DISSERTATION

submitted to the

Combined Faculty of Mathematics, Engineering and Natural Sciences

of Heidelberg University, Germany

for the degree of

Doctor of Natural Sciences

Put forward by

LIQING QIN

born in Shanxi, China

Oral examination: 26.11.2025

Scintillating Fiber-based Beam Profile Monitor for Particle Therapy

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

Every step counts.

Abstract

A real-time beam profile monitor (BPM) has been developed for particle therapy and evaluated at the Heidelberg Ion Beam Therapy Center (HIT). The system is designed to provide real-time measurements of the ion beam position and width, thereby enabling precise radiation delivery in future clinical applications.

The BPM measures the one-dimensional projections of the beam using two layers of 250 µm scintillating fibers, which are read out by Hamamatsu S11865-64 photodiode arrays equipped with amplifier and integration circuits. The photodiode channels have dimensions of 0.7 mm (w) × 0.8 mm (h) with a pitch of 0.8 mm. The signals from the photodiodes are digitized on a custom PCB hosting voltage regulators and ADCs. Both the photodiodes and ADCs are controlled by a MAX 10 FPGA, which, in addition to signal control, performs ADC data packaging and beam profile reconstruction before transmitting the processed data via Ethernet to a PC equipped with dedicated data acquisition software.

This thesis investigates the signal characteristics of the SciFi-BPM, develops a tailored beam profile reconstruction algorithm, and implements it on the FPGA, enabling real-time determination of the beam position and width. Beam tests at HIT with protons and carbon ions demonstrated that the performance is sufficient for the majority of beam settings, specifically achieving an overall resolution better than 0.05 mm for the beam position and 0.12 mm for the beam width at high intensities.

Zusammenfassung

Ein Echtzeit-Strahlprofilmonitor (BPM) wurde für die Partikeltherapie entwickelt und am Heidelberger Ionenstrahl-Therapiezentrum (HIT) evaluiert. Das System ist darauf ausgelegt, die Position und Breite des Ionenstrahls in Echtzeit zu messen und dadurch eine präzise Strahlapplikation in zukünftigen klinischen Anwendungen zu ermöglichen.

Der BPM erfasst die eindimensionalen Projektionen des Strahls mithilfe von zwei Lagen aus 250 µm szintillierenden Fasern, die von Hamamatsu S11865-64 Photodiodenarrays mit Verstärker- und Integrationsschaltungen ausgelesen werden. Die Photodiodenkanäle haben Abmessungen von 0.7 mm (B) × 0.8 mm (H) bei einer Pitch von 0.8 mm. Die Signale der Photodioden werden auf einer kundenspezifischen Leiterplatte mit Spannungsreglern und ADCs digitalisiert. Sowohl die Photodioden als auch die ADCs werden von einem MAX 10 FPGA gesteuert, das neben der Signalsteuerung auch die Verpackung der ADC Daten und die Rekonstruktion des Strahlprofils übernimmt, bevor die verarbeiteten Daten über Ethernet an einen PC mit spezieller Datenerfassungssoftware übertragen werden.

In dieser Arbeit werden die Signalcharakteristika des SciFi-BPM untersucht, ein speziell angepasster Algorithmus zur Strahlprofilrekonstruktion entwickelt und auf einem FPGA implementiert, wodurch eine Echtzeitbestimmung der Strahlposition und -breite ermöglicht wird. Strahltests am HIT mit Protonen- und Kohlenstoffionenstrahlen zeigten, dass die Leistung für die Mehrheit der Strahleinstellungen ausreichend ist. Insbesondere wurde eine Gesamtauflösung von besser als 0.05 mm für die Strahlposition und 0.12 mm für die Strahlbreite bei hohen Intensitäten erreicht.

Preface

The measurements presented in this work have not yet been published in a peer-reviewed journal. A submission to Nuclear Instruments and Methods in Physics Research (NIM) or Journal of Instrumentation (JINST) is planned for later this year.

The SciFi-BPM detector was developed within a small collaboration. This work was conducted under the supervision of Dr. Blake Leverington, who designed the detector. Dr. Michał Dziewiecki, formerly at Heidelberg University, moved to GSI before the author began this work. He developed the readout electronics, including the ADC boards, the original FPGA firmware, and the DAQ software. A prototype test was carried out in 2016 and published in 2018 using an LHCb-style fiber plane by Dr. Blake Leverington and Dr. Michał Dziewiecki [1].

A PhD student, Qian Yang, was involved in this project between October 2020 and April 2023. She analyzed the detector behavior under a magnetic field, which is not included in this thesis.

Dr. Blake Leverington has been continuing the development of the scintillating fiber planes with the aim of reducing the material budget and increasing the active area. In addition, he designed the mechanical frame to combine two fiber planes for measuring the one-dimensional projections in both horizontal and vertical directions, referred to as XY stations. Dr. Blake Leverington constructed three copies of such XY stations.

Building on this foundation, the author further investigated the signal characteristics and implemented the reconstruction algorithm within the electronics framework developed by Dr. Michał Dziewiecki. In addition, the author evaluated the performances of different types of fiber planes developed by Dr. Blake Leverington with different test beam measurements.

The test beam measurements were carried out eight times at HIT during the author’s PhD, each as an overnight session. Each measurement involved preparation of the detector in the laboratory, transport to HIT, setup, data taking, and subsequent return of the detector. These measurements were conducted in collaboration with Qian Yang during the first four sessions, with Dr. Blake Leverington throughout all eight, and with the accelerator team at HIT during all of them. The author analyzed six of the eight test beam measurements. In this thesis, the test beam results from August 2024 are presented, focusing on the performance of a relatively advanced type of fiber plane, namely the full-size, two-layer glueless fiber plane.

The author of this thesis is primarily responsible for the analysis, algorithm design, implementation, verification, and performance evaluation presented herein. All analysis steps reported in this work were carried out by the author.

This work was supported in part by the Deutsche Forschungsgemeinschaft (German Research Foundation, DFG) - Projektnummer(419255448).

Contents

Chapter 1

Introduction

Cancer is a major public health concern worldwide. Since 1903, when William Henry Bragg discovered the Bragg peak phenomenon [2], particle therapy has attracted significant attention due to its ability to deliver highly localized radiation to destroy cancer cells.

The central challenge of particle therapy is to deliver the prescribed radiation precisely to the target. Modern raster-scanning techniques employ a narrow pencil beam to paint the tumor volume [3], enabling flexible shaping with high spatial accuracy. Real-time monitoring of the beam's position, width, and intensity is therefore essential, as it directly determines the quality of radiation delivery.

At present, only 15 ion therapy centers and 121 proton therapy centers are in operation worldwide [4]. Among them, the Heidelberg Ion Beam Therapy Center (HIT) is the first dedicated hospital-based facility in Europe capable of treating patients with both protons and heavier ions [5]. At HIT, the pencil beam intensity is continuously monitored online using ionization chambers (ICs), while its position and width are measured with multi-wire proportional chambers (MWPCs).

The scintillating fiber-based beam profile monitor (SciFi-BPM) is proposed as the next-generation detector for position and width measurements at HIT. Its scalability and adaptability also make it suitable for implementation in particle therapy centers worldwide for ion beams.

The aim of this dissertation is to develop the SciFi-BPM, with a focus on enabling real-time reconstruction of the pencil beam position and width.

Chapters 2 and 3 provide the background and motivation for this work. Chapter 2 introduces the physics of particle therapy, while Chapter 3 discusses its role in cancer treatment, together with a detailed description of HIT and its beam characteristics. The current MWPC-based monitoring system and its limitations are presented, along with the general requirements for a new BPM.

Chapters 4, 5, 6, and 7 form the core of this dissertation, where the main work is presented. Chapter 4 introduces the principle, setup, and functionality of the SciFi-BPM, while Chapter 5 describes the signal characteristics and the reconstruction algorithms derived from them. Chapter 6 details the FPGA implementation of the full reconstruction chain and verifies its functionality, demonstrating high accuracy. Chapter 7 reports on beam test measurements at HIT and evaluates the performance of the SciFi-BPM. Finally, Chapter 8 provides a summary and an outlook on the future development of the detector.

Part I

Motivation

Chapter 2

Physics of Particle Therapy

The terminology used in particle therapy differs between medical physics and particle physics due to the distinct focus of each field. The following definitions are adopted throughout this thesis to maintain consistency:

- **Radiation therapy** encompasses all forms of cancer treatment using radiation.
- **Particle therapy**, also known as hadron therapy, is a subset of radiation therapy and includes only proton therapy and ion beam therapy:

  $$\text{Particle Therapy} = \text{Proton Therapy} \cup \text{Ion Beam Therapy}$$

  It explicitly excludes other radiation modalities:

  $$\text{Particle Therapy} \cap \{\text{Photon Therapy}, \text{ Neutron Capture Therapy}\} = \emptyset$$

- **Ion beam therapy** refers exclusively to the treatment using heavier ions (e.g., carbon, helium, oxygen) and does not include proton therapy:

  $$\text{Proton Therapy} \cap \text{Ion Beam Therapy} = \emptyset$$

Careful monitoring of the pencil beam with a dedicated detector (Beam Profile Monitor, BPM) is essential in particle therapies. The primary goal of particle therapy is to deliver the prescribed dose to the target tumor while minimizing exposure to the surrounding health y tissues. Section 2.1 provides a comprehensive overview of the properties of photon, proton, and ion beams as they traverse materials, focusing on their lateral and longitudinal dose distributions, as well as their biological effectiveness in destroying tumor cells. It explains the theoretical foundation for the superior potential of proton and ion therapy in treating tumors compared to photon therapy. Section 2.2 introduces the dose delivery methods of particle therapy, highlighting the active scanning dose delivery system and its associated challenges, in which the beam profile monitor plays a crucial role.

2.1 Physics of Particle Therapy

The aim of all radiation therapies is to destroy cancer cells using radiation while minimizing damage to healthy tissue. The amount of radiation is quantified by the dose, as described in Subsection 2.1.1. The dose can be delivered using photons in photon therapy, protons in proton therapy, or ion beams in ion therapy. Subsections 2.1.2, 2.1.3, and 2.1.4 explain the depth dose distribution for photons, protons, and heavier ions, as well as the dose tail and the lateral dose distribution—referred to as the penumbra—for protons and heavier ions. Subsection 2.1.5 discusses the biological effects of photons, protons, and heavier ions. These physical and biological properties are essential for understanding the advantages of proton and ion therapy over photon therapy, as well as the distinctions between proton and ion therapy.

2.1.1 Dose

Dose dE/dm is the amount of energy deposited in the tissue per unit mass. Its unit is Gray(Gy) or Rad.

$$1\,\mathrm{Gy} = 1\,\mathrm{J\,kg}^{-1} = 100\,\mathrm{rad} \tag{2.1}$$

The purpose of radiation therapy is to deliver the prescribed amount of dose to the target tumor while minimizing the dose exposure to the surrounding healthy tissues. A standard total irradiation dose, for instance in the treatment for lung cancer, is 60 Gy in 30 fractions. If the dose to the tumor is insufficient, it may fail to effectively kill or suppress the tumor, leading to local and/ or regional recurrence [6]. On the other hand, excessive dose to healthy tissues or organs can cause complications, such as cardiopulmonary toxicity after treatment for lung cancer [7], which will reduce the patient's quality of life and threaten their health. Therefore, precise dose delivery is crucial.

Doses can be delivered using different types of radiation, such as photons, electrons, protons, neutrons, and heavier ions. In particle therapy, doses are delivered using protons or heavier ions. By leveraging the energy deposition properties of charged heavy particles, particle therapy can deliver doses to tumors with high precision.

2.1.2 Specific Energy Loss and Bragg Peak

Particle therapy delivers protons or heavier ions to target tumors. As these ions pass through both healthy and malignant tissues, they deposit energy, leading to DNA strand breaks or enhancing the immune response, ultimately resulting in tumor cell death. The energy loss per distance in a material is called the specific energy loss, denoted as dE/dx. The specific energy loss dE/dx for proton or ions is described empirically by the Bethe-Bloch formula [8]:

$$-\frac{dE}{dx} = \rho K \frac{Z}{A}\frac{z^2}{\beta^2}\left[\ln\frac{2m_e c^2\beta^2\gamma^2}{I} - \beta^2\right] \tag{2.2}$$

The unit of dE/dx is $\mathrm{MeV}\cdot\mathrm{cm}^{-1}$. z, β and γ are the charge, velocity relative to the speed of light and Lorentz factor of the incident ions, respectively. Z, A, ρ and I are the atomic number, atomic mass, mass density and mean excitation energy of the material, respectively. K is a constant, given by

$$K = 4\pi N_A r_e^2 m_e c^2 = 0.307\,\mathrm{MeV}\cdot\mathrm{cm}^2/\mathrm{mol} \tag{2.3}$$

This is a simplified version of Bethe-Bloch formula, applicable to ions with $M \gg \gamma m_e$ and $\beta\gamma$ between 0.3 to 3 [8]. It fits the scenario of particle therapy. At HIT, the $\beta\gamma$ of ion beams is between 0.3 to 1.1.

When $\beta\gamma$ is smaller than 3, the dE/dx is dominated by the $1/\beta^2$ term. As the velocity of the ions decreases, the energy loss increases, resulting in the majority of the ions' kinetic energy being deposited near the end of their range. This phenomenon is known as the Bragg peak.

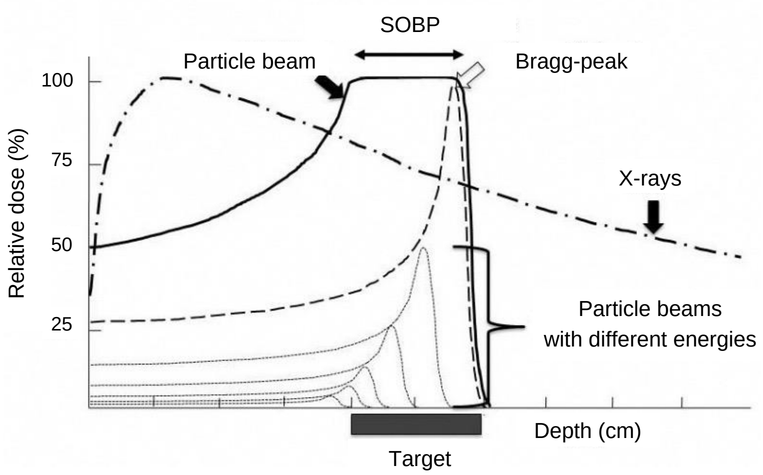


FIGURE 2.1: The depth-dose distribution of X-ray and ion beams [6]. The Bragg peaks of beams with different energies contribute to the spread-out Bragg peak(SOBP), which covers the target region.

Figure 2.1 illustrates the depth-dose distribution of X-ray and particle beams. The Bragg peaks of ion beams at different energies are combined to create a flat dose plateau region, known as the spread-out Bragg peak (SOBP), which can be precisely tailored to cover the target region. Compared to X-ray beams, the SOBP delivers higher doses to the target region while minimizing exposure to normal tissues both before and after the target region. This characteristic makes particle therapy especially advantageous for treating deep-seated tumors located near critical structures.

Figure 2.2 compares the dose distributions of X-ray and proton beams for the treatment of a patient with head and neck cancer. The proton beam delivers a tightly focused dose around the tumor, whereas the X-ray beam spreads the dose over a broader area, affecting both the tumor and surrounding healthy tissues. Thus, proton beam therapy allows for dose escalation to the tumor while sparing normal tissues, potentially improving local control and survival rates while reducing side effects [9].

2.1.3 Fragmentation

Figure 2.3 illustrates the relative dose-depth distributions of protons and carbon ions. Unlike protons, carbon ions exhibit a dose tail beyond the Bragg peak. This occurs because, for hadrons with mass $Z > 1$, ions can fragment into lighter ions upon interaction with nuclei. The nuclear fragments generate forward-peaked secondary beams, which contribute to the dose tail.

This dose tail can potentially expose surrounding healthy tissues to radiation. Therefore, treatment planning must account for heavy-ion fragmentation to minimize its impact on healthy tissues. On the other hand, this phenomenon provides an opportunity for real-time therapy monitoring, as the detection of fragmentation products enables monitoring of the ion beam range and beam spot position [11].

2.1.4 Penumbra

Penumbra [95-50%] is defined as the lateral width of the dose band at the field edge where the dose decrease from 95% to 50% of the maximum dose. It is an important

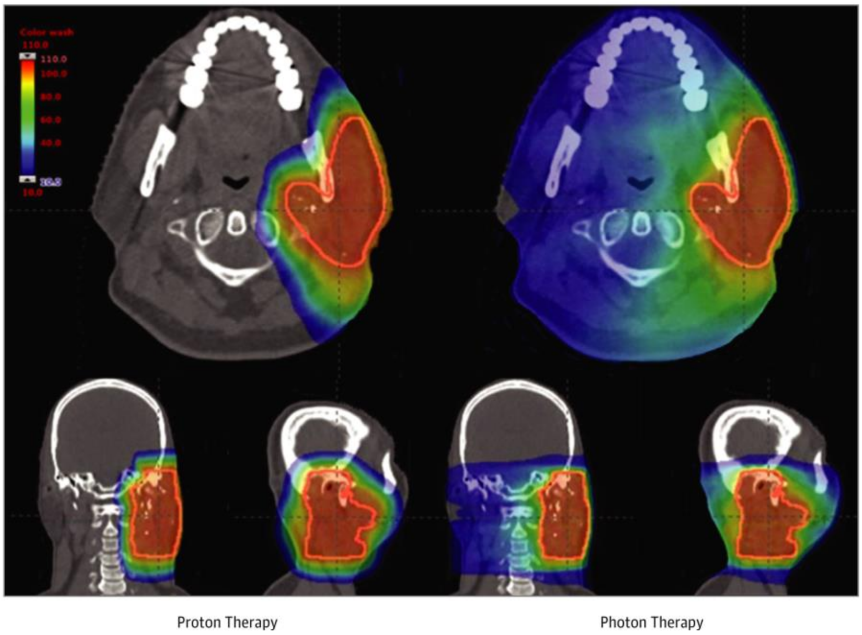


FIGURE 2.2: Representative dose distributions of protons and photons for a patient with head and neck cancer [10]. The tumors are indicated by the red circles. Dose levels are color-coded, with red representing the highest dose and blue the lowest.

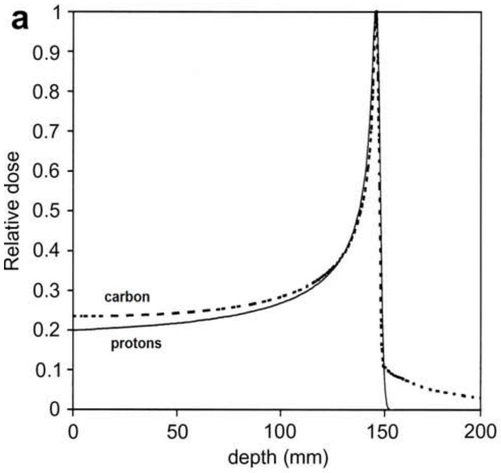


FIGURE 2.3: The depth-relative dose distributions of protons and carbon ions [12].

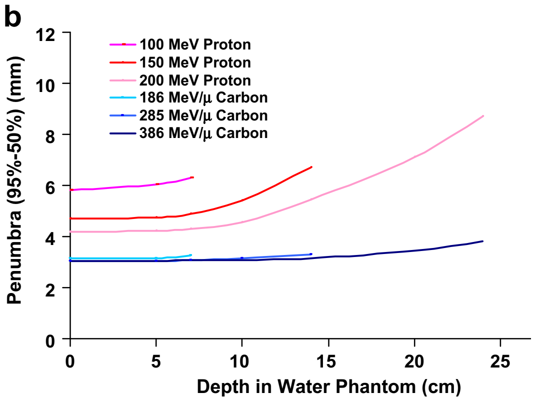


FIGURE 2.4: Calculated 95-50% penumbra widths along the beam vs depths for proton and carbon ion pencil beams at GSI [12]. The initial penumbra at 0 cm depth reflects the difference in beam width across various energies.

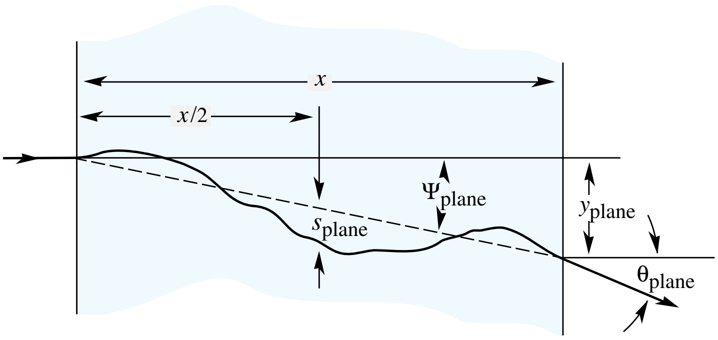


FIGURE 2.5: Sketch for multiple scattering [8]. The particle is incident horizontally from the left. After a distance x, it is deflected by an angle θ_{plane}.

parameter in radiation therapy, as it determines the sharpness of the dose gradient. A smaller penumbra indicates a steeper dose gradient, which is desirable for reducing the dose exposure to surrounding normal tissues.

Figure 2.4 compares the penumbra widths of proton and carbon ion beams. The penumbra of carbon ions is smaller than that of protons. This is related to multiple scattering.

When a parallel beam penetrates a material, it undergoes numerous small-angle scatterings by Rutherford scattering, as shown in Figure 2.5. This phenomenon is known as multiple scattering.

This scattering angle distribution is well-represented by the theory of Molière [13]. The RMS width θ_0 for the projected scattering angle is given by [8]:

$$\theta_0 = \theta_{\text{plane}}^{\text{RMS}} = \frac{13.6\text{MeV}}{c\beta^2\gamma}\frac{z}{m}\sqrt{\frac{x}{X_0}}\left[1 + 0.038\ln\left(\frac{xz^2}{X_0\beta^2}\right)\right] \tag{2.4}$$

where p, βc, z, m are the momentum, velocity, charge number, and mass of the incident particle, X_0 is the radiation length of the material. The X_0 for polystyrene is 41.31 cm, for PMMA is 34.07 cm, and for air is 3.039×10^4 cm [14]. These values are used to estimate the multiple scattering by BPM materials in later sections.

Since the RMS scattering angle increases with penetration distance x, the penumbra of both protons and carbon ions increases with depth. However, carbon ion

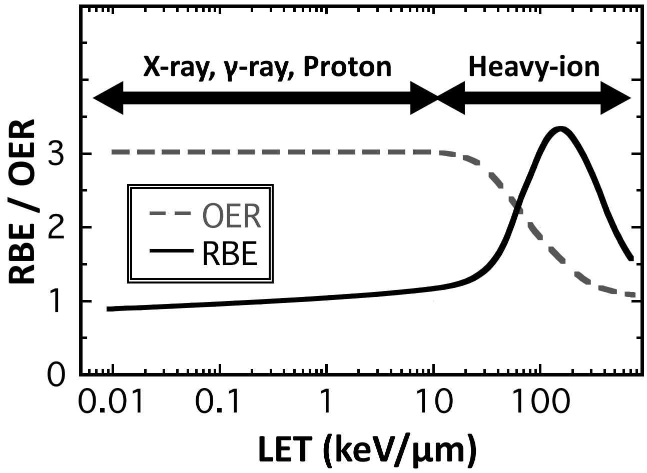


FIGURE 2.6: Relationship between relative biological effectiveness(RBE), oxygen enhancement ratio(OER) and LET [6].

beams maintain a smaller multiple scattering angle than proton beams. This is due to carbon's lower charge-to-mass ratio and the fact that for the same range in tissue, carbon ions require roughly twice the energy per nucleon as protons [12]. This results in smaller penumbra for carbon ions, leading to better dose confinement and precision in carbon ion beam therapy.

This trend is not limited to carbon ions; other ion species used in ion beam therapy, such as helium, oxygen, and neon ions, exhibit similar behavior. Due to their higher mass and lower charge-to-mass ratio, these ions experience less multiple scattering compared to protons, leading to a smaller penumbra and better dose localization. As a result, ion beam therapy, in general, offers superior lateral dose confinement compared to proton therapy, making it a highly precise modality for treating deep-seated tumors while sparing surrounding healthy tissue.

2.1.5 Linear Energy Transfer, Relative Biological Effectiveness, and Oxygen Enhancement Ratio

According to the Bethe-Bloch formula 2.2, the specific energy loss, dE/dx, is proportional to z^2, where z represents the charge of the ion. This means that for ions with the same velocity, the energy loss of heavy ions is significantly higher than that of protons in the same material. In medical physics, dE/dx is referred to as the linear energy transfer (LET), denoted as dE_l/dx.

Figure 2.6 illustrates the relationship between LET and two key parameters related to the efficacy of cancer cell eradication: the relative biological effectiveness (RBE) and the oxygen enhancement ratio (OER). High LET is associated with greater effectiveness in killing cancer cells, as the DNA damage induced by high-LET radiation is more difficult for cells to repair. The relative biological effectiveness (RBE) is defined as the ratio of the dose from reference photon radiation to that from particle radiation required to produce the same biological effect. The RBE increases with LET and reaches its maximum at approximately 100–200 keV$\cdot\mu$m^{-1}. Clinically, the standard RBE for protons is 1.1, while the RBE for carbon ions is approximately 3.0 [6]. The oxygen enhancement ratio (OER) is defined as the ratio of the dose required to achieve the same biological effect in the absence of oxygen to that required in the presence of oxygen. The OER is approximately 2.5–3.0 at low LET and decreases to

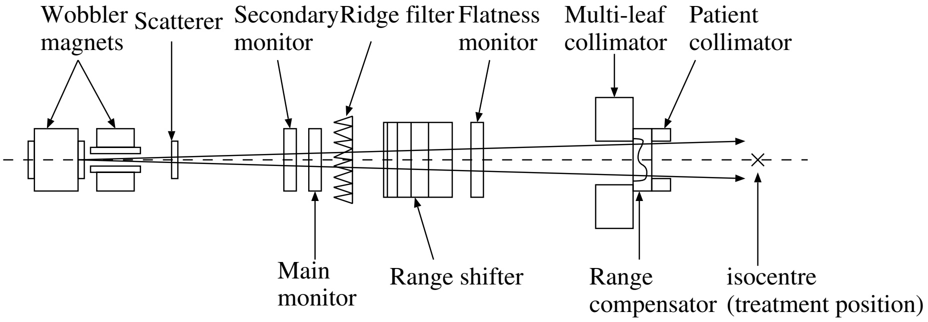


FIGURE 2.7: A schematic layout of a passive scattering dose delivery system [18].

1 at high LET. A low OER is generally considered beneficial, because it reduces the dependence of radiation effectiveness on tissue oxygenation.

Comparing with proton therapy, ion beam therapy, with high RBE and low OER, is particularly effective at killing cancer cells, especially those in hypoxic regions or those classified as radioresistant [6, 15].

2.1.6 Discussion

In conclusion, considering both physical and biological properties, ion beam therapy is more precise and effective than proton therapy, while proton therapy is more precise and effective than photon therapy.

However, in practice, achieving precise target volume coverage in particle therapy is more challenging than in photon therapy. This is because protons and ions are more sensitive to organ motion and patient setup errors due to their steep dose fall-off beyond the Bragg peak [16]. Fully realizing the benefits of particle therapy depends on continuous technological advancements, particularly in enhancing dose delivery precision.

At the same time, proton therapy is more complex and costly than photon therapy due to the need for a specialized accelerator, dose delivery and monitoring system, advanced imaging, shielding, and precise patient positioning systems. In particular, ion beam therapy requires higher beam energy than proton therapy, resulting in larger and more expensive accelerators compared to those used for proton treatment.

To facilitate the broader adoption of particle therapy, the development of more compact and affordable treatment systems and stronger clinical evidence demonstrating its advantages over photon therapy are essential [9, 17].

2.2 Dose Delivery in Particle Therapy

The beams for particle therapy are typically generated by cyclotrons or synchrotrons. Cyclotrons provide mono-energetic continuous beams, while synchrotrons deliver pulsed beams with adjustable energy for each pulse. Both types of beams must be shaped into a Spread-Out Bragg Peak (SOBP) to adequately cover the tumor. This can be achieved through two main techniques: passive modulation of the primary beam or scanning tumor slices with pencil beams of varying energies. These are known as passive scattering and active scanning, respectively.

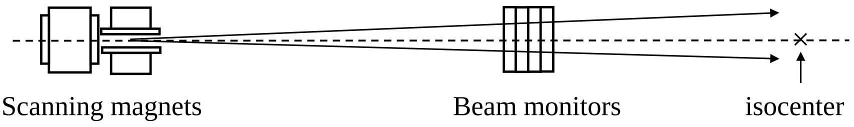


FIGURE 2.8: A schematic layout of an active scanning dose delivery system.

Figure 2.7 illustrates a typical passive scattering system. In this system, the transverse beam spread is achieved using a wobbler magnet and a scattering device in the beamline. The collimator shapes the beam to conform to the tumor's contour. Longitudinally, a range shifter adjusts the depth of the beam in the patient by degrading the beam energy, while a range compensator fine-tunes the beam depth to match the tumor's depth variations. To create a SOBP, a ridge filter is used to broaden the energy distribution. A major drawback of the passive scattering system is the generation of secondary radiation—particularly neutrons—caused by the interaction of the beam with the devices in the beamline, which may increase the risk of secondary cancers. Additionally, this method requires patient-specific hardware, adding complexity to treatment planning and quality assurance.

Figure 2.8[1] shows an example of an active scanning system. In this system, the beam's energy is adjusted by the accelerator before it reaches scanning magnets, while the scanning magnets steer the beam transversely to irradiate the tumor region slice by slice. HIT employs the active scanning system. More details about the active scanning system at HIT will be introduced in subsection 3.3.3.

Compared with the passive scattering system, the active scanning is more advantageous in several aspects. The beamline contains only beam monitors, reducing neutron production and thus minimizing secondary radiation exposure. At the same time, it reduces the penumbras with the thin pencil beams and improves dose localization. It is more flexible for treating irregularly shaped tumors than the passive scattering method. In addition, it allows for daily adaptive treatment planning without the need for patient-specific hardware, making it more efficient.

At the same time, the active scanning system has some limitations. It requires a more sophisticated accelerator capable of delivering narrow pencil beams with different energies. Moreover, since the tumor is irradiated in a sequential manner rather than simultaneously, tumor motion during treatment must be carefully managed [19].

Clinical studies have shown mixed results depending on the cancer type. For example, the active scanning method has been found superior to the passive scattering method for pancreatic cancer due to reduced secondary irradiation [20]. In contrast, for breast cancer, both methods have been shown to be equally effective [21].

[1] The figure was prepared by the author, using a plotting style similar to Fig 2.7 [18].

Chapter 3

Practice of Particle Therapy: From History to Beam Profile Monitor

Chapter 2 introduces the physics and dose delivery concepts of particle therapy. This chapter focuses on the practical aspects, presenting the actual application scenario of the beam profile monitor (BPM). Section 3.1 provides a brief history of particle therapy. Section 3.2 outlines the role of particle therapy in cancer treatment. Section 3.3 provides a detailed introduction to the Heidelberg Ion Therapy Center (HIT), the facility where our BPM is tested and intended to be implemented. Section 3.4 discusses the multi-wire proportional chambers (MWPCs) currently used at HIT as BPMs, their limitations, the requirements for the new BPMs and several alternatives.

3.1 Short Review of Particle Therapy

In 1903, William Henry Bragg discovered the Bragg peak phenomenon [2], laying the foundation for particle therapy. However, it was not until 1946 that Robert Wilson proposed the use of protons and ion beams for cancer treatment, recognizing their potential for precise dose delivery [22].

The first particle therapy treatment was performed in 1954 at the Lawrence Berkeley National Laboratory (LBL), where the proton beams produced by the 4.6-meter cyclotron were used to treat brain tumors. However, the Bragg peak was not utilized back then, as the essential peripheral technologies, such as CT imaging, were not available [17].

Over the following decades, advancements in imaging, accelerator technology, dose delivery, and treatment planning transformed particle therapy from an experimental technique to a clinically viable treatment option. The first hospital-based proton therapy center was established at Loma Linda University Medical Center in 1983. It featured a synchrotron-based system capable of delivering 0.5 Gy protons per minute to a 20 cm-diameter field with a 9 cm-long Spread-Out Bragg Peak (SOBP). This facility incorporated beam-spreading devices to produce the uniform treatment field and a range modulator to create the SOBP, allowing treatment of larger tumor regions [23].

Meanwhile, in 1994, Japan built the first dedicated hospital-based ion therapy center, the Heavy Ion Medical Accelerator in Chiba (HIMAC). HIMAC introduced a synchrotron-based system with a maximum dose rate of 5 Gy per minute, employing wobbler scanning, ridge filter, and range shifter to passively shape the dose distribution both transversely and longitudinally [24].

At the same time, GSI (Gesellschaft für Schwerionenforschung) in Germany developed pencil beam scanning (PBS) method for ion therapy. Based on GSI's experience, the Heidelberg Ion Therapy (HIT) Center project was approved and achieved. It is Europe's first hospital-based ion beam therapy center, with the first gantry on world for ion beam therapy[17]. In 2009, HIT began patient treatments with protons and carbon ions. More detailed information about HIT will be introduced in the following section.

By the 2000s and 2010s, further innovations, including compact superconducting cyclotrons, facilitated the development of single-room proton therapy systems, making the technology more accessible and cost-effective. Companies such as IBA and Varian introduced superconducting cyclotrons for proton therapy, with designs as compact as 2 to 3 meters in diameter [25, 26].

Recent research has focused on several advanced approaches, including FLASH proton therapy [27], which requires an ultra-high dose rate of 40 Gy per second; helium ion therapy [28], which provides higher linear energy transfer (LET) than proton therapy and exhibits less fragmentation compared to carbon ion therapy; and image-guided particle therapy [29], which enables real-time adaptive treatment. These researches aim to further enhance the treatment efficiency, improve the dose delivery precision, and reduce the side effects caused by unwanted doses.

3.2 Role of Particle Therapy in Cancer Treatment

Cancer is a major public health issue worldwide. According to the International Agency for Research on Cancer (IARC), there were 20 million new cancer cases in 2022, and this number is projected to reach 35 million by 2050 due to population growth and aging [30]. The leading cause of cancer death in 2022 was lung cancer, followed by cancers of colorectal, liver, female breast, and stomach [30]. Cancer treatment typically involves surgery, radiation therapy, chemotherapy, or combinations of these approaches [31]. Radiation therapy can be employed as a neoadjuvant treatment (before the primary therapy), an adjuvant treatment (after the primary therapy), or as the primary treatment itself, with over 50% of all cancer patients likely to receive radiation therapy during their treatment [32]. And the major form of radiation therapy is photon therapy, as it is widely available in hospitals.

Due to particle therapy's unique physical properties, as discussed earlier, it provides greater precision compared to photon therapy. This precision increases treatment effectiveness, improving survival rates and reducing toxicity. It is especially advantageous for pediatric patients, as it reduces the risk of secondary malignancies and other long-term side effects [6, 33]. Ion beam therapy has proven superior for rare cancers in the head and neck, as well as bone and soft tissues, and at the same time, evidence from Japan's 2022 public health insurance data shows its advantage over photon therapy for certain common cancers, including pancreatic and liver cancer [17].

However, because of its complexity and high costs, the scale of particle therapy remains significantly smaller than photon therapy. It requires an accelerator to generate ion beams, a beam delivery system, and a beam monitoring system. Additionally, its operation depends on a multidisciplinary team of engineers, physicists, oncologists, and other medical staff to maintain the facility, develop treatment plans, deliver treatments, and monitor patients. Currently, there are only 15 ion therapy centers and 121 proton therapy centers in operation worldwide, mainly in high-income countries [4]. On the contrast, photon therapy is widely available in hospitals. In the United States, statistics indicate that only 1.6% of all external beam radiation therapy treatments involve proton beam therapy [34].

In conclusion, particle therapy is a promising cancer treatment modality that reduces toxicity and improves patient outcomes. However, its high cost and complexity limit its widespread availability. Further studies are needed to assess the cost-effectiveness of particle therapy compared to photon therapy [35], to develop technologies that enhance its precision, efficiency and reduce costs, and to generate more clinical evidence supporting its use in cancer treatment [36].

3.3 Heidelberg Ion Therapy Center

In 1994, a pilot project was started at GSI to conduct ion beam therapy at its own accelerator complex. Within this project, several techniques were developed to improve ion beam therapy. In 1997, the first patients were treated successfully, demonstrating the reliability of the techniques. This expertise led to the launch of the Heidelberg Ion Therapy Center (HIT) project, which was approved finally in 2003 and began patient treatment in 2009. Many design elements from GSI were carried over to the HIT system, including the pencil beam raster scanning method. [17, 37, 38]

The objective of this thesis is to develop a new beam profile monitor (BPM) for HIT to replace the original multi-wire proportional chambers (MWPCs), which play a crucial role in the pencil beam raster scanning method. This section provides the necessary background on HIT and its pencil beams, which are the objects that the BPM is designed to monitor. Subsection 3.3.1 offers an overview of HIT's resources, focusing mainly on its extensive variety of the pencil beams; Subsections 3.3.2 and 3.3.3 describe the accelerator and dose delivery system at HIT; and Subsection 3.3.4 details the beam characteristics of the pencil beams.

3.3.1 Resources at HIT

The Heidelberg Ion Beam Therapy Center (HIT) is located on the campus of the University Hospital Heidelberg, in close proximity to the National Center for Tumor Diseases (NCT), Heidelberg University Hospital (UKHD), and German Cancer Research Center (DKFZ), serving as a hub for both particle therapy research and clinical practice.

Figure 3.1 provides an overview of the HIT accelerator system. It features four beamlines: two horizontal beamlines that terminate in separate horizontal treatment rooms; one horizontal beamline leading to the experimental cave for quality assurance (QA) and research activities; and one beamline dedicated to the heavy ion gantry.

In the gantry room, the beam can be directed in all directions while the patient remains stationary. The heavy ion gantry itself has a diameter of 13 meters, a length of 25 m and an overall weight of 600 tons. It is the world's first gantry for ion beam therapy. In comparison, a proton therapy gantry at the same time was just about 10 m in length and 200 tons in weight [17], due to the lower magnetic rigidity of protons compared to heavier ions required to reach the same penetration depth in tissue.

HIT employs the active scanning method for dose delivery [40]. As discussed in section 2.2, this method requires the accelerator to generate thin pencil beams with varying energies, along with a scanning system to steer and monitor the beams. All four beamlines at HIT are fully equipped with such scanning systems. Further

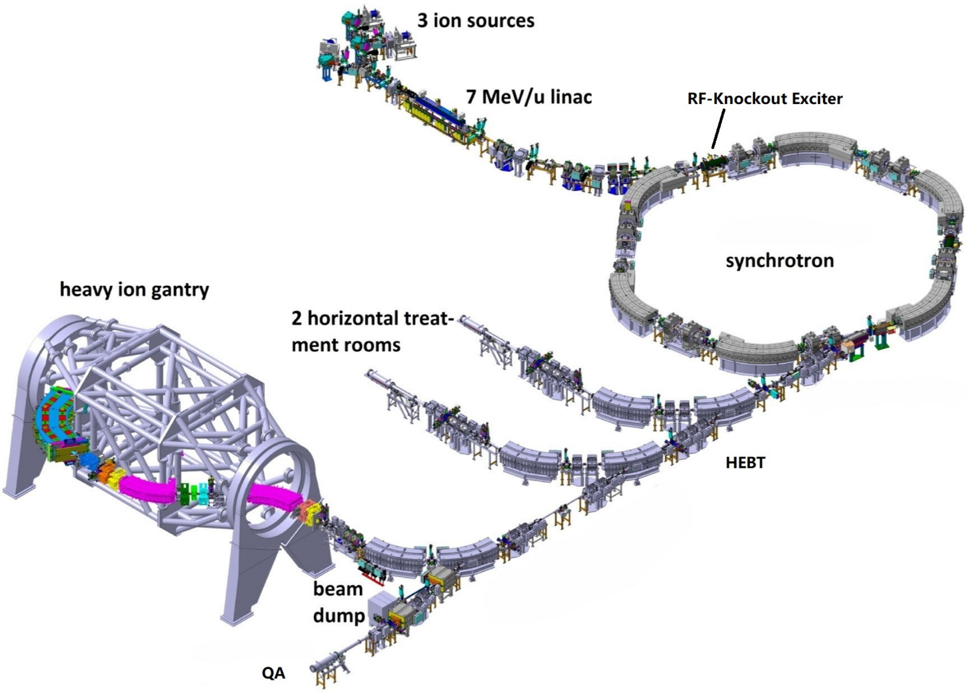


FIGURE 3.1: The overview of the accelerator complex at HIT [39].

details about the scanning system will be provided in subsection 3.3.3. The pencil beams at HIT are organized within the Library of Beams, which will now be explained.

Library of Beams

The HIT facility provides an extensive library of pencil beams (LIBC) containing 61 200 unique combinations of beam properties—defined by 255 energy steps, 10 intensity levels, 6 beam size steps, and 4 ion species—for each beamline. This results in a total of 244 800 configurations across the four beamlines. Because each pencil beam configuration requires its own unique set of accelerator control parameters, obtaining these parameters for the entire LIBC is extremely time-consuming. The commissioning of the HIT accelerator, which involved acquiring the control parameters for the subset of the LIBC corresponding to proton and carbon ion beams across three fixed beamlines, 4 beam size steps, 255 energy steps, and 10 intensity levels—took 22 months.

TABLE 3.1: Beam parameter ranges in the LIBC at HIT.

Parameter	Proton	Carbon	Helium	Oxygen
Energy (MeV/u)	48 – 221	88 – 430	50 – 221	104 – 515
Intensity (10^7particles / s)	8 – 320	0.2 – 8	2 – 80	0.1 – 4
Beam Size at isocenter (mm)	8.1 – 32.9	3.4 – 20.0	4.9 – 26.9	2.7 – 21.7
Maximum Intensity (10^8 particles/ spill) [37]	400	10	100	5

In the LIBC, 255 energy steps are designated as E1 through E255, intensities as I1 to I10, and beam sizes as F1 to F6. Although these labels are uniformly applied to the four ion species, the underlying beam parameters vary depending on the ion

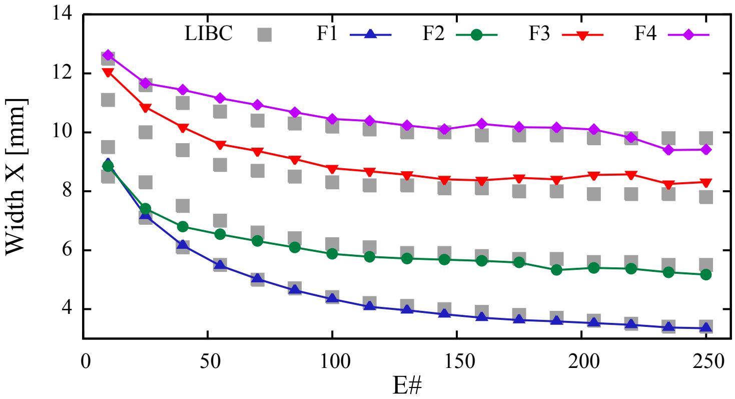


FIGURE 3.2: The horizontal focus for carbon ion beams at HIT in the isocenter for different energy and focus steps, measured with a viewing screen. The gray blocks show the reference values from LIBC [40].

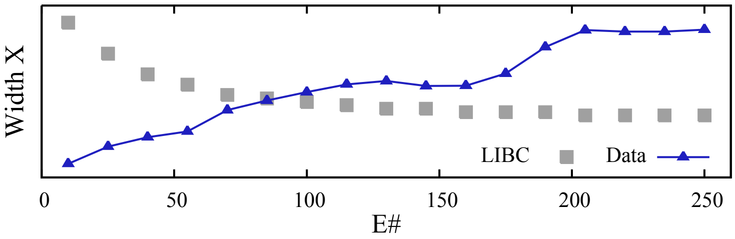


FIGURE 3.3: Horizontal beam width before commissioning [40].

species. Table 3.1 summarizes the range of beam parameters for the four ion species at HIT. The energy range is chosen to cover water penetration depths from 20 to 300 mm—suitable for treating the entire human body—with proton energies ranging from 48 to 221 MeV/u and carbon ion energies from 88 to 430 MeV/u. The beam size parameters depend not only on the ion species but also on the energy level; typically, higher-energy beams in the LIBC correspond to narrower beam sizes. Figure 3.2 illustrates the beam size parameters in the LIBC for different beam sizes F1 to F4 as a function of energy steps Ei, along with the corresponding measured beam sizes.

Higher-energy beams exhibit narrower beam sizes at the isocenter, which is a manual setting of the LIBC rather than solely a consequence of reduced multiple scattering; this design ensures that, although higher-energy beams penetrate deeper before reaching the Bragg peak and thus undergo more scattering, they are delivered with narrower widths to maintain a lateral distribution comparable to that of lower-energy beams. Figure 3.3 shows the measured beam size before commissioning. The increase of the size with energy is a consequence of optimizing the synchrotron for extraction efficiency [40]. During commissioning, the focusing strength of the quadrupoles was carefully adjusted for each energy step to achieve the desired beam width [40].

In addition to the pencil beam available in the LIBC, HIT can also provide high-intensity beams for FLASH research. For carbon, the system can deliver $\geq 5 \times 10^8$ ions within approximately 150 milliseconds [41]. However, extensive adaptations and optimizations of the accelerator parameters and dose delivery system are required to provide and monitor these high-intensity beams. In addition, HIT offers low-intensity beams for research applications, and a dedicated beam monitor for these beams is currently under development [42].

With a wide variety of beams available at HIT and its prime location in close proximity to hospitals and research centers, it supports a broad range of treatment options for patients while also serving as an invaluable hub for advanced research activities.

3.3.2 Accelerator System

Figure 3.1 shows an overview of the HIT accelerator system. It consists of four functional components: one injection chain, one synchrotron, one high-energy beam transport system (HEBT), and four beam terminals, including one gantry and three fixed beam lines, all of which are equipped with a raster scanning system.

The injection chain comprises ion sources, a complex linac, and an injection line into the synchrotron. Its main functions are to produce and monitor beams of desired particles, control the beam intensity by adjusting the beam optics (thereby regulating the beam loss rate), accelerate beams up to 7 MeV/u, and fully strip the ions. In the LIBC setting, the intensity step is controlled by the injection chain.

The synchrotron is equipped with dipole magnets providing a maximum magnetic field of 1.5 T, corresponding to a maximum magnetic rigidity of 6.6 T m. For carbon ions, this rigidity translates to an energy of 430 MeV/u, resulting in a penetration depth of 30 mm in water.

The primary functions of the synchrotron are to accelerate ions to the desired energy and to provide spill-to-spill energy variation. In the LIBC setting, the energy step is controlled directly by the synchrotron and is highly precise, with magnetic field deviations better than 1×10^{-4} [43]. The acceleration to the desired energy takes about 0.5 s.

On the synchrotron, an RF-knockout exciter is responsible for extracting the beam from the synchrotron to the HEBT. The extraction time controls the beam intensity; with a fixed number of particles per spill, a longer extraction time yields a lower instantaneous intensity. In general, the extraction is set to a slow mode—taking about 5 seconds—for tumor painting and beam monitoring, but it can also be tuned for ultrafast extraction of the entire spill within 100 to 200 milliseconds [41].

The HEBT consists of quadrupole lenses, bending magnets, and a fast beam abort chopper located immediately after the synchrotron. It transports the beams from the synchrotron to the selected beam terminals, tunes the beam size for the fixed beam lines, and switches off the beam in case of an emergency or interruption within 100 ms. The HEBT is also responsible for providing the required beams for the gantry system, while the beam size for the gantry is managed by the gantry itself.

In summary, the system operates as follows: the injection chain determines the ion species and, along with the RF-knockout exciter, regulates the beam intensity; the synchrotron controls the energy level; and the HEBT adjusts the beam size before delivering the beams.

At each beamline, the beam size is further refined by quadrupole magnets, while the beam position is controlled and monitored by the raster scanning system, which will be explained in the next subsection.

3.3.3 Pencil Beam Raster Scanning System

Figure 3.4 illustrates the concept of pencil beam scanning. The tumor target volume is divided into iso-energy slices (IES), with each slice irradiated by a pencil beam of a specific energy. The monoenergetic pencil beam scans each slice spot by spot, delivering a defined number of particles to each spot. Once the required number of

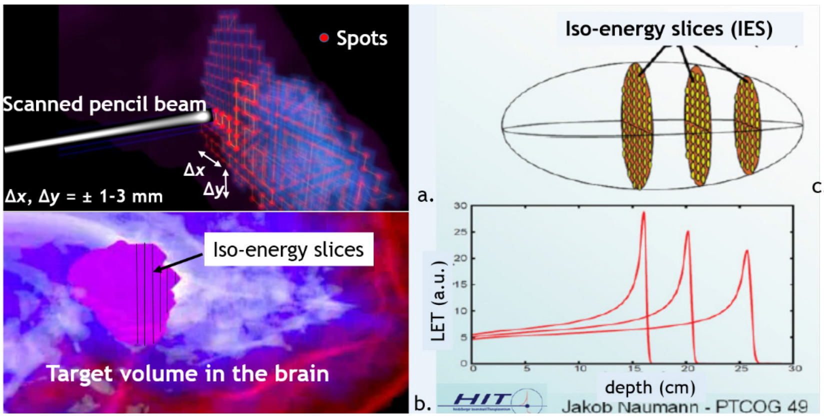


FIGURE 3.4: Example of the pencil beam scattering method [3]. (a) spot distribution, (b) iso-energy slices for a simulated target volume in the brain, (c) iso-energy slices treated with different Bragg peaks.

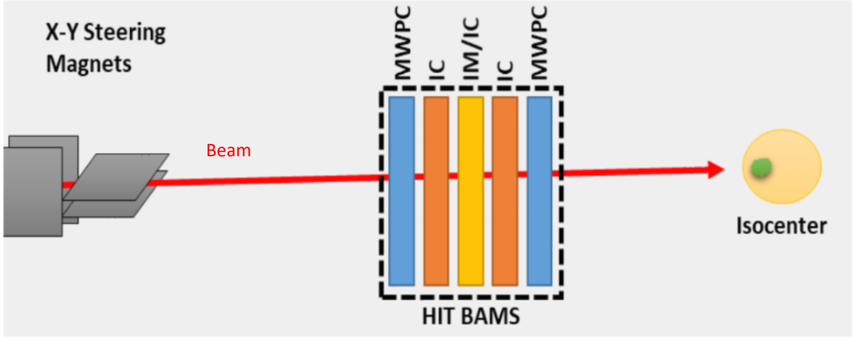


FIGURE 3.5: Sketch of the beam application monitors (BAMs) at HIT. The BAMs are placed at the nozzle exist, 1 m before the isocenter, measuring the beam intensity, position and width in real-time.

particles is delivered, the beam moves to the next spot until the entire IES is treated. After completing one IES, the beam is switched off, and the accelerator system is adjusted to the next IES. The accelerator then delivers a beam with a different energy for the new IES, and the raster scanning continues for the new slice. This process creates a 3D dose distribution that conforms precisely to the target volume. In raster scanning, the beam remains on between spots, making it faster than spot scanning, where the beam is switched off between spots.

The implementation of the pencil beam raster scanning at HIT consists of three main components:

- **X-Y steering magnets**: These magnets steer the beam in a zigzag pattern, delivering it to the desired positions by varying their strength. Each steering magnet's strength is controlled by its power supply, which is daily adjusted and calibrated based on the beam position measured by the BAMs. The beam steering speed is of the order of 10 mm per microsecond, which is limited by the ramp rate of the power supply [44].

- **Beam Application Monitors (BAMs)**: Figure 3.5 shows the components of the BAMs, which include three ionization chambers (ICs) and two multi-wire proportional chambers (MWPCs). The beam intensity is monitored by the ionization chambers (ICs), while the position and width are measured by the multi-wire proportional chambers (MWPCs) [45]. One MWPC measures the beam's position and focus in both the horizontal and vertical directions. With two ICs and two MWPCs, the beam's intensity, position, and focus are each measured twice. The IM/IC in the middle provides redundancy. The intensity and position are sent to the real-time control system.

Notably, the BAMs are located at the nozzle exit, approximately 1 m upstream of the isocenter. The beam size range of the LIBC, where the MWPCs are situated, is empirically between 2 mm and 8 mm. The beam size is carefully controlled to ensure that it does not fall below 2 mm at the monitor planes.

- **Real-time Control system**: This system compares the desired beam parameters from the treatment planning with the measured values from the BAMs and takes actions accordingly. It tracks the beam intensity to determine when to move to the next spot and uses position feedback from the MWPCs to adjust the steering magnets for the precise positioning of the next spot [46]. If the parameters from BAMs deviate beyond acceptable tolerances, the beam is interrupted. Notably, the control system is a purely hardware-based system designed for real-time control, free from computer intervention during irradiation [44]. This fact motivates the use of FPGAs in this thesis for the detector readout as well as the real-time reconstruction of beam position and width.

The precision of ICs and MWPCs determines the precision of the treatment. The ICs monitor the beam intensity. When enough particles have been delivered, the beam moves to the next spot. The dose delivered and its precision directly depend on the accuracy of the ICs. The MWPCs monitor and control the beam position via the real-time control system. The precision of the MWPCs directly determines the accuracy of the dose delivery's spatial distribution. If the MWPCs provide an incorrect position, the dose would be delivered to the wrong location, which must be avoided.

The response rate of the MWPCs limits the efficiency of the treatment [45]. The irradiation time for each raster point is longer than the time required to measure its position, ensuring that each point has a valid position measurement. Currently, the irradiation time per point exceeds 1 ms, and a typical treatment consists of approximately 6 million points. If the position measurement could be performed in less time, the overall irradiation time could be reduced, enabling more patients to be treated per day and improving patient comfort, as they must remain in a fixed position throughout the procedure.

3.3.4 Beam Characteristics: Microstructure

The pencil beams at HIT is in pulse/spill mode. In general, every spill lasts about 5 s. Figure 3.6 shows the time structure of one spill of a carbon beam. The beam intensity takes some time to rise from zero to the desired level, whereas the drop is sharp and fast. And its intensity fluctuates over short intervals but remains stable on a microsecond scale. Figure 3.7 shows a spill with three interruptions. The beam is interrupted for a short time, and then it is re-established. The intensity quickly returns to the previous level from zero. The transitions in beam intensity around interruptions are sharp and rapid.

The beam characteristics is important. As we will see later, increasing the integration (shutter) time from 100 µs to 250 µs results in a stronger signal. This is related to the continuous characteristic of the beam during one spill. Additionally, at the beginning of one spill, the BPM occasionally fails to detect the beam, due to the low and unstable beam intensity at the start of the spill.

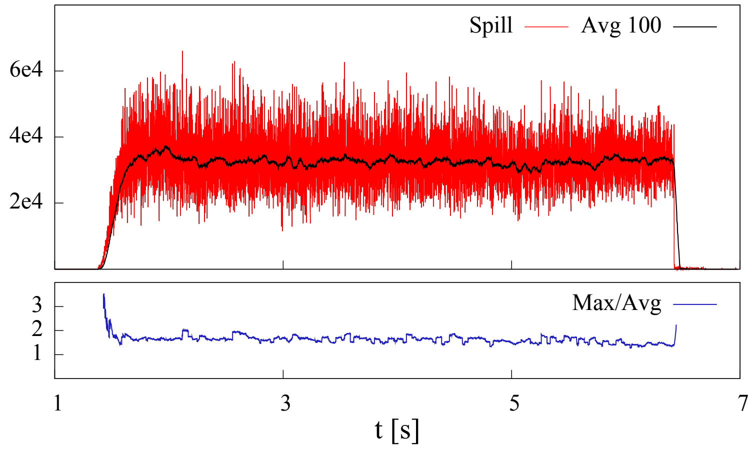


FIGURE 3.6: The time structure of a 250 MeV/u carbon spill, measured with an ionization chamber in the HEBT [40]. The red line shows the particle number in 500 µs time intervals, and the black line shows the particle number in 50 ms time intervals. The blue line shows the maximum over average of the red curve in a sliding window of 50 ms.

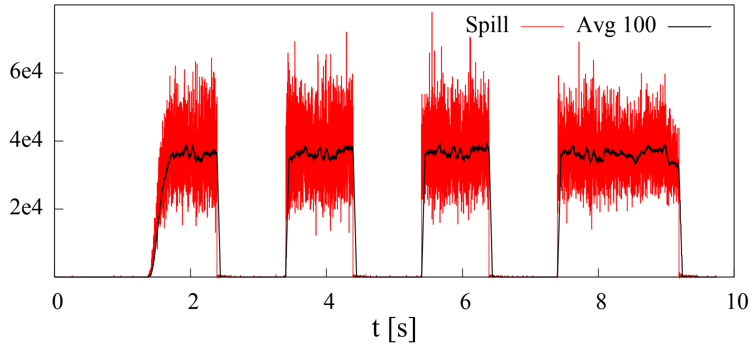


FIGURE 3.7: A spill with three interruptions of 1 s [40].

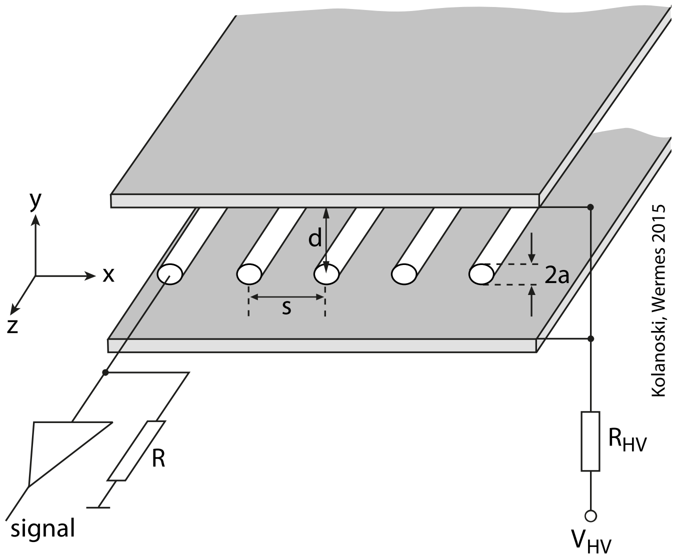


FIGURE 3.8: Perspective view of an MWPC. The two parallel planes are cathodes, and the white wires are anodes. Signals are drawn from the anodes. Typical values for the dimensional parameters are: $a = 0.01\,\text{mm}$, $s = 2\,\text{mm}$, and $d = 8\,\text{mm}$. [47]

3.4 Beam Profile Monitor

The previous section introduced the beam characteristics and the pencil beam raster scanning system at HIT. The multi-wire proportional chambers (MWPCs) in the BAMs determine the position of the pencil beam and therefore play a critical role in treatment precision. This section focuses on the MWPCs and various types of beam profile monitors (BPMs) that could serve as replacements.

Subsection 3.4.1 describes the MWPCs and their limitations, which have driven the need for new BPMs. Subsection 3.4.2 outlines the requirements for the next-generation BPMs, while subsection 3.4.3 introduces the scintillating fiber-based BPM developed in this work. Finally, subsection 3.4.4 briefly discusses other detector technologies that can be used as BPMs.

3.4.1 Multi-Wire Proportional Chamber and its Limitations

Principle

Multi-Wire Proportional Chambers (MWPCs) are a type of gas-based particle detector that convert the ionization energy loss of incident particles into electrical signals on nearby wires, thereby providing positional information. Fig. 3.8 depicts the structure of a typical MWPC, in which evenly spaced anode wires are situated between two cathode plates. A high voltage is applied to the cathodes, creating an electric field that intensifies near the anode wires.

Electron–ion pairs are produced by the ionization process. The primary electrons drift toward the anode wires. The electric field around the anodes is so strong that an electron can gain enough energy between collisions to ionize a molecule in the next collision, thereby producing additional ion–electron pairs. Consequently, the signal is multiplied, and the number of secondary pairs is proportional to the number of primary ionization electrons; hence, this region is called the “proportional region.”

The gain in this region depends on factors such as the electric field strength, the gas composition, and the gas pressure. The typical gas gain at HIT is 1.

The electrons drift rapidly and are typically collected within about 100 ns. Meanwhile, the positively charged ions—created mainly during the multiplication process—travel toward the cathode plates, either recombining with electrons along the way or upon reaching the cathodes. Since ions move much more slowly than electrons, they generally require up to milliseconds reach the cathodes [47].

Both the electron and ion motion generate current signals. A capacitor can be connected to each anode wire to collect charge within a defined integration window, or a resistor can be connected to each anode to provide a dynamic voltage signal proportional to the current.

