technische universität
dortmund

Dissertation for the acquisition of the academic degree
Dr. rer. nat.

Beam monitoring at the Large Hadron Collider beauty experiment

Development of Experiment Control Systems and characterisation of beam losses

David Rolf

January 2026

Working Group Albrecht
Faculty of Physics
TU Dortmund University

First supervisor:	Prof. Dr. Johannes Albrecht
Second supervisor:	Dr. Federico Alessio
First corrector:	Prof. Dr. Johannes Albrecht
Second corrector:	PD Dr. Dominik Elsässer
Chairman of the examination committee:	Jun.-Prof. Dr. Wolfram Helml
Representative of the scientific staff:	apl. Prof. Dr. Alex Greilich
Submission date:	12. January 2026
Defence date:	24. March 2026

Abstract

In particle physics, accelerators and detectors are continually improved to enhance the precision and functionality of experiments. An important part is to continue to ensure the safety of the experiment under the new conditions. The Large Hadron Collider beauty (LHCb) detector was upgraded for the data-taking period beginning in 2022. As part of the upgrade, the Beam Conditions Monitor (BCM), the safety system of the LHCb experiment, got new readout hardware: the Machine Interface Beam Abort Decision (MIBAD) system. In addition, a new detector is planned to enhance the beam-monitoring capabilities at the experiment. The Timepix4 telescope is used as a prototype for this detector.

In this work, the control system of both detectors is fully implemented. For the BCM, the focus is on the integration of the new MIBAD system, operating since June 2023. For the Timepix4 telescope, the focus is on the reusability of the components for the future beam-monitoring detector. Finally, beam losses are studied, which can damage the LHCb detector and contribute to a machine-induced background. The losses are characterised, and the contribution to the physics operations is discussed.

Kurzfassung

In der Teilchenphysik werden Beschleuniger und Detektoren kontinuierlich optimiert, um die Präzision und Funktionalität der Experimente zu erhöhen. Ein wichtiger Teil ist, die Sicherheit des Experiments unter den neuen Bedingungen weiterhin zu gewährleisten. Der Large Hadron Collider beauty (LHCb)-Detektor wurde für die Datennahmeperiode ab 2022 aufgerüstet. Als Teil des Upgrades wurde der Beam Conditions Monitor (BCM), das Sicherheitssystem des LHCb-Experiments, mit neuer Hardware ausgestattet: dem Machine Interface Beam Abort Decision (MIBAD)-System. Des Weiteren ist ein neuer Detektor in Entwicklung, um die Strahlüberwachungsfähigkeiten am Experiment zu verbessern. Das Timepix4-Teleskop dient als Prototyp für diesen Detektor.

Diese Arbeit beschreibt die Implementierung der Kontrollsysteme für beide Detektoren. Für den BCM liegt der Fokus auf der Integration des neuen MIBAD-Systems, das seit Juni 2023 in Betrieb ist. Für das Timepix4-Teleskop liegt der Fokus auf der Wiederverwendbarkeit der Komponenten für den zukünftigen Strahlüberwachungsdetektor. Zum Schluss werden Strahlverluste untersucht, die den LHCb-Detektor beschädigen können und zu einem maschineninduzierten Untergrund beitragen. Die Strahlverluste werden charakterisiert und der Beitrag zur Physikoperation wird diskutiert.

Contents

Glossary

ADC analogue-to-digital converter

ALICE A Large Ion Collider Experiment

API application programming interface

ASIC application-specific integrated circuit

ATLAS A Toridal LHC Apparatus

BAI Beam Abort Interface

BCM Beam Conditions Monitor

BIC Beam Interlock Controller

BIS Beam Interlock System

BLM Beam Loss Monitor

BxID bunch identification number

CAN Controller Area Network

CERN Conceil Européen pour la Recherche Nucléaire

CFC charge-to-frequency converter

CIBU beam interlock system user interface

CID card identification number

CMS Compact Muon Solenoid

CP violation charge parity violation

CRC cyclic redundancy check

CU Control Unit

DAC digital-to-analogue converter

DAQ data acquisition

DCS Detector Control System

DCU Diamond Connection Unit

DIM Distributed Information Management

DNS Domain Name System

DP data point

DU Device Unit

DUT device under test

ECAL electromagnetic calorimeter

ECS Experiment Control System

ELMB Embedded Local Monitoring Board

FID frame identification number

FPGA field-programmable gate array

FSM finite-state machine

GPIB General Purpose Interface Bus

HCAL hadron calorimeter

HLT High Level Trigger

HV high voltage

IP interaction point

IR insertion region

JCOP Joint Controls Project

LBDS LHC beam dumping system

LHC Large Hadron Collider

LHCb Large Hadron Collider beauty

Linac4 Linear accelerator 4

LS2 Long Shutdown 2

LV low voltage

MCP microchannel plate

MIB machine-induced background

MIBAD Machine Interface Beam Abort Decision

NC normal conducting

OPC UA OPC Unified Architecture

pCVD polycrystalline chemical vapor deposited

PID particle identification

PLUME Probe for Luminosity Measurement

PMT post-mortem Trigger

PS Proton Synchrotron

PS Booster Proton Synchrotron Booster

RF radio frequency

RICH detector Ring Imaging Cherenkov detector

RMS root mean square

RS running sum

SC superconducting

SciFi Tracker Scintillating Fibre Tracker

SCPI Standard Commands for Programmable Instruments

SiPM silicon photomultiplier

SM standard model of particle physics

SPIDR4 Speedy Pixel Detector Readout 4

SPS Super Proton Synchrotron

TCLA Target Collimator Long Absorber

TCP Target Collimator Primary

TCPH Target Collimator Primary Horizontal

TCPS Target Collimator Primary Skew

TCPV Target Collimator Primary Vertical

TCSG Target Collimator Secondary Graphite

TCT Target Collimator Tertiary

TCTH Target Collimator Tertiary Horizontal

TCTV Target Collimator Tertiary Vertical

TFC Timing and Fast Control

ToA Time of Arrival

ToT Time over Threshold

UI user interface

USB Universal Serial Bus

UT Upstream Tracker

VELO Vertex Locator

VSS VELO Safety System

WinCC OA WinCC Open Architecture

XCVR transceiver

1 Introduction

Physicists around the world are constantly striving to improve the description of the fundamental forces in our universe. In the standard model of particle physics (SM), the three fundamental forces, the electromagnetic, the weak and the strong force, as well as the interaction between all known particles, are described. The SM is a quantum field theory and the best-tested theory of today.

However, not all physical observations can be explained with it. The gravitational force is not described, and massless neutrinos are predicted in contrast to observed neutrino oscillations, which require a finite neutrino mass. Dark matter observations from gravitational interactions between galaxies in astroparticle physics cannot be explained, to name a few. In addition, the difference between matter and antimatter in the universe cannot accurately be described by the SM. In the SM, this difference is a result of charge parity violation (CP violation) of the particles. However, the predicted CP violation is not large enough to account for the observed asymmetry [1, 2, 3].

The last discovered particle in the SM is the Higgs particle. The Higgs particle was discovered in 2012 at the Large Hadron Collider (LHC) [4, 5, 6]. It is located at Conceil Européen pour la Recherche Nucléaire (CERN) near Geneva at the border between Switzerland and France. At the LHC particles are accelerated to almost the speed of light and collided with each other at four interaction points (IP), where the A Toridal LHC Apparatus (ATLAS), A Large Ion Collider Experiment (ALICE), Compact Muon Solenoid (CMS) and Large Hadron Collider beauty (LHCb) detectors have been built [7, 8, 9, 10]. By precise measurement of the collision and its decay products, fundamental predictions of the SM can be tested. Significant deviations can indicate or prove the existence of unknown physical aspects. The CERN accelerator complex, with a focus on the LHC and machine-related concepts, is described in Chapter 2.

The energy of the colliding beam at the LHC is currently at 6.8 TeV [11]. As such, the particle beam possesses great potential to damage the machine and the experiments. To prevent damage from the beam, the beam is constantly monitored all around the LHC ring. This includes beam monitoring at the LHCb experiment. The LHCb experiment specialises in studying rare beauty (b) and charm (c) quark decays, as well as precision CP violation measurements. The LHCb experiment is described in detail in Chapter 3.

The safety of sensitive detector components is ensured by beam-monitoring detectors. The main detector responsible for the LHCb experiment is the Beam Conditions Monitor (BCM) [12]. It consists of two stations, each with eight diamond sensors located around the beam pipe. Based on the beam flux monitored by the sensors, the BCM decides to continue operations or to abort the beam to protect the LHCb experiment. The hardware responsible for making the beam abort decision is the Machine Interface Beam Abort Decision (MIBAD) board [13]. It was introduced as a part of the Long Shutdown 2 (LS2) upgrade of the experiment from 2019 to 2022.

The current BCM lacks detailed information on the beam conditions at the collision point of the particles. For this purpose, a new detector is considered. The detector will be based on a beam telescope design. A telescope currently used at the LHCb experiment is the Timepix4 telescope [14, 15]. Its design is an excellent prototype for a future beam monitoring detector. The BCM detector and the Timepix4 telescope are described in Chapter 4.

For both, the upgraded BCM and the new telescope-based detector, an Experiment Control System (ECS) is developed. Control systems continuously monitor the status and health of a system, provide slow control of hardware components, and provide live feedback to the operator. In addition to detecting errors, standard procedures can be automatised, and complex procedures can be managed and synchronised [16]. An introduction to the development of control systems is given in Chapter 5.

In Chapter 6, the ECS of the BCM is redesigned to adapt to the new MIBAD system. The ECS of the BCM monitors the fluxes of the diamond sensors. It takes action on any state change of the BCM, including a triggered beam abort or a system failure. The chapter focuses on implementing the ECS components and the underlying software.

At an early stage of development for the future beam-monitoring detector, the ECS of the Timepix4 telescope is being developed, with a focus on component reusability. The components of the control system are outlined in Chapter 7.

Essential for the beam-monitoring detectors are changes in the beam intensity and effects of stray particles (beam losses) hitting the detector. Beam losses can damage the detector and contribute to a machine-induced background (MIB), with potential impact on physics analysis [17, 18]. To study the effects of the LHC beam on the LHCb detector, it is important to understand beam losses. This is the topic of Chapter 8. For the first time, beam losses in the LHCb detector are characterised, and the contribution of MIB to physics operations is discussed.

Each main chapter has its own conclusion. Chapter 9 summarises the overall results, and an outlook is given for future beam monitoring developments at the LHCb experiment.

This work is based on my previous work [19]. Parts of the descriptive text and created figures in Chapters 2 to 5 are reused but heavily expanded upon to provide a cohesive understanding of the underlying concepts. The upgrade of the BCM ECS is closely related to the MIBAD system. The communication server for the ECS and parts of the Timepix4 ECS are upgrades to the implementation described in my previous work. The main work is presented in Chapters 6 to 8.

2 The Large Hadron Collider

The Large Hadron Collider (LHC), located at Conceil Européen pour la Recherche Nucléaire (CERN) at the border between France and Switzerland, is the largest particle accelerator in the world, with a circumference of 27 km. Protons and ions are accelerated in opposing beams and collide at the experiments with a centre of mass energy of up to 13.6 TeV [4, 11].

This chapter will introduce the LHC in Section 2.1, focusing on its technical aspects. The physical limitations of the machine and risks to the experiments from the beam are then described in Section 2.2. The implemented solutions for monitoring and controlling the beam from the machine side to prevent damage to machine components are presented.

2.1 The accelerator

Different types of particle accelerators have been invented over the years, each with its respective advantages and disadvantages. However, the fundamental concept behind them remains the same. Electromagnetic particles are accelerated by an electromagnetic field to high speed, either in a linear fashion, or by bending the particles with dedicated electromagnets into a circular path [20].

The LHC is a circular synchrotron accelerator. Synchrotron accelerators have the advantage that particles traverse a closed loop and can be gradually accelerated over multiple circulations, whereas linear accelerators are limited by available space. On the contrary, a synchrotron accelerator suffers energy losses due to synchrotron radiation. This radiation is emitted by the charged particles when they are accelerated perpendicularly to their velocity and increases with the energy of the particles [21].

This section will first briefly introduce the acceleration of particles by electromagnetic fields and magnets. Then the acceleration of the particles in the LHC is described from start to collision energy. Finally, the nominal operation cycle is presented.

2.1.1 Electromagnetic fields

The physics and the design of accelerators are based on the interaction between charged particles with an electromagnetic field. This interaction is described by the Lorentz force [22]. If an electromagnetic particle with charge q is traversing with the velocity $\boldsymbol{v}$ in an electromagnetic field with a magnetic field component $\boldsymbol{B}$ and an electric field component $\boldsymbol{E}$, a force $\boldsymbol{F}_\mathrm{L}$ is acting on the particle. In a right-handed coordinate system, the force is orthogonal to $\boldsymbol{B}$ and $\boldsymbol{v}$ and parallel to $\boldsymbol{E}$. The strength is proportional to q. The force is described as

$$\boldsymbol{F}_\mathrm{L} = q(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B}). \tag{2.1}$$

A particle is accelerated when a force is acting parallel to its flight direction. When the force is acting perpendicular to its flight direction, the orthogonal component of the force is bending the path of the particle. The particles can thus be accelerated by creating an electric field and bent with a magnetic field. In the case of the Lorentz force, as described in Equation 2.1, the bending component of the force is $|\boldsymbol{F}_\mathrm{LT}| = q|\boldsymbol{v}||\boldsymbol{B}|$. A bending radius R can be defined by setting this part of the force equal to the centripetal force [23]

$$\boldsymbol{F}_\mathrm{C} = -m|\boldsymbol{\omega}|^2\boldsymbol{r}, \tag{2.2}$$

with the mass m and the angular velocity $|\boldsymbol{\omega}|$ of the particle. The radius $\boldsymbol{r}$ is defined as the perpendicular distance from the axis of rotation $\boldsymbol{\omega}$. With $|\boldsymbol{\omega}| = |\boldsymbol{v}|/|\boldsymbol{r}|$ and $|\boldsymbol{r}| =: R$ the formula for R can be derived with Equations 2.1 and 2.2:

$$\begin{aligned} |\boldsymbol{F}_\mathrm{LT}| &= |\boldsymbol{F}_\mathrm{C}| \\ q|\boldsymbol{v}||\boldsymbol{B}| &= \frac{m|\boldsymbol{v}|^2}{R} \\ R &= \frac{m|\boldsymbol{v}|}{q|\boldsymbol{B}|} = \frac{|\boldsymbol{p}|}{q|\boldsymbol{B}|}, \end{aligned} \tag{2.3}$$

where $|\boldsymbol{p}|$ is the momentum of the particle. Equation 2.3 shows that the bending radius of the particles increases proportionally to the momentum. In contrast, as the magnetic field strength increases, the bending radius of the particles decreases. This means that the size of the accelerator and the magnetic field strength directly limit the energy of the circulating particles. By resolving the equation for $|\boldsymbol{p}|$, the momentum of the particle can be reconstructed, if the bending radius, electric charge and magnetic field strength are known [20].

2.1.2 Magnets

There are two types of magnets relevant for the LHC. Ferromagnets provide a constant, stable magnetic field and are robust; however, their maximum field strength is limited by material saturation. Electromagnets, on the other hand, can create much larger magnetic fields, but are more delicate. In ferromagnets, the magnetic field strength depends on the material. In electromagnets, the electromagnetic field is created by an electric current through a wire coil. The magnetic field strength B is given as

$$B = I\frac{n}{l}, \tag{2.4}$$

with I the current through the coil, n its windings and l its length. As can be seen in Equation 2.4, B is proportional to I and n, while being anti-proportional to l. Therefore, a short coil with many windings and a high current will create the strongest magnetic field. However, the field strength is limited, due to the heat development in the coil from the resistance of the material [24].

The efficiency of the magnet can be improved by cooling it and reducing its resistance. In certain materials, when the coil is cooled to a few kelvin, the material becomes superconducting [25]. These superconductors allow exceptionally high currents because the resistance in the material disappears. In turn, the magnetic field can be much larger than in a typical electromagnet. However, this effect is only present below a critical temperature; thus, cooling of the magnets is crucial. If the temperature of the coil exceeds its critical temperature, the superconducting effect is lost, and the high current in the coil causes a significant temperature rise. The magnet has to be turned off. This is called a quench. Therefore, superconducting magnets are only used where the high energy and momentum of the particles require such large electromagnetic fields.

The electromagnetic fields at the LHC are created by so-called dipole and quadrupole magnets [4]. The dipole magnets are used to bend the particles into a circular path around the LHC with a magnetic field strength of 8 T. They are superconducting electromagnets cooled with liquid helium to around 2 K. The quadrupole magnets are dedicated focusing magnets. They have electromagnetic fields optimised to concentrate the particles and to control their spread. They work analogously to focusing lenses in optical physics, but instead of focusing light, they concentrate the particles.

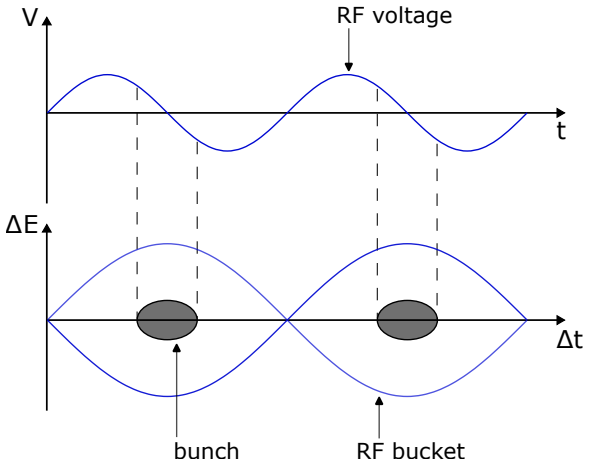


Figure 2.1: Schematic of an RF bucket. The top part of the figure shows the time-varying RF voltage $V(t)$ that generates the RF field. The bottom part shows the energy spread ΔE and the time spread Δt of the particles within the corresponding created RF bucket. Based on reference [27].

2.1.3 Radio frequency cavities

The acceleration of the particles at the LHC is done via radio frequency (RF) cavities [26, 27]. RF cavities create an electromagnetic field modulated with a frequency. Oppositely charged conductors are positioned in an array. The created field pushes particles away from a conductor as they pass and pulls them toward the following conductor. By alternating the field polarity, charged particles injected into the RF field are accelerated.

The frequency of the RF cavities determines the number of RF buckets of a beam. RF buckets are places around the beam where particles can be contained in the form of bunches. Not every RF bucket needs to be filled with a bunch. This allows for operational tuning by injecting bunch trains that conform to a so-called filling scheme.

A schematic view of the concept of a RF bucket is shown in Figure 2.1. When a particle is not in the centre of the RF bucket, the forces of the electromagnetic field will gradually push it to the central position. This is because particles arriving early experience a positive RF voltage and are accelerated, while those arriving late experience a negative RF voltage and are decelerated. In the same way, particles with higher and lower energies will arrive sooner or later, due to being faster and slower compared to the particles at the centre of the RF bucket. Because the applied forces do not perfectly bring all particles to the centre, some will start oscillating around it. These oscillations are called betatron oscillations.

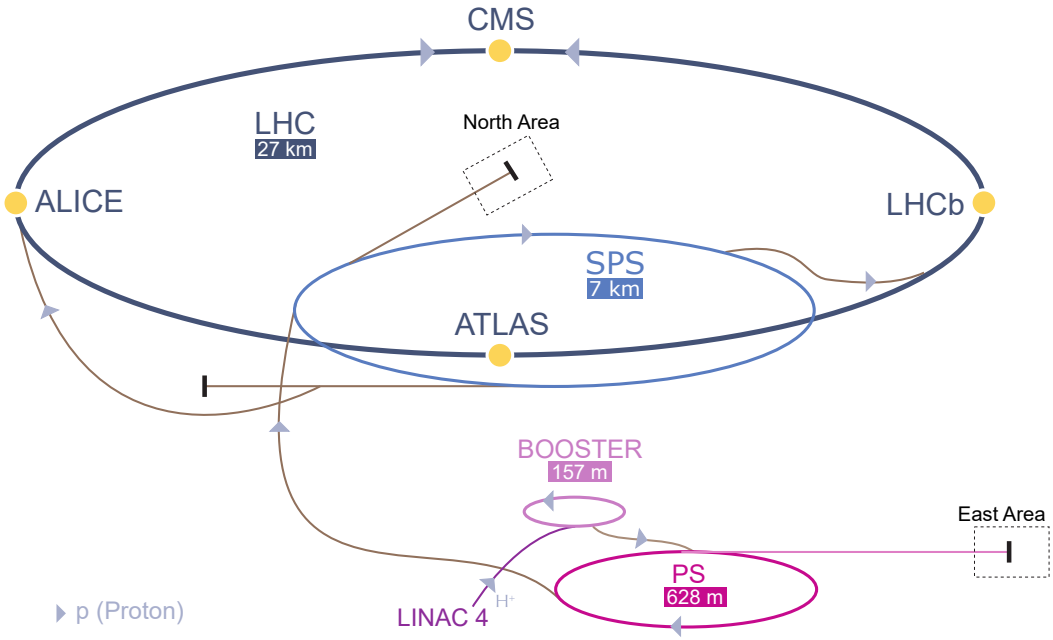


Figure 2.2: Schematic of the accelerator complex at CERN. The four main experiments are highlighted. Below the accelerators, the circumference is given. Adapted from reference [28].

2.1.4 Pre-accelerators

To bring the protons to their target collision energy, a set of pre-accelerators is required, since the magnets that guide the particles along their paths are not infinitely variable in field strength. Therefore, the CERN accelerator complex is built in multiple stages [4, 11]. A schematic of the CERN accelerator complex is shown in Figure 2.2. At the Linear accelerator 4 (Linac4), negatively charged hydrogen ions are created by adding electrons to hydrogen gas. The ions are then accelerated to an energy of 160 MeV in multiple stages using RF cavities, before being injected into the Proton Synchrotron Booster (PS Booster).

The PS Booster consists of four stacked synchrotron rings. This allows more protons to be filled into the booster simultaneously, thereby increasing the proton density of the beam. At injection to the PS Booster, an electric field is applied to strip the electrons from the ions to get a proton beam. The beam is pulsed with the RF frequency of 400 µs, creating the bunches. The PS Booster accelerates the particles to an energy of 2 GeV. The particles of the four rings are combined before being injected into the next acceleration stage.

From the PS Booster, the particles are then injected into the Proton Synchrotron (PS) and afterwards into the Super Proton Synchrotron (SPS). In these stages, the energy of the particles is further increased to 25 GeV and 450 GeV, respectively. From the SPS, the particles are injected into the LHC. Both the PS and the SPS have additional extraction lines to provide particle beams to various physics experiments and test beam facilities operating within the respective energy range [4].

2.1.5 Physics operations

The LHC is the final accelerator stage at CERN. The particles are brought to collision in the four experiments at a centre of mass energy of 13.6 TeV. The beams inside the LHC consist of trains of bunches. The collisions of the particles in the experiments are referred to as bunch crossings. Depending on the operation mode, the LHC is filled with a different number of bunches. The filling scheme determines which RF buckets are filled and which are left empty. During 2024 proton-proton collisions, the LHC is filled with up to 2350 nominal bunches, where a nominal bunch consists of up to $\sim 1.6 \times 10^{11}$ protons [29]. During machine development and commissioning, it is possible to adapt the number of bunches and the number of particles per bunch injected into the beam. The bunch spacing is 25 ns, at a revolution frequency of 11 245 Hz, resulting in a bunch crossing rate of 40 MHz.

A nominal machine cycle for physics operations, referred to as a physics fill, follows a sequence of defined beam modes [30]. The most relevant are given. The bunches are not all injected at the same time. Before each fill, several tests are performed to confirm the proper operation of all machine components. Then, the RF buckets are gradually filled up with bunches from the SPS that conform to the filling scheme. The period before the beam is filled is called the `INJECTION` mode. While the energy of the particles is increased to their destination energy, the machine is in the `RAMP` mode. When the energy of the beam is fully ramped, the machine will enter a `FLAT TOP` followed by a `SQUEEZE` mode to focus the beam by reducing the spread of the particles in the RF bucket. Afterwards, the beam collision is prepared and optimised for data taking in the `ADJUST` mode. When the beam has stabilised, for the remaining time period of the fill, the LHC is in the `STABLE BEAMS` physics mode for data taking. The beam is aborted either when safe operations can not be ensured or when the number of protons in the beam becomes too low. The remaining particles are extracted from the machine by the beam dump facility, and the machine enters the `BEAM DUMP` mode. The magnets are ramped down (`RAMP DOWN` mode), and pre-cycled (`CYCLING` mode) before beginning the next LHC cycle.

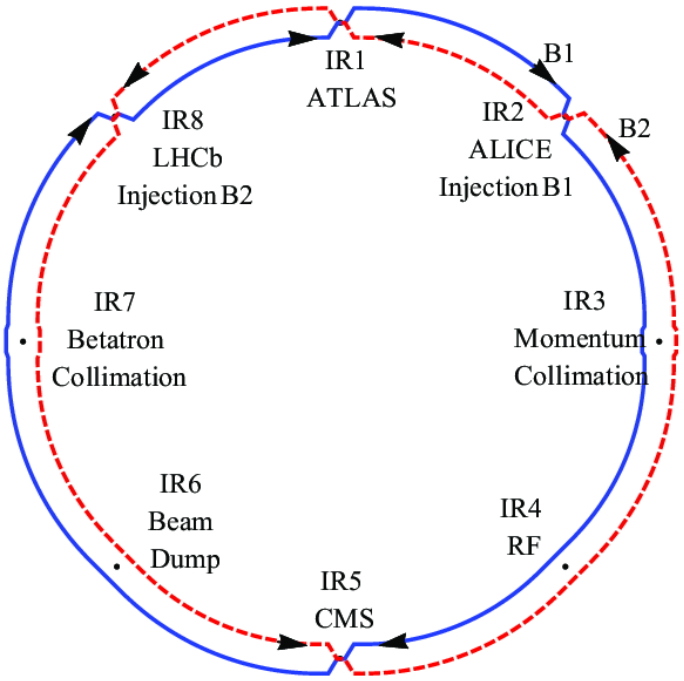


Figure 2.3: Schematic of the LHC. The eight IR are highlighted. The beams B1 and B2 are depicted in red and blue. Adapted from reference [31].

