signals is required. These are the nominal reported values, the actual duration of the gate has been determined via measurements with an oscilloscope.

This modifies the scenario shown in Figure 9.3. The minimal overlap of the final coincidence as well as the shaped PMT signal will impact the actual gate. As the theoretical pulse durations are at 6 ns and the minimal overlap is 3 ns, the final gate length would turn out to be 6 ns again. A more realistic scenario is shown in Figure 9.4.

The ideal PMT delta peaks (indicated in red, dotted) have a pulse duration in the order of the gate length, which again is in the order of the required overlap for the logic.

There are several factors which can have impact on these properties. The stated values are performance characteristics the vendor guarantees to meet, depending on the actual used components and their variation. The attenuation will decrease the signal amplitudes, attenuation will be due to cables but also by the attached NIM units. Improper
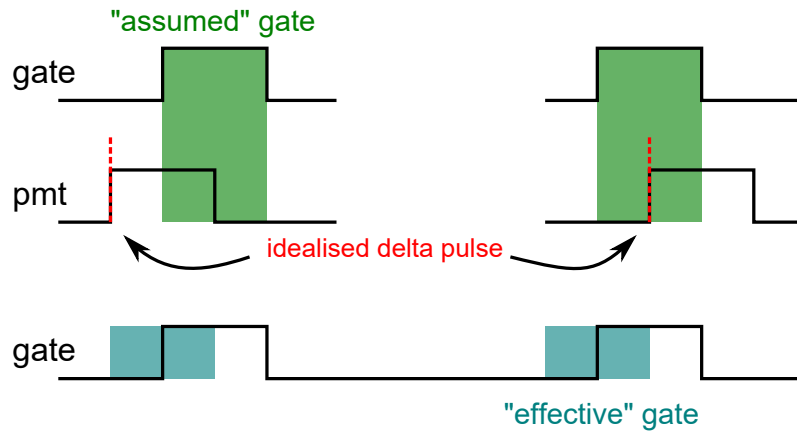
Figure 9.4.: Gating with PMT long duration PMT pulses and required overlap time.

signal termination will cause reflections which also impact the signal. An attached oscilloscope which is used to monitor the signals can either improperly terminate with a high-input impedance or also terminate with 50 Ω and hence drain current from the signal. Ultimately, the non-zero rise- and fall-times will impact the measurement.

9.2.1. Set-up Validation

In order to validate the properties of the set-up, several measurements were performed. In particular, the gate duration was investigated. It was ensured that adding the delay would not modify the gate signal. The measurements were carried out with a Tektronix DPO 4104 digital phosphor oscilloscope.

If the addition of extra delay would modify the gate signal, then the different points at various delays could not be compared to each other. The most probable error which could be introduced would be a changing gate length for different delays. This would bias the final results, as for example the measured efficiencies would be for different gate widths. Other undesired effects could be shifts not according to the specified delay or attenuation of the signal because of the extra cables.

The first measurements, given in Figure 9.5, show the gate length after the coincidence unit in the signal path of the clock signal. Channel 1, the blue signal, shows the NIM logic signal. A logical one in NIM-standard fast negative logic signals is defined as -12 mA into 50 Ω load (which is the lowest current the input needs to respond to), corresponding to a voltage of -0.6 V. In addition, the clock signal to the FE is probed at the USBpix adapter card. This signal is shown on channel 4 (magenta) and simply is used as a reference signal to trigger on. In particular, this also allows to validate if the amount of the shift is correct.

The measurement functionality of the oscilloscope is used to quantify the negative width of the signal. It was measured to be (5.65 ± 0.03) ns. The same measurement was
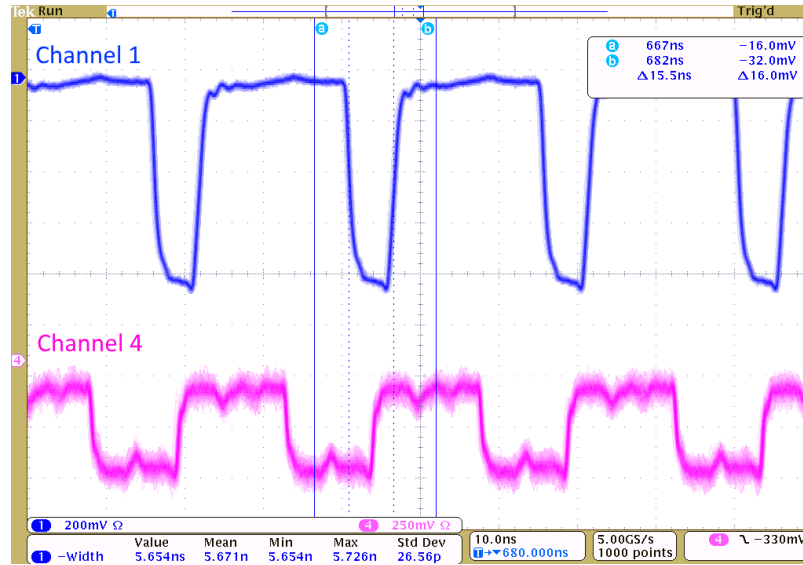
Figure 9.5.: Validation of the output pulse shape and measurement of the negative pulse width.

performed with 5 ns and 11 ns delay. The results are given in detail in Appendix E. Summarising, all delays give consistent shifts with respect to the clock signal to the FE, and yield gate widths of (5.61 ± 0.02) ns and (5.66 ± 0.02) ns, respectively.

In order to quantify the gate duration, as discussed in the previous section, again an oscilloscope based measurement was used. In Figure 9.6 the measured signals are given. Shown in channel 2 (red) is the gate signal right in front of the final coincidence unit, which is repeated every 25 ns. Channel 1 (blue) shows the shaped signal in the PMT path, also right in front of the final coincidence unit. Channel 3 (green) is the discriminator output of the TLU, i.e. it is the signal if the TLU registered a hit. A falling slope trigger is used on the TLU's signal to filter only events where there was coincidence in the clock and PMT signal. A measurement of the delay between the falling edge on channel 2 and the next falling edge on channel 1 was carried out. The measurement was only carried out between the on-display cursors and not over the entire acquisition.

The measurement revealed a range of -1.4 ns to 2.6 ns for the measured delays, this yields a total length of 4.0 ns for the *effective* gate width, as sketched in Figure 9.4.

The final delay unit in Figure 9.2 is there to correctly set the time the trigger from the USBpix card is sent to the FE. As in test beam, there is no central unit for triggering, timing, and control, the issued trigger is typically not synchronised with any other part of the set-up. This can lead to the case in which the events, which are locked to the FE clock, are read out with a 25 ns delayed trigger. If this happens, the hit will appear to be further in the past, i.e. seem to have arrived earlier, reflected by a shift in the LV1 distribution.

In Figure 9.7 the case of this glitching behaviour is shown. Channels 1 to 3 show the same quantities as before. Channel 4 (magenta) probes the differential command signal
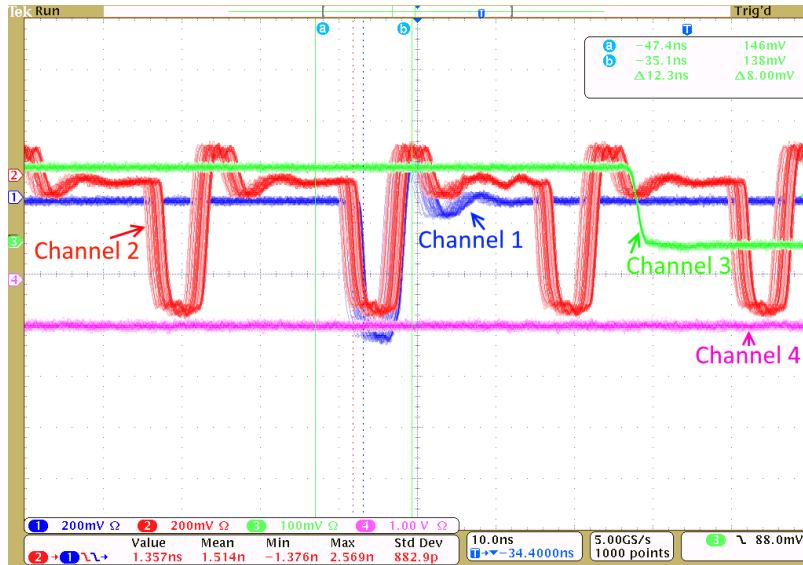
Figure 9.6.: Measurement of the gate length by probing the minimal and maximal delay between the two input pulses.

on the USBpix BIC adapter card sent to the FE. The so-called *LV1 trigger command* sent to the FE is 11101. This corresponds to the command being probed on channel 4. In Figure 9.8 the delays have been adjusted to ensure that the trigger is sent in a correctly timed manner and the LV1 command is even more clearly visible.

9.2.2. Samples Used

For this analysis, two FE-I4 modules were used. The reference module is a standard IBL style module, but with a quad chip sensor assembly. In particular, the Göttingen **W13** module acts as a reference sample. It has been shown, that these modules have an efficiency above 98% [95]. During the intime-efficiency studies, an overall efficiency above 97.5% was recorded. However, the efficiency quoted here is the module efficiency in the fiducial area. The module efficiency is not limited to a *good area*, which is the efficiency typically quoted. The good area is selected by removing areas with not-responding or noisy pixels. The W13 module features a sensor with a thickness of 200 µm, a p-type bulk with n-type pixel implants, and an FE-I4B read-out chip bump-bonded to the sensor. It is unirradiated, hence it is operated at -80 V bias voltage, exceeding the depletion voltage, which was determined to be slightly above -10 V [96], by a large margin. As the DUT is a single FE, only one of the four FEs on the quad chip module is used, it is the FE with the global address (GA) of 2.

The module under test is the **ADVACAM NP-100-5-3B**. Unlike the W13 it features n-in-n pixels and has a sensor thinned down to 100 µm. This poses a challenge to the attached FE-I4B, as merely half the charge is deposited, compared to IBL style modules, like the W13. This batch of ADVACAM modules was used to investigate certain types
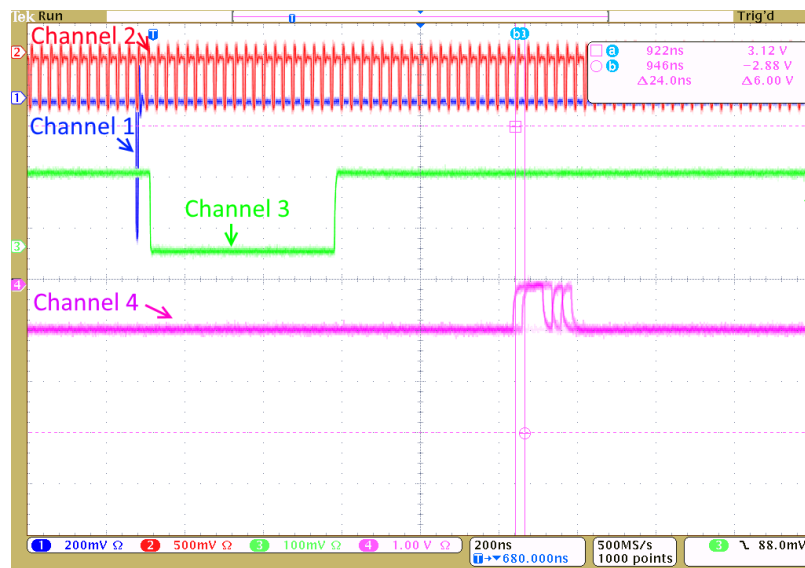
Figure 9.7.: Wrongly timed trigger being sent to the FE by the USBpix board.
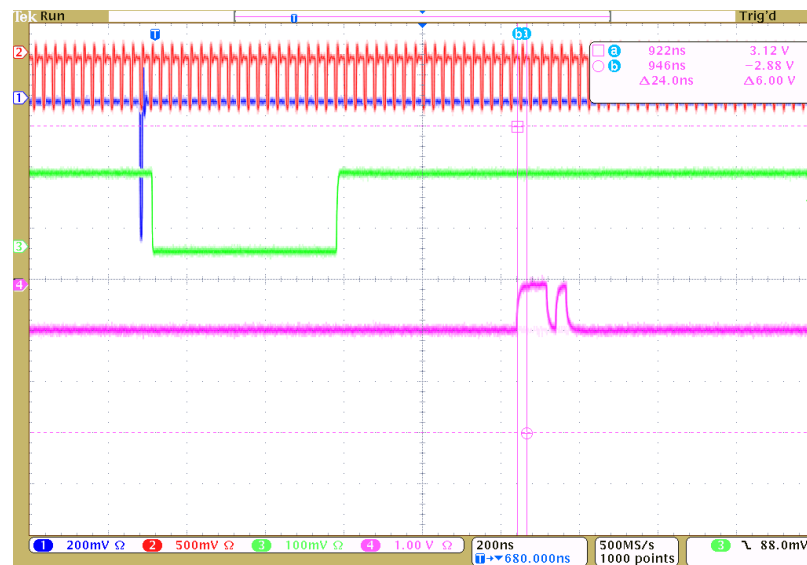