Limitations

The MWPCs used at HIT as part of the BAMs are industrial products developed by Siemens and are essentially "black boxes" to researchers. It has been communicated that obtaining an identical MWPC is no longer possible. One such black box provides beam position and width measurements in both the horizontal and vertical directions. This can be achieved by arranging the cathode wires perpendicular to the anode wires, with signals read from both; alternatively, a single black box may house two orthogonal MWPCs.

It has been noted that obtaining a valid position measurement requires about 1 ms, which is considered the primary factor limiting treatment speed at HIT [45] (Schoemer et al., 2015). Unfortunately, the origin of this "1 ms" delay is not explained in the paper. One plausible explanation is that space-charge effects caused by the slow drift of ions contribute to the delay. For instance, while the X–Y steering magnets can move the beam spot by 1 mm in less than a microsecond, the ions produced around the original beam spot continue to drift toward the cathode, generating a current that overlaps with the new signal and causes distortion. Additionally, HIT may require sufficient statistical averaging for a valid measurement, further extending the measurement time.

At high intensity, the gain of the MWPCs is reduced due to space-charge effects, which weaken the local electric field and degrade the gain. This effect can become dominant, leading to saturation of the beam profile. For the flash-related research, the gas mixture is changed from an Ar–CO_2 mixture to helium, which has a lower multiplication power [41]. Although in this way MWPCs can cover a large dynamic range, changing the gas mixture is tedious and time-consuming.

The resolution of MWPCs is also limited. A typical MWPC has an anode wire spacing of 2 mm, which imposes a fundamental limit on the position resolution. Assuming a square signal distribution, the position resolution is given by

$$\frac{2}{\sqrt{12}}\ \text{mm} = 0.58\ \text{mm}.$$

Manufacturing wires with spacing smaller than 2 mm is challenging due to the repulsive forces between the wires and constraints related to wire tension [48]. Although the position resolution can be improved to better than 1 mm by applying amplitude information from multiple wires using the center-of-gravity method, the inherent 2 mm wire spacing ultimately limits the achievable resolution of MWPCs.

This 2 mm wire spacing is also the reason why the beam size at the nozzle exit is constrained to be larger than 2 mm.

TABLE 3.2: The requirements for new beam profile monitors

Requirement	Value
Active Area	$> 20\text{cm} * 20\text{cm}$
Beam Size Range	$2\text{mm} - 33\text{mm}$
Material in Beam	< 0.35mm Water Equiv./ plane(4 planes in total)
Position Resolution	$< 0.2\text{mm@}4 - 8\text{kHz}$
Focus Resolution	$< 0.4\text{mm@}4 - 8\text{kHz}$

Moreover, due to the MGy-level dose deposition at the (0,0) position over the course of one year at HIT, detector aging is a concern. The plasma discharges in the MWPCs can lead to the accumulation of polymer deposits on the electrodes, reducing the electric field and consequently weakening the signal strength [49]. To address this aging effect, a new BPM must be integrated into the HIT system in the near future.

3.4.2 Requirements for new BPMs

Table 3.2 outlines the requirements for the new BPM. To cover the $20\text{cm} \times 20\text{cm}$ beam scanning area at the isocenter, the BPM must have an active area larger than that dimension.

The beam size range reflects a combination of the beam size at the nozzle exit (from 2 mm to 8 mm) and at the isocenter (2.7 mm to 32.9 mm) of the LIBC, since the new BPM will be installed at the current location of the MWPCs and may also be used for beam quality assurance at the isocenter. The beam size at the nozzle exit is constrained to be larger than 2 mm due to the 2 mm wire spacing of the MWPCs.

The material budget in the beam should be minimized to reduce multiple scattering. The requirement is below 0.35 mm of water equivalent per plane. There can be four planes in total.

The position resolution should be better than 0.2 mm at 4–8 kHz, and the focus resolution, i.e. resolution of beam width, should be better than 0.4 mm at 4–8 kHz. These values are preliminary specifications, as the ideal resolution for particle therapy remains under investigation.

Beyond the requirements in Table 3.2, for the HIT, the BPM must be integrated into the existing raster scanning system. This means that the position information should be calculated and delivered entirely by hardware in real time for the BAMs' control system, and it must also be incorporated into the facility's quality assurance routines.

Additionally, the BPM should be capable of operating under flash conditions at extremely high intensities, and it also has the potential to operate in magnetic resonance imaging (MRI) environments for developing MRI-guided particle therapy.

Last but not least, the BPM should be cost-effective (reducing costs for patients), simple to manufacture, and easy to maintain.

3.4.3 Scintillating Fiber-based BPM

In this thesis, a real-time scintillating fiber-based beam profile monitor (BPM) for ion beams has been developed and tested at HIT to meet the requirements outlined above.

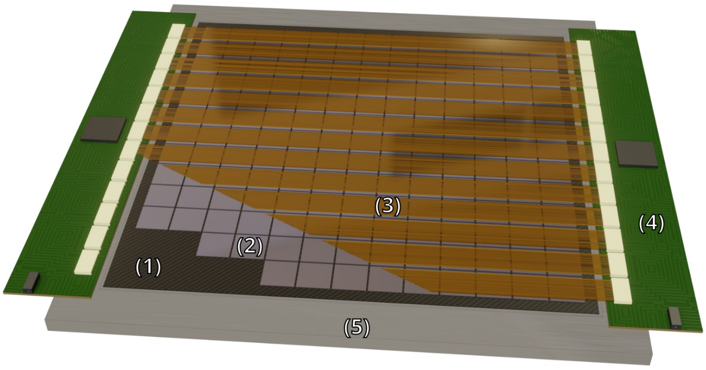


FIGURE 3.9: Schematic 3D rendering of the HV-CMOS BPM with 13 × 13 chips (2), interconnected by flex cables (3), assembled on a thin carbon fiber plate (1). The surrounding electronic boards (4) are mounted on a stabilizing frame (5) [50].

Each BPM plane consists of four components: a 25 cm × 25 cm low-mass scintillating fiber plane (with a water-equivalent thickness below 0.35 mm), five silicon photodiode arrays, a custom ADC board, and a FPGA development board. Together, these components measure the one-dimensional projection of the pencil beam.

MRI compatibility was evaluated in a preliminary test with a version in which only the fiber plane was placed inside the MRI scanner, yielding promising results. However, MRI compatibility is beyond the scope of this thesis.

The beam position and width are reconstructed in real time using the FPGA board, as the requirements for the new BPMs necessitate a hardware-based solution.

The scintillating fiber-based BPMs represent a mature technology that has already demonstrated successfully in the LHCb experiment at CERN. Moreover, they are straightforward to produce and maintain, as they do not require a gas system or high-voltage supply.

For this detector, my colleague Dr.Blake Leverington designed the system, ensuring that the requirements for active area and material budget were met while minimizing the overall size of the BPM. Building upon the work of Dr.Michał Dziewiecki[1], who developed the electronics readout, my contribution focuses on studying the signals from the scintillating fiber-based BPM, developing real-time beam profile reconstruction on the FPGA based on Michał's readout system, and the commissioning and operation of this detector. These aspects are discussed in detail throughout this thesis.

3.4.4 Other Alternatives

High-voltage CMOS (HV-CMOS) pixel sensors are monolithic detectors that integrate both the sensor and the readout electronics onto a single chip, which itself can be thinner than 50 µm. The sensor part consists of a reverse-biased p-n junction, to which a high voltage is applied to generate a large depletion region for particle detection. The integrated readout electronics comprise specialized CMOS circuits capable of detecting and processing signals from the sensor region. Due to the rapid

[1] Dr. M. Dziewiecki was with Heidelberg University. He is now with the Hardware and Electronics group, GSI Helmholtzzentrum für Schwerionenforschung GmbH, Darmstadt, Germany (e-mail:m.dziewiecki@gsi.de)

drift of charge carriers and the minimal distance between the sensor and electronics, these sensors can achieve nanosecond-level time resolution.

The operating principles of the p-n junction as a sensor are detailed in section 4.3, where the photodiode component of the scintillating fiber-based BPM is explained. Further details on CMOS technology are provided in Appendix A, as part of the explanation of FPGA technology.

One candidate for the new BPM is the HV-CMOS sensor named HitPix, developed at the Karlsruhe Institute of Technology (KIT)[50]. Figure 3.9 shows a schematic 3D rendering of the HV-CMOS BPM, which consists of an array of 13 × 13 HitPix chips, each with dimensions of 20 mm × 20 mm, to achieve an active area exceeding the specified requirement. Each HitPix chip will contain 96 × 96 200 µm × 200 µm pixels. Currently, the first version, HitPix1, has 24 × 24 pixels per chip, and the HitPix2 has 48 × 48 pixels.

The sensor can operate in 2D mode or 1D mode. In the faster 1D mode, only the summed counts for each row and column, so the projections, are read out, which takes approximately 30 µs for HitPix2 [50], making it suitable for real-time beam monitoring. In the slower 2D mode, count values from all pixels are read out individually. This mode is not suitable for real-time monitoring but can be employed for diagnostic purposes to assess beam shape when necessary.

The HitPix2 achieves a position resolution of 0.15 mm for carbon ion beams at a low intensity of 5×10^6 ions/s using an integration window of 30 µs. However, at higher beam intensities, the hit-detection efficiency decreases due to accumulated charges causing a drop in baseline voltage. Furthermore, radiation hardness studies indicate the necessity of incorporating a cooling system to mitigate increased leakage currents resulting from radiation damage, which adds complexity to the material budget. Additionally, these sensors cannot withstand rapid changes caused by scanning MRI devices due to induced voltages. Currently, the HitPix3 is under development with the aim of improving radiation hardness and enhancing hit-detection efficiency at high beam intensities.

Another example is the multi-strip ionization chamber, which, unlike MWPCs, utilizes thin aluminum or copper film strips coated on Mylar foils as anodes. Such detectors are currently in use at the CNAO (National Center for Oncological Hadrontherapy, Italy) and the PSI (Paul Scherrer Institute, Switzerland) to monitor beam position and size in clinical applications [3]. At PSI, the achieved position resolution is approximately 0.1 mm, albeit with a relatively long integration time of around 20 ms. Although it can measure beam profiles at a frequency of 5 kHz, the dead time exceeds 50 µs [51].

Another detector type is the strip-segmented silicon Low-Gain Avalanche Detector (LGAD). This device functions similarly to standard silicon diode detectors, relying on a p-n junction structure, but it incorporates a highly doped layer that generates a substantial electric field to multiply charges, similar to a proportional chamber. An LGAD developed for CNAO contains 146 strips, each with a pitch size of 180 µm and an inter-strip dead region of 80.8 µm. The detector itself is relatively compact, measuring 114 µm thick and 26.2 mm wide [52]. LGADs feature excellent temporal resolution, achieving nanosecond-level performance. The signal generated by a single particle on a strip lasts roughly 2 ns and can be clearly distinguished from noise [53].

A further approach for monitoring beam position within the patient is to measure secondary ions produced through fragmentation interactions in carbon ions therapy. In this method, two mini-trackers placed downstream of the patient track secondary ions, enabling reconstruction of the pencil beam position. At HIT, the precision and uncertainty of the position achieved range between 0.6 and 1.5 mm, meeting clinically acceptable requirements [54].

In conclusion, the gas-based detectors benefit from mature technology, ease of construction, and capability to achieve a large active area. However, they are hindered by long dead times due to slow ion drift and the complexity of auxiliary systems required for operation. The silicon-based detectors offer superior temporal and spatial resolution with rapid response times, yet they suffer from small active areas, high costs, and considerable R&D efforts needed for clinical adoption.

This thesis focuses specifically on the scintillating fiber-based BPMs, a mature technology that offers a large active area, good spatial resolution, fast response, and cost-effective implementation.

Part II

Detector

Chapter 4

Scintillating Fiber-based Ion Beam Profile Monitor

The scintillating fiber-based Beam Position Monitor (SciFi-BPM) measures the one-dimensional ion beam projections using scintillating fiber planes coupled to photodiode arrays. The photodiode signals are digitized by ADCs on a custom PCB that also integrates voltage regulators. A MAX 10 FPGA controls the photodiode arrays and ADCs, processes the digitized data to reconstruct the beam position and width, and transmits both the results and data via Ethernet to a PC with dedicated acquisition software.

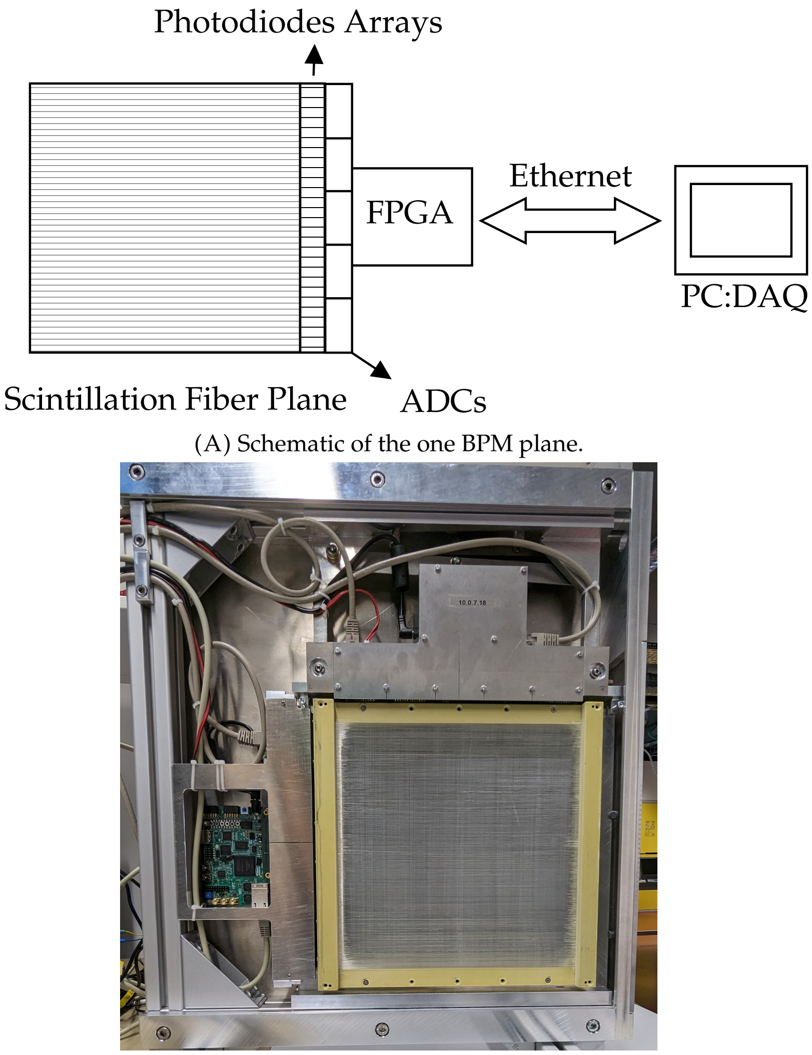
This chapter begins with an overview of the SciFi-BPM in Section 4.1. It then systematically presents the individual components in detail in the order of the signal production chain: starting with the scintillating fiber plane, followed by the photodiode array and the sensor readout system. Finally, the chapter concludes with a version history of the detector's development in Section 4.5.

4.1 Setup

Fig. 4.1A presents the a simplified schematic of one SciF-BPM plane, which consists of a scintillating fiber plane, five 64-channel silicon photodiode arrays, five ADCs, and one FPGA. From left to right, the granularity becomes coarser: the fiber plane is composed of fibers with a diameter of 250 µm, the photodiode arrays have a pitch of 0.8 mm, and in total 320 photodiode channels are digitized by five ADCs, with one ADC assigned to each array. These five ADCs, together with the corresponding photodiode arrays, are controlled by a single FPGA. Finally, several SciF-BPM planes can be connected to one PC for data acquisition.

The scintillating fibers used are Kuraray SCSF-3HF (1500 ppm) fibers with a diameter of 250 µm [55]. This diameter was chosen to ensure compatibility with the winding techniques developed for LHCb, to minimize the material budget in the beam, and to provide sufficient light yield for reliable beam profile reconstruction. Further details about the fibers and the production of the fiber planes are provided in Section 4.2.

The photodiode arrays are Hamamatsu S11865-64 devices [56]. Each array contains 64 photodiodes, each equipped with an amplifier and an integration circuit. They were selected for several reasons: their commercial availability with a standardized geometry also offered by other manufacturers; a readout frequency compatible with HIT specifications; linearity and dynamic range suitable for ion therapy; the presence of an on-board integration circuit; and the expectation that non-ionizing radiation damage will only increase the noise baseline or leakage current without significantly affecting the signal. Additional information is provided in Section 4.3.



(A) Schematic of the one BPM plane.

(B) One XY-station.

FIGURE 4.1: Schematic of one SciF-BPM plane and picture of one SciF-BPM XY-station.

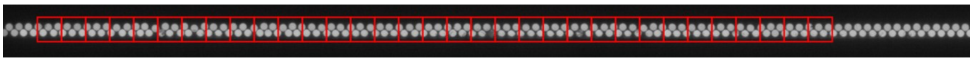


FIGURE 4.2: Photograph of the cross-section of a two-layer glueless fiber plane with a schematic red rectangle indicating the photodiode array.

The ADCs are 16-bit AD7983 devices [57]. A custom PCB hosts five of these ADCs, one for each photodiode array, together with voltage regulators supplying power and voltage references to the ADCs and photodiode arrays, a synchronization port for connecting to other SciFi-BPM planes, and signal lines bridging the signals from the FPGA to the ADCs and photodiode arrays. An Altera DK-DEV-10M50-C Max10 development board [58] was selected to read out the five ADCs in parallel and to provide control and clock signals to both the ADCs and photodiode arrays. These readout components are described in more detail in Section 4.4.

Figure 4.1B is a photograph of the mechanical assembly of one SciF-BPM XY-station, consisting of two orthogonal planes for measuring the 1D projection in both vertical and horizontal direction. Details about the mechanism of the XY-station see Section 4.5. Dr. Blake Leverington has built 3 XY-stations, and In Chapter 7, they were placed in beamline for performance testing.

The SciFi-BPM will be installed in the beamline upstream of the patient to provide real-time measurements of the ion beam position and width. The fiber planes, placed directly in the beamline, produce scintillation photons when traversed by the ion beam through fluorescence.. A fraction of these photons is guided to the fiber ends by total internal reflection.

At one end of the fibers, one-dimensional photodiode arrays convert the scintillation photons into photoelectrons, which are then integrated over a defined integration time. The resulting signal amplitudes are digitized by ADCs. The signals collected within one integration period from all photodiodes are referred to as a *frame*. Each frame is collected and further processed by the FPGA before being transferred to a PC. The typical integration time is 94.4 µs at a readout rate of 10 kHz (10 000 frames per second), with a dead time of 5.6 µs.

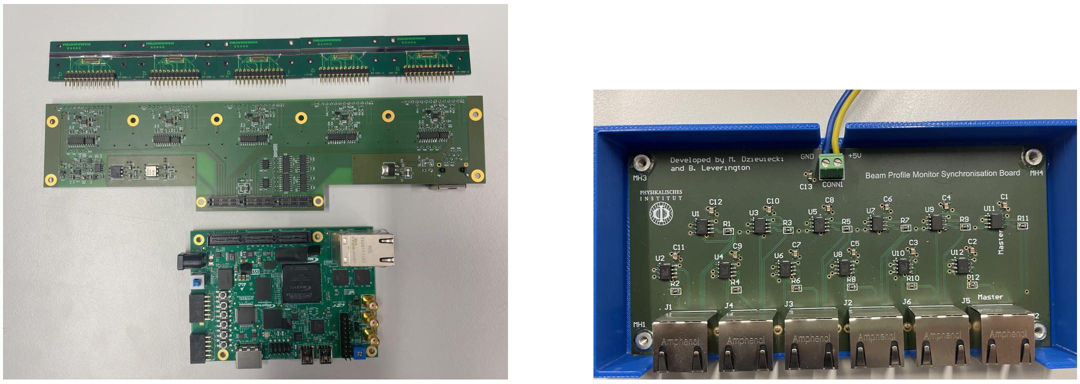
The relative size and alignment of the fibers and photodiodes are illustrated in Fig. 4.2. The fiber diameter is 250 µm, while the photodiode pitch is 0.8 mm. The number of fibers coupled to each channel varies, the channel response also differs and requires calibration. On average, each photodiode channel covers 4.3 fibers. If a single fiber is missing, the signal decreases by approximately $1/4.3 \approx 23\%$.

The fiber plane has a height of 0.3 mm, while the photodiode height is 0.8 mm. The photodiode arrays are not glued to the fiber ends; instead, an air gap of 0.5 mm is left to allow for the removal and replacement of aged fiber planes. As expected, the optical alignment between the 0.3 mm-high fiber plane and the 0.8 mm-high photodiode array is critical: the signal amplitude is highly sensitive to the quality of their coupling, as poor coupling can lead to a reduced signal or even complete signal loss.

These aspects are discussed in detail in Chapter 7.

Fig. 4.3 shows the electronic components of one BPM plane, together with the synchronization board used to synchronize up to six planes.

The following sections are structured to introduce the scintillating fiber plane, the silicon photodiode array, and the electronic components sequentially, following the order of the signal production chain. In this way, both the principle and functionality of the detector become clear step by step.


Developed by M. Dziewiecki and B. Leverington
Beam Profile Monitor Synchronisation Board
Master

(A)

(B)

FIGURE 4.3: Electronics of the SciFi-BPM. (A) One BPM plane, from top to bottom: five photodiode arrays, the sensor readout board (hosting five ADCs), and the FPGA board. (B) Synchronization board.

4.2 Scintillating Fiber Plane

The scintillating fiber plane is the only BPM component placed directly in the beam and is made of plastic scintillating fibers. As beam-intercepting material, its thickness must be below 0.35 mm per mat, as specified in Tab. 3.2. In addition, the light yield of the fiber planes and the quality of fiber alignment directly determine the signal quality of the detector.

Subsection 4.2.1 introduces the principle of the scintillating fibers, while Subsection 4.2.2 describes the size, alignment, and production of the fiber planes. In addition, this section provides an estimate of the photon yield per photodiode pitch.

4.2.1 Scintillating Fiber

The SciFi-BPM uses double-clad, green-emitting plastic scintillating fibers, SCSF-3HF (1500 ppm) from Kuraray [55], with a circular cross-section and a diameter of 250 µm. They were selected for their higher radiation resistance compared to the blue-emitting SCSF-78 fibers used in the LHCb SciFi tracker, and for their emission spectrum, peaking at 530 nm, which is well matched to the photodiode sensitivity peaking at 700 nm [1]. The green fibers have a scintillation decay time of 7 ns, longer than the 2.8 ns of the blue fibers, but the microsecond signal integration regime employed here permits the use of scintillators with relatively long decay times.

When a charged particle traverses a scintillating fiber, it loses energy through ionization. This energy is absorbed by the scintillator, exciting molecular states, whose subsequent de-excitation leads to the emission of light. Part of the emitted light can be reabsorbed within the scintillator. Since the emitted photons have lower or nearly equal energy compared to the absorbed photons, the emission spectrum is shifted relative to the absorption spectrum. This difference is known as the Stokes shift. A larger Stokes shift reduces reabsorption of emitted photons in the material, thereby improving the light yield. [47]

The SCSF-3HF fiber is a ternary scintillator based on polystyrene, which acts as the solvent and primary scintillator, doped with 1% p-terphenyl as the primary dye and 0.10% 3-hydroxyflavone (3HF) as a wavelength shifter. The dye is added to the polystyrene base to improve the efficiency of the scintillation process. Energy is transferred from the base to the dye within sub-nanosecond timescales via Förster

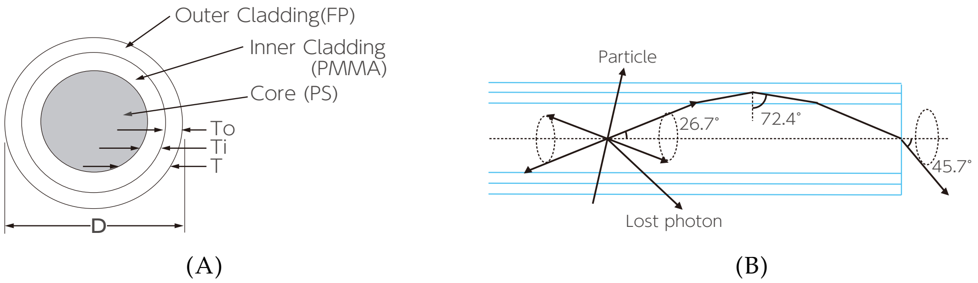


FIGURE 4.4: The SCSF-3HF fiber [55]. (A) Cross-sectional view. (B) Lateral view.

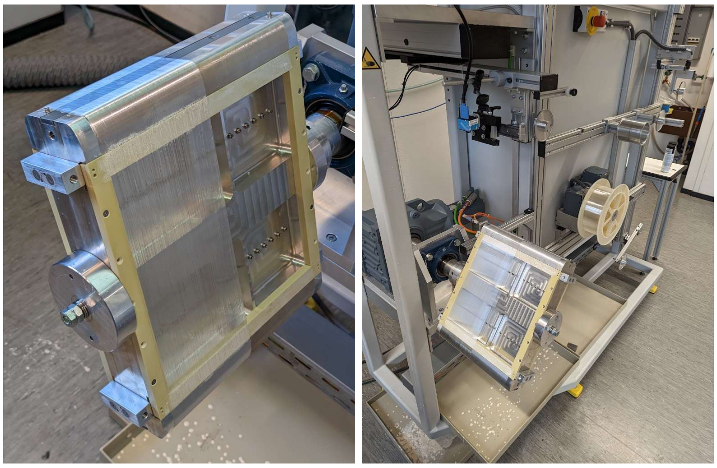


FIGURE 4.5: The winding hub (left). The winding hub and fiber spool on the LHCb winding machine (right).

transfer, after which the excited dye molecules relax by emitting photons. The wavelength shifter then absorbs these photons and re-emits them at longer wavelengths in the green spectrum. [59]

As a result, the emission spectrum spans from 480 nm to 650 nm, with a peak at 530 nm, and does not overlap with the absorption spectrum.

The schematic of the fiber is shown in Fig. 4.4. The core, inner cladding, and outer cladding are made of polystyrene (PS), polymethylmethacrylate (PMMA), and fluorinated polymer (FP), with refractive indices of 1.59, 1.49, and 1.42, respectively. Photons emitted within a cone of half-angle 26.7° are confined by total internal reflection in the double cladding, resulting in a trapping efficiency of 5.4%. This efficiency refers to photons propagating toward a single fiber end.

The scintillating fibers have an intrinsic light yield of about 8000 photons per MeV of deposited ionization energy. For a proton with an energy of 220 MeV, the specific energy loss in polystyrene is approximately 0.4 MeV mm^{-1}. If the proton traverses 250 µm of fiber, the resulting light yield is about 800 photons. With a trapping efficiency of 5.4%, this is reduced to 43.2 photons.

The SCSF-3HF fibers have a light attenuation length of 4.5 m [55]. According to the Beer–Lambert law, after passing through a fiber length of 25 cm, the remaining light intensity is 94.5% of the initial value, reducing the yield from 43.2 photons to about 40.8 photons per 220 MeV proton per 0.25 mm-thick fiber.

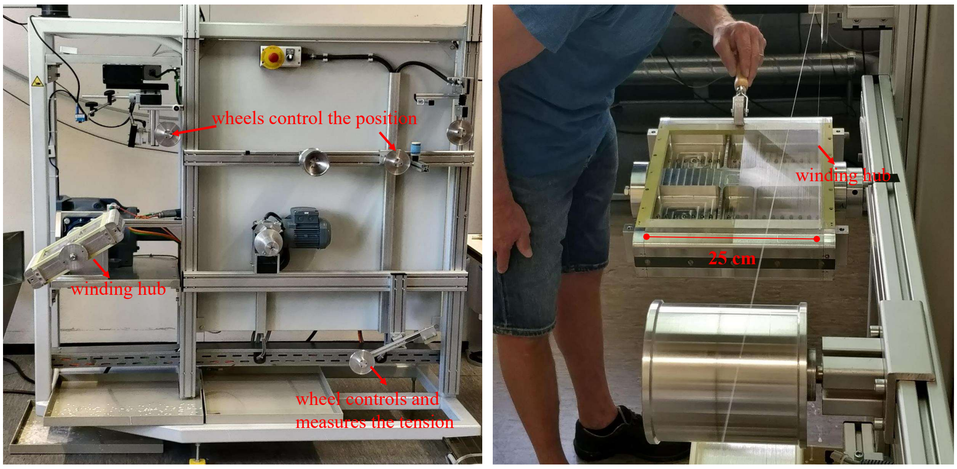

wheels control the position
winding hub
wheel controls and measures the tension
winding hub
25 cm

FIGURE 4.6: The winding process.

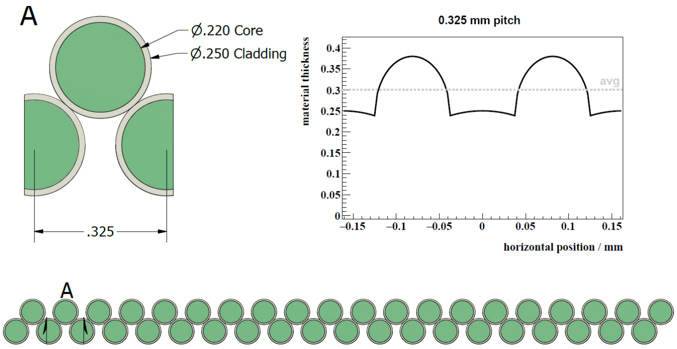


FIGURE 4.7: Schematic cross-section of a two-layer glueless fiber plane and its thickness distribution [1].

4.2.2 Fiber Planes

The fibers, delivered on spools from Kuraray, are mounted on the LHCb winding machine using adapted technology from the LHCb scintillating fiber tracker [60]. Figure 4.5 shows the fiber spool, the winding machine, and the winding hub with two installed fiber frames, on which the fibers are wound.

The scintillating fibers are wound into fiber planes on threaded winding wheels, as shown in Fig. 4.6. They are guided through several wheels, one of which is attached to a tension control arm. The final wheel before the winding frame is mounted on a linear stepping motor, which shifts the horizontal position by the winding pitch (0.325 mm) after each rotation.

Before winding begins, glue is applied to the fiber frames on the winding hub to fix the fiber ends under tension. First, one half of the fiber plane is wound in two layers; then the hub is rotated by 180°, and the second half is wound.

After winding is completed, an additional layer of the fiber frame material is placed on top of the glued region where the fibers are wound, so that the fibers are clamped between two frame layers. The hub is then kept in place until the glue dries, after which the fiber mats are cut out. Finally, the ends of the fiber planes are polished to enhance optical efficiency.

The winding process ensures that the fibers are placed in a regular pattern with a pitch of 0.325 mm, forming a matrix of two staggered layers. As illustrated in Fig. 4.7, the material thickness ranges from 0.23 mm to 0.38 mm, with an average of 0.3 mm. This corresponds to approximately 0.3 mm water-equivalent thickness,

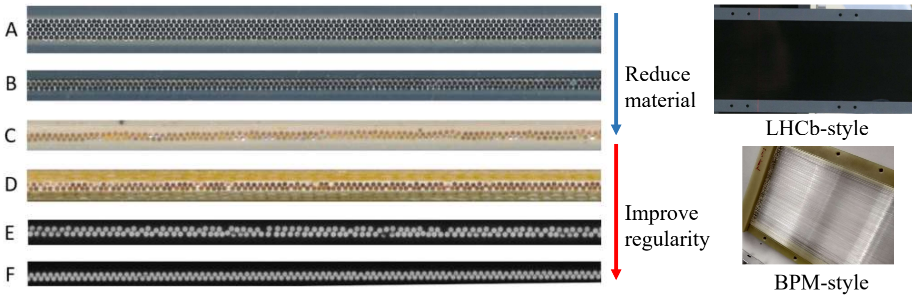


FIGURE 4.8: Iterations of fiber planes: Type-A 5-layer LHCb-style, Type-B 2-layer LHCb-style, Type-C 2019 BPM-style, Type-D 2021 BPM-style, Type-E 2022 BPM-style, and Type-F 2023 BPM-style [61].

since the density of the fiber material is similar to that of water. It therefore satisfies the material budget requirement of 0.35 mm water-equivalent thickness, as specified in Tab. 3.2.

In this way, a 25 cm × 25 cm two-layer glueless fiber plane is produced, which both meets the material budget requirement and provides sufficient coverage for the 20 cm × 20 cm active area.

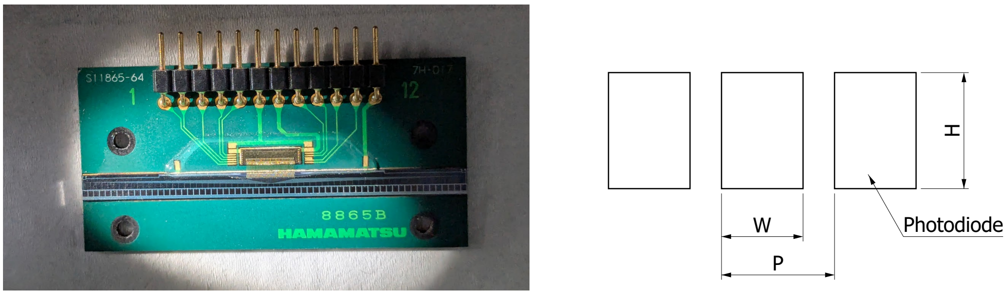
The original LHCb fiber planes were six-layer structures with a winding pitch of 0.275 mm. Titanium dioxide epoxy glue was applied between the fibers to hold them together, and a thin black polyimide foil was glued to both sides to provide mechanical stability [60]. This design, referred to as *LHCb-style*, results in a total thickness of about 1.35 mm, far exceeding the 0.35 mm material budget specified in Tab. 3.2. To reduce the material in the beam, several alternatives have been developed since 2017: five-layer LHCb-style mats [1], two-layer LHCb-style mats, and two-layer glueless mats, the latter being referred to as *BPM-style*.

Fig. 4.8 shows cross-sectional photographs of fiber plane iterations over the years. From Type-C to Type-F, the regularity of the BPM-style fiber matrix improved significantly. Key developments included refining glue application, mitigating static charge, and minimizing external pressure, all of which enhanced fiber uniformity [61]. The most advanced winding process was applied to the Type-F fiber planes, achieving consistent material uniformity. These Type-F planes were used in the beam tests from 2023, and their performance is presented in Chapter 7.

At one end of the fiber plane, outside the active area, photodiode arrays collect the scintillation photons. As shown previously in Fig. 4.2, each photodiode accommodates on average 4.3 fibers. Variations in the number of fibers per photodiode, together with the regularity of the fiber matrix, contribute to channel-to-channel differences in light yield. This will be shown in detail in Section 7.2.

At the opposite end, the fibers can be coupled to a mirror. The mirror reflects photons back toward the photodiodes, thereby increasing the detected light intensity. However, the extended optical path introduces a delay in the photon arrival time, which can distort the time structure of the signal. For applications where timing resolution is critical, this effect must be considered. In the case of the BPM, for a fiber length of 25 cm, the additional delay is about 1.3 ns, which is negligible compared to the BPM integration time of 94.4 µs. Therefore, mirrors were applied to five out of the six planes tested in Chapter 7, with one plane left without a mirror to evaluate the signal enhancement effect by the mirror.

As estimated in the previous subsection, the light yield per 0.25 mm-thick fiber per 220 MeV proton is 40.8 photons. For fiber planes with a thickness of 0.3 mm, this



(A)

(B)

FIGURE 4.9: Hamamatsu S11865-64 photodiode array: (A) front side, (B) enlarged schematic of the photosensitive area [56]. W = 0.7 mm, P = 0.8 mm, H = 0.8 mm.

corresponds to about 49 photons.

At the lowest intensity and under standard beam width settings at HIT, the beam delivers a flux of approximately 8×10^7 protons per second with a width of 5 mm. This corresponds to a flux density of 1.6×10^7 protons per second per millimeter.

The active area of each the photodiode is 0.7 mm wide. In one 0.7 mm-wide channel with a 0.3 mm-thick fiber plane, this results in approximately 5.5×10^8 photons per second per channel, peaking at 530 nm. This corresponds to about 0.18 lx incident on the photodiode area of 0.7 mm in width by 0.8 mm in height, assuming a coupling efficiency of 100% between the fiber planes and the photodiode arrays.

4.3 Silicon Photodiode Array

The photons from the scintillating fibers are read out by silicon photodiode arrays. The device used in this study is the commercial S11865-64 photodiode array chip from Hamamatsu [56]. Each array contains 64 photodiode channels, each with its own charge amplifier and integrator. All channels are integrated simultaneously; the resulting analog voltage amplitudes are then output sequentially. The chip is sensitive to photons in the range of 300–1000 nm, with a peak sensitivity at 700 nm, well matched to the emission spectrum of the green-emitting fibers.

The properties of the photodiode array directly influence the detector performance: its physical dimensions determine the spatial resolution; its output voltage amplitude, once digitized by a 16-bit ADC, defines the usable signal, with the maximum amplitude setting the detector's upper limit and the noise level reflecting the detector noise; and the signal timing characteristics, including the waveform shape in relation to the system clock, constrain the integration and dead times, which are accommodated by the FPGA firmware.

This section introduces these aspects of the photodiode array chip and explains how it is employed in the SciFi-BPM. The photodiode arrays were not characterized independently in this thesis. The dimensional specifications, saturation voltage, maximum non-uniformity and typical photosensitivity are taken from the manufacturer's datasheet [56].

Dimensional Information

Each photodiode array contains 64 channels arranged in a line, as shown in Fig. 4.9A. Figure 4.9B illustrates the schematic of the active photodiode area. The channel pitch is 0.8 mm, of which 0.7 mm is photosensitive.

The 64 channels span a total length of 5.12 cm. To cover a 25 mm-wide fiber plane, five such arrays are aligned with a 0.2 mm gap between adjacent arrays. This gap is accounted for in the position and width reconstruction by treating it as a 0.2 mm extension on one side of the neighboring channel. Discussion about the bias introduced by this gap is included in Section 7.5.

Output Voltage Characteristics

The output signal of this chip is the voltage amplitude for each channel, with respect to its reference voltage. This reference voltage is provided by the sensor readout board as 4.5 V relative to ground. The maximum output voltage is 3.5 V with respect to this reference voltage [56], which defines the saturation voltage.

The voltage amplitude is digitized by a 16-bit ADC (range 0–65535) on the sensor readout board. The ADC reference voltage is set to 4.55 V, corresponding to the maximum count of 65535. Thus, one ADC count corresponds to approximately 6.9×10^{-2} mV.

The 0.05 V difference between the ADC reference voltage and the sensor reference voltage introduces a pedestal of about 720 ADC counts. Section 5.1 discusses the pedestal and the subtraction method in detail.

The sensor output thus saturates at 3.55 V relative to the ADC reference, which is about 51 500 ADC counts. This value is close to the maximum ADC count and defines the detector's upper detection limit.

In addition, instabilities in the reference voltages can induce noise to all the channels at the same time. This is referred to as common-mode noise, which is investigated in Section 5.2.

The typical photosensitivity of the chip is $2200\,\mathrm{V\,lx^{-1}\,s^{-1}}$ [56]. With an incident light level of 0.18 lx and an integration time of 100 µs, this corresponds to an output of about 40 mV for the proton beam at the lowest intensity setting at HIT, or roughly 580 ADC counts above the pedestal. In practice, the observed value is lower since the coupling efficiency between the fiber planes and the photodiode arrays is less than 100%. At the highest intensity, as discussed in Section 7.3.6, the maximum observed value is around 12 400 ADC counts, corresponding to approximately 0.8 V above the pedestal, which is well below the 3.5 V saturation limit.

The maximum photoresponse non-uniformity between channels specified in the datasheet is 10% (defined as the difference between maximum and minimum response over the average). This non-uniformity contributes to the channel-to-channel variation in output voltage and is corrected through calibration, together with the variations caused by the fiber alignment in Section 7.2.

Signal Timing Structure

Fig. 4.10 shows the signal timing chart of the sensor. The `CLK` signal is the sensor clock. The sensor can operate at clock frequencies between 40 kHz and 4 MHz. In this work, the clock was set to 3.57 MHz, derived from the 50 MHz FPGA clock. The `RESET` signal controls the integration time (shutter) of the sensor; its frequency also defines the measurement rate, i.e., the readout rate per frame. The video signal corresponds to the analog voltage amplitudes of the channels, which are output sequentially channel by channel. Both the `RESET` and `CLK` signals are generated by the FPGA, while synchronization of the `RESET` signal across multiple BPM planes is ensured by the synchronization board, as will be seen in Section 4.4. The video signal is connected to the ADC, which digitizes the output at the correct timing under FPGA control.

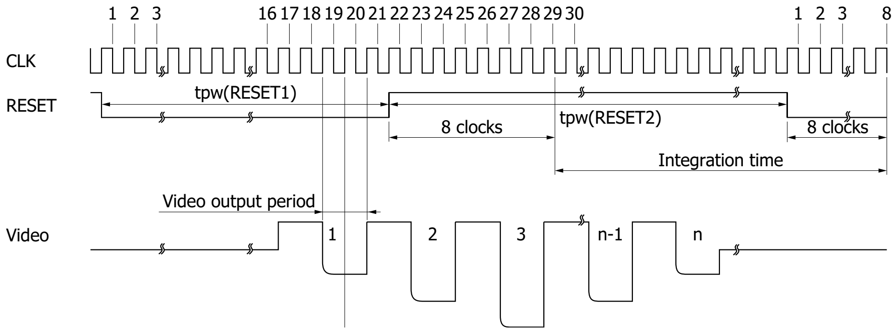


FIGURE 4.10: Signal timing chart of the photodiode array [56].

Integration begins 8 clock cycles after the RESET signal goes high and ends 8 clock cycles after it goes low. The video output starts 18 clock cycles after the falling edge of RESET, with the first channel lasting 2 clock cycles. Subsequently, each channel is output for 2 clock cycles, followed by 2 idle clock cycles, until all 64 channels have been read.

The period of the RESET signal must be longer than the total video streaming time, given by

$$18 + 2 + 4 \times 63 = 272 \text{ CLK cycles,}$$

which corresponds to 76.16 µs at a 3.57 MHz clock. This defines the maximum readout rate of 13 kHz. In addition, the low duration of the RESET signal must be at least 20 clock cycles, and the rising edge must occur outside of the video output period. These constraints introduce a minimum dead time of about 5.6 µs at a 3.57 MHz sensor clock.

The longest integration time achievable at 3.57 MHz is determined by a 12-bit down-counter for the RESET high signal on the FPGA, which results in a minimum readout rate of 0.87 kHz.

In conclusion, the detector can operate at readout rates between 0.87 kHz and 13 kHz with a dead time of about 5.6 µs. Longer integration times can be achieved by reducing the sensor clock frequency.

Readout Rate (kHz)	Integration Time (µs)	Dead Time (µs)
10	94.4	5.6
8	119.3	5.7
6.7	144.2	5.8
5	194.3	5.7
4	244.2	5.8

TABLE 4.1: Readout rate, integration time, and dead time.

Tab. 4.1 summarizes the five readout rates and corresponding integration times used in this thesis for the noise measurement in Section 5.2 and the linearity measurement in Section 7.3. The dead time is not always exactly 5.6 µs, as a few additional clock cycles are sometimes inserted to synchronize the sensor clock between BPMs and ensure that the rising edge of RESET does not occur during the video output period. For all other measurements, the detector was operated at a readout rate of 10 kHz with an integration time of 94.4 µs.

4.4 Sensor Readout

The previous two sections described the fiber planes and the photodiode arrays, which together form the sensor part of the detector. After the sensor, the signal is processed by the readout system.

The readout consists of four main modules, as illustrated in Fig. 4.3: the sensor readout board, the synchronization board, the FPGA board, and, not shown in the photographs, the data acquisition software (HITDAQ). The FPGA board is the core of the readout system, while HITDAQ is a convenient user interface software of the SciFi-BPMs.

This section briefly introduces these four modules and their functions. It explains how the five photodiode arrays are connected to form a single BPM plane, how synchronization between different boards is achieved (which is later used to verify that the firmware processes all frames completely and on time), and how the sensor operation can be flexibly controlled via HITDAQ. More detailed information can be found in the manual [62].

4.4.1 Sensor Readout Electronics Board

Fig. 4.11 shows the top and bottom sides of the sensor readout board, its main components, and its connections to the FPGA board and the five photodiode arrays. The board interfaces with the FPGA board via an HSMC connector, with the five photodiode arrays via five 12-pin connectors, and with the synchronization board via a dedicated connector. It comprises five identical ADC blocks, each assigned to one photodiode array. In this way, the board serves as the central interface between the photodiode arrays, the ADCs, the synchronization board, and the FPGA.

With five 64-channel sensors, the total number of channels is 320. All five photodiode arrays are controlled by the FPGA through the readout board: their integration windows open simultaneously, and the five ADCs digitize the corresponding signals in parallel. The resulting 320 channels of 16-bit digitized data are stored in a large buffer on the FPGA for further processing, which is described in detail in Chapter 6. One frame of raw data from a BPM plane thus contains 320 16-bit values.

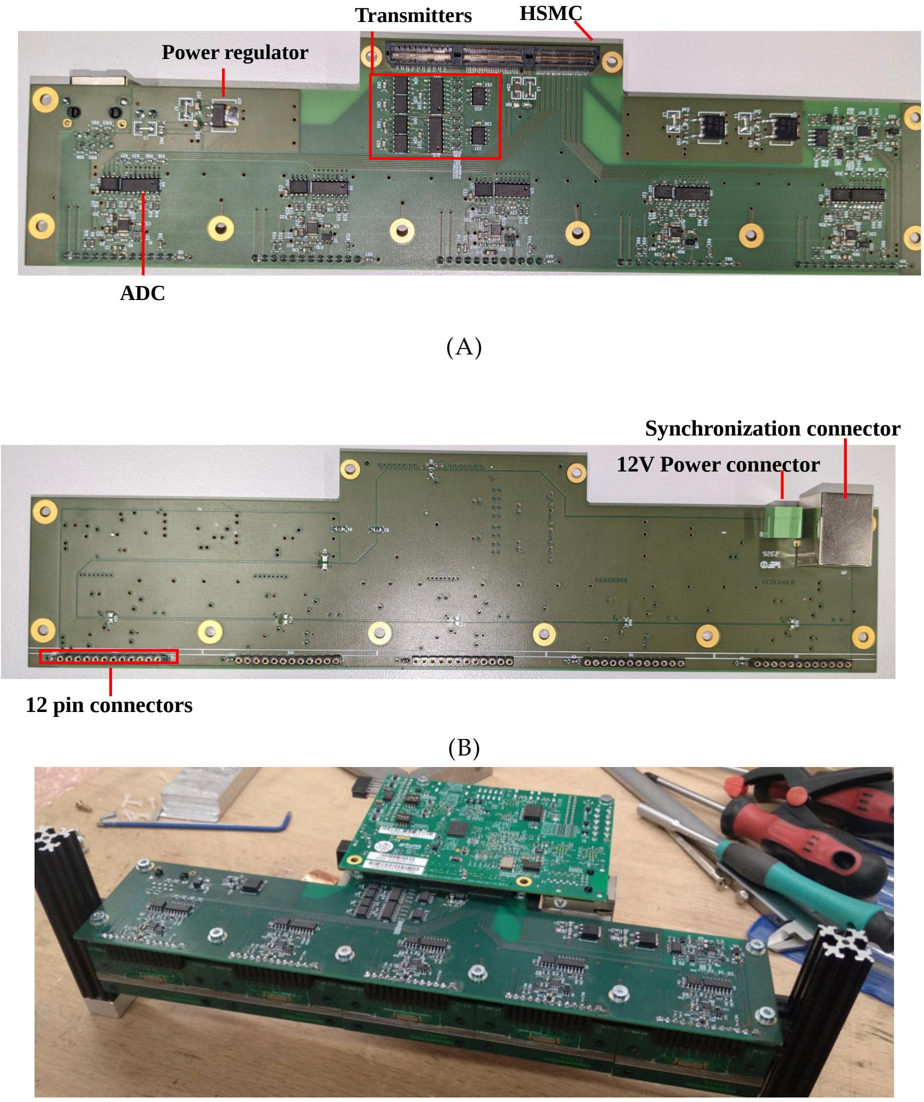
The board supplies power and provides reference voltages to both the photodiode arrays and the ADCs. Each array receives a dedicated reference voltage for its output, and each ADC is supplied with its own reference voltage as well. Consequently, the module consisting of five arrays operates with five separate reference voltages. As shown later in Section 5.1, the array-dependent pedestal behavior can be attributed to small variations between these reference voltages.

Fluctuations in these voltages directly translate into fluctuations of the signal amplitude, thereby introducing noise. Since the same reference voltage is distributed to all five arrays and subsequently across all channels, such fluctuations affect them simultaneously, resulting in correlated noise between channels. As will be discussed in Chapter 5, two categories of boards were identified: one with low common-mode noise and one with high common-mode noise.

4.4.2 Synchronization Board

The synchronization between BPM planes is achieved at two levels via the synchronization board: fine synchronization and frame-level synchronization.

When several SciFi-BPMs are set up together, all synchronization connectors on their sensor readout boards are connected to the synchronization board shown in



(A)

(B)

(C)

FIGURE 4.11: Sensor readout board: (A) top view, (B) bottom view, (C) assembled with photodiode arrays and FPGA board.

Fig. 4.3B. One BPM is designated as the leader board, while the others act as follower boards. Two signals are generated on the leader FPGA and transmitted to the follower FPGAs via the synchronization board: a frame clock and an 8-bit global frame ID, corresponding to fine synchronization and frame-level synchronization, respectively. The function of the synchronization board is to broadcast these two signals from the leader to all follower boards.

The frame clock, generated by the leader FPGA or received by the follower FPGAs, triggers the photodiode RESET (shutter) signal through delay logic implemented on the respective FPGAs. The delay for the leader board is set slightly longer than for the follower boards, ensuring that the shutter times of all BPM planes remain synchronized. The delay can be finely adjusted with a granularity of 20 ns, corresponding to the 50 MHz FPGA clock, and is tuned using an oscilloscope to align the RESET signals across all planes. This procedure is referred to as *fine synchronization*, as it ensures simultaneous shutter timing across BPM planes. The photodiode CLK signal is also synchronized by similar mechanism; further details are provided in [62].

After each aligned shutter time, each board generates a frame package and sends it to HITDAQ. For frame-level synchronization, each package contains not only the board ID, used by HTIDAQ to distinguish the data source, but also a local frame ID and a global frame ID. The local frame ID is a 16-bit counter of the number of frame clock cycles in each FPGA. The global frame ID is derived from the lower 8 bits of the local frame ID of the leader FPGA. It is transmitted from the leader board to the follower boards and included in the package alongside the data of the subsequent frame.

HITDAQ collects all frame packages from every plane and assembles the data into a combined frame according to the frame ID information. This procedure is referred to as *frame-level synchronization*.

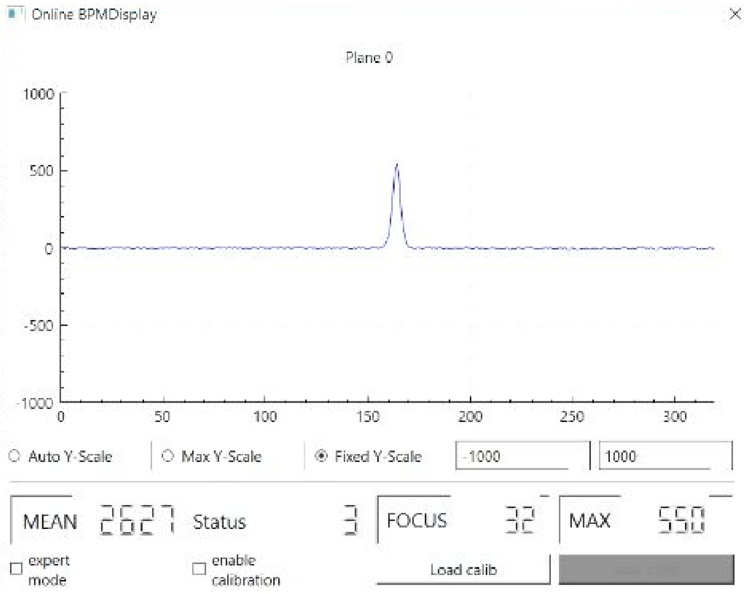
The frame ID information is stored by HITDAQ together with the signal data for offline analysis. The global frame ID in each frame should always be offset by −1 relative to the lower 8 bits of the local frame ID, and both values should increment by one from frame to frame. This is later used to monitor potential frame losses and to verify in Section 6.8.5 that the firmware processes all frames correctly and on time.

4.4.3 FPGA and Firmware

The FPGA board, shown in Fig. 4.3, is the MAX10 FPGA Development Board from Altera [58]. Its HSMC connector interfaces with the sensor readout board, while its Ethernet link is used to stream data and communicate with HITDAQ.

The FPGA firmware functions as the brain of the detector. It provides the CLK and RESET signals for the five photodiode arrays on a plane and controls the five ADCs, which digitize the outputs of the arrays in parallel. The firmware collects this data and streams it via Ethernet. It also implements the synchronization logic, allowing a plane to be configured as a leader (transmitting synchronization signals) or as a follower (receiving synchronization signals), as described previously.

In addition, the FPGA is configured as an embedded TCP/IP server, enabling control of the detector via HITDAQ. Users can start or stop measurements, check detector status, and configure parameters such as the readout rate, integration time, sensor clock frequency, synchronization delays, and leader/follower mode. Notably, these settings can be modified directly via HITDAQ without the need to reprogram the FPGA, providing a highly flexible and convenient interface.



○ Auto Y-Scale ○ Max Y-Scale ● Fixed Y-Scale -1000 1000

MEAN 2627 Status 3 FOCUS 32 MAX 550

□ expert mode □ enable calibration

FIGURE 4.12: HITDAQ interface shortcut displaying one frame data.

On top of these functions, Chapter 6 integrates reconstruction algorithm into the firmware, enabling real-time determination of the beam position and width. Some parameters related to the reconstruction algorithm can also be configured via HITDAQ ensure the flexibility of the reconstruction process.

4.4.4 DAQ Software

As described earlier, the DAQ software, named HITDAQ, allows connection to and control of multiple BPM planes, performs frame-level synchronization, assembles the packages with the same frame ID into combined frames, and stores the data to disk. In addition, it can display the data in real time at 10 Hz. Fig. 4.12 shows the interface for displaying the data from one BPM plane.

For synchronization with the BAM system (MWPCs, ICs) at HIT, HITDAQ can generate a timestamp that is transmitted to the BAM system. This timestamp is stored by both the BAM system and the BPM, enabling offline time alignment between detectors. This functionality was used to study the correlation between the SciFi-BPM and the MWPCs, showing an excellent agreement of 0.9987 for a high-intensity scan. This study is not presented in this thesis.

HITDAQ was originally developed by Dr. Michał Dziewiecki and later refined by Dr. Blake Leverington, Julian Horn, and the author to support different firmware versions.

4.5 Version History

The SciFi-BPM has been under development since 2016. Fig. 4.13 and Tab. 4.2 summarize four versions and their main characteristics.

The first test beam was performed in 2016 [1]. The earliest prototype, shown on the left in Fig. 4.13, used fiber planes with a thickness of 1.2 mm and was coupled with the Hamamatsu photodiode array tested at a readout rate of 1 kHz at HIT, with an integration time of 100 µs. These measurements demonstrated that the detector's linearity and dynamic range were suitable for operation at HIT.

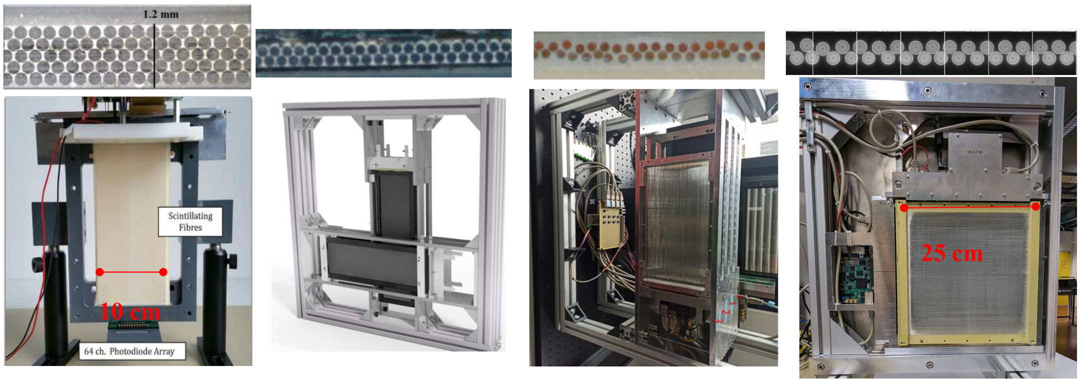


FIGURE 4.13: Version history of the SciFi-BPM system. From left to right: 2016, 2021, 2022, and 2023.

Year	2016	2021	2022	2023–present
Progress	Proof of principle	Resolution study	Correlation with MWPCs	FPGA implementation
Style of mats	LHCb	LHCb	BPM	BPM
Layers of fiber	5	2	2	2
Thickness	1.2 mm	0.46 mm	0.3 mm	0.3 mm
Width	10 cm	10 cm	10 cm	25 cm
Photodiodes per plane	64	128	128	320
Readout rate	1–3 kHz	0.87–13 kHz	0.87–13 kHz	0.87–13 kHz

TABLE 4.2: Version history of the SciFi-BPM system.

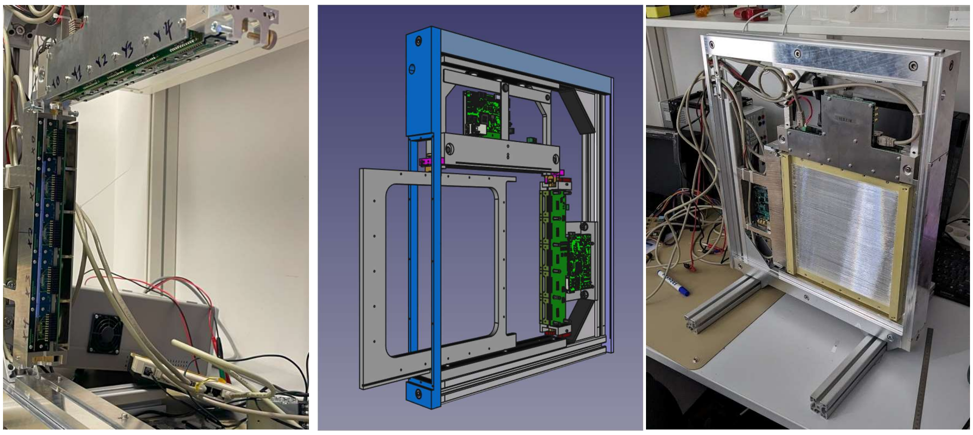


FIGURE 4.14: Left: mechanical support structure. Middle: aluminum plate for installing two fiber planes. Right: assembled BPM XY-station.

Between 2016 and 2020, customized readout boards, FPGA firmware, and HITDAQ were developed, enabling readout rates from 0.87 to 14 kHz with a dead time of about 5.6 µs. At this stage, the FPGA streamed out only the raw data.

From 2021 onward, new versions of fiber planes were developed to reduce the material budget, improve regularity (see Section 4.2.2), and increase the width to cover the 20 cm × 20 cm active area. In parallel, this work focused on developing a reconstruction algorithm for beam position and width, which was later implemented on the FPGA. The most recent FPGA firmware is capable of streaming raw data, processed data, and reconstruction results, while the latest version of HITDAQ supports real-time display of all three and simultaneously stores them to disk.

The most recent version, shown on the right in Fig. 4.13, is the SciFi XY-station. Fig. 4.14 illustrates the mechanical structure and the assembly of fiber planes. Two scintillating fiber planes are mounted on the aluminum frame and inserted into the support structure, allowing easy replacement. The alignment between the fiber planes and the photodiode arrays is adjusted manually using pins that push or pull the aluminum plate.

In conclusion, this detector utilizes well-aligned scintillating fibers to measure the one-dimensional projection of ion beams, with photodiode arrays converting the scintillating photon intensity into electrical signals for processing. As expected, the two most important factors determining detector performance are the signal-to-noise ratio (SNR) and the alignment of the system (mainly the fiber-to-fiber alignment). A higher SNR directly translates into better resolution, while precise alignment determines the accuracy of the measurement. These aspects will be examined throughout this thesis in the context of developing the reconstruction algorithm and evaluating the performance of the detector.

Chapter 5

Signal Characteristics and Beam Profile Reconstruction Algorithm

Chapter 4 introduced the components and operating principle of the scintillating fiber-based BPM stations. This chapter focuses on the signal characteristics and especially the reconstruction algorithm chain derived from them. For real-time beam position and focus measurement, the reconstruction algorithm is implemented on an FPGA, with further details provided in Chapter 6. Or for high precision offline analysis, it is implemented as a C++ software package on CPU highlighted in the calibration section in Chapter 7.