2.1.6 Insertion Regions

A schematic view of the LHC is given in Figure 2.3. The LHC tunnel is not a perfect circle, but consists of eight straight sections, referred to as octants. Each section has an insertion region (IR), hosting beam operating facilities and the four physics experiments. The octants are connected by arcs, where the superconducting dipole magnets (refer to Section 2.1.2) are bending the beams.

The beam acceleration RF system (refer to Section 2.1.3) is located at IR4. Undesired particles accompanying the beam, called beam halo, are removed by the momentum and betatron collimation facilities located at IR3 and IR7, respectively. The process is called beam cleaning. The origin of the beam halo, and the collimation system are described in Sections 2.2.1 to 2.2.4. The beam dump system, described in Section 2.2.5, is located at IR6. The four experiments A Toridal LHC Apparatus (ATLAS), A Large Ion Collider Experiment (ALICE), Compact Muon Solenoid (CMS) and Large Hadron Collider beauty (LHCb) are located at the remaining IR1, IR2, IR5 and IR8, respectively. The LHCb experiment is described in detail in chapter 3.

2.1.7 Secondary beam test facilities

At CERN, part of the beam is extracted for use in secondary test facilities [32, 33] with fixed target experiments and for beam tests. They can, for example, be used for experimental studies and development studies on prototype detector components.

The two main test facilities are the east and north areas, extracting beams from the PS and the SPS, respectively. The extracted beam is split and shot onto fixed targets, resulting in secondary mixed or pure hadron beams, pure electron beams or pure muon beams. Particle energies range up to 15 GeV in the east area and up to 400 GeV in the north area, enabling various studies within their respective energy ranges. Various locations along the secondary beam lines allow for experimental setups. Control of the beam parameters allows a wide range of settings to be adjusted to the needs of the experimental groups using the facilities.

2.2 Machine protection

The particle beam is not perfect, but constrained by physical limitations of the accelerator [17, 18]. Undesired particles accompany the beam in the form of a beam halo. Halo particles hitting machine components cause so-called losses that can shower onto the experiments, leading to an undesired background signal. These concepts are explained in the first part of this section.

The second part focuses on protecting the LHC from a misbehaving beam. The LHC implements procedures to reduce undesired losses and risks to the machine and the experiments. The beam collimation system removes halo particles from the beam. The beam dump system is responsible for aborting the beam in case of critical scenarios. Both systems are described.

2.2.1 Beam halo

The beam has a Gaussian distribution around its centre. The particles at the centre define the nominal orbit. Particles further away from the centre position are off orbit. The beam halo is characterised by particles accompanying the beam furthest from the centre.

The buildup of halo particles can have various sources [34]. The types considered are collimation-induced particles from the betatron cleaning inefficiency at IR7 and the momentum cleaning inefficiency at IR3. Particles not entirely removed by the cleaning process contribute to halo buildup. Additionally, the beam can interact with the residual gas in the LHC beam pipe, contributing to halo buildup.

2.2.2 Beam losses

Beam halo particles that are off orbit can interact with machine components. This results in particle showers called losses. Due to the interaction of the losses with the material, energy is deposited, and the machine component heats up. As seen in Section 2.1.2, this can cause superconducting magnets to quench, leading to machine failure. In addition, losses can shower onto the experiments and can give a non-negligible machine-induced background (MIB) contribution. As such, it is of importance to control the beam halo as much as possible [35].

The MIB is differentiated based on the source of the losses. Losses originating from betatron and momentum cleaning inefficiency are referred to as betatron and momentum losses, respectively. The losses originating from beam-gas interactions are referred to as beam-gas losses. Local beam-gas losses originate directly within the long straight section of the experiment, while global beam-gas losses originate from beam-gas interactions in other sections of the LHC.

To keep track of the losses happening around the accelerator, they are monitored by Beam Loss Monitors (BLMs) [36]. If losses are deemed too high, a beam abort is requested. BLMs are, for example, located around the superconducting magnets, the injection lines, and at the machine components related to the beam cleaning. The beam losses are of special interest for the analysis of losses in LHCb in chapter 8.

2.2.3 Beam dispersion

The particles that are not on the nominal orbit can be off-momentum and are affected by beam dispersion [37, 38, 39]. The magnetic field of the magnets bends high-momentum particles less than low-momentum particles. If the particles are offset from the nominal position in the longitudinal or transverse plane, they will start to oscillate around this position as mentioned in Section 2.1.3. If the entire beam is off momentum, for example, due to a misconfigured RF frequency, this can lead to a drift in the beam orbit, potentially hitting machine components.

Figure 2.4 shows the effect of beam dispersion at one of the experiments, where the beams are crossing. The collision point of the particles shifts, dependent on the amount of dispersion. This, in turn, leads to more or less head-on collisions of the bunches due to the changed beam separation.

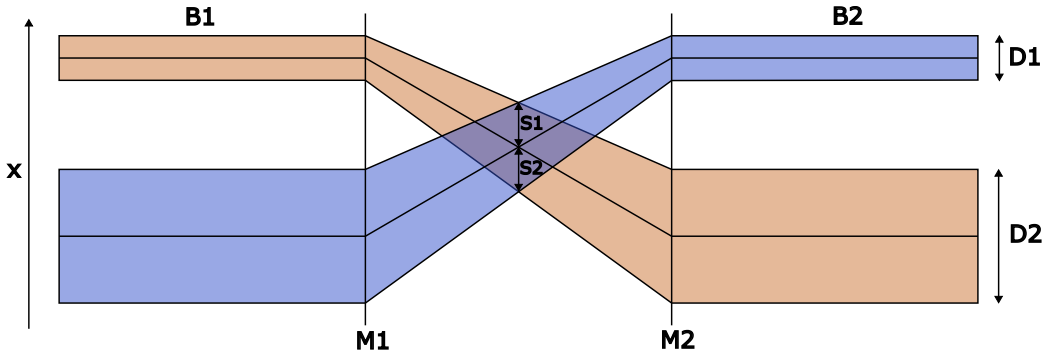


Figure 2.4: Schematic of beam dispersion. Beam 1 (B1) and beam 2 (B2) are bent at the magnets M1 and M2. The collision point along x is shifted by S1 and S2, depending on the amount of beam dispersion D1 and D2.

2.2.4 Beam collimation

Beam cleaning is performed using a multi-stage collimation system [40, 41]. A collimator consists of a set of collimator jaws that are used to reduce the halo particles of the beam. The collimators do not require cooling. As such, the regions of the collimators are called normal conducting (NC) or *warm* regions. They are meant to protect the superconducting (SC) or *cold* regions with the superconducting magnets from heat development due to excess losses. An exemplary schematic of the collimation system is shown in Figure 2.5.

In a first stage, the beam is intercepted by a set of Target Collimator Primarys (TCPs). The primary halo particles of the beam are colliding with the collimator material and are partially absorbed. However, some particles are scattered, and the surviving particles form a secondary beam halo. The secondary beam halo, in turn, is intercepted in a second stage by another set of Target Collimator Secondary Graphites (TCSGs), leading to the creation of a tertiary beam halo. Target Collimator Long Absorbers (TCLAs) behind the collimators are used to absorb local secondary showers that do not contribute to halo build-up. Before the experiments, in a third stage, Target Collimator Tertiarys (TCTs) are added to further reduce the halo contribution.

Each collimation stage can contain multiple collimators to clean the beam in the horizontal or vertical plane. For better cleaning performance, an additional so-called skew collimator can be used, rotated by 45° around the beam axis. For the primary collimation stage, for example, the possible collimator orientations are Target Collimator Primary Horizontal (TCPH), Target Collimator Primary Vertical (TCPV) and Target Collimator Primary Skew (TCPS).

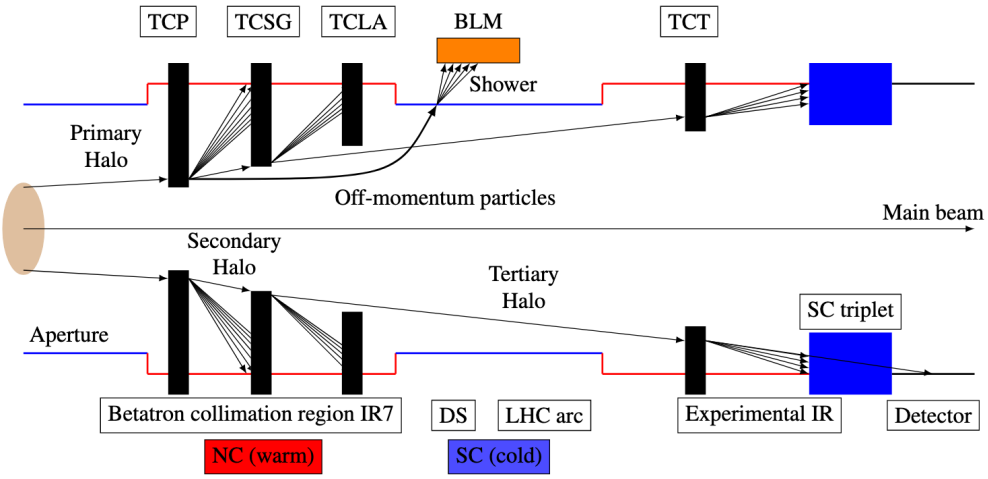


Figure 2.5: Schematic of a multi-stage collimation system at the LHC. The primary TCPs are removing the primary beam halo. Secondary TCSGs or tertiary TCTs remove surviving particles. TCLAs behind the collimators are used to absorb secondary showers. Losses at the collimators are monitored by BLMs. Taken from reference [42].

2.2.5 Beam dump system

The LHC beams are constantly monitored by a variety of systems to prevent damage to the machine from misaligned, unfocused, or otherwise unwanted beam conditions. The primary underlying system is the Beam Interlock System (BIS) [43, 44, 45]. The BLMs are part of this system, as well as Beam Conditions Monitors (BCMs) located at the experiments. In total, several hundred systems are interfaced to the BIS. The individual systems are integrated into the BIS over a beam interlock system user interface (CIBU). They are connecting the systems to various Beam Interlock Controllers (BICs) around the LHC. The BIS is schematically depicted in Figure 2.6.

The BICs are interconnected with each other in two serial current loops, one for each beam. If just one of them detects adverse conditions, the global beam permit is withdrawn. In addition to the beam permit, two injection permits are provided for each beam. Injection permits are used to allow injection into the machine only when all systems are ready. All the permits are redundant. Redundancy is crucial to prevent undesired beam aborts caused by signal failures. In addition, feedback on the combined global permit status is provided via a beam info channel, which is the logical `AND` of all related permits (see Figure 2.6). This information is provided to indicate if a permit was withdrawn by another system.

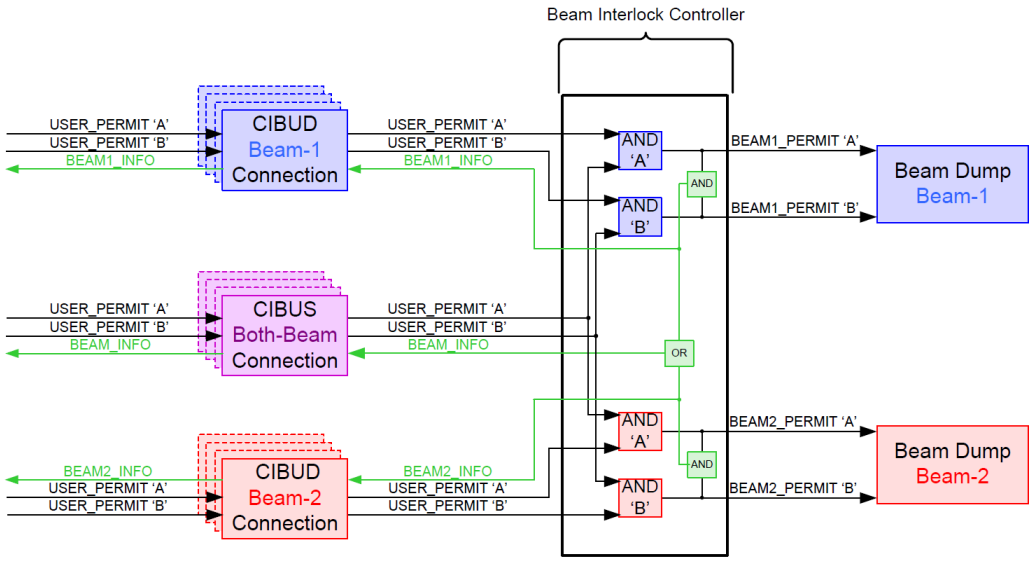


Figure 2.6: Schematic of the BIS of the LHC. The CIBU units can be of type D or S, meaning they are connected to one beam separately or to both beams simultaneously. The abort logic is indicated by the logical operands. Taken from reference [45].

When the current loop is broken by one of the BICs, the LHC beam dumping system (LBDS) at IR6 monitoring the current loops, start the beam dump procedure. The time between triggering an abort by one of the systems and extracting the beam is determined by the time it takes for the abort signal to reach IR6 and by the extraction process. By design specifications this is ensured to be below 189 µs, where 89 µs correspond to one turn of the particle beam around the LHC ring, and 100 µs are allocated for the extraction process. The limit thus assumes the worst-case scenario, in which the beam must complete a full turn before the LBDS can extract it at the abort gap.

The LBDS extracts the beam by ramping up a set of 15 kicker magnets in the time of a so-called abort gap and redirecting the particles away from their nominal trajectory. The abort gap is a set of empty RF buckets in the bunch train of the LHC providing the time to fully ramp the kicker magnets for extraction. The particles are then directed into a dedicated cylindrical graphite absorber in a sweeping pattern to distribute the damage from the high-energy beam across the absorber material.

3 The Large Hadron Collider beauty experiment

This chapter gives a detailed overview of the Large Hadron Collider beauty (LHCb) experiment [10]. The LHCb detector is optimised to measure charge parity violation (CP violation) of the particles, and the study of rare hadrons containing beauty (b) and charm (c) quarks. CP violation is introduced in the SM by the mixing of the weak and mass eigenstates of the quarks. At the LHCb experiment, oscillations have been measured, for example, in the B_d and B_s Systems.

Section 3.1 describes the LHCb detector after its upgrade during the Long Shutdown 2 (LS2) from 2019 to 2022. The infrastructure for operating the detector is described in Section 3.2. This includes the readout of detector data, the timing and clock distribution, and the control of the experiment.

3.1 The detector

In contrast to the other three experiments at the Large Hadron Collider (LHC), the LHCb detector is a single-arm forward spectrometer. This is a design choice based on the angular distribution of b quarks, which are produced in high-energy collisions and are boosted along the beam, and therefore mainly directed at small angles to it.

Since 2022, the detector has been running with upgraded subdetectors and software [16]. One of the main changes from previous data-taking periods is the removal of the hardware-level trigger. As a result, the electronics for all subdetectors are upgraded to readout at the 40 MHz bunch crossing rate. Most of the subdetectors are replaced or refurbished with new hardware. A schematic view of the upgraded LHCb detector is shown in Figure 3.1.

The standard LHCb coordinate system is used [46], with the z-axis along the beam direction, the y-axis pointing upwards, and the x-axis pointing outwards of the LHC ring. The origin is placed at the interaction point (IP). The particles of beam 1 traverse the LHCb detector towards the positive z direction, while the particles of beam 2 traverse it towards the negative z direction. In the following, the individual detector components are described.

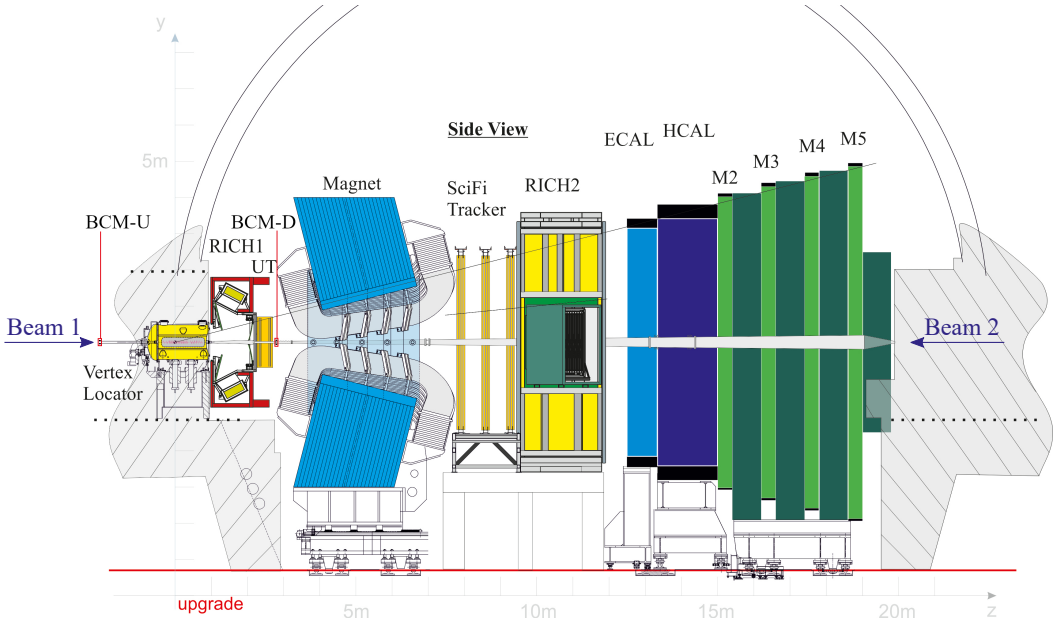


Figure 3.1: Schematic of the LHCb detector with its individual components. Adapted from reference [16].

3.1.1 Tracking system

The tracking system of the LHCb detector is composed of the Vertex Locator (VELO) [47, 48], Upstream Tracker (UT) [49] and Scintillating Fibre Tracker (SciFi Tracker) [49, 50]. The VELO surrounds the collision region. It is used to reconstruct the tracks and vertices around the collision point. The primary vertex is the point of interaction between the protons. Secondary vertices are from particle decays. Typically, due to their lifetime, b hadrons can travel short distances of the order of 2 cm within the VELO, before they decay further into other particles. The UT and the SciFi Tracker are positioned in front (upstream towards negative z) and behind (downstream towards positive z) the LHCb dipole magnet, respectively.

The magnet consists of two aluminium coils and provides a magnetic field of 4 T. By bending the tracks with the magnet, the momentum and charge of the particles can be determined (refer to Section 2.1.1). The magnet bends the particles in the horizontal plane and can be operated with reversed polarisation.

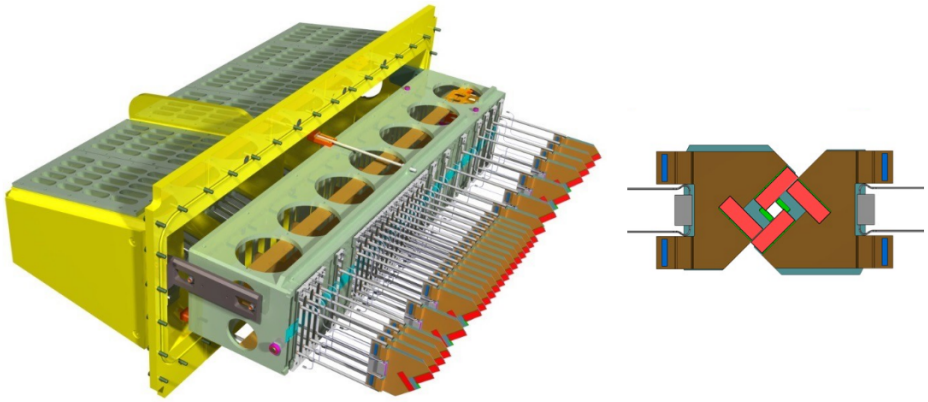


Figure 3.2: Schematic of one side of the VELO (left), as well as the front view of two modules in the closed position (right). Taken from reference [51].

Vertex Locator

The VELO consists of two movable halves that open along the x axis during injection and close once stable beams are declared. In its closed position, the minimal distance of the VELO sensors to the beam is 5.1 mm, which makes it the closest detector to the beam at the four experiments and most susceptible to beam damage. Opening the detector during injection prevents unstable beams or injection losses from damaging the sensitive parts. The VELO is interfaced with the LHCb safety system, the Beam Conditions Monitor (BCM), so the VELO can be opened in case safety cannot be ensured by the BCM. The BCM and the interface to the VELO are described in detail in chapter 4.

A total of 26 modules per VELO half are placed along the z axis surrounding the IP. A schematic of one side of the upgraded VELO and of two modules, viewed from the front in the closed position, is shown in Figure 3.2. Each module consists of four silicon pixel tiles arranged in an L-shaped geometry. Two of the tiles are placed on the front side and two on the back side of the module. Each tile is composed of a 1.4×4.2 mm sensor with 55×55 µm pixels and a thickness of 200 µm, bump-bonded to three VeloPix application-specific integrated circuits (ASICs). The tiles are cooled by a CO2 biphase microchannel cooling system embedded in a silicon substrate, which also provides mechanical support for the module.

The two sides of a VELO station are positioned with a slight offset along the z direction, which allows for an overlap of the modules in the x direction for full coverage. The VELO modules are enclosed in a separate vacuum with respect to the beam by two 200 µm thin aluminium radio frequency (RF) foils, one for each side of the VELO.

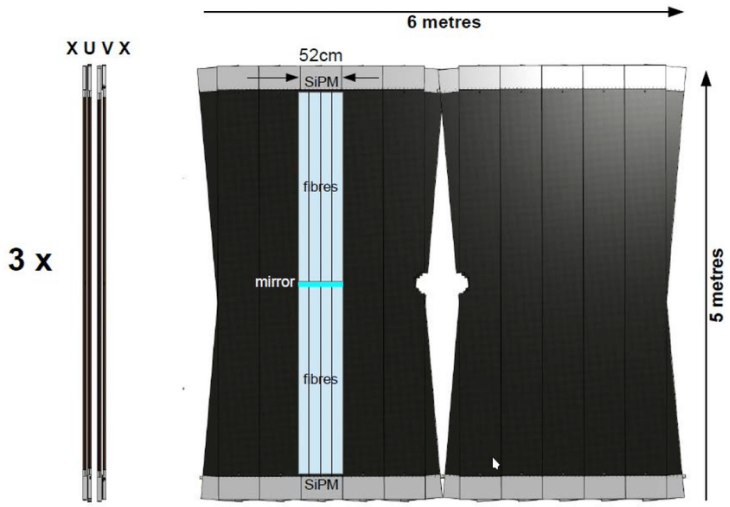


Figure 3.3: Schematic of one station of the SciFi Tracker viewed from the side (left) and the front (right). Taken from reference [50].

Upstream Tracker

The UT is the tracking detector upstream of the magnet. It has an acceptance of 1.5 m in x and 1.3 m in y. It is composed of silicon strips arranged in a X, U, V, X configuration. The strips are oriented along the y axis in the LHCb coordinate system for the x layers. The inner layers U and V are tilted by 5° in opposite directions. This allows reconstruction of particle tracks in all three dimensions, with the highest granularity in the x direction, which corresponds to the bending direction of the magnet and is thus most relevant for momentum reconstruction. The inner regions of all layers have a higher density of silicon strips than the outer regions of the detector. This is done to compensate for the higher track density in these regions.

Scintillating Fibre Tracker

The SciFi Tracker is positioned downstream of the magnet. A schematic of one SciFi Tracker station is shown in Figure 3.3. The SciFi Tracker consists of tightly packed layers of scintillating fibres. When a particle passes through the fibres, the atoms are excited and emit light as they return to their ground state. The fibres consist of a scintillation core surrounded by a cladding material that reflects the light back into the fibre. This way, the light gets transported to the silicon photomultipliers (SiPMs) with minimal losses.

The fibres in the stations are packed into fibre modules with a length of 4850 mm and a width of 523 mm, each consisting of eight fibre mats. A mirror in the middle of a module is used to reflect the light towards the SiPMs, positioned at the top and bottom of the modules, to increase the light yield. The modules are arranged over three stations $T1$ to $T3$. Every station consists of four layers, using the same X, U, V, X geometry as the UT. The granularity of the SciFi Tracker in the bending plane of the magnet is 250 µm, resulting in a resolution below 100 µm.

3.1.2 Particle identification

At the LHCb experiment, specialised detectors are used to perform particle identification (PID) [52]. The detectors for PID are two Ring Imaging Cherenkov detectors (RICH detectors) [53], the electromagnetic calorimeter (ECAL) and hadron calorimeter (HCAL) [54], and four muon stations (M2 to M5) [55].

The RICH detectors make use of Cherenkov radiation. Cherenkov radiation is emitted when charged particles travel through a radiator medium faster than the speed of light in the medium. The Cherenkov radiation is emitted in a cone shape from the position of the traversing particle. The opening angle of the cone depends on the velocity of the particles. In combination with the momentum information, the masses of the particles can be calculated.

Photons and electrons can be identified by exploiting the fact that they deposit all their energy in the ECAL. Photons can be differentiated from electrons because they do not leave a track in tracking systems due to their lack of charge. The energy deposition of the electrons must match the momentum information obtained from the tracking system.