Figure 9.8.: Introducing (or reducing) an additional delay ensures a proper trigger condition. Clearly visible is the LV1 trigger command sent to the FE on channel 4 (green).

of active- and slim-edge designs and to probe various types of under-bump-metallisation (UBM). The ADVACAM NP-100-5-3B module exhibits a highly increased noise in a

ring structure around the sensor edge, an effect most probably due to a deficient UBM. Hence, this area is excluded for the analysis. An active area of 8 mm × 8 mm centred inside the sensor is used.
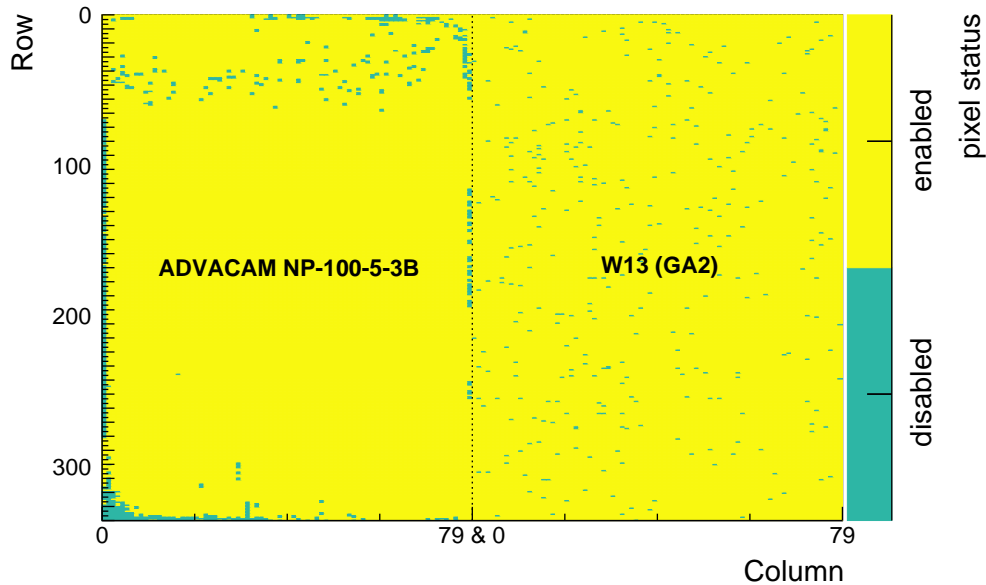
Figure 9.9.: The status of the individual pixels on the reference module W13 and the DUT ADVACAM.

In Figure 9.9 the status of the pixels is shown. Known noisy pixels were disabled during data acquisition. They were determined by performing an analogue scan, a scan where a test charge is injected into the analogue part of the pixel cell and the response of the pixel is recorded. Noisy pixels detect more hits than the number of times the charge is injected.

On the ADVACAM module, the disabled noisy pixels, arranged in the circular ring around the edge, are due to the faulty UBM. The W13 features noisy pixels homogeneously spread over the pixel matrix.

9.3. Reconstruction and Analysis

The collected data are reconstructed with EUTelescope. No hardware configuration, i.e. FE parameter, is used to limit the accepted LV1 windows. Hence, 16 subsequent LV1 windows are read out. Data are collected with a MIO3 board and a BIC adapter card. The default, newly implemented, STcontrol producer, as described in Section 6.2, is used for DAQ.

Prior to clustering, noisy pixels are removed. A newly introduced processor allows the selection of pixel hits with a specified LV1 value for a given DUT. Furthermore, the possibility to set a cut on the charge, i.e. ToT, was implemented. A flow diagram of the
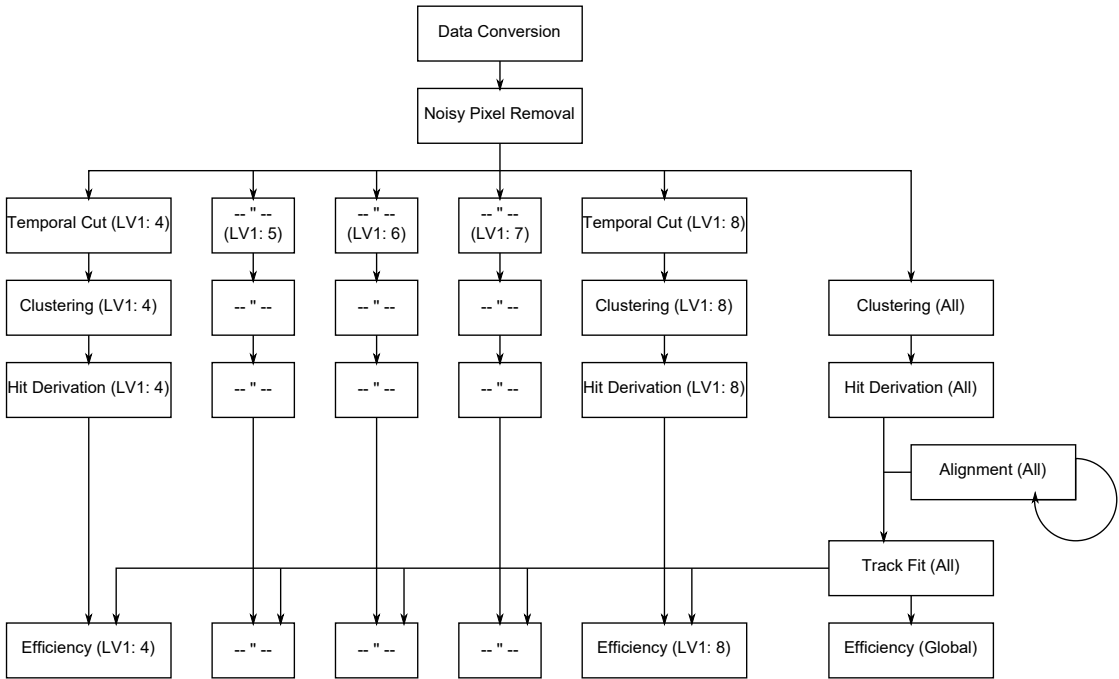
Figure 9.10.: The strategy used in the reconstruction and analysis of the in-time efficiency measurements. On the left, the same reconstruction steps are performed for the various LV1 bins.

analysis is shown in Figure 9.10, where the mentioned initial steps are the same for all the following reconstruction branches.

In order to only look at data from a single LV1 bin, raw data collections for the various LV1 windows are created and processed. This is done by using the LV1 information and cloning entries with an appropriate LV1 value into the new collection. To obtain a precise alignment, also a collection without any cuts is processed and used (the branch on the right in Fig. 9.10). This is done to not reduce statistics or to bias the hits because of time-walk effects.

Every raw data collection is processed in the clustering and hit position derivation step. The unmodified collections are used for pre-alignment and alignment. Alignment constants are obtained in an iterative two-step alignment approach.

In order to validate the track fit, every run was analysed and a fit to the residual distribution was performed. This is exemplarily shown for the first run used in the analysis, run 4009 in Figure 9.11.

The parameters for the residual fit in x-directions are the following, where $H(x)$ is the unit step function and erf() the (Gauss) error function:
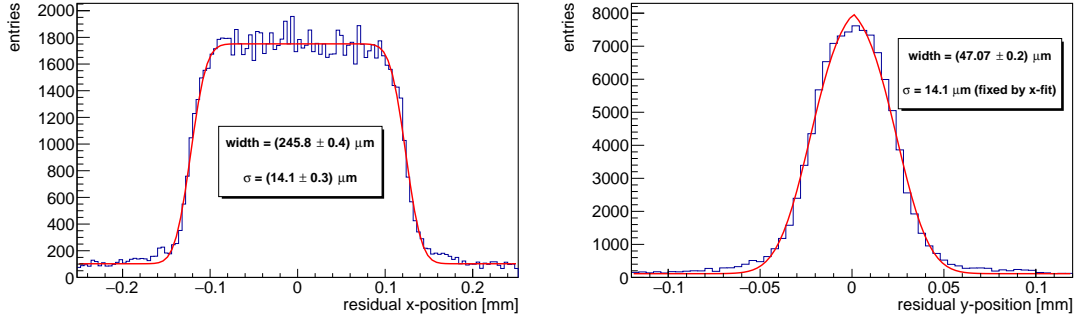
(a) Fit with Eq. 9.1 of the residual distribution in x-direction. The obtained fit results are given in the box.

(b) The same histogram and fit for the residual distribution in y-direction. The σ parameter is derived from the x-direction fit.

Figure 9.11.: Residual distribution after the fitter step used to validate the data. Shown is run 4009.

$$
\begin{aligned}
f(y_{\text{max}}, x_{\text{fall}}, x_{\text{rise}}, \sigma, y_{\text{offset}}; x) = & \frac{y_{\text{max}}}{2}\left[1 - \text{erf}\left(\frac{x - x_{\text{fall}}}{\sqrt{2}\sigma}\right)\right] H(x) \\
& + \frac{y_{\text{max}}}{2}\left[1 - \text{erf}\left(\frac{x_{\text{rise}} - x}{\sqrt{2}\sigma}\right)\right] H(-x) \\
& + y_{\text{offset}}
\end{aligned}
\tag{9.1}
$$

The obtained σ from the residual fit in x-direction is used as a fixed parameter in the y-direction fit. This is done because the residual distribution contains multiple sized clusters and hence the y-direction distribution is assumed to have a more pronounced peak in the centre due to the smaller pixel pitch in that direction. The σ however is merely dominated by the tracking resolution of the telescope and charge sharing properties in the pixel sensors. Assuming similar charge sharing along both pixel edges and the tracking resolution being isotropic, the σ from the x-direction residual fit is used to obtain a better estimate of the y-distribution's width.

The distance between the falling edge and rising edge, as given by x_{fall} and x_{rise} in the fit, are used to extract the width. For all runs used in the analysis, the width in both directions as well as the σ parameter are fitted and plotted in Figure 9.12. The deviations from the nominal pixel pitch of 250 µm in x-direction and 50 µm in y-direction are approximately 3.5 µm in both directions. The smaller residual width is expected due to charge sharing between adjacent pixels. No difference of the absolute deviation between expected and measured residual width in x- and y-direction is anticipated, i.e. the same value is expected in both directions. This is reflected in the observed result.

The σ parameter is a measure of the intrinsic tracking resolution convolved with the turn on behaviour of the pixel, governed by the charge collection properties. The fit result of $\sigma = 14.5$ µm are slightly higher than the previously reported 10 µm we
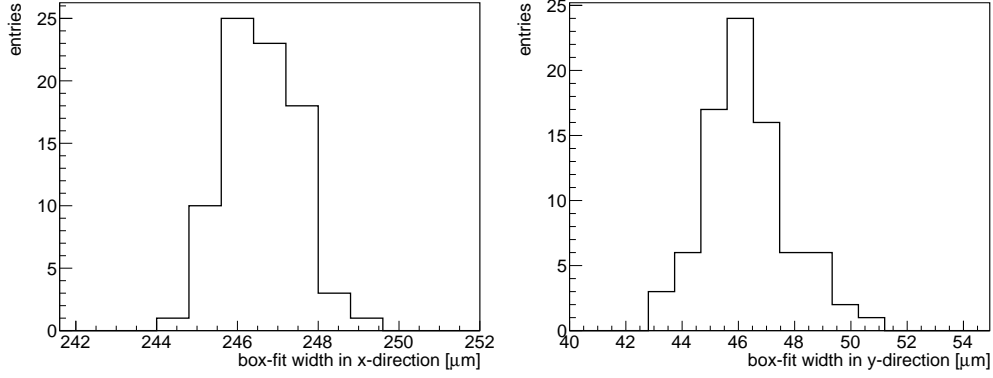
(a) The obtained box-width in x-direction. The mean width is at (246 ± 0.9) µm.