The whole reconstruction chain for a single frame of a BPM plane consists of pedestal subtraction, common mode subtraction, clustering, calibration and finally the beam position and width calculation by RMS, Linear Regression (LR) or fitting. Section 5.1 shows the pedestal over channels and over time, and the pedestal subtraction methods. Section 5.2 presents the noise characteristics, focusing on the common mode noise and its subtraction. Depending on the level of common mode noise, the BPM planes can be categorized into two types: the board 0 type with low common mode noise and the board 1 type with high common mode noise. Section 5.4 describes the clustering algorithm and its efficiency. Section 5.5 presents the calibration methods briefly, with more details in Section 7.2. Finally, Section 5.6 compares the beam position and width calculation methods for gaussian-like beams, including RMS, LR and fitting. And the whole reconstruction chain is summarized in Section 5.7.

5.1 Pedestal and Pedestal Subtraction

Pedestal refers to the ADC amplitude when no signal is present. As described in Section 4.3, a non-zero ADC amplitude arises from the difference between the ADC reference voltage and the photodiode array reference voltage.

The first step in the reconstruction chain is therefore to subtract the pedestal, which is measured across channels in Subsection 5.1.1, over time in Subsection 5.1.2, and finally subtracted using the method described in Subsection 5.1.3.

5.1.1 Pedestal Measurement

For convenience, the noise data for channel i is noted as the average pedestal value μ_i with a noise fluctuation δ_i:

$$\mathrm{S_i} = \mu_i + \delta_i \tag{5.1}$$

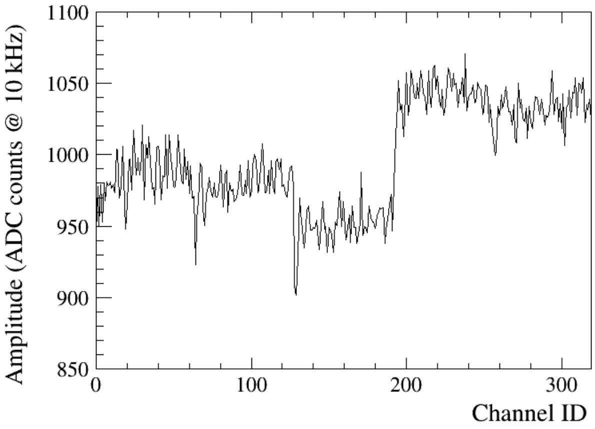


FIGURE 5.1: One frame of raw ADC amplitudes during noise run from one BPM plane without beam

One frame of the raw amplitudes for one BPM plane contains 320 channels from 5 arrays, each with 64 channels:

$$\{S_0, S_1, S_2, \ldots, S_{319}\}$$

The channel ID 0 to 63 are from the first array, channel ID 64 to 127 are from the second array, and so on.

Fig. 5.1 shows a raw frame ADC amplitudes of a SciFi-BPM plane during a noise run. The value does not center around zero. The ADC measures the voltage difference between its own reference voltage 4.55 V and the Video signal from the photodiode array. The Video signal sits on a 4.5 V reference voltage. The voltage difference results in a −0.05 V offset. This offset is intentionally introduced to lift the noise baseline, enabling the signal to utilize the central portion of the ADC's dynamic range and thereby improving the effective resolution.

Due to the slight variations of the reference voltages of each array and channel, the pedestal value varies across channels and arrays. This pedestal variation is clearly visible in Fig. 5.1, where the five distinct pedestal levels correspond to the five photodiode arrays.

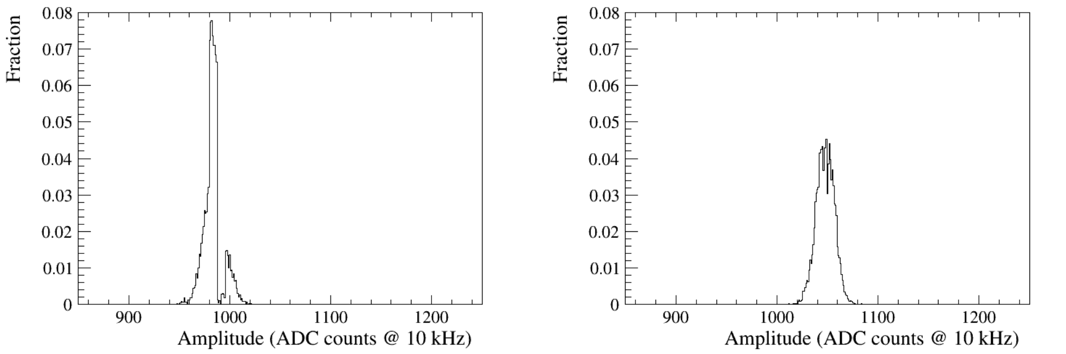
To determine the average pedestal, the mean $\bar{\mu}_i$ over N frames is evaluated as

$$\bar{\mu}_i = \frac{1}{N} \sum \mu_i \,. \tag{5.2}$$

The associated uncertainty $\sigma_{\bar{\mu}_i}$ is given by

$$\sigma_{\bar{\mu}_i} = \frac{\sigma_{ni}}{\sqrt{N}} \,, \tag{5.3}$$

where σ_{ni} denotes the standard deviation of the noise distribution (i.e., the pedestal distribution) of channel i. The uncertainty of the pedestal measurement $\bar{\mu}_i$ decreases as $1/\sqrt{N}$. And this is independent of the shape of the noise distribution, according to the central limit theorem.



(A) Channel ID 3. $\mu_{202} = 983.28, \sigma_{202} = 9.36$

(B) Channel ID 202. $\mu_{202} = 1048.29, \sigma_{202} = 8.95$

FIGURE 5.2: Normalized histograms of the raw amplitudes for two channels on board 0.

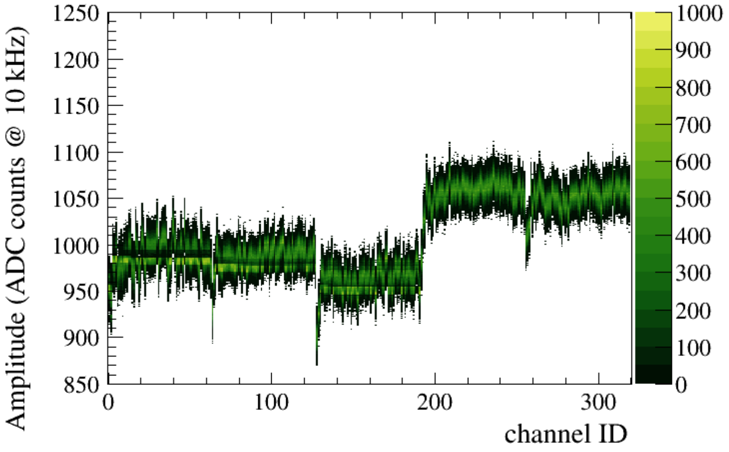


FIGURE 5.3: 2D-Histogram of 10 000 frames raw amplitudes on board 0.

Noise Distribution

To determine the standard deviation σ_{ni}, noise data were collected at a measurement rate of 10 kHz with an integration time of 94.4 µs[1].

Fig. 5.2 shows the normalized histograms of the raw amplitudes for two specific channels on board 0. The number of entries is one million for each plot, corresponding to a measurement duration of 100 seconds at 10 kHz.

For channel 202, the distribution is approximately Gaussian with $\sigma_{202} = 8.95$ ADC counts. However, for channel 3, the distribution is discontinuous, exhibiting a clear gap around 990 ADC counts. This gap is consistently observed for all channels within the same photodiode array as in Fig. 5.3.

Fig. 5.3 illustrates the 2D histogram of the raw amplitudes on board 0. The black stripes indicate gaps that occur at the same positions for all channels within an array. Since each array is read out by a single ADC, these gaps may be associated with the ADC readout process. Notably, the gap structure is seen for three ADCs.

The ADCs used in the BPM are AD7983 devices, which are 16-bit successive approximation register (SAR) ADCs [57]. SAR ADCs evaluate the input voltage

[1] This corresponds to 10 000 data frames per second, with each frame having a shutter time of 94.4 µs.

through successive comparisons with voltages generated by an internal digital-to-analog converter (DAC). As explained by Dr. Michał Dziewiecki, the DAC can exhibit nonlinearity and, in the worst case, may even be non-monotonic, potentially leading to gaps in the ADC readout. Although such occurrences are rare—since most ADCs are guaranteed to be monotonic—in our case, the gap issue occurs on three out of five ADCs. The exact cause of these gaps remains under investigation. Nevertheless, they do not impact the uncertainty of the pedestal measurement due to the central limit theorem.

Fig. 5.3 also shows that the width of the noise distribution is relatively uniform across all channels in the five arrays. The average RMS is

$$\sigma_n \simeq 9\,.$$

This characteristic, namely that the noise is uniform across channels, is later used in the clustering algorithm, where the same threshold is applied to all the channels on a BPM plane. A more detailed analysis of the noise will be presented in Section 5.2.

Uncertainty

For board 0, when averaging over 10 000 frames for the pedestal calculation, the resulting uncertainty of the individual pedestal value is:

$$\sigma_{\bar{\mu}_i} = \frac{\sigma_{ni}}{\sqrt{N}} = \frac{9}{\sqrt{10000}} = 0.09$$

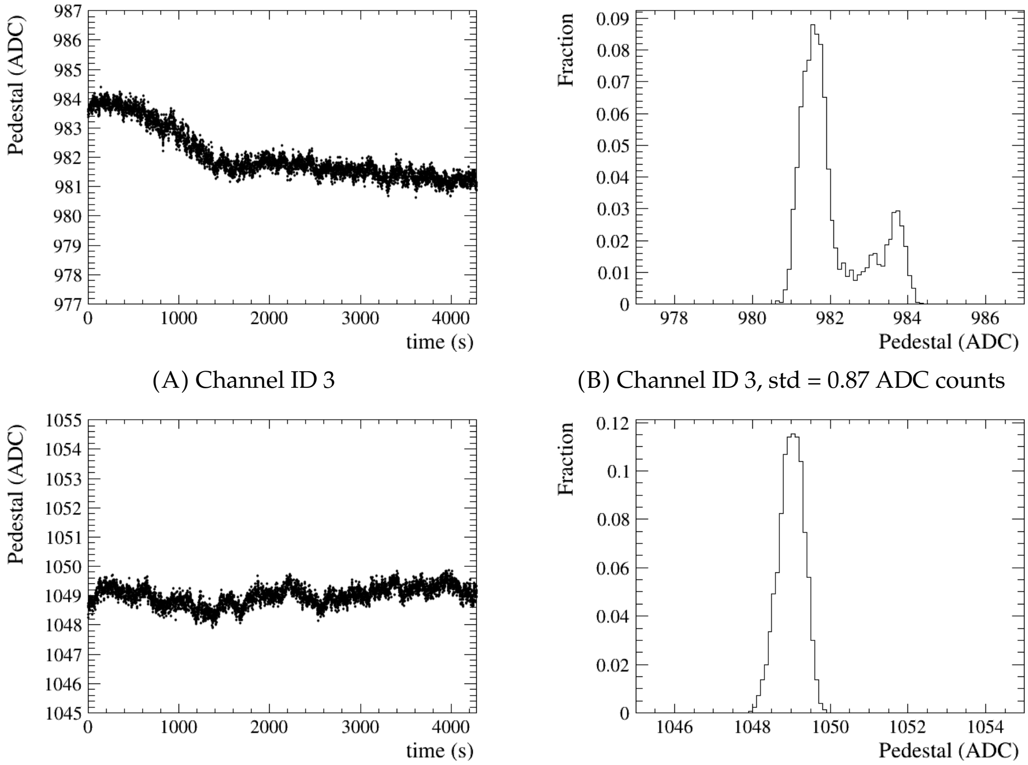
This means that the average pedestal value can be determined with a precision of 0.09 ADC counts. A typical sample of the pedestal value which is used later in the fpga real-time analysis uses 8 192 frames leading to an uncertainty of 0.10 ADC counts.

5.1.2 Pedestal over Time

At HIT, depending on the tumor size, a radiation treatment typically lasts about 10 to 30 minutes. To verify whether the average pedestal shifts over time, the raw amplitudes were continuously recorded for more than one hour at a readout rate of 10 kHz. Fig. 5.4 illustrates the average pedestal evolution over time for the two channels on board 0 as shown previously in Fig. 5.2. The average pedestal was calculated every 10 000 frames.

The pedestal of channel 202 remains stable throughout the entire measurement duration. In contrast, channel 3 exhibits a clear pedestal shift over time, decreasing by 2.5 ADC counts during the first 25 minutes. After that, the pedestal stabilizes for the remaining 45 minutes of the measurement. The cause of this shift is not yet clear.

Nevertheless, the shift is small enough to be buried in the noise, which has a standard deviation of 9 ADC counts for this board. In Section 5.4, the signal amplitude is constrained to be more than three times the noise width, i.e., above 27 ADC counts. This shift is therefore negligible, demonstrating that a single pedestal calculation at the beginning of a measurement (lasting 10 to 30 minutes, corresponding to a radiation treatment in future clinical use) is sufficient.



(A) Channel ID 3

(B) Channel ID 3, std = 0.87 ADC counts

(C) Channel ID 202

(D) Channel ID 202, std = 0.33 ADC counts

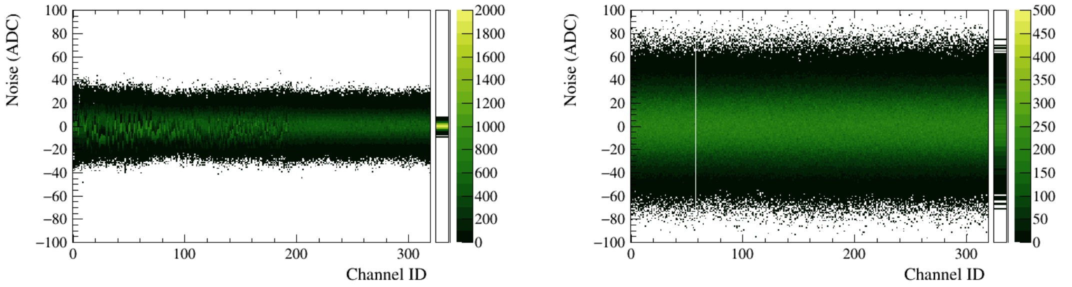
FIGURE 5.4: Pedestal (N=10000 average) over time and its 1D histogram for 2 specific channels on Board0 for 70 minutes.

5.1.3 Pedestal Subtraction

On the MAX10 FPGAs, the pedestal subtraction procedure is implemented such that the first 8 192 frames of each measurement are used to calculate the pedestal. During this period (0.8192 s at a 10 kHz measurement rate), the BPM indicates that the detector is not yet ready. Once the pedestal calculation is completed, the detector signals readiness, and the computed pedestal is subtracted from the raw amplitudes until the next measurement begins. Further details on the FPGA implementation and verification of the pedestal subtraction are provided in Chapter 6.

The remaining time-dependent pedestal shift affects the intensity measurements. A shifted pedestal value would be added to the signal, resulting in an incorrect intensity value. For high-precision offline analysis—such as the calibration factor measurement in Section 7.2, where all frames containing beam need to be summed—the measurement time is kept short (several minutes) to reduce the pedestal uncertainty. The pedestal subtraction is performed as follows: all frames outside the beam time are divided into two groups. The first group is used to calculate the pedestal, while the second group is used to evaluate the uncertainty in the summed signal after pedestal subtraction using the first group.

In Fig. 5.3, there are two types of channels. The first type, represented by channel 3, is observed in the first three arrays and exhibits the gap structure. The second type, represented by channel 202, is of the last two arrays. Channel 3 also serves as the worst-case scenario for pedestal shifting. It should be noted that the gap structure and the pedestal shift are not related. The pedestal shift has also be observed in the channel-202 type channels.



(A) Board 0

(B) Board 1

FIGURE 5.5: The 2D histograms of the noise distribution of 10 000 frames for all the channels for board0 and board1.

The cause of the gap structure and the pedestal shift warrants further investigation. However, for the time being, a mean pedestal shift of one or two ADC counts is unlikely to affect beam position measurements, as it is much smaller than the standard deviation of the pedestal.

5.2 Noise and Common Mode Subtraction

For a noise measurement, after pedestal subtraction on S_i as defined in Eq. 5.1, the noise δ_i remains, so that the entire frame becomes

$$\{\delta_0, \delta_1, \delta_2, \ldots, \delta_{319}\} \ .$$

The pedestal subtraction shifts the distribution of the raw noise signal from the average pedestal to zero, without changing its shape. Thus, the remaining noise follows the pedestal distribution centered around zero.

The width of the noise distribution for each channel can be expressed as

$$\sigma_n^2 = \sigma_\rho^2 + \sigma_\eta^2 \,,$$

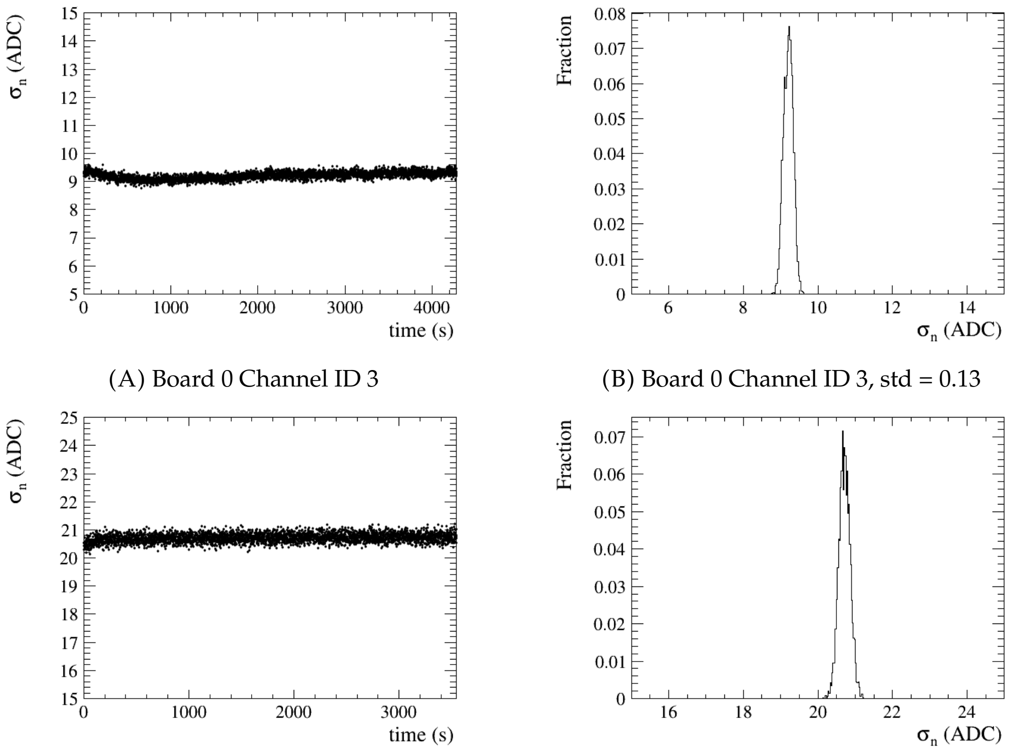
where σ_n is the total noise, σ_ρ is the noise correlated between channels (referred to as common-mode noise), and σ_η is the uncorrelated noise.

This section examines the characteristics of the noise. Subsection 5.2.1 shows that the noise is uniform across channels, while Subsection 5.2.2 demonstrates that it is stable over time. Subsection 5.2.3 quantifies the common-mode noise and describes the corresponding subtraction method. Finally, Subsection 5.2.5 shows that the noise remains stable with increasing integration time. Overall, this section establishes the noise characteristics that form the basis for further development of the reconstruction algorithm.

5.2.1 Noise on Channels

Fig. 5.5 shows histograms of the noise δ_i for all channels on two different BPM planes. The independent box represents the histogram of the mean noise, $\bar{\delta}$, for all 320 channels:

$$\bar{\delta} = \frac{1}{N} \sum_{i=0}^{N-1} \delta_i, \text{where } N = 320 \qquad (5.4)$$



(A) Board 0 Channel ID 3

(B) Board 0 Channel ID 3, std = 0.13

(C) Board 1 Channel ID 4

(D) Board 1 Channel ID 4, std = 0.15

FIGURE 5.6: The standard deviation of noise σ_n over time for two specific channels, one from board 0 and one from board 1.

Board 1 exhibits higher noise levels compared to board 0. The variation in $\bar{\delta}$ for board 0 is much smaller than the variation of any individual δ_i. However, for board 1, the variation in $\bar{\delta}$ is comparable to that of the individual δ_i

For both boards, the noise is uniform across their channels. This uniformity can be expressed by the equality of the noise standard deviation σ_n for any two channels i and j on the same board:

$$\sigma_{n_i} = \sigma_{n_j}$$

5.2.2 Noise over Time

Fig. 5.6 shows σ_n over time for one representative channel from board 0 and another from board 1. σ_n is the standard deviation of the pedestal-subtracted signal every 10 000 frames. For both boards, the value of σ_n remains stable over time.

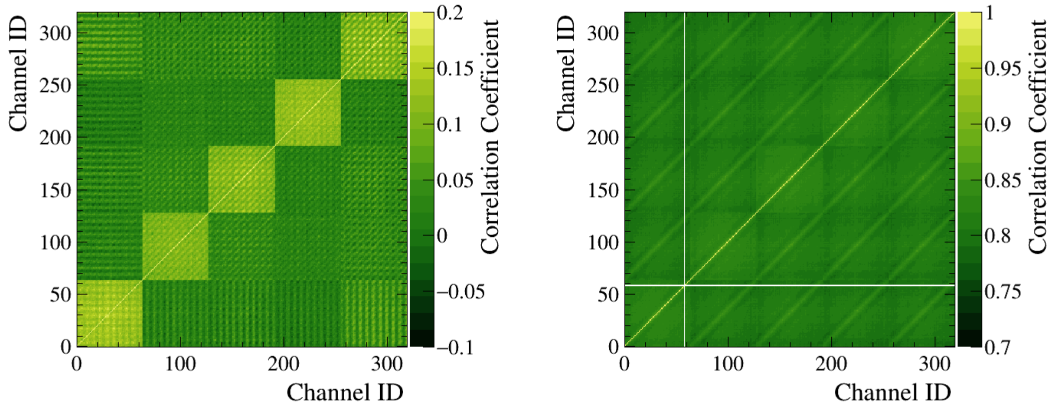
5.2.3 Noise Correlation

The correlation coefficient ρ_{ij} between channel i and channel j on the same board is calculated as:

$$\rho_{ij} = \frac{\mathrm{Cov}_{ij}}{\sigma_{ni}\sigma_{nj}} \tag{5.5}$$

where Cov_{ij} is the covariance between channels i and j, which is determined by:

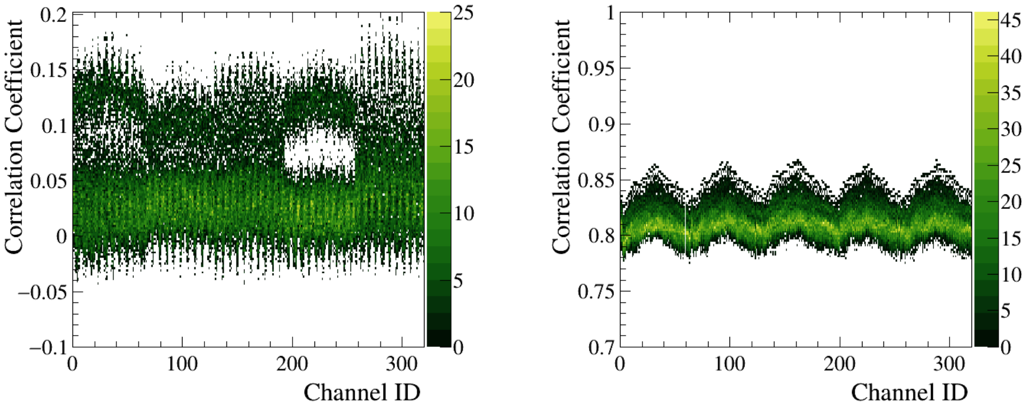
$$\mathrm{Cov}_{ij} = \frac{1}{N}\sum_{n}^{N}\left(\delta_i(n) - \bar{\delta}_i\right)\left(\delta_j(n) - \bar{\delta}_j\right) \tag{5.6}$$



(A) Board 0

(B) Board 1

FIGURE 5.7: Noise correlation ρ_{ij} maps for board 0 and board 1.



(A) Board 0

(B) Board 1

FIGURE 5.8: Noise correlation histograms for each channel.

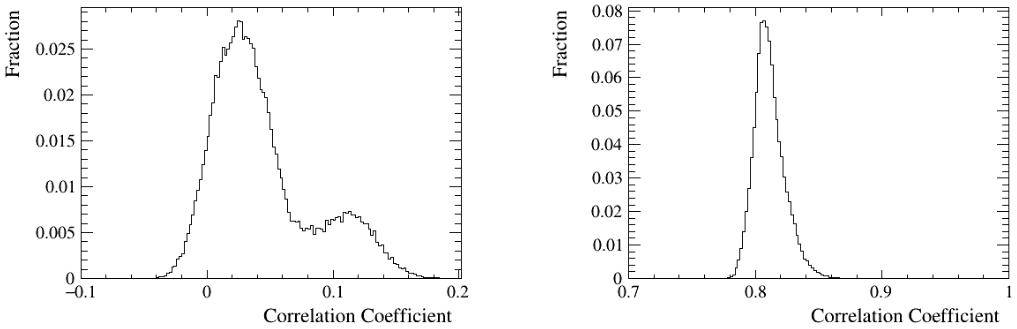
with N being the number of sampled frames, $\delta_i(n)$ the noise value of channel i at frame n, and $\bar{\delta}_i$ the mean noise of channel i over all sampled frames. Cov_{ij} represents the correlated noise, also known as common mode noise. The correlation coefficient ρ_{ij}, thus, indicates the portion of the correlated noise relative to the total noise.

ρ_{ij} ranges from -1 to 1. A value of 1 indicates that the two channels are perfectly correlated, meaning that all the noise is correlated noise. A value close to 0 indicates that the two channels are not correlated.

Fig. 5.7 shows the correlation coefficient ρ_{ij} maps for board 0 and board 1, with the x- and y-axes denoting the channel IDs and the z-axis representing ρ_{ij} in a color scale. The values of ρ_{ij} are calculated using Equations 5.5 and 5.6 over 10 000 noise frames.

In both plots, the diagonal elements from the bottom left to the top right are 1, representing the self-correlation of each channel. The horizontal and vertical white lines in the correlation map for board 1 are caused by a dead channel at channel ID = 58.

The Z-axis color scale range of the two plots is different. The correlation coefficient for board 0 is much smaller than that for board 1. The mean correlation coefficient is 0.04 for board 0 and 0.81 for board 1. This indicates that the noise on board 0 is mostly uncorrelated, while the noise on board 1 is highly correlated. This is consistent with the observation that the noise in board 1 is higher, indicating additional



(A) Board0, the mean is 0.04

(B) Board1, the mean is 0.81

FIGURE 5.9: Noise correlation 1D histograms.

correlated (common-mode) noise.

In Fig. 5.7A, the lighter squares around the diagonal represent the channels that belong to the same photodiode array. The correlation between channels in the same photodiode array is slightly larger than between channels from different arrays, as they share the same ADC, voltage supply, and readout line. The cause of the grid pattern inside an array is not yet clear. This is not a binning effect, as each bin in the x- and y-axes corresponds to a single channel, and the exact value of ρ_{ij} is used along the z-axis without any binning.

Fig. 5.7B shows a different pattern. Except for the dead channel, all other channels exhibit strong correlations. This may be caused by an unstable voltage level across the entire ADC board. The corner channels of each array appear slightly less correlated, indicating that the voltage level may be somewhat more stable in these regions. Furthermore, the diagonal strips from the bottom left to the top right indicate that channels from different arrays with the same channel order are more strongly correlated. For example, the first channel of the first array is more correlated with the first channel of the second array than with other channels in the second array.

Fig. 5.8 presents 2D histograms of the correlation coefficients, with 319 entries for each channel. Each x-bin represents a channel, and the y-axis denotes its correlation coefficient with other channels. Fig. 5.9 shows the corresponding projection on the y-axis for all channels. The values of ρ_{ij} are relatively low and sparsely distributed for board 0, whereas for board 1, they are high and densely clustered.

The channels on the same ADC board exhibit similar distributions of correlation coefficients with the other channels. Aside from the dead channel, there is no specific channel that is significantly more or less correlated with the remaining 319 channels. Therefore, the mean correlation coefficient provides a reasonable approximation for the correlation coefficient of each individual channel.

$$\rho = 0.04 \text{ for board } 0\text{, and } 0.81 \text{ for board } 1$$

The presence of high common-mode noise is problematic, as it makes it difficult to distinguish between noise and beam profile. Strong common-mode noise can produce bumps across several channels that resemble a beam profile.

In a Gaussian distribution, the one-sided tail probability above $+3\sigma$ is about 0.15%. If a single channel has an amplitude above 3σ, where σ is the standard deviation of its noise, the probability that it is due to noise is 0.15%. If this channel and its neighboring channel are both above 3σ, and the channels are independent, the probability that this is due to noise is 2.25×10^{-6}. However, if the two channels are highly correlated, for example with $\rho = 1$, the probability that it is due to noise remains 0.15%. Thus, with strong common-mode noise, the probability of neighboring channels simultaneously exceeding the threshold is much higher than in the case of weak common-mode noise. This increases the probability of a false positive signal. For this reason, common-mode noise must be carefully studied and subtracted.

5.2.4 Common Mode Subtraction

In the presence of correlated noise—also known as common mode noise—which affects all channels equally, the signal values shift up and down simultaneously across all channels. In such cases, common mode subtraction is essential for precise signal processing.

The common mode noise can be calculated as the mean value of all the channels, $\bar{\delta}$, as shown in Eq. 5.4. The variance of the common mode noise $\bar{\delta}$ is given by:

$$\mathrm{Var}\left(\bar{\delta}\right) = \mathrm{Var}\left(\frac{1}{N}\sum_{i=0}^{N-1}\delta_i\right) = \frac{1}{N^2}\sum_{i=0}^{N-1}\mathrm{Var}\left(\delta_i\right) + \frac{2}{N^2}\sum_{0\leqq i<j\leqq N-1}\mathrm{Cov}_{ij} \tag{5.7}$$

With the definition of the correlation coefficient in Eq. 5.5, and assuming that σ_n is equal across all channels, and ρ represents the average correlation coefficient, the equation above can be rewritten as:

$$\mathrm{Var}\left(\bar{\delta}\right) = \frac{\sigma_n^2}{N} + \frac{N-1}{N}\rho\sigma_n^2 = \left[\rho + \frac{1}{N}\left(1-\rho\right)\right]\sigma_n^2 \tag{5.8}$$

When

$$\rho = 0, \mathrm{Var}\left(\bar{\delta}\right) = \frac{\sigma_n^2}{N},$$

this represents the variance of the mean of independent variables. When

$$N \to \infty, \mathrm{Var}\left(\bar{\delta}\right) = \rho\sigma_n^2$$

This is the variance of the common mode noise σ_ρ^2.

With the common mode noise $\bar{\delta}$ subtracted:

$$\delta_i' = \delta_i - \bar{\delta}$$

the variance of δ_i' is:

$$\mathrm{Var}\left(\delta_i'\right) = \mathrm{Var}\left(\delta_i - \bar{\delta}\right) = \mathrm{Var}\left(\delta_i\right) + \mathrm{Var}\left(\bar{\delta}\right) - 2\mathrm{Cov}_{i\bar{\delta}} \tag{5.9}$$

The covariance between the channel i and the common mode noise $\bar{\delta}$ is:

$$\mathrm{Cov}_{i\bar{\delta}} = \frac{1}{N}\sum_{j=0}^{N-1}\mathrm{Cov}_{ij} = \frac{1}{N}\left(\sigma_n^2 + (N-1)\rho\sigma_n^2\right) = \mathrm{Var}\left(\bar{\delta}\right) \tag{5.10}$$

The channel i is also included in the common mode noise $\bar{\delta}$. When $i = j$, $\mathrm{Cov}ij = \sigma_n^2$, and when $i \neq j$, $\mathrm{Cov}ij = \rho\sigma_n^2$.

Thus, the variance of the noise for each channel after common mode subtraction is:

$$\mathrm{Var}\left(\delta_i'\right) = \mathrm{Var}\left(\delta_i\right) - \mathrm{Var}\left(\bar{\delta}\right) = (1-\rho)\left(1 - \frac{1}{N}\right)\sigma_n^2 \tag{5.11}$$

With $N \to \infty$, Var (δ_i') is $(1-\rho)\,\sigma_n^2$, which is the variance of the uncorrelated noise σ_η^2. This means that when N is really large, the common mode noise can be subtracted efficiently.

In practice, we usually choose several reference channels and ensure that these channels are not seeing signal, meaning they only measure noise. Instead of averaging the signal from all the channels, which is not realistic, the common mode noise is calculated by averaging the signals of the reference channels. Thus, the common mode noise is represented as:

$$\bar{\delta}_{\text{ref}} = \frac{1}{n}\sum_{\text{ref}} \delta_i$$

where n is the number of reference channels.

Since the channel itself is not included in $\bar{\delta}_{\text{ref}}$, Eq. 5.10 changes. The covariance between channel i and the common mode noise $\bar{\delta}_{\text{ref}}$ is:

$$\text{Cov}_{i\bar{\delta}_{\text{ref}}} = \frac{1}{n}\sum_{\text{ref}} \text{Cov}_{ij} = \rho\sigma_n^2 \tag{5.12}$$

Thus, the variance of the noise δ_i'' for each channel after the realistic common mode subtraction is:

$$\text{Var}\left(\delta_i''\right) = (1-\rho)\left(1+\frac{1}{n}\right)\sigma_n^2 \tag{5.13}$$

The common mode subtraction reduces the variance, if the factor $(1-\rho)\left(1+\frac{1}{n}\right)$ is smaller than 1, which can be rearranged to:

$$n > \frac{1}{\rho} - 1, \text{when } \rho > 0 \tag{5.14}$$

In this case, n should be larger than 24 for board 0 and larger than 0.2 for board 1 in order to reduce the noise through common-mode subtraction. Since n must be an integer, this means that for board 1 a single reference channel is sufficient to reduce part of the common-mode noise, whereas 25 channels are required for board 0.

Because board 0 exhibits low common-mode noise, common-mode subtraction is unnecessary, while any value of n would only worsen the noise unless $n \geq 25$. Therefore, common-mode subtraction is applied only to board 1.

The larger the value of n, the more common-mode noise can be subtracted; however, as n increases, the number of channels available for the signal decreases. For the board 1 type BPM, n is chosen to be 4. For convenience in the FPGA implementation, the first four channels of the first array are used.

Discussion

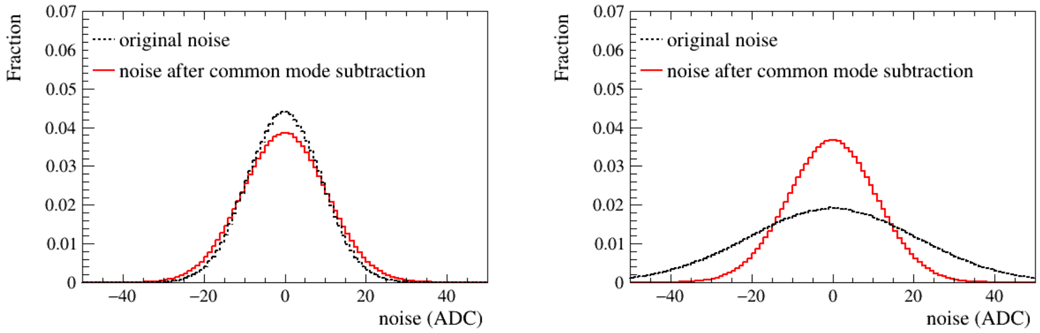
The total noise σ consists of two components: the common mode noise σ_ρ and the uncorrelated noise σ_η. These are related by:

$$\sigma_n^2 = \sigma_\rho^2 + \sigma_\eta^2 \tag{5.15}$$

With ρ is the correlation coefficient, the individual components can be expressed as:

$$\sigma_\rho^2 = \rho\sigma_n^2, \quad \text{and} \quad \sigma_\eta^2 = (1-\rho)\sigma_n^2 \tag{5.16}$$

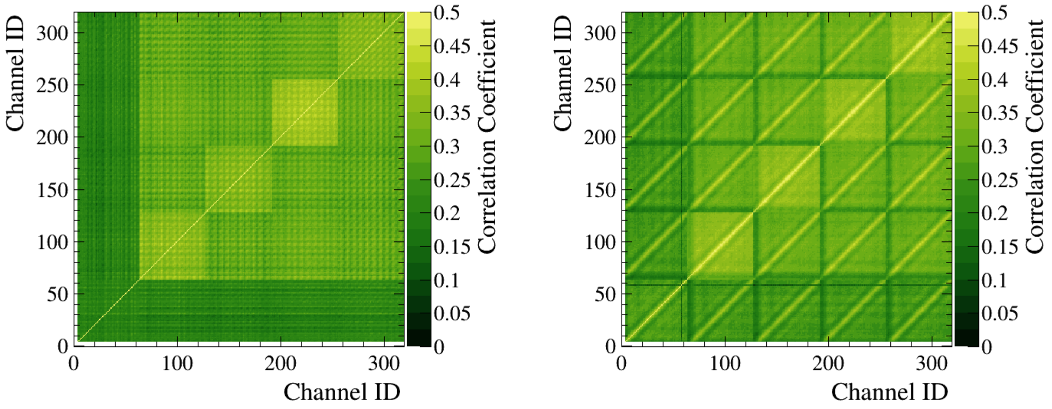
Fig. 5.10 shows the noise histograms before and after the common mode subtraction. Fig. 5.11 shows the correlation coefficient maps after the common mode



(A) Board 0

(B) Board 1

FIGURE 5.10: Noise histogram for all the channels before and after common mode subtraction.



(A) Board 0

(B) Board 1

FIGURE 5.11: Noise correlation ρ''_{ij} after the common mode subtraction for board 0 and board 1.

Quantity	**Condition**	**Board 0**	**Board 1**
σ_n	Before subtraction	9.00	21.00
	After subtraction	10.29	10.88
ρ	Before subtraction	0.04	0.81
	After subtraction	0.28	0.30

TABLE 5.1: Standard deviation of the total noise σ_n (in ADC counts) and the correlation coefficient ρ before and after common mode subtraction. Values are calculated over 10 000 frames. Reference channels are excluded after subtraction.

subtraction. Tab. 5.1 summarizes the standard deviations of the noise distributions and the correlation coefficient ρ before and after common mode subtraction. The common mode subtraction increases the common mode noise for Board0, while it nicely reduces the common mode noise for Board1, as expected.

Quantity	Board 0	Board 1
σ_n	9.00	10.88
σ_ρ	1.80	5.78
σ_η	8.82	9.10

TABLE 5.2: Noise components in ADC counts with the common mode subtraction applied only to Board 1.

Tab. 5.2 summarizes the noise value with the common mode subtraction for board 1 only. The uncorrelated noise is the dominant component and is similar for both boards.

We have six ADC boards in total: two of them (board 0 type) were manufactured in 2018, and four (board 1 type) in 2022. Their noise characteristics are consistent with those of board 0 and board 1 discussed here; specifically, all boards manufactured in 2022 exhibit high common-mode noise of about 0.81, while all boards from 2018 exhibit low common-mode noise. Further studies at the electronics level are required to understand the observed behavior. If all the boards behave similarly to Board 0, the common mode subtraction is not necessary.

5.2.5 Noise with Integration time

Readout Rate	Integration Time	σ_n Board 0	σ_n Board 1
(kHz)	(µs)	(ADC counts)	(ADC counts)
10	94.4	9.01	10.86
8	119.3	9.06	10.85
6.7	144.2	9.06	10.90
5	194.3	8.93	10.88
4	244.2	9.08	10.92

TABLE 5.3: The standard deviation of the total noise σ for different integration times.

Raw noise was collected at different integration times. Tab. 5.3 presents the noise values at these integration times, calculated as the standard deviations over 10 000 data frames each. Common-mode subtraction is applied to board 1 but not to board 0. Each entry in the table corresponds to one measurement. For both boards, the noise remains stable across the range of integration times, indicating that the noise level is not affected by the integration time. Consequently, the integration time can be increased to improve the signal without introducing additional noise.

5.3 Beam Profile Characteristics

After pedestal subtraction and common-mode subtraction, the remaining amplitudes have a mean around zero and a standard deviation of 9 or 11 ADC counts, depending on the board type, in the case of pure noise. In the presence of a beam, the beam profile is detected.

For further development of the reconstruction algorithm, this section describes the characteristics of the beam profile measured by the SciFi BPM. The beam profile typically spans a few channels and may contain outliers or dead channels, which motivates the clustering algorithm in Section 5.4. Its generally Gaussian shape further motivates the Gaussian-like beam reconstruction in Section 5.6.

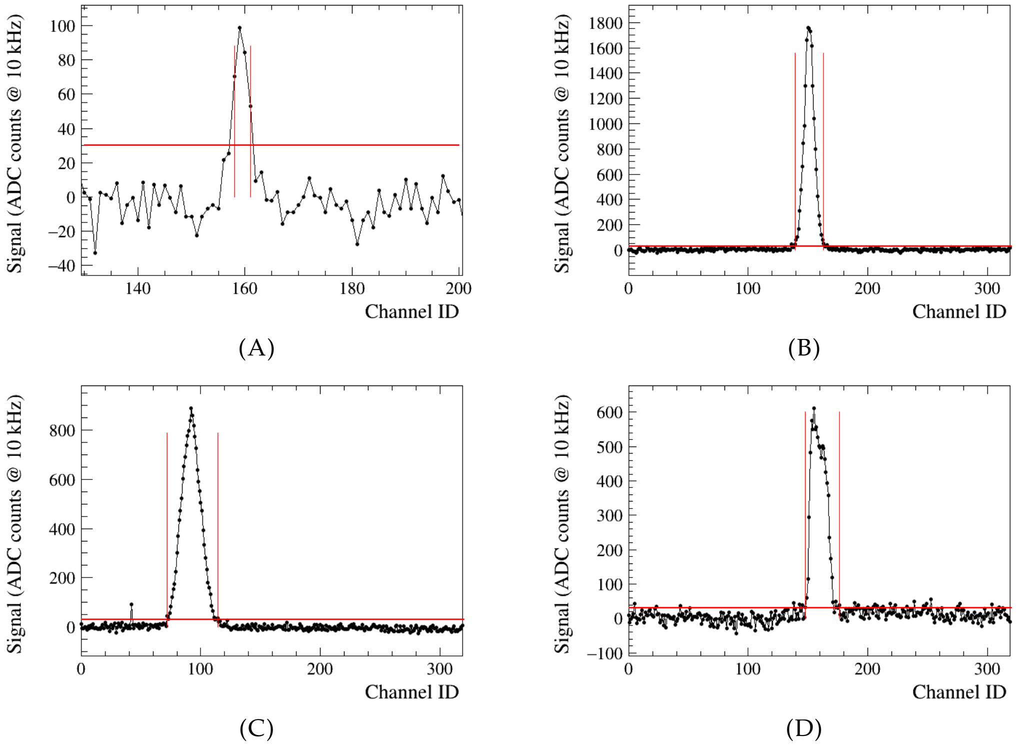


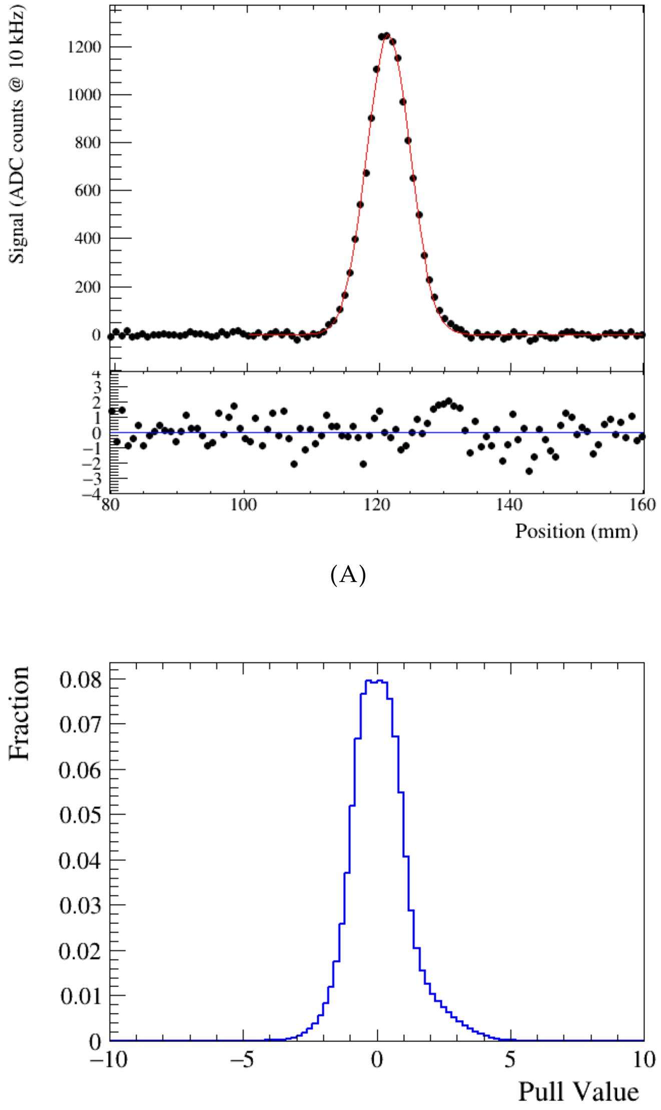
FIGURE 5.12: Examples of beam profiles from the test beam. (A) Low-intensity proton beam, 1.2×10^8 s^{-1}; (B) high-intensity proton beam, 2×10^9 s^{-1}; (C) beam profile with an outlier; (D) asymmetric beam profile.

Fig. 5.12 shows beam profile examples collected during the beam tests at HIT, when the SciFi-BPMs were placed close to the nozzle, as in Fig. 7.1. These beam profiles exhibit the following features:

- The ion beams do not illuminate all 320 channels. In the test beam measurements, the beam size at the location of the SciFi-BPMs ranges from 2.5 mm to 10 mm. According to the beam specifications in Tab. 3.1, the beam size at the isocenter should range from 2.7 mm to 32.9 mm. Based on measurements from the MWPCs at the nozzle, the beam size at the nozzle ranges from 2 mm to 8 mm. For future measurements performed at the nozzle for online monitoring and at the isocenter for quality assurance (QA), the beam profiles typically span more than 3 channels but fewer than 50 channels.
- Most signals are symmetrical, although the horizontal profiles of some special carbon beams are not, as shown in Fig. 5.12D. This asymmetry is related to the RF-knockout beam extraction process [40]. The position and width reconstruction for these asymmetric signals can be performed using the RMS method in Section 5.6.1.

- Outliers are observed in the form of single channel with unusually high signals, as seen in Fig. 5.12C. This might be related to scattered particles or δ-electrons irradiating the photodiode directly.

Additionally, there are dead channels whose amplitudes remain zero. In the test beam campaigns in 2024, only one dead channel was observed out of six 320-channel planes.



(A)

(B)

FIGURE 5.13: (A) A typical proton beam profile measured by the SciFi-BPM and its Gaussian fit, along with the corresponding pull distribution. (B) The pull distribution histogram for 262 901 frames of such a beam as shown in (A).

To investigate whether the symmetric beam profile follows a Gaussian shape, Fig. 5.13A shows a proton beam profile fitted with a Gaussian function, along with its corresponding pull distribution. The x-axis represents the position in millimeters, with the relation between position and channel ID given by Eq. 5.30. Each dot in Fig. 5.13A corresponds to the amplitude or the pull of one channel.

The beam is a high-intensity 95.3 MeV proton beam with an intensity of $2 \times 10^9\,\mathrm{s}^{-1}$. The pull for each channel is defined as

$$\frac{y - y_{\mathrm{fit}}}{\sigma}, \tag{5.17}$$

where y is the amplitude of the channel, y_{fit} is the fitted Gaussian value, and σ is the associated uncertainty. For each channel, σ is defined as

$$\sigma = \sqrt{|y| + \sigma_\eta^2}, \tag{5.18}$$

where $\sqrt{|y|}$ represents the statistical uncertainty of y[2], and σ_η is the electronic noise contribution, i.e., the uncorrelated noise.

Fig. 5.13B presents the 1D histogram of the pull values for 262 901 frames of proton beams. In these frames, the proton beam spot position on the detector is not fixed but corresponds to a line scan (see the explanation in Subsection 7.1.3). The pull distribution for this scan has a mean of approximately 0.10 and a standard deviation of 0.97, indicating that the Gaussian function provides a good description of the beam profile.

5.4 Clustering

The beam profile spans only a few channels, which must be identified using a clustering algorithm. This section describes the clustering algorithm and evaluates its performance at different levels of common-mode noise.

5.4.1 Clustering Algorithm

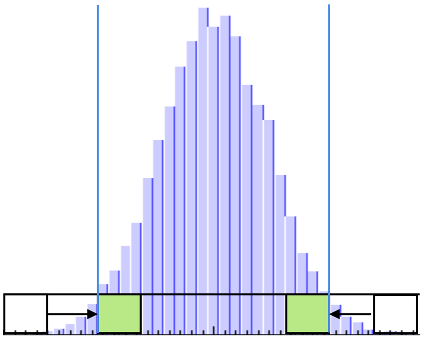


FIGURE 5.14: Sketch of the clustering algorithm.

The clustering algorithm is designed to determine the presence of a signal and, if present, to identify the beam profile edges while ignoring outliers and dead channels.

Fig. 5.14 illustrates the clustering algorithm. The purple histogram represents a beam profile, with the boxes searching for its edges. The clustering algorithm has two parameters: the cluster threshold T_c, represented by the height of the boxes and

[2] The beam profiles used for the pull calculation are calibrated profiles. Thus, the statistical uncertainty $\sqrt{|y|}$ accounts not only for the intrinsic Poisson fluctuations of the raw ADC amplitude, but also for contributions from the calibration process. Details of the calibration procedure is given in Section 7.2.

given in units of amplitude; and the cluster size N_c, represented by the width of the boxes and given in units of channel number. The term "cluster" here denotes a contiguous group of N_c channels with amplitudes exceeding the threshold T_c. In the sketch, the green boxes highlight the leftmost and rightmost clusters. The clustering algorithm records the positions of these edge clusters, assigning the left edge of the leftmost cluster as the left boundary of the beam profile and the right edge of the rightmost cluster as the right boundary. The channels between these two edges, including the edge channels, define the window of interest, shown between the two vertical blue lines in the sketch. The channels within this window are used to reconstruct the beam profile.

In the case where only one cluster is found, its edges directly define the window of interest. In this situation, the window length is N_c channels, which is the minimum possible length of the window of interest defined by the clustering algorithm at setting N_c.

Outliers appear as single channels with unusually high amplitudes. Since the clustering algorithm requires several consecutive channels to form a cluster, isolated outlier channels outside the beam region are effectively ignored.

Dead channels are those whose amplitudes remain zero. The clustering algorithm only considers the two edge clusters, and therefore does not depend on the channels in between. Consequently, dead channels within this region do not affect the window of interest. This is advantageous when dead channels are present, as it allows the clustering algorithm to encompass the entire beam region, even if some channels are non-functional.

The performance of the clustering algorithm is a probabilistic problem, where T_c, N_c, and the noise level jointly determine the trade-off between the false positive rate and the efficiency.

The probability that an amplitude exceeds the threshold T_c due to noise is given by the upper-tail probability of the noise distribution,

$$p(A > T_c) = 1 - F(T_c),$$

where $F(T_c)$ is the cumulative distribution function of the noise amplitude. For example, when $T_c = 3\sigma_n$, with σ_n denoting the standard deviation of the noise, $p(A > T_c) \approx 0.15\%$. This probability depends on the shape of the noise distribution and on the ratio T_c/σ_n, rather than on the absolute value of T_c.

In the absence of common-mode noise, the channels can be considered independent, and the probability that the clustering algorithm produces a false positive due to noise is

$$p_{\text{false\_positive}} = \left[p(A > T_c)\right]^{N_c}. \tag{5.19}$$

In the presence of common-mode noise, the assumption of independence no longer holds, as correlations between channels increase the likelihood of simultaneous threshold crossings. In this case, the false positive probability depends on the correlation coefficient ρ between channels. A general expression can be written as

$$p_{\text{false\_positive}} = P\big(A_1 > T_c, A_2 > T_c, \ldots, A_{N_c} > T_c\big), \tag{5.20}$$

which corresponds to the upper-tail probability of an N_c-dimensional correlated normal distribution with a covariance matrix determined by the common-mode noise level ρ.

According to Eq. 5.19, increasing T_c/σ_n reduces $p_{\text{false\_positive}}$, as does increasing

N_c. At the same time, according to Eq. 5.20, stronger correlations (higher common-mode noise) increase the probability that several consecutive channels simultaneously exceed T_c, thereby raising the false positive rate.

On the other hand, larger values of T_c/σ_n and N_c may cause the algorithm to miss low-amplitude or narrow beam profiles, thereby reducing its detection efficiency.

Subsection 5.4.2 evaluates the performance of the clustering algorithm for different values of T_c/σ_n, N_c, and common-mode noise levels ρ, focusing on the false positive rate and the efficiency.

The red vertical lines in Fig. 5.12 indicate the windows of interest identified by the clustering algorithm, with $T_c = 30$ ADC counts shown as the horizontal red line and $N_c = 4$ channels. T_c is chosen to be the same for all channels of a BPM plane, since the noise level is stable over time and it is uniform across channels on the same plane, as discussed in Section 5.2. In Fig. 5.12A, only four channels exceed the threshold, and the edges are correctly identified. In Fig. 5.12C, the algorithm successfully ignores the outlier channel.

The key features of this clustering algorithm are its flexibility and simplicity. The flexibility lies in the ability to optimize performance by adjusting the parameters T_c and N_c. The simplicity of the algorithm stems from the fact that it requires only a single pass through the data to identify the edge clusters, making it computationally efficient and well-suited for implementation on FPGAs.

5.4.2 Performance

The performance of the clustering algorithm can be evaluated with respect to the false positive rate $p_{\text{false\_positive}}$ and the efficiency ϵ. Ultimately, the acceptable levels of $p_{\text{false\_positive}}$ and ϵ are constrained by the tolerable interlock rate in clinical operation.

False Positive Rate, Efficiency, and Interlock Rate

The false positive rate $p_{\text{false\_positive}}$ is defined as the probability of identifying a window of interest when no signal is present, while the efficiency ϵ is the probability of identifying a window of interest when a signal is present.

In this section, $p_{\text{false\_positive}}$ is evaluated for different values of T_c/σ_n, N_c, and ρ using Monte Carlo simulations. In the toy simulation, only noise data are generated, and the clustering algorithm is applied to search for false positives. $p_{\text{false\_positive}}$ and its uncertainty are then computed as

$$p = \frac{k}{N} \pm \sigma_{prob}, \tag{5.21}$$

where N is the total number of simulated noise frames and k is the number of frames in which the algorithm identifies the presence of the cluster. The uncertainty σ_{prob} is calculated using Bayesian statistics [63]:

$$\sigma^2_{prob} = \frac{(k+1)(k+2)}{(N+2)(N+3)} - \frac{(k+1)^2}{(N+2)^2}. \tag{5.22}$$

In the presence of a signal, the false positives can cause the window of interest to be determined incorrectly, leading to a bias in the reconstructed beam position. Therefore, the false positive rate determined using the noise-only frames can also be interpreted as the probability of incorrectly identifying the window of interest.

Analogous to the false positive rate, the efficiency ϵ is determined using a toy simulation that includes both noise and simulated Gaussian beam profiles. The efficiency and its uncertainty are then calculated using Eqs. 5.21 and 5.22.

During an irradiation campaign at HIT, four consecutive "position not confirmed" signals within a time window of 1 ms result in an interlock. For the SciFi-BPM, HIT requested that an interlock be triggered after three such consecutive signals.

A "position not confirmed" event occurs when the beam is on, either because no cluster is found or because the window of interest is determined incorrectly, resulting in the position not being confirmed. The former can be quantified by $1 - \epsilon$, and the latter by $p_{\text{false\_positive}}$. Therefore, the interlock rate $p_{\text{interlock}}$ can be expressed as

$$p_{\text{interlock}} = (1 - \epsilon + p_{\text{false\_positive}})^3. \tag{5.23}$$

A strict interlock rate target is less than one per day. In order to reduce $p_{\text{interlock}}$ to less than one per day, $\epsilon - p_{\text{false\_positive}}$ should be greater than 99.89 %. So one option is:

$$\epsilon > 99.9\% \text{ and } p_{\text{false\_positive}} < 0.01\%.$$

The following subsections evaluate $p_{\text{false\_positive}}$ using noise-only simulations, determine ϵ using simulations including both signal and noise, and finally discuss the interlock rate as a function of the common-mode noise level ρ.

False Positive Rate Evaluation

Four values of ρ were considered: 0.00, 0.04, 0.30, and 0.81. A value of $\rho = 0.00$ corresponds to the case with no common mode noise; $\rho = 0.04$ represents the typical value for the board 0-type detector plane; and $\rho = 0.81$ and $\rho = 0.30$ correspond to the board 1-type detector plane before and after common mode subtraction, respectively.

To evaluate $p_{\text{false\_positive}}$, for each value of ρ, 10^8 frames of noise-only data were generated, with each frame consisting of 320 channels. At a readout rate of 10 kHz, 10^8 frames correspond to approximately 2.8 hours of data acquisition.

For channel i in frame k, the noise value is generated as

$$\delta_N^{ik} = \delta_\eta^{ik} + \delta_\rho^k, \text{ where } \delta_\eta \sim \mathcal{N}(0, \sigma_\eta^2), \quad \delta_\rho \sim \mathcal{N}(0, \sigma_\rho^2), \tag{5.24}$$

where $\mathcal{N}(0, \sigma^2)$ denotes a Gaussian distribution with mean 0 and standard deviation σ. σ_η and σ_ρ are the standard deviations of the uncorrelated noise and the common-mode noise, respectively. They are calculated from ρ and the total noise σ_n according to Eqs. 5.16.

δ_ρ^k, the common-mode noise, is sampled once per frame from $\mathcal{N}(0, \sigma_\rho^2)$, whereas δ_η^{ik}, the uncorrelated noise, is sampled independently for each channel in each frame from $\mathcal{N}(0, \sigma_\eta^2)$.

The clustering algorithm was then applied to these frames, and $p_{\text{false\_positive}}$ was calculated for various combinations of T_c/σ_n and N_c at different values of ρ.

Fig. 5.15 shows $p_{\text{false\_positive}}$ at different conditions. The horizontal thick red line shows the 0.01% line. Solid lines are with $T_c = 3\sigma_n$, while dashed lines are for $T_c = 2\sigma_n$[3]. The bands show $\pm 3\sigma_{prob}$ region. For probability equal to 10^{-8}, zero clusters are found in the 10^8 noise frames. Larger T_c/σ_n and N_c values and smaller ρ decreases the probability of false positive.

[3] In this study, $T_c/\sigma_n = 2$ or 3 are considered, although it does not need to be an integer.

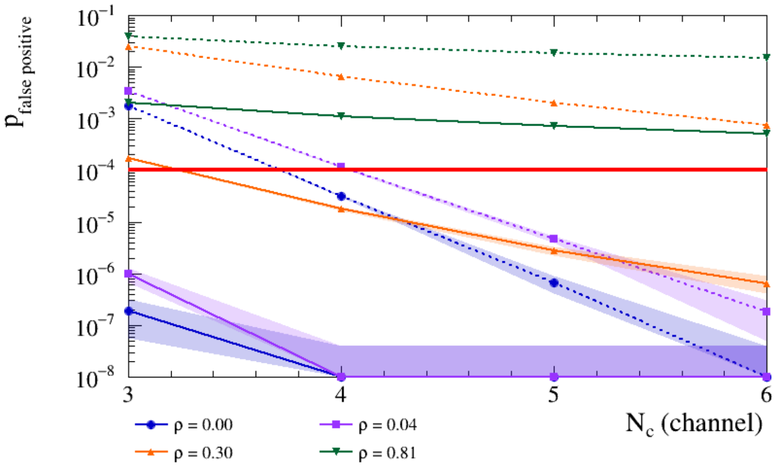


FIGURE 5.15: The probability of false positive at different T_c/σ_n, N_c and ρ.