Charged hadrons can be differentiated from neutral hadrons analogously, except that their energy signature is in both calorimeters. Different hadrons can be identified by measuring the dependence of the energy loss in the HCAL on the momenta of the particles, as provided by the tracking system. The signature signal depends on the mass and velocity of the particle; therefore, the mass can be inferred.

Muons do not deposit much of their energy when they interact with detector material. Therefore, they pass through the entire detector and leave a signature in the muon stations. Other particles are stopped by absorbers between the stations.

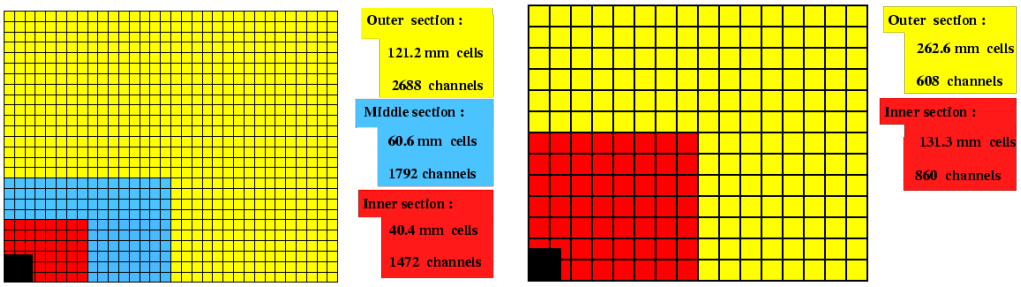


Figure 3.4: Schematic of the module placement and the regions of the ECAL (left) and the HCAL (right). Taken from reference [16].

Ring Imaging Cherenkov detectors

The RICH detectors determine the velocity of particles by measuring Cherenkov light over a set of mirrors with SiPMs. The velocity of the particles can be determined by measuring the opening angle. The opening angle θ of the cone is given by $\cos(\theta) = 1/(n\beta)$, where β is the speed of the particle relative to the speed of light and n is the refractive index of the material.

The two RICH detectors use different radiators to cover different wavelengths between 200 nm and 600 nm, corresponding to different momentum ranges covered by the detectors. The RICH1 detector is placed upstream of the magnet and identifies charged particles within a momentum range from 1 GeV to 60 GeV. The RICH2 detector is placed downstream of the magnet and covers a momentum range from 15 GeV to 100 GeV.

Calorimeters

The two calorimeters measure particle energy deposition using layers of sensitive scintillating material. The total energy is proportional to the amount of light emitted. The light is guided through optical fibres to the readout photomultiplier tubes. Absorbers between layers are used to reduce particle energy, ensuring the electromagnetic shower is fully contained.

A schematic of the ECAL and HCAL is given in Figure 3.4. The detectors are divided into multiple regions each. For the ECAL, it is distinguished between the inner, the middle and the outer region. For the HCAL, it is only distinguished between the inner and outer region. The calorimeter modules are arranged with higher density toward the beam centre to compensate for the higher occupancy there.

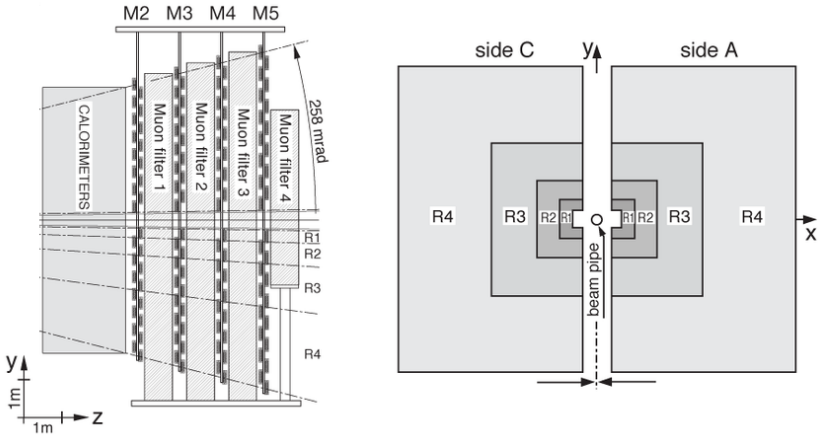


Figure 3.5: Schematic of the side view of the muon stations with respect to the calorimeters (left) and a schematic of the four regions of the front view of a station (right). Taken from reference [52].

Muon chambers

Muons are identified by four muon stations, each consisting of multiwire proportional chambers. Passing muons interact with particles, ionising the gas. High-voltage electrodes in the gas collect the ions and electrons. Using the drift time and signal strength, the muon position and transverse momentum can be determined.

A scheme of the muon stations is given in Figure 3.5. As with the calorimeters, the density of the wire chambers is non-uniform, and all stations are split into four regions based on their distance from the beam centre. The left side of the figure also depicts the absorbers (muon filters) between the stations, meant to stop all particles except muons.

3.1.3 Luminosity

The instantaneous luminosity $\mathcal{L}_{\text{inst}} = f_{\text{rev}}(\mu_{\text{vis}}/\sigma_{\text{vis}})$ describes the number of particles produced per time interval and cross-section. It is given by the revolution frequency $f_{\text{rev}} = 11\,245\,\text{Hz}$ of the LHC, the number of visible proton-proton interactions mu_{vis} at the IP and the visible production cross-section σ_{vis}. At the LHCb experiment, the beam separation at the IP can be controlled. This is used for luminosity levelling, where particles are not collided head-on but are separated so that only a fraction of them collide, allowing control over μ_{vis}. Since the upgrade, the value of μ_{vis} is steadily increased from former $\mu_{\text{vis}} = 1$ to the upgrade design value of $\mu_{\text{vis}} = 5$.

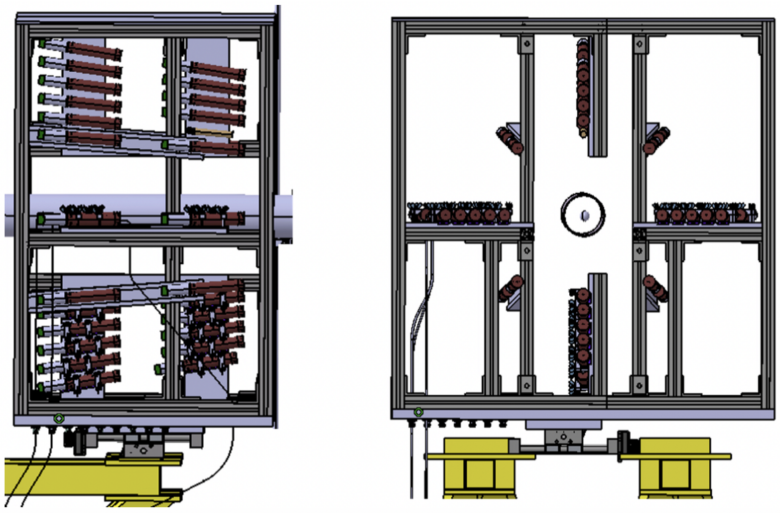


Figure 3.6: Schematic of the PLUME detector viewed from the side (left) and from the front (right). Taken from reference [56].

A dedicated subdetector Probe for Luminosity Measurement (PLUME) [56] is used to measure the luminosity and provide it to the experiment and the LHC. A schematic of PLUME is shown in Figure 3.6. PLUME consists of 48 photomultiplier tubes arranged in two layers around the beam pipe, detecting Cherenkov light produced by particles passing through a quartz material. The luminosity is determined by the number of counts detected in the tubes. Additionally, various counters associated with the subdetectors provide luminosity measurements by leveraging the linearity between luminosity and the number of clusters or tracks within the subdetectors.

3.2 Operating the experiment

To operate the LHCb experiment, detector data must be read out and made available for physics analysis. The so-called online architecture implements the hardware, software, and infrastructure for data acquisition (DAQ), Timing and Fast Control (TFC), and the control of the experiment via the Experiment Control System (ECS) [16, 57]. The DAQ system is standardised for all subdetectors and synchronised with the clock distributed by the TFC. Everything is connected to, controlled and monitored by the ECS. Figure 3.7 shows the scope of the online system.

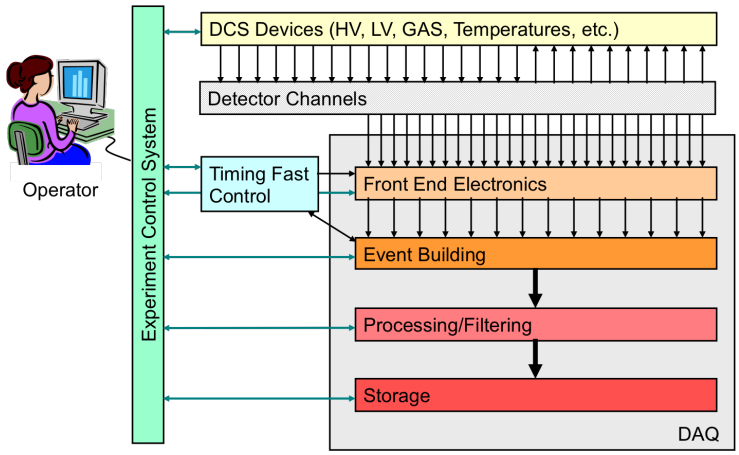


Figure 3.7: Schematic of the LHCb online architecture. The components are described in their respective sections. Taken from reference [16].

3.2.1 Data acquisition

The data is read out from the LHCb detector at every bunch crossing at 40 MHz. To reduce data volume, it is filtered by a trigger, selecting only data relevant to physics analysis. Since the upgrade, all front-end electronics send out their data without reducing it by a hardware-based low-level trigger. Instead, the data is reduced at a later stage in software only, in the High Level Trigger (HLT) [58]. Standardised back-end boards (TELL40 boards) receive data from the front-end electronics, hosted on event builder servers that run the HLT sequence.

The trigger is processing the data at an effective rate of 30 MHz, since not all bunches in the machine are filled (see the abort gap in Section 2.2.5). In the first step, partial detector reconstruction and pre-selection of the data are performed in the HLT1 application, reducing the rate to a few MHz. Complete reconstruction is done in the HLT2 application, reducing the data further by a factor of ten. The final events after processing and filtering are saved to permanent storage.

3.2.2 Timing and Fast Control

The TFC is responsible for synchronising the subcomponents to the LHC clock to allow the readout at 40 MHz. The clock allows associating subdetector data with the correct bunch crossing during event building. The LHC clock acts as the master clock of all electronics. All commands related to the DAQ are distributed by the TFC system. The readout supervisor (SODIN) is the master component generating the information. The information is then distributed via standardised interface boards (SOL40 boards) to the front-end electronics.

3.2.3 The Experiment Control System

The LHCb experiment is controlled by a global ECS. It monitors, configures and controls the subcomponents of the detector. The Detector Control System (DCS) is responsible for controlling the subdetectors, like setting the high and low voltages. The DAQ and TFC systems are controlled and monitored by their respective components in the ECS. This includes the front-end electronics responsible for detector readout, the HLT responsible for data reduction, and other infrastructure (INFR.) components, such as the servers running the system. The ECS framework is explored in detail in Section 5.2.

The experiment is built on a tree-like structure that includes the various components. This allows for high independence and versatility. A simplified schematic of the tree structure is given in Figure 3.8. The tree nodes are either Control Unit (CU) or Device Unit (DU). The DUs drive the equipment, while the CUs are logical nodes that control and monitor the nodes below them. The tree structure is implemented by integrating the `SMI++` [59] language. This allows for object-oriented implementation of finite-state machines (FSMs). For each tree node, a user interface (UI) panel displays the relevant data for the component. The status and alarms of the subcomponents are propagated from the lower tree levels to the top, while commands are sent from the top to all subcomponents. This allows for automating all standard procedures and enabling quick error detection and recovery.

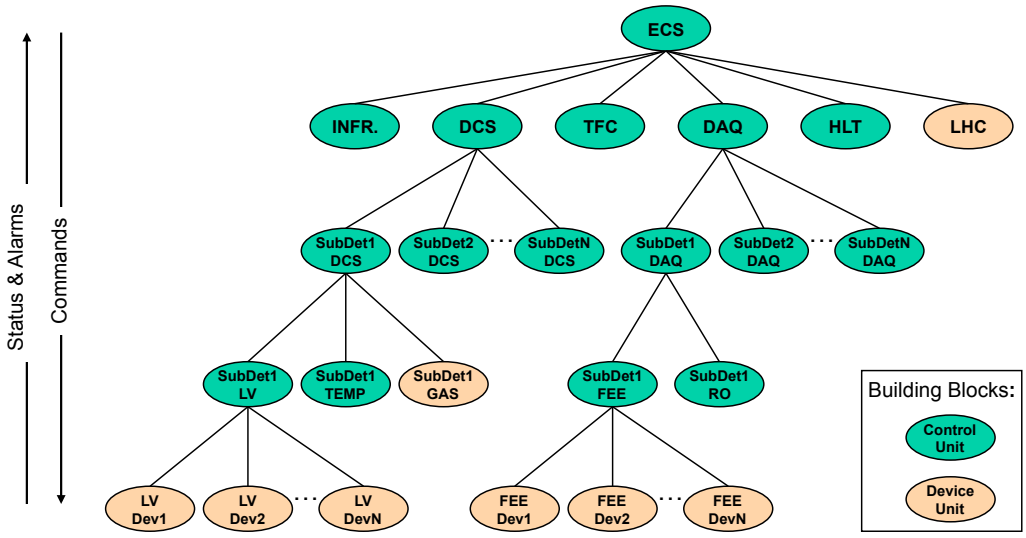


Figure 3.8: Schematic of the tree structure of the global LHCb ECS. Taken from reference [16].

4 Beam monitoring at the experiment

The Large Hadron Collider beauty (LHCb) detector components are exposed to radiation during operation. Should the radiation level increase drastically due to a critical beam scenario, such as beam misalignment, the sensitive parts of the LHCb detector might be damaged or destroyed.

A first line of defence to protect the experiment from stray particles is provided by the collimation system of the Large Hadron Collider (LHC) (refer to Section 2.2.4). The local tertiary collimation system of the LHCb experiment is briefly described in Section 4.1. The primary protection system of the LHCb detector is the Beam Conditions Monitor (BCM), specifically constructed to prevent damage to the detector components. It continuously monitors the beam conditions and triggers a beam abort in the event of adverse beam conditions, preventing damage to the detector. The BCM detector is described in Section 4.2. For a future addition to the monitoring, the Timepix4 telescope is acting as a prototype beam telescope for the upgrade detector. The Timepix4 telescope is described in Section 4.3.

4.1 Local collimation

As described in Section 2.2, halo particles from various sources accompany the beam and can reach the LHCb long straight section. They are cleaned by the tertiary collimators protecting the experiment. This produces secondary particle showers reaching the detector, resulting in machine-induced background (MIB).

The collimation system of the LHCb experiment consists of a vertical and a horizontal collimator for beam 1 and beam 2, respectively. The Target Collimator Tertiary Verticals (TCTVs) are positioned $\pm 116\,\mathrm{m}$ upstream and downstream of the interaction point (IP) for beam 1 and beam 2, respectively. The Target Collimator Tertiary Horizontals (TCTHs) are positioned $\pm 118\,\mathrm{m}$ upstream and downstream of the IP for beam 1 and beam 2, respectively. Beam Loss Monitors (BLMs) behind the collimators give information on the number of halo particles reaching the experiment.

The injection line for beam 2 is positioned 190 m downstream of the LHCb experiment. At this location, the injection kicker magnets are transferring the beam from the SPS into the LHC. Particle absorbers are placed behind the kicker magnets to protect the local aperture [60].

(a) BCM-U (b) BCM-D

Figure 4.1: Pictures of the BCM-U and BCM-D stations. The picture of the BCM-U station depicts one half of the station in its moved-out position.

4.2 The Beam Conditions Monitor

The BCM [12] consists of a BCM-U station upstream and a BCM-D station downstream of the IP, located at $z = -2131\,\text{mm}$ and $z = 2765\,\text{mm}$, respectively. The BCM-U station is mounted upstream of the Vertex Locator (VELO) on the concrete wall of the LHCb cavern, while the BCM-D station is mounted onto the beam pipe between the Upstream Tracker (UT) and the magnet (refer to Section 3.1). Both BCM stations are depicted in Figure 4.1.

Each BCM station consists of a set of eight polycrystalline chemical vapor deposited (pCVD) diamond sensors, symmetrically arranged around the beam pipe centre. Each sensor has a size of 10 mm by 10 mm and a thickness of 0.5 mm. The sensitive area of the BCM-U station begins at a radial distance of 50.5 mm, while that of the BCM-D station begins at a radial distance of 37.0 mm. The artificially created pCVD diamond material was chosen for its high radiation resistance and charge-collection capabilities, making it suitable for operation close to the beam pipe. The sensors are operated with a 200 V bias voltage.

The BCM detector was refurbished for Run III. All the sensors have been replaced. In addition, the BCM-U station mechanical support has been updated with two sliding halves for easier access, required by the inclusion of the Probe for Luminosity Measurement (PLUME) system in front of the BCM station. As a result, the sensors of the BCM-U station are rotated by 22.5° [61].

4.2.1 Front-end data acquisition

The detector outputs a current proportional to the energy deposited in the diamond sensors. The current is digitised by the front-end charge-to-frequency converter (CFC) cards [62]. They are the same cards originally used for the LHC BLMs. Due to their radiation hardness, they are suitable for operation with the BCM. A CFC card supports eight channels, resulting in one card per BCM station, for each of the diamond sensors.

A capacitor is charged for each diamond channel with an integration time of 40 µs. At a certain integrated charge, the capacitor is discharged. The CFC card is calibrated so that one integration cycle corresponds to a current of 5 µA. At the end of a measurement period, the integration level of the capacitor is sampled by a 12 bit analogue-to-digital converter (ADC). The number of discharges $n_{\text{discharge}}$, combined with the difference in two consecutive ADC values, gives the flux value of a sensor [13]

$$I(t_i) = 5\,\mu\text{A}\left(n_{\text{discharge}} + \frac{\texttt{ADC}(t_{i-1}) - \texttt{ADC}(t_i)}{\texttt{ADC}_{\text{range}}}\right), \tag{4.1}$$

where i denotes the measurement period. The $\texttt{ADC}_{\text{range}}$ used to normalise the ADC values depends on the electronics. It is slightly below the maximum theoretical 12 bit value of 4096 and determined for each channel individually.

The CFC card is operated at high irradiation levels with up to 400 Gy integrated dose during its lifetime. As irradiation increases, the integrator amplifier produces an offset current that must be actively compensated. For this purpose, a small current is injected by a digital-to-analogue converter (DAC). Its value steadily increases as the CFC card is powered until a current of 10 pA is measured.

The CFC card sends the data with 25 kHz in 40 byte data frames. Each data frame includes a 16 bit card identification number (CID) to identify the card. A 32 bit status value is used to indicate the health of the card. The bits include monitoring of correct voltage, temperature, and DAC values. The discharge and ADC counts for each sensor are packed as 8 bit and 12 bit values, respectively. The 8 bit DAC values for each channel are sent as well. A 16 bit frame identification number (FID) is incremented with each package, to detect lost packages during the transmission. Finally, a 32 bit cyclic redundancy check (CRC) is used for error detection in the data frame. The CRC-32/MPEG2 algorithm is used to calculate the CRC value from the bits of the data frame. By recalculating the value after transmission and comparing the CRC values, errors in the data frame can be detected. The data is sent via two redundant optical links to the back-end electronics.

4.2.2 Interface to other systems

The BCM is in constant communication with the accelerators Beam Interlock System (BIS) (refer to Section 2.2.5). The BCM is connected to the BIS via three beam interlock system user interface (CIBU) units. The first unit is used to provide the beam permit. The other two units provide the injection permits for beam 1 and beam 2, respectively. All signals are transmitted by two redundant links. In a failing-beam scenario, the BCM removes the beam permit signal, and the beam is dumped. The injection permit is used to indicate that beam injection to the LHC is permitted by the BCM. The BCM is interfaced with the Beam Interlock Controllers (BICs) at IR6, for the beam permit, IR2 for the injection permit of beam 1, and IR8 for the injection permit of beam 2.

To provide information on the dump cause, a diagnostic post-mortem system is implemented. The LHC sends a post-mortem Trigger (PMT) to all interfaced systems, including the BCM. The BCM keeps the most recent relevant data in transient storage. After the PMT is received, the data is saved to disk for later analysis.

The BCM is critical to ensure the safety of the VELO sensors. Therefore, in addition to providing the permits to the LHC, the BCM is interfaced with the VELO Safety System (VSS). The BCM continuously monitors its own health. In case the BCM is showing errors, or the status of the system becomes unknown, the `BCM OK` signal provided to the VSS is withdrawn. In this case, the VELO is moved into a safe state: the high voltage is ramped down, and the VELO is moved into its opened position. The beam permit is not withdrawn.

4.2.3 Beam dump logic

The BCM must reliably dump the beam to prevent permanent detector damage. At the same time, the detector should not dump the beam without good reason, as a dump wastes valuable time for all the physics experiments. The BCM implements the following dump logic [13], where both stations are treated independently of each other.

For each diamond sensor, a set of running sum (RS) is calculated over a different number n of timeframes of 40 µs. Each RSn is a summation over the n most recent current readouts. As n increases, outliers are smoothed out, reducing noise in the signal and statistical fluctuations. At the same time, the responsiveness of the algorithm decreases. Therefore, three abort criteria are provided to cover different use cases based on the responsiveness and stability of the signal.

The first abort criterion is the RS1 abort. It continuously monitors the RS1 value for each sensor. It is compared to a threshold. If a sensor current exceeds the threshold, it is flagged. When three adjacent sensors in one BCM station are flagged for two consecutive time frames, the beam is dumped. The triple spatial and the temporal redundancy reduce the risk of triggering a beam dump due to dark current fluctuations or noise.

If the threshold is exceeded in only one timeframe, the RS1 abort does not trigger. Therefore, the RS2 abort is introduced. The RS2 values are continuously compared to a threshold. Due to the nature of the RS, this value is also responsive to single bunch events, making it especially useful during the injection of the beam. The spatial coincidence with three neighbouring sensors is the same as for the RS1 abort.

To monitor the current on a larger time scale with reduced noise, a third RS32-SUM abort criterion is implemented. The fast criteria of the RS1 and RS2 require significantly higher thresholds to compensate for current fluctuations. The advantage of integrating over a more extended time period is that most of the noise is averaged out [63]. A so-called RS32-SUM is computed by adding up the RS32 values of the eight sensors of a station, leaving out the two highest values and the lowest value to compensate for outliers:

$$\text{RS32-SUM} = \sum_{i=0}^{4} \text{RS32}[i]. \tag{4.2}$$

For all abort criteria, different threshold values are set depending on the operation mode. Up to four different modes are supported in the firmware. Two of them are implemented: mode 0 during stable beams, and mode 3 during injection. This is done because the beam behaves very differently in these scenarios. Additionally, the VELO is retracted during injection, allowing higher thresholds without increasing the risk to the detector.

4.2.4 The Machine Interface Beam Abort Decision system

The Machine Interface Beam Abort Decision (MIBAD) [13] system implements the logic that determines the dump decision based on information received from the CFC cards. It implements the described abort criteria and sets or removes the permits to the CIBU. It monitors the BCM health and deasserts the `BCM OK` signal to the VSS. It receives the PMT and sends out the data for post-mortem analysis.

The MIBAD is based on a field-programmable gate array (FPGA), implementing the firmware logic. The data from the CFC cards is received by an optical mezzanine card. A custom mezzanine card is used to send the signals to the BIS and VSS. A standard redundant power supply powers the board.

Figure 4.2 depicts the toplevel structure of the MIBAD firmware components. The data is received from the CFC cards by four front-end link transceivers (XCVRs). They receive and deserialise the data. The assignment of the XCVR links to the stations is done over the CID of the CFC card by the router, which monitors the links. After the station is assigned, the data is processed. The health of the CFC card is verified by checking the status bits of the CFC frame. Further, the RS are calculated and compared to the thresholds according to the abort criteria. The beam permits and the `BCM OK` signal are set in the cavern interface.

Over the back-end link, the data is sent via twisted pair copper cables to the Experiment Control System (ECS), monitoring and controlling the MIBAD. Additionally, the data is sent via twisted-pair copper cables or optical fibres to the post-mortem system, storing the data in transient storage for post-mortem analysis. The data is preprocessed and packaged according to the Ethernet standard. For debugging purposes, a front-end emulator is implemented that can generate CFC data when no CFC card is connected.

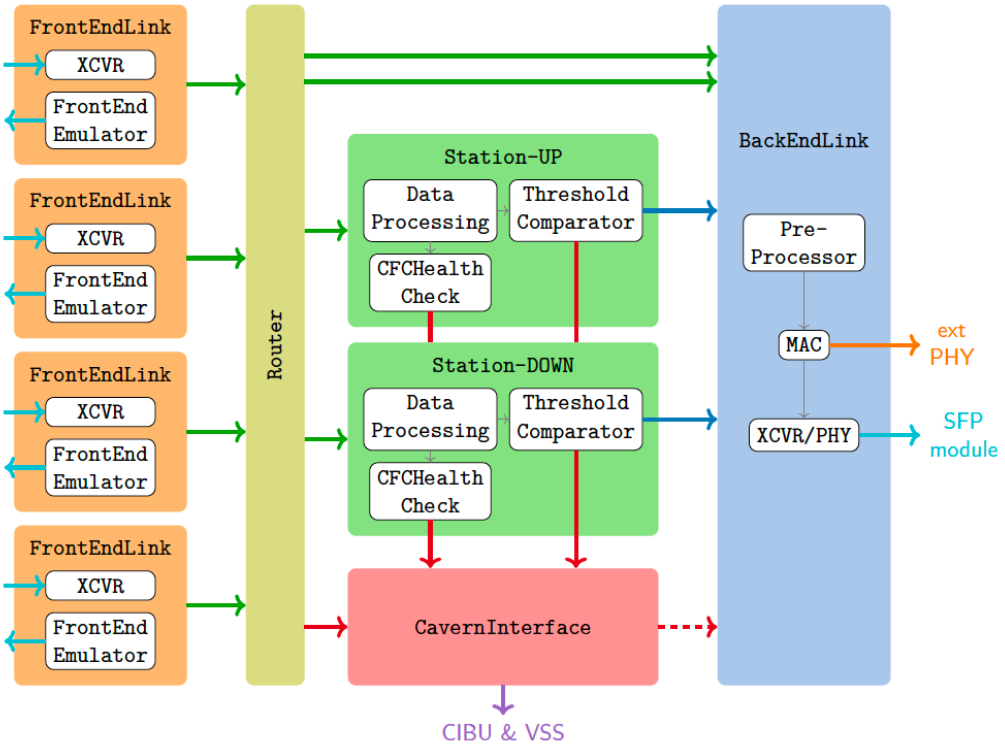


Figure 4.2: Schematic depiction of the main firmware components of the MIBAD system. Taken from reference [13].