(b) The obtained box-width in y-direction. The mean width is at (46.4 ± 1.6) µm.
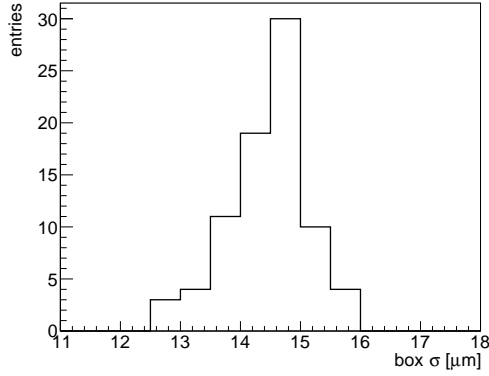
(c) The fitted box σ with a mean value of (14.5 ± 0.7) µm.

Figure 9.12.: The box widths and σ from the fit for runs used in the subsequent analysis. All the processed runs are well centred with no significant outliers.

obtained during the HV-CMOS measurements with a similar set-up [97]. However, the HV-CMOS test beam investigated sub-pixel encoding and the set-up was optimised for in-pixel studies and the beam energy was at 5 GeV compared to the 4 GeV. The 4 GeV were used to increase the trigger rate which is already decreased by a factor of five due to the gating window. The tracking resolution is sufficient for any global in-time studies.

9.3.1. Efficiency Definition

The analysed runs all have at least 10^5 triggers with 1-2 tracks per trigger in the active time window of the FE-I4, which all contribute to the global efficiency. The statistical uncertainty in the case of non-extreme efficiencies, i.e. away from 1 and 0 with a sufficient number of events, should be estimated as a binomial uncertainty. This yields $\delta\varepsilon =$

$1/N\sqrt{N\hat{\varepsilon}(1-\hat{\varepsilon})}$ where the measured efficiency as given by the number of observed tracks divided by the expected tracks is used as a best estimator for $\hat{\varepsilon}$. Assuming $N \approx 10^5$ the statistical uncertainties are in the order of 0.04% and thus neglected as they are a magnitude smaller than the assumed systematical uncertainty. The above definition is used as the estimator for the efficiency, the number of observed tracks in the fiducial region divided by the number of expected tracks in that region:

$$\varepsilon = \frac{\#\text{observed tracks}|\text{fiducial region}}{\#\text{expected tracks}|\text{fiducial region}} \tag{9.2}$$

The expected tracks are determined by the reference sample. This poses a possible bias and would be true if an inefficiency of the reference sample is correlated to an inefficiency on the DUT. For example, if the FE-I4 suffered from an inefficient lower left corner region and this were the case for the reference sample as well as the DUT, the measurement of the overall efficiency would be biased as the sensors' inefficiency is spatially correlated. Another bias could arise if the FE-I4 were inefficient in a temporal fashion as the FE clocks are completely correlated by the DAQ system.

No such behaviour of the FE-I4 has been reported. Assuming no hidden correlated inefficiencies, the determined efficiency is not biased by the reference sample.

9.4. Results

Aside from establishing the measurement technique, two parameters of the FE were investigated: the impact of the tuning as well as the discriminator bias (DisVbn). The discriminator bias is known to impact the speed of the discriminator and effect of the time-walk [98]. The DisVbn parameter is therefore also used in timing the IBL modules in the actual ATLAS detector.

Table 9.1 gives an overview of the five different sets the runs were divided into. The ToT tuning is given as the target ToT value at a given injected charge, e.g. *7 at 16 ke* is a tuning where at 16 ke injected charge the pixel is tuned to respond with a ToT value of 7.

	Threshold [e]	ToT tuning	DisVbn [DAC unit]
Sample 1	1200	7 at 14 ke	26
Sample 2	2500	7 at 16 ke	26
Sample 3	2500	7 at 16 ke	30
Sample 4	2500	7 at 16 ke	28
Sample 5	2500	8 at 16 ke	26

Table 9.1.: Samples used in the different analyses.

All the measurements were taken with *HitDiscCnfg* set to 0 and *SmallHitErase* enabled, which is the default test beam recommendation for measurements with the FE-I4.

	global efficiency [%]
Sample 1	97.5 ±0.2
Sample 2	97.4 ±0.1
Sample 3	97.4 ±0.1
Sample 4	97.4 ±0.1
Sample 5	97.4 ±0.1

Table 9.2.: Global efficiencies obtained for the five different samples.

9.4.1. Alignment Studies

Sample 1 aims at a low threshold tuning. As signals induced by small charges are most affected by time-walk effects, this sample exhibits the most distinct temporal features. However, it should also show the largest overall efficiency, assuming that the tuning to low threshold does not increase the noise.

All the in-time efficiencies quoted are relative to the global module's efficiency, i.e. if the module efficiency is at 97% and at a given delay point the global efficiency for a LV1 accept bin is 97% as well, the quoted relative efficiency is 100%.

Hence, it is important to quote the global efficiency. For this sample, without any ToT cut, the observed global efficiency was determined to be $(97.5 \pm 0.2)\%$. This is inside a fiducial area of 8 mm × 8 mm in the sensor's centre without any exclusion of noisy or disabled pixels, i.e. the module efficiency. The quoted uncertainty is the standard deviation from all measurement points and used to assess the systematic uncertainty. The results for the other runs are also indicated in Table 9.2.

During data acquisition, several runs at a higher beam energy of 5 GeV were recorded. No gating was superimposed. These runs are used as so-called alignment runs. As the alignment was not modified during the entire data collection period, the alignment from the alignment runs was grouped into a single overall alignment. Using this alignment ensures that the alignment and analysis data are uncorrelated, i.e. the alignment is not biased towards the analysed tracks.

Alignment values were however still computed for every single run. The derived alignment values were compared to the obtained alignment from the alignment runs. This ensured that no systematic drift, for example due to a mechanical displacement over time, took place.

The actual analysis was performed with the alignment-run based alignment as well as the alignment on a run-to-run basis. The results were compared and no significant difference was observed. No impact of the chosen alignment is expected as the alignment fluctuations and the alignment bias are beyond the DUT's pixel pitch.

In Figure 9.13 the results for Sample 1 are shown. Subfigure 9.13a shows the in-time efficiency for three subsequent LV1 accept bins. The efficiency of a single bin must not exceed 100%, however a single hit can be detected in more than one LV1 window if charge detection is delayed. Hence, the efficiency for a given delay point for multiple LV1 accept windows may exceed 100%.

Subfigure 9.13b shows the comparison of the different alignments. The efficiency obtained by the run-by-run derived alignment is divided by the efficiency obtained by the alignment-run based alignment. The error bars are not the statistical uncertainty of that fraction, but indicate the relative statistical uncertainty, i.e. $\delta\varepsilon/\varepsilon$, for the obtained efficiency at that point, i.e. the binomial uncertainty as discussed above. It can be seen, that the impact of the alignment is even smaller than the expected statistical fluctuation.

The impact of the alignment was also studied with the other samples. Equally, the choice of alignment mode does not impact the obtained results. In consequence, it is concluded that for a global efficiency analysis the alignment is sufficiently good.

9.4.2. Low Threshold: Sample 1

In the temporally resolved in-time efficiency distributions for Sample 1 shown in the previous section, at the rising edges in LV1 accept bin 2 and 3 a flat increase is observed up to the point where the previous LV1 accept bin rapidly drops and the next LV1 bin takes over. The observed efficiency is expected to be due to time-walk of small charge deposits. Hence, this effect should decrease if an additional ToT cut is applied. A value of ToT ≥ 4 was used for this study. The obtained in-time efficiency distribution is shown in Figure 9.14a. Additionally, the shape was fitted with an S-curve fit, motivated by the box-shaped turn on behaviour which is smeared out by the temporal signal formation and FE response.

Compared to the full data set, only a very small number of hits are detected by the LV1 accept bin 3. In order to compare the distributions with and without ToT cut, the obtained fit from Figure 9.14a is shown in Figure 9.14b, which is the distribution already shown in Figure 9.13a.

It can be noted that despite the ToT cut, the relative in-time efficiency approaches 100% in the plateau regions, and mostly only the turn-on behaviour of the accept LV1 bins is changed. This is in contrast to the higher threshold and different ToT tuning for the subsequent samples.

### 9.4.3. DisVbn: Sample 2, Sample 3, and Sample 4

The Samples 2, 3, and 4 (cf. Table 9.1) are used to determine the impact of the DisVbn parameter. Additionally, the impact of the ToT cut was studied with Sample 4. In order to quantify the timing behaviour, an S-curve fit, similar to the fit given in Equation 9.1 but with a y_{offset} of zero and using only the rising or falling part in a single fit, was used to determine the precise time in the region of interest. The obtained rise or fall times are referred to as t_{rise} or t_{fall}, respectively. To reduce the impact of time-walk a ToT cut of ≥ 4 was used. The results are given in Figure 9.15. A tuning with a threshold of 2500 e and a ToT response of 7 ToT at 16 ke injected charge was done.

The $t_{\mathrm{rise/fall}}$ parameters obtained by the fit are used to indicate the relative crossing time. The rising and falling slope fits are performed independent of each other. In Sample 2 (Fig. 9.15a) the falling slope does not feature any measurements in the region where the plateau is expected to be at its maximum. This results in a larger uncertainty
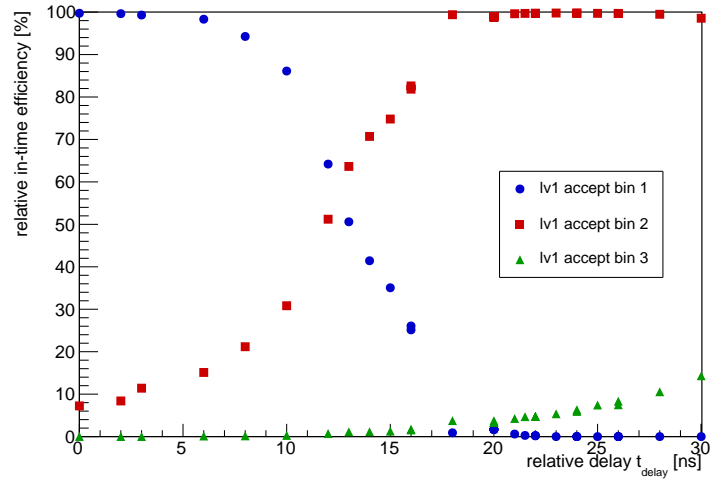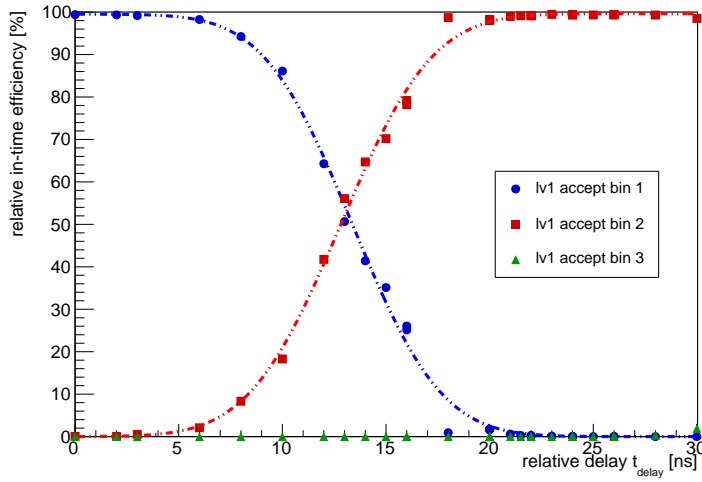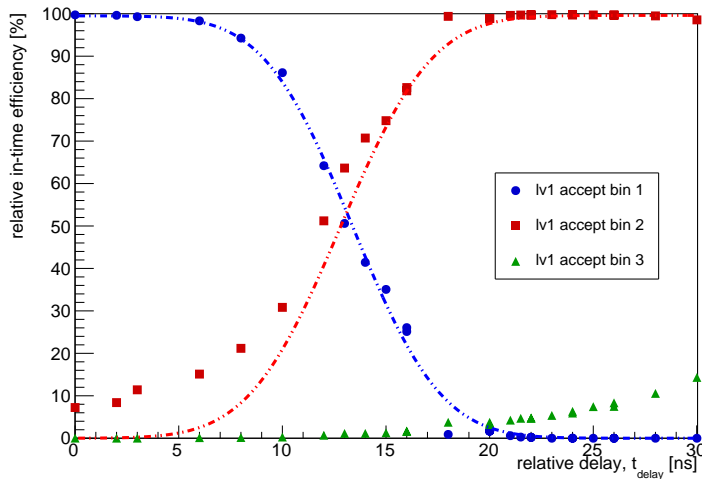
(a) The derived in-time efficiency distribution for Sample 1. No ToT cut is applied.
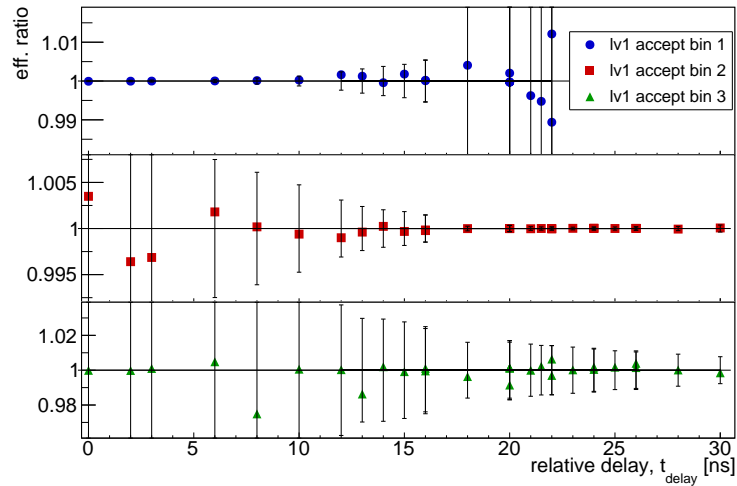
(b) Ratio of the efficiencies obtained by a run-by-run based alignment (nominator) and an alignment-run based alignment (denominator).