For board 0-type planes, as shown in Fig. 5.15, $p_{\text{false\_positive}}$ remains below 0.01% when the clustering algorithm is configured with

$$N_c = 3, T_c = 3\sigma_n.$$

With the common-mode subtraction, $p_{\text{false\_positive}}$ also remains below 0.01% for board 1-type planes with

$$N_c = 4, T_c = 3\sigma_n.$$

T_c sets a constraint for the amplitude of the beam profile, while N_c sets a constraint for the width of the beam profile. For a narrow, high-amplitude beam, T_c can be set high, while N_c does not need to be large to maintain low $p_{\text{false\_positive}}$. For a wide, low-amplitude beam, N_c can be set high with a loose T_c. The difficulty arises in the case of narrow and low-intensity beam profiles.

The narrowest beam at the nozzle has a width of 2 mm, which at low intensities may cover only three channels. To detect such beams, N_c has to be set to 3. In this case, board 0 type performs well at $T_c = 3\sigma_n$, while the $p_{\text{false\_positive}}$ of board 1 type is slightly larger than 0.01%.

This indicates that, for a system containing both board 0 type and board 1 type, the clustering algorithm results from board 0 type provide the trustworthy signal for determining the presence of the beam, even with the narrowest beam. The board 0 type planes are employed for determining the beam-on time in Subsection 7.1.3.

The clustering algorithm on FPGA is implemented with the flexibility to modify N_c and T_c via HITDAQ.

Efficiency Evaluation

The efficiency of the clustering algorithm is dependent on the signal amplitude, as a larger signal has a higher probability of being detected with the same clustering settings. To evaluate the efficiency, the definition of the signal-to-noise ratio (SNR) is introduced. Beam profiles with different SNR values are then generated using a toy Monte Carlo simulation.

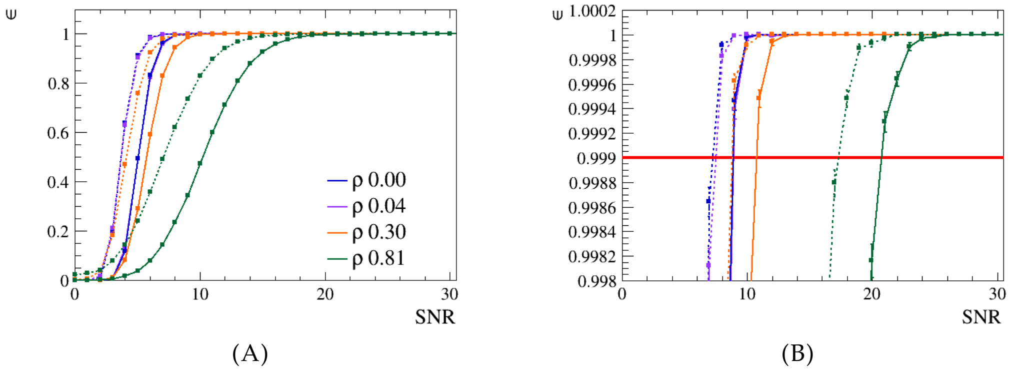


FIGURE 5.16: (A) Efficiency of the clustering algorithm for a beam with FWHM of 5 mm. (B) Zoomed-in plot of (A).

The ideal Gaussian beam is defined as:

$$y = y(x, A, \mu, \sigma) = Ae^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{5.25}$$

where A is the amplitude, μ is the mean, and σ is the standard deviation of the beam profile. The relation between the FWHM[4] and the standard deviation σ of a Gaussian beam is given by:

$$\text{FWHM} = 2\sqrt{2\ln(2)}\,\sigma \approx 2.355\,\sigma. \tag{5.26}$$

The signal-to-noise ratio (SNR) is defined as:

$$\text{SNR} = \frac{A}{\sigma_\eta} \tag{5.27}$$

where σ_η is the standard deviation of the uncorrelated noise. With this definition, the SNR is independent of the common mode noise. It reflects the amplitude of the signal relative to the noise within each frame σ_η. As discussed in Section 5.2.4, σ_η is similar for both Board 0 and Board 1 types. Therefore, the SNR primarily indicates the signal amplitude.

To evaluate the efficiency ϵ of the clustering algorithm under different ρ and SNR conditions, a simulated beam with a fixed FWHM of 5 mm was used. The integer SNR varied from 0 to 100 in steps of 1, and the beam position μ was randomly assigned for each frame. For each SNR value, 10^5 frames were generated. For frame k, the signal in channel i was generated as:

$$S^{ik} = y^{ik} + \delta_y^{ik} + \delta_N^{ik}\text{ , where } y^{ik}(x_i, \text{SNR} \times \sigma_\eta, \mu, \sigma) \text{ and } \delta_y^{ik} \sim \mathcal{N}(0, y^{ik}) \tag{5.28}$$

The term δ_y^{ik} represents the statistical fluctuation of the signal y^{ik}, with a standard deviation equal to $\sqrt{y^{ik}}$, while δ_N^{ik} denotes the total noise contribution as in Eq. 5.24.

Fig. 5.16 shows the efficiency ϵ as a function of SNR for different values of ρ. The clustering settings are $N_c = 4$, with $T_c = 3\sigma_n$ for the solid lines and $T_c = 2\sigma_n$ for the dashed lines. The higher the T_c/σ_n and ρ, the higher the SNR required to achieve 99.9% efficiency. The plot shows that, ϵ exceeds 99.9% for $\rho = 0.04$ when the SNR is larger than 7 at $T_c = 2\sigma_n$ and larger than 9 at $T_c = 3\sigma_n$. For $\rho = 0.30$, an SNR greater

[4] In this thesis, the terms FWHM, width, and focus are used to refer to the beam width.

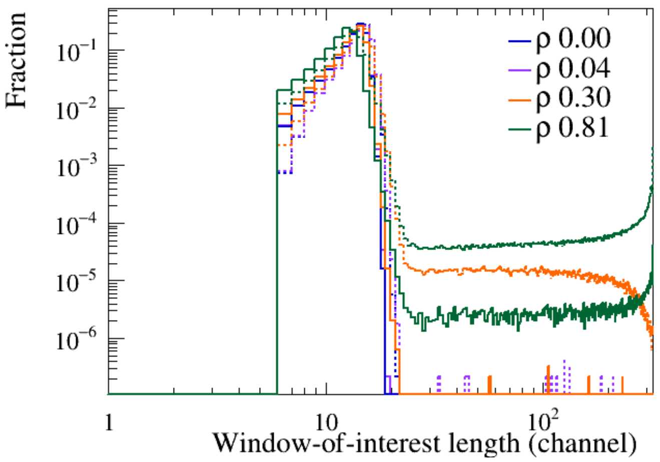


FIGURE 5.17: Normalized histograms of the window-of-interest length for events containing a cluster, obtained from the simulation of a 5 mm-wide beam with SNR values ranging from 0 to 100. The number of entries is approximately 9×10^6 for each curve.

than 9 or 11 is required to achieve the same efficiency. For $\rho = 0.81$, the required SNR increases to above 18 or 21.

This means that the common mode subtraction is necessary for the board 1-type detector plane to increase the efficiency of the clustering algorithm.

Fig. 5.17 shows the distribution of the window-of-interest length for this simulation. The clustering settings are $N_c = 4$, with $T_c = 3\sigma_n$ for the solid lines and $T_c = 2\sigma_n$ for the dashed lines. The tails above the peak region are caused by fake clusters, with a probability of $p_{\text{false\_positive}}$. The incorrectly identified windows in the tails lead to a biased position. A setting of $T_c = 3\sigma_n$ is therefore necessary to suppress $p_{\text{false\_positive}}$ and reduce the number of events with biased position.

5.4.3 Discussion

This section describes the clustering algorithm and its performance in the presence of varying levels of common-mode noise.

$$N_c = 4 \text{ channels}, \quad T_c = 3\sigma_n$$

is chosen as the default setting, where σ_n denotes the total noise, which is stable over time and uniform across all channels on the same BPM plane, as observed in Section 5.2.

Fig. 5.18 shows the simulated interlock rate per day as a function of the correlation coefficient ρ for at this setting with SNR = 10. For each data point, 10^8 noise frames are generated to calculate the false positive rate $p_{\text{false\_positive}}$, and another 10^8 signal frames at SNR 10 are generated to determine the efficiency ϵ. These two quantities are then used to compute the interlock rate $p_{\text{interlock}}$ using Eq. 5.23, which is scaled to a 24-hour period with a 10 kHz event rate.

For $\rho = 0.30$, the interlock rate is 0.04 per day. This means that the detector is free of interlock as long as the SNR is larger than 10, and the common mode subtraction is applied for the board 1-type detector plane.

For implementation of the clustering algorithm on FPGA, both T_c and N_c are registers that are controlled and can be modified by the HITDAQ.

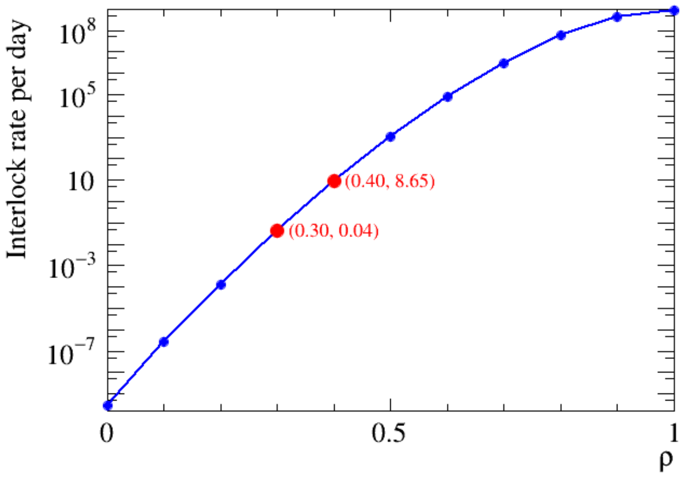


FIGURE 5.18: Interlock rate per day versus the correlation coefficient ρ, with $T_c = 3\sigma_n$ and $N_c = 4$, for SNR = 10.

The current clustering algorithm does not consider the channels between the two edge clusters, to ignore dead channels. It was originally developed at the beginning of this project, when several dead channels were intentionally left in place so that their impact could be studied. However, recent discussions with Dr. Jakob Naumann and Dr. Thomas Hansmann at HIT clarified that dead channels should be avoided or minimized before clinical use. Since 2024, after all photodiode arrays were replaced with new ones, only one dead channel remains. From the perspective of clustering, dead channels should no longer be a concern. Therefore, the algorithm can be improved in the future by adopting a box search method that requires the peak amplitudes to lie within the window of interest. This approach eliminates false clusters outside the true signal region and thus enables a more accurate determination of the beam profile edges.

5.5 Calibration

Calibration is necessary to ensure a uniform response across all channels for a flat particle distribution. The photodiode array consists of 320 channels, each coupled to several fibers. However, the number of fibers coupled to each channel can vary, as shown in Fig. 4.2, leading to differences in signal response between channels. Additionally, variations of light yield per fiber, reflectivity, the variation of the alignment of fibers and coupling between fibers and photodiodes are also accounted for in the per-channel calibration.

Calibration is necessary to ensure a uniform response across all channels of a BPM plane for a flat particle distribution. One BPM plane consists of 320 channels, each coupled to several fibers. However, the number of fibers coupled to each channel can vary, as shown in Fig. 4.2, leading to differences in signal response between channels. In addition, variations in light yield per fiber, reflectivity, fiber alignment, and the coupling between fibers and photodiodes are also accounted for the per-channel response variation.

where $\hat{y}_i$ is the calibrated signal and f_i is the calibration factor for channel i. The calibration factor f_i is determined in section 7.2. The value is around 1 with up to 35% variations between channels.

After pedestal subtraction, common-mode subtraction, and clustering, the signal inside the window of interest is multiplied by a calibration factor:

$$\hat{y}_i = f_i \cdot y_i \tag{5.29}$$

where $\hat{y}_i$ is the calibrated signal and f_i is the calibration factor for channel i. The calibration factor f_i is determined in Section 7.2 and has a value around 1, with variations of up to 35% between channels.

The clustering algorithm is applied prior to calibration, since calibration multiplies the noise σ_n, resulting in different calibrated noise levels for each channel. Consequently, different thresholds would be required for clustering after calibration, which would be much more complicated than applying the clustering algorithm before calibration with the same threshold for all the channels.

5.6 Gaussian-like Beam Profile Reconstruction

The goal of the reconstruction is to estimate the parameters $\hat{\mu}$, $\hat{\sigma}$, and $\hat{A}$ from the measured Gaussian-like beam profile $(x_n, \hat{y}_n)$. Here, $\hat{y}_n$ is the calibrated amplitude, and x_n is the position of channel n. The relative position of channel n on array m within a single plane is defined as:

$$x_n = (0.8n + 0.2m)\ \text{mm} \tag{5.30}$$

where 0.8 mm is the channel pitch, and 0.2 mm accounts for the gap between adjacent arrays.

The reconstruction can be performed using three methods: root mean square (RMS), linear regression, and Gaussian fitting. The RMS method is introduced as the simplest approach, as it is suitable for implementation on FPGAs. The linear regression method is included because it is expected to offer better performance than RMS and remains feasible for FPGA implementation, although it requires more FPGA resources and development effort. The Gaussian fitting method is presented as the most accurate approach; however, it is not feasible for FPGA implementation and is therefore used as an accurate reference for offline analysis.

This section introduces these methods and evaluates their performance.

5.6.1 Profile Estimation

RMS

In the RMS method, the first and second moments are used as estimators for $\hat{\mu}$ and $\hat{\sigma}$, respectively, and the amplitude $\hat{A}$ is estimated by the maximum value $\hat{y}_{\max}$ of the signal:

$$\begin{aligned} \hat{\mu} &= \frac{\sum x_n \hat{y}_n}{\sum \hat{y}_n} \\ \hat{\sigma} &= \sqrt{\frac{\sum (x_n - \hat{\mu})^2 \hat{y}_n}{\sum \hat{y}_n - 1}} \\ \hat{A} &= \hat{y}_{\max} \end{aligned} \tag{5.31}$$

The arithmetic operations required for the RMS method include addition, multiplication, division, and square root. All of these operations are supported by the

MAX10 FPGA, and the RMS method has been successfully implemented on the FPGA, as will be shown in Chapter 6.

Linear Regression

Taking the natural logarithm of the Gaussian equation 5.25, one obtains:

$$\ln y = \ln A - \frac{(x-\mu)^2}{2\sigma^2} = \ln A - \frac{x^2}{2\sigma^2} + \frac{x\mu}{\sigma^2} - \frac{\mu^2}{2\sigma^2} = a + bx + cx^2 \tag{5.32}$$

where

$$a = \ln A - \frac{\mu^2}{2\sigma^2}, \quad b = \frac{\mu}{\sigma^2}, \quad c = -\frac{1}{2\sigma^2}$$

This transformation results in a polynomial equation. The parameters a, b, and c can be estimated using the linear regression method described below.

The principle of linear regression is to minimize the sum of squared residuals. The residual ϵ is defined as:

$$\epsilon = \ln \hat{y} - \ln y = \ln \hat{y} - (a + bx + cx^2) \tag{5.33}$$

To minimize the sum of squared residuals, $\sum \epsilon^2$, one sets the partial derivatives of $\sum \epsilon^2$ with respect to a, b, and c equal to zero. This leads to the following 3 × 3 matrix equation:

$$\begin{bmatrix} N & \sum x_n & \sum x_n^2 \\ \sum x_n & \sum x_n^2 & \sum x_n^3 \\ \sum x_n^2 & \sum x_n^3 & \sum x_n^4 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \sum \ln(\hat{y}_n) \\ \sum x_n \ln(\hat{y}_n) \\ \sum x_n^2 \ln(\hat{y}_n) \end{bmatrix} \tag{5.34}$$

By solving the matrix equation 5.34, the coefficients a, b, and c can be determined. Subsequently, the parameters μ, σ, and A can be computed using the following expressions:

$$\hat{\mu} = -\frac{b}{2c}, \quad \hat{\sigma} = \sqrt{\frac{-1}{2c}}, \quad \hat{A} = e^{a - \frac{b^2}{4c}} \tag{5.35}$$

A simplified version of the linear regression method is proposed in [64]. The basic idea is to use the area under the curve to estimate σ, and then estimate the remaining parameters μ and A using a linear regression for the remaining two parameters.

The area under the curve is calculated as:

$$\Lambda = \sum \hat{y}_n \Delta x_n \tag{5.36}$$

where Δx_n is the channel width. The area under a Gaussian curve is given by:

$$\Lambda = \int_{-\infty}^{+\infty} A e^{-\frac{(x-\mu)^2}{2\sigma^2}} \, dx = \sqrt{2\pi} A \sigma \tag{5.37}$$

Thus, the standard deviation σ can be estimated as:

$$\hat{\sigma} = \frac{\sum \hat{y}_n \Delta x_n}{\sqrt{2\pi} \hat{y}_{\max}} \tag{5.38}$$

With the parameter c known, the 3×3 matrix equation 5.34 can be simplified to a 2×2 matrix equation:

$$\begin{bmatrix} \sum \hat{y}_n^2 & \sum x_n \hat{y}_n^2 \\ \sum x_n \hat{y}_n^2 & \sum x_n^2 \hat{y}_n^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum \hat{y}_n^2 \ln(\hat{y}_n) - c \sum x_n^2 \hat{y}_n^2 \\ \sum x_n \hat{y}_n^2 \ln(\hat{y}_n) - c \sum x_n^3 \hat{y}_n^2 \end{bmatrix} \tag{5.39}$$

After solving a and b from this matrix equation, the amplitude $\hat{A}$ and the mean $\hat{\mu}$ can be calculated using Equation 5.35.

Both the 3×3 and the simplified 2×2 matrix linear regression methods require the logarithmic operation on the signal, $\ln \hat{y}_n$, in addition to the arithmetic operations used in the RMS method. The 2×2 matrix linear regression method requires less memory and computational resources than the 3×3 method, as solving a 2×2 matrix is less computationally intensive. To implement linear regression on the FPGA, the first milestone is the 2×2 matrix method, with the 3×3 method considered for later implementation.

In this section, the simplified 2×2 matrix linear regression method is evaluated as the next generation of the reconstruction algorithm following the RMS method.

Fitting

In this work, the chi-square method fitting tool of the ROOT framework [65] is used to fit the beam profile using Gaussian curve.

Unlike RMS and linear regression, fitting involves an iterative process. It begins with an initial guess for the parameters and assesses the goodness of fit. Subsequently, it adjusts these parameters iteratively to improve the goodness of fit. While fitting is a more computational expensive method, it tends to yield more accurate results.

5.6.2 Performance

Toy Monte Carlo simulations were conducted to estimate the initial performance of three methods: RMS, linear regression, and Gaussian fitting. A beam with a fixed FWHM of 5 mm was generated under varying SNR conditions and different levels of common mode noise ρ. The integer SNR was varied from 0 to 100 in steps of 1, and the beam position μ was randomly assigned for each frame. For each SNR value, 10^5 frames were generated for a given ρ value. Two sets of simulations were performed, corresponding to $\rho = 0.04$ and $\rho = 0.30$. The clustering parameters were set to $T_c = 3\sigma_n$ and $N_c = 4$ channels. The three reconstruction methods were then applied to estimate μ, σ, and A within the window of interest identified by the clustering algorithm.

Fig. 5.19 presents an example of a simulated beam profile and the corresponding reconstruction results obtained from the three methods. Only the channels within the window of interest indicated by the dashed lines are used for the reconstruction. Performance is assessed in terms of the uncertainty and bias of the estimated μ and FWHM, by comparing the results with the known simulation parameters.

Focus Bias Due to Cut-Off Tails

As shown in Fig. 5.19, the tails of the beam profile are not included in the window of interest. The RMS method estimates σ as the second moment of the amplitudes inside the window. Since the tails are excluded, the RMS method underestimates σ. Similarly, in the 2×2 linear regression method, σ is estimated using the ratio of

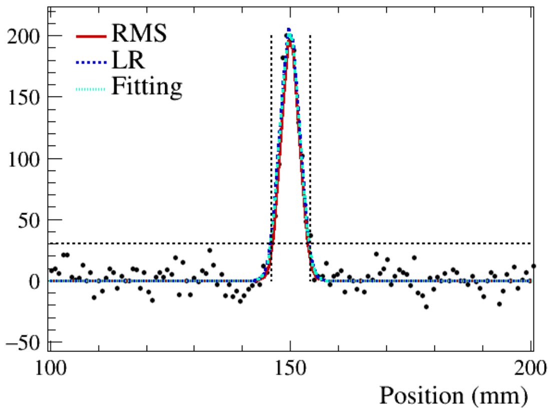


FIGURE 5.19: A simulated beam profile at SNR = 20 with a FWHM of 5 mm.

the area under the curve to the peak value, as given in Eq. 5.38. This area is also reduced due to the missing tails. As a result, both RMS and the 2 × 2 linear regression methods underestimate the beam width/focus. The 3 × 3 linear regression method is expected to perform better, as it includes more information into the least-squares regression.

Fig. 5.20 presents the 2D histograms of focus bias versus the window-of-interest length for all three methods. The focus bias is defined as the difference between the estimated and true focus. For a fixed beam width, the width bias of the RMS and linear regression methods decreases as the number of channels in the window of interest increases, while the fitting method does not introduce a systematic bias.

Improved Resolution with Increasing SNR

Fig. 5.21 presents 2D histograms of the position and focus biases for the RMS method with $\rho = 0.04$ and $\rho = 0.30$. As the SNR increases, the distributions of both position and focus biases become narrower, indicating improved resolution. The strip-like structure observed in the focus bias plot is caused by the quantization effect due to the discrete channel positions.

The 2D histograms for the linear regression and fitting methods exhibit the same trend, with the distributions becoming narrower as the SNR increases.

Impact of Common-Mode Noise on Performance

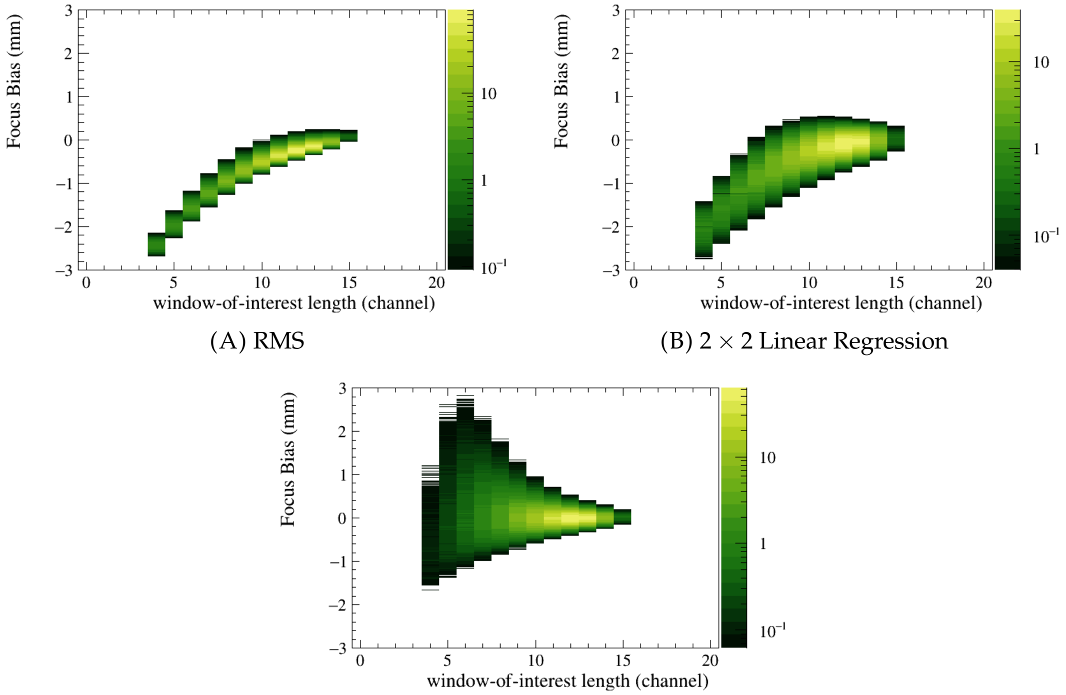
Comparing the bias plots for $\rho = 0.04$ and $\rho = 0.30$ in Fig. 5.21 reveals that common-mode noise increases the biases in both position and focus, with a more pronounced deterioration in focus.

5.6.3 Comparison of the Three Methods

The reduced chi-squared statistic, χ^2_ν, is used to evaluate how well the reconstruction results describe the observed data. It is defined as:

$$\chi^2_\nu = \frac{1}{\nu}\sum\frac{(y-\hat{y})^2}{\sigma^2} \tag{5.40}$$

where y is the estimated value based on the reconstructed results, $\hat{y}$ is the measured value, and $\sigma^2 = |\hat{y}| + \sigma^2_\eta$ denotes the error variance as defined in Eq. 5.18. The

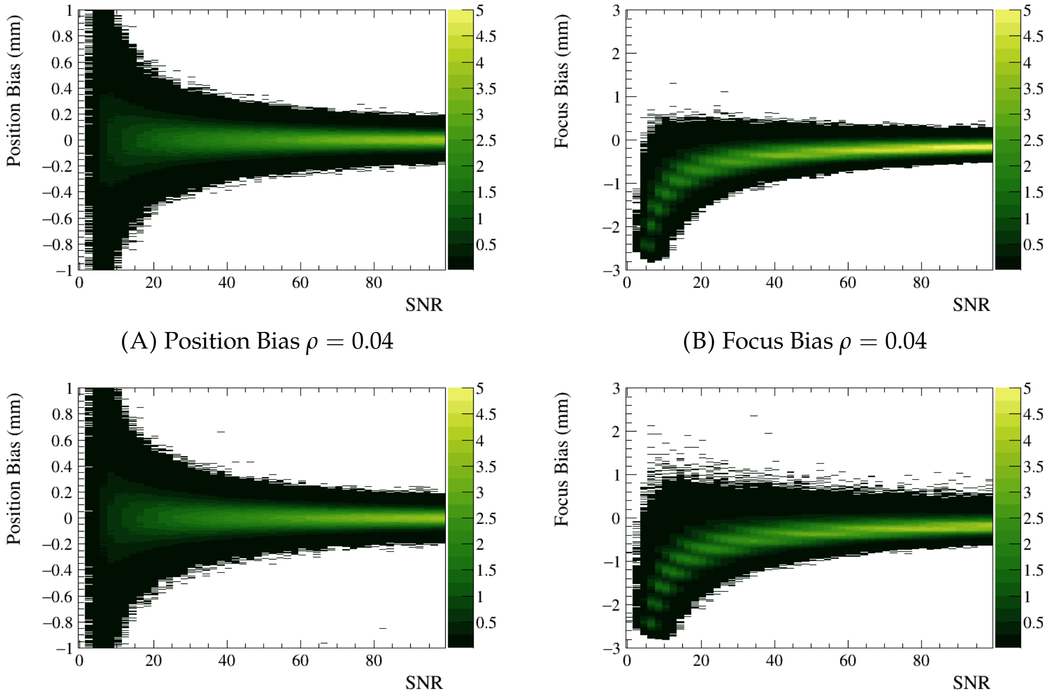


(A) RMS

(B) 2 × 2 Linear Regression

(C) Fitting

FIGURE 5.20: Two-dimensional histograms of focus bias versus window-of-interest length for the three methods, based on simulations with $\rho = 0.04$. The beam FWHM is fixed at 5 mm. The Z-axis color scale represents the number of entries, in units of 10^3.



(A) Position Bias $\rho = 0.04$

(B) Focus Bias $\rho = 0.04$

(C) Position Bias $\rho = 0.30$

(D) Focus Bias $\rho = 0.30$

FIGURE 5.21: Two-dimensional histograms of position and focus biases from the RMS reconstruction results, based on simulations with (A) and (B) $\rho = 0.04$, and (C) and (D) $\rho = 0.30$. The Z-axis color scale represents the number of entries in units of 10^3.

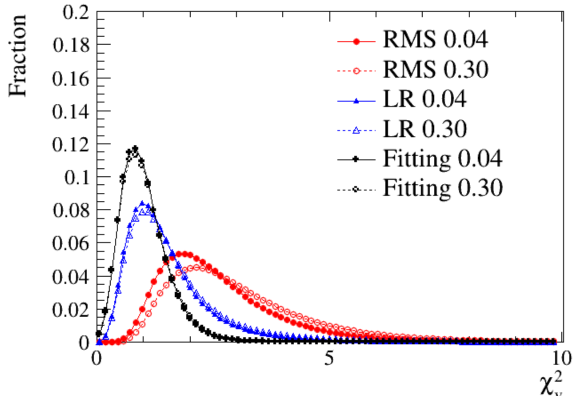


FIGURE 5.22: Normalized histograms of the reduced chi-squared χ^2_ν for reconstructions using RMS, linear regression (LR), and Gaussian fitting, based on simulations with $\rho = 0.04$ and $\rho = 0.30$.

number of degrees of freedom is given by $\nu =$ number of bins $- 3$. The closer χ^2_ν is to 1, the better the reconstruction describes the data. A value significantly larger than 1 indicates that the reconstruction does not fit the data well, while a value significantly smaller than 1 suggests overfitting or underestimation of errors.

Fig. 5.22 presents the reduced chi-squared χ^2_ν distributions for the three reconstruction methods based on the simulations in subsection 5.6.2. As expected, Gaussian fitting yields the most accurate results, followed by linear regression and then RMS. The presence of common mode noise (ρ) degrades the reconstruction accuracy, with RMS being the most affected.

The reduced chi-squared statistic evaluates the overall goodness for the reconstructed beam profile parameters μ, σ, and A.

However, since the primary goal of this thesis is to measure position and focus in real time, reconstructing the amplitude A is not required. Therefore, the specific metrics of the resolution and mean bias for position and focus are used to compare the three methods, as shown in Fig. 5.23.

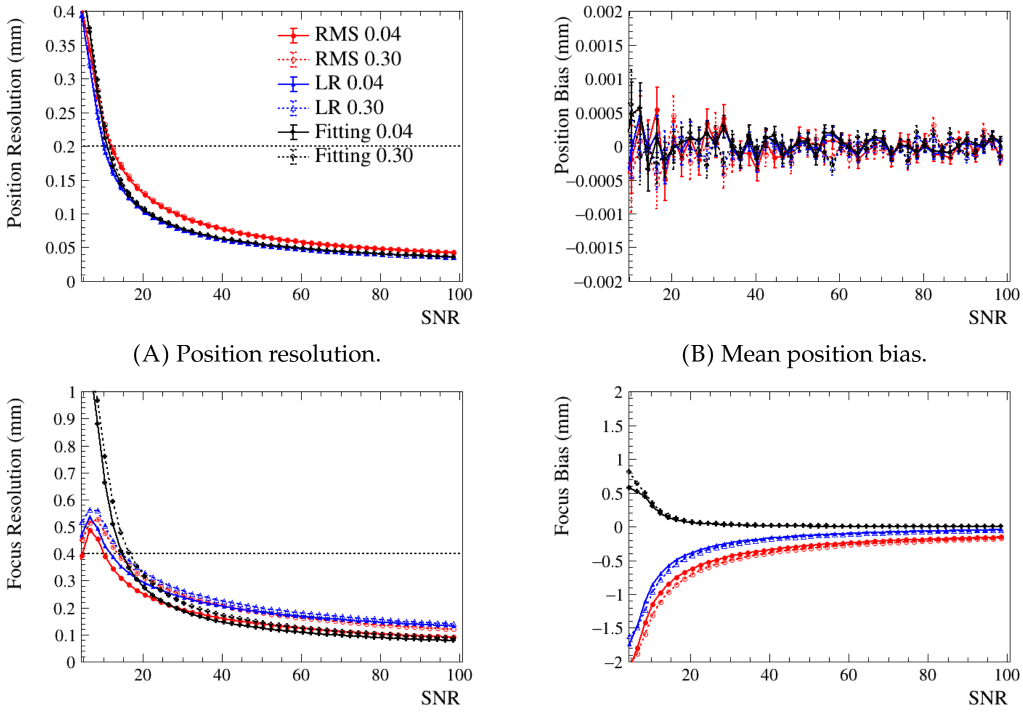
The resolution in Fig. 5.23 is determined as the standard deviation of the bias distribution, as in Fig. 5.21, with its outliers removed. The mean bias is defined as the mean of the bias distribution, also with its outliers removed. Outliers are excluded by imposing a threshold on the bin entry count.

Fig. 5.23 shows the following characteristics:

- Both the resolution and bias improve with increasing SNR.
- None of the three methods introduces a systematic position bias.
- The common-mode noise has little impact on the position resolution, mean position bias, and mean focus bias, but it greatly degrades the focus resolution in the RMS method.
- The linear regression method provides better position resolution than the RMS method.
- The RMS method achieves better focus resolution than the linear regression method at $\rho = 0.04$, but with greater systematic bias.

From the plots, one can estimate the following two requirements:

- For a 5 mm-wide beam, achieving a position resolution better than 0.2 mm requires an SNR greater than 15.



(A) Position resolution.

(B) Mean position bias.

(C) Focus resolution.

(D) Mean focus bias.

FIGURE 5.23: Resolution and bias of the position and focus with SNR in the range $[5, 100)$ for $\rho = 0.04$ and $\rho = 0.30$. Results are shown for RMS, linear regression (LR), and Gaussian fitting methods. The beam FWHM is fixed at 5 mm. The horizontal dashed lines represent the requirements for position resolution better than 0.2 mm and focus resolution better than 0.4 mm.

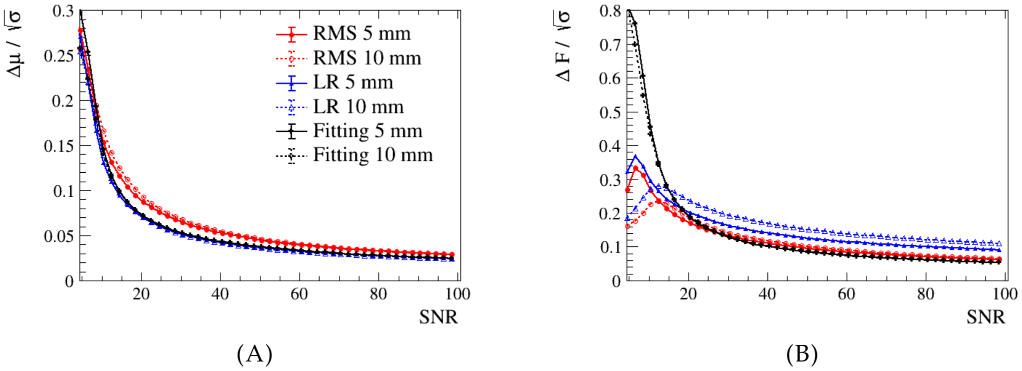


FIGURE 5.24: (A) $\Delta\mu/\sqrt{\sigma}$ and (B) $\Delta F/\sqrt{\sigma}$ as a function of SNR in the range $[5, 100)$ from simulations of beams with FWHM values of 5 mm and 10 mm at $\rho = 0.04$.

- For a 5 mm-wide beam, achieving a focus resolution better than 0.4 mm requires an SNR greater than 10 for $\rho = 0.04$ and 15 for $\rho = 0.30$.

For wider beam, the resolution scales with $\sqrt{\sigma}$. The resolution of the mean value from a set of N independent measurements, $\Delta\mu$, is given in Eq. 40.5 of [8]:

$$\Delta\mu = \frac{\sigma}{\sqrt{N}}, \tag{5.41}$$

where σ is the beam width and $N = \sum \hat{y}_n$, i.e., the area under the curve. Eq. 5.41 holds when the noise is negligible.

It can be further simplified using $N = \sqrt{2\pi}A\sigma$ from Eq. 5.38:

$$\Delta\mu = k\frac{\sqrt{\sigma}}{\sqrt{A}}, \tag{5.42}$$

with k as a constant. Since the SNR is proportional to A (see Eq. 5.27), the resolution $\Delta\mu$ scales with $\sqrt{\sigma}$ at a fixed SNR.

Fig. 5.24 shows $\Delta\mu/\sqrt{\sigma}$ and $\Delta F/\sqrt{\sigma}$ (where ΔF denotes the focus resolution) as functions of SNR for beams with widths of 10 mm and 5 mm. The ratio $\Delta\mu/\sqrt{\sigma}$ for all three methods follows the expected $\sqrt{\sigma}$ scaling. The ratio $\Delta F/\sqrt{\sigma}$ for the RMS and fitting methods also follows this rule.

This indicates that, to achieve the same resolution for a wider beam, the SNR must scale linearly with the beam width.

For the linear regression method, the beam width is calculated using Eq. 5.38, i.e., as the ratio between the area and the peak. As the beam width σ increases, the uncertainty of the area also increases due to more channels contributing noise. Consequently, ΔF for the linear regression method scales with σ rather than with $\sqrt{\sigma}$.

5.7 Summary

This chapter has described the signal characteristics and the reconstruction algorithm for the BPM. Fig. 5.25 shows the reconstruction chain.

The voltage differences between ADCs and photodiodes generate a pedestal. This pedestal can be subtracted accurately with an uncertainty better than 1 ADC count using the average of 8192 noise data frames.

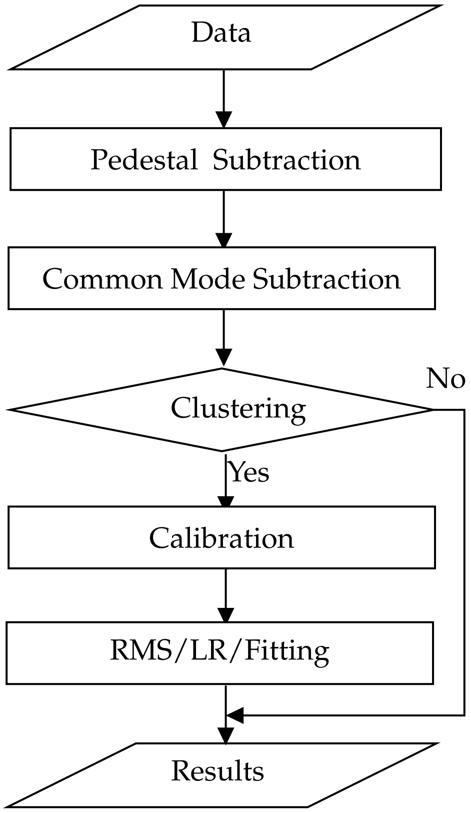


FIGURE 5.25: The reconstruction algorithm for one frame of data. Common mode subtraction should be applied only for the Board 1-type detector plane.

The Board 1-type detector plane exhibits significant common mode noise with a correlation factor $\rho = 0.81$, which should be further investigated in the lab. It can be reduced to $\rho = 0.30$ by subtracting the average of 4 reference channels. The board 0-type detector plane has a much lower common mode noise, with $\rho = 0.04$, and should not be subjected to common mode subtraction, as it would degrade the performance.

Due to the different levels of the common mode noise, the subsequent reconstruction algorithms—clustering and the three reconstruction methods (RMS, linear regression, and Gaussian fitting)—are studied under various noise conditions.

The clustering algorithm locates the window of interest with two flexible parameters: the threshold T_c and the cluster size N_c. The default values are $T_c = 3\sigma_n$ and $N_c = 4$ channels. With these settings, for SNR larger than 10 the expected interlock rate for $\rho = 0.30$ is below 0.04 per day, and for $\rho = 0.04$ it is below 10^{-7} per day.

Once the window of interest are located, the reconstruction algorithm is applied to the calibrated data within the window. Three methods are evaluated by toy simulations: RMS, linear regression, and Gaussian fitting. In the case of high common-mode noise event, both position and focus(width) reconstruction are influenced. Statistically, the dominant effect is the degradation of the focus resolution in the RMS method, while the impact on position resolution and mean bias remains minimal.

The estimated BPM performance requirements from the toy simulations are as follows: for a beam with a FWHM of 5 mm, an SNR greater than 15 is required to achieve a position resolution better than 0.2 mm. For focus resolution better than 0.4 mm, an SNR greater than 10 is required for a board with $\rho = 0.04$, and greater than 15 for a board with $\rho = 0.30$. For wider beams, the required SNR scales linearly with the beam width. It is important to note that SNR is defined as the ratio of signal amplitude to uncorrelated noise and is equally applicable to both board 0 and board 1 types.

In Section 5.3, for the pull calculation, $\sqrt{y}$ represents the statistical part of the error, where y is the calibrated amplitude. This results in a pull distribution from a beam scan with a standard deviation of 0.97, confirming the correct estimation of the error. However, $\sqrt{y}$ accounts not only for the intrinsic Poisson fluctuations of the ADC amplitude but also for contributions from the calibration process over the scan region.

The toy simulation is generated with a statistical fluctuation of $\sqrt{y}$. The estimated resolution obtained from this toy simulation therefore reflects the overall resolution across the scan region. The resolution at a fixed position would be better than the estimation from the toy simulation, as fewer variations are involved.

In the next chapter, this reconstruction chain—based on the RMS method—will be implemented on the Max10 FPGA.

For all detectors, performance or resolution is determined by the signal amplitude and the noise level. To achieve better performance, either the signal must be increased or the noise reduced. In this case, there are two types of noise: uncorrelated noise and common mode noise shared among all channels. And they deteriorate the performance in different ways. The presence of common mode noise can lead to extreme cases in which reconstruction failures and thus interlocks occur, while the uncorrelated noise primarily affects the resolution of the position and focus measurements.

When the common mode noise is below $\rho = 0.30$, and the SNR exceeds 10, the likelihood of erroneous interlocks is low. If this SNR threshold cannot be reliably maintained, efforts should be made to reduce the common mode noise. The source of the common mode noise may be related to the voltage regulator on the ADC board. The board 1-type plane should be investigated in the lab, with the goal of achieving a common mode noise level comparable to that of the board 0-type plane.

With the common mode noise below $\rho = 0.30$, the resolution is primarily determined by the signal amplitude and the uncorrelated noise. A strict requirement is that an SNR greater than 15 is needed to achieve a position resolution better than 0.2 mm for a beam with a full width at half maximum (FWHM) of 5 mm. If the SNR is insufficient, efforts must be directed toward either increasing the signal amplitude or reducing the uncorrelated noise. The signal amplitude can be improved by increasing the integration time or by enhancing the light yield of the detector, for example, through better coupling between the fibers and the photodiodes. The uncorrelated noise can be reduced by adding a cooling system.

In Chapter 7, where different beams at HIT were measured, the typical SNR for most beams is well above 10 at a measurement rate of 10 kHz. At this level, the interlock rate is not a concern, but the resolution needs to be considered at low intensities.

Chapter 6

Implementation of the Beam Profile Reconstruction on a FPGA

For online beam position monitoring, the SciFi-BPM detector requires real-time reconstruction. To ensure compatibility with the HIT BAM system, the reconstruction must be performed entirely in hardware, and in this case, the Intel MAX10 FPGA [58] is used.

Chapter 5 explains the reconstruction algorithm for the detector, which is intended to be implemented on the FPGA for online data processing. The purpose of this chapter is to present the implementation and verification of the reconstruction algorithm on the FPGA. This constitutes the main focus of the thesis.

This chapter can be divided into two parts. The first part introduces the background knowledge necessary for understanding the design of the reconstruction firmware, while the second part describes the design, implementation, and verification of the firmware itself.

The first part consists of four sections. Section 6.1 provides a general overview of digital circuit design, introducing key concepts such as logic elements, pipelining, and finite state machines. Section 6.2 focuses on the available resources of the MAX10 FPGA, which serves as the hardware platform—or the canvas—for this design implementation. Section 6.3 presents the arithmetic operations on FPGAs, including addition, subtraction, multiplication, division, and square root, all of which are essential for implementing the reconstruction algorithm. Section 6.4 describes the firmware frame developed by Dr. Michał Dziewiecki [62], which serves as the original framework upon which the reconstruction firmware is built.

The second part also includes four sections. Section 6.5 presents the design concept of the reconstruction firmware, discussing how it is shaped by hardware resource constraints and based on the original design. Section 6.6 provides a detailed explanation of the implementation and timing behavior of the reconstruction firmware. Section 6.7 explains how the bit widths and LSBs of operands and results are determined, along with the precision of each operation. Finally, Section 6.8 verifies the functionality of the reconstruction firmware by comparing its output with the results from the C++ implementation, confirming that the firmware accurately realizes the reconstruction algorithm.

6.1 General Introduction to FPGA

A field-programmable gate array (FPGA) is a reconfigurable digital semiconductor device primarily composed of programmable logic blocks, programmable interconnections, and programmable I/Os. It can be configured as various digital circuits, enabling parallel computation and real-time data processing. FPGAs are versatile

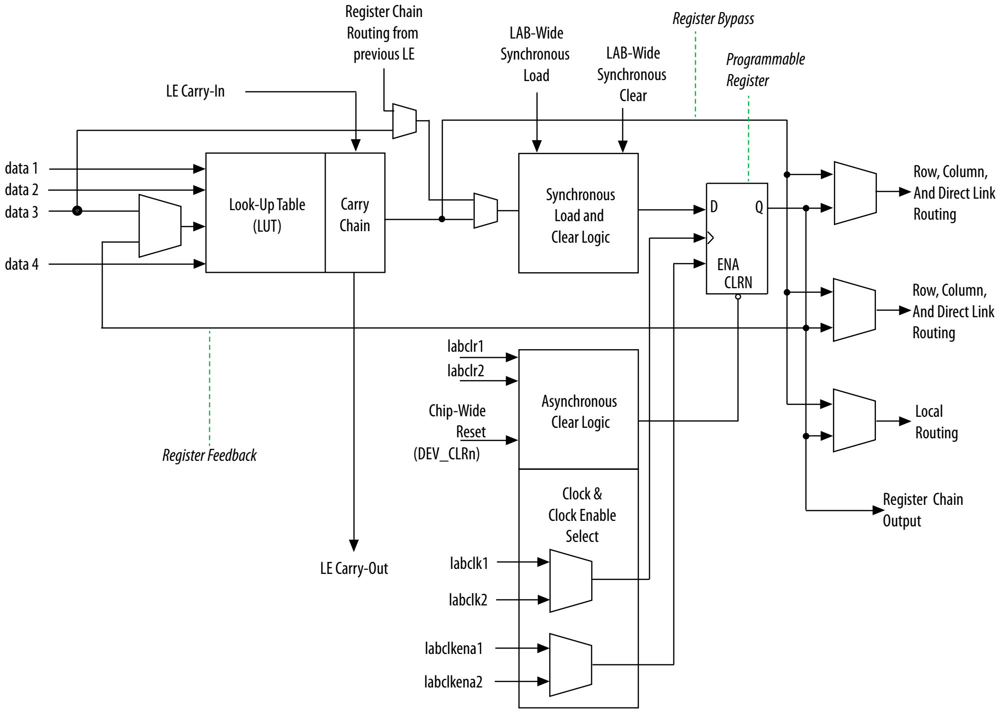

Register Chain Routing from previous LE
LE Carry-In
LAB-Wide Synchronous Load
LAB-Wide Synchronous Clear
Register Bypass
Programmable Register
data 1
data 2
data 3
data 4
Look-Up Table (LUT)
Carry Chain
Synchronous Load and Clear Logic
D
Q
ENA
CLRN
Row, Column, And Direct Link Routing
Row, Column, And Direct Link Routing
Local Routing
Register Chain Output
labclr1
labclr2
Chip-Wide Reset (DEV_CLRn)
Asynchronous Clear Logic
Register Feedback
Clock & Clock Enable Select
LE Carry-Out
labclk1
labclk2
labclkena1
labclkena2

FIGURE 6.1: High-level block diagram of a Logic Element (LE) in MAX10 FPGAs [66]. One LE consists of one look-up table (LUT), one flip-flop, and some interconnections.

and powerful tools in detector physics, which often requires real-time processing of large-scale data from detectors.

In this thesis, the MAX10 FPGA is programmed to control the photodiode arrays and ADCs, handle data collection and transfer, and perform real-time reconstruction, as described in Section 4.4. The real-time reconstruction is the major focus of this work. To understand the design concepts behind the reconstruction firmware, it is important to first understand the basic logic elements, the concept of pipelining, and the finite state machine.

Logic Element

A look-up table (LUT), a flip-flop, and interconnections together form a Logic Element (LE), which is the basic building block of an FPGA. The LUT can be configured to implement any combinational logic function of its inputs. The flip-flop, which updates its output on each clock edge, can store the LUT output and form sequential logic. The interconnections allow LUTs and flip-flops to be connected together, enabling the construction of more complex circuits using multiple LEs. Further explanation of how LUTs, flip-flops, and interconnections are constructed from CMOS transistors can be found in Appendix A.

Fig. 6.1 shows the LE architecture of MAX10 FPGAs. It consists of a 4-input LUT and a D flip-flop with enable and clear logic, along with carry-in, carry-out, and register-chain interconnections. The MAX10 FPGA used in this work, which is the largest in the MAX10 series, provides 49 760 user-available LEs.

Pipelining

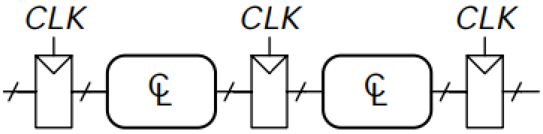


FIGURE 6.2: A typical digital circuit [67]. *CLK* stands for clock. The boxes with a *CLK* input represent flip-flops, and the rounded-corner boxes represent combinational logic. The slanted line on the horizontal signal line indicates that the signal has an arbitrary bit width.

Fig. 6.2 illustrates a typical digital circuit. It consists of interleaved combinational and sequential logic blocks. The combinational logic, which can be implemented entirely using LUTs, does not retain any memory of previous states. Its outputs change immediately in response to changes in inputs, subject to propagation delays. The sequential logic, represented by flip-flops, retains the output of the previous combinational stage until the next clock cycle and provides the input for the next stage. The clock signal synchronizes the sequential logic, ensuring that all flip-flops update their outputs simultaneously. This allows precise control over the timing of the final output signal.

When the combinational logic becomes too complex and the propagation delay exceeds acceptable limits—such that the flip-flop cannot capture the output of the combinational logic before the next clock edge—the output of the flip-flop becomes unpredictable. In such cases, the clock frequency must be reduced, or additional flip-flops must be inserted to store intermediate results and reduce the long propagation delay. The latter technique is known as *pipelining*. It is one of the main reasons why digital circuits typically consist of interleaved combinational and sequential logic blocks. As will be seen later, addition and multiplication can be completed within a single clock cycle, while division and square root operations require pipelining due to their greater complexity.

Finite State Machine

Feeding the output of a flip-flop back into the input of the combinational logic creates a different type of sequential logic known as a finite state machine (FSM). In this configuration, the outputs of both the flip-flop and the combinational logic determine the next state, which is stored in the flip-flop on the next clock edge. As a result, the flip-flop outputs change sequentially, depending on the current state and inputs. In the reconstruction firmware design, FSMs are extensively used to control data flow and manage operational sequences. Appendix A provides more details on FSMs.

6.2 Resources on the FPGA Board

The MAX10 FPGA development board is a commercially available platform that integrates one MAX10 FPGA alongside various peripherals and interfaces, making it suitable for a wide range of applications. For the SciFi-BPM detector, it is used both as the sensor interface and as the platform for implementing the reconstruction firmware.

This section provides an overview of the available resources and development tools for designing and implementing firmware on the MAX10 FPGA.

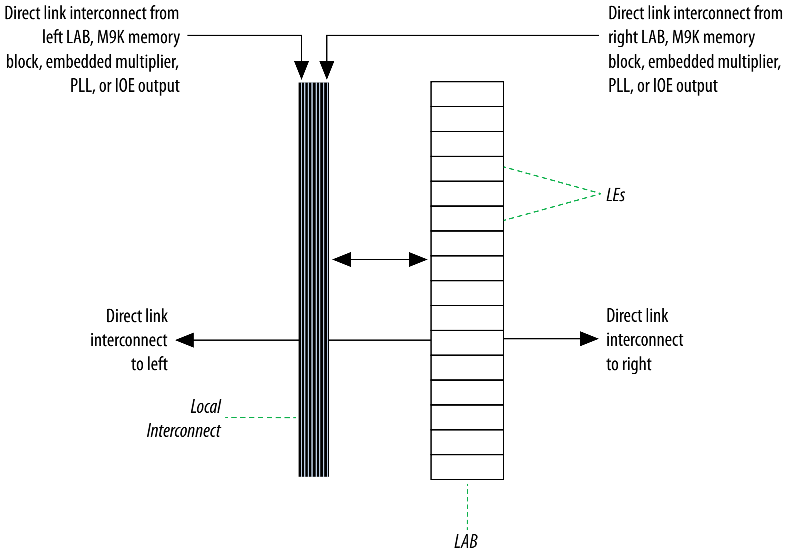

Direct link interconnect from left LAB, M9K memory block, embedded multiplier, PLL, or IOE output
Direct link interconnect from right LAB, M9K memory block, embedded multiplier, PLL, or IOE output
LEs
Direct link interconnect to left
Direct link interconnect to right
Local Interconnect
LAB

FIGURE 6.3: A Logic Array Block (LAB) and the links to its neighbors or other blocks [66]

6.2.1 Max10 FPGA

A group of LEs forms a Logic Array Block (LAB). In MAX10 FPGAs, each LAB consists of 16 logic elements (LEs), which can collectively form a 16-bit shift register. Fig. 6.3 shows a single LAB and its connections, illustrating how it can link to adjacent LABs or other blocks within the FPGA.

While LABs can be combined to implement complex logic, they are generally about an order of magnitude less efficient in terms of both speed and area than dedicated hard-wired implementations of the same functions [67]. Consequently, FPGAs often incorporate specialized embedded components such as memory, multipliers, and even entire microprocessors. Fig. 6.4 shows the embedded components within MAX10 FPGAs.

The MAX10 FPGA used in this thesis is the 10M50 [66], which contains 3 200 LABs. 90 of the 3 200 LABs are reserved and not available for user logic, resulting in 49 760 user-available LEs. In addition to the LABs, the MAX10 also includes the following embedded components:

- **Embedded multiplier blocks:** 144 18-bit× 18-bit multipliers capable of operating at 183 MHz. These are used for fixed-point multiplication.
- **Embedded memory blocks:** 182 M9K blocks, each 9 216 bits, 1 638 kbits in total. These resources can be configured as random-access memories (RAMs), read-only memories (ROMs), first-in-first-out buffers (FIFOs), or shift registers. In this work, they are extensively used to store pedestal values and calibration factors, buffer calibrated data for RMS calculations, and support FIFO-based data streaming.
- **Embedded clocks and phase-locked loops (PLLs):** Used to generate clocks for the sensor interface.
- **General-purpose and high-speed LVDS I/Os:** Used for interfacing with photodiode arrays.
- **External memory interface:** Provides connectivity to external DDR3 memory.

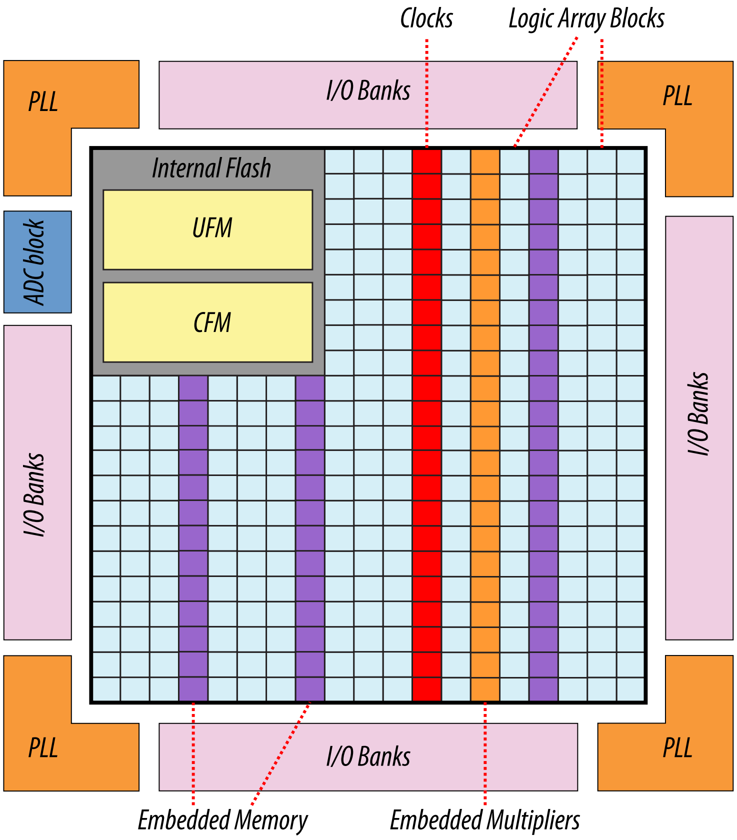


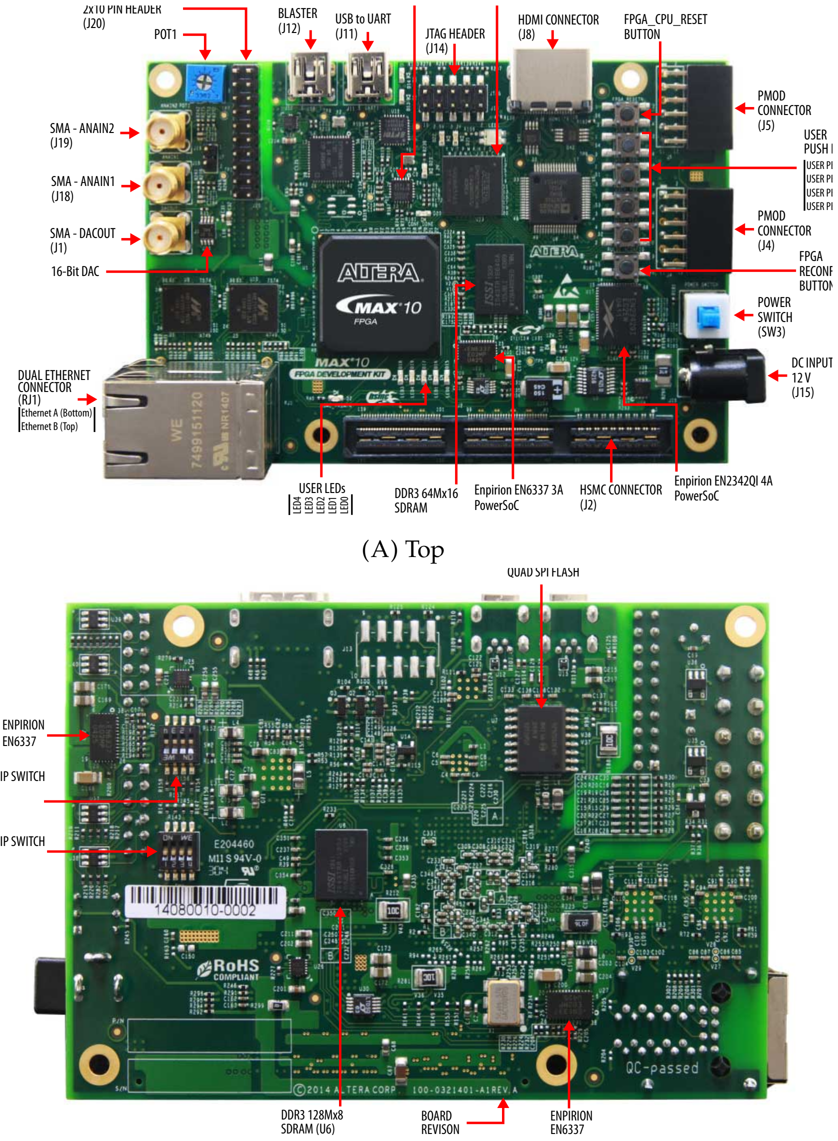
FIGURE 6.4: Embedded elements on MAX10 FPGAs [66]

- **Configuration flash memory (CFM):** Stores the FPGA configuration bitstream. Upon power-up, the image is loaded from the CFM, making the MAX10 a non-volatile FPGA.
- **Analog-to-digital converters (ADCs):** Not used in this thesis. It can be used in future for monitoring the temperature of the FPGA.
- **User flash memory (UFM):** Used for storing user data and initializing memory. Not used in this thesis.

Each SciFi-BPM plane, consisting of 320 photodiodes, is processed by a single FPGA. Each photodiode signal is digitized into 16-bit data. For one frame of a BPM plane, 320 16-bit ADC values must be processed by the FPGA. To collect and process this data, at least two sets of 320 16-bit buffer blocks are required on the FPGA, as will be shown in Section 6.4.2. These buffers can only be implemented using the flip-flops within the LABs, with one flip-flop required per bit. As a result, this part alone consumes 20% of the available flip-flops on the FPGA.

6.2.2 Intellectual Property Cores

In addition, there are many modular FPGA Intellectual Property (IP) cores available for the MAX10 FPGA, such as the NIOS II processor, Ethernet interface, on-chip memory, and fixed-point division and square root modules. These IP cores are similar to libraries in C++: they allow designers to implement complex functionality in firmware without writing the entire logic from scratch. Instead, only a few parameters—such as the bit width of input and output signals or the number of



(A) Top

(B) Bottom

FIGURE 6.5: The Max10 FPGA development board[58].

pipeline stages—need to be configured. They serve as convenient building blocks for constructing the complete firmware. However in order to use them effectively and correctly, it is important to understand the underlying principles of these IP cores. Section 6.3 explains the principles of arithmetic operations on FPGAs, which form the foundation of the reconstruction firmware.