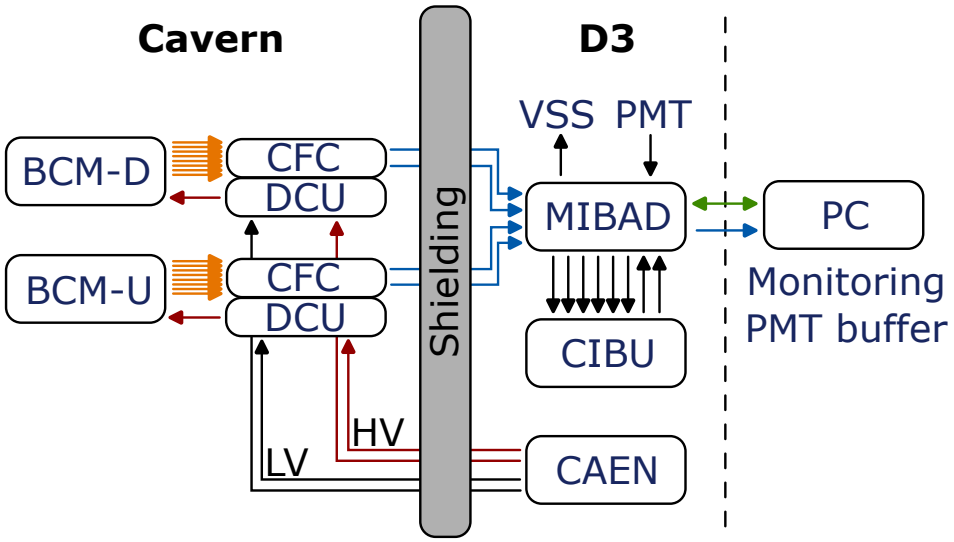


Figure 4.3: Schematic depiction of the BCM components. Twisted copper cables are orange, optical fibres blue, the Ethernet link is green, and the MIBAD cavern interface signals are black. The voltage lines are depicted in red for the high voltage and black for the low voltage.

4.2.5 Readout chain

The full readout chain of the BCM is given in Figure 4.3. The BCM stations are connected to the front-end CFC cards via twisted copper cables. The CFC cards are housed in a mini crate in the LHCb cavern near the detector stations. The mini crates have Diamond Connection Units (DCUs) attached, providing the 200 V bias voltage to the sensors. The CFC cards are connected to the MIBAD system via redundant optical fibres. The MIBAD is interfaced with the CIBU units, the VSS, and the post-mortem system, to provide the necessary signals. The MIBAD and CIBU are behind a 4 m shielding wall in the so-called D3 barrack. A CAEN [64] power supply crate provides the high and low voltages for the BCM system.

The monitoring of the MIBAD is done by a remote machine connected by a Gigabit Ethernet link. The data for post-mortem analysis can be transmitted by this link as well, or by a dedicated optical link. The post-mortem data is stored in a circular PMT buffer running on a server.

4.3 The Timepix4 telescope

A telescope-based design is under consideration for a future detector to enhance the beam-monitoring capabilities at the LHCb experiment. Telescopes are tracking detectors that precisely reconstruct particle tracks using multiple sensor planes. The detector will enhance the beam-monitoring capabilities by reconstructing tracks from the collision and interaction region. The information is then used to reconstruct the beam conditions at the IP.

In an early stage, the Timepix4 telescope [14, 15], successor to the Timepix3 telescope [65], is used as a development baseline and prototype for the detector. The Timepix4 application-specific integrated circuit (ASIC) specifications and the layout of the Timepix4 telescope are described in this section.

4.3.1 Specifications

The Timepix4 ASIC is the successor to the Timepix3 ASIC. The Timepix4 ASIC can simultaneously measure the Time of Arrival (ToA) and the Time over Threshold (ToT) of the particles. The former relates to the timestamp, and the latter to the charge collected. For the Timepix4 ASIC the ToA binning resolution is 195 ps, and the ToT energy resolution is $< 1\,\mathrm{keV}$. This makes it an excellent prototype for future four-dimensional tracking detectors. The main characteristics for the Timepix4 ASICs are listed in Table 4.1. The values of its predecessor, the Timepix3 ASIC, are included for reference.

Table 4.1: Properties of the Timepix3 and Timepix4 ASIC.

	Timepix3 (2013)	Timepix4 (2019)
Technology	130 nm	65 nm
Pixel size	$55 \times 55\,\mu\mathrm{m}$	$55 \times 55\,\mu\mathrm{m}$
Pixel matrix	256×256	512×448
Sensitive area	$1.98\,\mathrm{cm}^2$	$6.94\,\mathrm{cm}^2$
Event packet	48 bit	64 bit
Max rate	$0.43 \times 10^6\,\mathrm{mm}^{-2}\,\mathrm{s}^{-1}$	$3.58 \times 10^6\,\mathrm{mm}^{-2}\,\mathrm{s}^{-1}$
Max rate per pixel	1.3 kHz/pixel	10.8 kHz/pixel
ToT energy resolution	$< 2\,\mathrm{keV}$	$< 1\,\mathrm{keV}$
ToA binning resolution	1.56 ns	195 ps
ToA dynamic range	409.6 µs	1.638 µs
Readout bandwidth	$\leq 5.12\,\mathrm{GB}$	$\leq 164\,\mathrm{GB}$
Target global minimum threshold	$< 500\,\mathrm{e}^-$	$< 500\,\mathrm{e}^-$

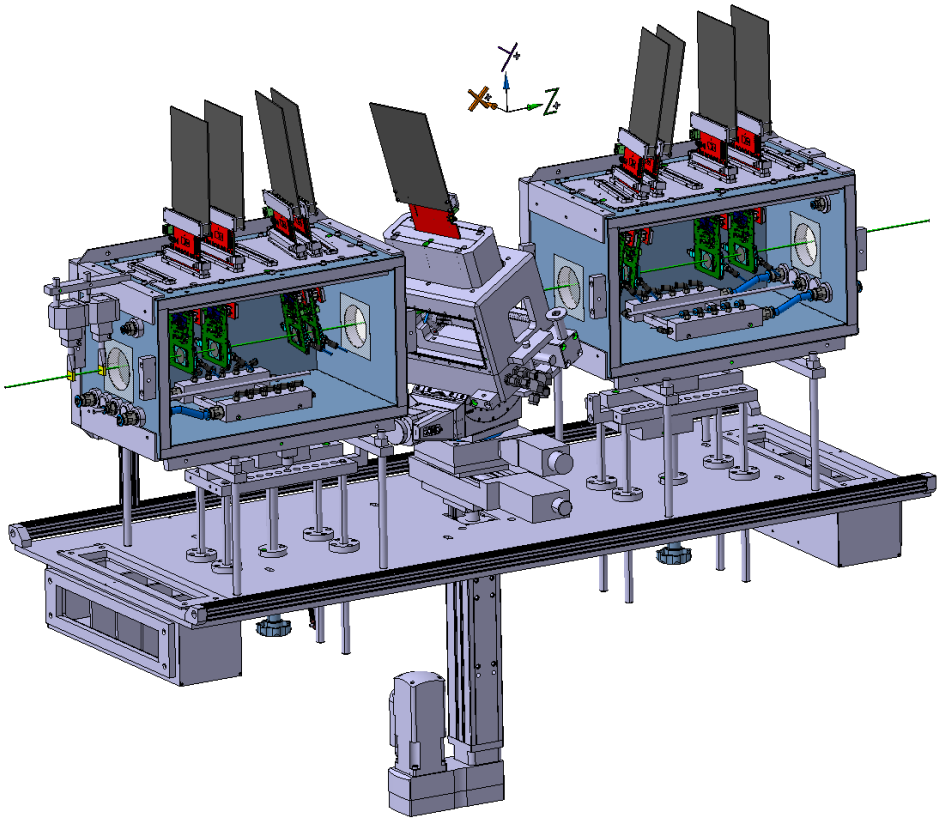


Figure 4.4: Schematic of the Timepix4 telescope. Taken from reference [14].

4.3.2 Layout

The Timepix4 telescope is operated at the Super Proton Synchrotron (SPS) test beam facility in the north area of the LHC. A mixture of 100 nm and 300 nm thick n-on-p planar silicon pixel sensors is bump bonded to Timepix4 ASICs. A device under test (DUT), typically a prototype for a sensor, is characterised by comparing its response with the information provided by the telescope.

A schematic view of the Timepix4 telescope is given in Figure 4.4. A right-handed coordinate system is used, where the y axis points upwards, and the z axis points in the direction of the beam. The telescope consists of two arms with four sensor planes each. Each telescope arm is enclosed in a dry-air-filled container. The sensor planes are water-cooled to around 18 °C. The telescope uses two planes with 300 nm thick sensors angled 9° both in x and y direction optimised for spatial resolution, and two planes with 100 nm thick sensors placed perpendicular to the beam optimised for temporal resolution.

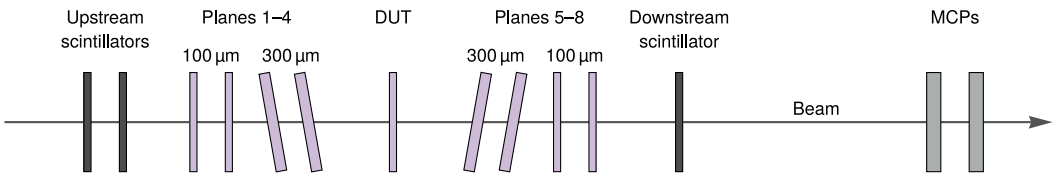


Figure 4.5: Schematic of the arrangement of the Timepix4 telescope sensor planes, the DUT, MCPs and scintillators with respect to the beam. Adapted from reference [14].

The DUT can be mounted at the centre of the telescope, between the telescope arms, referred to as the centre stage. The telescope also provides an outer stage at the back, where larger DUTs can be mounted. A vacuum box can be installed on the centre stage to test irradiated devices at high bias. The main telescope motion stage can be moved in the x and y directions to align the telescope with the beam. The telescope arms are mounted on rails, allowing translation along the z direction to increase or decrease the distance from the DUT. The DUT is mounted on translation stages that allow it to move in the x and y directions to perform scans. The DUT is additionally mounted on rotation stages that enable it to be rotated and tilted around the x and y axes.

For the data acquisition (DAQ), the sensor planes are read out by the fourth-generation Speedy Pixel Detector Readout 4 (SPIDR4) boards. With one SPIDR4 board per sensor plane, this results in four SPIDR4 boards per telescope arm. The SPIDR4 board was designed to handle the full bandwidth of 160 Gbit s^{-1} of the Timepix4 ASIC.

Positioned downstream of the telescope, two microchannel plates (MCPs) are used to provide a timing reference for the telescope. The MCPs have a time resolution of 5 ps. Three plastic scintillators, with individual time resolutions of 120 ps to 150 ps, mounted on the enclosing boxes of the telescope arms, complement the measurements of the MCPs. The arrangement of the telescope components is shown in Figure 4.5.

5 Development of Experiment Control Systems

To operate any experiment successfully, it is necessary not only to develop the detector hardware but also to have precise control and feedback on the system status. This is achieved by implementing an Experiment Control System (ECS). The ECS enables the operator to set different configurations for the experiment, monitor the status of all subcomponents, and automatically respond to and manage actions triggered by events during the operation of the experiment.

General considerations for designing an ECS are given in Section 5.1. The framework used to build the control systems is described in Section 5.2. The WinCC Open Architecture (WinCC OA) software suite [66] is introduced, and general concepts are explained. Section 5.3 describes communication between the software and the hardware, with a focus on the software implementation. The Beam Conditions Monitor (BCM) ECS and the Timepix4 telescope ECS are developed. Their implementation is described in dedicated chapters, in Chapters 6 and 7, respectively.

5.1 General considerations

The ECS must be reliable and robust in operation. At the same time, the ECS must be efficient, responsive, and easy to operate. In this section, general design considerations are given, taking into account the purpose of the ECS for use at a large experiment and the fundamental workings of the underlying framework. An insight into event-based and parallel code execution is provided.

5.1.1 Requirements

Every ECS has different requirements. However, there are a few standards applicable to all ECS. The health state of the ECS needs to be continuously monitored to quickly address system errors. Therefore, errors should be propagated and made visible as they occur. Any change in the system state needs to be apparent. For robustness, automatisms should be considered, where possible, to recover the system into a healthy state.

While at the software level, a more complex but efficient and robust implementation is preferable, at the operator level, the most intuitive solution is preferable. Making the ECS intuitive reduces the risk of user error. Regarding the layout of the information, related data should be displayed close together, and groups and tabs can be used to organise it. Graphical representation of the data is preferred, where applicable.

For efficiency, independent tasks should be executed in parallel. Bit and pointer operations can improve memory usage and speed. The ECS requires taking action in response to external input, whether an operator executes a command or the system changes state. Therefore, asynchronous, event-driven code execution is required to handle the simultaneous tasks. This is elaborated on in the following part of this section.

5.1.2 Event driven code execution

Event-driven code execution in ECS systems is an essential concept throughout and differs from the classical linear programming approach. Instead of executing commands consecutively, external actions trigger an event that executes the code associated with it. These actions can be triggered by other functions, by changes in the controlled and monitored hardware, or by user input.

The concept of event-based code execution is visualised in figure 5.1. An input action is triggered. This is the event source. An event listener is subscribed to the event source and creates a task. The task is added to an event queue, which stores the tasks for further processing. An event loop cycles through the tasks in the queue and passes them to the corresponding event handler. The event handler is a callback function that executes the code.

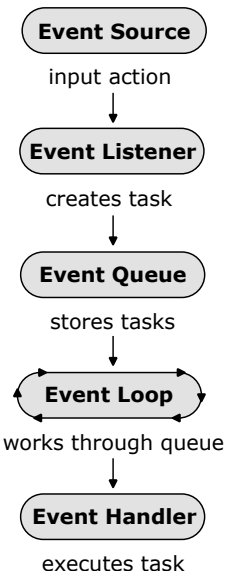


Figure 5.1: Event-driven code execution.

For event-driven code execution, care must be taken when a task triggers another event. This event can, in turn, trigger the original event, causing an infinite loop. This becomes exponentially more convoluted as the number of connected events increases. Furthermore, tasks can operate on the same memory. If two or more processes read or write the same memory, the order in which read and write operations are performed matters. If no hierarchy is defined, this can lead to undefined behaviour.

5.1.3 Parallel code execution

When optimising performance, parallel code execution can increase speed by executing tasks independently in their own processes, each with local computing resources. Therefore, executing independent events in parallel is a good way to increase the processing speed and responsiveness of the ECS.

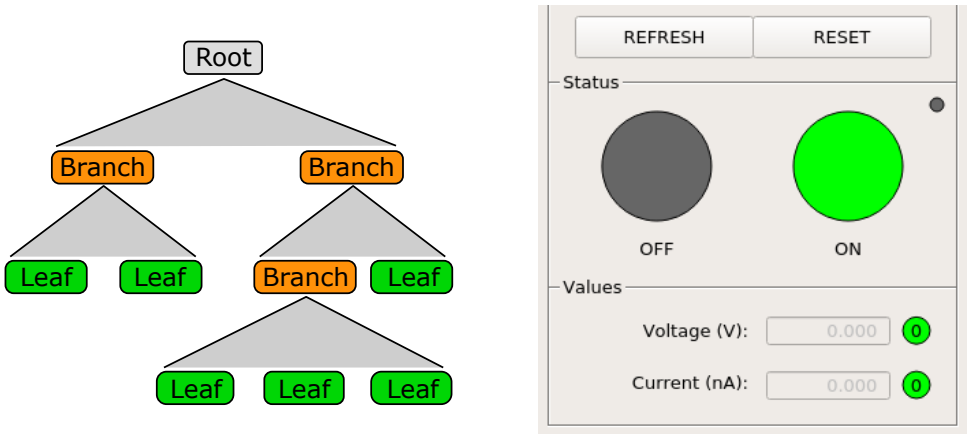
However, some of the events are not entirely independent of each other and instead share resources. This shared memory is needed if the processes, while in principle not dependent on each other, are working with the same information of the underlying hardware. Writing and reading memory at the same time causes undefined behaviour.

Therefore, locks are used to claim the memory for the process currently accessing the data and to prohibit other processes from accessing the same memory at the same time. The locks work on a first-come, first-served basis. Another process can change memory during code execution if a lock is not taken at the appropriate time. This, in turn, can cause the results of a task to change, dependent on another task, potentially leading to undesired results. It is thus of great importance to create a well-defined structure for the ECS to prevent undefined and undesired behaviour during operation.

5.2 Control system framework

WinCC OA is used to build the ECSs. The entire infrastructure is built on its design, enabling a unified system and feel across the experiments [67, 68]. This removes the need to reimplement standard procedures, as they can be implemented in separate components that are then shared across projects.

The framework implemented for this purpose is the Joint Controls Project (JCOP) framework [69]. Components are included in a project over a framework installation tool. It is installed individually for each project and allows for easy management of the JCOP components. Components outside the framework can be developed and included easily via the framework manager. A component contains all the control scripts, libraries, user interfaces, and other requirements, depending on the use case. This provides the added advantage of upgrading a project by reinstalling the underlying packages.



(a) DP tree (b) example panel

Figure 5.2: Visualisation of a DP tree structure and a basic panel.

5.2.1 Data management

WinCC OA is based on a data point (DP) system. All information of a device is stored in DPs. This allows for device description and access by setting and reading from DP elements. Alarm handling and archiving, described later on, can be performed by monitoring the DP values. WinCC OA provides components to visualise DP element values and to allow operator interaction via graphical widgets.

Data points

To create the DPs, first a DP type is defined. This DP type serves as a template to create the DPs. Every DP type is unique, but many DPs can be created from the same DP type. Within a DP, variables are stored in a tree hierarchy. The variables are equivalent to variables known in other common programming languages, like `bool`, `int` or `float`. The tree hierarchy is visualised in Figure 5.2 (a). The top tree node is the *root*. Every subsequent *branch node* has both a parent and a child node defined. A branch node can have any number of children, but only one parent node. If no children are defined, the nodes are referred to as *leaf nodes*. The depth of the tree is variable. Branch nodes can contain further branch nodes. Leave nodes are final.

Data visualisation

Functionalities are provided to trigger an event on change of a DP value. The functions on state change are used to connect the DP values to so-called interface widgets. These widgets can be text fields, tables, shapes, or graphs. They are used to visualise data in a way that is easily comprehensible to the end user on an operator panel. A panel is a space where all the widgets are displayed together.

Important widgets are those that take user input, such as buttons, allowing the user to trigger actions like setting values or navigating the panel hierarchy. An example panel is shown in Figure 5.2 (b) showing buttons to perform operations on the controlled hardware, circles used to indicate the status, and textfields to display hardware DP values. The functionality behind the widgets is implemented in their respective widget control script. The code for the widgets is executed based on the panel state, unlike general code that runs behind the scenes.

Managers

WinCC OA operates on an event basis (refer to Section 5.1.2) using managers. Usually, the events are triggered by user input or changes in the DP elements; however, timed execution is also possible. There are four layers of managers: the driver layer, the communication and memory layer, the processing layer and the user interface (UI) layer [68]. The managers are shown in Figure 5.3.

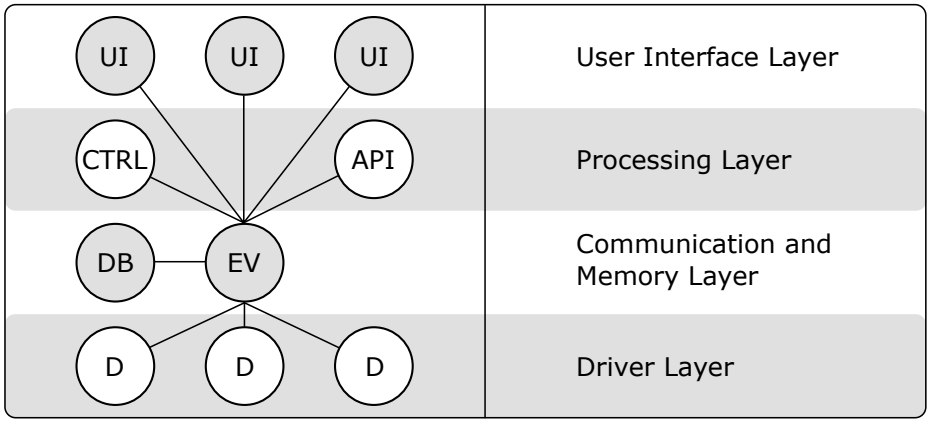


Figure 5.3: Schematic overview of the WinCC OA managers, with the diver manager (D), event manager (EV), database manager (DB), control manager (CTRL), API manager (API), and UI manager (UI). Based on reference [68].

The driver layer includes driver managers that handle communication with the hardware. This allows reading and setting the DP element values from the hardware side. The data exchange between the managers is event-driven. The event manager handles this in the communication and memory layer. It can be seen as the main manager. In the same layer is the database manager. Its task is to manage and store the DP values in a database. This is done by exporting the data at a fixed time specified in the database options. The processing layer executes scripts through control managers. They are executed by events handled by the event manager. WinCC OA uses a `C` like coding language. When triggered, the code is sequentially executed. It is possible to run multiple control managers simultaneously, as each manager starts in its own thread. The application programming interface (API) manager allows the inclusion of external libraries into the project. The last layer is the UI layer. The UI managers are responsible for the graphical representation and for executing all panel scripts.

System types

When creating a project, a system type is specified. It generally defines how data is managed within and between systems. It is distinguished between a single system, a scattered system, and a redundant system. The system types are visualised in Figure 5.4.

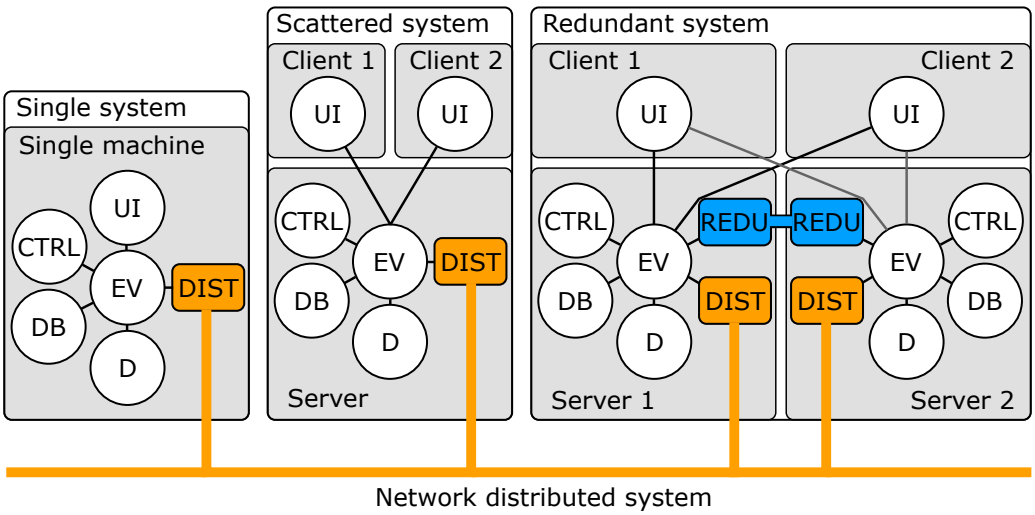


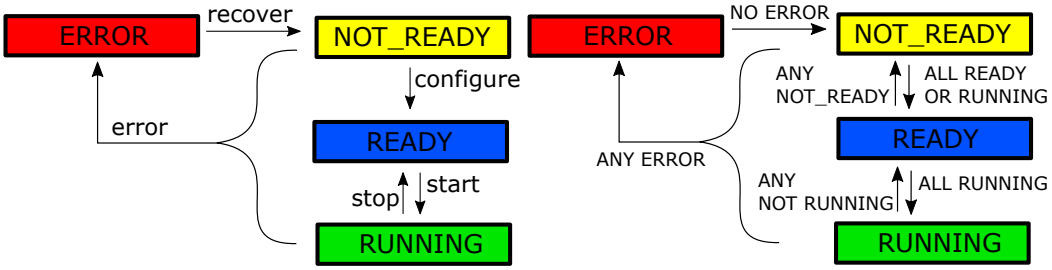
Figure 5.4: Schematic overview of the WinCC OA system types: the single, the scattered, the redundant, and the distributed system. The managers are as given in Figure 5.3, with the additional redundancy manager (REDU) and distribution manager (DIST). Based on reference [68].