Figure 9.13.: Shown are the overall in-time efficiency measurements as well as the studies of the impact of the alignment for Sample 1. Multiple data points, for example at $t_{\text{delay}} = 20$ ns, are due to multiple measurements for that delay value.

on the obtained normalisation (max). The rising slope fit was limited to a region before

(a) Sample 1 with an applied ToT cut of ≥ 4.

(b) The in-time efficiency distribution without any ToT cut with the overlaid fit result from (a).

Figure 9.14.: The effect of a ToT cut on Sample 1. The removal of low ToT values is visible at the rising slopes due to hits effected by time-walk being cut away.
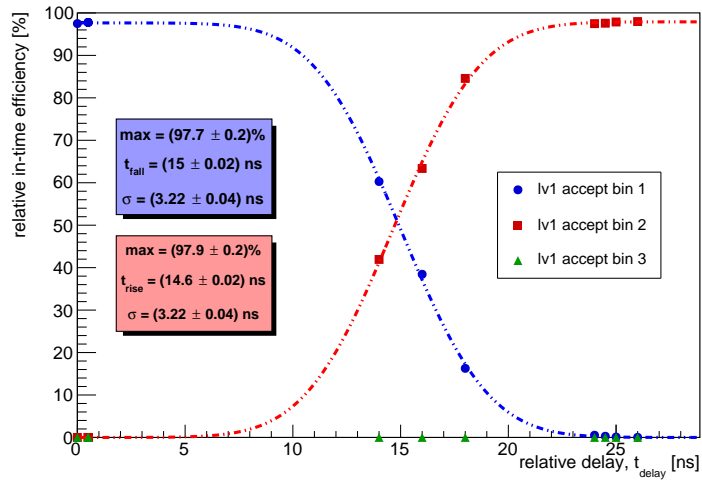
the plateau starts to decrease again.

The obtained crossing time values from those fits are extracted and shown in Figure 9.16. A clear delay of the relative crossing time with increased DisVbn is visible. This behaviour is expected and exploited in adjusting the timing of the FE-I4 chips of IBL within the ATLAS detector.
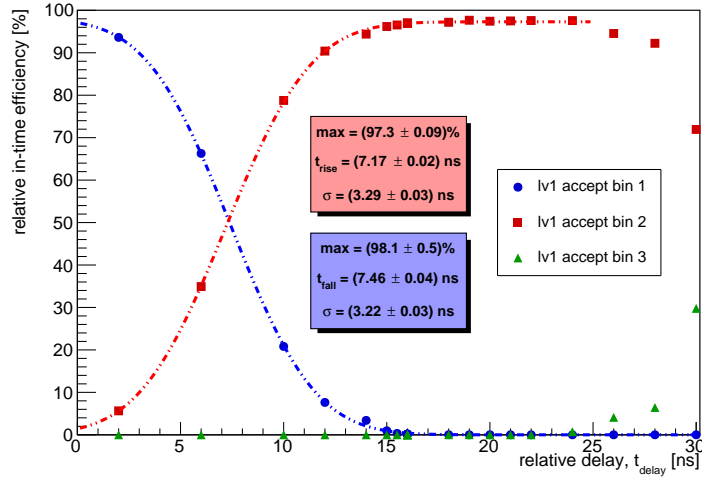
(a) Sample 2 with DisVbn 26.

(b) Sample 3 with DisVbn 30.

Figure 9.15.: Timing plots for different DisVbn parameters. Shown are the obtained in-time efficiency distributions for a ToT ≥ 4 with the corresponding fit results (continued on next page).
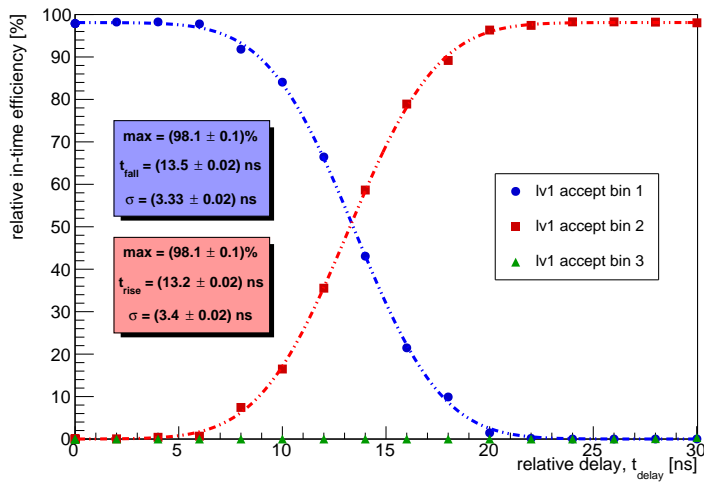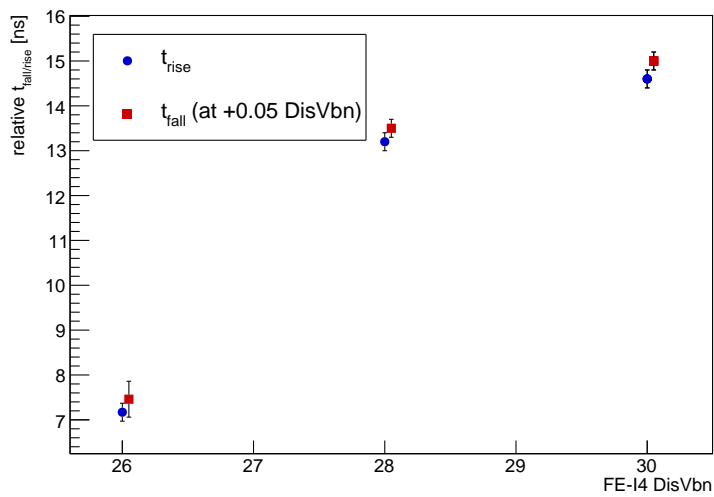
(c) Sample 4 with DisVbn 28.

Figure 9.15.: Timing plots for different DisVbn parameters. Shown are the obtained in-time efficiency distributions for a ToT ≥ 4 with the corresponding fit results (continued from previous page).


Figure 9.16.: Relative crossing times with respect to the DisVbn parameter. The t_{fall} are slightly displaced for better visibility.

121

9.4.4. Effect of ToT cut: Sample 4

In order to study the impact of time-walk, various ToT cuts are applied and the obtained in-time efficiency distributions are compared. They are shown in Figure 9.17, where a sample without ToT cut, a ToT cut of ≥ 2, of ≥ 3, and ≥ 4 are compared. It can be seen that the low ToT hits are delayed and registered in the subsequent bin, most clearly visible when comparing the LV1 accept bin 3 where there is a rise for long relative delays, t_{delay}.
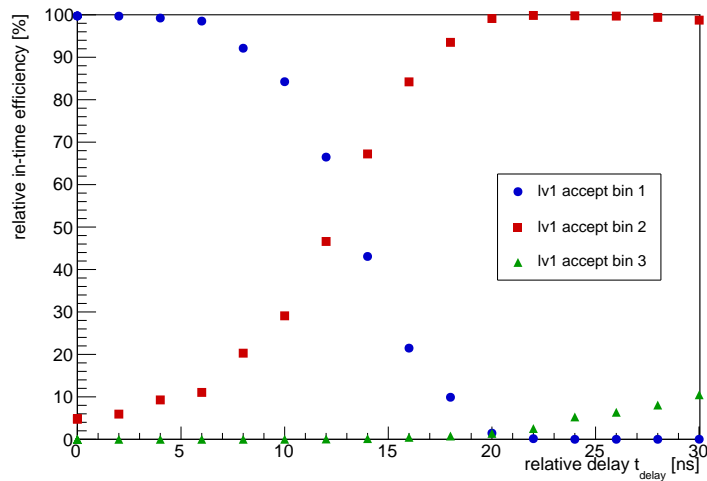
(a) In-time efficiency with no ToT cut applied.
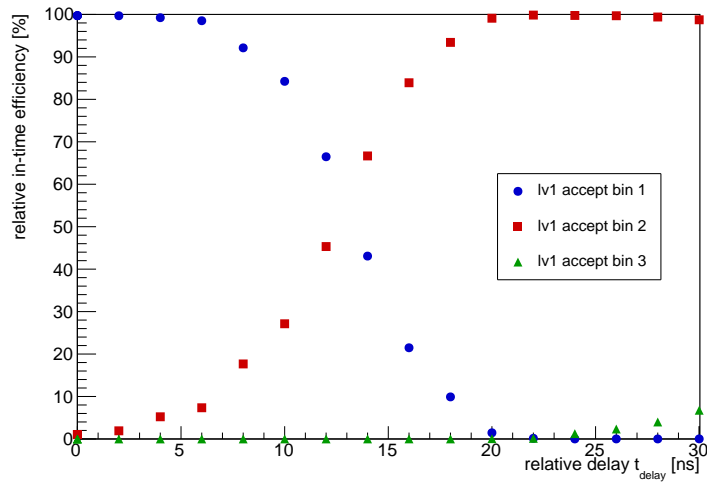
(b) In-time efficiency with a ToT cut of ≥ 2 applied.

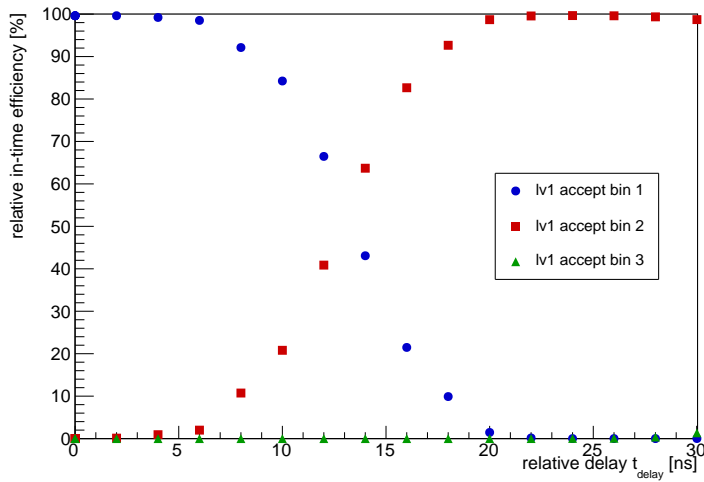Figure 9.17.: Different in-time efficiency distributions with various ToT cuts applied (continued on next page).

(c) In-time efficiency with a ToT cut of ≥ 3 applied.