6.2.3 MAX10 FPGA Development Board

As described in Chapter 4, the FPGA sits on a development board referred to the MAX10 FPGA Development Board from Altera[58]. Fig. 6.5 shows the front and back sides of the board. It features a MAX10 FPGA, an embedded USB-Blaster II, JTAG interface, DDR3 SDRAM, a 512 Mbit Quad Serial Peripheral Interface (Quad SPI) flash, two Gigabit Ethernet RJ-45 ports, one Universal High-Speed Mezzanine Card (HSMC) connector, a Silicon Labs clock generator with default output frequencies of 25 MHz, 50 MHz, 100 MHz, 125 MHz, etc.

As will be explained in Fig. 6.7, the FPGA together with some components on the development board forms a server that communicates with the data acquisition software HITDAQ via Ethernet.

6.3 Arithmetic Operations on FPGA

To implement the reconstruction algorithm described in Chapter 5 on FPGAs, five basic arithmetic operations are required: addition, subtraction, multiplication, division, and square root. While IP cores are available for these operations, the user must configure the number of pipeline stages together with the number format. Therefore, it is essential to understand how these operations are implemented on FPGAs.

This section describes different number representations, while briefly introducing the basic arithmetic operations along the way. The key concepts to take away from this section are:

- The operations are implemented using fixed-point arithmetic, meaning that both the bit widths and the weight of the least significant bit (LSB) of the operands and results must be fixed.
- Addition and multiplication can be completed within a single clock cycle.
- Division and square root are iterative operations that require multiple clock cycles to complete and should be pipelined.

In the digital world, numbers are represented using combinations of 0s and 1s. While there is considerable flexibility in how these combinations are interpreted, the most common method is binary representation. In binary, each bit has a base of 2, and its weight is a power of 2 greater than that of the bit to its right. Depending on the format, the representation can be signed or unsigned, and either fixed-point or floating-point.

Unsigned Integers

For unsigned integer numbers, each bit represents a power of two, where the weight of bit position N is 2^N, with $N = 0$ for the rightmost bit. For example, in the 3-bit binary number 101_2:

$$101_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 5$$

Addition of two unsigned integers is performed by summing the corresponding bits from right to left, carrying over any overflow to the next higher bit. For example, adding 101_2 and 011_2:

$$101_2 + 011_2 = 1000_2$$

Multiplication of two unsigned integers is carried out by multiplying each bit of one number by each bit of the other, shifting the results according to their bit positions, and then summing all intermediate results.

Signed Integers

For subtraction and division, signed integers in two's complement representation must be introduced. In two's complement, the leftmost bit (most significant bit, MSB) has a weight of -2^N instead of 2^N. To convert a number to its two's complement negative form, invert all bits and add 1 to the least significant bit. For example:

$$0101_2 = 5_{10} \quad \text{and} \quad 1011_2 = -5_{10}$$

The addition of two signed integers is performed in the same way as for unsigned integers, but with a slight difference in handling overflow. For signed integer addition, overflow occurs only if the carry into the sign bit differs from the carry out of the sign bit. This situation arises when adding two positive numbers results in a negative number, or when adding two negative numbers results in a positive number. If there is no overflow, the carry bit is simply ignored. For example, adding 5 and -5:

$$0101_2 + 1011_2 = 0000_2$$

The carry into and out of the sign bit are both 1, and the carry out is ignored. The result is 0, which is correct.

Subtraction of two signed or unsigned integers is thus performed by adding the two's complement of the subtrahend to the minuend; thus, subtraction is equivalent to addition.

Multiplication of the signed integers are performed similarly to those of unsigned integers by converting the operands to two's complement form and carefully handling the sign bit.

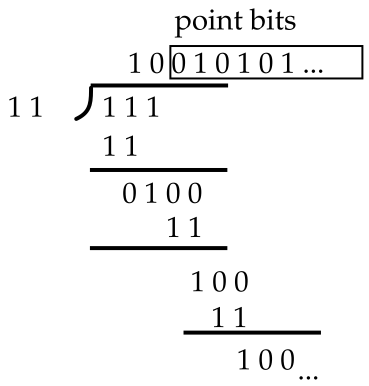


FIGURE 6.6: Division of 7 by 3.

The division is performed as an iterative process. Fig. 6.6 illustrates the binary division of 7 by 3. The procedure begins by aligning the most significant bits of the dividend with the divisor. The divisor is then compared with this portion of the dividend by subtracting it. If the remainder is non-negative, the corresponding quotient bit is set to 1; otherwise, it is set to 0 and the remainder is restored to its previous value.

At each step, the remainder is updated, and the divisor is compared with the new remainder to determine the next quotient bit. This process continues bit by bit until all quotient bits have been obtained. Since the quotient in binary fractional form may have unlimited precision, the iterative process for dividing 7 by 3 can, in principle, continue indefinitely.

In division, each bit of the quotient depends on the previously determined bits, since the remainder from the previous step is required. Division is therefore an iterative operation that requires multiple sequential additions.

The square root operation, similar to division, is also performed iteratively using sequential subtractions [68]. The most significant bit of the result is determined first, followed by the iterative determination of the remaining bits by comparing with the remainder.

Fixed-Point Numbers

In Fig. 6.6, the quotient is 2 in the integer part, and the fractional part becomes increasingly precise as more bits are added after the binary point.

In the number 10.010101_2, the LSB has a weight of 2^{-6}. This representation corresponds to an 8-bit unsigned fixed-point number, yielding $10.010101_2 = 2.32815$ instead of the exact value 2.33333. This difference arises because the remainder is constrained to be non-negative, meaning that part of 7 is always left undivided by 3. In fixed-point calculations, the remainder can be set to either non-negative or non-positive. In either case, the difference is smaller than the weight of the LSB of the result, $2^{-6} = 0.015625$. If the remainder is constrained to be non-negative, the resulting quotient is consistently smaller than or equal to the true value, with an error bounded by 2^{-6}.

As with division, the remainder in the square root calculation can be set to non-negative or non-positive. In this work, for both division and square root operations, the remainder is set to be non-negative, so the result is always smaller than or equal to the true value, with an error bounded by the weight of the LSB.

Signed fixed-point numbers are similar to signed integers, except that the rightmost bit has a fractional weight. The definition of the LSB weight is customizable depending on the specific application, but it must be fixed for a given implementation. Later, for position reconstruction, the LSB is defined as 0.05 mm. Arithmetic operations on signed fixed-point numbers follow the same rules as those on signed integers, since in practical FPGA implementations they are stored as fixed-bit signed integers but interpreted differently by the user.

For 8-bit signed integers, the representable range is from -128 to 127. To represent numbers outside this range, additional bits are required. It is therefore essential to fix the bit width of operands and results and ensure that it is sufficiently large for each operation to avoid overflow or loss of precision.

Floating-Point Numbers

In addition to fixed-point numbers, floating-point numbers are also used in digital circuits. A floating-point number consists of one sign bit, several bits for the exponent, and several bits for the mantissa. In floating-point arithmetic, the mantissa and exponent are processed differently. For example, when adding two floating-point numbers, the exponents must first be aligned, and then the mantissas are added as if they were signed fixed-point numbers. For multiplication, the exponents are added, and the mantissas are multiplied. Division is similar, except the exponent of the divisor is subtracted from that of the dividend.

Floating-point arithmetic is thus significantly more complex than fixed-point arithmetic. The MAX10 FPGA does not natively support floating-point operations; that is, there is no ready-to-use floating-point IP core available for the MAX10 FPGA.

Discussion

The MAX10 FPGA does not natively support floating-point operations, while the floating-point operations are more complex than fixed-point. Therefore, the reconstruction firmware uses fixed-point arithmetic for all operations. The bit widths and the weight of the least significant bit of the operands and results are thus fixed, explained in Section 6.7.

The clock for the reconstruction firmware is 50 MHz, resulting in a clock period of 20 ns. To determine the pipeline depth required for each arithmetic operation, a rough estimation of their propagation delays is provided in the following.

For the addition of two 16-bit unsigned integers, the carry may need to propagate from the rightmost bit to the leftmost bit. This results in a worst-case delay through 16 LUTs, each with an approximate propagation delay of 0.1 ns, leading to a total delay of 1.6 ns. This delay is well below half of the 20 ns clock period, meaning that the addition can be completed within a single clock cycle, with sufficient time remaining for the flip-flop to store the result.

For multiplication of two 16-bit numbers, the operation involves up to 16 additions, which can be parallelized. As a result, the propagation delay of a fully combinational multiplier is comparable to that of an adder. Moreover, the MAX10 FPGA includes dedicated hard-wired multiplier blocks that can perform 18-bit×18-bit multiplication at up to 183 MHz.

For the division of a 16-bit number by another 16-bit number, the operation requires approximately 16 sequential additions, which cannot be parallelized, as each result depends on the outcome of the previous step. Therefore, the propagation delay is roughly 16 times that of a single addition.

Similarly, the square root operation also involves sequential processing, resulting in a much longer propagation delay than that of addition or multiplication.

The estimations above are rough, as there is no definitive source for the exact propagation delay through a single LUT—only approximate values are available. One could build a simple test circuit with a known number of LUTs and observe the emulation results to obtain a more accurate estimate. Additionally, the FPGA compiler may optimize the synthesized logic, potentially reducing the actual propagation delay. However, these estimates serve to highlight the fact that division and square root operations are an order of magnitude slower than addition and multiplication, and this order is directly related to the bit width of the operands.

Therefore, in this work, one clock cycle is allocated for addition and multiplication operations. For division and square root operations, $\frac{N}{2}$ clock cycles are allocated, where N is the bit width of the numerator in the case of division, and the bit width of the input in the case of the square root. This value might be conservative; further optimization is possible but not necessary in this context. In the end, no timing violations were found related to the reconstruction design, indicating that the propagation delays of the arithmetic operations are within the limits of the clock period.

6.4 Original Firmware

The design of the reconstruction algorithm on the MAX10 FPGA is based on the firmware frame developed by Dr. Michał Dziewiecki. While Chapter 4 focused on the signal interfacing with the photodiode arrays and ADCs of the firmware to explain the detector's operation, this section concentrates on its structure and timing of the data flow, which are essential for integrating the reconstruction algorithm into this firmware.

Section 6.4.1 provides an overview of the firmware. Section 6.4.2 describes the data handling approach within the sensor interface and the timing for streaming out the data, which is essential for understanding how the data will be streamed into the reconstruction firmware. Finally, Section 6.4.3 summarizes the resource usage of this firmware.

6.4.1 Structure

The MAX10 development board is configured into three functional blocks: the CPU, the sensor interface, and the output interface, as shown in Fig. 6.7A. The arrows in the schematic represent the data flow between these blocks. The CPU communicates with HITDAQ via the output interface and has access to several control and status registers within the sensor interface. The sensor interface collects data from the ADCs and transfers it to the output interface, which then forwards the data to HITDAQ.

Fig. 6.7B provides a more detailed view of each block.

The CPU block comprises the Nios II processor implemented on the FPGA, the DDR3 RAM and QSPI flash memory on the development board, and their respective interfaces on the FPGA. The external DDR3 serves as RAM for the processor, while the QSPI flash stores the software image for the Nios II. During the boot phase, this image is loaded from the flash into the DDR3. In addition, a USB port is available for communication with the CPU, which is used for MAC address configuration and debugging of the sensor interface.

The sensor interface will be described in detail in Section 6.4.2.

The output interface is made up of two components: a User Datagram Protocol (UDP) generator and an Ethernet block.

The UDP generator receives data packets from the sensor interface, attaches appropriate Ethernet and UDP headers, and forwards the formatted packets to the Ethernet block.

The Ethernet block includes both receiver (RX) and transmitter (TX) chains for Ethernet packets. On the RX side, a triple-speed Ethernet interface connects to an RX Direct Memory Access (DMA) module. The CPU continuously monitors this RX DMA to receive commands and data from the HITDAQ. On the TX side, the CPU can send data to Ethernet through a TX DMA module. A multiplexer selects between the TX DMA path and the UDP generator path, forwarding the selected data stream to the Ethernet interface, and ultimately to HITDAQ.

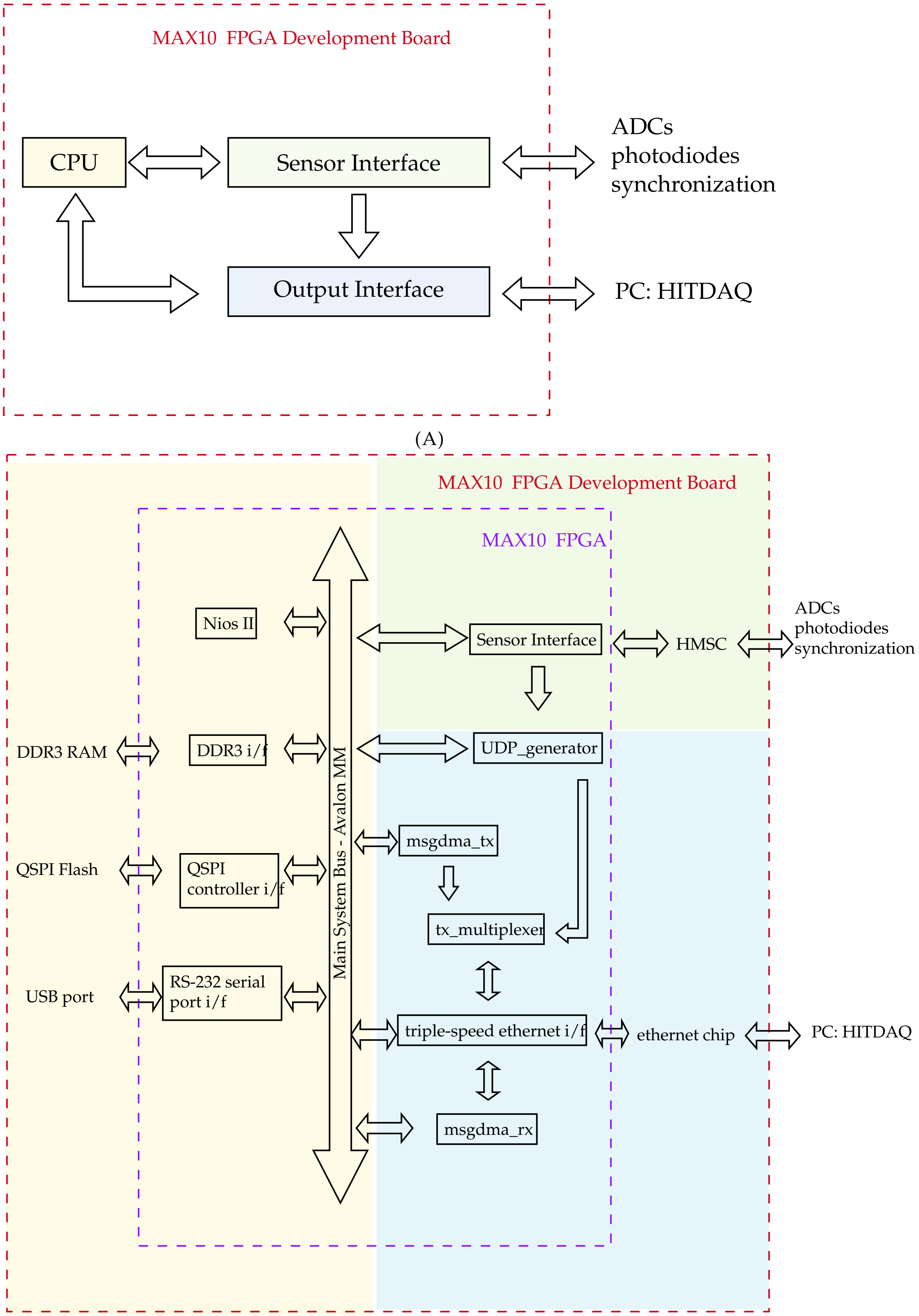
The reconstruction block will be inserted between the sensor interface and the output interface, as shown in Fig. 6.11.

This design is based on the Nios II Simple Socket Server (SSS) Ethernet design example provided by Altera for the Intel MAX10 FPGA [69]. The original SSS design sets up a basic TCP/IP server, allowing users to control the development board's LEDs by sending commands over Ethernet. Michał extensively modified this design to support control of a complex sensor system instead of simple LEDs, enabling the detector to be controlled and data packets to be transmitted via the Ethernet interface.

As will be shown in Tab. 6.2, this framework already utilizes 65.4% of the logic elements (LEs), with the CPU and Ethernet components together occupying approximately 33.6%, and the sensor interface logic occupying approximately 23.3%. The available resources for the beam profile reconstruction are thus very limited.

6.4.2 Sensor Data Flow

This section describes the data flow within the sensor interface, followed by an explanation of the timing of data transmission from the sensor interface to the output interface.



(A)

(B)

FIGURE 6.7: Simplified block diagrams for the original Firmware [62]. (A) shows the basic structure of the firmware, while (B) provides a more detailed view.

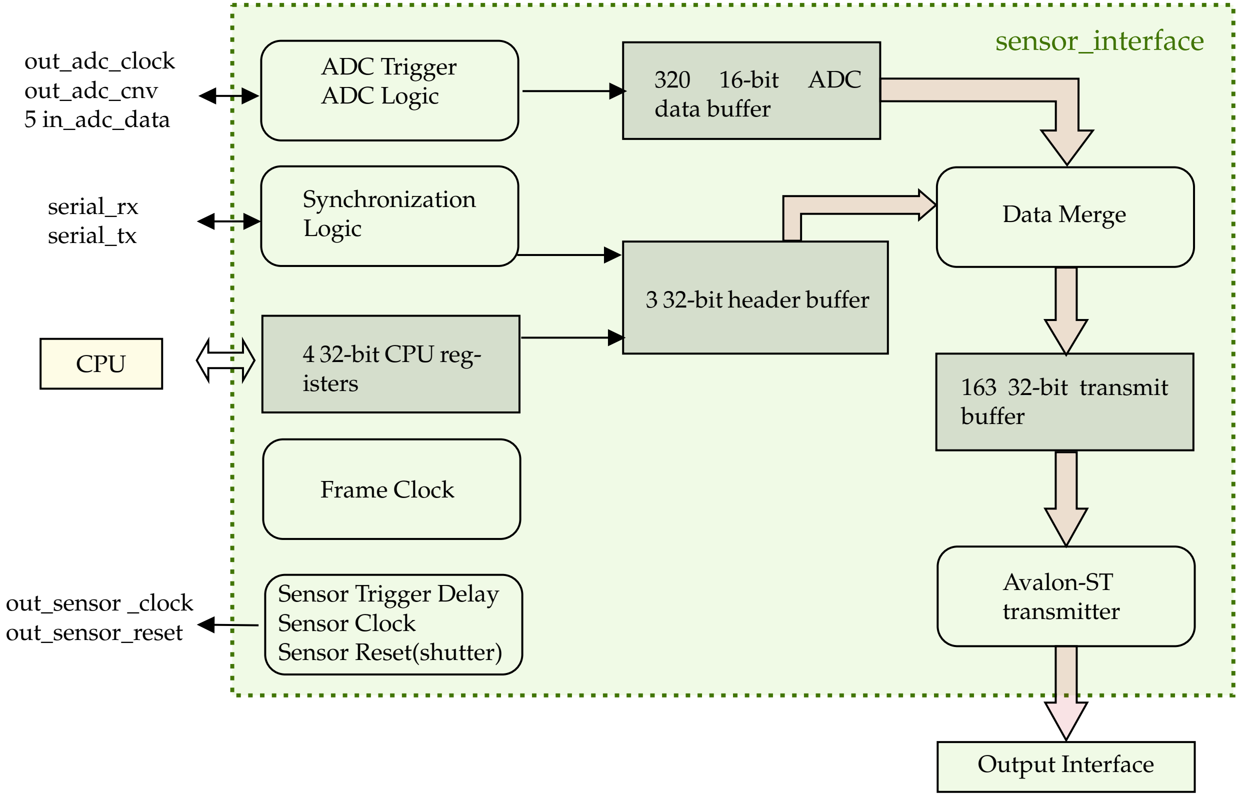


FIGURE 6.8: Logic block diagram of the sensor interface. The thick pink arrows indicate the data flow from the data buffer and header buffer to the output interface. The thin arrows represent input and output signals and data lines connected to the data buffer and header buffer.

Sensor Interface Data Flow

Fig. 6.8 shows the logic block diagram of the sensor interface. The logic blocks on the left side handle data collection, communication with the ADCs and photodiode arrays, synchronization logic, and interaction with the Nios II CPU. The logic blocks on the right side are responsible for transmitting data to the output interface. The central blocks serve as intermediate buffers.

There are four buffer regions in the sensor interface: the 4 32-bit CPU registers, the ADC data buffer, the header buffer, and the transmit buffer.The four 32-bit CPU registers are directly accessible by the CPU and control the entire sensor interface. These registers are only partially used by the sensor interface; the remaining unused bits are later utilized by the reconstruction firmware. The header buffer stores header information from the CPU registers and synchronization data from the synchronization logic. The ADC data buffer collects 16-bit ADC values from all 320 channels for a single frame. The contents of the ADC buffer and header buffer are merged into a 163-word 32-bit transmit buffer, which is then sent to the output interface word by word via the Avalon Streaming (ST) transmitter.

As described in Section 4.3, the photodiode arrays output video signals channel by channel, while the ADCs digitize them simultaneously. For all 64 channels, this process takes 76.16 µs, as described in Section 4.3. Once the data collection is complete, the contents of the ADC buffer are merged into the transmit buffer in a single 20 ns clock cycle, based on the 50 MHz system clock. The transmit buffer then begins sending the data to the output interface.

After the data merge, the ADC buffer is cleared and prepared to receive data for the next frame. The transmit buffer must be fully transferred before the next frame's data can be merged into it. At a readout rate of 10 kHz, the ADC data buffer is filled every 100 µs, meaning the transmit buffer must be completely transmitted within this 100 µs interval.

Due to the limited number of available LEs, as will be discussed in Section 6.4.3, there is not enough logic to implement an additional buffer. Consequently, the reconstruction firmware shares the same transmit buffer with the sensor interface, and both the transmission and reconstruction processes must be completed within 100 µs:

$$\text{Data Transmission} + \text{Reconstruction} < 100\,\mu\text{s}$$

Avalon Streaming Interface

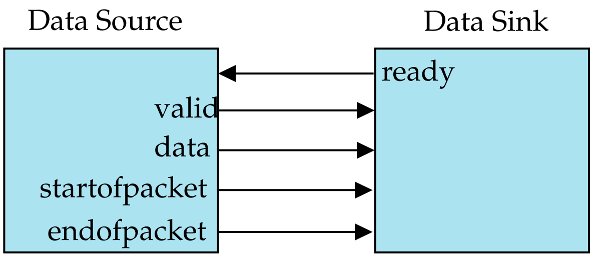


FIGURE 6.9: Avalon Streaming Interface signal details [70].

The Avalon Streaming (ST) interface is a standard protocol for streaming data word by word. It supports both backpressure and packet-based data transfer.

Fig. 6.9 shows the interface signals of the Avalon ST interface used in this thesis, and Fig. 6.10 illustrates the corresponding timing behavior.

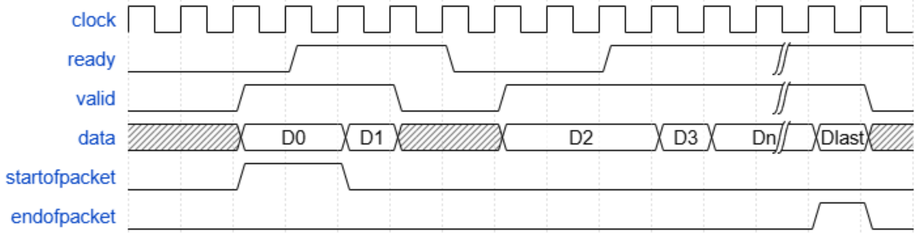


FIGURE 6.10: Avalon Streaming Interface signal timing chart [70].

The `valid` and `ready` signals manage the backpressure mechanism. The `valid` signal from the data source indicates that valid data is available on the data line. When the `ready` signal from the data sink is high, it means the sink has accepted the data, and the source will proceed to send the next word. If the `ready` signal is low, the data source must hold the current data until `ready` goes high again. With the correct assignment of these two signals, different data streams and functional modules can be connected in a flexible and efficient manner.

The `startofpacket` and `endofpacket` signals indicate the first and last words of a packet, respectively. They are used to mark the beginning and end of a data packet, allowing the receiving module to recognize the boundaries of the packet.

<——————-32 bits——————->

Header0	
Header1	
Header2	
channel 0	channel 1
channel 2	channel 3
⋮	⋮
channel 318	channel 319

TABLE 6.1: Data format in the transmit buffer of the sensor interface.

Tab. 6.1 shows the data format in the transmit buffer of the sensor interface, which is also the packet transmitted from the sensor interface to the output interface. The 163 words in the transmit buffer are sent one by one: the first 3 words form the header, which includes synchronization data, and the remaining 160 words contain the ADC data. Each of these 160 words holds two 16-bit ADC values from two consecutive channels, in the order of channel IDs 0, 1, 2, 3, and so on. The `startofpacket` signal is high for the the first header word. This is later used to trigger the reconstruction process in the reconstruction firmware.

This Avalon ST protocol is used not only between the sensor interface and the reconstruction firmware but also within the reconstruction firmware itself, providing a flexible mechanism for data transfer.

6.4.3 Resource Usage

Tab. 6.2 presents the resource usage of the original firmware without the reconstruction block. It occupies 65.4% of the LEs on the MAX10 FPGA. Notably, the

Component	**LUT**	**Flip-Flop**	**LE Percentage**
CPU	7881	5177	19.1%
Output Interface	5446	5451	16.0%
Sensor Interface	4202	10695	21.8%
Interconnection and Others	3135	2626	8.5%
Total	20664	23949	65.4%

TABLE 6.2: Resource usage of the original firmware. LUT stands for Look-Up Table, and LE stands for Logic Element.

sensor interface accounts for 21.8% of the total usage. This is large due to the presence of two large buffers for storing 320 16-bit ADC data, which alone consume 20% of the flip-flops.

The reconstruction firmware, which will be explained later in this thesis, consumes an additional 8.5% of the LEs, including 6.9% dedicated to RMS calculation. The pedestal subtraction, calibration and clustering modules together occupy only 1.6% of the LEs.

Details about how to get the figures in the Table are provided in the following:

The number of LUTs and flip-flops for each component is obtained from the Quartus Resource Utilization by Entity report. This report estimates the resource usage of each component independently. However, these numbers do not directly reflect the actual number of LEs used in the final implementation, as the entire design is optimized and synthesized into a single block with logic sharing among components.

As Fig. 6.1 show, one LE contains one LUT and one flip-flop. However, the total number of LEs used is not simply the sum of the LUT and flip-flop counts, since certain LUTs and flip-flops can be derived from the same LE.

The Quartus Flow Summary provides the final LE usage after logic placement and routing. The ratio between this final LE usage and the sum of LUTs (20 664) and flip-flops (23 949) in the Table is 72.9%. This factor accounts for both logic sharing among components and cases where a single LE provides both a LUT and a flip-flop.

The LE percentage values shown in the table are estimated using this ratio, according to the formula:

$$\text{LE Percentage} = \frac{\text{LUT} + \text{Flip-Flop}}{\text{Total LEs}} \times 72.9\%$$

where the total number of user LEs on the MAX10 FPGA is 49 760.

6.5 Reconstruction Firmware Design Concept

This section discusses the design concept of the reconstruction firmware, which is developed based within the original firmware framework and the resource constraints of the MAX10 FPGA.

6.5.1 Positioning

The purpose of the reconstruction firmware is to receive data from the sensor interface, process it using the reconstruction chain described in Chapter 5, and send the results to the output interface. For firmware verification purposes, the reconstruction block should also be capable of sending the raw data, the processed data, or both to the output interface.

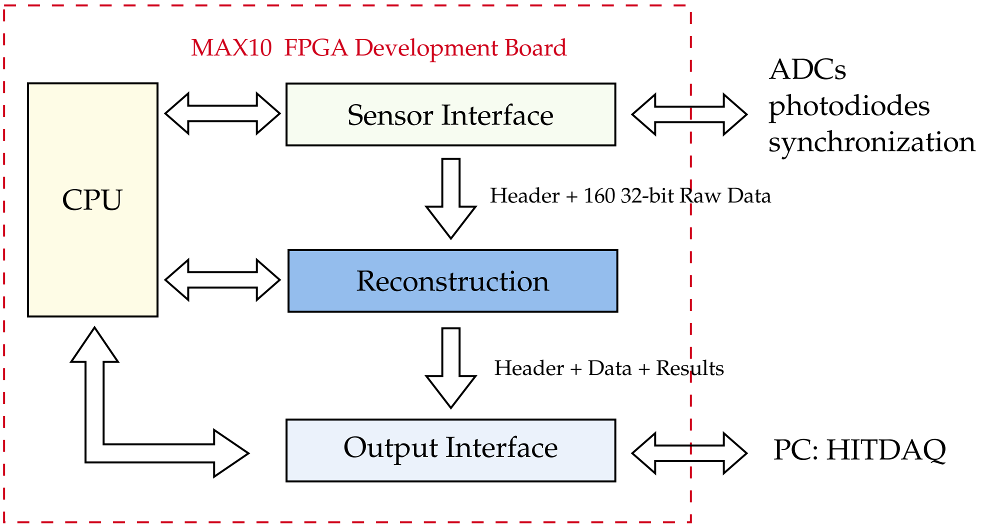


FIGURE 6.11: The reconstruction firmware is inserted between the sensor interface and the output interface.

In the original firmware design, the ADC data is packed in the sensor interface and sent to the output interface via the Avalon ST interface. Following this structure, the reconstruction module is inserted after the sensor interface and before the output interface, as illustrated in Fig. 6.11.

In addition, for flexible configuration of the reconstruction firmware, a 32-bit register configurable by the CPU is used to control the reconstruction process, allowing certain parameters of the reconstruction chain to be set by the CPU.

6.5.2 Limitation from Resource

The original firmware imposes two constraints on the reconstruction firmware. The first is a timing constraint: data transmission and reconstruction must be completed within 100 µs. The second is a resource constraint: the original firmware already utilizes 65.4% of the logic elements on the MAX10 FPGA, leaving less than 34.6% available for the reconstruction firmware.

The first step of the reconstruction chain is the pedestal subtraction. A natural approach would be to parallelize the pedestal subtraction across all 320 channels and perform the subtraction within a single clock cycle. However, this requires two additional large buffers: one to store the pedestal values and another to hold the subtracted data. This consumes a significant amount of resources, both in terms of flip-flops and the interconnections between the buffers.

An attempt was made to add an additional 320-channel 16-bit buffer for pedestal storage on top of the original design. As a result, the LE usage reached 90%, making it clearly infeasible to implement the full reconstruction chain on the MAX10 FPGA in this manner.

Following the original firmware structure, the reconstruction module is inserted between the sensor interface and the output interface. The reconstruction module therefore acts as a data sink that receives data from the sensor interface and as a data source that sends processed data and results to the output interface. Since the data is streamed from the sensor interface word by word, it is natural to also process it sequentially, word by word. In this case, pedestal values can be stored in memory and accessed sequentially during processing. Subtraction can then be performed one word at a time, significantly reducing LE usage.

Although this sequential approach requires more time, it remains well within the available timing budget. At a clock frequency of 50 MHz, processing all 320 channels once (one clock cycle per channel) takes 6.4 µs. Within a 100 µs time window, it can go through the 320 channels 15 times. This is sufficient for the reconstruction chain, which in principle requires only two passes over the data: one for pedestal subtraction, calibration, and clustering, and another for the RMS calculation. (The current reconstruction module performs four passes over the data.)

Due to the resource constraints, the reconstruction module is therefore designed to process data in a sequential manner.

6.5.3 Modularity

Implementing the complete reconstruction chain described in Chapter 5 is a challenging task. It is not feasible to realize the entire chain as a single unit at once. Therefore, the reconstruction chain is broken down into four main submodules: pedestal subtraction, calibration, clustering, and RMS calculation. Each submodule is developed and verified independently. The integrated reconstruction module, consisting of all connected submodules, is also verified after integration to ensure correct functionality.

To facilitate modular design and integration, the Avalon ST interface in Section 6.4.2 is employed across nearly all submodules, and all submodules operate on the same clock—the 50 MHz system clock. This setup allows for straightforward interconnection between modules and enables individual submodules to be easily replaced with alternatives implementing different algorithms, thereby enhancing the flexibility and reusability of the overall design.

6.5.4 Fixed-Point Arithmetic

The reconstruction firmware uses fixed-point arithmetic for all operations, as described in Section 6.3. For addition and multiplication, one clock cycle is allocated for each operation. For division and square root, $\frac{N}{2}$ clock cycles are allocated, where N is the bit width of the numerator in the case of division, and the bit width of the input in the case of square root.

6.6 Reconstruction Firmware Implementation

This section describes the implementation of the reconstruction firmware. It begins with an overview of the firmware, followed by detailed descriptions of the FSMs for the pedestal subtraction, calibration, clustering, and RMS calculation modules. It concludes with a discussion of the overall timing behavior of the reconstruction firmware, which must complete processing of the current frame before the next one arrives.

6.6.1 Structure

Fig. 6.12 shows the structure of the reconstruction firmware. The cyan arrows represent the Avalon Streaming data paths. The `startofpacket` signal for each data

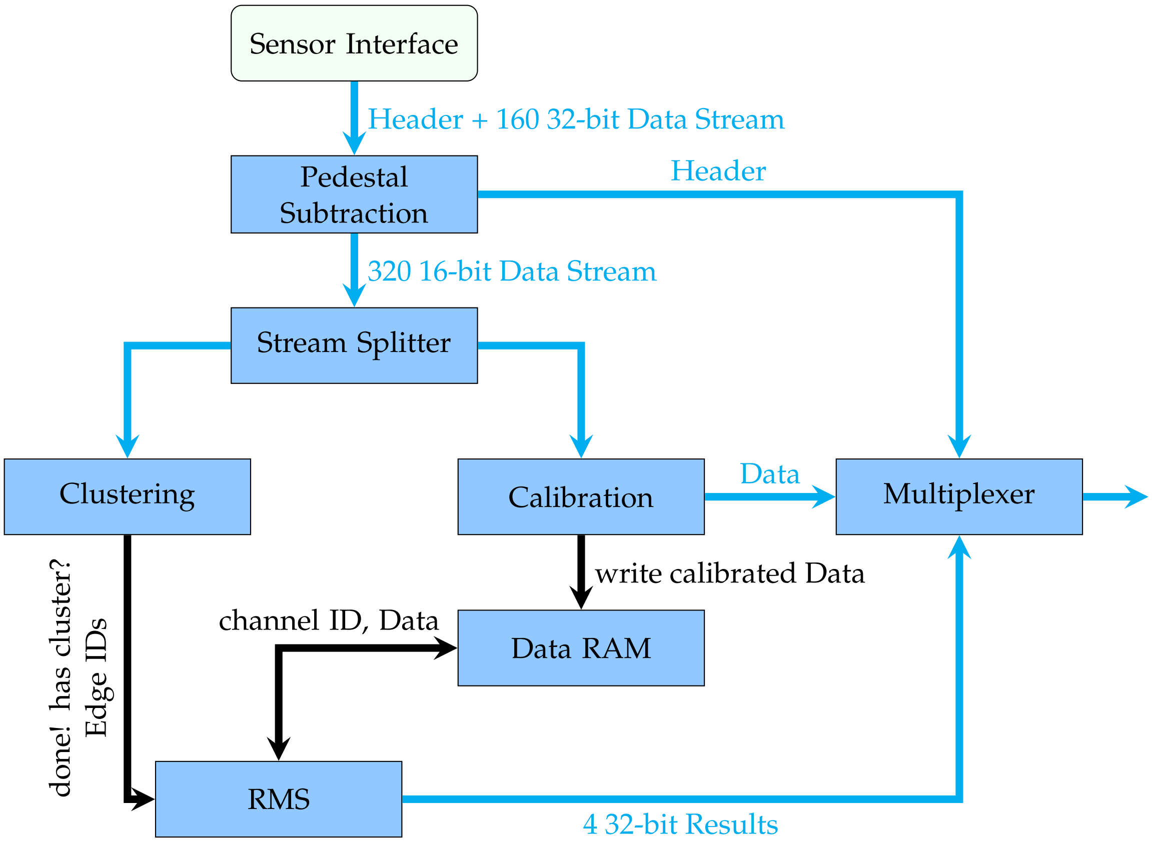


FIGURE 6.12: Logic Blocks and signal paths of the reconstruction firmware. The cyan arrows represent the data paths following the Avalon Streaming Interface protocol.

stream is used to trigger the subsequent module in the reconstruction chain. The `endofpacket` signal indicates the end of the data stream. And the `valid` and `ready` signals control the data flow between modules with the backpressure mechanism.

The entire reconstruction process is carried out for every single frame as follows. The sensor interface sends raw data to the pedestal subtraction module, which removes the pedestal from the input signals. The resulting 16-bit pedestal-subtracted data is then forwarded to the stream splitter in channel order. The stream splitter duplicates the data into two parallel streams: one directed to the clustering module and the other to the calibration module. The clustering module identifies the window of interest in the pedestal-subtracted data stream, while the calibration module calibrates the data and stores the calibrated values in the DATA RAM.

Once all channels have been processed, the clustering module sends a signal, along with the clustering results, indicating completion to the RMS module. If cluster is present, the RMS module initiates the calculation. It reads the calibrated data within the cluster, from the leftmost to the rightmost channel, and computes the mean and standard deviation step by step.

Regarding data streaming to the output interface, the pedestal subtraction module transmits the header, the calibration module provides the data, and the RMS module sends the results. These outputs are forwarded to the multiplexer, which selects the appropriate data stream to be sent to the output interface in the correct time sequence. Ultimately, the output interface receives the header, data, and results in this order and transmits them to the HITDAQ system. Multiple versions of the calibration module have been developed to support different data output modes, including raw data mode, calibrated data mode, and a combined mode that streams both raw and calibrated data.

6.6.2 Pedestal Subtraction

Besides data transmitting, the pedestal subtraction module performs two main functions: firstly, it calculates the pedestal value for each channel; and secondly, it subtracts the calculated pedestal from the raw data.

The core components of the pedestal subtraction module include a 32-bit frame ID counter, a 1-bit status indicator, and two 256-word 32-bit RAM blocks.

When the HITDAQ system issues the command to start a measurement, the 32-bit frame ID counter begins incrementing by one from zero for each incoming frame from the sensor interface. At a readout rate of 10 kHz, the counter can operate continuously for up to 4.9 days before overflowing, which is sufficient for typical measurement durations. The frame ID determines the operational mode of the pedestal subtraction module.

The two RAM blocks are used to store pedestal values separately for even and odd channels, as the ADC data is streamed in channel pairs.

One FSM controls the two RAMs according to the frame ID counter:

- When the frame ID is 0, both RAMs are initialized to zero.
- For frame IDs from 1 to 8192, the raw data value at each address is added to the value already stored in RAM, and the updated sum is written back to the same address. After 8192 frames, each address holds the accumulated sum:

$$\text{Sum}_i = \sum_{j=1}^{8192} S_{i,j}$$

- From frame ID 8193 onward, accumulation stops and the stored value is output to calculate the pedestal by right-shifting the sum by 13 bits, as $2^{13} = 8192$:

$$\mu_i = \text{Sum}_i \gg 13$$

  where μ_i is the estimated pedestal value for channel i. The least significant bit of μ_i is 1 ADC count. Notably, as in Chapter 5, the precision of this pedestal measurement is better than 1 ADC count.

From frame IDs 0 to 8192, the pedestal subtraction module outputs the raw data. Starting from frame ID 8193, the pedestal value is subtracted from the raw data of each channel, and the pedestal-subtracted data are output.

The frameID is reset to zero at the end of each measurement and starts to increment again at the beginning of the next. If the measurement continues without interruption, it takes 4.9 days before it overflows back to zero. Thus, the pedestal is recalculated either at the beginning of each measurement or, at the latest, after 4.9 days when the frameID overflows.

The 1-bit status indicator is used to mark whether the output data is raw or pedestal-subtracted. It is set to 0 for frame IDs from 0 to 8192, and to 1 from frame ID 8193 onward. This signal is passed to the RMS module and is streamed together with the RMS calculation results. It also instructs the subsequent modules not to perform calculations for frame IDs 0 to 8192, and to start calculations from frame ID 8193 onward.

For every two channels, a total of two clock cycles are required: one to read the pedestal values from the two RAMs and one to subtract them from the raw data. This process can be optimized by prefetching the pedestal values, allowing the subtraction to occur concurrently with reading the pedestal for the next word. As a result, the operation can, on average, be completed within a single clock cycle per two channels. However, this optimization does not reduce the overall processing time, as the module must also stream all 320 channels of 16-bit pedestal-subtracted or raw data to the stream splitter in channel order, so that at least 2 clock cycles are required to stream out the two channels of data.

In conclusion, the pedestal is automatically calculated and subtracted from the raw data. The pedestal subtraction module streams out raw data for frame IDs 0 to 8192, and pedestal-subtracted data from frame ID 8193 onward. The pedestal subtraction indicator also instructs the subsequent modules not to perform calculations for frame IDs 0 to 8192, and to start calculations from frame ID 8193 onward. This indicator can also be used in future HIT BAM systems to signal when the detector is ready for beam measurements.

6.6.3 Stream Splitter

The stream splitter module is responsible for duplicating the 320-channel, 16-bit data stream from the pedestal subtraction module into two separate streams: one directed to the clustering module and the other to the calibration module. All three interfaces follow the Avalon Streaming (ST) protocol. As a result, the pedestal subtraction module can transmit the next data only after both the clustering and calibration modules have successfully received the current data.

6.6.4 Clustering

The clustering module identifies the window of interest in the data stream for each data frame. As illustrated in Fig. 5.14, it searches for the leftmost and rightmost boundaries of clusters that exceed the threshold T_c and span at least N_c consecutive channels of the frame.

Since the pedestal-subtracted data is streamed in channel order, the FPGA implementation differs slightly from the conceptual sketch in Fig. 5.14. Instead of scanning from both directions using two search boxes, a single search box moves from left to right as the data streams in channel order. The channel IDs of the left edge of the first cluster and the right edge of the last cluster that satisfy the clustering settings are recorded as the leftmost and rightmost boundaries.

As the clustering module receives the data stream, it scans each channel sequentially, processing one channel per clock cycle (assuming the data is streamed at one channel per clock, which is not the case with the calibration module).

After all channels have been scanned, the module sends a signal indicating the completion of the clustering process for the frame, together with the clustering results. The clustering results include a flag specifying whether at least one cluster was found, as well as the channel IDs of the boundaries of the window of interest.

For flexibility, both T_c and N_c are controlled by a CPU-configurable register, allowing them to be set by the HITDAQ via the Nios II CPU.

6.6.5 Calibration

The calibration module multiplies the pedestal-subtracted data by a calibration factor f_i for each channel. These calibration factors are stored in a dual-port RAM on the FPGA: one port is accessible to the Nios II CPU for configuration via HITDAQ, and the other is used by the calibration module to get the factor for each channel.

The calibration factors are determined through offline analysis in Section 7.2, and are loaded into the volatile RAM on the FPGA by HITDAQ. Since they are lost after each FPGA power cycle, they must be reloaded. This is done automatically when the HITDAQ connects to the Nios II server.

The calibrated data is then written to the DATA RAM, from which it is later read by the RMS module.

Currently, each channel requires one clock cycle to read the calibration factor and another clock cycle to perform the multiplication. Since the pedestal subtraction, clustering, and calibration modules are pipelined, the calibration module becomes the bottleneck, as it is the slowest among the three—requiring two clock cycles to process each channel. This limitation can be addressed by prefetching the calibration factor in advance, allowing the calibration for each channel to be completed within a single clock cycle. With this improvement, as shown in Fig. 6.14, the total time required for the pedestal subtraction, clustering, and calibration would be reduced by 320 clock cycles (6.4 µs).

As mentioned previously, multiple firmware versions of the calibration module have been developed to support different data output modes, including raw data, calibrated data, or both. The version that outputs both raw and calibrated data was specifically developed for validating the functionality of the pedestal subtraction and calibration modules.

6.6.6 RMS

Once the RMS module receives a has cluster signal from the clustering module, it begins computation while reading the calibrated data from the DATA RAM, from the leftmost to the rightmost channel within the window of interest. The mean $\hat{\mu}$, standard deviation $\hat{\sigma}$, and amplitude $\hat{A}$ are computed according to Equations 5.31 step by step:

$$\hat{\mu} = \frac{\sum x_n \hat{y}_n}{\sum \hat{y}_n}$$

$$\hat{\sigma} = \sqrt{\frac{\sum (x_n - \hat{\mu})^2 \hat{y}_n}{\sum \hat{y}_n - 1}}$$

$$\hat{A} = \hat{y}_{\max}$$

According to the equations above, the calibrated data would need to be read from the DATA RAM twice: once to calculate the mean $\hat{\mu}$, and a second time to compute the standard deviation $\hat{\sigma}$ using the previously obtained $\hat{\mu}$, since $\hat{\mu}$ appears inside the parentheses. However, the expression for $\hat{\sigma}$ can be simplified to

$$\hat{\sigma} = \sqrt{\frac{\sum x_n^2 \hat{y}_n - \hat{\mu}^2 \sum \hat{y}_n}{\sum \hat{y}_n - 1}}, \tag{6.1}$$

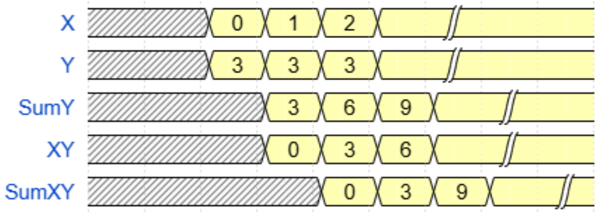


FIGURE 6.13: The timing diagram illustrates the first loop of the RMS calculation. X represents x_n, and Y represents $\hat{y}_n$. The numbers in the X waveform indicate the channel ID. The Y waveform represents the calibrated data corresponding to the X channel, which is set to 3 in this example for illustration purposes.

so that only a single pass over $\hat{y}_n$ is required. This is a useful feature of the moments calculation: the term in parentheses can be simplified by the first moment equation, so that only a single pass over the data is required.

The current RMS module in this work, however, uses the non-simplified equations and makes two passes over the data.

According to the equations, four types of operations are required: addition, multiplication, division, and square root. As discussed in Section 6.5, the fixed-point arithmetic is used for all operations. For addition and multiplication, each operation is completed in a single clock cycle. For division and square root operations, $\frac{N}{2}$ clock cycles are allocated, where N is the bit width of the numerator in the case of division, and the bit width of the input in the case of the square root. Methods for fixing the bit width are discussed in Section 6.7.

Fig. 6.13 shows the timing diagram of the first loop of the RMS calculation. The calibrated data is read one clock cycle before being used in computation. The calculation is pipelined such that each addition and multiplication operation is completed within a single clock cycle. During the first pass through the $\hat{y}_n$ values, the quantities $\sum x_n \hat{y}_n$ and $\sum \hat{y}_n$ are computed for all channels in the window of interest. Simultaneously, the maximum value $\hat{y}_{\max}$ is determined.

A division IP core with a 48-bit numerator and a 32-bit denominator is used to calculate the mean $\hat{\mu}$ after the first pass. This division core is also reused for calculating the standard deviation $\hat{\sigma}^2$. It is pipelined for 24 clock cycles.

In the second pass, the term $\sum (x_n - \hat{\mu})^2 \hat{y}_n$ is calculated. After this, the division module and a square root module implemented by the square root IP core are used to compute the standard deviation $\hat{\sigma}$. The square root IP core is configured as a 26-bit input and a 13-bit output, which takes 13 clock cycles to complete.

Once the RMS calculation is completed, the results—including the mean $\hat{\mu}$, standard deviation $\hat{\sigma}$, amplitude $\hat{A}$, the pedestal subtraction indicator, and the clustering results—are sent to the multiplexer, marking the end of the reconstruction process for the current frame of data.

In the worst-case scenario, where the window-of-interest length is 320 channels, the RMS module requires 640 clock cycles to go through y_n twice, taking 12.8 µs.

If the RMS module receives a clustering-complete signal with the flag indicating that no cluster was found, it directly enters the result-transmission state and transmits RMS results with all values set to zero, together with the pedestal subtraction indicator and the clustering results.

The RMS module should be simplified in the future using Eq. 6.1. With this form, only a single pass over the data is required to accumulate $\sum \hat{y}_n$, $\sum x_n \hat{y}_n$, and $\sum x_n^2 \hat{y}_n$. Once these sums are available, the first moment $\hat{\mu}$ can be computed, followed by the

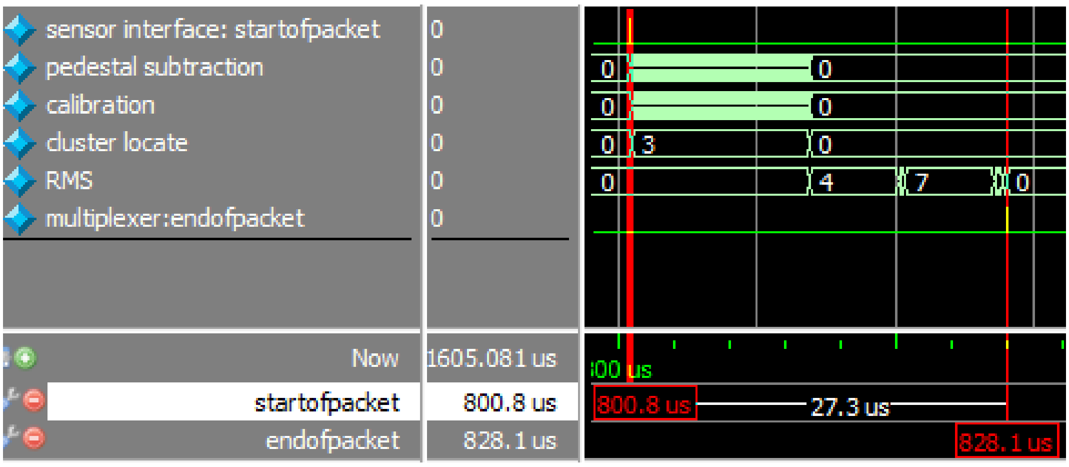


FIGURE 6.14: Simulated timing diagram of the reconstruction firmware for processing one frame of data. The waveforms for pedestal subtraction, calibration, clustering, and RMS represent the state of the FSMs in each respective module, where 0 indicates the idle state for all modules.

second moment $\hat{\sigma}$. This approach also extends naturally to higher moments (third, fourth, etc.).

The current RMS module is more complex and has higher latency than the simplified design. Nevertheless, it meets the timing requirement and performs with the required precision, as will be shown in the following sections.

6.6.7 Overall Timing and Resource Usage

Fig. 6.14 shows a simulated timing diagram of the reconstruction firmware for processing one frame of data. In this simulation, the input signal is configured to be high and wide, resulting in a window of interest that spans all 320 channels, which leads to the longest possible data processing time.

The pedestal subtraction, calibration, and clustering modules are pipelined, allowing them to process data in parallel. The dense switching of states in the pedestal subtraction and calibration waveforms corresponds to transitions between reading from RAM (for pedestal values or calibration factors) and executing the calculation. A state value of 3 in the clustering module indicates that it is processing the first cluster. For each channel, this stage requires two clock cycles to process the data, which is limited by the calibration module, as described in Section 6.6.5. For 320 channels, this results in a processing time of 12.8 µs plus several clocks of pipeline latency.

The RMS module is triggered by the clustering module and processes data sequentially. A state value of 4 in the RMS waveform indicates that the module is performing the first pass through y_n, while a state value of 7 corresponds to the second pass. The RMS module adds an additional two clock cycles per channel in the cluster, along with some pipeline latency and arithmetic processing time.

In this worst-case scenario, where the window of interest spans all 320 channels, the total processing time for the entire frame is 27.3 µs, from the `startofpacket` signal of the sensor interface to the `endofpacket` signal of the multiplexer. During this time period, not only are the whole reconstruction chain performed, but the header, data, and results are also streamed out. This total processing time remains well within the 100 µs time budget allocated for data processing.

All modules are reset to their idle state once processing of the current frame is completed, and they wait for the next `startofpacket` signal from the sensor interface to begin processing the next frame. In Fig. 6.14, the pedestal subtraction is also initialized to the idle state after the current frame, indicating that for the next frame it will process the pedestal starting from the first channel. The frame ID, which controls the pedestal subtraction module and determines when to calculate the pedestal, is reset only at the start of each measurement.

As mentioned in Section 6.4.3, the whole reconstruction firmware occupies only 8.5% of the logic elements (LEs) on the MAX10 FPGA, including 6.9% dedicated to RMS calculation. Together with the original firmware framework, the total resource usage is 74% of the LEs on the MAX10 FPGA.

6.7 Constrain the Fixed-point Operations

Section 6.6 describes the implementation of the reconstruction firmware and its timing behavior. The final remaining step in the implementation is to fix the bit width and the weight of the least significant bit (LSB) in the fixed-point arithmetic. This is essential to ensure that no overflow occurs during calculations and that the results remain within the expected numerical range.

This section discusses how the bit width and LSB weight are determined for each operation in the reconstruction chain. It also highlights the precision and computational limitations of the current reconstruction firmware.

6.7.1 Pedestal Subtraction

The formula for pedestal subtraction is:

$$y_i = S_i - \mu_i$$

where S_i is the raw amplitude, and μ_i is the pedestal for channel i.

The raw amplitude S_i is a 16-bit unsigned integer output from the ADC:

$$S_i \in \{0, 1, 2, \ldots, 2^{16} - 1\} = [0, 65535]$$

In practice, its dynamic range spans from a typical pedestal value of around 1000 to a saturation level of approximately 51 500 ADC counts.

The pedestal μ_i, as described in Section 5.1, is also computed as a 16-bit unsigned integer, with a precision better than 1 ADC count. Its least significant bit (LSB) is defined as:

$$\mathrm{LSB}_{S_i} = \mathrm{LSB}_{\mu_i} = 1 \text{ ADC count}$$

The result of subtracting two 16-bit unsigned integers is natively a 17-bit signed integer. To keep y_i in 16-bit format for streaming convenience, the least significant bit is discarded by right-shifting the result by one bit. Therefore, y_i becomes a 16-bit signed integer, and the pedestal subtraction implemented on the FPGA is expressed as:

$$y_i = (S_i - \mu_i) \gg 1 \tag{6.2}$$

with

$$\mathrm{LSB}_{y_i} = 2 \text{ ADC counts}$$

and the precision is thus:

$$\sigma_{y_i} = \frac{\text{LSB}_{y_i}}{\sqrt{12}} = 0.58 \text{ ADC counts}$$

and the dynamic range of y_i is:

$$y_i \in \{-32768, -32767, \ldots, 32767\} = [-32768, 32767]$$

In practice, the dynamic range of y_i is approximately $[-200, 25000]$. This quantization step introduces a precision loss, but it remains acceptable, as the resulting LSB and precision are smaller than the typical standard deviation of the noise, which is approximately 9 ADC counts.

6.7.2 Calibration

The formula for calibration is:

$$\hat{y}_i = f_i \cdot y_i$$

where f_i is the calibration factor for channel i, and y_i is the pedestal-subtracted signal from the previous step, $\hat{y}_i$ is the calibrated signal.

The calibration factor f_i is represented as a 17-bit signed fixed-point number, structured as follows:

$$f_i = \text{Q}(1, 3, 13)$$

This denotes 1 sign bit (which is always 0, since $f_i \geq 0$), 3 integer bits, and 13 fractional bits. Therefore, the value of f_i lies in the range:

$$f_i \in [0, 2^3) = [0, 8) \text{ with } \text{LSB}_{f_i} = 2^{-13} \approx 0.00012$$

In practice, the default calibration factor is 1, and the actual values typically range from 0.5 to 2.0.

The multiplication is performed as a signed multiplication between the 16-bit signed value y_i and the 17-bit fixed-point calibration factor f_i, resulting in a 32-bit signed product P_i in Q-format:

$$P_i = f_i \cdot y_i = \text{Q}(1, 3 + 15, 13) = \text{Q}(1, 18, 13)$$

To obtain the final calibrated value $\hat{y}_i$ as a 16-bit signed integer, the result P_i is truncated by discarding the rightmost 13 bits (corresponding to the fractional part of f_i) and the leftmost 3 non-sign bits:

$$\hat{y}_i = \text{Q}(1, 18 - 3, 13 - 13) = \text{Q}(1, 15, 0)$$
$$\hat{y}_i \in \{-32768, -32767, \ldots, 32767\} = [-32768, 32767]$$

When y_i is large, the multiplication can potentially lead to overflow. However, since the calibration factor f_i is typically around 1, and as will be shown in the next chapter, y_i remains relatively small for all types of pencil beams, the multiplication does not result in overflow for the application as BPM at HIT.

The precision of $\hat{y}_i$ is approximately given by:

$$\sigma_{\hat{y}_i} \simeq \sqrt{\sigma_{f_i}^2 \cdot y_i^2 + f_i^2 \cdot \sigma_{y_i}^2}$$

For the extreme case of $f_i = 2$ and $y_i = 25000$, this results in:

$$\sigma_{\hat{y}_i} \simeq \frac{1}{\sqrt{12}}\sqrt{(2^{-13})^2 \cdot 25000^2 + 2^2 \cdot 2^2} \approx 1.44 \text{ ADC counts}$$

This is still much smaller than the typical standard deviation of the noise, which is around 9 ADC counts. Therefore, the precision loss introduced by the calibration step is acceptable.

6.7.3 Clustering

The clustering parameters include the cluster size N_c, defined in terms of the number of channels, and the cluster threshold T_c, defined as an 8-bit unsigned integer ranging from 0 to 255. The threshold T_c is compared with the pedestal-subtracted signal y_i.

$$\text{LSB}_{N_c} = 1 \text{ channel}, \quad \text{LSB}_{T_c} = 2 \text{ ADC counts}$$

The clustering module produce the leftmost and rightmost channel IDs of the cluster, which are 9-bit unsigned integers. The range of these IDs is from 0 to 319, corresponding to the 320 channels in the detector.

6.7.4 RMS Calculation

The RMS calculation module includes 5 additions, 3 multiplications, 2 divisions, and 1 square root operation.

To fix the bit width and the LSB of each operation, the LSB of the final results are defined first. Both the position μ and the standard deviation σ are with an LSB of 0.05 mm, in order to meet the resolution requirement of 0.2 mm for beam position.

With this definition, the relative position of channel n on array m is represented on the FPGA as an integer value x_n:

$$x_n = \frac{0.8\,\text{mm} \cdot n + 0.2\,\text{mm} \cdot m}{0.05\,\text{mm}}$$

Here, the 0.2 mm term accounts for the gap between adjacent fiber arrays, and the 0.8 mm term accounts for the pitch size of the photodiodes. To accommodate all 320 channels, x_n is represented as a 13-bit unsigned integer, covering a valid range from 0 to approximately 409.55 mm.

With 0.05 mm as the unit, each channel width of 0.8 mm corresponds to a value of 16 in binary, and the inter-array gap of 0.2 mm corresponds to a value of 4. Thus, the positions and gaps are naturally quantized in the fixed-point representation.

Then, starting with the defined values of x_n and simulated beam $\hat{y}_n$ for various beam widths and amplitudes, each arithmetic operation in the RMS module is performed using integer arithmetic to fit the bit width. The simulated beam $\hat{y}_n$ covers an extremely wide range of beam parameters, with the peak value $\hat{y}_{\max}$ ranging from 20 to 32767, beam widths from 2 mm to 50 mm, and beam positions spanning from channel 20 to channel 300.

The bit width required for each intermediate result is determined by identifying the maximum bit size needed across all the simulation cases. To support this analysis, the GNU Multiple Precision Arithmetic Library (GNU MP) [71] is used. The GNU MP enables arbitrary-precision arithmetic, allowing the bit widths of operands and results to be flexibly adjusted.

Several additional bits are allocated to each operation to provide margin and ensure that no overflow occurs during computation.

TABLE 6.3: Bit Width of the Operands and Results in RMS Calculation

ID	Expression	Bit Width	ID	Expression	Bit Width
1	x_n	13	2	$\hat{y}_n$	15
3	$\sum \hat{y}_i$	31	4	$x_n \hat{y}_i$	31
5	$\sum x_n \hat{y}_n$	34	6	$\hat{\mu}$	13
7	$x_n - \hat{\mu}$	14	8	$(x_n - \hat{\mu})^2$	28
9	$(x_n - \hat{\mu})^2 \hat{y}_n$	43	10	$\sum (x_n - \hat{\mu})^2 \hat{y}_n$	43
11	$\frac{\sum (x_n - \hat{\mu})^2 \hat{y}_n}{\sum \hat{y}_n - 1}$	26	12	$\hat{\sigma}$	13

Tab. 6.3 summarizes the bit width (excluding the sign bit) of each operand in the RMS calculation. Except for operands 11 and 12, all other operands are implemented as signed integers in the FPGA for convenience.