In the single-system case, everything is set up on a single machine. The operator will use this machine to communicate with the hardware. In the scattered system, the main project is set up on a server, and the user interface is available across multiple operator devices that communicate with the server via the event manager. This way, the server can spread the processing load. The redundant system is an extension of the scattered system. In this case, a second server is set up as a clone of the first server. In the event of a primary server failure, the project will automatically switch to the secondary server. The redundancy is handled by a redundancy manager. Additionally, the project can be set up as a distributed system. This will add a distribution manager to the project. It enables data exchange between multiple autonomous systems.

5.2.2 Finite-state machines

To control the data flow and give structure to the ECS, finite-state machines (FSMs) are implemented. Each subcomponent of the experiment has to be in a defined state at any given time. The state changes according to the rules that restrict state transitions. The transition rules can be hardware-related (like a critical change in temperature) or software-related (like a change in throughput).

There are two types of FSM units: Device Units (DUs), which operate on the hardware state directly, and Control Units (CUs), which operate on the states of their children. Each state specifies the actions that can be triggered in that state. Actions triggered on a parent state can also be triggered on its children. Figure 5.5 (a) shows a basic example FSM DU with four states: `NOT_READY`, `READY`, `RUNNING` and `ERROR`.



(a) child FSM (b) parent FSM

Figure 5.5: Schematic of a simple FSM. The child FSM is a device unit that operates directly on a hardware state. The parent FSM is a logical control unit operating on its child states.

The FSM is initialised in its default state NOT_READY. An external action is required to configure the system. All the hardware attributes and configurations are set correctly. Only then does the state transition to READY. If everything is fine, data taking can begin, and the state transitions to RUNNING.

If, instead, in any state an error is detected, the FSM will go into the ERROR state. To get out of ERROR and back to RUNNING, either an automated routine recovers from the error, or an external recovery action must be performed on the system. Dependent on the state of the system, the FSM will transition into one of the non-error states after recovery.

To reduce complexity, the possible transitions are limited to a minimum. All non-error states can transition into ERROR, but the transition from the ERROR state is only allowed into the NOT_READY state. For once, data taking is stopped due to the error, so it is not sensible to return to RUNNING immediately after recovery. Additionally, the configuration requirements might have changed, so it is better to transition through the default NOT_READY state with the fewest requirements. This simplifies the flow chart by reducing unnecessary redundancy. The transition rules from NOT_READY to READY to RUNNING automatically take care of the transition into the correct state.

Figure 5.5 (b) shows the states for a parent CU, where all the children are FSM DUs as in the previous example. The state of the parent FSM is dependent on all children, so the transition rules are made of logical operands. The whole system is RUNNING only when all children are RUNNING (logical AND). The system is in ERROR if any of the children are in ERROR (logical OR). The system is in READY if all children are either RUNNING or READY and not all are RUNNING (combination of different logical operands), and so on. A FSM gets exponentially more complex with the number of states added to the system. Therefore, the number of states should be kept as simple as necessary. Maintaining a good flow between states by minimising allowed transitions is essential to reducing complexity.

5.2.3 Alarm handling

To ensure safe detector operation, alarms can be implemented to alert the operator when a value exceeds a defined alarm range. Alarms are handled via alarm classes with an assigned priority indicating the severity of the triggered alarm. Two of them are introduced. The first is the warning class triggered at 80 % of the limit value. The second is the alarm class triggered at the full limit. The alarm classes are assigned to value ranges defined by the limits for each class. The concept is shown on the right side of Figure 5.6.

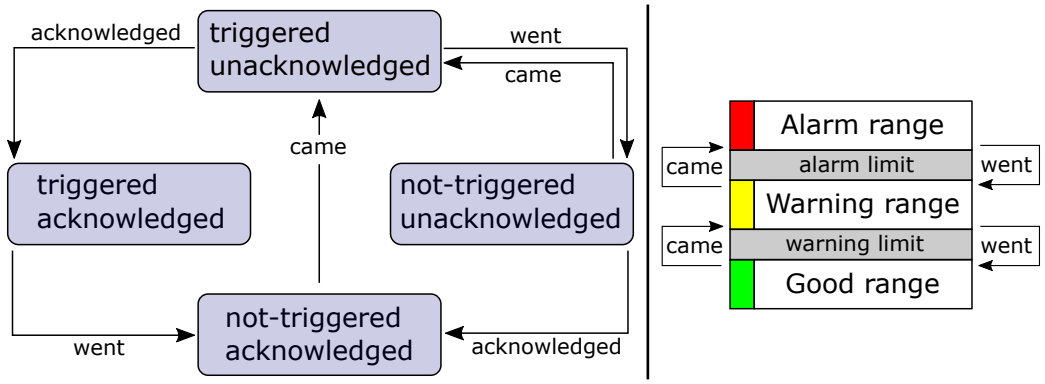


Figure 5.6: Visualisation of the alarm handling in WinCC OA. The alarm states (left) and the alarm ranges (right) are depicted. Based on reference [68].

The alarm class includes a colour code for each possible state. The possible states for an alarm are `triggered-unacknowledged`, `triggered-acknowledged`, `not-triggered-unacknowledged` and `not-triggered-acknowledged`. The alarm states and the transition between the states are shown on the left side of Figure 5.6. The `triggered` state is based on the limit set on the monitored value. The limit can be a fixed value, or a hysteresis band can be defined around it to prevent the alarm from constantly triggering when the signal fluctuates. The alarm can be activated and deactivated by disabling the limit. The `acknowledged` state indicates whether the operator has acknowledged the alarm. There are three possible transitions between the states. The three transitions are `came`, `went` and `acknowledged`. For each transition, a message can be set, or a specific action can be implemented.

5.2.4 Data logging

DP values are exported to the hard drive or uploaded to a cloud file system at regular intervals. This preserves the relevant information for future data analysis. The database managers handle data logging. Archiving can be configured to include the DP element values that are supposed to be logged. To reduce the data size, data smoothing can be used. Data smoothing means that not every value is saved to disk. If, for example, the smoothing is set to save data only on change, all values in between are ignored.

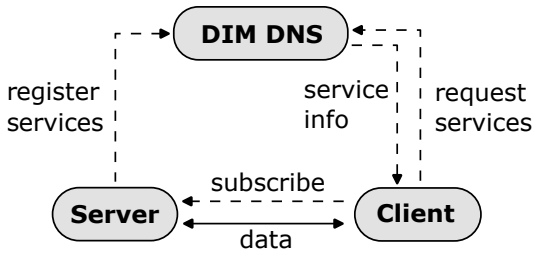


Figure 5.7: Visualisation of the data flow between the DIM components. Inspired by reference [71]

5.3 Communication with hardware

To make the data from the hardware accessible to the user, it must be transferred from the hardware to the software. For this purpose, communication protocols are introduced. The communication protocols adopted throughout are the OPC Unified Architecture (OPC UA) [70] and the Distributed Information Management (DIM) system [71]. In this section, the focus is on DIM, but OPC UA works analogously. The software implementation to connect the hardware over DIM to the ECS is provided.

5.3.1 Distributed Information Management

DIM was developed to enable efficient, reliable communication across platforms for large data transfers. DIM is a server-client communication protocol. The server provides services to the client that the client can subscribe to. There are four types of subscriptions usually of interest: a once only request from the client to the server to provide information of a value; a timed subscription, where the server provides a value to the client at a regular interval; a monitored subscription, where the server will provide updated information on a value to the client, whenever the value changes; and a command service, where the client can execute a callback function on the server.

The request for information based on a subscription allows for sending the subscription request only once at the beginning. The data is then retrieved as specified during the initial subscription. Meanwhile, the client can continue with other tasks without waiting for the response of the server, as would be the case if the client were to send a request each time. Giving the server the power to update all values whenever they change prevents unnecessary client-side requests. Over the command service, clients can perform actions on the server.

The primary data flow of DIM is visualised in Figure 5.7. The server registers the DIM services with a Domain Name System (DNS) during initialisation. Every service has a unique identifier, and the DNS maintains a directory of all available services across all servers in the system. The client requests a subscription to a server service from the DNS over the unique identifier. The DNS assigns the corresponding server to the client, if the requested service exists. The client then establishes a direct connection to the server for the data transfer. This includes the server sending data to the client and executing commands on the server. To execute commands on the server, callback functions are defined on the server side that are executed when the client calls the command.

5.3.2 Control system integration

WinCC OA communicates with the hardware on a server-client basis as introduced in the previous section. The WinCC OA specific implementation is visualised in Figure 5.8. The communication protocols are used to read and set the DP elements from the hardware side. The JCOP framework provides packages that allow setting up WinCC OA as a client or server in the communication protocol. The server provides services to which the client can subscribe and commands that the client can execute. Communication with the hardware is handled via drivers. They are specifically written to allow communication between the hardware and the operating system.

In Figure 5.8 WinCC OA is set up as the client and the hardware side is set up as the server. The DP elements are defined with two different attributes: `actual` and `setting`. The `actual` DP elements are intended to represent the actual values of the device. They are displayed on the panels and set by the WinCC OA client by subscribing to the values of the server. The server will update the values at a fixed time interval by reading them from the hardware. The `setting` DP elements are set from the WinCC OA panels. The value is read by the WinCC OA client, which, in turn, executes a server command. This can be used to assign values to the device or request the execution of an action.

5.3.3 Communication server

A Python-based communication server [19] is used to communicate with the hardware over DIM for both ECS of this work, adapted to the specific needs of the systems. The general server structure is described in this section. The specific changes to the use case of the ECS systems are described in their respective chapters.

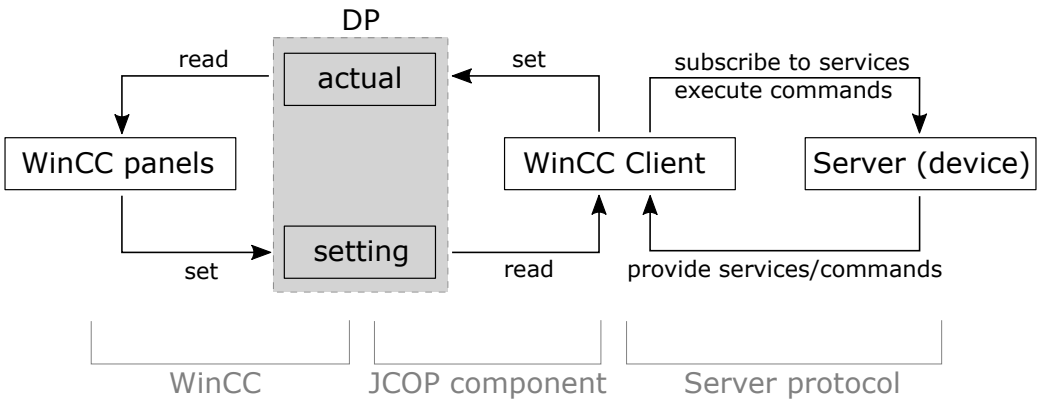


Figure 5.8: Visualisation of the communication between the server (device) and WinCC OA via a communication protocol.

The server is implemented in Python [72]. The class heavily relies on the PyDIM package [73], and the asyncio package [74]. The PyDIM package brings the DIM functionalities to Python. Figure 5.9 visualises the concept of executing a command on or reading a value from the server over its callback functions. The server communicates with WinCC OA over DIM as described in the previous section. The server registers services and commands with the DIM DNS. The DIM DNS assigns client subscriptions to the server. The WinCC OA client will then call a command or subscribe to a service provided by the server.

Command and service functions trigger their assigned callback functions, defined in the server. Commands are used to execute an action, and services are used to receive values from the server. From the callback functions, the server submits a task to an asynchronous loop. The loop runs an operation that reads or writes to the device. The values of the read operation are propagated back to the WinCC OA client. The commands are always submitted on the client side, and the services are updated on the server side.

The asynchronous loop is implemented using the asyncio package. When initialised, the server thread is launched in an asynchronous loop. Functions are defined as asynchronous functions. The main loop can create tasks to run multiple of these functions asynchronously. The asynchronous approach is chosen as an in-between solution of running the code sequentially and avoiding complications of parallel threading, as described in Section 5.1.2. With this approach, the server allows event-driven execution of functionalities by scheduling tasks to the asynchronous loop. It does not become unresponsive when executing a command, and its behaviour is well-defined at all times.

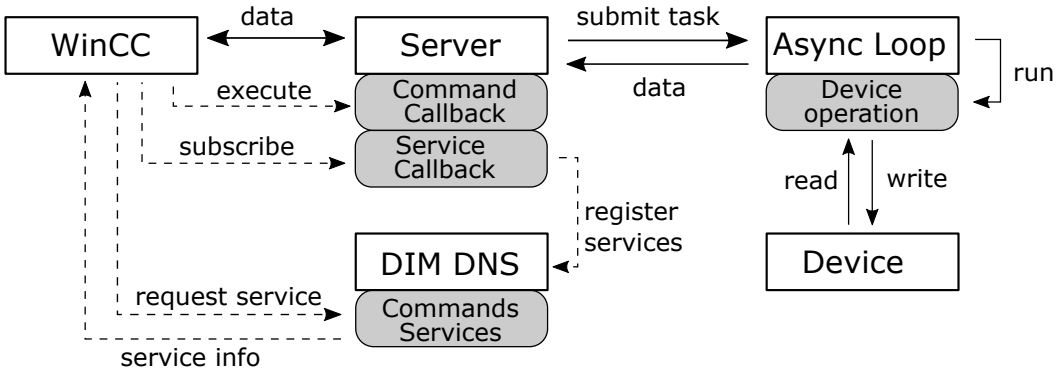


Figure 5.9: Schematic of the data flow for reading and writing data to the device.

The main feature for flow control in the code are `Future` objects provided by the asyncio package. These objects monitor the state of an asynchronous function. When an asynchronous function is executed, its `Future` object is awaited. The executing function will continue once the state of the `Future` object is `Done`. These `Future` objects can be awaited anywhere in the code, making them especially useful when performing `write()` and `read()` operations on the device.

When the server starts, and the device is initialised, a read and write operation is performed to check whether the device is operational. Only then is the server put into the `OK` state. If communication with the device cannot be established, the server is put into the `Dummy` state. This allows starting the server independent of the device state. When the device is initialised but communication with it breaks, the server is put into the `ERROR` state. The main loop periodically triggers the `update_server()` function of the server, which periodically reads the information of the device. If the server is in `ERROR` state, instead of reading all the values, a `probe()` command is used to try to reinitialise the device. If successful, the server is returned to the `OK` state. The update function also serves as the heartbeat of the server, sending a periodic `Alive` signal to the WinCC OA client. A reasonable heartbeat update interval is around 1 s. Should a connection to the device be established after server initialisation, the `Dummy` state is unset. The server provides logging capabilities using the logging package [75]. Device-specific configurations are read in via a configuration file using the configparser [76] and argparse [77] packages.

Figure 5.10 visualises the basic structure of the server. At the top of the server class, decorators are defined. Decorators are wrappers applied to defined server functions to differentiate the functionality they provide. Basic functions without a decorator mainly handle internal server functionalities.

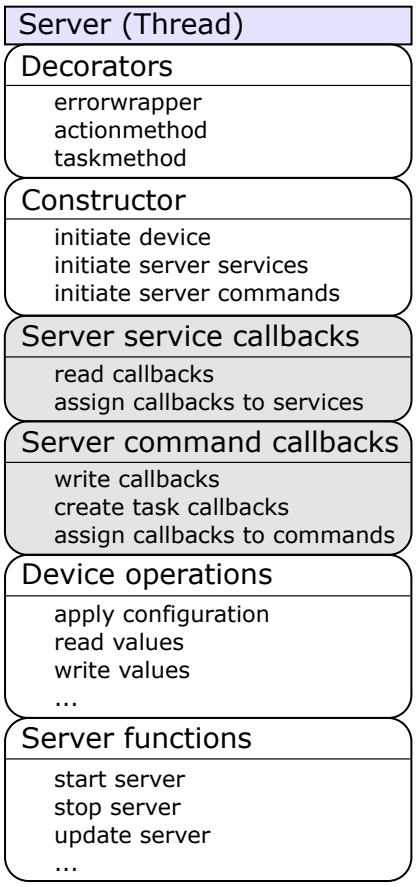


Figure 5.10: Server structure.

The `actionmethods` are functions that perform a `write()` or `read()` operation on the device. Only one `actionmethod` can be executed at a time by claiming the device, blocking access to any other function call during the read or write operation. This ensures that no read and write operations are performed simultaneously. The `taskmethods` are used to submit new tasks to the asynchronous loop. The `taskmethods` are the callback command functions provided to be executed by the WinCC OA client. The `errorwrapper` decorator is introduced to unify the error handling of the functions.

In the class constructor, the device is initialised. All the commands and services of the server are also initialised in the constructor. Service and command callbacks are defined, providing the necessary functionality for communicating with WinCC OA. The services for the ECS systems vary and are implemented individually. Service callbacks are used to periodically read actual values from the device. Command callbacks are used to set values or execute operations on the device. The device operations include the `read()` and `write()` functions. Additionally, server control functions are implemented. They include, among others, launching the main asynchronous loop and starting, stopping, and updating the server.

6 The Beam Conditions Monitor control system

The Beam Conditions Monitor (BCM), described in Section 4.2, was upgraded with new hardware. Therefore, the Experiment Control System (ECS) needs to be updated to include the new Machine Interface Beam Abort Decision (MIBAD) system. The full BCM ECS is implemented. It consists of several components, as depicted in Figure 6.1.

The Beam Abort Interface (BAI) component is responsible for monitoring the BCM health and setting or withdrawing the `BCM OK` signal to the Vertex Locator (VELO). It also sets the thresholds based on the Large Hadron Collider (LHC) machine mode. Further, it is used to rearm the BCM in case of a beam abort and sets the injection permits at handshake with the LHC. It is described in Section 6.1. Another part of the ECS is the monitoring of the CAEN power supply. This is done by the CAEN component. It was adapted from the previous version of the BCM ECS. It is briefly introduced in Section 6.2. For feedback on the abort reason, post-mortem data readout is implemented for the MIBAD. It handles writing the post-mortem data to disk for later post-mortem analysis after a post-mortem Trigger (PMT) is received by the BCM. It is presented in Section 6.3. The ECS components are combined into a main BCM ECS, and integrated into the global ECS of the Large Hadron Collider beauty (LHCb) experiment. This is outlined in Section 6.4.

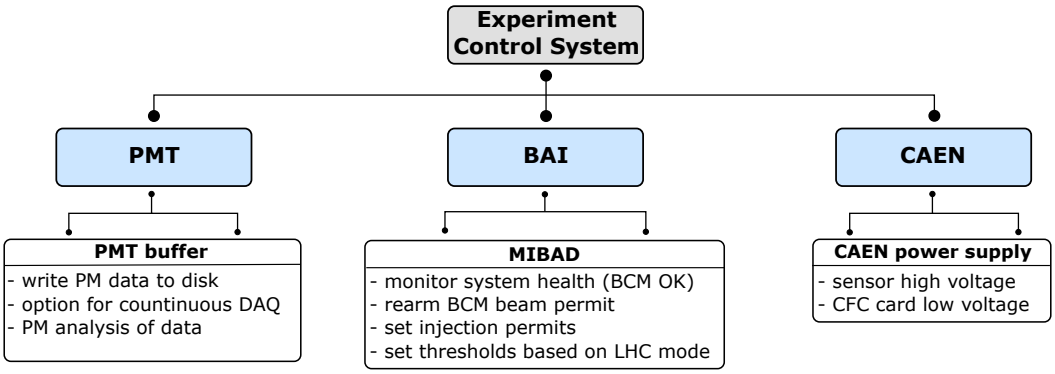


Figure 6.1: Schematic overview of the components of the BCM ECS, including the BAI, the PMT, and the CAEN components.

6.1 The Beam Abort Interface component

The MIBAD system requires the development of a new control system to accommodate the updated hardware. This section describes the BAI component of the BCM ECS implemented for this purpose.

First, communication must be established with WinCC Open Architecture (WinCC OA). The communication between the MIBAD board and the ECS is established over the Distributed Information Management (DIM) communication service. The server described in Section 5.3 is adapted to the requirements of the MIBAD system. The most noteworthy change is the read-out of the MIBAD data.

Second, the monitoring and control tasks implemented in the ECS are described. This includes continuous health checks by the software and actions to set or unset the permits of the MIBAD. Finally, the finite-state machine (FSM) of the system is presented, and the operator panels are explained.

6.1.1 Register definition

The MIBAD stores its data in registers. The registers are defined in a central `xml` file. The file is used by the DIM server and the firmware to assign hexadecimal addresses to the registers. It is also used to generate the data points (DPs) of the WinCC OA project. This is done to ensure compatibility between the firmware registers and the DP values stored in WinCC OA. Each entry in the file is given a type. They are `typedef`, `struct`, `register`, `field` and `subfield`.

The `typedef` types are used to define DP types. DP types in WinCC OA are acting as templates for the actual DPs (refer to Section 5.2.1). This means that by defining a set of DP types, the creation of the final DPs becomes much simpler, as all the structural information is defined only once. Structures are then included by reference rather than defining everything manually multiple times.

The `struct` types are used to structure and define the register data hierarchy, but they have no real representation in the firmware. A `struct` can hold any number of other types. The actual registers, as implemented in the firmware, are defined as `register` types. Each `register` is given a unique `name` attribute for identification, a `size` attribute to determine the total bit length of the data in the register in multiples of 32 bits, and an `access` attribute used to declare whether WinCC OA has read and write permission on the register.

Every `register` can contain several `field` types. A `field` defines what data is stored in the registers. There are different flavours: `standard field`, `bit field` and `custom field`. Each `field` is given a `name` attribute for identification, a `len` attribute defining the bit length, a `num` attribute defining how often the field is duplicated, and an `access` read-write permission overwrite attribute.

The `standard fields` are `bool`, `uint`, `ulong`, `float`, and `string`, corresponding to the equivalent types in common programming languages. In principle, other data types, such as arrays, are supported as well, but are not relevant for the definition of the MIBAD registers. The `bit field` type is a bit special, as it defines a field of 32 or 64 bits. It contains further `subfield` types that can be any number of `standard fields` defining the contents of the `bit field`. The `bit fields` are especially useful, since the MIBAD registers are defined in multiples of 32 bits. The `custom field` type is used to include a `register` or a `struct` by reference. This is done by setting its `name` attribute to the `name` of a previously defined `register` or `struct`.

A script ensures the entries in the `xml` file conform to the defined syntax so that the registers can be implemented in the firmware as defined in the file. This includes checks for the correct format of types and attributes, and for the consistent length of elements within `registers` and `bit fields`. Additionally, the `access` attribute of the `registers` is used to verify correct read and write access.

6.1.2 Synchronisation of registers with the control system

Read and write access to the registers is implemented as follows: registers are read in a single request, while data is written in a point-to-point format. In the point-to-point format, a complete multiple of 32 bit addresses is reserved for a single value, even if it is a 1 bit boolean. As data is written to the registers in multiples of 32 bits, this ensures that no other unrelated data is affected by the write operation. The compromise here is between secure writing and efficient data reading. When combining multiple read-only values into 32 bits (e.g., 32 boolean values) and reading all values in a single go, the data can be sent in a more compact format. Data decoding is performed by the DIM server. A schematic overview of this concept is given in Figure 6.2.

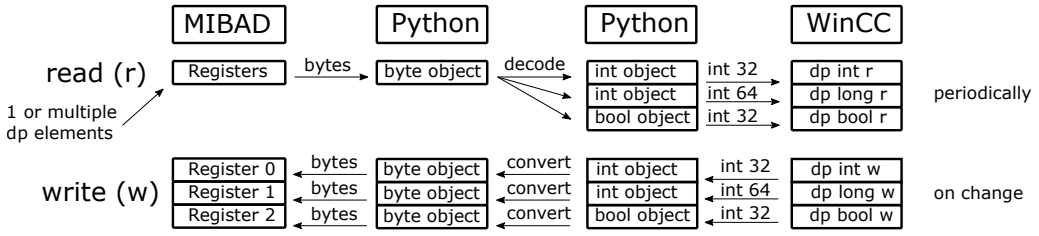


Figure 6.2: Schematic of the data read-out flow between the MIBAD and WinCC OA, exemplary for `uint`, `ulong` and `bool`.

A few of the register addresses are fixed, so they can always be read out correctly by the DIM server. The address `0x0` is used to reset the board. Address `0x1` is an unused register used for confirming read and write operations. The server will write a random number upon initialisation and read back the register value. If the operation is successful and the read data matches the random number written, the MIBAD device is initialised. The following two registers, `0x3` and `0x4`, provide a timestamp. The timestamp must increase with each read of the register values; otherwise, the server will mark the MIBAD as erroneous. Address `0x5` contains the firmware git hash. Using the git hash, the server can find the correct register file to initialise the remaining addresses, corresponding to the firmware version of the MIBAD. This ensures that the DIM server reads the correct data from any given address.

To perform the read and write operations to the MIBAD registers, a proxy server is implemented [13]. A MIBAD Python class accesses the read and write commands provided by the proxy server. The MIBAD class is included in the DIM server and also provides the capability to read and write registers from the command line. The DIM server reads out the MIBAD values every 1.3 s and publishes them via DIM to WinCC OA. The 1.3 s readout period is a standard for beam-monitoring applications across the LHC. It defines the update rate of the WinCC OA DPs.

In WinCC OA, the DPs are created based on the `xml` file description. Changes in the register addresses are irrelevant, as long as the register structure is not changed, as the assignment over DIM is dereferenced from the address. If, however, the structure changes, the DPs needs to be recreated to represent the changes. It is important to note that the history of deleted DPs is lost, so extra care must be taken when making firmware changes. Since the firmware is not expected to have frequent significant changes, this is acceptable.