(d) In-time efficiency with a ToT cut of ≥ 4 applied.

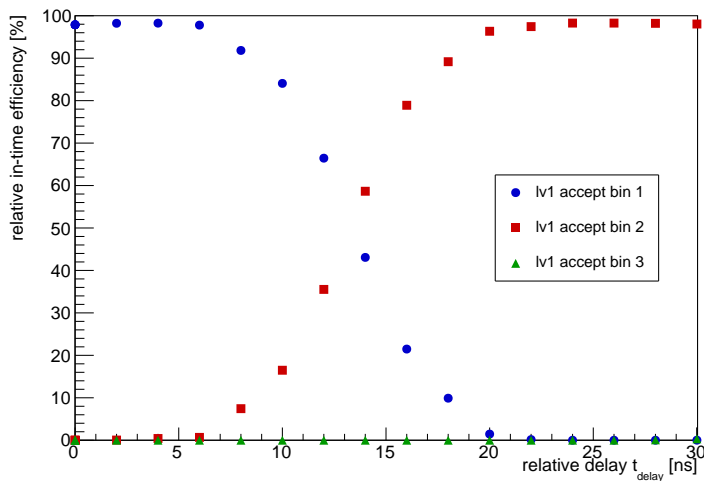Figure 9.17.: Different in-time efficiency distributions with various ToT cuts applied (continued from previous page).

This effect is also visible in the rising slope of the LV1 accept bin 2. It decreases for increasing ToT cut values. With a ToT cut of ≥ 3 and the given tuning, nearly no hits are detected in the LV1 accept bin 3, even for the largest relative delay times, shown in Fig 9.17c.

As visible in Figure 9.17d, the maximum achieved efficiency decreases in the plateau regions. This is in contrast to Sample 1, cf. Figure 9.14a, where the maximal efficiency

was not effected by the same ToT cut, due to the lower threshold and different ToT tuning.

9.4.5. Plateau length: Sample 4

In order to estimate the width of the region where the timing is ideal, data points from the previous LV1 bin were duplicated and shifted by 25 ns in the positive direction. The same was done for the subsequent bin with a negative shift of 25 ns. In order to reduce the effect of time-walk, the ToT cut of ≥ 4 was used, reducing the relative maximal in-time efficiency slightly.
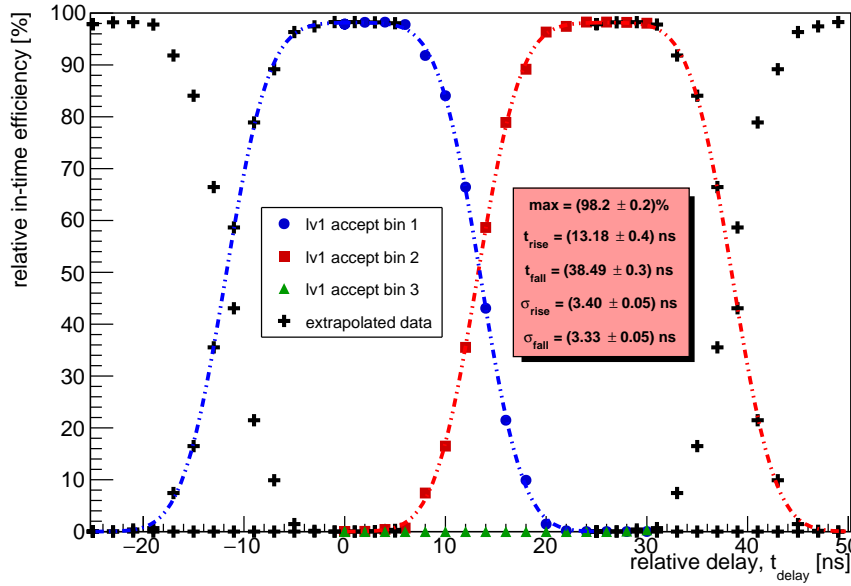

Figure 9.18.: Extracting the duration of the efficient plateau from Sample 4.

For both LV1 accept bins a fit similar to Equation 9.1 is performed, with the difference that the rising and falling slope are not required to be equal and the unit step function was offset appropriately. The results as well as the obtained fit parameters for the LV1 accept bin 2 are shown in Figure 9.18. The fit for the LV1 accept bin 1 uses the exact same data, hence there is no additional information gain as the data are fully correlated. However, it was carried out to verify that the fitting procedure works, as any deviation would hint at an error in the fit procedure. Aside from the shifted $t_{\text{fall/rise}}$, no deviation was observed.

The obtained value for the duration is $t_{\text{plateau}} = t_{\text{fall}} - t_{\text{rise}} = (25.31 \pm 0.5)$ ns, with a $\sigma_{\text{rise}} = 3.4$ ns and a $\sigma_{\text{fall}} = 3.3$ ns.

It must be emphasised that in order to obtain these results, a repetitive behaviour every 25 ns is assumed. In consequence, the duration is biased by this assumption.

9.4.6. ToT Tuning: Sample 5

Sample 5 aimed at investigating the effect of a slightly deviating ToT tuning compared to Sample 2. Limited beam time reduced the amount of data points and mostly the plateau region was probed. Assuming a linear relation between ToT and charge (this is not true, though it holds as a first approximation), a higher ToT response at the same injected charge at the same threshold, implies that a ToT cut of ≥ 4 should have less impact on the overall relative efficiency. This is because the same ToT corresponds to less physical charge with this tuning. Compared to Sample 2, a slightly higher efficiency is observed.
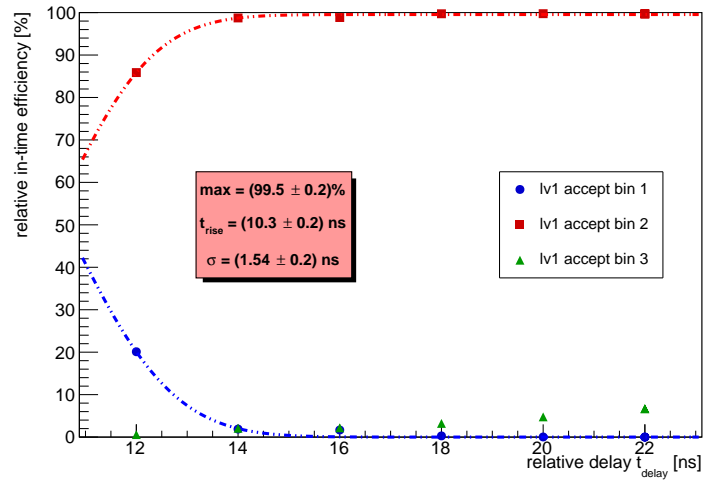
(a) Sample 5 without a ToT cut.
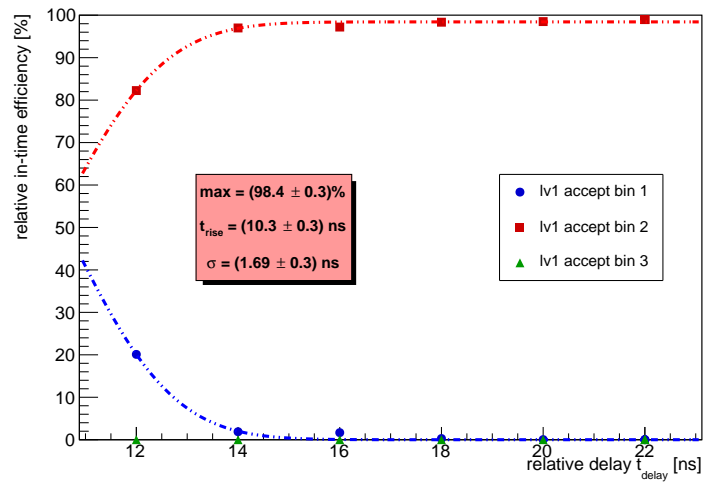
(b) Sample 5 with ToT ≥ 4.

Figure 9.19.: Results of the measurements for Sample 5.

Conclusion and Outlook

10.1. USBpix EUDAQ Integration

The new integration of the EUDAQ interface has been used since early 2017. Not only is the USBpix system with STcontrol one of the default DAQ systems for the ATLAS ITk Pixel community, it has also been extensively used by the Göttingen group for measurements of novel diamond modules within the RD-42 collaboration. Furthermore, all measurements for the in-time efficiency studies have been carried out with the new STcontrol version using USBpix boards of the third generation.

The integration with a clear separation of tasks which are not synchronised by means of inter-thread synchronisation, but with thread-safe data structures has proven to be very successful. Up to today, no bugs regarding the actual producer have been reported.

The persisting issues all relate to the synchronisation of the DAQ boards and the TLU. For the third generation board, the philosophy of EUDAQ2 and the miniTLU are incorporated, meaning there is no exchange of trigger number between the TLU and DAQ boards. This no-trigger exchange mode is a valid option for EUDAQ(1) as well, however, it has not been used in STcontrol before. Since no trigger number is exchanged and merely a trigger is issued, the DAQ systems internally need to keep track of the trigger number.

Given that the synchronisation problems only effect the second generation boards at the time when the trigger *rolls-over*, i.e. the internal counter overflows and resets, it seems likely that there is either a problem with the TLU not resetting this counter properly or the USBpix firmware. A modification in which no trigger number is used, has been proposed and implemented, but still needs to be tested in a test beam environment together with a telescope.

A similar approach for the design of further EUDAQ integrations is hence suggested, in particular in consideration of the upcoming DAQ hardware and software for the RD-53 upgrade project.

10.2. EUTelescope

EUTelescope has changed over the past few years and is an important tool for many test beam analyses. Many of the changes aimed at making EUTelescope an easier and more maintainable tool for the future. In particular, mechanisms were introduced to validate that new changes to the EUTelescope code only have the desired effect, and do not change existing reconstruction code.

This is particularly important as test beam measurements will be used by the ATLAS ITk upgrade community to make a final market survey. Chosen vendors will provide samples which all need to be tested and compared. It is of utmost importance that measurements are done with comparable standards. Moreover, test beams will be utilised in the quality assurance during the process of module building where again comparable results are needed.

If any modifications to the code are made, they need to be fully understood so that measurements done with them can be put in comparison. Properties like the efficiency depend on the physical properties of the module, on the configuration, on the applied bias voltage, or on other physical properties as well as the definition of the quoted efficiency. They must neither depend on the test beam facility used for the measurement, nor on the version of the reconstruction or the analysis framework.

As one can guess, measurements at lower particle energies might be more difficult to perform as scattering has a larger effect. However, this must be reflected in the quoted systematic uncertainty, and be consistent (within statistical fluctuation) with other results. For example, modifications to the GEAR-file allow to easily monitor fluctuations of the alignment. This can be used to estimate the fluctuation in the alignment.

Moving to the GBL track-fit has the advantage that the GBL algorithm is also widely used by the users of the EUDET-type telescopes. They have, for example, used this fit to not only study properties of the telescope, but also showed the power of using it in multiple scattering tomography at the DESY test beam facility [99]. This is of particular interest, as the DESY test beam site will be used in the upcoming studies by the ATLAS community due to the shut-down of the SPS in the next few years.

A simulation framework like Allpix^2 is needed to validate the algorithms and basic functionality of test beam reconstruction. While data driven validations give a good initial *gut feeling* if things go right, the ultimate validation must be performed with a full physical simulation and access to the Monte Carlo truth level data. It is not without reason, that Monte Carlo frameworks play a crucial role, for example in cross-section measurements. Similarly, at test beams they should be utilised to fully understand one's set-up.

10.3. In-Time Efficiency Measurement

The measurement set-up described in Chapter 9 was the first time the ATLAS Pixel test beam groups had experimental access to the time behaviour properties of the tested modules. Together with the functionality implemented in EUTelescope, this set-up enabled

time-resolved measurements. This becomes even more important as for the upcoming RD-53 chip parameters like the in-time efficiency are explicitly stated in the design documents. Hence, having the possibility to perform these measurements in a test beam campaign, is an important tool for testing the new read-out chip.

While in principle it would be desirable to perform all test beam measurements time-resolved, this is not feasible. To obtain in-time efficiency distributions, the parameter space that is needed to be measured is increased by a factor corresponding to the number of points for the relative delay parameter. Furthermore, the trigger rate is reduced due to the gating. With the approximately 5 ns gating window for the performed measurements, the reduction is by a factor of five.