The square root operation is implemented with a 26-bit input and produces a 13-bit output, with 13 clock cycles allocated for its completion. The division operation is implemented as an unsigned division with a 48-bit numerator and a 32-bit denominator, and 24 clock cycles are allocated for its completion.

Besides the limitation on bit width, a limitation of this RMS calculation scheme is that both the numerator and the denominator of the division must be greater than zero. This implies that in Tab. 6.3 operand 3 must be greater than 1, and operands 5 and 9 must be strictly positive. These conditions are satisfied as long as $\hat{y}_i$ is positive, which is guaranteed by the clustering module that correctly identifies the clusters above the positive cluster threshold T_c.

6.8 Firmware Verification

Section 6.6 describes the implementation of the reconstruction firmware and its timing behavior, while section 6.7 describes the precision and limitation of each operation in the reconstruction chain.

In this section, the reconstruction firmware is implemented on the FPGA, and the complete detector system is set up for testing. Some tests are performed at HIT using pencil beams, while others are conducted in the laboratory using an ultraviolet (UV) LED light source. The objective is to verify the functionality of the reconstruction firmware. The verification is carried out by comparing the results obtained from the FPGA, which uses fixed-point numbers, with those calculated by the CPU using floating-point numbers.

6.8.1 Pedestal Subtraction

Fig. 6.15 illustrates the reconstruction firmware settings for verifying the pedestal subtraction and clustering modules. The calibration factor for all channels is set to 1.0, so the calibrated data is equivalent to the data after pedestal subtraction, which is also the input to the clustering module. Both the raw and calibrated data, along with the results, are streamed out and then saved by HITDAQ for offline comparison.

To verify the pedestal subtraction, data are saved starting from frame ID 0. The pedestal on the CPU is calculated as the average of the raw data from frame ID

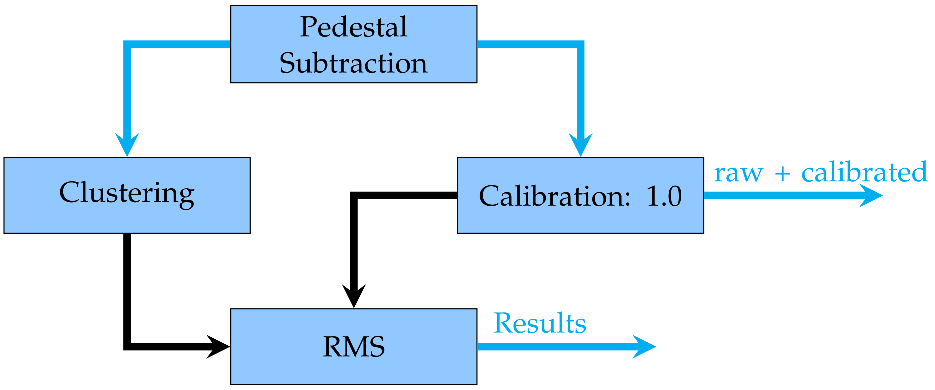


FIGURE 6.15: Simplified diagram of Fig. 6.12, showing the settings for verifying the pedestal subtraction and clustering modules.

1 to frame ID 8192, following the same procedure as on the FPGA. The pedestal-subtracted data is then obtained by subtracting the computed pedestal from the raw data.

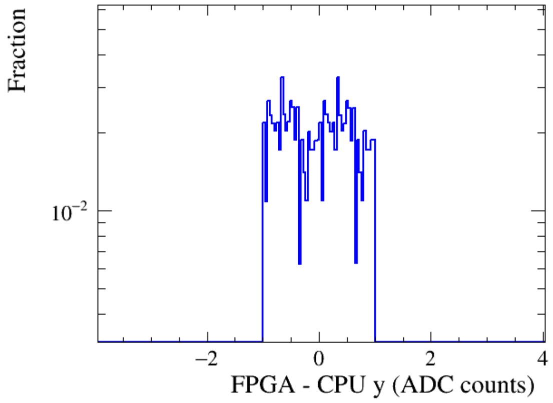


FIGURE 6.16: Normalized histogram of FPGA–CPU differences of pedestal-subtracted data for all 320 channels over one hour of data at 10 kHz (1.1×10^{10} entries).

The difference between the pedestal-subtracted data calculated by the FPGA and the CPU is shown in Fig. 6.16 for 1.1×10^{10} entries. The y-axis is plotted on a logarithmic scale to show that all differences lie fully within the range $[-1, 1]$.

To understand why the range is $[-1, 1]$: the calculated pedestal value on the FPGA, μ_F, is a 16-bit unsigned number with an LSB of one ADC count, as described in Subsection 6.6.2. As μ_F is equivalent to the CPU-calculated floating-point pedestal μ_C by truncating the fractional part, their difference is

$$\mu_C - \mu_F \in [0, 1) \text{ ADC count.} \tag{6.3}$$

The FPGA-calculated pedestal-subtracted data is

$$y_F = (S - \mu_F) \gg 1, \tag{6.4}$$

where S is the 16-bit unsigned raw data. The right-shift operation drops the least significant bit, so

$$y_F - (S - \mu_F) \in [-1, 0] \text{ ADC count.} \tag{6.5}$$

In contrast, the CPU calculates

$$y_C = S - \mu_C. \tag{6.6}$$

Thus, the overall difference is

$$\begin{aligned} y_F - y_C &\in (S - \mu_F) + [-1, 0] - (S - \mu_C) \\ &\in (\mu_C - \mu_F) + [-1, 0] \\ &\in [0, 1) + [-1, 0] \\ &\in [-1, 1) \text{ ADC count.} \end{aligned} \tag{6.7}$$

Hence, the differences in Fig. 6.16 are confined to the range $[-1, 1]$.

This histogram confirms that the pedestal subtraction on the FPGA operates as expected, with deviations remaining well within the quantization precision.

6.8.2 Clustering

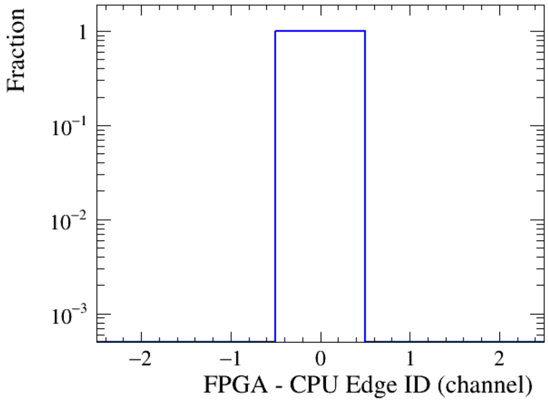


FIGURE 6.17: Normalized histogram of FPGA–CPU differences in window-of-interest edge channel IDs (bin size: 1 channel). All differences are exactly zero channel.

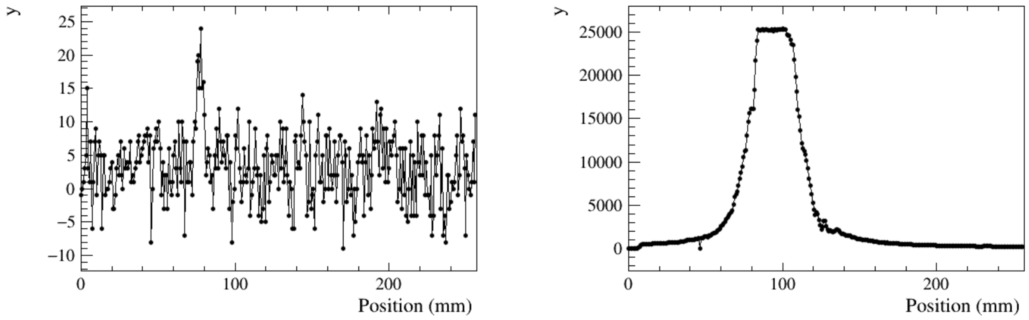
To verify the clustering module, the same clustering algorithm is implemented on the CPU. The calibrated data obtained in Fig. 6.15 are processed with the clustering algorithm on the CPU using the same clustering settings as on the FPGA. The calibration factor is set to 1, so that the input data to the CPU clustering are identical to those used on the FPGA.

Fig. 6.17 shows the histogram of the difference between the window-of-interest edges calculated by the FPGA and the CPU. The histogram shows a zero difference across all events, indicating that the clustering module on the FPGA performs as expected and matches the CPU implementation exactly. More than 22 million frames of data, covering a wide range of window-of-interest lengths and positions, were collected for the verification.

This histogram confirms that clustering on the FPGA operates correctly.

6.8.3 Calibration

To verify the calibration module—specifically, that the calibration factor for each channel can be correctly assigned and the calibrated data can be accurately computed—a set of random calibration factors f ranging from 0.5 to 2.0 is generated and applied in the firmware. The amplitude values y are produced using a UV LED beam in the laboratory, covering a wide range of signal amplitudes.



(A) Narrow, low-intensity signal

(B) Wide, high-intensity signal

FIGURE 6.18: Examples of signals generated by the UV LED point source. Y axis is the amplitude of the pedestal subtracted data.

Fig. 6.18 shows two examples of signals generated by the UV LED source. It can produce a wide range of signals, from narrow and low-intensity to wide and high-intensity, capable of saturating the sensor.

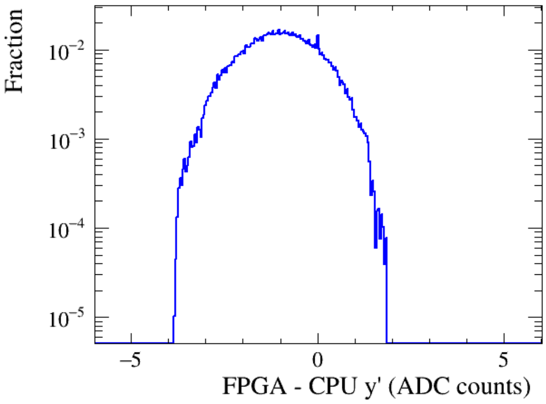


FIGURE 6.19: Normalized histogram of FPGA–CPU differences in calibrated data.

Fig. 6.19 shows the 1D histogram of the difference between calibrated data computed by the FPGA and the CPU for all the 320 channels, excluding overflow data. For this histogram, the amplitude y generated by the UV LED and natural light ranges from -119 to 24574.

The calibrated data by the CPU y'_C are obtained by multiplying the CPU-calculated pedestal-subtracted data y_C by the corresponding 17-bit signed calibration factor f:

$$y'_C = f \cdot y_C.$$

On the FPGA, the 16-bit signed calibrated data y'_F are calculated as

$$y'_F = \text{integer}(f \cdot y_F),$$

where only the integer part is kept, with LSB of y'_F = 2 ADC counts.

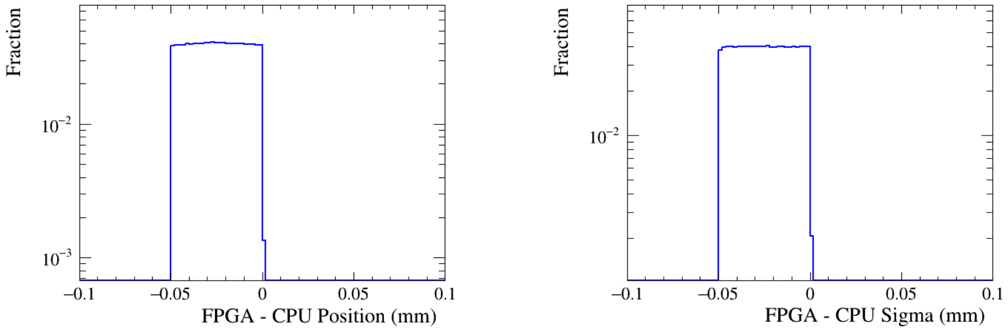
Thus, the difference between y'_F and y'_C is

$$\begin{aligned} y'_F - y'_C &= \text{integer}(f \cdot y_F) - f \cdot y_C \\ &\in f \cdot y_F + (-2, 0] - f \cdot y_C \\ &\in (-2, 0] + f \cdot (y_F - y_C) \\ &\in (-2, 0] + 2 \cdot [-1, 1) \\ &\in (-4, 2) \text{ ADC counts.} \end{aligned}$$

In Fig. 6.19, y'_F and y'_C differ within the range $(-4, 2)$ ADC counts, showing that the calibration module on the FPGA operates correctly. The standard deviation of the difference is 0.97 ADC counts, well below the noise level, ensuring sufficient precision for single-frame beam profile reconstruction.

6.8.4 RMS

To verify the RMS calculation module, the calibrated data and the window-of-interest edges from the FPGA are used to compute the RMS results on the CPU. The position μ, beam size σ, and amplitude are compared with the results produced by the FPGA.



(A) Difference in position (μ) between FPGA and CPU

(B) Difference in beam size (σ) between FPGA and CPU

FIGURE 6.20: Normalized histograms of the difference between the FPGA and CPU results for (a) position and (b) beam size (σ) in the RMS calculation.

The test was conducted at HIT in August 2024, with the firmware streaming out the calibrated data and RMS results. Six planes were set up, and a total of $71,927,652 \times 6$ frames were collected across 40 different runs, covering a wide range of beam sizes, intensities, energies, and positions.

As shown in Fig. 6.20, the differences in position and σ are within $[-0.05, 0]$ mm. As described in Subsection 6.3, for division and square root operations on FPGA the remainder is set to be non-negative. Thus, the result is always smaller than the true value by no more than the weight of the LSB. For both position and σ, the LSB is 0.05 mm. These results confirm that the RMS module operates as intended.

Abnormal Event

However, 383 out of the $71,927,652 \times 6$ recorded frames (0.887 per million) show a discrepancy in the calculated beam σ. Fig. 6.21 presents one such abnormal event. All these events exhibit high common-mode noise like Fig. 6.21, however, these 383

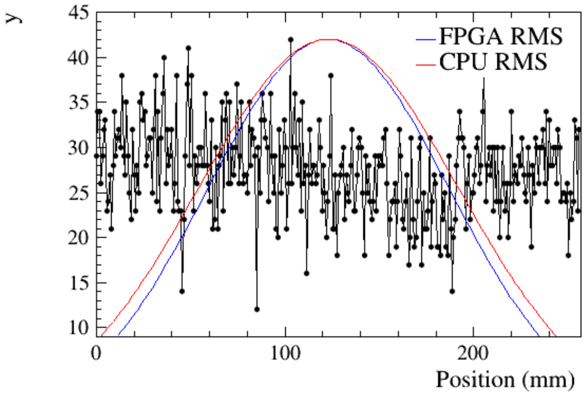


FIGURE 6.21: An example of an abnormal event with mismatched σ between FPGA and CPU.

frames are not the only frames with high common-mode noise, as many other frames with similar noise levels are processed correctly by the firmware.

These 383 frames all originated from the same detector board and occurred in a burst during a single run. Fig. 6.22 shows the evolution of the σ difference between FPGA and CPU and the common mode noise over time. The σ discrepancy appears suddenly after around 300 seconds, while the common mode noise distribution remains stable throughout the run.

The 383 abnormal frames were reprocessed using the GNU MP library with fixed-point arithmetic on the CPU. The results matched the CPU's floating-point calculations, but differed from the FPGA results. No overflow or bit-width-related error occurred in the CPU calculation.

Extensive tests were conducted in the lab to reproduce the anomaly, but the issue could not be recreated. All lab test results, including those shown in Fig. 6.18, demonstrated full agreement between FPGA and CPU results.

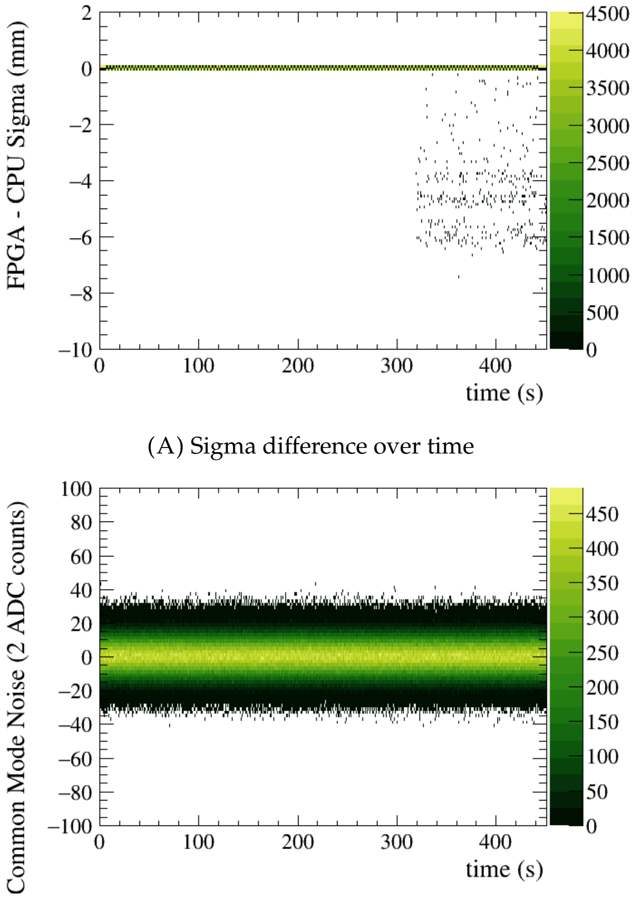
Although the exact cause of the 383-frame discrepancy remains unknown, both the test beam at HIT and the lab tests confirm that the reconstruction firmware operates correctly under normal and varied conditions. Moreover, the issue appears to be correlated with high common-mode noise and may be mitigated by further reducing such noise in future setups by the methods discussed in Chapter 5.

6.8.5 Timing Verification

The timing of the reconstruction firmware is verified by simulating the entire reconstruction chain using ModelSim, as shown in Fig. 6.14. The simulation confirms that the total processing time for one frame of data is 27.3 µs, which is well within the 100 µs time budget allocated for data processing.

Timing performance is further verified by examining the frame IDs saved in each frame header. As described in Section 4.4, frame-level synchronization is achieved using two frame ID counters: a global counter from the leader plane and a local counter for each frame. Both counters are incremented by one for every processed frame, and their values are saved in the header. By checking the continuity of these frame IDs, any frame loss can be easily detected. If no frame loss is observed, it indicates that the firmware is capable of processing data at the required readout rate of 10 kHz, and that HITDAQ is successfully storing all frames.

For all data collected at HIT in August 2024, only one frame loss event was observed. In that case, 4441 consecutive frames for all 6 boards were lost. As the loss



(A) Sigma difference over time

(B) Common mode noise over time

FIGURE 6.22: The abnormal events occurred in a burst from a single board during one run.

occurred all at once for all boards, it is most likely attributable to a software issue, since hardware issues typically manifest as one board failing. No other frame loss events were found in the remainder of the dataset. This result demonstrates that the reconstruction firmware is capable of sustaining real-time processing at the full readout rate.

6.9 Discussion

This chapter presents the detailed implementation of the reconstruction firmware on the MAX10 development board. The reconstruction chain features pipelined pedestal subtraction, calibration, and clustering for all channels, followed by a sequential RMS calculation. The complete firmware currently occupies 75% of the logic elements (LEs) on the MAX10 FPGA, with only 8.5% attributed to the reconstruction chain. The total processing time for one frame of data is 27.3 µs, which is well within the 100 µs time budget allocated for data processing.

Fixed-point arithmetic is employed throughout the reconstruction chain. The precision of the pedestal subtraction and calibration stages is significantly finer than the typical noise level, which is around 9 ADC counts. The RMS calculation yields a least significant bit (LSB) of 0.05 mm for both position and beam width, sufficient to meet the resolution requirement of 0.2 mm for beam monitoring.

The functionality of the reconstruction firmware is verified by comparing the results from the FPGA (fixed-point) with those computed by the CPU (floating-point). The verification confirms that all modules perform as intended and the firmware produces accurate and consistent results.

Two anomalies were observed during testing: 383 frames (0.887 per million) with mismatched σ values between the FPGA and CPU, and 4441 consecutive frames lost during data collection at HIT. The former shows high common-mode noise, while the latter appears to be a one-time software issue. Nonetheless, these isolated and not reproducible cases do not impact the overall performance or reliability of the firmware, as verified by extensive testing both in the lab and at HIT.

The reconstruction firmware is designed with flexibility in mind. Pedestal subtraction is performed automatically when measurement begins, and the calibration factors can be configured and updated via the HITDAQ software. Additionally, parameters such as cluster size and cluster threshold are fully configurable through HITDAQ.

The firmware is also designed to be modular. This allows for easy extension with additional features, such as more advanced clustering algorithms or alternative calculation methods. For example, the linear regression method introduced in Chapter 5 offers higher precision for position and beam width estimation compared to the RMS approach. Implementing linear regression requires not only addition, multiplication, division, and square root operations—as used in RMS—but also logarithmic functions.

The logarithm operation can be implemented using lookup tables stored in on-chip memory, where the input serves as the address and the output provides the corresponding logarithmic value. However, further investigation is required to assess the size and depth of memory needed for this implementation and whether it is feasible within the resource constraints of the MAX10 FPGA (1 638 kbits on-chip memory in total). All other operations in the linear regression method can be implemented sequentially, and their bit widths and LSBs can be determined in a similar manner as in the RMS calculation.

Based on the experience gained from implementing RMS on the MAX10 FPGA, the linear regression implementation is expected to be straightforward. While resource usage may be tight on the MAX10, a slightly larger FPGA with enough on-chip memory should be more than capable of supporting it.

The version of the reconstruction firmware documented in this thesis does not include the common-mode subtraction module. A version with this module has been prepared, and the common-mode subtraction occupies less than 1% of the logic elements (LEs).

Chapter 7

Measurements and Performance

This chapter presents the results of the test beam campaign conducted on 8 August 2024. On that day, three SciFi XY stations, comprising a total of six planes, were tested at HIT. Each plane was equipped with a 25 cm-wide, two-layer, glueless fiber plane and five photodiode arrays. The main objective of this test beam was to study the detector resolution over the range of energies and intensities available at HIT.

This chapter is organized as follows: Section 7.1 introduces the test beam setup and beam plan. Section 7.2 describes the per-channel calibration together with the associated uncertainty analysis. Section 7.3 examines the detector linearity and compares the beam dynamic range at HIT with that of the SciFi-BPM, providing a method to estimate the lowest detectable intensity under different beam energies and widths.

Finally, to assess the resolution of the detector, Section 7.4 presents the resolutions at a fixed spot, and Section 7.5 extends the evaluation to a 128 mm by 132 mm area.

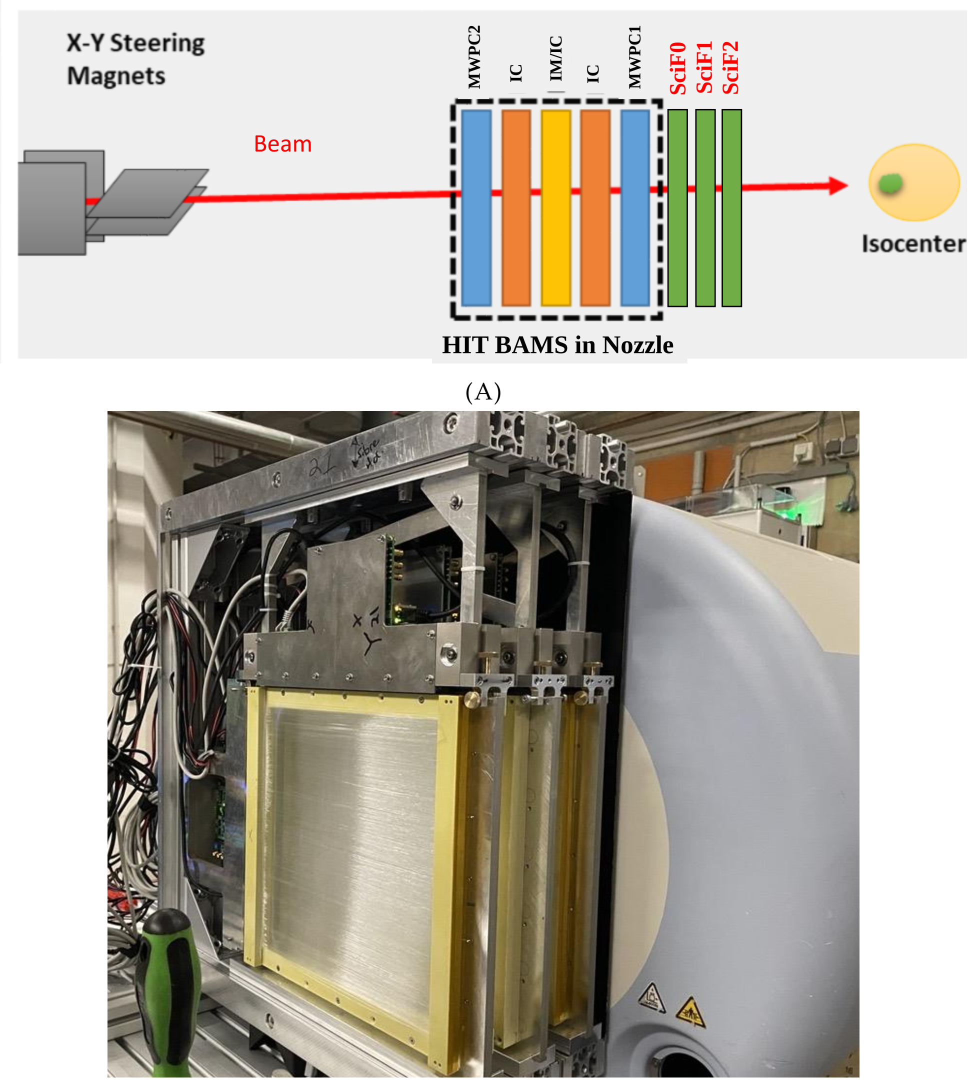
7.1 Test Beam at HIT

For the August 2024 test beam campaign, three XY stations were placed near the beam nozzle in the experimental cave at HIT. Proton and carbon beams covering a wide range of energies and intensities were measured to investigate the calibration procedure, test the linearity of the detector, and evaluate its resolution performance. All measurements were carried out using the setup and beam plans described in this section.

This section describes the experimental setup in Subsection 7.1.1. For convenience in the subsequent explanation and analysis, the naming convention of the detectors and the coordinate systems are also presented.

As introduced in Section 3.3.1, each type of beam has 15 300 combinations, defined by 255 energy steps, 10 intensity levels, and 6 beam size steps. It is not feasible to perform measurements for a single beam type covering all these 15 300 combinations. Therefore, in Subsection 7.1.3, typical beam scan plans are introduced, which sample representative energy and intensity steps to cover a wide range of conditions. In addition, spatial scans with fixed beam settings were performed over different detector positions and are also explained in this subsection.

Finally, in Subsection 7.1.4, the firmware version, the specific considerations regarding data uncertainty for the August 2024 test beam, as well as the unit of all signal amplitudes used in this chapter, are provided.



(A)

(B)

FIGURE 7.1: The layout for the test beam at HIT. (A) Sketch for the relative positions of the steering magnets, HIT BAMs, SciFi-BPMs and isocenter (B) Photograph for the nozzle and SciFi-BPMs.

7.1.1 Experimental Setup and Layout

Figure 7.1A shows the layout of the beamline for test beam. From upstream to downstream, the beamline consists of magnets for beam steering, the BAMs (beam application monitors) in the nozzle containing two MWPCs and two ICs for beam position and intensity monitoring respectively, and three SciFi XY stations positioned immediately downstream of the nozzle.

The beam profile along the beamline varies with distance due to multiple scattering and the intrinsic focusing or defocusing of the beam. Since the SciFi XY stations are intended to replace the MWPCs in the nozzle in the future, they are positioned close to the nozzle to obtain a beam profile similar to that at the nozzle.

The detector remained fixed throughout the entire experiment and was covered to shield it from surrounding light.

Order	Name	XY Station	photodiodes orientation	Common Mode Noise	Mirror
0	H0	SciFi0	horizontal	high	on
1	V0	SciFi0	vertical	low	on
2	H1	SciFi1	horizontal	low	on
3	V1	SciFi1	vertical	high	off
4	H2	SciFi2	horizontal	high	on
5	V2	SciFi2	vertical	high	on

TABLE 7.1: Naming convention of the SciFi-BPMs and their characteristics.

Tab. 7.1 lists the naming convention of the SciFi-BPMs. From upstream to downstream, the three XY stations are SciFi0, SciFi1, and SciFi2. The individual planes are denoted as H0, V0, H1, V1, H2, and V2, where “H” indicates that the photodiodes are oriented horizontally and measure the horizontal beam profile, and “V” indicates a vertical orientation for measuring the vertical beam profile.

The common mode noise has two possible states: "high", corresponding to the board 1 type described in Chapter 5 with $\rho \simeq 0.8$, and "low", corresponding to the board 0 type with $\rho \simeq 0$. Common mode subtraction is applied to the planes with high common mode noise during offline analysis.

All planes except V1 have mirrors on their fiber planes. Since V1 also exhibits high common-mode noise, its signal-to-noise ratio is the lowest among all planes. The definition of SNR is given in Eq. 5.27.

7.1.2 Coordinate System

Three Cartesian coordinate systems are used in the following analysis, each serving a different purpose.

The first coordinate system is defined in terms of the channel ID for each plane, as shown in Fig. 7.2. It indicates the positions of channel ID 0 and channel ID 319 for both the horizontal and vertical planes. This system is used for studying the calibration factors for each channel.

The second coordinate system is defined in millimeters for each plane. In this system, one channel corresponds to 0.8 mm with a 0.2 mm gap between photodiode arrays. The origin (0 mm) lies at the center of the middle photodiode array. The X and Y axes are oriented away from the Y and X photodiode arrays, respectively. Coordinates in this system are written as $(X_{\mathrm{H}}, Y_{\mathrm{V}})$. For example, X_{H0} (mm) denotes

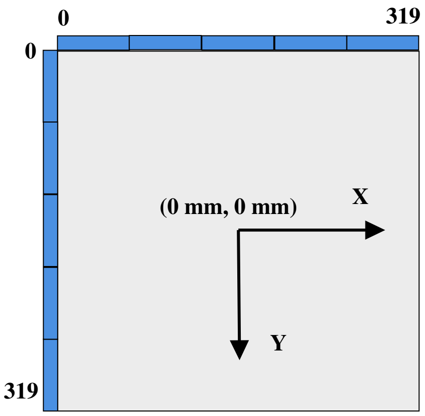


FIGURE 7.2: Sketch of the coordinate systems. The blue rectangles represent the 64-channel photodiode arrays, and the grey square represents one set of XY fiber planes.

the position from SciFi-H0, with zero at the center of the plane. This system is used for mapping the resolution in 2D in Section 7.5.

As shown in Fig. 7.1A, the three XY stations are located at different positions along the beamline, and the beam traverses them at an angle depending on the target position at the isocenter. Consequently, the reconstructed positions for the three stations differ. To compare the positions reconstructed from different planes, the positions measured by SciFi0 and SciFi2 are aligned to those of SciFi1 by translating them into SciFi1's coordinate system using the distances from the detectors to the magnets. The resulting positions are then used to compare differences between planes for resolution and bias study. In this system, coordinates are denoted as (X, Y), where X_0 represents the position measured by H0 and aligned to H1, and Y_0 represents the position measured by V0 and projected to V1.

7.1.3 Beam Plan

This section describes the beam plan, which is required both for delivering specific beams and for providing beam energy and intensity for further analysis, as accurate knowledge of these parameters is essential. With a well-designed beam plan, a large range of beam conditions or a wide spatial region of the detector can be covered within a short measurement time. Moreover, a carefully designed beam plan allows the energy and intensity of the beams to be determined using only the data collected from the SciFi-BPM together with the beam plan itself, thereby simplifying the data analysis.

In Section 3.3.1, the extensive library of pencil beams (LIBC) at HIT with a vast collection of beam settings was described. It contains four types of ion beams (proton, helium, carbon, and oxygen), each with 255 energy steps (E1 to E255), six beam width settings (F1 to F6), and ten intensity settings (I1 to I10).

To deliver a beam with specific settings to a given location, a beam request is submitted to the accelerator control center. The beam request is an `.xml` file, referred to as a beam plan file.

A beam plan file specifies parameters including the treatment room ID, ion type, energy setting, focus setting, spot position at the isocenter, and the number of ions or protons for each spot. It is organized into iso-energy slices (IES), where each slice is irradiated with a fixed energy and focus. The particle number for each lateral spot at the isocenter is defined within its respective IES.

The accelerator executes the beam plan on an IES-by-IES basis, as described in Section 3.3.3. For each IES, it delivers the prescribed number of particles to the designated spots according to the plan. Once an IES is completed, the accelerator prepares the next spill for the subsequent IES. Each IES is delivered in a separate spill, and a single IES may be delivered in one or multiple spills, depending on the number of required particles. As shown later in Fig. 7.3D, the beam energy changes spill by spill according to the IES settings. This makes it possible to assign the beam energy to each spill without the need for external detectors. As described in Subsection 3.3.2, the energy is determined by the synchrotron magnetic rigidity, which is highly precise, with magnetic field deviations better than 1×10^{-4} [43].

The intensity setting needs to be treated differently. It is not included in the beam plan file, as the facility is designed to deliver a precise number of particles rather than a specific intensity. Unless otherwise specified, the intensity is chosen automatically so that each spot is irradiated for at least 1 ms, as limited by the MWPC described in Section 3.4.1. When the intensity is specified manually (I1 to I10), the number of particles planned per spot must be sufficient to ensure a spot duration of at least 1 ms. Nevertheless, for some beams—mostly at low energies—the actual intensity does not strictly follow the specification.

Fig. 3.6 shows that the intensity fluctuates measured by ICs with 2 kHz readout rate, ranging by a factor of 2 to 6, but remains stable with 20 Hz readout rate. Since the accelerator can deliver the particle number with better than 1% precision [45] and the intensity remains stable over longer time intervals, the intensity of each IES is calculated as the number of particles specified in the beam plan divided by the beam-on time.

The beam-on time is determined using SciFi-V0 and SciFi-H1, as these two planes exhibit low common-mode noise. The probability of a false positive beam signal is very low for one plane (less than 10^{-8}), as explained in Section 5.4. The bin width for determining beam-on and beam-off times is 1 ms. While a finer binning is possible, it is not necessary, since each spill lasts typically several seconds.

Four types of beam plans were employed in this test beam, as shown in Fig. 7.3: line scan, box scan, intensity scan, and energy scan.

For the line scan and box scan plans, the energy, focus, and number of particles per spot are fixed. The spot positions form either a line or a rectangular grid with a fixed step size. Notably, the detector remained fixed throughout the measurements; scanning was performed by moving the pencil beam rather than the detector.

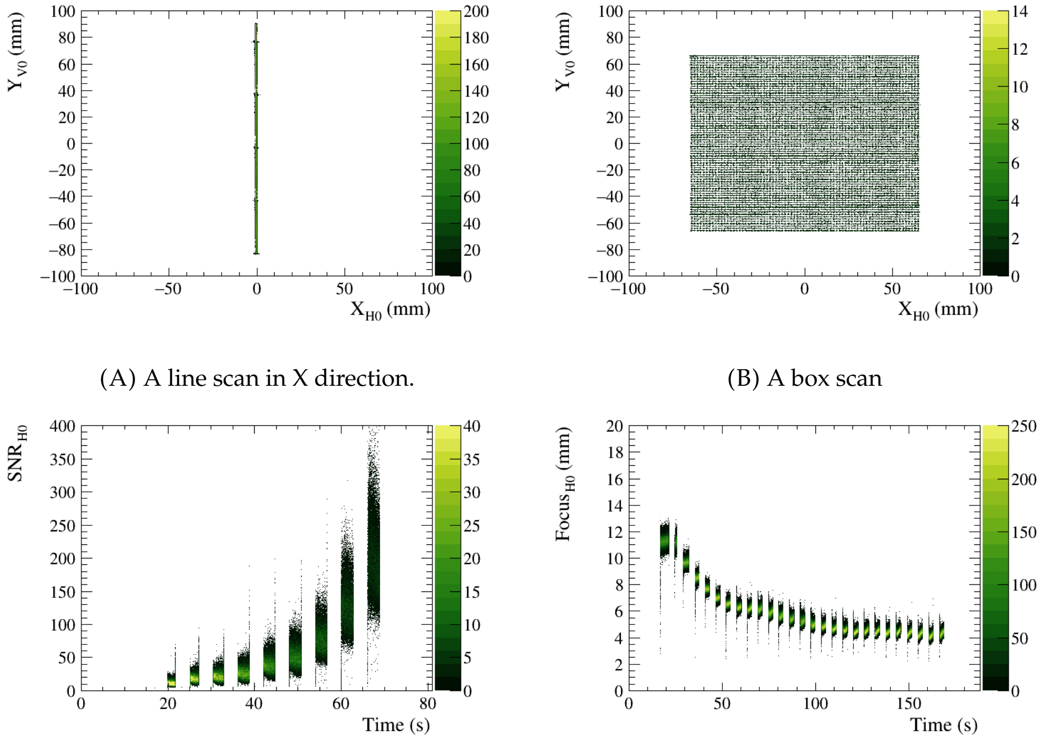
In Fig. 7.3A, the line scan delivers beams uniformly along the Y position in fixed steps, where the axes correspond to positions measured by H0 and V0 in the second coordinate system, i.e., SciFi0's own coordinate system. Line scans in both directions were performed at different Y/X positions, in order to study the detector response along the fiber planes for all vertical and horizontal planes in Section 7.2.

Two box scans were carried out: one smaller scan with a proton beam covering an area of 128 mm by 132 mm at SciFi0, and one larger scan with a carbon ion beam covering an area of 168 mm by 174 mm at SciFi1. These measurements are used to determine the two-dimensional resolution performance of SciFi-BPM in Section 7.5.

In both the line and box scans, the beam width was wider than the step size, producing a uniform dose region. This uniformity is essential for the per-channel response calibration described in Section 7.2.

For the intensity (energy) scans, the spot position remained fixed while the intensity (energy) was varied.

In an energy scan plan, the beam energy is increased from one IES to the next. Fig. 7.3D shows that the beam width decreases spill by spill, indicating increasing



(A) A line scan in X direction.

(B) A box scan

(C) An intensity scan.

(D) An energy scan.

FIGURE 7.3: Four types of beam plans for the test beam at HIT. Shown are 2D histograms of the reconstructed beam position, reconstructed beam width, or measured SNR per data frame for the corresponding beam plan.

beam energy, consistent with the energy dependence of beam width described in Fig. 3.2 for the HIT beam settings. The energy of each spill is therefore known from the beam plan within the precision of the synchrotron. Energy scans are used to study the linearity with respect to the specific energy loss in Section 7.3.5.

In an intensity scan plan, the particle number per spot is progressively increased from one IES to the next, causing the accelerator to automatically raise the intensity. Fig. 7.3C shows the gradual increase in SNR of H0 spill by spill, indicating rising beam intensity. It also reveals fluctuations in the intensity with 10 kHz readout rate.

TABLE 7.2: Settings for the energy and intensity scans.

	Proton		Carbon	
	Energy (MeV/u)	Intensity (10^6particles/s)	Energy (MeV/u)	Intensity (10^6particles/s)
LIBC range	48 – 221	80 – 3200	88 – 430	2 – 80
Intensity scan1	81	160 – 3030 (9 IES)	204	5 – 75 (8 IES)
Intensity scan2	169	160 – 3030 (9 IES)	430	5 – 75 (8 IES)
Energy scan	48 – 218 (26 IES)	1900 (low E: 760, 1420)	88 – 425 (26 IES)	47 (low E: 23, 37)

Tab. 7.2 summarizes the settings of the intensity and energy scans for proton and carbon beams, together with the corresponding ranges in the LIBC. The scan intensities listed in the table were determined from the measurements by dividing the number of particles in the beam plan by the beam-on time.

The intensity scans with 169 MeV/u protons were performed five times with different integration times to measure the linearity with respect to integration time and intensity (Sections 7.3.3 and 7.3.4). The remaining three intensity scans were performed once each. These four types of scans, with beams of different widths, are used in Section 7.4 to evaluate the position and width resolution as a function of intensity and beam width. The intensity range of the scans does not reach the lowest intensities available in the LIBC. This is addressed in Section 7.3, where the SNR of V1 is used to estimate the SNR of the other boards at lower intensities.

The energy scans, consisting of 26 energy steps for both protons and carbon ions, were each performed once. The intensities for each energy step were relatively constant, except for the two lowest energy steps in both the proton and carbon scans.The energy scans are used to measure the linearity with specific energy loss (Section 7.3.5).

Overall, the scans span a wide range of beam conditions and are used to comprehensively evaluate the linearity, dynamic range, and resolution performance of the SciFi-BPM.

7.1.4 Firmware Version and Data Acquisition Notes

The firmware used for this test beam campaign implemented the reconstruction chain described in Chapter 6, which includes pedestal subtraction, clustering, per-channel calibration, and RMS calculation. It did not include common-mode subtraction, which was applied only during offline analysis for planes affected by high common-mode noise.

The data format of each BPM in this firmware consists of the header containing the frame ID, the calibrated values, and the reconstruction results, which were stored for offline analysis. The raw ADC values were not included and, consequently, were neither stored nor available for offline analysis.

During the test beam, an initial per-channel calibration of the SciFi-BPMs was performed onsite. The resulting calibration factors were written to the FPGAs, as described in Subsection 6.6.5, and applied to all subsequent measurements.

It was later found that the preliminary calibration was not sufficiently precise. Therefore, in the offline analysis, this calibration was removed by dividing each channel by its preliminary calibration factor to restore the pre-calibration data. A more accurate calibration was then determined in Section 7.2 and applied in the subsequent analysis.

As explained in Chapter 6, the pedestal subtraction and the calibration performed on the FPGA introduces an uncertainty of less than 1 ADC count in the calibrated value, which is negligible compared to the typical uncorrelated noise level of 9 ADC counts. The restoration of the pre-calibration data is performed using floating-point calculations on the CPU. Consequently, the uncertainty of the restored data relative to the true data is negligible.

Since the least significant bit of the ADC is discarded during pedestal subtraction on the FPGA (16-bit unsigned ADC converted to a 15-bit magnitude plus one sign bit after subtraction), the unit for all signal amplitudes after pedestal subtraction is 2 ADC counts. For simplicity, the signal amplitude in this chapter is expressed in units of 2 ADC counts unless stated otherwise. For example, the uncorrelated noise is not 9 ADC counts, but 4.5.

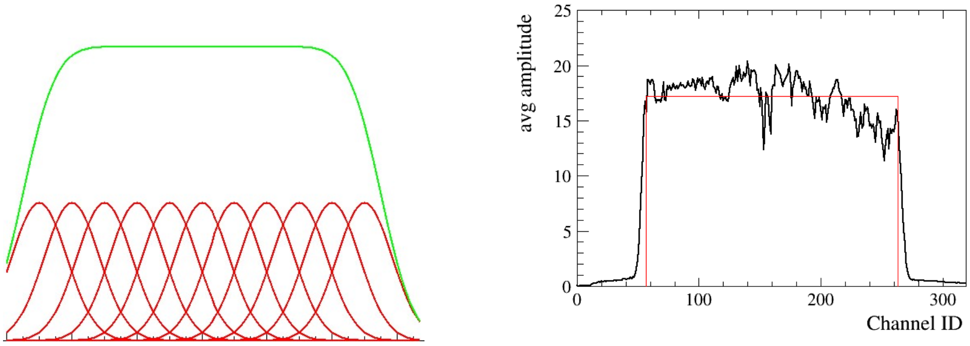
7.2 Calibration

For each SciFi-BPM plane, there are 320 channels. Variations in the number of fibers coupled to each channel, differences in light yield and fiber reflectivity, variations in the coupling efficiency between fibers and photodiodes, and gain variations of the photodiodes all contribute to the non-uniformity of the channel responses. Calibration is therefore essential to ensure a uniform ADC signal across all channels.

This section begins with the calibration method description in Subsection 7.2.1, followed by an example line scan used for calibration. Subsections 7.2.3 and 7.2.4 address the uncertainties arising from the detector measurements and from the beam used for calibration. The precision of the one-dimensional calibration for the two-dimensional fiber planes is then evaluated using line scans at different positions in Subsection 7.2.2. Finally, Subsection 7.2.6 presents the overall calibration uncertainty, and Subsection 7.2.7 discusses potential future improvements in calibration.

7.2.1 Methods

The idea of the per-channel calibration is to apply the same beam signal to all channels and measure their channel-dependent responses. To achieve this, the same beam is scanned across all channels to obtain a scan profile. Fig. 7.4 illustrates the calibration principle in (A) and an example of a measured scan profile in (B). The profile in (B) is the per-frame average of the summed beam profiles from a line scan; this type of plot is referred to as a scan profile. Ideally, a scan profile has a flat plateau



(A) The sum (green) of the beams (red) forms a flat plateau.

(B) A scan profile of SciFi-H1 from a line scan with proton beams.

FIGURE 7.4: principle of calibration. The red lines in (b) indicate the boundaries and the average signal $\overline{H}$ of the plateau region, which spans 206 channels.

region, as shown in (A). In practice, the measured scan profile from a real scan exhibits variations between channels. These variations reflect the detector properties and are used to determine the calibration factor for each channel.

The calibration factor f_i for channel i can be expressed as the ratio of the average height of the scan profile in the plateau region, $\overline{H}$, to the scan profile height of that channel, H_i:

$$f_i = \frac{\overline{H}}{H_i}. \tag{7.1}$$

The calibration factor f_i is then multiplied by the signal y_i to obtain the calibrated signal, as shown in Eq. 5.29. The essence of this calibration method is to use a scan profile as a standard ruler for measuring other beams. It is, therefore, a relative measurement.

The uncertainty of the calibration factor depends, on the one hand, on the precision with which the detector determines the H_i values, and on the other hand, on the uniformity of the beam scan. In addition, since this per-channel calibration is one-dimensional, variations in the fiber plane along the second dimension may introduce further uncertainties. The following sections present an example scan and then discuss the sources of uncertainty in the calibration factor arising from the detector (H_i), the beam scan uniformity, and considerations related to two-dimensional calibration.

7.2.2 Line Scan

To form a plateau region across channels, the scan can be either line scans or box scans. A box scan delivers the beam in two dimensions, painting over the fiber planes as Fig. 7.3B, whereas a line scan delivers the beam in only one dimension as Fig. 7.3A, with a fixed X or Y position. In both cases, the scan should deliver a high-intensity beam in fine and constant steps between spots, in order to form a high and flat profile.

As an example, a X line scan was performed at a vertical position of $Y = 0\,\mathrm{mm}$ using a 221 MeV proton beam at an intensity of $2 \times 10^9\,\mathrm{s}^{-1}$ (setting I9 E255), as shown in Fig. 7.5. The beam width was 4.5 mm at H1, with an amplitude of approximately 700 (unit: 2 ADC counts). The scan was performed in 0.086 mm steps at H1, covering

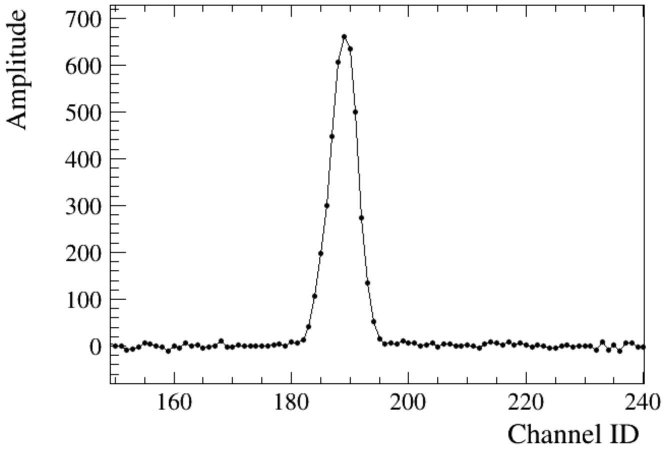


FIGURE 7.5: A frame data measured by SciFi-H1 from a scan.

a horizontal range from −86 mm to 86 mm in a total of 2001 steps. At the isocenter, this corresponds to a step size of 0.1 mm, covering a range from −100 mm to 100 mm.

At each spot, approximately 2×10^7 protons were delivered in 10 ms. The entire scan lasted 100 s and was completed in 10 spills.

The average signal per frame obtained from this scan at SciFi-H1 is shown in Fig. 7.4B. The average amplitude in the plateau region, $\overline{H}$, was 17.23 in units of 2 ADC counts. Since it was averaged over 4.21×10^5 frames, the statistical uncertainty of H_i is $\frac{\sqrt{17.23}}{\sqrt{4.21 \times 10^5}} \simeq 0.006$, which is a 0.03% uncertainty relative to 17.23. This uncertainty is therefore negligible.

7.2.3 Uncertainty in H_i

Assuming that the pencil beam delivery system delivers the beam uniformly, the uncertainty of the calibration factor f_i is determined by the uncertainty of H_i, the average amplitude per frame. Since the statistical uncertainty of H_i is negligible, it is dominated by the uncertainty of the pedestal subtraction.

As described in the FPGA chapter on pedestal subtraction (Section 6.6.2), the least significant bit (LSB) of the pedestal-subtracted value is 2 ADC counts. Therefore, any pedestal smaller than 1 ADC count is not subtracted and remains in the data streamed from the FPGA. If the FPGA output is summed directly, this leads to an uncertainty in the calibration factor of approx $1/17.23 \approx 5.8\%$ for the line scan described in Section 7.2.2.

The preliminary calibration factor applied during the test beam, as mentioned in Subsection 7.1.4, was determined using the FPGA pedestal-subtracted values directly and thus contains this 5.8% uncertainty.

Additional pedestal subtraction should be performed in the offline analysis to reduce this uncertainty.

To measure the pedestal precisely, two steps are taken. First, all scans were kept relatively short (500 seconds in the longest case for a large box scan) to limit the effect of pedestal drift over time (Fig. 5.4). Second, all beam-off frames are divided into two groups: group A and group B, as shown in Fig. 7.6. Group A is used to determine the pedestal, while group B, after subtraction of the group A pedestal, is used to estimate the uncertainty of the pedestal for the beam-on frames, which is the uncertainty in H_i.

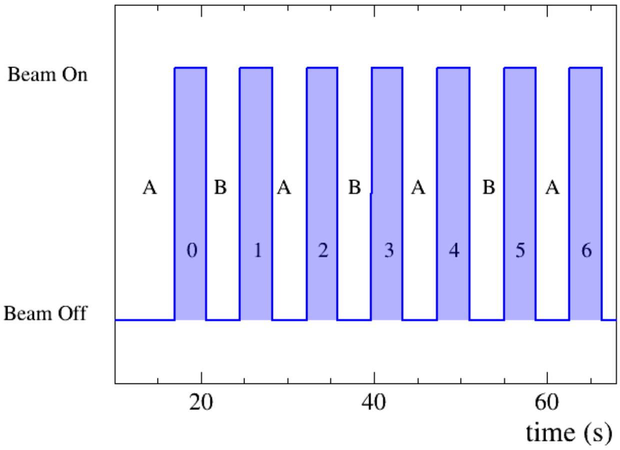


FIGURE 7.6: Beam-on and beam-off periods over time for part of a measurement. The beam-off frames are separated into two groups: group A and group B.

Groups A and B are separated in an interleaved manner, as shown in Fig. 7.6. This ensures that the pedestal in both groups closely represents the pedestal during the beam-on frames, and that the difference between the two groups reflects the uncertainty of the pedestal measurement for the beam-on frames.

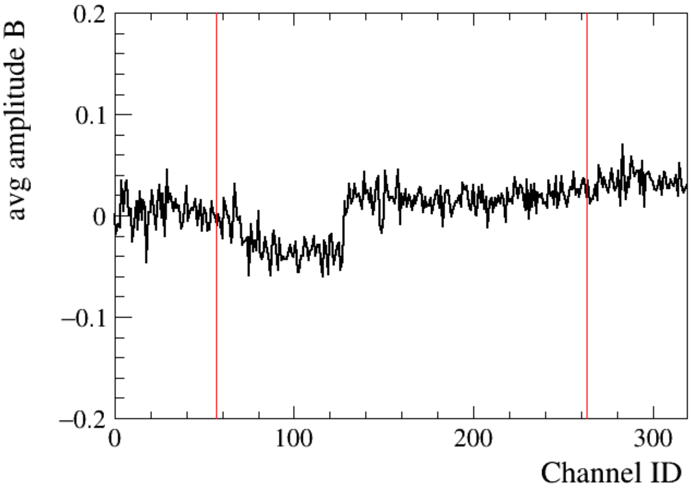


FIGURE 7.7: Average amplitude per frame for group B after pedestal subtraction using group A for SciFi-H1 of the line scan in Fig. 7.4B, with the corresponding H_i for beam on frames shown in Fig. 7.4B. The red lines indicate the boundaries of the plateau region.

Fig. 7.7 shows the residual of group B after pedestal subtraction of group A for the line scan in Fig. 7.4B. The sharp jump in the pedestal values between channels occurs at the boundary between the second and third photodiode arrays. This array-dependent behavior is likely caused by slight variations of the reference voltages between arrays.

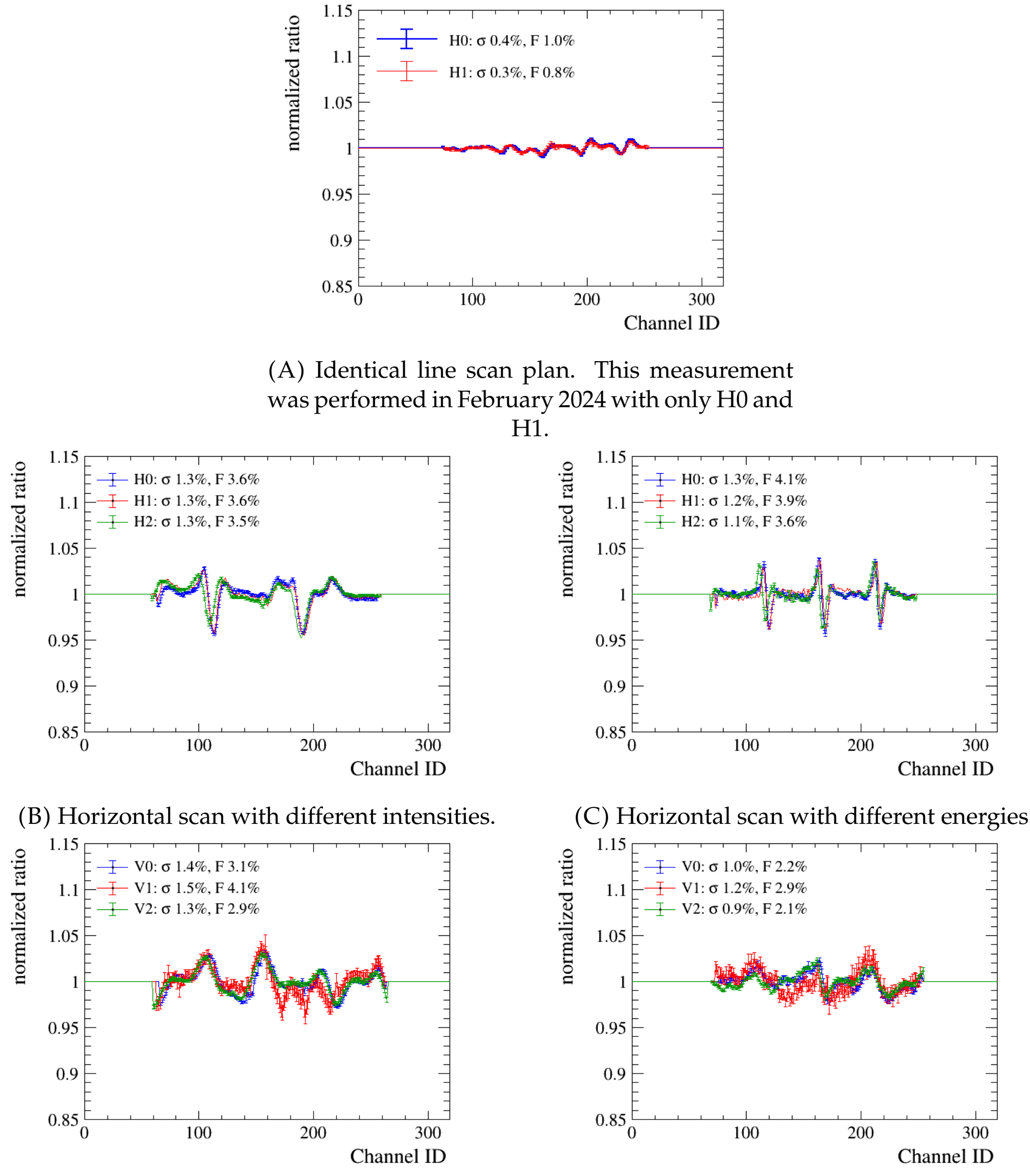
The standard deviation of the residual in the plateau region is 0.01. This is the uncertainty of H_i for Fig. 7.4B. H_i varies and is calculated for each measurement using the standard deviation of the residuals of group B in the plateau region.

The minimum, maximum, and mean values of H_i in the plateau region in Fig. 7.4B are 11.25, 20.36, and 17.23, respectively. This corresponds to relative uncertainties from H_i related to the pedestal subtraction of 0.09%, 0.05%, and 0.06% for the worst, best, and mean cases, respectively.

The same line scan was also performed in the Y direction to obtain the scan profile for the vertical planes. Among all channels in the plateau region of all six planes, the largest relative standard deviation in H_i is 0.27%. This implies that, assuming the beam is perfectly uniform, the overall uncertainty of the calibration factor H_i is at most 0.27%, which is sufficient for measuring the 10% relative variations between channels.

7.2.4 Uncertainty Due to Scan Uniformity

This section evaluates the scan uniformity by comparing the scan profiles from different line scans taken at the same positions of the detector.



(A) Identical line scan plan. This measurement was performed in February 2024 with only H0 and H1.

(B) Horizontal scan with different intensities.

(C) Horizontal scan with different energies.

(D) Vertical scan with different intensities.

(E) Vertical scan with different energies.

FIGURE 7.8: Normalized ratio of scan profiles from different line scan measurements. The legend labels indicate the SciFi plane name, the relative standard deviation σ, and the flatness F.

Fig. 7.8 shows the normalized ratio of scan profiles, defined as the channel-by-channel ratio of two profiles. The values within the plateau region are normalized, while those outside the plateau are set to 1.0. This ratio can be interpreted as using one scan to calibrate the other, providing an indication of the beam scan uniformity and calibration uncertainty. The uncertainties of H_i from both the numerator and denominator scans are included in the plots. The x-axis for each curve represents the channel ID of the SciFi plane labelled with that curve.

The legend in the plots provides the relative standard deviation σ and the flatness F for each curve. Both quantities indicate the variation in uniformity between two scan measurements.

The relative standard deviation σ of the normalized ratio is defined as the standard deviation of the ratio in the plateau region divided by its mean, which equals 1 because the ratio is normalized. This represents the mean variation and is taken as the uncertainty of the calibration.

The flatness F of the dose distribution in radiation therapy is defined as

$$F = \frac{\mathrm{D_{max}} - \mathrm{D_{min}}}{\mathrm{D_{max}} + \mathrm{D_{min}}}, \tag{7.2}$$

where $\mathrm{D_{max}}$ and $\mathrm{D_{min}}$ are the maximum and minimum dose values in the plateau region, respectively. This parameter quantifies the extreme variation of the dose distribution. The flatness requirement for the beam at the isocenter for patient treatment is less than 5 %. Since, for a fixed beam energy, the dose is proportional to the number of particles, the flatness in Fig. 7.8 is calculated using the maximum and minimum of the normalized ratio.

In Fig. 7.8A, the two scan profiles compared were from two measurements with identical line scan plan. The relative standard deviation is 0.3 % for H0 and 0.4 % for H1. The variations in H0 and H1 are highly correlated, indicating that the scan uniformity uncertainty is the dominant contribution, with a value of approximately 0.4 %.

Figures 7.8B and 7.8D compare two horizontal line scans and two vertical line scans, respectively, with different beam intensities: one with $2.0 \times 10^9\ \mathrm{s^{-1}}$ protons and the other with $3.2 \times 10^9\ \mathrm{s^{-1}}$ protons. Figures 7.8C and 7.8E compare two scans with different energies: one with 95.3 MeV proton beams and the other with 221 MeV proton beams.

In these four plots, for all planes except V1, the uncertainty is dominated by the scan uniformity. For V1, which is the only plane without a mirror on the fiber plane, the low value of H_i leads to a significant contribution from its uncertainty, as the error bar in Fig. 7.8 for V1 is comparable to the variation of the scan.