All main firmware components of the MIBAD, described in Section 4.2.4, are mirrored to the ECS as DPs. Additional DPs are created for the writeable registers, as indicated by the `access` attribute of the entries in the `xml` file. This separation is necessary, as otherwise the writeable registers could not be monitored. The data of the charge-to-frequency converter (CFC) stations is streamed to the ECS only once per station. The redundancy is removed at this stage, because the ECS is not critical to safety. The MIBAD writes the raw values of the CFC cards in dedicated registers. The correct assignment of the links to their respective station is done over the card identification number (CID) given in the CFC data (refer to Section 4.2.1).

Due to the update interval of 1.3 s, the currents transmitted to the ECS do not contain the information for every 40 µs. Instead, the maximum, minimum and a sample value are given within the update interval. The values are given for all calculated running sum (RS) of the three implemented abort criteria (refer to Section 4.2.3). The currents are given in ADC ticks. They are transformed into nA with Equation 4.1 from Section 4.2.1.

The thresholds are set depending on the machine mode that is constantly monitored. For this purpose, the BAI component is interfaced with the LHC ECS. All LHC modes related to injection set the BCM into injection mode; all other states are treated as BCM physics mode. When the machine mode changes, the BCM mode is changed accordingly and is written to the MIBAD. In the MIBAD, the thresholds are updated accordingly.

Relative currents are calculated based on the thresholds for each RS of the abort criteria. The values are synchronised into dedicated DPs by watching the original MIBAD register values and performing the calculation whenever they are updated. The relative currents make it easy to see how close they are to the abort thresholds.

6.1.3 Monitoring of the system health

The health of the BCM is constantly monitored. The health checks are executed when a monitored DP changes (event-based code execution). The purpose is to withdraw the `BCM OK` signal to the VELO if any of the monitored values indicate system misbehaviour or failure. Most importantly, the data acquisition of the MIBAD is monitored. For this, the following checks are performed.

First, the connection to the DIM server is monitored. If the DIM server itself is unresponsive, no information can be given on the MIBAD. Therefore, the ECS will go into an `UNKNOWN` state. If the server sends a heartbeat but the MIBAD still cannot be reached, the DIM server sends an error to the ECS. In this case, the ECS indicates a connection error between the DIM server and the MIBAD.

After communication is established and the MIBAD registers are synchronised with the DPs, the data in the MIBAD registers is checked for errors. There are three main checks performed that are also implemented in the hardware. The redundancy of software and hardware checks ensures error detection even if the firmware fails.

The health of the redundant data links is checked by the MIBAD router. The links receiving data are assigned a station via the CID provided by the CFC card. If a link does not receive data, it is marked as unassigned. The expected station CID is read from the firmware. If one of the stations has no operational link assigned, a router error is thrown. If some links are not receiving data or show transmission errors, a warning is issued. After the health of the links is checked, their correct assignment to the stations is verified using the CID. Each station must have one correctly assigned CID. If this is not the case, a CID error is thrown. Warnings are issued if not all stations have both redundant links assigned. Finally, the CFC health is checked. The status bits of the CFC data are checked. The critical error bits are throwing a CFC error; other error bits will issue a warning. The digital-to-analogue converter (DAC) values are monitored separately. If one of the DACs has a value over 155, a warning is issued; at the limit of 255, an error is thrown.

Another important aspect is monitoring the CAEN high- and low-voltage channels. The supplied high voltages for the BCM sensors and the low voltages for the CFC cards are monitored, along with their respective currents. If any of the monitored values fall outside the allowed range, the channel is erroneous. If the voltage supplied to the BCM sensors is incorrect, the affected sensors are no longer operational. One CAEN channel provides voltage to all sensors in a station, so if one sensor causes the high-voltage supply to fail, the entire station becomes non-operational. Changing the voltage supply to provide the high voltage per sensor can be considered for the future to mitigate this problem.

For any error thrown, the ECS will go into `ERROR` state and unset the `BCM OK` signal of the MIBAD. In addition, an error is raised to the VELO at the software level. It can be considered a soft removal of the `BCM OK` signal. In both cases, the VELO is moved into a safe position, and the voltage is ramped down. Again, the redundancy between software and hardware adds an additional safety level to the system.

6.1.4 State machine

The BAI component implements a FSM (refer to Section 5.2.2) for the MIBAD device to monitor the system state. The `UNKNOWN` and `ERROR` states are set as described in the previous section by monitoring the system health. Additionally, several states can occur during operations. The states are based on the permits read from the cavern interface register of the MIBAD. A flow chart of the MIBAD FSM is given in Figure 6.3.

The default state of the FSM is the `UNKNOWN` state at initialisation. The FSM needs to be recovered to bring it into a good state. To recover, the FSM transitions through a `RECOVERING` state. The recover function performs all system health checks and sets the cavern permits, as well as the `BCM OK` signal accordingly. The FSM state is then updated based on the state of the cavern interface. This ensures that the system is always in a state corresponding to the actual hardware state. The only exception is if setting the permits of the cavern interface fails. In this case, the FSM will retry the operation a few times and then manually enter the `ERROR` state, until communication with the MIBAD is fully restored.

After successful recovery, the default good state of the MIBAD is the `ARMED` state. In this state, the MIBAD is fully operational, and no errors are detected. The `BCM OK` and the beam permit are set. The second good state is during the injection of the beam into the LHC. To distinguish this important state from the `ARMED` state, the MIBAD FSM transitions into the `INJECTION` state. This is done over the global ECS of the LHCb experiment. It requests setting the injection permits for the BCM for both beams by calling the corresponding MIBAD FSM functions, which, in turn, set the injection permits in the cavern interface.

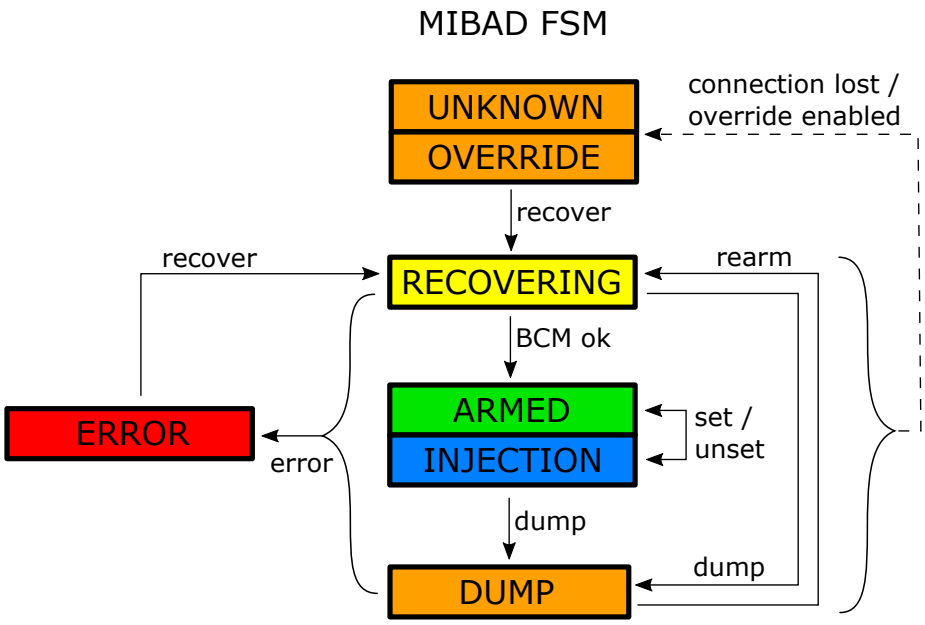


Figure 6.3: FSM of the MIBAD device. State transitions are allowed as indicated by the arrows.

The FSM functions can be called manually by the operator by clicking the respective button on the operator panel, by managers running in the background, or by other interfaced systems. This allows the implementation of automatic error-recovery functions, as the cavern interface and system health are continuously monitored. If the server monitoring function detects a connection to the DIM server, while in `UNKNOWN` state, the FSM recover function is called by the script. The same is done by the health monitoring functions. If the health check indicates that the BCM is healthy while in `ERROR` state, the recover function of the FSM is called. It attempts to restore the BCM to a healthy state.

When the beam permit is withdrawn by the BCM, a beam abort will be triggered. In the FSM, this is indicated by going into the `DUMP` state. After a beam abort, the system must be rearmed manually by the operator. This is done by calling the corresponding rearm function implemented in the BAI FSM. The rearm function transitions to the `RECOVERING` state, as does the recover function. In addition, it attempts to reset the beam permit. The FSM transitions to the correct state based on the success of the health checks and the permits indicated in the cavern interface.

In addition, an `OVERRIDE` state is implemented. In this state, all permits of the cavern interface of the MIBAD are ignored and overridden by the ECS. It is helpful for tests, such as verifying the permits individually with the LHC. However, as this state overrides the hardware permits, it effectively disables the BCM completely.

It has a severe impact on the operation of the BCM and cannot be easily set. To enable it, a multi-step expert intervention is required. First, the MIBAD has to be put into maintenance mode. This is done by issuing the correct command via the command line interface. Only afterwards can the ECS enable the override by setting the corresponding register entry.

The states have an assigned priority, defining a hierarchy to prevent ambiguity. The `ARMED` and `INJECTION` states have the same priority, dependent on whether the injection permits are set. The `DUMP` state has higher priority and is set independently of injection or normal operations. The `ERROR` state has even higher priority. The `ERROR` state has, in principle, the highest priority, because functionality cannot be ensured when the system is in error. However, the `UNKNOWN` state takes precedence over the `ERROR` state to indicate a problem with the DIM server. The `OVERRIDE` state is a special state that overrides all other states.

The states are colour-coded from green (blue) for low priority to increasingly red for high priority. The good state `ARMED` is thus green. The `INJECTION` state is blue to differentiate it from the default `ARMED` state due to its special importance during the injection of the beam into the LHC. The critical `ERROR` state is red for highest attention. The transitional `RECOVERING` state is yellow, as it is neither good nor bad. The `UNKNOWN`, `DUMP` and `OVERRIDE` states are orange for high attention.

6.1.5 Operator panels

Operator panels are implemented to provide a user interface (UI) to the user. The most important panels are explained. Additional panels are added to Appendix A. The panels are visually heavily inspired by the previous implementation. However, they are recreated to work with the new DP register structure of the MIBAD. The main overview panel shows the most important information to the operator. A screenshot is given in Figure 6.4.

Depicted in the centre of the panel are the currents of both BCM stations over time. To avoid cluttering the graph with 16 trends for all sensors for each RS, only four are depicted, representative of the others. The RS32-SUM is used because it accounts for the entire station. Using the same methodology, an RS2-SUM is calculated for both stations. This RS2-SUM is not used in any abort criteria but is a good representative of the RS1 and RS2 values, while requiring only one trend per station. All currents are given in per mill of the threshold. This allows for easy identification of significant current spikes.

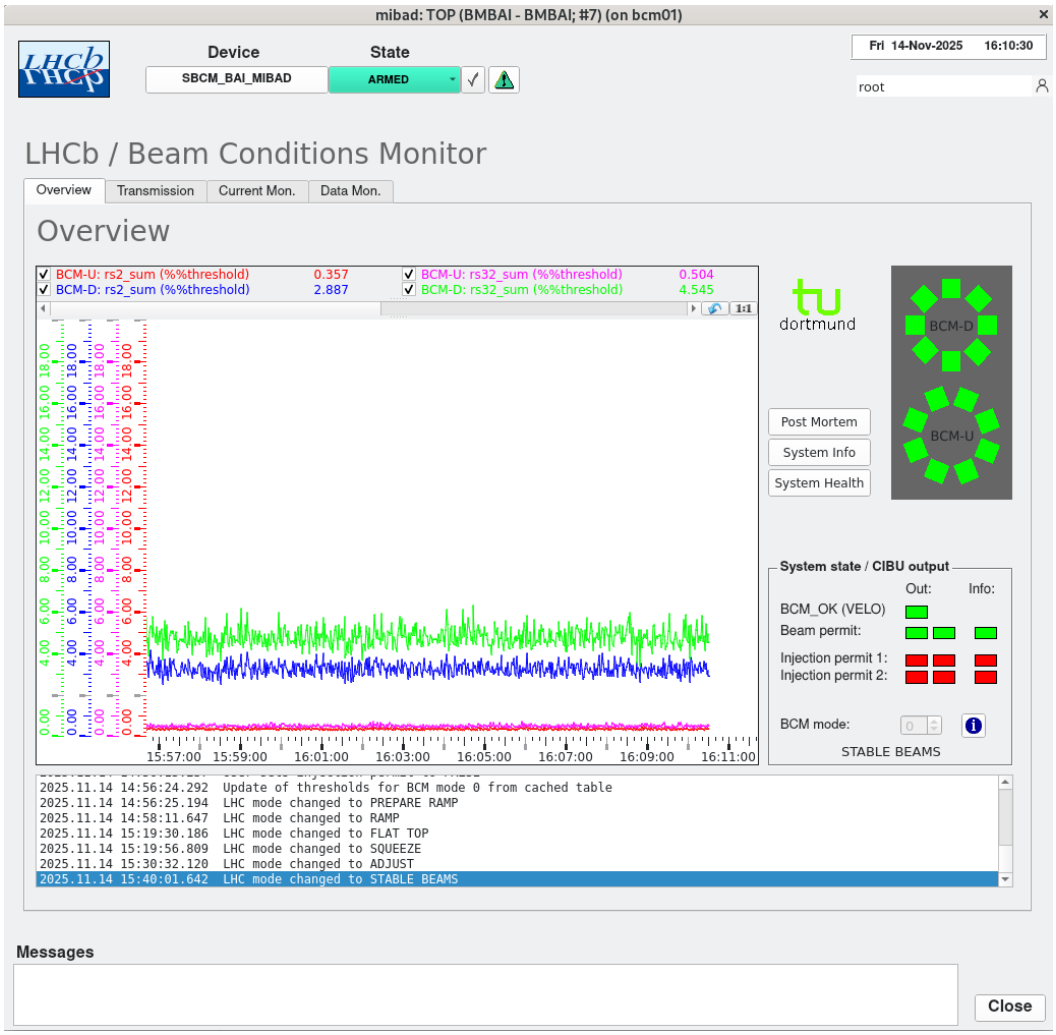


Figure 6.4: Screenshot of the main MIBAD overview panel.

Additionally, on the right-hand side of the trend, the diamond currents are colour-coded relative to the thresholds for each sensor of both stations. The colour ranges from green (far below threshold) to red at the threshold level. Below the diamonds, the cavern interface is represented. It shows whether the `BCM OK` signal is sent to the VELO, and if the beam and injection permits are set. It displays redundancy where applicable and provides feedback on the global information provided by the beam interlock system user interface (CIBU) unit. The BCM mode regarding physics (0) or injection (3) and the LHC mode are given as well.

At the top of the panel, the FSM state of the device is shown. At the bottom, the most relevant operational log messages are given. They range from a change of beam mode to critical error messages. The abort reason is provided for a beam abort. The abort reason provides information on the abort criteria and the sensor currents that triggered the dump. The sensor currents are continuously monitored. A counter is incremented every time the currents go above 20 % of the threshold. A message is printed displaying the number of overshoots and the maximum current within the last hour.

Figure 6.5: Screenshot of the MIBAD transmission panel. Only one link is active for the BCM-D station.

In addition to the general overview panel, three more tabs provide more detailed information on the MIBAD. The transmission panel is given in Figure 6.5. This panel provides detailed information on the CFC and router health. The top-left part of the panel displays the CFC error and integration-level bits for each sensor, together with the DAC values of the channels. The integration level bit indicates whether the current injected by the DAC is high enough to meet the 10 pA design requirement. If the integration level is subsequently too low, the CFC error will be set. The DAC will increase the injected current until both flags are in a good state or the maximum DAC value of 255 is reached. At this point, the CFC card needs to be recalibrated, and a DAC error is thrown. To indicate the DAC health, it is displayed as a relative value in a status bar.

To the right, the CFC status bits are displayed. They are to be referred to in case the monitoring function raises a CFC error or warning to identify the source of the error. The `status HV` bit is currently not used by the BCM and is thus greyed out. The CFC card sends additional status bits that indicate the results of internal card tests. These tests are currently not used by the BCM and are not relevant for operation. They are included in the panel for completeness.

The health of the links is monitored at the bottom of the panel. This part of the panel is to be referred to when a router error or warning is raised. For the four links, the station is assigned by matching the CID sent by the CFC card to the expected CID set in the firmware. A frame identification number (FID) is incremented to indicate that the link is active and no packages are lost. If errors occur during data transmission, the respective field is incremented by one. They can be a failing CRC check, a wrong length of the received data package, an error with the transceiver (XCVR), or a sequence error if the FID is indicating package loss. At the start-up of the CFC cards, one sequence error is expected, as the FID resets.

The last two tabs provide a deeper insight into the BCM currents. In the current monitor, the current trends are displayed for all RS for both stations and for all sensors. These trends should be consulted when the information in the overview panel on the currents is insufficient. The fluxes here are not only the maximum value within the readout period of 1.3 s, but also the minimum and a sample value, as well as the average flux within the period. This allows for a deeper understanding of the current fluxes for each sensor. The data monitor tab also provides the current values of each RS for each sensor, but not in a time trend, but in a periodically updated table. The table shows the maximum current value and can be displayed in ADC ticks or nA. The ADC ticks can be interesting because they are the units used for calculations in the firmware.

6.2 The CAEN component

The CAEN power supply of the BCM is monitored by a separate CAEN project. The power supply was not exchanged during the upgrade; however, the opportunity was taken to introduce a state machine to monitor the CAEN channel operation. Communication to the CAEN is established over the OPC Unified Architecture (OPC UA) protocol. The CAEN project is implemented solely to monitor the status of the high- and low-voltage channels. Interventions to change the voltages are not permitted by the ECS. As the safety system, the BCM is always operational. Therefore, the voltages are always on and supplied independently of other systems. If interventions are necessary on the CAEN, an expert command line interface is provided by the CAEN crate.

As a standard component, the CAEN is included in the Joint Controls Project (JCOP) framework, and the provided functions and panels are adapted to the needs of the BCM. A selection of the panels is added to Appendix A. For each active channel, the voltage and currents are monitored to be in a healthy range. A graph provides information about the stability of the values over time. An expert panel for each channel provides further insight into the health of the channels, such as whether the power tripped or whether one of the limits was triggered.

The FSM for the CAEN is basic. If the channel is on and healthy, the default good state is `READY`. When the channel is unhealthy, it enters the `ERROR` state. In addition, the channel can be `OFF`. For the BCM, the `OFF` state is equivalent to an `ERROR` state. The parent FSM is a simple superposition of all the channels. When all channels are `READY`, the parent state is `READY` as well. In every other case, it is in `ERROR` state. The colour code for the states is green for the good `READY` state, grey for the `OFF` state, and red for the `ERROR` state.

6.3 The post-mortem Trigger component

The BCM is interfaced with the post-mortem system of the LHC. After a beam dump, the MIBAD receives a PMT. This section describes the PMT system implemented to handle the event. A circular PMT buffer is written. It stores the most recent data in transient storage and saves it to disk at a PMT for post-mortem analysis. A FSM is defined to integrate the buffer into the ECS. At the end of the section, the procedure for analysing the post-mortem data is outlined, taking an example dump triggered by the BCM on June 24th 2023.

6.3.1 Data format

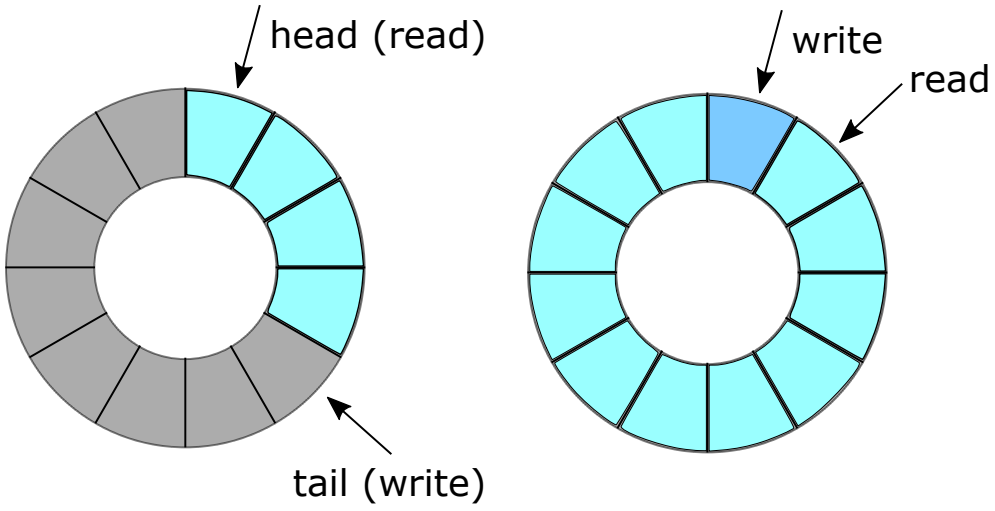
The MIBAD sends out the data in BCM packages. The following BCM data format is defined. Each BCM package contains a header and a body. The header consists of an 8 byte timestamp and 1 byte indicating the data package type. The format and length of the package depend on the type. There are seven different types defined.

The `permit_change` package is sent whenever one of the permits in the cavern interface is changed, including the `BCM OK` signal, the beam and injection permits, and the PMT. Other firmware flags, such as whether an abort criterion was triggered, are also included. This package is used to identify a beam abort or PMT. It has a length of 16 byte plus header. The `cfc_raw` packages contain the unprocessed raw data of the CFC cards as given in Section 4.2.1. This data can be used to reconstruct the BCM currents from the ADC values. The package has a length of 52 byte plus header. Additionally, `data` packages are defined. They contain the processed RS calculated for the abort criteria, namely the RS1, RS2, RS32 and RS32-SUM. They are sent for all sensors of the station. The package has a size of 112 byte plus header.

The PMT buffer also supports the data format used throughout the LHCb experiment, named the DAQ40 format [78]. This data format was created specifically for use with the PCIE40 cards used at the LHCb subdetectors. The BCM packages are stored in 1056 byte data fragments. Each fragment has a 4 byte header that includes the fragment size and a bunch identification number (BxID). Data padding is used to keep the data 8 byte aligned. Fragments are bundled into blocks with an additional 24 byte header that provides the number of fragments in the block, the total size, and further information about the data. The full stream is given yet another 84 byte header. This header is sent only once at the beginning of the data stream. It includes a timestamp, information about the sender and receiver, and details on the type of transmission and the data. This data format has been implemented for a future upgrade of the BCM utilising the PCIE40 readout. This allows for quick buffer adaptation when upgrading the readout to that format.

6.3.2 Circular buffer

The BCM data is streamed continuously at 40 µs intervals from the MIBAD to the PMT buffer. The challenge for the PMT buffer is to store the relevant information for the post-mortem analysis in a memory-efficient manner while maintaining good performance. This is achieved with a circular buffer implemented in `C++`, enabling highly efficient programming through pointer arithmetic and byte operations for both speed and memory efficiency. A circular buffer prevents extensive memory usage by keeping only the most recent data.



(a) partially filled (b) filled

Figure 6.6: Concept of the circular buffer. The write operation wraps around and overwrites the oldest data. The head pointer is moved to the next position, now containing the oldest data in the buffer.

The concept of a circular buffer is illustrated in Figure 6.6. A circular buffer adds data to a memory space of a defined size (the buffer size) and assigns head and tail pointers. Data is written to the tail pointer and read from the head pointer. The memory space can be imagined as a circle, so the tail pointer will wrap around to the beginning of the allocated memory when it reaches the end. The oldest data will be overwritten as the circular buffer fills, leaving only the most recent and relevant data. The buffer size of the BCM has been set to include the last 200 s of data before the dump, corresponding to 582 Mbyte of data sent to disk after a beam dump.

The circular PMT buffer is filled with the received BCM packages. When receiving a package, the buffer checks if a PMT was triggered. If so, the buffer starts a dump thread to write the data to disk. In order not to block the data stream of the MIBAD, additional memory is allocated. The buffer writes incoming data to this secondary memory, so that the read operations from saving the data to disk do not interfere with the write operations from the incoming data. After a BCM package is written to disk, its memory is freed. Once all BCM packages are written, the primary memory is thus cleared, and the secondary memory becomes the new main memory. At the next dump, the roles of the two memory spaces are reversed.

At initialisation, the memory can be allocated in two different ways. On the one hand, the primary memory can be allocated statically at initialisation. The secondary memory is allocated dynamically as needed. This way, the memory overhead is marginal, as long as the new BCM packages are not received much faster than the old packages are written out to disk. The maximal memory overhead is twice the buffer size.

On the other hand, both primary and secondary memory can be allocated statically, resulting in a total buffer size twice that of the original buffer. This is reasonable, if enough memory is available, so no memory needs to be allocated dynamically. Dynamic memory allocation can be dangerous as it can lead to out-of-memory system failures. The second memory allocation method is used for the PMT buffer during operation, as the server running the buffer provides sufficient memory.

In addition to writing the buffer to disk after receiving a PMT signal, the PMT buffer also provides continuous streaming of data to disk. The BCM data packages sent by the MIBAD are processed in the same way, but are directly written to disk without waiting for the PMT. The continuous data stream does not interfere with writing the buffer to disk after receiving a PMT, as both actions are executed in parallel. The streamed files are not limited in size, and it is the responsibility of the operator to start and stop the data stream. The streaming mode is used when the BCM data is relevant for analysis in special runs performed at the LHCb detector.

6.3.3 State machine

The PMT buffer is derived from an implemented FSM class providing DIM capabilities. This way, the PMT buffer is interfaced with the PMT component of the ECS. The FSM of the PMT buffer is visualised in Figure 6.7. The default state is `NOT_READY`. At initialisation, the server is configured by internally calling a configure command. The configure command opens a socket to stream the BCM packages from the MIBAD by connecting to the corresponding host and port. When a socket connection is established, the FSM enters the `READY` state.