Time-resolved measurements allow to probe the temporal properties of the read-out chip. In the case of the FE-I4, the speed of the discriminator stage as well as the effect of time-walk can be investigated. This has been done in the context of this thesis by comparing different tunings and various DisVbn parameters. Moreover, the taken measurements helped to develop and test the possibility to perform time-resolved measurements and lay the foundation for future test beam campaigns.

Concerning time-resolved measurements with the FE-I4, a scan over a larger DisVbn range would be desirable, in particular with IBL-style modules with the same tuning as in the IBL detector. Better understanding of the temporal properties of signal formation and processing within the read-out chip could help to determine operational parameters for detector operation.

The FE-I4 has a parameter to correctly assign so-called *small hits* to the correct LV1 accept bin, the HitDiscCnfg parameter. Measurements have only been performed with this parameter set to 0, where no hits are classified as *small hits*. The effect of the HitDiscCnfg could be investigated in time-resolved test beam studies in the future.

Regarding the experimental set-up, the used NIM modules are widely available. As mechanical delay lines were used, physically entering the test beam area to modify the delay parameter was necessary. Also, it is not possible to reduce the gate length below 5 ns with most NIM components. A shorter gate duration would increase the temporal resolution, yet it reduces the data rate even further. It could open the window to investigate temporal signal formation properties, however this will still be convolved with the response of the read-out chip. Using ECL components might enable processing logical signals in the ns to sub-ns range, though high speed electronic engineering in the GHz frequency range is needed to achieve this. A remote possibility to modify the delay parameters would be desirable.

A set-up and measurement which acquires data over a longer delay than 30 ns would provide an unbiased measurement of the exact in-time efficiency plateau length, without the need of assuming a 25 ns repetitive behaviour. With even longer delays, the migration of the signal into further, i.e. more than three, LV1 bins should become possible.

Given the upcoming tasks for the ATLAS ITk Pixel community, in particular a broad quality assessment of the available vendors as well as quality assurance during the subsequent module production, it would be naive to assume as of right now that in-time measurements are a high priority task. Nonetheless, they are important to quantify the

temporal properties of the new read-out chip as well as give crucial information for the final detector operation, and for this reason need to be carried out.

[1] J. J. Thomson, *XL. Cathode Rays*, Philos. Mag. **44** (1897) 293–316.

[2] A. Einstein, *Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen*, Ann. Phys. **322** (1905) 549–560.

[3] E. Rutherford, *LXXIX. The scattering of α and β particles by matter and the structure of the atom*, Philos. Mag. **21** (1911) 669–688.

[4] J. Chadwick, *Possible Existence of a Neutron*, Nature **129** (1932) 312–312.

[5] J. Chadwick, *The Existence of a Neutron*, Proc. Royal Soc. Lond. A **136** (1932) 692–708.

[6] P. Dirac, *The principles of quantum mechanics.* Oxford University Press, 1930.

[7] C. D. Anderson, *The Apparent Existence of Easily Deflectable Positives*, Science **76** (1932) 238–239.

[8] C. D. Anderson, *The Positive Electron*, Phys. Rev. **43** (1933) 491–494.

[9] E. Fermi, *Tentativo di una Teoria Dei Raggi β*, Nuovo Cim. **11** (1934) 1–19.

[10] E. Fermi, *Versuch einer Theorie der β-Strahlen. I*, Zeitschrift für Phys. **88** (1934) 161–177.

[11] H. Yukawa, *On the Interaction of Elementary Particles. I*, Proc. Physico-Mathematical Soc. Japan. 3rd Ser. **17** (1935) 48–57.

[12] C. D. Anderson and S. H. Neddermeyer, *Cloud Chamber Observations of Cosmic Rays at 4300 Meters Elevation and Near Sea-Level*, Phys. Rev. **50** (1936) 263–271.

BIBLIOGRAPHY

[13] C. M. G. Lattes, et al., *Processes involving charged mesons*, Nature **159** (1947) 694–697.

[14] S.-I. Tomonaga, *On a Relativistically Invariant Formulation of the Quantum Theory of Wave Fields*, Prog. Theor. Phys. **1** (1946) 27–42.

[15] Z. Koba, T. Tati, and S.-I. Tomonaga, *On a Relativistically Invariant Formulation of the Quantum Theory of Wave Fields. III: Case of Interacting Electromagnetic and Electron Fields*, Prog. Theor. Phys. **2** (1947) 198–208.

[16] S. Kanesawa and S.-I. Tomonaga, *On a Relativistically Invariant Formulation of the Quantum Theory of Wave Fields V: Case of Interacting Electromagnetic and Meson Fields*, Prog. Theor. Phys. **3** (1948) 101–113.

[17] S.-I. Tomonaga and J. R. Oppenheimer, *On Infinite Field Reactions in Quantum Field Theory*, Phys. Rev. **74** (1948) 224–225.

[18] J. Schwinger, *On Quantum-Electrodynamics and the Magnetic Moment of the Electron*, Phys. Rev. **73** (1948) 416–417.

[19] J. Schwinger, *Quantum Electrodynamics. I. A Covariant Formulation*, Phys. Rev. **74** (1948) 1439–1461.

[20] R. P. Feynman, *Space-Time Approach to Non-Relativistic Quantum Mechanics*, Rev. Mod. Phys. **20** (1948) 367–387.

[21] R. P. Feynman, *Relativistic Cut-Off for Quantum Electrodynamics*, Phys. Rev. **74** (1948) 1430–1438.

[22] J. A. Wheeler and R. P. Feynman, *Interaction with the Absorber as the Mechanism of Radiation*, Rev. Mod. Phys. **17** (1945) 157–181.

[23] F. J. Dyson, *The Radiation Theories of Tomonaga, Schwinger, and Feynman*, Phys. Rev. **75** (1949) 486–502.

[24] C. L. Cowan, et al., *Detection of the Free Neutrino: a Confirmation*, Science **124** (1956) 103–104.

[25] C. S. Wu, et al., *Experimental Test of Parity Conservation in Beta Decay*, Phys. Rev. **105** (1957) 1413–1415.

[26] J. H. Christenson, et al., *Evidence for the 2π Decay of the K_2^0 Meson*, Phys. Rev. Lett. **13** (1964) 138–140.

[27] S. L. Glashow, *The renormalizability of vector meson interactions*, Nucl. Phys. **10** (1959) 107–117.

[28] S. Weinberg, *A Model of Leptons*, Phys. Rev. Lett. **19** (1967) 1264–1266.

[29] A. Salam and J. C. Ward, *Weak and electromagnetic interactions*, Nuovo Cim. **11** (1959) 568–577.

[30] F. Englert and R. Brout, *Broken symmetry and the mass of gauge vector mesons*, Phys. Rev. Lett. **13** (1964) 321–323.

[31] P. W. Higgs, *Broken symmetries and the masses of gauge bosons*, Phys. Rev. Lett. **13** (1964) 508–509.

[32] M. Gell-Mann, *THE EIGHTFOLD WAY: A THEORY OF STRONG INTERACTION SYMMETRY*, tech. rep., Divison of Technical Information Extension, U.S. Atomic Energy Commission, CTSL-20, TID-12608, Mar, 1961.

[33] M. Gell-Mann, *Symmetries of Baryons and Mesons*, Phys. Rev. **125** (1962) 1067–1084.

[34] M. Gell-Mann, *A schematic model of baryons and mesons*, Phys. Lett. **8** (1964) 214–215.

[35] G. Zweig, *An SU_3 model for strong interaction symmetry and its breaking*, CERN-TH-401, 1964. `https://cds.cern.ch/record/352337`.

[36] G. Zweig, *An SU_3 model for strong interaction symmetry and its breaking II*, CERN-TH-412, 1964. `http://cds.cern.ch/record/570209`.

[37] E. D. Bloom, et al., *High-Energy Inelastic e-p Scattering at 6° and 10°*, Phys. Rev. Lett. **23** (1969) 930–934.

[38] M. Breidenbach, et al., *Observed Behavior of Highly Inelastic Electron-Proton Scattering*, Phys. Rev. Lett. **23** (1969) 935–939.

[39] B. Bjørken and S. L. Glashow, *Elementary particles and SU(4)*, Phys. Lett. **11** (1964) 255–257.

[40] S. L. Glashow, J. Iliopoulos, and L. Maiani, *Weak Interactions with Lepton-Hadron Symmetry*, Phys. Rev. D **2** (1970) 1285–1292.

[41] J. E. Augustin, et al., *Discovery of a Narrow Resonance in e^+e^- Annihilation*, Phys. Rev. Lett. **33** (1974) 1406–1408.

[42] J. J. Aubert, et al., *Experimental Observation of a Heavy Particle J*, Phys. Rev. Lett. **33** (1974) 1404–1406.

[43] M. Kobayashi and T. Maskawa, *CP-Violation in the Renormalizable Theory of Weak Interaction*, Prog. Theor. Phys. **49** (1973) 652–657.

[44] S. W. Herb, et al., *Observation of a Dimuon Resonance at 9.5 GeV in 400-GeV Proton-Nucleus Collisions*, Phys. Rev. Lett. **39** (1977) 252–255.

[45] R. Brandelik, et al., *Evidence for planar events in e^+e^- annihilation at high energies*, Phys. Lett. B **86** (1979) 243–249.

[46] UA1 Collaboration, *Experimental observation of isolated large transverse energy electrons with associated missing energy at $\sqrt{s} = 540$ GeV*, Phys. Lett. B **122** (1983) 103–116.

[47] UA2 Collaboration, *Observation of single isolated electrons of high transverse momentum in events with missing transverse energy at the CERN $\bar{p}p$ collider*, Phys. Lett. B **122** (1983) 476–485.

[48] UA1 Collaboration, *Experimental observation of lepton pairs of invariant mass around 95 GeV/c^2 at the CERN SPS collider*, Phys. Lett. B **126** (1983) 398–410.

[49] UA2 Collaboration, *Evidence for $Z^0 \rightarrow e^+e^-$ at the CERN $\bar{p}p$ collider*, Phys. Lett. B **129** (1983) 130–140.

[50] DØ Collaboration, *Search for High Mass Top Quark Production in $p\bar{p}$ Collisions at $\sqrt{s}$=1.8 TeV*, Phys. Rev. Lett. **74** (1995) 2422–2426.

[51] CDF Collaboration, *Observation of Top Quark Production in $\bar{p}p$ Collisions with the Collider Detector at Fermilab*, Phys. Rev. Lett. **74** (1995) 2626–2631.

[52] ATLAS Collaboration, *Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC*, Phys. Lett. B **716** (2012) 1–29.

[53] CMS Collaboration, *Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC*, Phys. Lett. B **716** (2012) 30–61.

[54] ATLAS Collaboration, *Standard Model Summary Plots Spring 2019*, ATL-PHYS-PUB-2019-010, 2019. `https://cdsweb.cern.ch/record/2668559/`.

[55] ATLAS Collaboration, *Performance of b-jet identification in the ATLAS experiment*, JINST **11** (2016) P04008.

[56] ATLAS Collaboration, *Expected performance of the ATLAS b-tagging algorithms in Run-2*, ATL-PHYS-PUB-2015-022, 2015. `https://cds.cern.ch/record/2037697`.

[57] ATLAS Collaboration, *Search for the standard model Higgs boson produced in association with top quarks and decaying into a $b\bar{b}$ pair in pp collisions at $\sqrt{s}$=13 TeV with the ATLAS detector*, Phys. Rev. D **97** (2018) 72016.

[58] ATLAS Collaboration, *Projections for measurements of Higgs boson cross sections, branching ratios and coupling parameters with the ATLAS detector at a HL-LHC*, ATL-PHYS-PUB-2013-014, 2013. `https://cdsweb.cern.ch/record/1611186`.

[59] M. J. Dolan, C. Englert, and M. Spannowsky, *Higgs self-coupling measurements at the LHC*, J. High Energy Phys. **2012** (2012) 112.

[60] ATLAS Collaboration, *Prospects for measuring Higgs pair production in the channel $H(\to\gamma\gamma)H(\to b\bar{b})$ using the ATLAS detector at the HL-LHC*, ATL-PHYS-PUB-2014-019. `https://cds.cern.ch/record/1956733`.

[61] ATLAS Collaboration, *Higgs pair production in the $H(\to\tau\tau)H(\to b\bar{b})$ channel at the High-Luminosity LHC*, ATL-PHYS-PUB-2015-046, 2015. `https://cdsweb.cern.ch/record/2065974`.