These plots show that the beam uniformity uncertainty is approximately 1.3% with a flatness of about 4%. Depending on the beam settings, the beam exhibits some distinct structures.

One plausible explanation for the pronounced structures in Figures 7.8 B–E is systematic inaccuracies in the position provided by the MWPCs. As described in Section 3.3.3, the beam delivery process is monitored and controlled by the MWPCs and ICs. The beam position measured by the MWPCs is fed back to the steering magnets for position correction. The structures observed in Figures 7.8 can therefore be attributed to systematic errors in MWPC position reconstruction, with any such inaccuracies being directly reflected in the calibrated scan profiles.

The worst flatness in these plots is 4.1 %. Notably, this measurement is performed by the SciFi-BPM near the nozzle. At the isocenter, where the patient is located, the beam flatness is typically better, as multiple scattering smears out this variations.

This section shows that the consistency of the same scan is approximately 0.4%, while the consistency between different scans is about 1.3%. Since the calibration of the SciFi-BPM uses one scan as the reference for all beam types, the uncertainty of the one-dimensional (1D) calibration—applicable only to the current line scan position—is dominated by the scan uniformity and is of the order of 1.3 %.

7.2.5 Two-Dimensional Response

The channel response along the fiber direction is not constant. It depends on the quality of the fiber alignment—curved or missing fibers can lead to differences—as well as on the distance to the photodiode, due to light attenuation. The farther from the photodiode, the lower the signal. In addition, radiation damage can create dark spots and alter the response. However, the study of radiation damage is beyond the scope of this thesis. All boards used in the test beam were equipped with new fiber planes and electronics that had not been exposed to radiation.

To study the channel response along the fiber direction, line scans at different positions, with beams of the same energy, focus, and intensity settings, were performed by moving the beam rather than the detector. A total of ten line scans were carried out: five horizontal and five vertical, performed at vertical or horizontal positions of -100, -50, 0, $+50$, and $+100$ mm at the isocenter, corresponding to approximately -90 to $+90$ mm in the fiber plane coordinate.

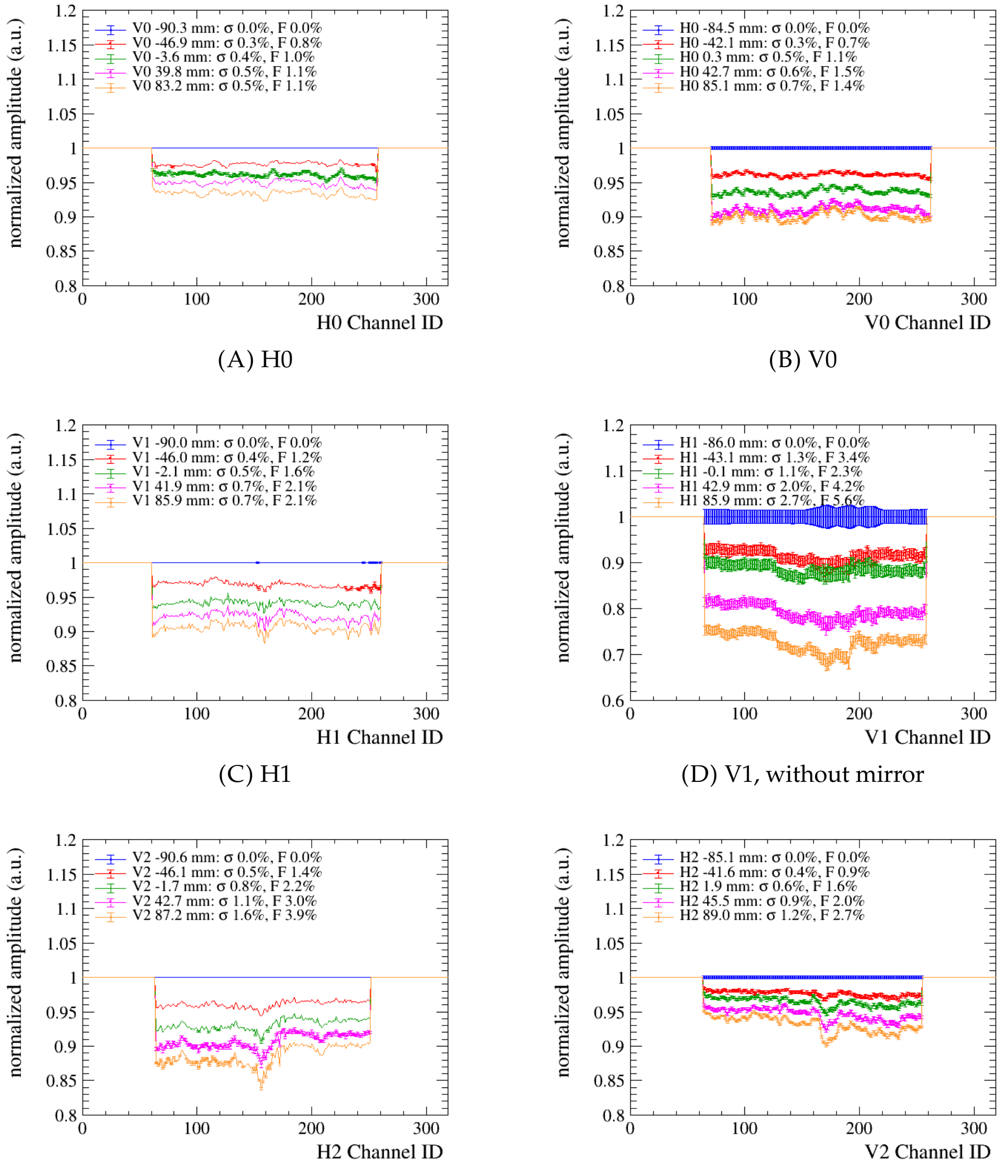
Fig. 7.9 shows the scan profile amplitudes at different positions, each normalized to a reference scan profile. For each plane, five line scans are displayed. The top curve in each plot corresponds to the reference scan, which is taken at the position closest to the photodiode arrays and therefore exhibits the highest amplitude. The gradual decrease of the other curves with increasing distance illustrates the attenuation effect along the fiber plane. In addition, some structures developing with increasing distance may be related to imperfections in the fiber alignment.

The position of each line scan is expressed by the position of the corresponding orthogonal plane in the same SciFi XY station. For example, in Fig. 7.9A, the five line scans correspond to H0 at different Y positions: -90.3, -46.9, -3.6, 39.8, and 83.2 mm on V0, moving progressively farther from the photodiodes. Since V0 and H0 are located very close to each other, these values can be directly taken as the Y positions on the H0 plane.

The uncertainty bars in these plots originate from the uncertainty of H_i. Since the signal amplitude of V1 is low, the uncertainty on H_i is correspondingly high, which explains the relatively large uncertainty bars for this plane. Note that the y-axis scale for V1 differs from that of the other five planes.

The mirrors at the ends of the fiber planes reduce the effect of light attenuation along the fibers. The mean amplitude as a function of distance relative to the photodiode arrays is shown in Fig. 7.10. Without mirrors (V1), the amplitude after 180 mm of fiber decreases to 72.7%, whereas with mirrors it decreases only to about 90%. This occurs because the reflected light from the mirror contributes additional signal, which increases with distance from the photodiodes, thereby mitigating the attenuation effect. V1 was intentionally left without a mirror in order to study this effect. For practical clinical use, however, all planes will be equipped with mirrors.

A closer inspection of Fig. 7.10 shows that H0, V0, and H1 exhibit stronger attenuation than H2 and V2. This could be related to the coupling of the fiber planes with the mirror. If the fiber plane and the mirror plane are not perfectly perpendicular, less reflected light will contribute to the signal. Such misalignment could originate



(A) H0

(B) V0

(C) H1

(D) V1, without mirror

(E) H2

(F) V2

FIGURE 7.9: The amplitude of line scan profile at different positions for six planes. The legend labels indicate the position coordinate, the relative standard deviation and the flatness.

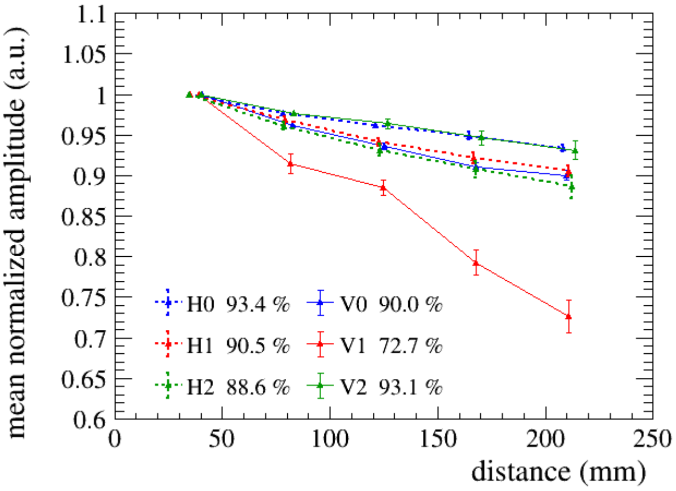


FIGURE 7.10: Mean normalized amplitude of line scans as a function of distance to photodiode arrays. The mirrors are located at 250 mm.

from the optical milling process at the end of the fiber plane. However, these effects result in only about a 3% reduction in the amplitude at the furthest point in Fig. 7.10 and can therefore be considered negligible for the beam position and focus reconstruction.

As shown in the previous section, identical line scans at the same detector position vary by about 0.4%, while line scans with different energy or intensity settings vary by about 1.3%. The variation of line scans at different detector positions includes contributions from both the beam and the detector, with the beam contribution remaining unclear. These two effects are difficult to separate precisely for an individual plane unless the detector itself is moved instead of the beam. However, by comparing two planes with some prior knowledge, it is possible to estimate the origin of certain variation structures.

In Fig. 7.9, for all planes, both the relative standard deviation σ and the flatness F increase as the line-scan position moves farther away from the reference scan. This indicates that the variation develops with increasing distance from the reference scan. Such variation can originate either from the beam or from the detector itself.

For H0, the variation remains small, with $\sigma = 0.5\%$ for the furthest scan, making it difficult to determine whether the source is the beam or the detector. In Fig. 7.8A, the σ for two line scans with identical settings is 0.4%. This indicates that the variation from the beam must be at least of the same order as that of the detector.

H0 and H2 were irradiated by the same line scans, so variation caused by the beam would appear in both planes. Compared with H0, H2 exhibits a clear valley structure in the middle of the plane that becomes more pronounced with distance from the photodiodes. In addition, the left half of H2 shows lower amplitude than the right half. Since these structures are absent in H0, they must originate from the detector.

As described in Section 4.2, the winding of fibers onto the winding hub is first performed on one half, after which the hub is rotated by 180° to wind the second half. This procedure may explain the structural differences observed in H2, namely the amplitude asymmetry between the left and right halves as well as the valley in between. These differences between the two halves are also observed in H1 compared with H0, and in V1 and V2 compared with V0. The resulting bias and resolution deterioration will be discussed in Section 7.5, where the two-dimensional distributions of bias and resolution are presented.

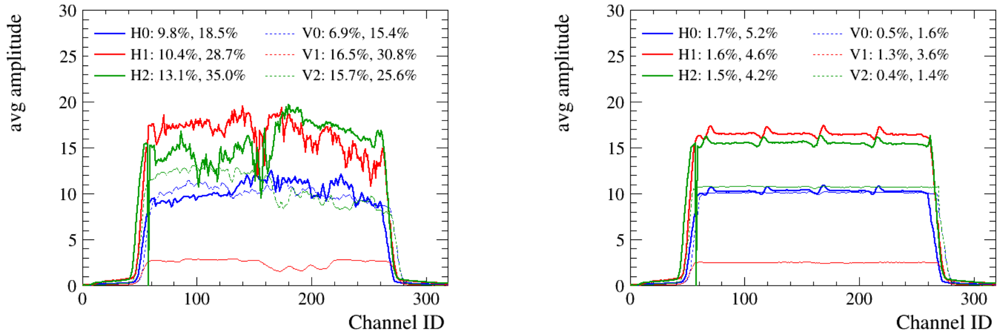
The calibration applied in this thesis is a one-dimensional, per-channel calibration. The same calibration factor is used for a channel regardless of whether the beam is close to or far from the photodiode along the fiber direction. Variations along the fiber, caused by detector-related effects, therefore introduce an additional contribution to the overall calibration uncertainty.

The variation along the fiber is relatively small. The worst case is observed in V1, with a flatness of 5.6%; however, V1 will not be used for clinical scenarios in this configuration, since it lacks a mirror. Among the mirrored planes, H2 shows the largest variation, with a relative standard deviation of $\sigma = 1.6\%$ and a flatness of 3.9% between the two furthest scans. Notably, these values include contributions from both the beam and the detector.

These fiber planes are early prototypes of the large-area glueless fiber planes. With practice, the engineers are becoming increasingly familiar with the fabrication process, and therefore future mats are anticipated to demonstrate superior performance.

7.2.6 Combined Uncertainty

Section 7.2.4 provides the uncertainty of the 1D calibration at a fixed line-scan position, while Section 7.2.5 provides the uncertainty along the fiber direction. To combine these two contributions, a line scan in the middle of the fiber plane is used to calibrate a wide box scan. Fig. 7.11 shows the box scan profiles before and after calibration for all six planes, with the relative standard deviation and flatness in the plateau regions given in the legend.



(A) Before calibration

(B) After calibration

FIGURE 7.11: Scan profiles of the six planes from a box scan. (A) before calibration (B) calibrated using line scans.

If all planes were identical, the same scan would produce scan profiles of equal height, since the same beam scan should generate the same amount of signal, resulting in the same plateau height provided that the plateau widths are similar. The plateau widths are indeed similar for all six planes. However, the amplitudes of the scan profiles for H1 and H2 are higher than those for H0, V0, and V2. This indicates that the coupling between the fiber planes and the photodiode arrays is better for H1 and H2 compared with H0, V0, and V2. In addition, V1 shows the lowest amplitude, as it is the only plane without a mirror and therefore collects less signal. This highlights the importance of improving the coupling between the photodiode arrays and the fiber planes by enhancing the optical alignment and improving the milling quality.

For H1, the calibration reduces the flatness from 28.7% to 4.6% and the relative standard deviation σ from 10.4% to 1.6%. This significantly improves the channel-to-channel uniformity. The remaining 1.6% relative uncertainty and 4.6% flatness include contributions from the uncertainty of H_i, the scan uniformity, and the two-dimensional response variation along the fiber direction. Thus, the uncertainty of the one-dimensional (1D) calibration—applied to the entire plane—is of the order of 1.6 % for H1. It is dominated by the scan uniformity, as in the post-calibration scan profile the structures caused by the beam, discussed in Section 7.2.4, are still prominent. Similar numbers can be obtained from Fig. 7.11, where all six planes show a consistent improvement in both flatness and relative uncertainty after calibration.

7.2.7 Discussion

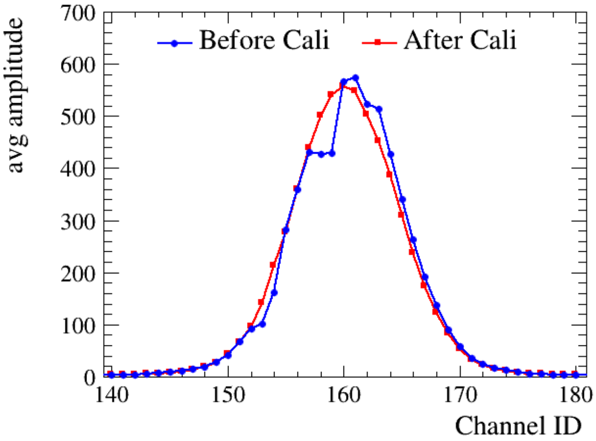


FIGURE 7.12: Beam profile before and after calibration for a 95.3 MeV proton beam at $2 \times 10^9\ \mathrm{s}^{-1}$ with FWHM = 8.4 mm, measured by H1.

Fig. 7.12 shows the averaged profiles of a proton beam before and after calibration, measured by H1. This beam is centered on H1, where the response fluctuations are most pronounced between channels. After calibration, the profile becomes smoother and more Gaussian-like, demonstrating the effectiveness of the calibration in equalizing the channel responses.

The uncertainty and precision of the calibration methods are limited by the beam uniformity, which in this case is determined by the MWPCs and ICs at HIT. To calibrate the SciFi-BPM independently of HIT, new calibration methods must be developed.

The essence of the calibration is to obtain the scan profiles. Such profiles can also be generated by irradiating the detector with a strip UV light source. A broad UV source provides a strong, wide, and flat signal that covers all 320 channels. The amplitude of a UV-induced scan profile can be up to 2000 times higher than that of a beam scan profile, limited only by the saturation of the photodiodes. This approach delivers scan profiles in real time—each frame corresponds to one complete scan profile—without the need to average over a Gaussian beam scanned across the detector. It has already been used in the laboratory for aligning the fiber planes with the photodiode arrays. UV light therefore has the potential to serve as a calibration standard for the SciFi-BPM detector. Further research should be carried out to compare UV-based scan profiles with beam-based scan profiles.

7.3 Dynamic Range and Linearity

This section examines whether the dynamic range of the beam in the LIBC at HIT matches that of the SciFi-BPM, i.e., whether the detector produces a sufficiently high SNR to achieve adequate spatial resolution while remaining within its operating range across the entire beam dynamic range at HIT.

Towards low intensities, the detector's dynamic range is limited by the SNR. The lowest detectable intensity was determined for a 4 mm-wide 169 MeV proton beam, for which the SNR remains greater than 10. Based on the observed linearity of the detector, the lowest detectable intensity at a fixed integration time determined for a specific beam can be extrapolated to other beam settings.

At high intensities, the detector's dynamic range is limited by the saturation of the photodiodes. As will be shown in Section 7.3.6, saturation is not a problem within the beam range available at HIT.

This section is structured to primarily present the linearity of the detector, during which the answer to whether the beam dynamic range matches that of the detector gradually becomes clear.

7.3.1 Linearity Equation

The linearity of the detector can be expressed as

$$A \propto \mathrm{SNR} \cdot \mathrm{FWHM} \propto \text{Intensity} \cdot \text{Energy loss} \cdot \text{Integration time}, \tag{7.3}$$

where the integrated amplitude A is proportional to the product of the SNR and the beam width (FWHM). Since A can be calculated as the product of the peak amplitude and the width, as shown in Eq. 5.37, and the SNR is proportional to the peak amplitude (see Section 7.3.2), this explains the first part of the relation. For the second part, this follows from the signal production process, which should be proportional to the beam intensity, its specific energy loss, and the integration time.

The linearity is investigated using the intensity and energy scans listed in Tab. 7.2. With the intensity scans, where the energy and beam width are fixed, the linearity of SNR with respect to intensity and integration time is shown in Section 7.3.3 and 7.3.4. With the energy scans, the linearity between A and the specific energy loss is demonstrated in Section 7.3.5, using A instead of the SNR. This is because both the FWHM and the energy loss depend on the beam energy and change simultaneously, so the SNR is not linear with the specific energy loss in energy scans.

7.3.2 SNR Concept

In this work, the signal-to-noise ratio (SNR) is defined in Eq. 5.27 as the ratio of the peak amplitude to the uncorrelated noise σ_η:

$$\mathrm{SNR} = \frac{\text{Peak Amplitude}}{\sigma_\eta}$$

This definition excludes the common-mode noise, which originates from voltage instabilities on the ADC boards. Since the common-mode noise does not significantly affect the position resolution (see Fig. 5.23), and because σ_η is similar across all boards (around 4.5, unit: 2 ADC counts), this definition provides a more meaningful measure of the signal response.

The peak amplitude refers to the maximum amplitude of the beam profile after pedestal subtraction. It is not calibrated. Since the noise is comparable across all channels, applying calibration would also rescale the noise channel by channel, making it non-uniform and thereby distorting its distribution. Therefore, it is more reasonable to compare the uncalibrated signal with the uncalibrated noise, and thus the peak used for the SNR is not calibrated. This is the same reason why the clustering is performed before the calibration.

With this definition, noise frames yield a SNR of around 3. This occurs because the maximum noise amplitude is taken as the peak amplitude, and with 320 channels, the probability is high that the largest fluctuation will reach approximately three times the noise level.

The SNR is a crucial parameter of the detector. It directly determines the performance in terms of both the interlock rate and the resolution. As shown by the simulation results in Chapter 5, a higher SNR leads to improved performance. For an SNR larger than 10, the interlock rate for a board with $\rho = 0.30$ common-mode noise is less than 0.04 per day. This means that, for all boards (with some boards having $\rho \simeq 0.30$ with common-mode subtraction and others $\rho \simeq 0$), the probability of producing three consecutive wrong results is less than 0.04 per day. Furthermore, for a beam width of 5 mm, if the SNR exceeds 15, the position resolution is better than 0.2 mm, and if it exceeds 20, the focus resolution is better than 0.4 mm.

It is reasonable to define the lowest detectable intensity as the level at which the SNR exceeds 10.

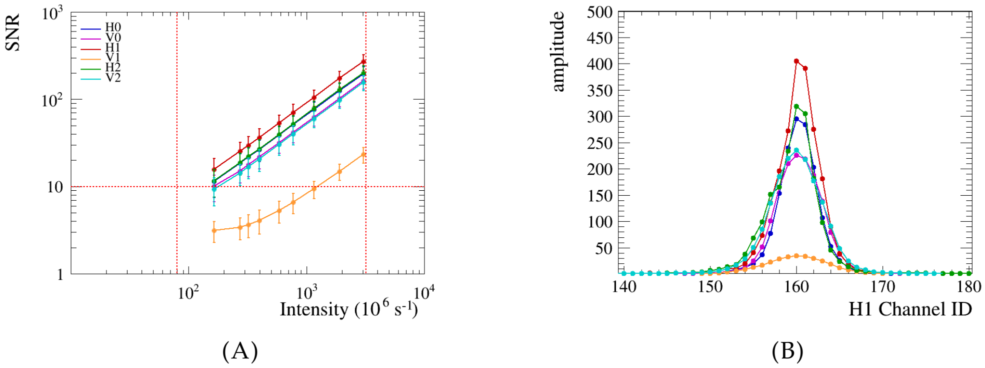
7.3.3 Linearity with Intensity

Fig. 7.3C shows the 2D histogram of the SNR of H0 as a function of time for an intensity scan. It was measured at a readout rate of 10 kHz with an integration time of 94.4 µs. A 169 MeV, at H1 4 mm-wide proton beam (LIBC setting: E174 F1) was irradiated at the center of the fiber planes, with the peak position located around channel ID 160 for all six planes.

The intensity scan consists of nine IES settings, corresponding to the nine spills shown in Fig. 7.3C. The beam intensity increases from spill to spill. The average intensity of each spill was estimated from the number of particles in the beam plan for that IES, together with the beam-on time of the spill. The resulting average intensities range from $160 \times 10^8\,\mathrm{s}^{-1}$ to $3.03 \times 10^9\,\mathrm{s}^{-1}$, as listed in Tab. 7.1.

Fig. 7.13A shows the mean SNR as a function of the average measured intensity for each spill. The vertical error bars represent the standard deviation of the SNR within each spill, reflecting fluctuations in the beam intensity rather than the uncertainty of the mean SNR. The vertical red lines indicate the LIBC proton beam intensity range. The measured intensity range does not extend to the lowest LIBC intensity setting. The horizontal red line marks SNR = 10. H1 maintains an SNR above 10 even at the lowest measured point, corresponding to fewer than 0.04 interlocks per day and ensuring good spatial resolution.

Fig. 7.13B shows the average beam profiles of the nine different intensities of the six planes, where the x-axis is the channel ID of H1. For H1, the peak is exactly at channel ID 160, while for the other five planes the peak is close to 160 but not



(A)

(B)

FIGURE 7.13: Results of the intensity scan with a 169 MeV proton beam, measured with an integration time of 94.4 µs. (A) SNR vs measured intensity. (B) Uncalibrated average beam profiles over the intensities.

perfectly aligned. For better comparison, all profiles were shifted so that their peaks align at channel ID 160.

From both the SNR–intensity curves and the average beam profiles, H1 shows the highest amplitude and SNR, while V1 performs the worst, as expected for the plane without a mirror. For H1, the mean SNR reaches 269.4 at the highest-intensity spill and 15.6 at the lowest-intensity spill. The SNR increases linearly with intensity for all data points in Fig. 7.13A, except for V1 at low intensity, where the SNR remains around 3—indicating essentially no visible signal.

The intensity scan was performed at the center of the fiber planes. As shown in Fig. 7.11A, for H1 and H2 the middle channels exhibit a reduced amplitude in the scan profiles. As explained in Section 7.2.5, this effect arises from the fiber winding process: the left and right halves were wound in two separate steps, which introduced a slight misalignment in the middle region. Consequently, the SNR at other positions is expected to be higher.

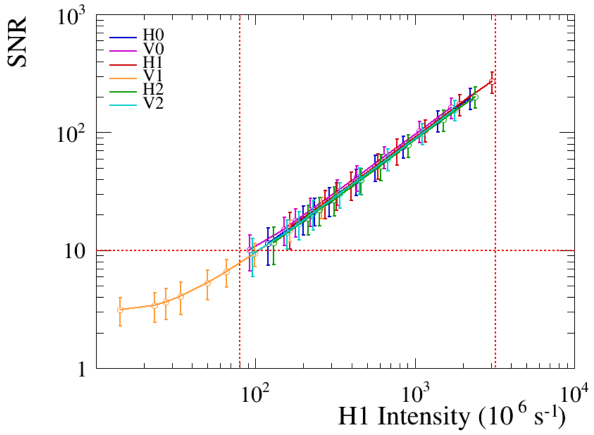


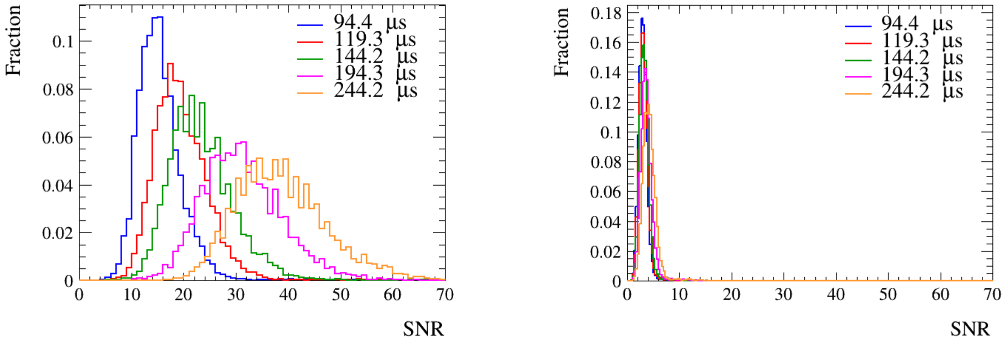
FIGURE 7.14: SNR with scaled intensity for 4 mm-wide 169 MeV proton beams, assuming H1-like response for all boards. The x-axis is labelled as "H1 Intensity".

The lowest LIBC proton intensity setting of $80 \times 10^6\,\mathrm{s}^{-1}$ was not included in this scan. To extend the intensity range to lower values, assuming that all boards has same SNR for same intensity as H1, the intensity for each plane can be scaled by the ratio of the peak amplitude of H1 to its own peak amplitude in Fig. 7.13B, thereby extending the effective intensity range using all six planes, as shown in Fig. 7.14. The scaled intensity is labelled as "H1 Intensity".

Fig. 7.14 shows the SNR over the scaled intensity range. At the lowest LIBC intensity setting, the SNR still scales linearly with intensity, even though it falls below 10. This measurement was performed with an integration time of 94.4 µs. To measure the lowest intensity, a longer integration time is required.

7.3.4 Linearity with Integration Time

Since the beam is continuous, delivered as a single long pulse per spill, and as shown in Tab. 5.3 the noise of the BPM plane does not increase with integration time, the SNR must increase with integration time as more signal is integrated. Therefore, at low intensity, the SNR can be improved by increasing the integration time.



(A) H1

(B) V1

FIGURE 7.15: SNR histograms at different integration times for a 169 MeV proton beam with an intensity of $160 \times 10^8\,\mathrm{s}^{-1}$.

The same 169 MeV proton beam intensity scan described in Section 7.3.3 was repeated with different integration times. Fig. 7.15 shows the SNR histograms for the lowest intensity spill in the scan for H1 and V1. For H1, a clear trend of increasing SNR with integration time is observed, whereas for V1 only a slight increase is visible (with the histogram heights decreasing).

Fig. 7.16 shows the SNR over scaled intensity at different integration times.

If the SNR increases linearly with integration time, the ratio of SNR to integration time for different integration times should overlap, as observed for intensities above the black star in Fig. 7.16 ($49 \times 10^6\,\mathrm{s}^{-1}$). At lower intensities, however, the SNR no longer scales linearly with integration time, since the SNR is not linear with intensity at short integration times, where essentially no signal is visible, as shown in Fig. 7.15.

Notably, at low intensities, the SNR still increases linearly with intensity at long integration times (244.2 µs). This implies that the lower intensity detection limit can always be extended by increasing the integration time.

At an intensity of $49 \times 10^6\,\mathrm{s}^{-1}$,as indicated by the black star in the plot, the SNR exceeds 10 for an integration time of 244.2 µs (4 kHz readout rate), while it remains below 10 for 94.4 µs. This implies that a 4 kHz readout rate can be used to detect this 4 mm-wide 169 MeV proton beam (LIBC setting: E174 F1) at $49 \times 10^6\,\mathrm{s}^{-1}$, ensuring an SNR greater than 10.

The horizontal red and blue lines in Fig. 7.16 indicate the intensity limit for this 4 mm-wide 169 MeV proton beam, where the SNR exceeds 10 at 4 kHz but drops

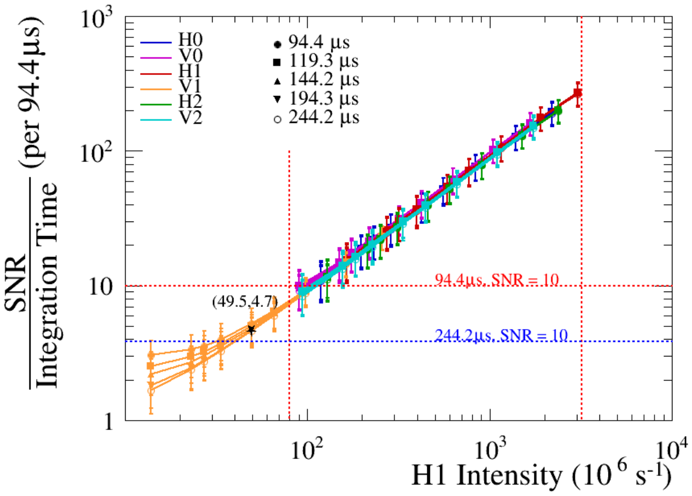


FIGURE 7.16: Ratio of SNR to integration time with scaled intensity. The vertical red lines indicate the LIBC proton beam intensity range, and the horizontal lines mark SNR = 10 for integration times of 94.4 µs (10 kHz readout rate) and 244.2 µs (4 kHz readout rate).

below 10 at 10 kHz. Within the available intensity range for this beam at HIT, the plot shows that a 4 kHz readout rate suffices at low intensities. For the higher intensities, the detector provides a good SNR at 10 kHz.

As described in Section 3.3.1, for protons there are 255 energy steps and 6 focus settings for each energy step. It is impossible to test all these settings with intensity scans one by one, as this would be extremely beam-time consuming. However, since the detector response is linear, the intensity limit can be projected to other settings.

The next section will provide further evidence of linearity with respect to the specific energy loss.

7.3.5 Linearity with Specific Energy Loss

Two high-intensity energy scans covering the entire LIBC energy range were performed, as listed in Tab. 7.2: one for proton beams and one for carbon beams, to measure the detector response as a function of specific energy loss.

The beam width varies with energy, as shown in Fig. 7.3D. It is manually set to decrease with increasing energy, as described in Section 3.3.1. The beam widths in the carbon energy scan are relatively narrow, ranging from 2 to 7 mm, while those for protons range from 4 to 15 mm.

Fig. 7.17 shows the calculated specific energy loss as a function of kinetic energy per nucleon for proton and carbon beams. The data were obtained from PSTAR database for protons [72] and MSTAR database for carbon [73]. The x-coordinates of the dots on the curves indicate the energies included in the two energy scans, with y representing the specific energy loss interpolated from the curves. In this energy range, the specific energy loss increases at lower energies, forming the Bragg peak that enables dose localization in particle therapy.

Fig. 7.18 illustrates the average integrated amplitude $\overline{A}$ per particle per second as a function of the specific energy loss. Here, $\overline{A}$ is the integrated amplitude of the calibrated averaged beam profile for each energy step. To obtain the value per particle per second, $\overline{A}$ is divided by the beam intensity, which defines the vertical

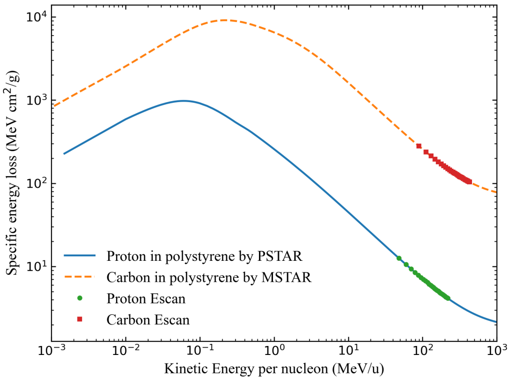
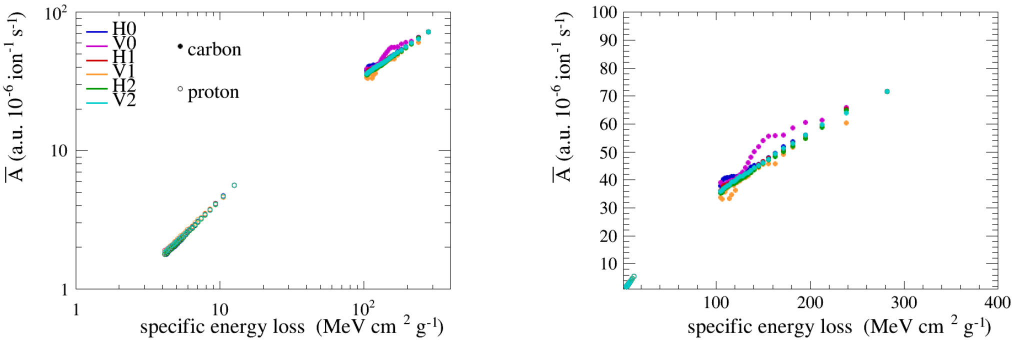


FIGURE 7.17: Calculated specific energy loss in polystyrene of protons from PSTAR and carbon ions from MSTAR. Points on the curves indicate the energy scans.



(A) Log-Log Scale.

(B) Linear Scale.

FIGURE 7.18: The average integrated amplitude $\overline{A}$ per particle per second for proton and carbon plotted as a function of their specific energy loss in polystyrene.

axis in the plots. As shown in Fig. 7.11B, the six planes exhibit different plateau heights due to variations in coupling efficiency. For better comparison, the $\overline{A}$ values from the six planes are normalized to the same reference level.

For protons in all planes and for carbon ions in planes H2 and V2, $\overline{A}$ is linear with the specific energy loss. At the same time, the proton and carbon curves exhibit different slopes in the plots, indicating ion-dependent linearity.

In 2016, a similar study was performed with the first prototype of the SciFi-BPM using an LHCb-style fiber plane for four types of ions [1] at the isocenter, where the beams are wider. There, $\overline{A}$ was also found to be linear with the specific energy loss for the same particle type, but not across different particles. This ion-dependent behavior was explained by saturation of the scintillator at high specific energy loss in the core region, where ionized electrons must escape the saturated core to produce light that is again proportional to the specific energy loss.

For carbon ions, the integrated amplitude versus specific energy loss curve is clearly not linear for V0, H0, and V1, particularly at high energies. This can be explained by the fact that, in the case of a narrow beam, the local (sub-channel) response uncertainty can have a large relative weight on the integrated amplitude.

The calibration is performed per channel, each 0.8 mm wide. Local variations below one channel width are therefore not calibrated. Since each channel contains 4–5 fibers, variations within the 0.8 mm width exist. For a narrow and tall beam, part of the beam may cross only half of a channel, in which case the calibration factor applied to the entire channel does not properly correct the partial response.

This sub-channel variation is expected to introduce a position bias of less than 0.2 mm. For a 2 mm-wide square beam, if 1.6 mm of its width is distributed across two channels and the remaining 0.4 mm falls on half of a channel, and if this half channel happens to contain no fibers, no signal is produced from that region. As a result, the reconstructed beam center would be shifted to the 0.8 mm position instead of the true 1.0 mm, introducing a bias of 0.2 mm.

In practice, however, the beam at the nozzle is typically set to be wider than 2 mm and has a Gaussian rather than a square profile, so the edge regions contribute less to the position bias. In addition, ongoing efforts to improve the uniformity of the fiber planes are expected to further reduce the position bias for narrow beams.

7.3.6 Higher Beam Limit

As described in Section 4.3, the photodiode array saturates at 3.5 V, which corresponds to an amplitude value of about 25 750.

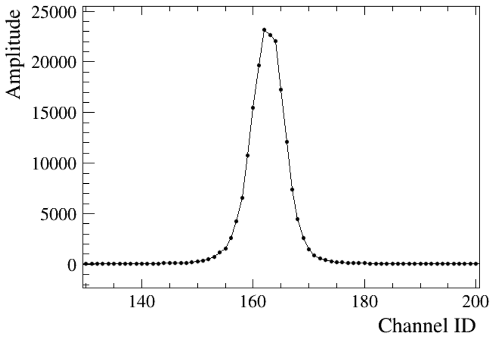


FIGURE 7.19: Maximum event observed. An interlock occurred within 1 ms after this event.

The maximum peak height observed by the SciFi-BPM is shown in Fig. 7.19, with a peak value of about 24 000, which is below the saturation limit. This was, however, an abnormal event occurring at the beginning of a spill. The accelerator went into safety interlock within 1 ms after this event due to the extremely high intensity.

In normal operation, when the accelerator does not enter safety interlock, the highest peak observed is about 6 200. This already includes fluctuations during high-intensity spills. Thus, 6200/26250 ≈ 24%, indicating that the dynamic range of the beam lies within the lower 24% of the detector's dynamic range.

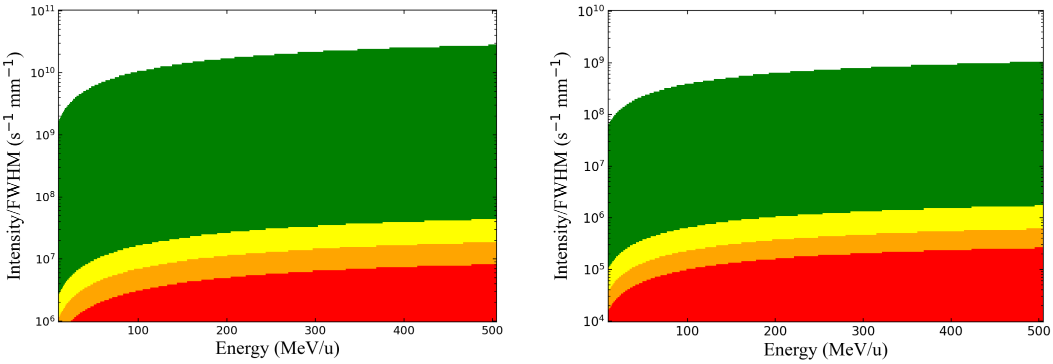
7.3.7 Detection Limit

In Section 7.3.4, the intensity limit for the 4 mm-wide 169 MeV proton beam (LIBC setting: E174 F1) at 4kHz and 10kHz is shown in Fig. 7.16 for detector plane H1.

Sections 7.3.3, 7.3.4, and 7.3.5 present the linearity of the SciFi-BPM, which is ion-type dependent. For the same type of beam, the following relation holds:

$$\mathrm{SNR} \propto \frac{\text{Intensity}}{\text{FWHM}} \cdot \text{Energy loss} \cdot \text{Integration time.}$$

Therefore, the intensity limit can be extrapolated to other beam settings of the same ion species.



(A) Proton

(B) Carbon

FIGURE 7.20: Dynamic range of H1 plane for proton and carbon beams. White region: above photodiode saturation; green region: SNR > 10 at 10 kHz; yellow region: SNR > 10 at 4 kHz; orange region: SNR > 5 at 4 kHz; red region: SNR below 5 at 4 kHz.

By extrapolating the intensity limits to the full phase space for both proton and carbon beams, Fig. 7.20 illustrates the dynamic range of H1. The x-axis represents the energy per nucleon, while the y-axis shows the ratio of intensity to FWHM, i.e., the particle rate per unit length. For this detector, which measures 1D projections, this representation is meaningful as it reflects the 1D particle flux density.

The green region corresponds to SNR > 10 at a 10 kHz readout rate. The white region marks intensities that are too high, where the photodiodes become saturated. The yellow region corresponds to SNR > 10 at 4 kHz. In the orange region, the SNR must be doubled to achieve SNR > 10 at 4 kHz. The red region represents SNR values below those of the orange region. Beams in this region can only be detected with even longer integration times.

Overall, the dynamic range of this detector spans three orders of magnitude at a 10 kHz readout rate and extends further with longer integration times, highlighting its versatility and suitability for a wide range of applications.

7.3.8 Conclusion

This section begins with an investigation of the linearity of the detector and concludes with the dynamic range of one plane, SciFi-H1.

The dynamic range is defined as the region where the SNR exceeds 10. The HIT beam range lies mainly in the green region, with a small portion in the yellow region of Fig. 7.20, indicating promising detector performance for this application.

The intensity scans were carried out at the center of the fiber plane, where (see Fig. 7.11A) the gain especially for H1 and H2 is lower than at other positions due to poorer fiber alignment in the middle region. Thus, the derived intensity limits provide only a conservative estimate of the detector performance and points to possible directions for improvement. One important improvement would be to increase the coupling efficiency. Board H1 achieves the highest amplitude under identical beam irradiation, and the remaining five boards should be optimized to reach this level or beyond. Efforts should therefore focus on improving the alignment and optical coupling between the photodiodes and the fiber planes.

7.4 Resolution

Section 7.3 has demonstrated the linearity of the detector, with particular emphasis on the SNR at low intensities. This section presents the spatial resolution at a fixed position on the detector, based on the four intensity scans, while Section 7.5 will extend the analysis to a two-dimensional resolution using a box scan. For simplicity, only the resolution of the H0–H1 combination is presented here, as the other combinations yield similar results, except for V1, which will not be used in clinical applications.

The spatial resolution determines the precision of the BPM station in measuring the beam profile. The specified requirements for the BPM station are 0.2 mm for position resolution and 0.4 mm for focus resolution as listed in Tab. 3.2.

As shown in Chapter 5, the resolution is a function of both the SNR and the beam width: the higher the SNR and the narrower the beam, the better the resolution. This section presents the position and focus resolution as functions of intensity and beam width, based on the four intensity scans listed in Tab. 7.2, measured at the center of the fiber plane at 10 kHz readout rate. The corresponding average beam widths at H1, from narrowest to widest, are 3.2 mm (carbon, 430 MeV/u), 3.6 mm (carbon, 204 MeV/u), 4.7 mm (proton, 169 MeV), and 7.1 mm (proton, 81 MeV).

The resolution is determined by calculating the standard deviation of the difference between two BPM planes and dividing by $\sqrt{2}$. The factor $\sqrt{2}$ reflects the assumption that the two BPM planes have similar and independent resolutions. Before taking the difference, both the position (Section 7.4.1) and the beam width (Section 7.4.2) are aligned.

The resolution results obtained using the RMS method are presented in Section 7.4.3, while comparisons with Gaussian fitting and linear regression are discussed in Section 7.4.4.

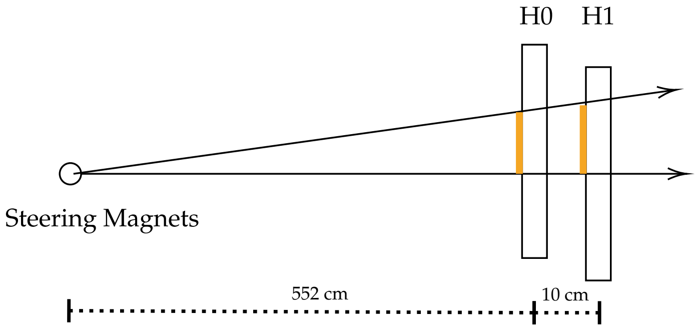


FIGURE 7.21: Sketch of the relative positions of H0, H1, and the steering magnets in beamline. The two arrows represent two beams entering H0 and H1: one perpendicular to the detectors and one under a small angle.

7.4.1 Position Alignment

Fig. 7.21 illustrates the relative positions of H0, H1, and the steering magnets. Since H0 and H1 may not be perfectly aligned with the beamline center, the measured beam positions can include different offsets. Furthermore, because H0 and H1 are located at different longitudinal positions along the beamline, a beam entering at a small angle will yield different measured positions at H0 and H1. Thus, their raw measurements are not directly comparable.

To account for these effects, the beam position measured at H0 is transformed into the coordinate system of H1 using the known distance from the steering magnets and the offsets determined from beam measurements.

As described in Section 7.1.2, the aligned positions are denoted as X_0, representing the position measured by H0 and transformed to H1, and X_1, representing the position measured directly by H1 after offset correction. Since X_0 and X_1 are directly comparable, the standard deviation of their difference is used to determine the position resolution:

$$\sigma_x = \frac{\sigma(X_1 - X_0)}{\sqrt{2}}. \tag{7.4}$$

7.4.2 Focus Alignment

The beam widths (focus, FWHM) measured by H0 and H1 per frame, denoted as F_{H0} and F_{H1}, respectively, are not directly comparable, as the beam naturally broadens due to multiple scattering. In addition, the beam may further converge or diverge optically, as illustrated in Fig. 7.22.

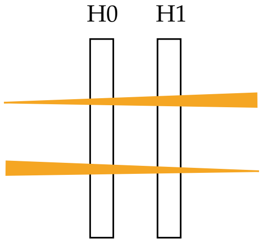


FIGURE 7.22: Schematic of the optical broadening/narrowing of the beam between BPM planes.

The width difference ΔFWHM can be defined as

$$\Delta\text{FWHM} = \begin{cases} \sqrt{\text{F}_{\text{H1}}^2 - \text{F}_{\text{H0}}^2}, & \text{if } \text{F}_{\text{H1}} \geq \text{F}_{\text{H0}}, \\ -\sqrt{\text{F}_{\text{H0}}^2 - \text{F}_{\text{H1}}^2}, & \text{if } \text{F}_{\text{H0}} > \text{F}_{\text{H1}}, \end{cases} \tag{7.5}$$

With this definition, ΔFWHM is positive when the beam becomes broader and negative when it becomes narrower.

The width difference caused by multiple scattering is

$$\Delta\text{FWHM}_{\text{MS}} = d \cdot \theta_0,$$

where $d = 10\,\text{cm}$ is the distance between H0 and H1, and θ_0 is the projected multiple scattering angle, as defined in Eq. 2.4.

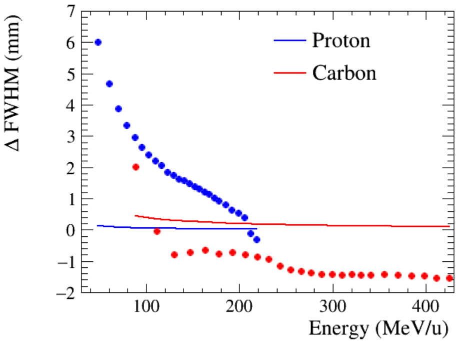


FIGURE 7.23: Measured ΔFWHM (data points) compared with the estimated contribution from multiple scattering $\Delta\text{FWHM}_{\text{MS}}$ (curves).

To compare the relative contributions of multiple scattering and optical broadening/narrowing, Fig. 7.23 shows the measured ΔFWHM from energy scans together with the estimated $\Delta\text{FWHM}_{\text{MS}}$.

For each data point in Fig. 7.23, $\overline{\text{F}}_{\text{H0}}$ and $\overline{\text{F}}_{\text{H1}}$, the mean beam widths of that energy spill, are used to calculate ΔFWHM by Eq. 7.5. $\overline{\text{F}}_{\text{H0}}$ is obtained as the mean of the F_{H0} histogram of that spill. Notably, the beam widths F_{H0} and F_{H1} are reconstructed using Gaussian fitting to avoid bias in the width determination.

For the $\Delta\text{FWHM}_{\text{MS}}$ curves in Fig. 7.23, the multiple scattering angle θ_0 is estimated for a 0.70 mm polystyrene layer, corresponding to the equivalent thickness of two fiber planes (H0 and V0), since V0 is located between H0 and H1. The contribution of the air between H0 and H1 is ignored.

Fig. 7.23 shows that the contribution of multiple scattering to the beam width difference is negligible. Therefore, the observed difference is predominantly caused by optical broadening or narrowing.

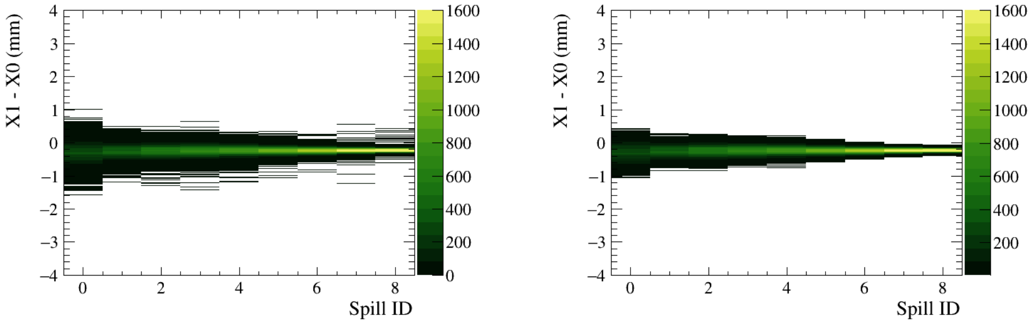
To align the widths, F_{X0} is scaled by the ratio of the average focuses for that spill, while F_{X1} is simply taken as F_{H1}:

$$\text{F}_{\text{X0}} = \frac{\overline{\text{F}}_{\text{H1}}}{\overline{\text{F}}_{\text{H0}}} \text{F}_{\text{H0}}, \qquad \text{F}_{\text{X1}} = \text{F}_{\text{H1}}.$$

The focus resolution is defined as

$$\sigma_{\text{F}} = \frac{\sigma(\text{F}_{\text{X1}} - \text{F}_{\text{X0}})}{\sqrt{2}}.$$

7.4.3 Resolution Results



(A) Z-axis color scale range: 0 to 1600

(B) Z-axis color scale range: 4 to 1600

FIGURE 7.24: Histograms of the position difference $X_1 - X_0$ for the 169 MeV proton beam intensity scan. The Z-axis shows the number of entries. (A) and (B) are different in Z-axis ranges.

Fig. 7.24 shows the distributions of $X_1 - X_0$ for each spill in the 169 MeV proton beam intensity scan, consisting of nine spills in total. The distribution becomes narrower as the beam intensity increases from spill to spill.

The outliers visible in Fig. 7.24A are caused by the common-mode noise of H0. After the common-mode subtraction, H0 still exhibits a noise level of about 0.30, while H1 remains close to 0. This common-mode noise of H0 can produce false clusters above threshold, leading the clustering algorithm to return the window of interest that include not only the beam profile but also additional noise channels, as shown previously in Fig. 5.17. This effect biases the reconstructed position and appears as the outliers in Fig. 7.24A. Applying common-mode subtraction to H0, reducing its level from 0.80 to 0.30, has significantly decreased the occurrence of such events.

For calculating the resolution, outliers are removed by excluding the bins in Fig. 7.24A with few entries. The resulting histogram, with outliers removed, is shown in Fig. 7.24B. The position resolution for each spill is obtained as the standard deviation of $X_1 - X_0$ for that spill in Fig. 7.24B, divided by $\sqrt{2}$, as given in Eq. 7.4. The focus resolution is calculated in the same manner.

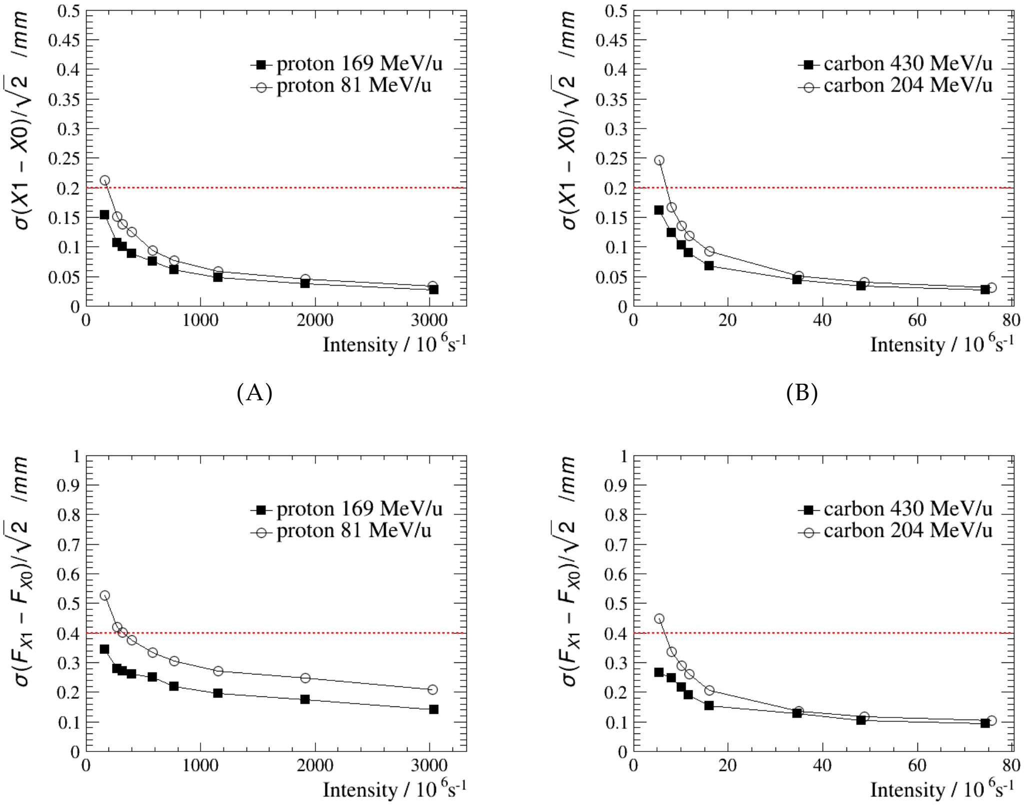
Notably, the mean of the position differences in Fig. 7.24 is not zero, indicating a position bias.

Fig. 7.25 shows the resolution as a function of intensity for the four intensity scans. The average beam widths for the four scans at H1 are 3.2 mm, 3.6 mm, 4.7 mm, and 7.1 mm, corresponding to carbon ion beams at 430 MeV/u and 204 MeV/u, and proton beams at 169 MeV and 81 MeV, respectively.

The beam position and focus in these plots are reconstructed using the floating-point RMS method. Except for several points at low intensities, all measured resolutions meet the specified requirements. These measurements were performed at a 10 kHz readout rate. As discussed in Section 7.3, operating at 4 kHz would provide improved resolution at low intensities.

7.4.4 Discussion

The spatial resolution is further influenced by the choice of reconstruction algorithm, such as RMS, linear regression, or fitting, as discussed in Section 5.6. Fig. 7.26 shows



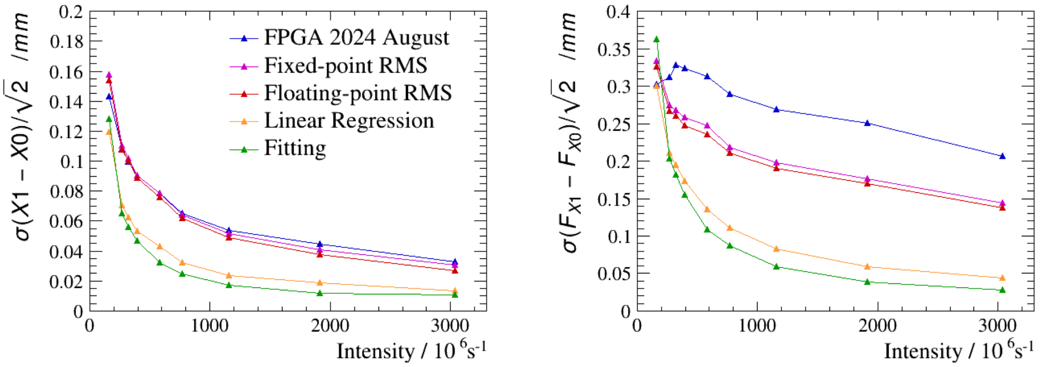
(A)

(B)

(C)

(D)

FIGURE 7.25: Spatial resolutions as a function of intensity for different intensity scans. (A) Proton position resolution. (B) Carbon position resolution. (C) Proton focus resolution. (D) Carbon focus resolution.



(A)

(B)

FIGURE 7.26: (A) Position resolution and (B) focus resolution of the 169 MeV proton intensity scan obtained with different reconstruction algorithms.

the position resolution and focus resolution of the 169 MeV proton intensity scan for these different algorithms.

The curve labelled as "FPGA 2024 August" is calculated from the FPGA reconstruction results obtained onsite at the test beam campaign in August 2024. The other four curves are obtained from offline analysis using the CPU. The difference between "FPGA 2024 August" and "Fixed-point RMS", besides the fact that one is reconstructed on the FPGA and the other on the CPU, is that the common-mode subtraction is applied to H0 in the latter but not in the former. They also use different calibration factors: the calibration for "FPGA 2024 August" is less precise than that used for the other four cases. Nevertheless, for the intensity scan at a fixed position on the detector, this has a negligible impact on the resolution. The differences among "Fixed-point RMS", "Floating-point RMS", "Linear Regression", and "Fitting" lie solely in the reconstruction algorithms themselves, as indicated by the labels.

As shown in Fig. 7.26, the position resolutions obtained with "FPGA 2024 August," "Fixed-point RMS," and "Floating-point RMS" are similar, further validating the FPGA implementation described in Chapter 6. The focus resolution from "FPGA 2024 August" is worse than that of both "Fixed-point RMS" and "Floating-point RMS," because the common-mode noise in the former is higher. This is consistent with the simulation results in Fig. 5.23, where a higher common-mode level deteriorates the focus resolution of the RMS method while having minimal impact on the position resolution.

Although linear regression and Gaussian fitting yield better resolutions, this improvement is not strictly necessary, as the RMS method already fulfills the specified requirements.

Combining these results with the resolutions in Section 7.4.3, calculated using the "Floating-point RMS", demonstrates that the detector meets the requirements for both position and focus resolution.

7.5 2D Maps of Bias and Resolution

Section 7.4 presents the resolutions for a fixed spot at the center of the fiber planes. In this section, the two-dimensional distributions of resolution and bias of both position and focus are studied over an area of 128 mm by 132 mm.

Three factors may contribute to the variations of bias and resolution in 2D:

- **Fiber alignment** — poor alignment introduces bias.
- **Gaps between arrays** — the photodiode pitch is 0.8 mm, but 0.2 mm gaps between arrays effectively increase the pitch and this may reduce the resolution around the gap region.
- **Distance to the photodiodes** — shorter distance gives stronger signals and better resolution.

This section begins by describing how the 2D distributions are generated in Subsection 7.5.1. It then explains the structures observed in these distributions, explaining how the three factors above contribute to the variations.

As an example, this section considers the H2–H0 combination. This choice is motivated by the observation in Fig. 7.9 that H0 is uniform along the fiber direction, while H2 exhibits defects in the center of the fiber plane that become more pronounced with increasing distance from the photodiodes. As explained in Section 7.2.5, these defects originate from the production of the fiber planes. It is therefore important to examine their impact on position and focus reconstruction.

7.5.1 Generate 2D Map

As mentioned in Section 7.1.3, two box scans were performed: one covering a larger region, whose line-scan-calibrated profile is shown in Fig. 7.11B, and another covering a smaller region.

The smaller scan is analyzed here to obtain the position and focus performance (bias and resolution) at each beam spot, while the larger scan is used for per-channel calibration to ensure that the calibration factors represent the average response of the 2D fiber plane.

The smaller box scan was performed with protons at 132.3 MeV and a high intensity of $3.2 \times 10^9\,\mathrm{s}^{-1}$. The beam width was 4.7 mm at H0 and 5.1 mm at H2. As shown in Fig. 7.3B, the delivered beam spots form a grid pattern covering 128 mm by 132 mm in 76 by 76 steps, corresponding to step sizes of 1.69 mm and 1.72 mm. At each position (x, y), the beam lasted for about 2.2 ms, yielding roughly 220 frames recorded by the SciFi-BPMs at 10 kHz.