As soon as the MIBAD starts sending data packages, they are filled into the PMT buffer, processed, and checked for a PMT signal. The buffer goes into `RUNNING` state. When finding a PMT signal and the buffer is written to disk, the FSM goes into `PM_RECEIVED` state. The state is kept for at least 4 s or until the buffer is fully emptied, to ensure the state transition is visible to the operator. While in `PM_RECEIVED` state, the buffer continues writing new data packages to the secondary memory space, so operability is not blocked. When continuous data streaming starts, the PMT buffer enters the `STREAMING` state. Data streaming can only be started when the buffer is in the `RUNNING` state.

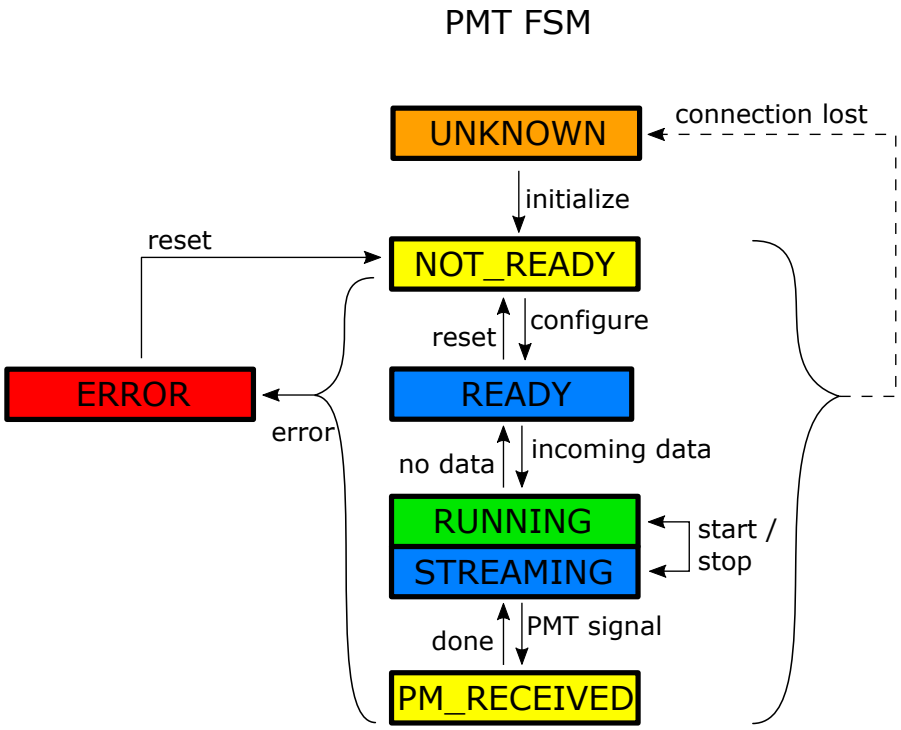


Figure 6.7: FSM of the PMT buffer. State transitions are allowed as indicated by the arrows.

In case of any errors, the PMT buffer enters the `ERROR` state. The server will automatically try to recover from the `ERROR` state by rerunning the configure command periodically. The status is provided to the ECS via DIM. Additional commands are provided by the server via DIM to start, stop, configure and reset the FSM. The continuous data stream is started and stopped by the provided `start-stream()` and `stop-stream()` commands. On the ECS side, an additional `UNKNOWN` state is defined, in case the connection to the buffer is lost and it stops updating its status.

The FSM states are colour-coded by priority as before. The `STREAMING` state is blue to separate it from the default good `RUNNING` state in green. The `READY` state is blue to indicate that it is a good state, ready to receive data, but it is not `RUNNING`. The `NOT_READY` state is yellow to indicate that the PMT buffer needs to be configured. The `PM_RECEIVED` state is also yellow to indicate the beam was aborted. The `ERROR` and `UNKNOWN` states are red and orange, respectively, as before.

6.3.4 Post-mortem analysis

For post-mortem analysis, the server will run an analysis script after writing the buffer to disk. The analysis script is not fully automated at the time of writing and requires manual intervention. The script is designed to create plots of the BCM currents directly before the beam abort. The plots are saved for the operator to analyse. The script creates two types of plots.

The first type of plot depicts the currents of the stations in a radial plot at the time of the dump. This plot helps identify the sensors above threshold that caused the dump. The radial plot is created for all the other abort criteria as well, even when they did not trigger the dump, so that they can be compared. An exemplary radial plot after a dump of the BCM-D station on June 24th 2023 is shown on the right-hand side of Figure 6.8. Three adjacent sensors at the station are above threshold; therefore, the dump was justified.

The second type of plot created is a plot of the BCM currents over the last 200 s. These plots are also created for all the abort criteria. These kinds of plots help determine how the fluxes evolved, providing more detailed insight into the behaviour leading up to the beam dump. An exemplary linear plot of the BCM currents for the same BCM-D station dump as before, zoomed in on the relevant time around the current increase, is shown on the left-hand side of Figure 6.8. The current increased significantly over a short time; thus, the beam was not misbehaving in general, but a single incident caused the dump.

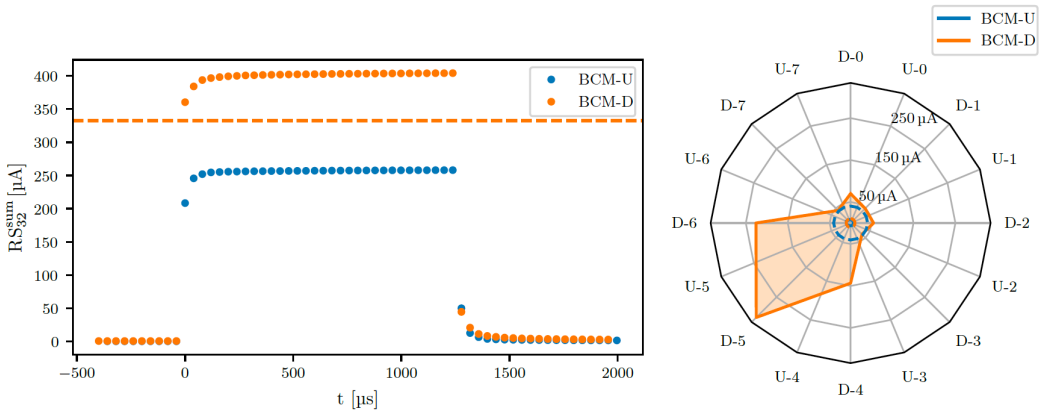


Figure 6.8: Plots generated for the PMT analysis for the beam dump on June 24th 2023. The time plot (left) shows the RS32-SUM currents around the time of increased currents. The radial plot (right) shows the angular distribution of the RS1 currents at the time of the dump.

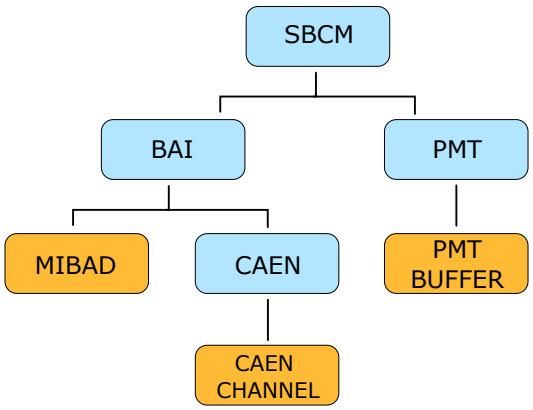


Figure 6.9: Schematic overview of the FSM tree structure. CUs are in blue, DUs in orange. State propagation is from the bottom to the top.

6.4 Combined project

The ECS components are combined into one single BCM ECS, including the FSMs and panels of the MIBAD, the CAEN power supply and the PMT buffer. This is done by adding a distribution manager to the projects. The BCM ECS is included in the global LHCb ECS. It transmits the BCM status to the control room and the shift crew. It indicates whether a beam abort was triggered by the system or whether a PMT was received so that the shifter can take the appropriate action. The BCM is not part of the detector readout for later physics analysis but is the safety system of the experiment. As such, it presents a special case and is treated accordingly when included in the LHCb ECS.

The main FSM tree structure is given in Figure 6.9. The BCM toplevel node is called SBCM. Beneath, it is distinguished between the BAI and the PMT unit of the BCM. The MIBAD and the CAEN are considered as a part of the BAI unit, due to their relevance for the beam abort. The PMT buffer is part of the PMT unit. Other hardware components can be added to the BAI and PMT units in the future if necessary.

The SBCM, BAI, and PMT units are Control Units (CUs), so their states depend on those of their children. The CUs all follow the same rules regarding error propagation. Errors are propagated in the FSM tree from the bottom to the top. A CU is in the `ERROR` (`UNKNOWN`) state if any of the children are in the `ERROR` (`UNKNOWN`) state. The `ERROR` state takes precedence, so the FSM can only be in the `UNKNOWN` state when none of the children have other errors. A CU is in their respective good state when all children are in a good state. The states are colour-coded by priority, analogous to the MIBAD and PMT buffer FSM states.

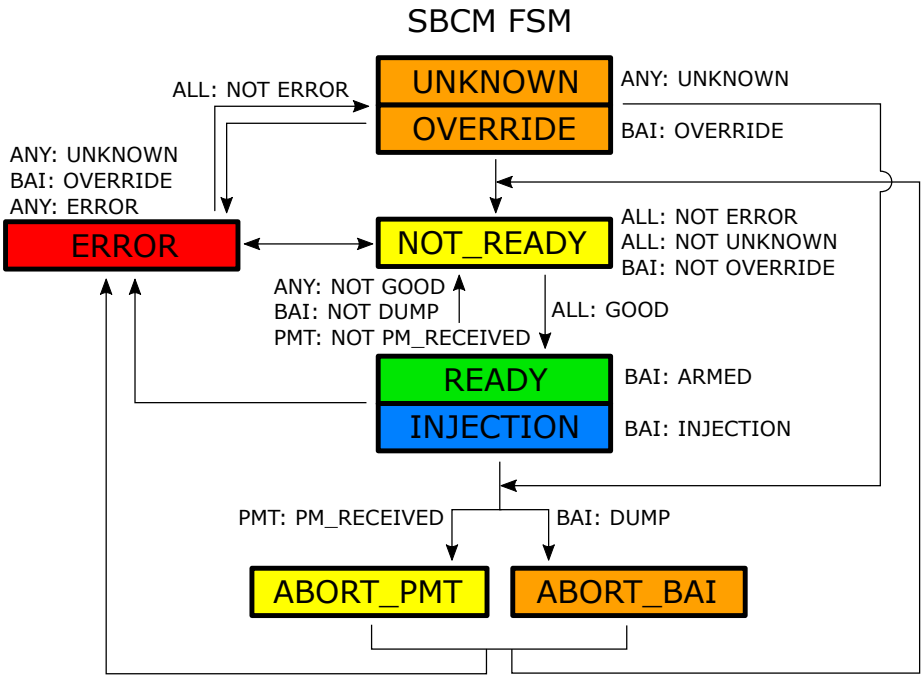


Figure 6.10: Schematic overview of the SBCM FSM. To transition between states, the requirements for the children must be met, and the transition must be allowed, as indicated by the arrows.

The BAI CU expands the FSM of the MIBAD with the FSM of the CAEN. The FSM states are the same as for the MIBAD FSM; the CAEN states contribute only to the `ERROR` state. This means as long as the CAEN FSM is not in `ERROR`, the MIBAD FSM states are propagated to the top. The PMT CU expands the PMT buffer FSM to work with multiple buffer instances. The states of the PMT buffer FSM are thus also propagated to the top.

In the last step, the SBCM CU combines the BAI CU and the PMT CU. The toplevel SBCM FSM is given in Figure 6.10. The default good state is the `READY` or `INJECTION` state. If the BAI and PMT CU are not in their respective good state, the SBCM FSM enters the `NOT_READY` state, unless any of the children are in an erroneous or beam abort state. At a beam abort, the SBCM FSM is set to the state `ABORT_BAI` or `ABORT_PMT`, depending on whether the `DUMP` state is propagated from the BAI CU or the `PM_RECEIVED` state is propagated from the PMT CU. The `ABORT_BAI` state takes precedence, since the dump was triggered by the MIBAD system; the `ABORT_PMT` state is triggered for all dumps. From the `ABORT_BAI` state, the BCM needs to be rearmed by the operator with the respective FSM function.

6.5 Conclusion for the control system

The control system of the BCM has been completely overhauled, along with hardware upgrades, to integrate the new MIBAD system. Read and write communication is established with the MIBAD. The register address space is defined and generated automatically to ensure version compatibility and coherence.

The BCM is continuously monitored in software for its health. This includes monitoring of the communication with the MIBAD, as well as the CFC card and router status. The CAEN power supply is also included in the BCM ECS. This allows monitoring the currents and voltages of the power supply.In the event of a detected system failure, the BCM is set to an error state. The checks are implemented redundantly in the hardware. This allows for soft interventions for the interfaced VELO system. A state machine is defined based on the status of the hardware cavern interface. Automatic error recovery is implemented when possible.

In addition, the BCM ECS is responsible for setting the BCM mode dependent on the LHC machine mode by being interfaced with the respective system. This is important to set the correct thresholds during injection and physics operations. Further, the system sets the injection permits for the MIBAD system. All the important data for the BCM is monitored and displayed on human-friendly operator panels. This includes detailed monitoring of the currents of the BCM sensors, as well as the transmission status of the CFC cards and the router. When the beam is dumped by the BCM, the system is rearmed by the ECS, by resetting the beam permit.

For post-mortem analysis of the data, a circular buffer server is implemented. It stores the most recent data and writes it to disk in case of a beam abort so that analysis can be performed on it. The buffer server is included in the BCM ECS. This allows for monitoring the operability of the post-mortem readout. For special runs, the BCM can be configured to continuously stream data to disk for analysis. This feature is implemented as a special case in the buffer server without impacting the post-mortem operability.

The BCM ECS is integrated into the global control system of the LHCb experiment. The MIBAD and ECS functionality has been confirmed over multiple beam dumps triggered by the system. Everything has been in continuous stable operation since the commissioning of the upgrade in June 2023.

7 The Timepix4 telescope control system

The Timepix4 telescope, described in Section 4.3, is used as a prototype for a future beam-monitoring detector. To already start development, the Experiment Control System (ECS) of the telescope is developed. The ECS remotely controls the power supplies and motion stages of the Timepix4 telescope and monitors its environmental conditions. The ECS is divided into multiple subcomponents taking care of the individual tasks, as outlined in Figure 7.1. Parts of the ECS have been implemented before, including control of the high voltage (HV) and environmental monitoring [19].

This work is expanded upon. The control of the HV is upgraded to a Power component, including control of the low voltage (LV). Its task is to set and monitor the voltages of the telescope. It is described in Section 7.1. The environmental monitoring is upgraded to an Environment component by creating a custom Environment box that simplifies monitoring of the temperature and humidity of the telescope. The upgrade is described in Section 7.2. Finally, a third completely new Motion component is added to control the translation and rotation of the motion stages of the telescope. They include the main translation stage of the telescope and the translation and rotation stages for the device under test (DUT). The Motion component is described in Section 7.3.

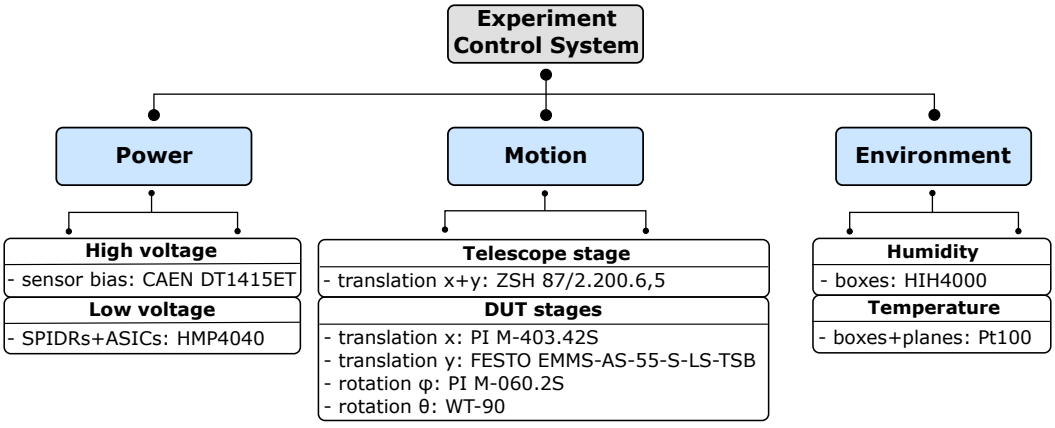


Figure 7.1: Schematic overview of the components and hardware of the Timepix4 ECS, including the Power component (HV and LV), the Motion component (main stage and DUT), and the Environment component (temperature and humidity).

(a) CAEN (b) Keithley (c) HMP4040

Figure 7.2: Pictures of the CAEN, Keithley and HMP4040 power supplies.

7.1 The Power component

The Power component is responsible for setting and monitoring the HV and LV of the telescope, including the Timepix4 sensors, the front-end electronics, and the DUT. For this purpose, different power supplies are integrated into the component. The functionalities include monitoring currents and voltages of the power supplies, handling alarms, and logging the values. Functions are provided to operate the power supplies, such as setting and ramping the voltages. In this section, hardware, driver, software, and user interface (UI) upgrades are described.

7.1.1 Hardware upgrade

The Power component of the ECS requires operating multiple power supply types. Three used power supplies are the Keithley 2410 [79], the CAEN DT1415ET [80] and the HMP4040 [81]. A picture of the three different power supplies is given in Figure 7.2. In the latest telescope setup, the CAEN provides the HV to the eight Timepix4 sensors. It has eight channels, which allows biasing each sensor individually. Keithleys are used to bias the DUTs at the centre and outer stages, depending on the setup. Up to five Keithleys can be operated simultaneously. Two HMP4040 power supplies are used to provide the LV to the Speedy Pixel Detector Readout 4 (SPIDR4) boards (split to two channels with 12 V each), and the eight telescope planes (four channels with 2 V each, corresponding to 2 planes per channel).

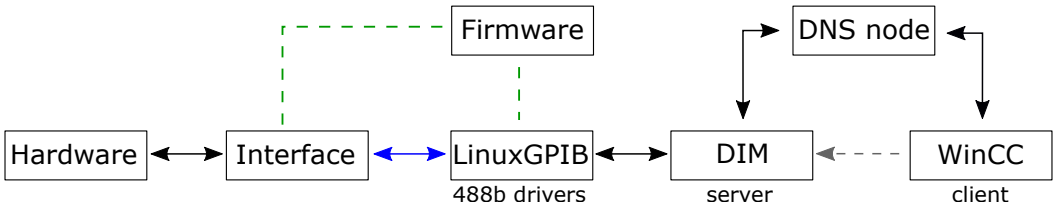


Figure 7.3: Visualisation of the drivers used to communicate between WinCC OA and the power supplies.

7.1.2 Drivers and software

The power supplies are connected to the ECS machine over their respective interfaces. The CAEN and HMP4040 can be communicated with over Universal Serial Bus (USB) or Ethernet. Communication with the Keithleys is established via the General Purpose Interface Bus (GPIB) interface. Standard Commands for Programmable Instruments (SCPI) are used for communication with the hardware. Each device is assigned a unique address for identification. After establishing communication, the voltages can be set remotely. Usually, this is done through a ramping action, with steps of fixed size and time, to give the current time to stabilise.

To establish communication with the hardware drivers are required, as visualised in Figure 7.3. For Linux, the used drivers for the 488b protocol are provided by the LinuxGPIB [82] project, an open-source GitHub [83] project. It communicates with the hardware over the interfaces and provides the necessary firmware. For communication with WinCC Open Architecture (WinCC OA), the Distributed Information Management (DIM) protocol, described in Section 5.3, is used. The DIM server accesses the LinuxGPIB drivers to retrieve the values from the hardware, and then propagates them to WinCC OA using the Domain Name System (DNS). The DIM server implementation specific to the Timepix4 telescope ECS is outlined in the next section.

7.1.3 Communication server upgrade

The server for the communication with the hardware is based on the server structure described in Section 5.3.3, adapted to the experimental requirements of the Power component. The adapted server class heavily relies on the PyVISA package [84]. It provides the commands that can be used from within Python to access the LinuxGPIB drivers via the pyVISA-py backend [85]. In Python, the device is defined as a resource, accessed over the unique address given to the power supply.

To support the different power supply types, the server is upgraded to include a `[Device]` section in the configuration file of the server. Different communication types can be specified to use the GPIB, USB, or Ethernet communication protocol. Because the device commands are not identical, a `[Commands]` section is added to the configuration file. The device communication commands are abstracted so the server can call the correct command for the corresponding device.

A channel class is added to the server to provide the functionalities required to address the multiple channels of the CAEN and HMP devices. The device variables, including the actual voltage and current, the source state, hardware limits, and many others, are defined in the channel class. The channels are initialised in the constructor of the server. Default channel configurations are provided in the configuration file. The `read()` and `write()` functions are adjusted to allow reading and writing values per channel. Tasks are defined per channel, queued individually. The main asynchronous loop iterates over the channels of each device, processing the tasks.

Error handling is added to the `write()` and `read()` functions to improve system stability. If a write or read command is interrupted, the operation is retried. The success of a write operation is checked by performing a read operation after the write. If the read operation does not return the same value as specified by the write operation, the write operation is retried.

The performance of the system is improved by updating the periodic readout of the data. The frequency of the readout is determined by the response time of the device: the longer the device takes to respond, the less often it is updated. This prevents a slow device from continuously blocking the readout of other devices. If a device exceeds a maximum response time, it is skipped and marked as not reachable. In this case, a `probe()` function is used to check connectivity. With certain devices, the probe operation can take a long time. Therefore, independent devices are started in separate threads so they do not interfere with each other.

7.1.4 Operator panel upgrade

The operator panels are upgraded to reflect the changes in the hardware. A selection of panels is added to Appendix B. Three main panels are created, one for each type of device, and updated to represent the per-channel settings for the CAEN and HMP4040 devices. The features are the same, including performing actions on the devices like starting and stopping the voltage source, ramping the voltage, monitoring the voltage and current values, setting the alarm limits, and regular automatic logging of the data. The general panel structure has been updated to a monitoring panel by default. Additional panels can be opened over their respective buttons.

The monitoring panel displays the current and voltage values more compactly than in the previous version. This allows the display of monitoring for all channels at the same time. The status of the channels and the alarm limits are displayed by coloured circles acting as lights. For colour-blind operators, the status values are additionally displayed as text: `-1` for the `invalid` state, `0` for the `normal` state, `1` for the `warning` state, and `2` for the `alarm` state. For displaying a binary state, the options are `ON` and `OFF`. Buttons for the `Log viewer`, the `DIM viewer`, the `Alert screen` and the `Expert settings` open their respective panels.

The part of the panel for applying and ramping the voltages is moved to a separate panel that can be extended via a button. This is done to separate monitoring of the values from taking action on the power supplies. This additionally allows for monitoring of the HV and LV values on one screen, as otherwise the panels would take up too much space. Actions are now performed not only across all devices, but also for each channel of the device. This leads to an updated panel layout, where channels are organised into tabs of four channels each. This allows all device channels to be displayed without the need for multiple panels. Actions performed on all channels are iterating over each channel internally.

Figure 7.4 displays a reference panel for setting and ramping the voltages for one of the Keithleys. The Keithley can be switched on and off, and the target voltage can be applied and ramped via the respective buttons. It is indicated whether a negative or positive bias is applied to the device. The hardware and software limits are displayed. Ramping is only allowed if all limits are enabled. Since, in the example, the software limits for the voltage are disabled, the buttons for ramping the voltage are also disabled. Two buttons at the bottom allow for refreshing or resetting the device, reapplying or reinitialising the settings and limits. The panels for the HV are colour-coded blue, and the panels for the LV are colour-coded yellow to differentiate between the two.

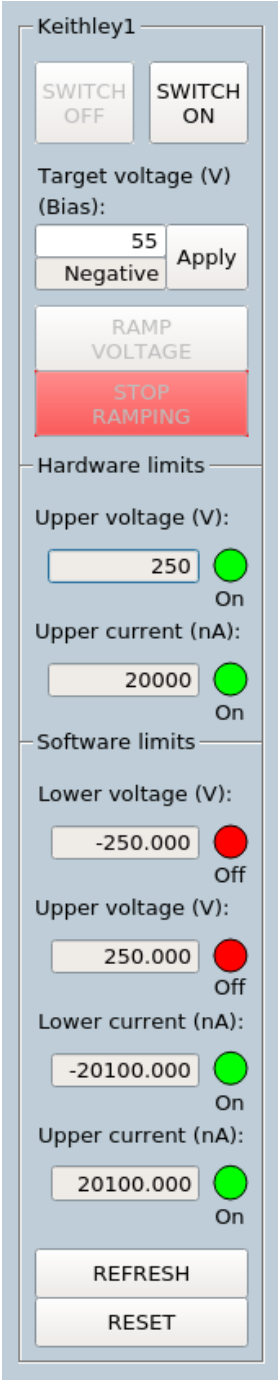


Figure 7.4: Reference panel of a Keithley.

7.2 The Environment component

The Environment component of the control system is continuously monitoring the environmental conditions of the Timepix4 telescope. The implemented functionalities include monitoring, alarm handling, and data logging of the sensor values. In this section, the component is briefly introduced. Then, the Environment box is described, which has been created to simplify the monitoring of the environmental conditions.

7.2.1 The component

The Environment component monitors temperature and humidity at different positions of the Timepix4 telescope. The following regions of the telescope are monitored: the temperature of the eight telescope planes, the temperature of the DUT, and the temperature and humidity of the dry-air-filled containers enclosing the two telescope arms and the DUT. The combination of monitoring temperature and humidity in each container allows for estimating the dew point. To monitor the temperature, four-wire Pt100 sensors are used; for the humidity, HIH-4000 sensors from Honeywell [86] are used.