[62] F. Zwicky, *Die Rotverschiebung von extragalaktischen Nebeln*, Helv. Phys. Acta **6** (1933) 110–127.

[63] F. Zwicky, *On the Masses of Nebulae and of Clusters of Nebulae*, Astrophys. J. **86** (1937) 217.

[64] R. Davis, D. S. Harmer, and K. C. Hoffman, *Search for Neutrinos from the Sun*, Phys. Rev. Lett. **20** (1968) 1205–1209.

[65] Y. Fukuda, et al., *Evidence for oscillation of atmospheric neutrinos*, Phys. Rev. Lett. **81** (1998) 1562–1567.

[66] S. Mertens, *Direct Neutrino Mass Experiments*, J. Phys. Conf. Ser. **718** (2016) 022013.

[67] G. Apollinari, et al., *High-Luminosity Large Hadron Collider (HL-LHC): Technical Design Report V. 0.1*, CERN-2017-007-M, 2017. `https://cds.cern.ch/record/2284929`.

[68] J. Coupard, et al., *LHC Injectors Upgrade Technical Design Report - Vol. I: Protons*, CERN-ACC-2014-0337, 2014. `https://cds.cern.ch/record/1976692`.

[69] J. Coupard, et al., *LHC Injectors Upgrade Technical Design Report - Vol. II: Ions*, CERN-ACC-2016-0041, 2016. `http://cds.cern.ch/record/2153863`.

[70] G. Lindström, *Radiation damage in silicon detectors*, Nucl. Instr. Meth. A **512** (2003) 30–43.

[71] ATLAS Collaboration, *ATLAS Insertable B-Layer Technical Design Report*, CERN-LHCC-2010-013, ATLAS-TDR-19, 2010. `https://cds.cern.ch/record/1291633`.

[72] ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST **3** (2008) S08003.

[73] ATLAS Collaboration, *Technical Design Report for the ATLAS Inner Tracker Strip Detector*, CERN-LHCC-2017-005, ATLAS-TDR-025, 2017. `https://cds.cern.ch/record/2257755`.

[74] ATLAS Collaboration, *Technical Design Report for the ATLAS Inner Tracker Pixel Detector*, CERN-LHCC-2017-021, ATLAS-TDR-030, 2017.
`https://cds.cern.ch/record/2285585`.

[75] ATLAS Collaboration, *ATLAS public plots: Study of SEU in FEI4*, PIX-2018-007, ATL-INDET-INT-2018-006, ATL-INDET-INT-2018-004, 2018.
`https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PLOTS/PIX-2018-007`.

[76] J. Lange, et al., *Radiation hardness of small-pitch 3D pixel sensors up to a fluence of 3×10^{16} n_{eq}/cm^2*, JINST **13** (2018) P09009.

[77] Particle Data Group, *Review of Particle Physics*, Chin. Phys. C **40** (2016) 100001.

[78] H. Bichsel, *A method to improve tracking and particle identification in TPCs and silicon detectors*, Nucl. Instr. Meth. A **562** (2006) 154–197.

[79] S. Meroli, D. Passeri, and L. Servoli, *Energy loss measurement for charged particles in very thin silicon layers*, JINST **6** (2011) P06013.

[80] W. Shockley, *Currents to Conductors Induced by a Moving Point Charge*, J. Appl. Phys. **9** (1938) 635–636.

[81] S. Ramo, *Currents Induced by Electron Motion*, Proc. IRE **27** (1939) 584–585.

[82] M. Moll, *Radiation damage in silicon particle detectors: Microscopic defects and macroscopic properties.* PhD thesis, Universität Hamburg, DESY-THESIS-1999-040, 1999.

[83] K. Dette, *Total Ionising Dose effects in the FE-I4 front-end chip of the ATLAS Pixel IBL detector*, JINST **11** (2016) C11028.

[84] K. Akiba, et al., *The Timepix Telescope for high performance particle tracking*, Nucl. Instr. Meth. A **723** (2013) 47–54.

[85] M. Benoit, et al., *The FE-I4 telescope for particle tracking in testbeam experiments*, JINST **11** (2016) P07003.

[86] H. Augustin, et al., *The MuPix Telescope: A Thin, High-Rate Tracking Telescope*, JINST **12** (2017) C01087.

[87] H. Jansen, et al., *Performance of the EUDET-type beam telescopes*, EPJ Tech. Instrum. **3** (2016) 7.

[88] D. Cussans, *Description of the JRA1 Trigger Logic Unit (TLU), v0.2c*, EUDET-MEMO-2009-04, 2009.

[89] T. Krämer, *Track Parameters in LCIO*, LC-DET-2006-004, 2006.

[90] M. Backhaus, et al., *Development of a versatile and modular test system for ATLAS hybrid pixel detectors*, Nucl. Instr. Meth. A **650** (2011) 37–40.

[91] S. Spannagel, et al., *Allpix^2: A modular simulation framework for silicon detectors*, Nucl. Instr. Meth. A **901** (2018) 164–172.

[92] S. Agostinelli, et al., *GEANT4 - A simulation toolkit*, Nucl. Instr. Meth. A **506** (2003) 250–303.

[93] J. Allison, et al., *Recent developments in GEANT4*, Nucl. Instr. Meth. A **835** (2016) 186–225.

[94] M. Habbaba, *Test beam analysis of pixel detector for the ATLAS upgrade.* Bachelor's thesis, Georg-August-Universität Göttingen, II.Physik-UniGö-BSc-2018/08, 2018.

[95] S. Terzo, *Development of radiation hard pixel modules employing planar n-in-p silicon sensors with active edges for the ATLAS detector at HL-LHC.* PhD thesis, Technische Universität München, 2015.

[96] J. Rieger, *Detector Development for the High Luminosity Large Hadron Collider.* PhD thesis, Georg-August-Universität Göttingen, II.Physik-UniGö-Diss-2016/01, 2016.

[97] T. Bisanz, et al., *Subpixel mapping and test beam studies with a HV2FEI4v2 CMOS-Sensor-Hybrid Module for the ATLAS inner detector upgrade*, JINST **12** (2017) P08018.

[98] FE-I4 Collaboration, *The FE-I4B Integrated Circuit Guide*, Version 2.3, 2012.

[99] H. Jansen and P. Schütze, *Feasibility of track-based multiple scattering tomography*, Appl. Phys. Lett. **112** (2018) 144101.

Appendices

A. Acknowledgements

GEAR File

The two valid syntaxes for a proper GEAR file to describe a telescope set-up are shown.

B.1. SiPlanes style layout

This is the old, initial syntax which has some ambiguities but is the most commonly used way of specifying a telescope. An example for merely two detector modules is given.

```
<gear>
  <!-- IBL testbeam June 2011 -->
  <global detectorName="EUTelescope"/>
  <BField type="ConstantBField" x="0.0" y="0.0" z="0.0"/>
  <detectors>
    <detector name="SiPlanes" geartype="SiPlanesParameters">
    <parameter name="Geometry" type="StringVec"
               value="Mimosa26.so Mimosa26.so"/>
    <siplanesID ID="95"/>
    <siplanesType type="TelescopeWithoutDUT"/>
    <siplanesNumber number="2"/>
    <layers>
      <!--Eudet-Plane 0 - EUD0 -->
      <layer>
        <ladder   ID="0"
          positionX="0.00"   positionY="0.00"   positionZ="0.00"
          rotationZY="0.00"  rotationZX="0.0"   rotationXY="0.0"
          sizeX="21.2"       sizeY="10.6"       thickness="0.030"
          radLength="93.660734"
```

```
        />
        <sensitive   ID="0"
          positionX="0.00"   positionY="0.00"   positionZ="0.00"
          sizeX="21.2"       sizeY="10.6"       thickness="0.02"
          npixelX="1152"     npixelY="576"
          pitchX="0.018402"  pitchY="0.018402"  resolution="0.0045"
          rotation1="-1.0"   rotation2="0.0"
          rotation3="0.0"    rotation4="-1.0"
          radLength="93.660734"
        />
      </layer>
      <!--Eudet-Plane 1 - EUD1 -->
      <layer>
        <ladder   ID="1"
          positionX="0.00"   positionY="0.00"   positionZ="83.00"
          rotationZY="0.0"   rotationZX="0.0"   rotationXY="0.0"
          sizeX="21.2"       sizeY="10.6"       thickness="0.030"
          radLength="93.660734"
        />
        <sensitive       ID="1"
          positionX="0.00"   positionY="0.00"   positionZ="83.0"
          sizeX="21.2"       sizeY="10.6"       thickness="0.02"
          npixelX="1152"     npixelY="576"
          pitchX="0.018407"  pitchY="0.018402"  resolution="0.0045"
          rotation1="-1.0"   rotation2="0.0"
          rotation3="0.0"    rotation4="-1.0"
          radLength="93.660734"
        />
      </layer>
    </layers>
  </detector>
 </detectors>
</gear>
```

The idea of separating the structural and sensitive originates from the fact that many sensors are thinned down and the remaining structural material is thicker. Hence, this ability is reflected in splitting the description into the ladder and sensitive part. In reality however, most processors will simply ignore this fact and users will be confused by the additional degree of freedom.

Another downside of the implementation is that the number of pixels, their pitch and the total active area should be correlated. This however is not reflected in this implementation. Providing mismatching number will merely result in wrong derived positions, but not cause an error or warning.

The geometry parameter, which links each module to an extended geometry descrip-

tion, is decoupled from the layers it relates to. As this has been introduced in retrospect, this is a limitation imposed by the initial formulation of the syntax.

B.2. TrackerPlanes style layout

A more novel approach, attempting to clarify the fields in the GEAR file, is the TrackerPlanes layout.

One new feature is that users can define materials which then can be used for the layers. Unfortunately, the material node in the XML file needs a value for the radiation and integration length. However, these values can be derived from the material properties, which happens if the initial values are set to zero.

```
<gear>
  <!-- new telescope geometry description -->
  <global detectorName="EUTelescope"/>
  <BField type="ConstantBField" x="0.0" y="0.0" z="0.0"/>

  <materials>
    <material name="Silicon" A="28" Z="14"
            density="2.329" radLength="0" intLength="0"/>
  </materials>
  <detectors>
    <detector name="SiPlanes" geartype="TrackerPlanesParameters">
      <LayoutID ID="225"/>
      <LayoutNumberOfLayers number="2"/>
      <layers>
        <layer ID="0" info="First tel. plane">
          <active ID="0" geometry="Mimosa26.so"
                sizeX="21.2" sizeY="10.6"
                thickness="0.030" material="Silicon"
          />
        </layer>
        <layer ID="1" info="Second tel. plane" positionZ="83.00">
          <active ID="1" geometry="Mimosa26.so"
                sizeX="21.2" sizeY="10.6"
                thickness="0.030" material="Silicon"
          />
        </layer>
      </layers>
    </detector>
  </detectors>
</gear>
```

EUTelescope Test Cases

As mentioned in Section 8.3, there are several Docker containers provided and used to execute the test cases. In particular, the following ten distributions, listed in Table C.1, are available.

Distribution	Package Manager	Comments
ArchLinux	pacman	rolling release
CernCentOS 7	yum	
Debian9	apt	
Fedora 26	dnf	
Fedora 28	dnf	
ScientificLinux 6	yum	baseline, using devtoolset
Ubuntu 16	apt	
Ubuntu 17	apt	
Ubuntu 18	apt	
openSUSE tumbleweed	zypper	rolling release

Table C.1.: List of available Docker containers.

The following test cases, listed in Table C.2, have been implemented. They use two different examples, shipped with a default EUTelescope installation, namely the *aconite-4chipLocal* example, which is an example with ATLAS pixel four-chip modules and the *gbl_local* example. The initial one exploits the extended geometry description within EUTelescope and can therefore validate the correct functioning of it. It also tests and validated the interplay of Mimosa26 and ATLAS pixel data. Moreover, it uses the Deterministic Annealing Filter (DAF) fitter. Contrary to that, the *gbl_local* example uses the GBL track fit. While some of the tests are repeated within the *gbl_local* test

cases, there is as little duplication as possible to keep the test cases maintainable.