From these 220 frames, the positions and focus values reconstructed in the two detector planes are compared to calculate the bias and resolution at each spot (x, y). The reconstruction uses the floating-point RMS, so the results represent the detector's real-time reconstruction performance, i.e., that of the FPGA reconstruction. Notably, the reconstructed focus is biased when using the RMS method, as discussed in Section 5.6: at lower SNR the reconstructed focus appears narrower than the true value.

The reconstructed positions and focus values are then aligned. As described in Subsection 7.4.1, positions are aligned to H1 considering the distance to the steering magnets; board rotations are not considered, assuming all are parallel. For focus alignment (Subsection 7.4.2), the focus is scaled to H1 using the ratio of the mean focus before alignment. While this ratio varies with energy, in the present box scan the energy is fixed, so constant scaling factors are applied: one for H0 and another for H2.

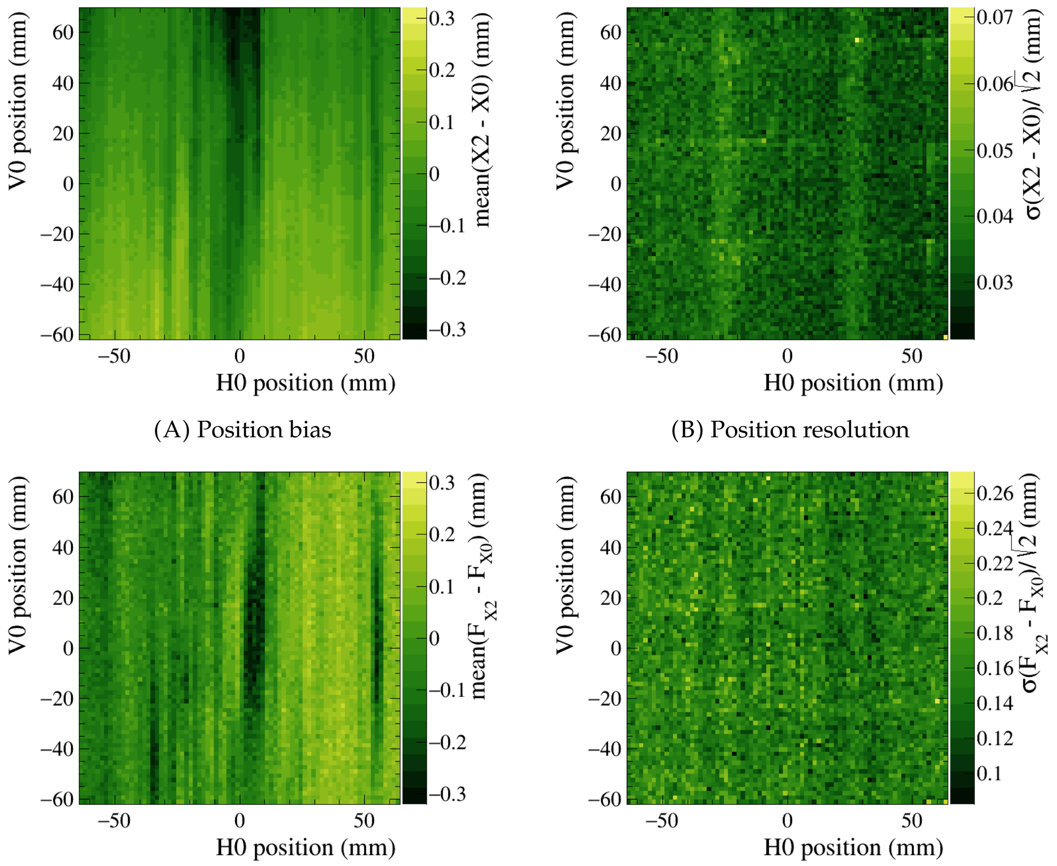
Finally, the differences in the aligned positions and focus values of these 220 frames for spot (x, y) are evaluated. Their mean gives the bias, and their standard deviation divided by $\sqrt{2}$ gives the resolution for this spot.

The bias and resolution are calculated for all the 76 by 76 spots. The coordinates of the spots are expressed as the positions reconstructed by H0 and V0, where negative values indicate locations closer to the photodiodes, and 0 mm corresponds to the center of the 320 channels, as explained in Fig. 7.2. The results are presented as 2D maps in Fig. 7.27, which reflect the characteristics of the two-plane combination.

The following sections explain these four 2D maps in detail.

7.5.2 Overall Performance

Fig. 7.27 shows the 2D maps of position and focus performance for the H2–H0 combination, measured with the high-intensity proton beam of about 5 mm width, while the corresponding 1D histograms, summarizing the distributions of all spots from these maps, are shown in Fig. 7.28.



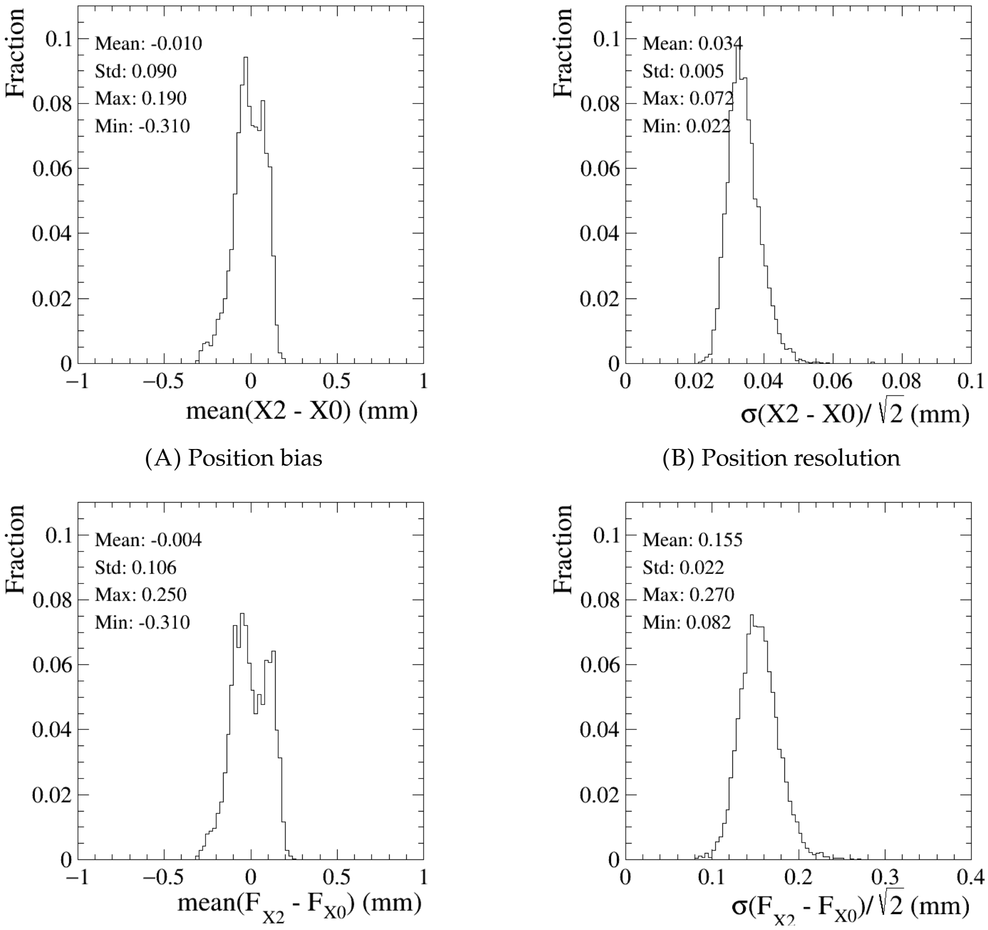
(A) Position bias

(B) Position resolution

(C) Focus bias

(D) Focus resolution

FIGURE 7.27: 2D maps of position and focus performance (bias and resolution) for the H2–H0 combination.



(A) Position bias

(B) Position resolution

(C) Focus bias

(D) Focus resolution

FIGURE 7.28: 1D histograms of position and focus performance of all spots for the H2–H0 combination. The mean, standard deviation, maximum, and minimum values are given for each distribution (units in mm).

As shown in Fig. 7.28, the mean position bias over the 2D region is −0.010 mm, and the mean focus bias is −0.004 mm. These values indicate that the alignment methods for position and focus are effective, as the resulting biases are close to zero.

The standard deviation of the position bias is 0.090 mm, and that of the focus bias is 0.106 mm. The position resolution has a mean of 0.034 mm with a small deviation of 0.005 mm, while the focus resolution has a mean of 0.155 mm with a small deviation of 0.022 mm. These values demonstrate the uniformity and good performance of the detector over the 2D region.

On top of this overall good uniformity, some small issues remain. The maximum position bias is 0.310 mm, which is on the same order as the fiber pitch size of 0.375 mm. In addition, a bimodal structure is observed in the focus bias distribution; nevertheless, the overall distribution remains very narrow.

7.5.3 Fiber Orientation

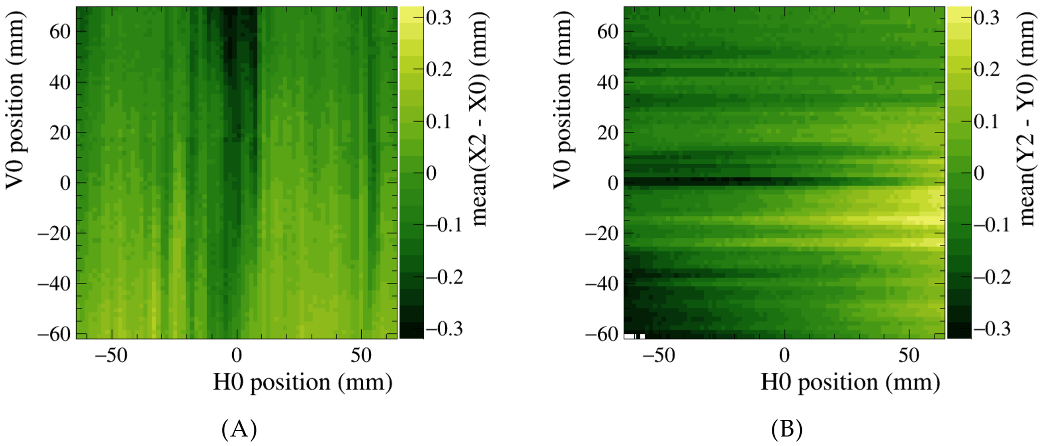


FIGURE 7.29: 2D maps of position bias. (A) H2–H0 combination. (B) V2–V0 combination

The most striking feature in the position bias and focus bias maps in Fig. 7.27 is the presence of vertical patterns. These structures follow the orientation of the scintillating fibers in H2–H0, which run along the vertical axis of the detector planes. This indicates that the observed variations in bias are strongly correlated with the fiber geometry.

Fig. 7.29 shows the position bias maps for the H2–H0 and V2–V0 combinations, with vertical and horizontal patterns that directly reflect the fiber orientations of the corresponding planes.

7.5.4 Fiber Alignment

Fig. 7.27A shows the position bias map, which can be interpreted as the bias of H2 relative to H0. In the middle of the plot (fiber plane), from bottom to top, the green shading becomes darker as the absolute value of the bias increases, with the darkest spot reaching −0.31 mm. The left panel of Fig. 7.30 illustrates a misaligned fiber overlaid on the position bias map. The misaligned fiber leads to an increasing position bias as the beam moves from bottom to top, since the true beam x-coordinate shifts away from the fiber end at the bottom, where the photodiode arrays are located.

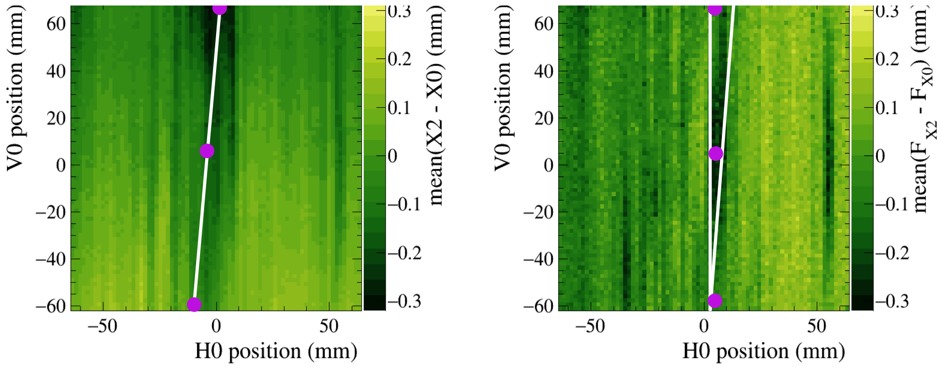


FIGURE 7.30: Exaggerated illustrations. Left: a misaligned fiber overlaid on the position bias map in Fig. 7.27A. Right: an aligned fiber and a misaligned fiber overlaid on the focus bias map in Fig. 7.27C. The purple dots represent a 5 mm FWHM beam.

The fibers are fixed with glue at both ends of the fiber frame, as described in Section 4.2. The pitch size is 0.375 mm. The maximum bias of −0.31 mm could therefore be caused by a shift of one fiber end by one or more pitch positions.

The dark green region appears in the middle of the fiber plane, consistent with the defect in the middle of the fiber plane observed in Fig. 7.9E.

This defect can also introduce a bias in the focus by causing non-parallelism between fibers, as shown in the right panel of Fig. 7.24. The purple beam spot in the middle irradiates a region of non-parallel fibers, resulting in a smaller focus projection on the photodiode array. The non-parallelism within the beam size is most pronounced in the middle of the fiber plane, resulting in the largest focus bias in this region.

The maximum position bias observed in this scan region is −0.31 mm. Since the upper limit of the *y*-coordinate (V0 position) in this scan is about 69 mm, whereas the fiber plane extends up to 128 mm, this bias would be even larger over the full detector plane. This highlights the need for either 2D calibration or improved fiber plane quality to ensure better alignment of the fibers, thus to reduce bias along the fiber direction.

7.5.5 Gaps between arrays

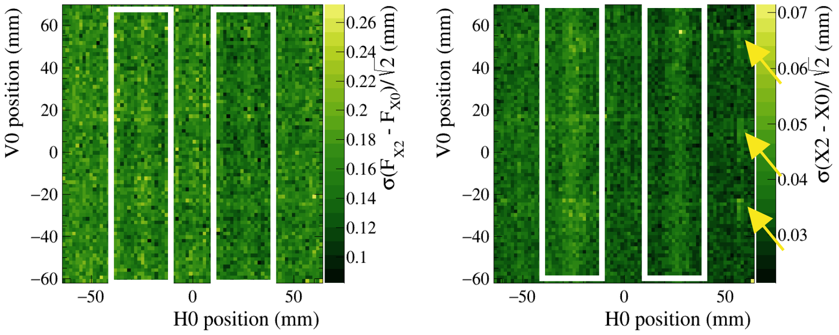


FIGURE 7.31: Illustration of regions affected by the 0.2 mm gaps between photodiode arrays. Left: the focus resolution map in Fig. 7.27D. Right: the position resolution map in Fig. 7.27B.

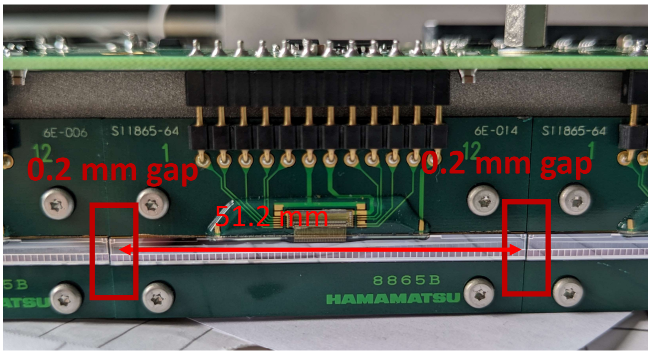


FIGURE 7.32: Photograph of the photodiode arrays and the gaps between them.

Fig. 7.27B and Fig. 7.27D show the position and focus resolutions in the scanned region. Fig. 7.31 highlights the features analyzed in this subsection.

The vertical features highlighted in the four boxes are uniform along the fiber direction, symmetric about the center, and spaced by about 50 mm. This corresponds to the 0.2 mm gaps between the photodiode arrays, as shown in Fig. 7.32. Each plane contains five arrays; the center of the middle array coincides with the center of the plane, with one gap on each side. The distance between the two gaps is 51.2 mm, further confirming that the observed structures are caused by the 0.2 mm gaps.

The position and focus biases associated with the left gap are also visible in Figures 7.27A and 7.27C, although they are less pronounced than the features caused by the fiber misalignment defects in the center of the fiber plane.

The horizontal strips in Fig. 7.31, indicated by the yellow arrows, may originate from variations in the beam intensity. In this region the beam has lower intensity, resulting in higher resolution values.

7.5.6 Distance to Photodiodes

In principle, as the distance to the photodiodes increases, the SNR decreases due to light attenuation. As shown in Fig. 7.10, an amplitude decrease of about 8% is expected for H2 over the 132 mm fiber length. The resolution map should therefore gradually worsen with increasing distance from the photodiodes. However, this feature is not observed in Figures 7.27B and 7.27D. This can be explained by the fact that at high SNR the resolution is not sensitive to such an 8% decrease. As shown in the resolution–SNR curve in Fig. 5.23, the curve flattens at high SNR.

7.5.7 Bimodal Structure in the Focus Bias

A bimodal structure is observed in the 1D distribution of the focus bias histogram in Fig. 7.28C. In the 2D distribution shown in Fig. 7.27C, the left half of the plot is dominated by negative bias values, whereas the right half is dominated by positive values.

This effect is related to the gain difference between the two halves of H2. As shown in Fig. 7.11A, the scan profile of the right half is about 20% higher than that of the left half.

For the same beam, the left side therefore reproduces a lower profile than the right side. The clustering algorithm then includes fewer channels from the lower profile, resulting in a smaller focus value when reconstructed with the RMS method.

Consequently, the left half of Fig. 7.27C appears darker than the right half, which explains the bimodal structure in Fig. 7.28C.

7.5.8 Discussion

This section presents the position and focus reconstruction performance of the H2–H0 combination for the 5 mm-wide, high-intensity proton beam. The standard deviation of the position bias is 0.09 mm, while that of the focus bias is 0.11 mm. The position resolution has a mean of 0.034 mm with a small deviation of 0.005 mm, while the focus resolution has a mean of 0.155 mm with a small deviation of 0.022 mm. These results indicate generally good performance across this region.

For the H2–H0 combination, fiber alignment is currently the dominant source of position and focus bias rather than the 0.2 mm gaps, indicating that the fiber misalignment exceeds 0.2 mm. This could be caused by one fiber end shifting by one or more pitch sizes of 0.375 mm. These fiber planes are early prototypes of the large-area glueless fiber planes. With growing experience, the engineers are becoming increasingly familiar with the fabrication process, and future mats are therefore expected to demonstrate improved performance.

The 0.2 mm gaps cause systematic biases and degraded resolution, which are uniform along the fiber direction. These biases may be mitigated by applying a smoothing algorithm that interpolates the signal amplitude across the gaps, effectively treating them as "virtual channels."

7.6 Summary

This chapter has evaluated the performance of the SciFi-BPM equipped with two-layer glueless fiber planes, demonstrating their feasibility as a beam profile monitor at HIT.

In Section 7.3, the linearity and dynamic range of the detector were presented. It was shown that the dynamic range of the clinical beams lies within the lower 24% of the detector range. For the beam settings at HIT, the expected SNR exceeds 10 at a 10 kHz measurement rate for most settings.

Sections 7.4 and 7.5 presented the resolution performance reconstructed mainly using RMS. While linear regression and fitting can achieve a position resolution better than 0.02 mm at high intensities, the RMS resolution achieve better than 0.04 mm. Across the intensity scans, the resolutions generally stayed within the HIT specifications, except at very low intensities where the requirement could still be met by extending the integration time. It is therefore evident that the RMS calculation implemented on the FPGA was sufficient to achieve the required resolution.

In Section 7.2, the per-channel calibration was performed. A variation of 13.1% (H3) in the channel response was reduced to 1.5% after calibration. This residual variation was mainly limited by the beam delivery precision at HIT.

The per-channel calibration did not account for variations along the fiber direction. As shown in Section 7.5, this effect, together with fiber alignment defects in the middle of the mats, led to sub-millimeter deviations in both position and focus. Addressing this issue requires 2D calibration methods or, more directly, improved fiber alignment in the fiber planes. Nevertheless, over the scanned 128 mm × 132 mm region, the standard deviation of the position bias was 0.09 mm, while that of the focus bias was 0.11 mm, indicating generally good performance across this region.

Chapter 8

Summary and Outlook

The work presented in this dissertation contributed to the development of the beam profile reconstruction algorithm, its implementation on FPGA, as well as the calibration and performance evaluation of the SciFi-BPM.

The SciFi-BPM measures the one-dimensional projection of the beam using two layers of 250 µm scintillating fibers, which are read out by Hamamatsu photodiode arrays with a pitch of 0.8 mm. The photodiode signals are digitized by ADCs, and both the photodiodes and ADCs are controlled by a MAX10 FPGA. In addition to signal control, the FPGA performs data packaging and beam profile reconstruction before transmitting the processed data via an Ethernet link to a PC equipped with dedicated data acquisition software. The detector operates flexibly with a readout rate ranging from 0.87 kHz to 13 kHz and a dead time of about 5.6 µs.

To establish the reconstruction chain for position and beam width, the noise characteristics were thoroughly studied. The non-zero baseline of the noise, referred to as the pedestal, required pedestal subtraction. The high level of common-mode noise observed in board-1-type planes necessitated an additional common-mode subtraction. The uniform and stable noise level across channels allowed for clustering, in which signals were identified using a fixed threshold T_c.

Pedestal subtraction is performed on the FPGA using the first 8192 noise data frames at the beginning of each measurement (after pressing the start button on the DAQ software). Some pedestals exhibited a drift of about 3 ADC counts over a one-hour measurement. However, this does not affect the per-frame position or width determination, as the drift remains well below the noise level of 9 ADC counts.

The total noise is quantified as the combination of common-mode noise and uncorrelated noise. It was observed that the difference between board 0 and board 1 lies only in the common-mode noise, with values of 0.04 and 0.81, respectively. The elevated common-mode noise of board 1 is likely related to the reference voltages provided by the ADC boards, and further investigation is needed to mitigate this effect. Nevertheless, with the common-mode subtraction the value is reduced from 0.81 to 0.3. At this level, clustering with a cluster threshold of $T_c = 3\sigma_n$, where σ_n is the standard deviation of the total noise (after the common mode subtraction for board 1 type), and $N_c = 4$ channels results in an expected interlock rate (false cluster in three consecutive frames) of less than 0.04 per day, demonstrating promising performance.

The next step of the reconstruction chain is the per-channel calibration. On the FPGA, the calibration factor is designed to be loaded into the detector via the DAQ and stored in on-chip volatile memory. Consequently, the calibration factor must be reloaded after every power cycle, which is done automatically when the DAQ connects to the FPGA.

The calibration factor is obtained from offline analysis of scan profiles. The profile height reflects the coupling efficiency of each board, with board H1 achieving the highest values. Variations in the scan profiles across channels reflect both large-scale alignment differences between the fibers and photodiode arrays and small-scale variations mainly caused by fiber alignment.

The per-channel calibration has an overall uncertainty of 1.7% (H0) and 1.5% (H1). These values include contributions from beam non-uniformity and detector response variation along the fiber direction. With this calibration uncertainty, the standard deviation of the measured position bias and width bias over a 128 mm by 132 mm region are 0.09 mm and 0.11 mm, respectively.

The final step of the reconstruction chain is the RMS calculation. Using RMS, either on the FPGA or on the CPU, the expected resolution is better than 0.05 mm for the position and better than 0.12 mm for the beam width at high intensities.

This thesis also estimated the dynamic range of the detector. At 10 kHz, it spans three orders of magnitude in the ratio of intensity to beam width, with the flexibility to expand further by increasing the integration time, thereby highlighting its versatility and suitability for a wide range of applications.

Future work should focus on several aspects to further improve this detector.

The noise performance of the ADC boards should be investigated to ensure that all board-1-type boards perform comparably to board-0-type boards, which exhibit very little common-mode noise. This is particularly important for narrow beams, where requiring $N_c = 4$ channels for the clustering may be too restrictive, and $N_c = 3$ or even $N_c = 2$ may become necessary. As N_c decreases, the probability of interlocks increases, making common-mode noise reduction essential.

Efforts should be made to improve the optical alignment between the photodiodes and the fiber planes, as the coupling efficiency between these two components directly affects the signal amplitude. Maximizing this coupling is therefore crucial.

Further improvements to the fiber planes with respect to fiber alignment are also required. The fiber winding pitch is 0.325 mm, and shifts at one end of the fiber have been observed to cause sub-millimeter biases in position and beam width. Such misalignments should be avoided in future fabrication. The fiber planes tested in this thesis are early prototypes of the large-area glueless fiber planes, and future mats are anticipated to demonstrate better performance as fabrication techniques continue to improve.

The current calibration process depends on the HIT pencil beam delivery system. For broader applications, calibration methods independent of HIT should be investigated. A UV strip light source may provide a suitable alternative.

The reconstruction algorithm developed in this work was only the first step toward the real-time position and width reconstruction. The FPGA firmware is modular, providing a framework for further development of reconstruction algorithms on the FPGA. Future efforts could focus on improved clustering algorithms or more precise linear regression methods.

In conclusion, the SciFi-BPM is a flexible detector capable of measuring the ion beam position and width with sub-millimeter precision. With further efforts to increase the SNR and improve fiber alignment, even higher precision can be expected.

Appendix A

FPGA Operating Principle and Finite State Machine

This appendix explains how logic gates are constructed from CMOS transistors, and subsequently how the basic components of FPGAs—look-up tables, flip-flops, and interconnections—are built from logic gates. The combination of these components forms the fundamental operating principle of FPGAs.

The content and most of the figures in this appendix are primarily adapted from Chapters 1 to 3 of the book *Digital Design and Computer Architecture* by *David Money Harris* and *Sarah L. Harris* [67]. The book provides an excellent and accessible explanation of the digital circuit design world, with well-structured content. It is recommended not only for FPGA beginners but also for experienced designers.

A.1 From CMOS to Logic Gates

Complementary metal-oxide-semiconductor (CMOS) is a technology used to produce p-type and n-type MOSFETs on a single silicon substrate. It is the fundamental building brick of modern integrated circuits. Fig. A.1 shows the cross-sectional structures of nMOS and pMOS transistors. Both have a substrate, a source, a drain, and a gate. The gate consists of a polysilicon layer placed over a thin layer of silicon dioxide, which lies on the substrate. Because silicon dioxide is an insulator, this arrangement forms a capacitor. This sandwich structure is known as a metal-oxide-semiconductor (MOS). While historically, the gate layer was metal, but it was replaced by polysilicon because metal cannot withstand the high temperatures used during fabrication. An nMOS transistor is formed on a p-type semiconductor substrate with n-type source and drain regions, whereas a pMOS transistor is formed on an n-type substrate with p-type source and drain regions.

Fig. A.2 illustrates the operation of an nMOS transistor. When a voltage is applied to the gate while the substrate is connected to ground, electrons accumulate in the lower layer of the MOS structure. If the voltage exceeds a certain threshold, the region beneath the gate inverts to n-type semiconductor. This forms a channel between the source and the drain, allowing current to flow. The voltage effectively controls the concentration of charge carriers, a phenomenon referred to as the field effect. Consequently, the full name of this type of transistor is the metal-oxide-semiconductor field-effect transistor (MOSFET), or simply a MOS transistor. In an nMOS transistor, the source and drain are connected by the n-type channel. A pMOS transistor behaves similarly, but its charge carriers are holes, and the gate is pulled to ground potential to form the channel.

MOSFETs function like switches. As illustrated by the schematics in Fig. A.1, when the gate is driven high (for nMOS) or low (for pMOS), the transistor is turned

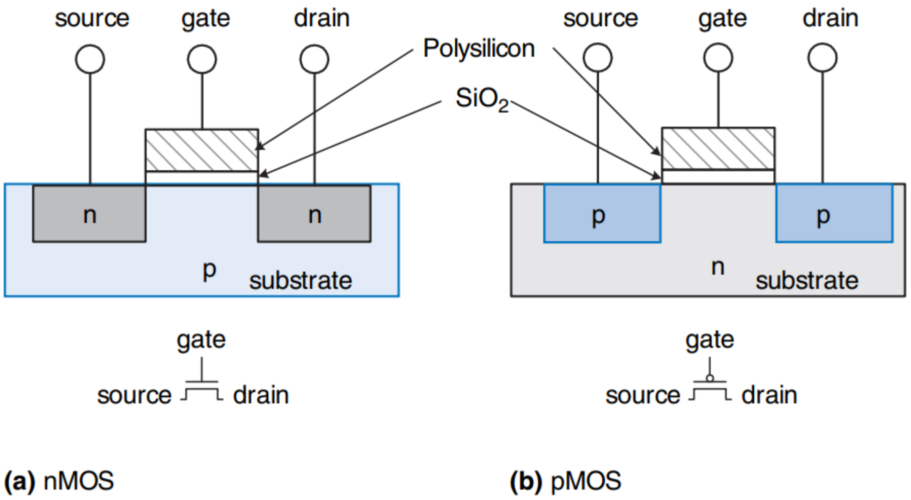


Figure 1.29 nMOS and pMOS transistors

FIGURE A.1: cross-sectional views and schematics of NMOS and PMOS transistors [67]

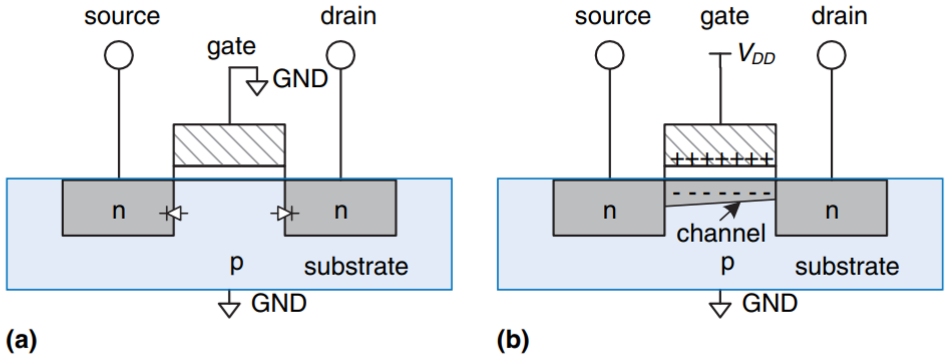
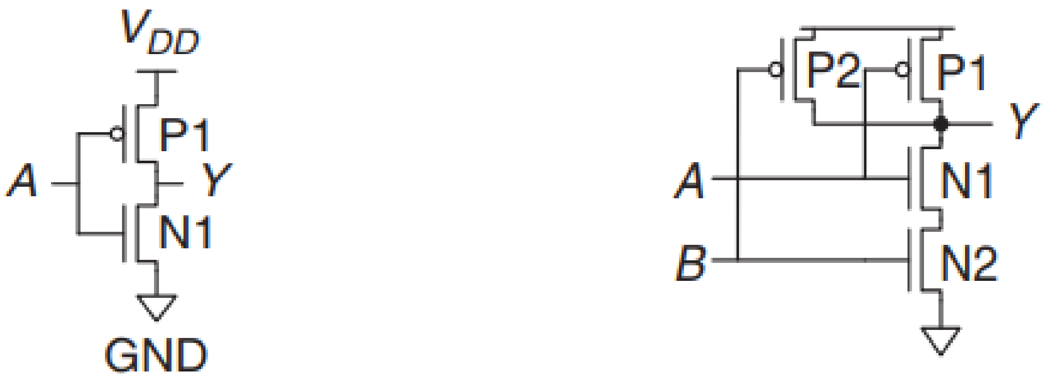


Figure 1.30 nMOS transistor operation

FIGURE A.2: NMOS transistor operation [67]



(A) NOT gate [67]

(B) Two-input NAND gate [67]

FIGURE A.3: CMOS logic gates

on. An nMOS transistor with its substrate connected to ground is most effective at passing a ground (0) signal, while a pMOS transistor with its substrate connected to the power rail is most effective at passing a high (1) signal. Consequently, CMOS circuits combine nMOS and pMOS transistors so that both 0 and 1 signals can be passed.

Fig. A.3 shows a NOT gate and a two-input NAND gate constructed from pMOS and nMOS transistors. In the NOT gate, when the input A is 0, the nMOS is off and the pMOS is on, pulling the output Y up to 1. When A is 1, the nMOS is on and the pMOS is off, driving Y down to 0 and thus inverting the input. In the two-input NAND gate, only when both inputs A and B are 1 do the two nMOS transistors in series conduct, pulling the output Y down to 0; otherwise, Y remains 1. In this way, different logic gates can be formed using CMOS.

A.2 From Logic Gates to Look-Up Table, Flip-Flop and Interconnection

Similarly to the NOT gate and NAND gate, CMOS can also be used to implement look-up tables (LUTs) for combinational logic, flip-flops for storing signal values in sequential operations, and interconnections to link these components into more complex digital circuits.

Look-Up Table

A LUT is a set of SRAM bit cells. Fig. A.4 shows the structure of a single SRAM bit cell. To read the stored bit, the bitline is initially floated, and then the wordline is enabled. To write a new bit, the bitline is driven to the desired value before the wordline is enabled.

Fig. A.5 illustrates a 2-input LUT. A decoder selects the appropriate SRAM bit cell based on the input signals. In this example, the LUT outputs 1 only when both inputs are 1 (i.e., $A = 1$ and $B = 1$), corresponding to the logic function $Y = AB$. By changing the bits stored in the SRAM cells, this LUT can implement any 2-input combinational logic function.

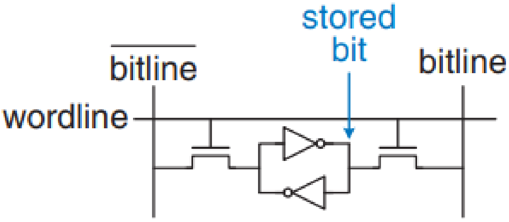
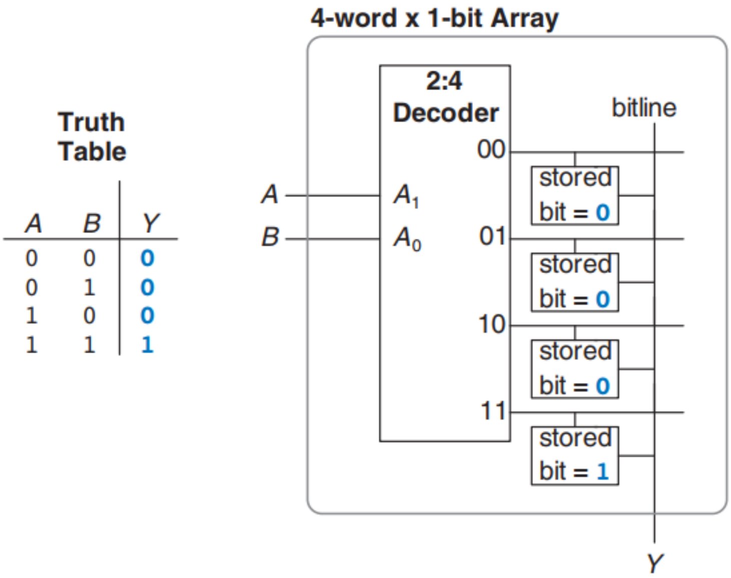


FIGURE A.4: SRAM bit cell [67]. The triangle with a circle is the symbol of the NOT gate. One bit cell costs 6 MOSFETs.



A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

FIGURE A.5: 2-input Look-Up Table Y=AB [67]

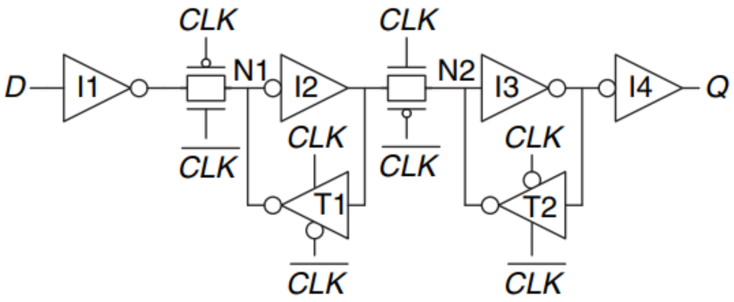


FIGURE A.6: D flip-flop schematic [67]

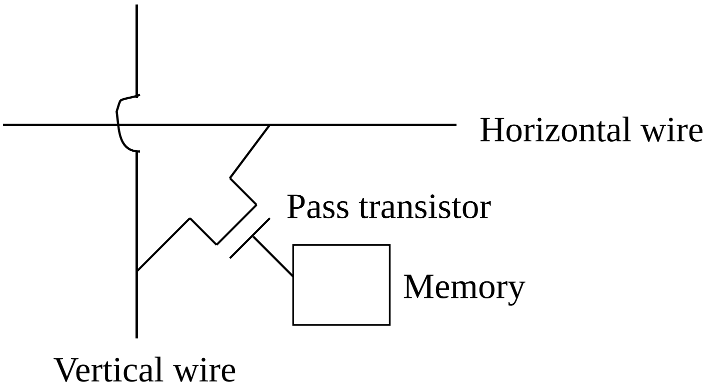


FIGURE A.7: Programmable interconnection

Flip-Flop

The most common flip-flop is the D flip-flop. Fig. A.6 shows the schematic of a D flip-flop, which requires 20 MOSFETs. The D flip-flop has a data input D and an output Q. When the clock (CLK) is low, D is passed to node $N1$. When the clock is high, the first switch is closed while the second switch is open, passing $N2$ to Q. Thus, the output Q copies the input D on the rising edge of the clock. A single D flip-flop constitutes a register, and a register file is a collection of D flip-flops sharing the same clock.

In FPGA design, flip-flops store intermediate results from the combinational logic and drive state machines, thereby enabling the implementation of sequential logic.

Programmable Interconnection

Fig. A.7 shows an example of a programmable interconnection. The value stored in the memory cell determines whether the pass transistor is on or off. By programming the memory cell, the interconnection can be configured as needed.

These interconnections make it possible to create circuits of varying sizes. They can connect LUTs to flip-flops or to other LUTs, or they can link these components to other blocks within the FPGA.

At the gate level, the LUT provides combinational logic, the flip-flop supplies sequential logic, and the programmable interconnection allows designers to scale circuits as needed. This is the basic principle of the Field-programmable Gate Arrays (FPGAs). With programmable I/Os, the FPGA can also interface with external

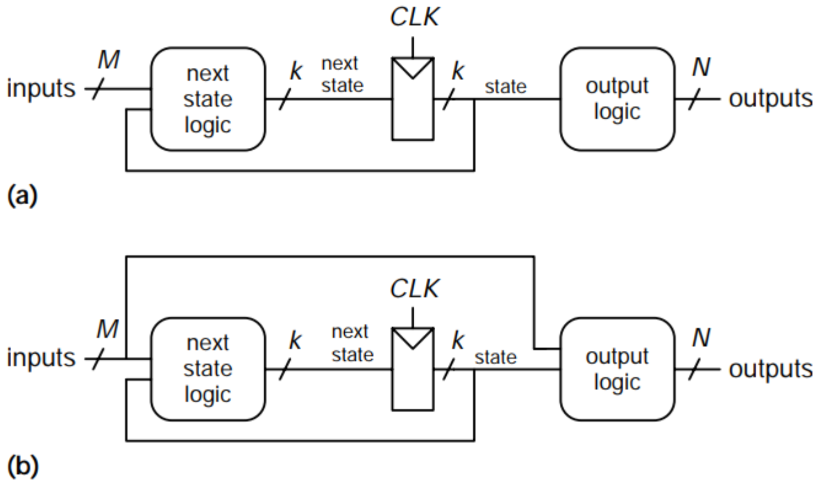


FIGURE A.8: Finite state machines (a) Moore machine (b) Mealy machine [67]

devices. Because the underlying concept of programmable I/Os parallels that of interconnections, it is not described further here.

A.3 Finite State Machine

A powerful design concept for sequantial logic is the finite state machine (FSM). Fig. A.8 shows two types of FSMs: the Moore machine and the Mealy machine. Both have a state register and a combinational logic to determine the next state. On each clock cycle, the state register transitions to this next state. In the Moore machine, the output depends only on the current state, whereas in the Mealy machine, the output depends on both the current state and the input.

FSMs are used throughout this design. For instance, in pedestal subtraction, the FSM determines when to reset the pedestal memory, when to calculate the pedestal, and when to subtract it. For RMS calculations, as Fig. A.9 shows, the FSM controls the work flow of the calculation.

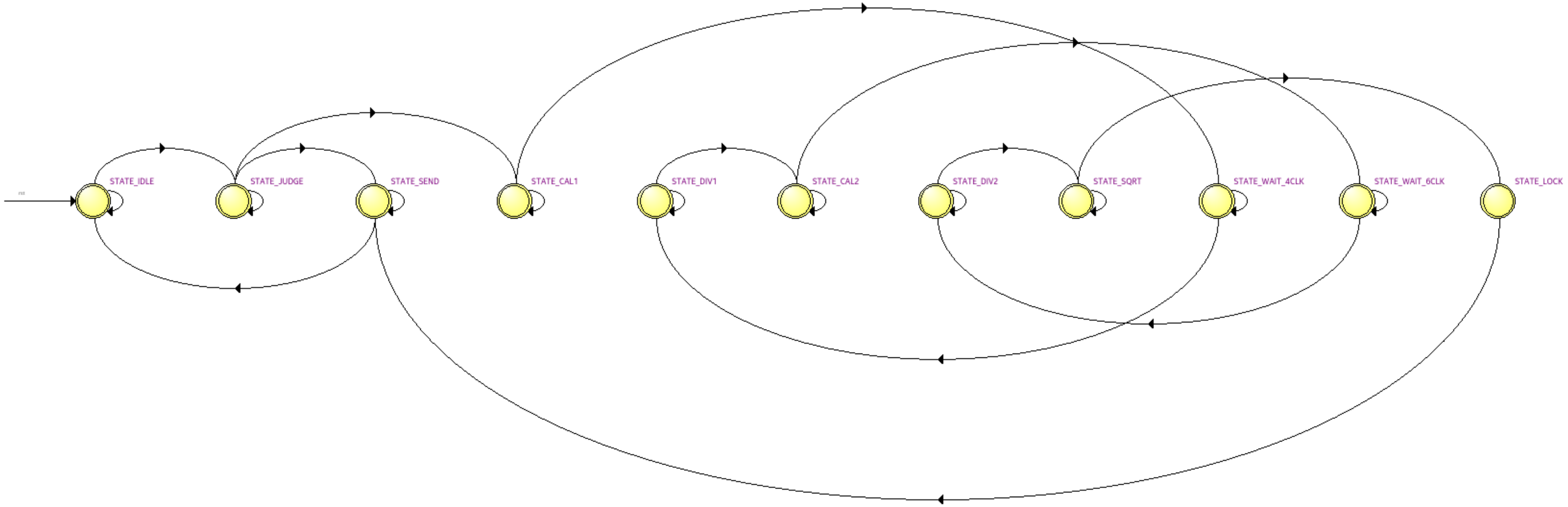


FIGURE A.9: The state machine for the RMS calculation. This state machine consists of 11 states. When a signal arrives, the machine transitions from `STATE_IDLE` to `STATE_JUDGE`. If the signal is valid, it proceeds to `STATE_CAL1` and then through additional states to compute the RMS before reaching `STATE_SEND`. If the signal is not valid, the machine transitions directly to `STATE_SEND`.

Bibliography

[1] B.D. Leverington et al. "A prototype scintillating fiber beam profile monitor for Ion Therapy beams". In: *Journal of Instrumentation* 13.05 (May 2018), P05030. DOI: 10.1088/1748-0221/13/05/P05030. URL: https://dx.doi.org/10.1088/1748-0221/13/05/P05030.

[2] W.H. Bragg. "LXXIII. On the absorption of α rays, and on the classification of the α rays from radium". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 8.48 (1904), pp. 719–725. DOI: 10.1080/14786440409463245. eprint: https://doi.org/10.1080/14786440409463245. URL: https://doi.org/10.1080/14786440409463245.

[3] Simona Giordanengo and Marco Donetti. "Dose delivery concept and instrumentation". In: *arXiv preprint arXiv:1803.00893* (2018).

[4] *Particle Therapy Co-Operative Group: Facilities in clinical Operation*. https://www.ptcog.site/index.php/facilities-in-operation-public. Accessed: 2024-05-27.

[5] Th Haberer et al. "The Heidelberg ion therapy center". In: *Radiotherapy and Oncology* 73 (2004), S186–S190.

[6] Yoshitaka Matsumoto et al. "A critical review of radiation therapy: from particle beam therapy (proton, carbon, and BNCT) to beyond". In: *Journal of personalized medicine* 11.8 (2021), p. 825.

[7] Katsuyuki Shirai et al. "Dose–Volume Histogram Parameters and Clinical Factors Associated With Pleural Effusion After Chemoradiotherapy in Esophageal Cancer Patients". In: *International Journal of Radiation Oncology*Biology*Physics* 80.4 (2011), pp. 1002–1007. ISSN: 0360-3016. DOI: https://doi.org/10.1016/j.ijrobp.2010.03.046. URL: https://www.sciencedirect.com/science/article/pii/S0360301610005973.

[8] S. Navas et al. "Review of particle physics". In: *Phys. Rev. D* 110.3 (2024). DOI: 10.1103/PhysRevD.110.030001.

[9] Radhe Mohan and David Grosshans. "Proton therapy–present and future". In: *Advanced drug delivery reviews* 109 (2017), pp. 26–44.

[10] Brian C. Baumann et al. "Comparative Effectiveness of Proton vs Photon Therapy as Part of Concurrent Chemoradiotherapy for Locally Advanced Cancer". In: *JAMA Oncology* 6.2 (Feb. 2020), pp. 237–246. ISSN: 2374-2437. DOI: 10.1001/jamaoncol.2019.4889. eprint: https://jamanetwork.com/journals/jamaoncology/articlepdf/2757520/jamaoncology_baumann_2019_oi_190095.pdf. URL: https://doi.org/10.1001/jamaoncol.2019.4889.

[11] Anissa Bey et al. "Nuclear fragmentation imaging for carbon-ion radiation therapy monitoring: an in silico study". In: *International journal of particle therapy* 8.4 (2022), pp. 25–36.

[12] Herman Suit et al. "Proton vs carbon ion beams in the definitive radiation treatment of cancer patients". In: *Radiotherapy and Oncology* 95.1 (2010), pp. 3–22.

[13] Hans A Bethe. "Moliere's theory of multiple scattering". In: *Physical review* 89.6 (1953), p. 1256.

[14] Particle Data Group. *Atomic and Nuclear Properties of Materials*. Accessed: 2025-05-27. 2023. URL: `https://pdg.lbl.gov/2023/AtomicNuclearProperties/`.

[15] Y Furusawa et al. "Inactivation of aerobic and hypoxic cells from three different cell lines by accelerated 3He-, 12C-and 20Ne-ion beams". In: *Radiation research* 154.5 (2000), pp. 485–496.

[16] Aswin Hoffmann et al. "MR-guided proton therapy: a review and a preview". In: *Radiation Oncology* 15 (2020), pp. 1–13.

[17] Masahiro Endo. "Creation, evolution, and future challenges of ion beam therapy from a medical physicist's viewpoint (part 1). Introduction and Chapter 1. accelerator and beam delivery system". In: *Radiological Physics and Technology* 15.4 (2022), pp. 271–290.

[18] Takashi Akagi et al. "Ridge filter design for proton therapy at Hyogo Ion Beam Medical Center". In: *Physics in Medicine & Biology* 48.22 (2003), N301.

[19] Marco Durante and Harald Paganetti. "Nuclear physics in particle therapy: a review". In: *Reports on Progress in Physics* 79.9 (2016), p. 096702.

[20] Miho Shiomi et al. "Comparison of carbon-ion passive and scanning irradiation for pancreatic cancer". In: *Radiotherapy and Oncology* 119.2 (2016), pp. 326–330. ISSN: 0167-8140. DOI: `https://doi.org/10.1016/j.radonc.2016.04.026`. URL: `https://www.sciencedirect.com/science/article/pii/S0167814016310726`.

[21] Hiroaki Matsubara et al. "Comparison of passive and scanning irradiation methods for carbon-ion radiotherapy for breast cancer". In: *Journal of Radiation Research* 59.5 (2018), pp. 625–631.

[22] Robert R Wilson. "Radiological use of fast protons". In: *Radiology* 47.5 (1946), pp. 487–491.

[23] G Coutrakon et al. "A prototype beam delivery system for the proton medical accelerator at Loma Linda". In: *Medical physics* 18.6 (1991), pp. 1093–1099.

[24] S. Yamada. "Commissioning and performance of the HIMAC medical accelerator". In: *Proceedings Particle Accelerator Conference*. Vol. 1. 1995, 9–13 vol.1. DOI: `10.1109/PAC.1995.504557`.

[25] D. J. Pearson. "IBA Superconducting Synchrocyclotron for Proton Therapy". In: *Proceedings of the CERN Accelerator School (CAS) on Medical Accelerators, Chios, Greece*. 2011. URL: `https://cas.web.cern.ch/sites/default/files/poster-sessions/chios-2011/pearson-poster.pdf`.

[26] Heinrich Röcken et al. "The VARIAN 250 MeV Superconducting Compact Proton Cyclotron: Medical Operation of the 2nd Machine, Production and Commissioning Status of Machines No. 3 to 7". In: *Proceedings of the 19th International Conference on Cyclotrons and their Applications (Cyclotrons 2010)*. 2010, pp. 283–285. URL: `https://accelconf.web.cern.ch/Cyclotrons2010/papers/tum1cco04.pdf`.

[27] Jiyuan Liu, Guangming Zhou, and Hailong Pei. “The clinical prospect of FLASH radiotherapy”. In: *Radiation Medicine and Protection* 4.4 (2023), pp. 190–196. ISSN: 2666-5557. DOI: https://doi.org/10.1016/j.radmp.2023.10.005. URL: https://www.sciencedirect.com/science/article/pii/S2666555723000576.

[28] Andrea Mairani et al. “Roadmap: helium ion therapy”. In: *Physics in Medicine & Biology* 67.15 (2022), 15TR02.

[29] Heng Li et al. “Image-guided particle therapy”. In: *Frontiers in Oncology* 13 (2023), p. 1175511.

[30] Freddie Bray et al. “Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries”. In: *CA: A Cancer Journal for Clinicians* 74.3 (2024), pp. 229–263. DOI: https://doi.org/10.3322/caac.21834. eprint: https://acsjournals.onlinelibrary.wiley.com/doi/pdf/10.3322/caac.21834. URL: https://acsjournals.onlinelibrary.wiley.com/doi/abs/10.3322/caac.21834.

[31] Kimberly D Miller et al. “Cancer treatment and survivorship statistics, 2022”. In: *CA: a cancer journal for clinicians* 72.5 (2022), pp. 409–436.

[32] World Health Organization. *Sustainable management of radiotherapy facilities and equipment*. World Health Organization, 2023.

[33] Vivek Verma, Mark V Mishra, and Minesh P Mehta. “A systematic review of the cost and cost-effectiveness studies of proton radiotherapy”. In: *Cancer* 122.10 (2016), pp. 1483–1501.

[34] William F Hartsell et al. “Temporal evolution and diagnostic diversification of patients receiving proton therapy in the united states: A ten-year trend analysis (2012 to 2021) from the national association for proton therapy”. In: *International Journal of Radiation Oncology* Biology* Physics* 119.4 (2024), pp. 1069–1077.

[35] Ravi A Chandra et al. “Contemporary radiotherapy: present and future”. In: *The Lancet* 398.10295 (2021), pp. 171–184.

[36] Radhe Mohan. “A review of proton therapy–Current status and future directions”. In: *Precision radiation oncology* 6.2 (2022), pp. 164–176.

[37] J Debus, KD Gross, and M Pavlovic. “Proposal for a dedicated ion beam facility for cancer therapy”. In: *University Heidelberg* (1998).

[38] H Eickhoff et al. “HICAT-The German hospital-based light ion cancer therapy project”. In: *Proceedings of the 2003 Particle Accelerator Conference*. Vol. 1. IEEE. 2003, pp. 694–698.

[39] E.C. Cortés García et al. “Optimization of the spill quality for the hadron therapy at the Heidelberg Ion-Beam Therapy Centre”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1040 (2022), p. 167137. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2022.167137. URL: https://www.sciencedirect.com/science/article/pii/S0168900222005228.

[40] David Ondreka and Udo Weinrich. “The Heidelberg Ion Therapy (HIT) accelerator coming into operation”. In: *European Physical Society Accelerator Group, Proceedings of EPAC* (2008), pp. 23–27.

[41] Walter Tinganelli et al. “Ultra-high dose rate (FLASH) carbon ion irradiation: dosimetry and first cell experiments”. In: *International Journal of Radiation Oncology* Biology* Physics* 112.4 (2022), pp. 1012–1022.

[42] R Hermann et al. "Advancements in the scintillation fibre beam monitor for low-intensity ion beams at HIT". In: *IPAC'23: 14th International Particle Accelerator Conference, 7-12 May 2023, Venice, Italy: proceedings,[Geneva]: JACoW Publishing,[2023], ISBN 978-3-95450-231-8, ISSN 2673-5490*. JACoW Publishing. 2023, pp. 4696–4698.

[43] E Feldmeier et al. "The first magnetic field control (B-train) to optimize the duty cycle of a synchrotron in clinical operation". In: *Proceedings of IPAC2012, New Orleans, Louisiana, USA* (2012).

[44] Th Haberer et al. "Magnetic scanning system for heavy ion therapy". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 330.1-2 (1993), pp. 296–305.

[45] Christian Schoemers et al. "The intensity feedback system at heidelberg ion-beam therapy centre". In: *Nuclear Instruments and Methods in Physics Research Section A* 795 (2015), pp. 92–99.

[46] E Badura et al. "Control system for cancer therapy with a heavy ion beam at GSI". In: *IEEE Transactions on Nuclear Science* 47.2 (2000), pp. 170–173.

[47] Hermann Kolanoski and Norbert Wermes. "Particle Detectors: Fundamentals and Applications". In: Oxford University Press, USA, 2020. Chap. Overview, history and concepts.

[48] Silvia Masciocchi. "Gaseous Detector". In: *talk for the 39th Heidelberg Physics Graduate Days* (2017).

[49] Muzaffer Atac. "Wire chamber aging and wire material". In: *IEEE Transactions on Nuclear Science* 34.1 (1987), pp. 475–480.

[50] Alexander Dierlamm et al. "A beam monitor for ion beam therapy based on HV-CMOS pixel detectors". In: *Instruments* 7.1 (2023), p. 9.

[51] Zhiguo Xu et al. "A new multi-strip ionization chamber used as online beam monitor for heavy ion therapy". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 729 (2013), pp. 895–899.

[52] OA Marti Villarreal et al. "Characterization of thin LGAD sensors designed for beam monitoring in proton therapy". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1046 (2023), p. 167622.

[53] DM Montalvan Olivares et al. "Monitoring of carbon ion therapeutic beams with thin silicon sensors". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1063 (2024), p. 169289.

[54] Renato Félix-Bautista et al. "Quality assurance method for monitoring of lateral pencil beam positions in scanned carbon-ion radiotherapy using tracking of secondary ions". In: *Medical physics* 48.8 (2021), pp. 4411–4424.

[55] *Kuraray Scintillating Fibers*. http://kuraraypsf.jp/psf/sf.html. Accessed: 2024-04-22.

[56] *Photodiode Arrays Hamamatsu*. https://www.hamamatsu.com/jp/en/product/optical-sensors/image-sensor/photodiode-arrays-with-amp/S11865-64.html. Accessed: 2024-04-23.

[57] *Analog Devices AD7983 Data Sheet*. https://www.analog.com/media/en/technical-documentation/data-sheets/ad7983.pdf. Accessed: 2025-07-28.

[58] *Max10 FPGAs Development Kit*. https://www.intel.com/content/www/us/en/products/details/fpga/development-kits/max/10m50.html. Accessed: 2024-04-24.

[59] LHCb Collaboration. *LHCb Tracker Upgrade Technical Design Report*. Tech. rep. CERN-LHCC-2014-001, LHCB-TDR-015. CERN, 2014. DOI: 10.17181/CERN.000E.909T.

[60] Christian Joram et al. *Lhcb scintillating fiber tracker engineering design review report: fibers, mats and modules*. Tech. rep. 2015.

[61] Blake Leverington. *DFG Final Report Individual Grant: A scintillating fiber-based beam profile monitor for ion therapy beams*. Contact Dr. Leverington for a copy. Physikalisches Institut, Heidelberg University, 2024.

[62] Michał Dziewiecki. *Beam Profile Monitor: A bundle of random information*. 1.0. Contact Dr. Leverington for a copy. Physikalisches Institut, Heidelberg University. Heidelberg, Germany, 2019.

[63] Diego Casadei. "Estimating the selection efficiency". In: *Journal of Instrumentation* 7.08 (2012), P08021.

[64] Ibrahim Al-Nahhal et al. "A fast, accurate, and separable method for fitting a Gaussian function [tips & tricks]". In: *IEEE Signal Processing Magazine* 36.6 (2019), pp. 157–163.

[65] Ilka Antcheva et al. "ROOT—A C++ framework for petabyte data storage, statistical analysis and visualization". In: *Computer Physics Communications* 180.12 (2009), pp. 2499–2512.

[66] *Intel Max 10 FPGA Architecture*. https://www.intel.com/content/www/us/en/docs/programmable/683105/current/fpga-device-architecture.html. Accessed: 2025-03-23.

[67] Sarah Harris and David Harris. *Digital design and computer architecture*. Morgan Kaufmann, 2015.

[68] Milos D Ercegovac and Tomas Lang. *Digital arithmetic*. Elsevier, 2004.

[69] *Intel Max 10 FPGA Nios II Processor Simple Socket Server Ethernet Design Example*. https://www.intel.com/content/www/us/en/design-example/714915/intel-max-10-fpga-nios-ii-processor-simple-socket-server-ethernet-design-example.html. Accessed: 2025-04-01.

[70] *Avalon Interface Specifications*. https://www.intel.com/content/www/us/en/docs/programmable/683091/20-1/introduction-to-the-interface-specifications.html. Accessed: 2025-07-02.

[71] Torbjrn Granlund. *GNU MP 6.0 Multiple precision arithmetic library*. Samurai Media Limited, 2015.

[72] J.M. Berger et al. *ESTAR, PSTAR, and ASTAR: Computer Programs for Calculating Stopping-Power and Range Tables for Electrons, Protons, and Helium Ions (Version 2.0.0)*. National Institute of Standards and Technology, Gaithersburg, MD. 2005. URL: https://physics.nist.gov/PhysRefData/Star/Text/PSTAR.html.

[73] Helmut Paul and Andreas Schinner. "An empirical approach to the stopping power of solids and gases for ions from 3Li to 18Ar". In: *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms* 179.3 (2001), pp. 299–315.

Acknowledgements

This thesis would not have been possible without the help of many wonderful people, to whom I am truly grateful.

First, I would like to thank my supervisor Ulrich Uwer for giving me the chance to work on this project and for his wisdom, mentorship, and constant support over the years.

I am especially thankful to Blake Leverington for his guidance, mentoring, and the many test beam times we shared throughout the project.

A warm thanks to Qian Yang, who started the PhD journey together with me, sharing both the pleasures and the stresses.

My heartfelt thanks go to Lucas Dittmann and Sebastian Bachmann, who were always around to help with daily work in the lab and for their kind support in reviewing this thesis.

I am deeply grateful to Sebastian Dittmeier, Michał Dziewiecki, Christian Färber and Konrad Briggl for their critical help whenever I was stuck in the FPGA jungle.

Special thanks to Alexey Zhelezov for setting up a flexible and efficient working platform on the LHCb servers and for his amazing support along the way, and Manuel Schiller for walking me through statistical puzzles and for opening the door to the world of digital algorithms.

The PI electronics workshop and the PI mechanical workshop deserve special thanks for producing and testing the custom ADC boards, building the mechanical frames, and polishing the fiber planes.

The whole LHCb Heidelberg group deserves my gratitude for creating such a friendly and inspiring working environment, for the shared knowledge, and for all the cakes and lunch breaks we enjoyed together.

Thanks also to Jakob Naumann, Julian Horn, and the wonderful HIT group for the smooth test beam times and for all the valuable information about HIT.

I also want to thank Hans-Christian Schultz-Coulon and the HighRR organisation for the opportunities to present my work and for the inspiring test beam workshop at DESY.

Finally, I owe the biggest thanks to my friends both in Germany and China, my boyfriend and my family for always being there for me, supporting me, and believing in me throughout these years.