Name	Example Used	Validation of
test_01-1_geoboxinitialisation	aconite-4chipLocal	Box of sensor in TGeo is initialised correctly
test_01-2_advancedgeometry	aconite-4chipLocal	Loading of shared extended geometry libraries
test_01-3_noisypixel_m26	aconite-4chipLocal	Number of noisy pixels on Mimosa26 sensors
test_02-fullconverter	aconite-4chipLocal	Execution of full converter step, writing output file
test_03-1_readnoisypixel_m26	aconite-4chipLocal	Reading back noisy pixel database for Mimosa26
test_03-2_readnoisypixel_apix	aconite-4chipLocal	Reading back noisy pixel database for ATLAS pixels (expected to be empty, but it must exist)
test_03-3_clustercount_m26	aconite-4chipLocal	Cluster count on Mimosa26
test_03-4_clustercount_apix	aconite-4chipLocal	Cluster count on ATLAS pixel
test_03-5_noisyclustermasker_m26	aconite-4chipLocal	Masking of correct number of noisy clusters
test_03-6_noisyclusterremover_m26	aconite-4chipLocal	Removing correct number of noisy Mimosa26 clusters
test_03-7_noisyclusterremover_apix	aconite-4chipLocal	Executing noisy cluster removal for ATLAS pixel (no noisy pixels, hence no removal expected)
test_04-fullclustering	aconite-4chipLocal	Execution of full clustering step, writing output file
test_05-1_prealigner	aconite-4chipLocal	Output of pre-alignment processor
test_05-2_correlator	aconite-4chipLocal	Output of correlator processor
test_06-fullhitmaker	aconite-4chipLocal	Execution of full hitmaker step, writing output file
test_07-1_daf_scatter_init	aconite-4chipLocal	The deterministic annealing filter (DAF) fitter initialises material correctly
test_07-2_daf_setup_init	aconite-4chipLocal	The DAF fitter initialises the set-up correctly, i.e. resolutions and sensors
test_07-3_daf_track_count	aconite-4chipLocal	DAF fitter finds a given amount of tracks
test_07-4_mille_track_accept	aconite-4chipLocal	Mille(pede) accepts agiven number of tracks from the DAF fitter
test_07-5_pede_call	aconite-4chipLocal	The Pede executable is called via pthreads
test_07-6_pede_result	aconite-4chipLocal	Results reported back by Pede are consistent
test_08_fullalign	aconite-4chipLocal	Execution of full align step, writing GEAR and log file
test_09-1_daf_setup_init	aconite-4chipLocal	Same as before, but also validates exclusion of DUTs in final track fit
test_09-2_daf_track_count	aconite-4chipLocal	Same as before now with final alignment
test_09-3_daf_track_stats	aconite-4chipLocal	Checks if DAF output like chi-square is consistent

test_01-1_noisypixel_m26	gbl_local	Number of noisy pixels on Mimosa26 sensors
test_02-fullconverter	gbl_local	Execution of full converter step, writing output file
test_03-1_clustercount_m26	gbl_local	Cluster count on Mimosa26
test_04-fullclustering	gbl_local	Execution of full clustering step, writing output file
test_05-1_prealigner	gbl_local	Output of pre-alignment processor
test_05-2_correlator	gbl_local	Output of corellator processor
test_06-fullhitmaker	gbl_local	Execution of full hitmaker step, writing output file
test_07-1_track_count_first_iteration	gbl_local	Initial GBL alignment iteration, number of found tracks
test_07-2_pede_first_iteration	gbl_local	Derived alignment from first Millepede iteration
test_07-3_track_count_second_iteration	gbl_local	Second GBL alignment iteration, number of found tracks
test_07-4_pede_second_iteration	gbl_local	Derived alignment from second Millepede iteration

Table C.2.: An overview of the implemented test cases for EUTelescope.

Emitter Coupled Logic Circuit

D.1. The Circuit

Shown in Figure D.1 is the initial attempt to impose gating without the NIM set-up as discussed in Section 9.2.

The section of the circuit outlined by the green (full) ellipse is the section deriving the gate pulse. A delay line between U$3 and U$4 delays the signal to the V_+ input of the comparator. If the 40 MHz clock is used as an input, the comparator will produce temporally correlated pulses, their length determined by the mentioned delay line.

Below, enclosed by the red (dashed) oval, the same type of comparator is used to produce an output pulse if there is a negative pulse from a PMT. This is done by comparing the PMT pulse, going to V_-, to a reference voltage on V_+.

Ultimately, both pulses are ANDed (logical AND) together. This is shown in the orange (dotted) circle.

The problems which arose with that circuit are that to derive a square pulse in the order of a (or some) ns, the circuit must be able to handle the high frequency components in that pulse. As a rule-of-thumb, for a square wave one needs to consider frequencies of at least three times the square pulse frequency. This pushes the frequencies of this gating circuitry to above a GHz. In this frequency domain, designing circuits is not an easy task, and ultimately improper termination and not perfectly adjusted trace impedances led to reflections which deteriorated the signals processed.
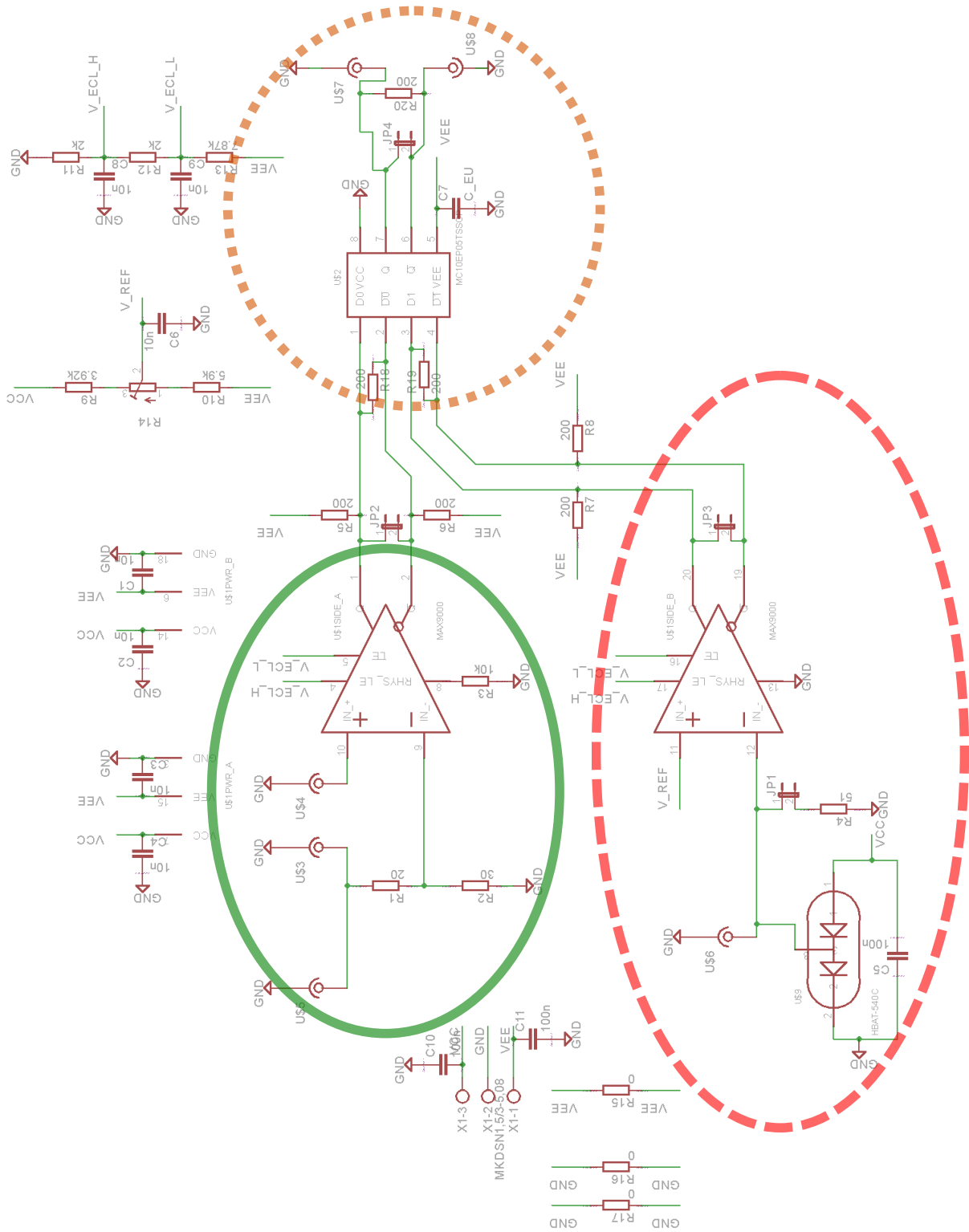
Figure D.1.: The NECL circuit, an attempt for the gating hardware for in-time measurements.

Supplemental In-Time Validation Plots

E.1. Gate Signal Shape

The further oscilloscope measurements for the impact of the delay on the gate width are shown below.
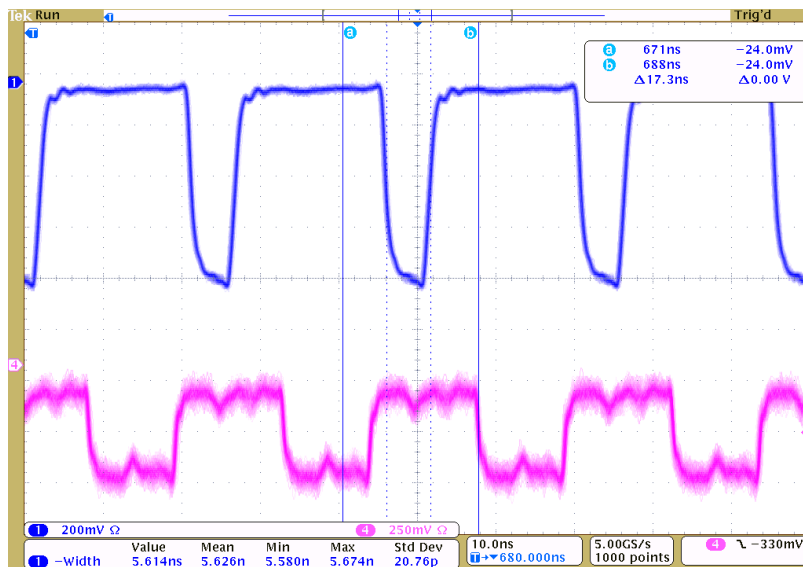

Figure E.1.: Measurement of the gate width with an additional 5 ns delay with respect to the measurement shown in Fig. 9.5.

In Figure E.1 an additional 5 ns of delay is introduced. The negative pulse width remains at (5.61 ± 0.02) ns. With a total of 11 ns delay, given in Figure E.2, a negative
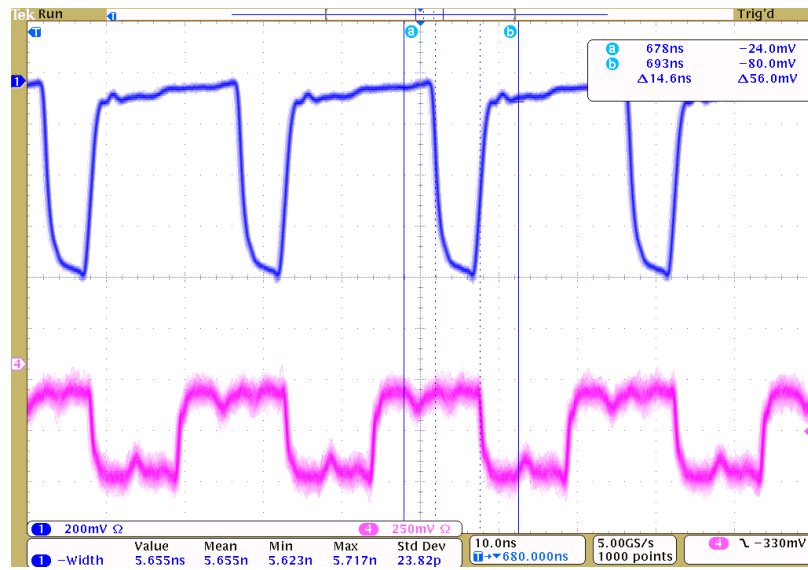
pulse width of (5.56 ± 0.02) ns is measured.

Figure E.2.: Measurement of the gate width with an additional 11 ns delay with respect to the measurement shown in Fig. 9.5.