The main hardware component, to which all the temperature and humidity sensors are connected, is an Embedded Local Monitoring Board (ELMB) [87, 88]. The used version is the ELMB128. The temperature and humidity sensors are connected to the analogue-to-digital converter (ADC) channels of the ELMB. The board is operated with two times 10 V to provide the voltages to the ADC channels and the Controller Area Network (CAN) bus. An additional 5 V is required to provide the voltage to the humidity sensors. A schematic overview of the ELMB setup is shown in Figure 7.5.

The ELMB is connected to the ECS machine over a sysWORXX interface, combined with a USB-CANmodul2 from SYS TEC electronic [89]. The communication protocol used to communicate with the software is CAN bus. The drivers needed to communicate with the hardware are visualised in Figure 7.6. To use CAN with Linux, additional SocketCAN drivers are required. Communication with WinCC OA is established using the OPC Unified Architecture (OPC UA) protocol and an OPC UA CAN server. The OPC UA communication protocol is an alternative protocol to DIM and uses the same server-client principles.

The UI panels in WinCC OA display the sensor values. This is done in the form of a table for the most recent values, and in four graphs displaying the time evolution of the sensor values. The dew point is calculated and displayed for all the enclosing containers. Combined, this gives a detailed overview of the environmental conditions. Alarms are set for all monitored values to indicate critical temperatures and humidity. The alarm limits can be set in an expert panel.

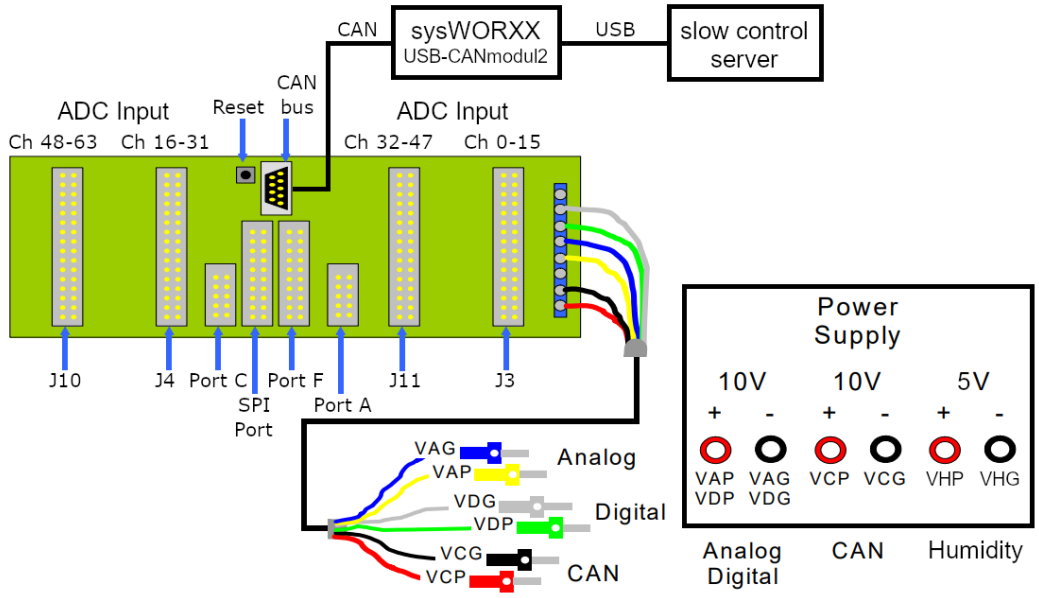


Figure 7.5: Schematic overview of the ELMB and its power connection. The board is connected to the control machine over the CAN bus. The sensors are connected to the ADC Input channels. Based on references [87, 88].

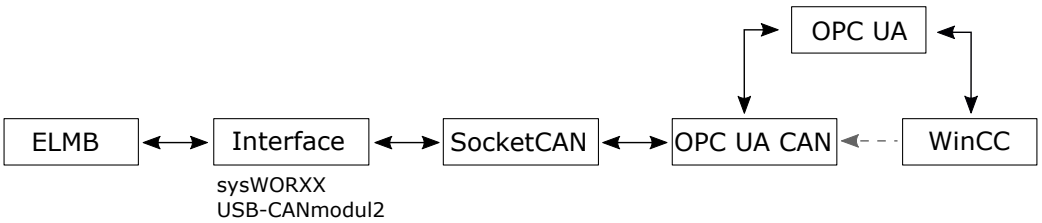


Figure 7.6: Visualisation of the communication layers between WinCC OA and the ELMB.

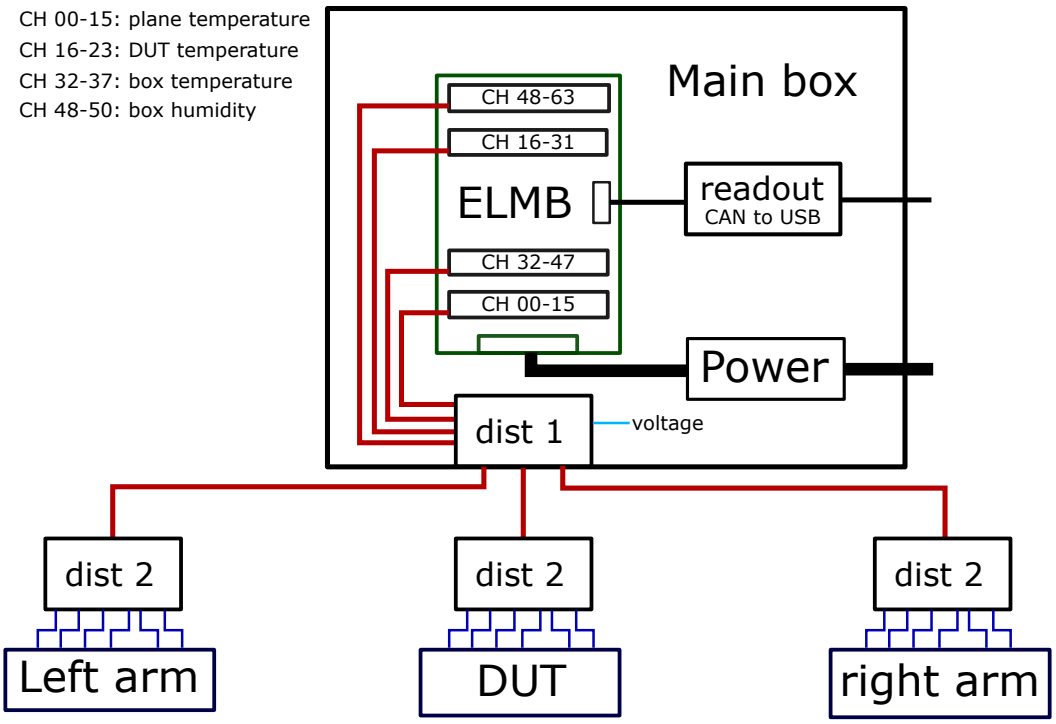


Figure 7.7: Visualisation of the environment box containing the ELMB and the distribution of the channel links to the telescope components.

7.2.2 The Environment box

The cabling of the temperature and humidity sensors, as well as the ELMB, has proven to be quite cumbersome during operation. Therefore, the setup is upgraded by combining all hardware components into a single monitoring box. The layout of the components and the channel assignment of this Environment box are given in Figure 7.7.

The box contains the ELMB, internally converting the CAN signal required by the ELMB to an USB signal for easy connection to the ECS machine. The operation voltage of two times 10 V for the ELMB, as well as the 5 V for the humidity sensors, is provided by a power converter inside the box. The box itself is connected via a standard power plug. The routing of the environmental sensors is enhanced by introducing signal distributors. The ELMB is connected to a distributor 1 inside the box, re-routing the 64 channels of the ELMB to three outputs for the left and right arm of the telescope, as well as the DUT. From there, a distributor 2 splits the channels for the individual sensors. The idea is that all three components of the telescope require the same environmental monitoring.

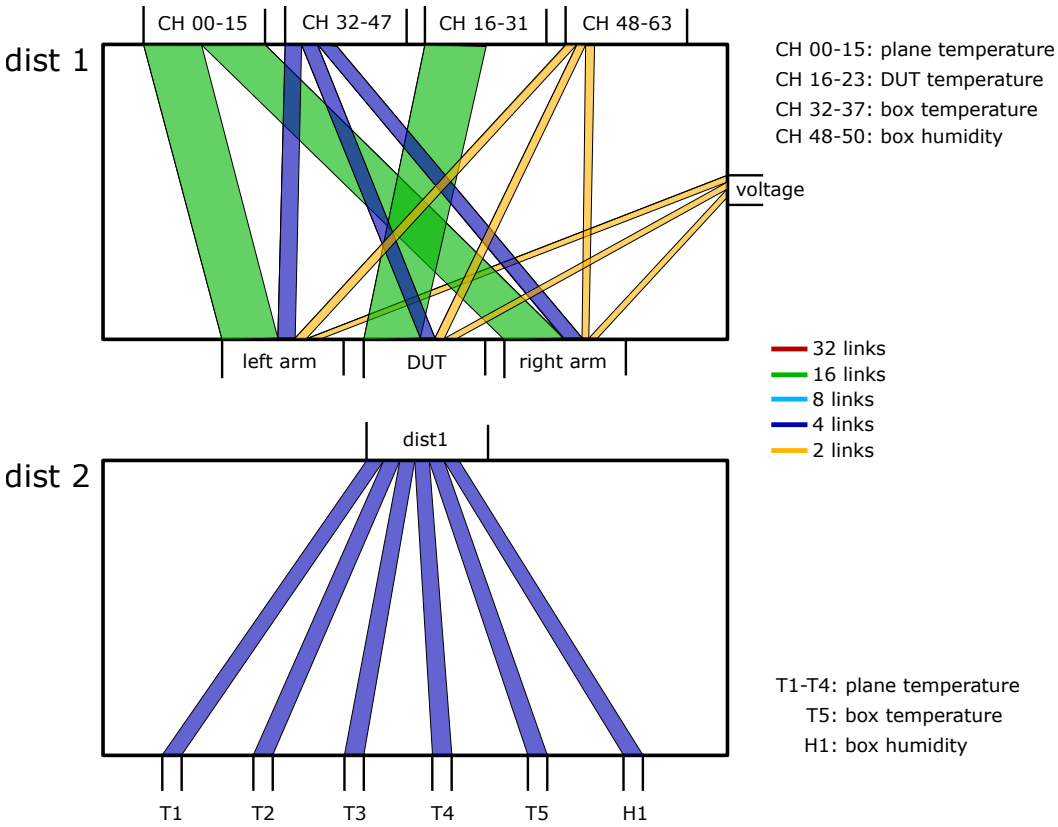


Figure 7.8: Visualisation of the internal routing of the wires inside the distributor 1 (dist 1) and distributor 2 (dist 2). Two links correspond to one ELMB channel.

The internal routing of the distributors is given in Figure 7.8. One ELMB channel corresponds to two links. The distributor 1 provides the channels for four temperature sensors, used to monitor the temperature of the sensor planes, and one temperature and humidity sensor each, to monitor the temperature and humidity of the enclosing containers. The inputs are routed to the corresponding ELMB channels; each output is connected to a distributor 2 of a telescope component to be monitored. The distributor 2 provides the channels for five temperature sensors and one humidity sensor. The outputs are connected to the designated sensor.

The temperature sensors are four-wire sensors, thus taking four links (two channels) per sensor, whereas the humidity sensors require two links (one channel) per sensor. The distributors are connected over 32 link cables, resulting in clean routing and cable management. The Environment box, together with the distributors, handles the assignment of the ELMB channels for all sensors and trivialises the voltage supply, making the environmental monitoring a plug-and-play system.

7.3 The Motion component

The Motion component is added to the Timepix4 ECS as a new element. It is used to remotely adjust the main telescope stage to align it with the beam. Further, the motion system is used to translate or rotate the DUT. This allows performing tests on the device based on its position or angular orientation. The hardware, drivers, software, and UI panels are described in this section.

7.3.1 Hardware

The translation and rotation of the telescope stages is performed with precision translation and rotation stages from PI [90], FESTO [91], and Phytron [92]. The ZSH step motors from Phytron are used to precisely move the main telescope stage in x and in y. They are built for harsh environmental conditions and have a step count of 200 (corresponding to a step angle of 1.8°). Motor rotation is converted to linear translation. In x, this is done with a screw drive, in y with a belt connected to a pulley.

For the translation of the DUT stages, high-precision translation and rotation stages from PI and FESTO are used. The repeatability of all stages is below 2 µm. In addition, the DUT can be rotated for angular analysis of the DUT. For this purpose, PI rotation stages are utilised. Both the centre and outer stage of the telescope allow the device to spin at an angle ϕ around the y-axis. The centre stage also allows tilting the device with a goniometer at an angle θ, with respect to the horizontal plane.

All the motors provide position feedback. To calibrate the position, a home position is defined. For the PI and FESTO stages used for the DUT, this is calibrated internally to be at the zero position of the translation or rotation stage. For the main telescope, a magnet is attached to the telescope stage for both x and y directions. Magnetic sensors on the non-translating part of the telescope are used to define the zero position. The same approach is used to determine the translation limits by placing additional magnetic sensors at the maximum and minimum translation points of the main telescope stage.

For safety purposes, the power to all motors is provided via a single power line and can be cut by pressing an emergency button. The button is used if the operator notices that the movement of the system could damage the telescope. This could, for example, occur when a cable is routed incorrectly and has too little flex to allow the full range of movement. The positional information is read out from the motors via motion controllers. They are powered individually and are used to initialise the motion stages. In case of errors, an error flag is set.

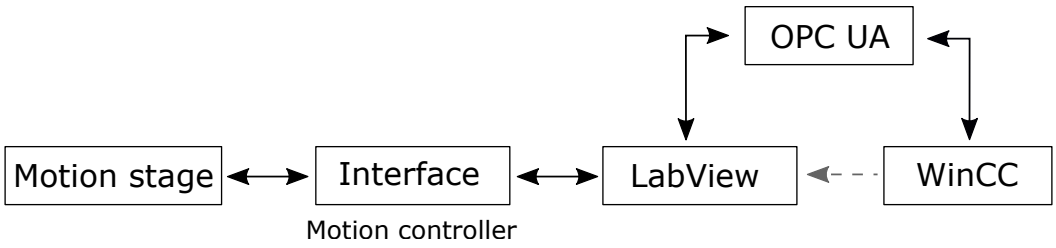


Figure 7.9: Schematic of the communication layers between WinCC OA and the motion stages.

7.3.2 Drivers and software

Communication between the motion controllers and WinCC OA is established in two stages. The complete chain is given in Figure 7.9. The translation and rotation information from the motion controllers is fed back to a Windows machine running the industry-standard LabVIEW [93]. The data is converted and transmitted to WinCC OA over OPC UA, to integrate the system into the main timepix4 ECS. LabVIEW is set up as the server, providing the values read by the hardware, while WinCC OA is set up as the client mirroring those values.

The information provided by the motion stages includes the position (in x, y) for the translation stages, or the angle (in ϕ, θ) for the rotation stages, as well as the target position (or angle). A byte array containing truth values provides information about the status of the motion stage. The status indicates whether the stage is in the home position, in error, or currently moving. In addition, a set of commands is implemented to move the stage to the home or target position, and a command is provided to reset the motion controllers in case of errors. Finally, a heartbeat is implemented by varying a single byte between zero and one every second.

7.3.3 Operator panels

The operator panels for the motion stages are used to remotely move the Timepix4 telescope stages during beam time. A selection of panels is added to Appendix B. The main panel is organised into three tabs: one for the main telescope stage, and one for the centre and outer DUT stage each. The functionality is very similar between all of them. Therefore, a panel reference is created. The reference panel supports all four available degrees of freedom: translation in x and y, and rotation around ϕ and θ. Because the rotation is not available for all telescope stages, the reference panel is configured to display only the available degrees of freedom. The rotation for the DUT stages is named spin (ϕ) and tilt (θ) to be more intuitive.

Figure 7.10: Reference panel for the centre DUT stage of the Motion component. The active motion stage is highlighted in red.

The reference panel for the centre DUT stage is shown in Figure 7.10. All motion, whether translation or rotation, follows the same principle: a target position or rotation is set, and the stage is moved to that value when pressing the `MOVE` button. Therefore, the motion for every degree of freedom is again implemented by reference. This way, all the main functionality has to be implemented only once, and the motion stage and axis of movement are provided via parameters.

The target position of the motion stage can be set. Below, the actual position is displayed. A text gives feedback if the target value has been applied to the motion controller. The status of the motion stage is displayed. The `UNKNOWN` state is the default state when no connection is established with the motion controllers. The `HOME OK` and `HOME BAD` states are set dependent on whether or not the home position is calibrated for the motion stage done by pressing the `HOME` button.

The motion stage can only be moved to the target position if the stage is not in `ERROR`. The `ERROR` state is set when the error flag of the motion controller is raised. By pressing the `RESET ERRORS` button on the main panel, the motion controllers are reset to clear the error. When the motion stage is moving, the state will enter the `MOVING` state accordingly. At the target position, the state is updated to the `OK` state. To cancel the movement of the stage, a `STOP` button can be pressed. In addition, a `STOP ALL MOVEMENT` button on the main panel will stop moving all the stages. All states are colour-coded for easy identification by the operator.

To be consistent with the other panels of the system, the project can be stopped from the main panel by pressing the `STOP PROJECT` button. Additionally, the path for automated data logging is displayed. On the right side of the main panel, the actual position information for all movement stages is summarised to provide a quick overview of the complete system. The panel can be extended to display an additional navigation panel. A figure of the telescope shows which part of the telescope is moved by the selected motion stage highlighted in red. The motion project is integrated into the main Timepix4 ECS, along with the other components.

7.4 Conclusion for the control system

The ECS of the Timepix4 telescope has been finalised. All components are fully implemented to provide control over the complete system. The power sources of the telescope can be controlled and monitored by an adaptive communication server, enabling easy switching between power supplies based on the needs of the system and making it suitable for future integration with other systems. The environment can be precisely monitored, including temperature, humidity, and information on the dew point relevant to the operation of the system. The sensor monitoring is designed as a plug-and-play system, making it an easily transferable standalone component. The motion stages of the telescope can be moved and operated remotely. This allows for positional and angular studies of the DUT.

Safety features are implemented to prevent damage to the hardware. Implemented alarms, limits, and logging provide a safety layer for the system and provide additional important information for data analysis. The system has been extensively used, tested and improved over the span of many test beams.

Due to their versatility, the ECS components can be easily adapted to other systems. The intention is to reuse components for the planned beam-monitoring upgrade at the Large Hadron Collider beauty (LHCb) experiment once the new detector is constructed. Therefore, the system not only provides precise control over the telescope itself but also serves as a starting point for future developments, saving time and resources.

8 Characterisation of beam losses

In this chapter, the impact of losses on the Large Hadron Collider beauty (LHCb) experiment is analysed for the first time. If the losses from the Target Collimator Tertiarys (TCTs) are large enough, they can have an impact on the physics result. Therefore, it is important to know the magnitude of such machine-induced background (MIB) particles on physics operations (refer to section 2.2.2). Characterising the losses can help to identify and potentially exclude them for physics analysis.

For this study, datasets have been recorded in specific fills in which losses are induced intentionally along the LHC ring, referred to as loss maps. The loss map methodology is described in Section 8.1. The analysis of the losses is performed in Section 8.2. The detector response to the losses is studied and characterised over time. Finally, the impact on physics operations is estimated.

8.1 Loss maps

Loss maps are a standard procedure for the operation of the Large Hadron Collider (LHC), performed to validate that the collimation system (refer to Section 2.2.4) is correctly protecting the machine components. For the loss maps, the LHC is filled with a different filling scheme compared to nominal physics operations. Two bunches are filled per beam with a semi-nominal bunch population of 1×10^{11} protons. They define the beam orbit and give a reference proton-proton collision signal at the experiments. Then, several bunches are filled with a lower bunch population of 1×10^{10} protons, so-called pilot bunches. The intensity of these bunches is reduced to minimise the risk of damaging the machine.

Losses are artificially induced [18], covering different loss scenarios: losses from the betatron source for beam 1 and beam 2 in the horizontal and vertical plane, and losses from the momentum source for positive and negative momentum variation. Every time losses occur, the beam intensity decreases. The losses are measured all around the LHC by Beam Loss Monitors (BLMs), resulting in a map of the losses as a function of the location s at the LHC ring. This is exemplarily shown in Figure 8.1.

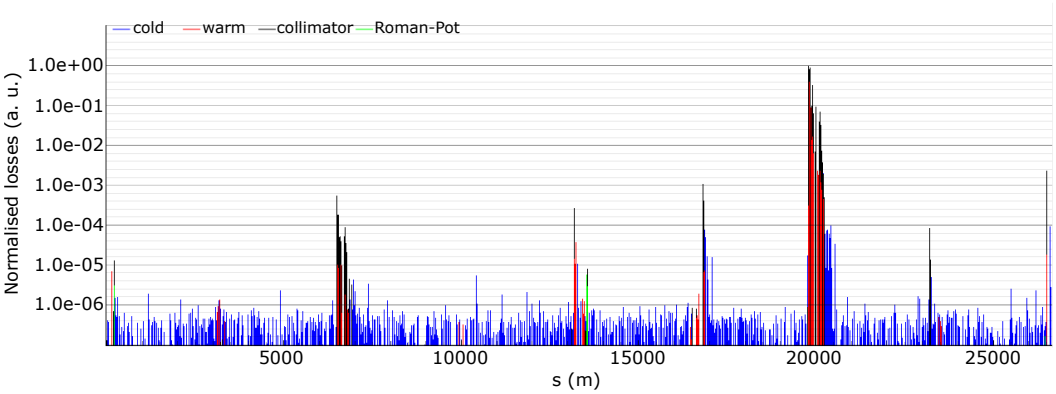


Figure 8.1: Beam loss measurements around the LHC ring, exemplary for a loss map fill on June 17th 2024.

8.1.1 Artificial creation of losses

To artificially create losses from the betatron source, the pilot bunches are exited before the collimators at the betatron cleaning section in IR7. This means they are blown up in either the vertical (V) or horizontal (H) plane, causing losses when they hit the collimator yaws. The secondary particles accompany the beam, reach the LHCb TCTs, and produce particle showers observed by the detector. This is done for both beams, resulting in the four scenarios B1V, B1H, B2V, and B2H. The number of protons per bunch is precisely known, so with the whole pilot bunch depleted, the loss readings of the BLMs behind the collimators give a representative value of the losses at the collimators.

To obtain losses from the momentum source, the semi-nominal bunches are used near the end of the fill. By varying the radio frequency (RF) frequency of the RF cavities, the average momentum of the protons in the beam changes, causing the beam to scrape at the horizontal collimator yaws of the momentum cleaning facility at IR3. This momentum variation is referred to as $\pm \mathrm{d}p/p$ for the positive and negative momentum variation, respectively. The secondary particles then accompany the beam to shower onto the LHCb detector from the TCTs. The beam orbit changes, affecting the beam across the entire LHC. This means that effects like beam dispersion as described in Section 2.2.3 play a role. Figure 8.2 shows the scenarios as a function of time for a typical loss map fill.

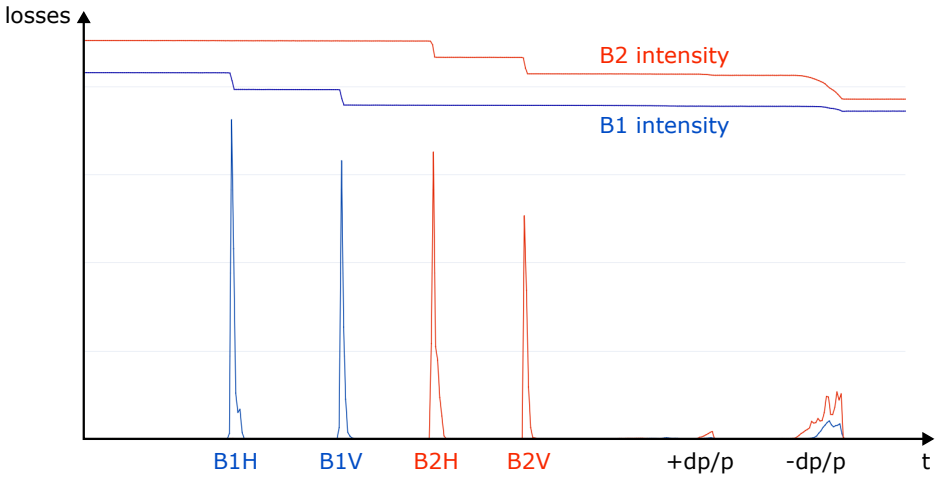


Figure 8.2: Schematic depiction of the beam intensity and the loss scenarios as a function of time.

8.1.2 Dataset

The data considered for this analysis is from 17th June 2024. The relevant bunches filled into the LHC and their respective population are given in Table 8.1. The bunches, identified by their bunch identification number (BxID), are accelerated to the energy of 6.8 TeV. Two pilot bunches are injected for each beam to create losses from the betatron source. Bunches 2085 and 1111 are excited in the horizontal plane (B1H and B2H scenario) and bunches 2200 and 1511 in the vertical plane (B2V and B2H scenario) for beam 1 and beam 2, respectively. The bunches 1785 and 0891 are semi-nominal bunches brought to collision at IP8 and relevant for the momentum variation ($\pm \mathrm{d}p/p$ scenario). The number of protons lost is given for each scenario and for the nominal bunches in both $\pm \mathrm{d}p/p$ scenarios. The losses for the momentum variation, especially the positive variation, are smaller by one to two orders of magnitude.

The LHCb detector is running in a special configuration for this study to avoid damage to sensitive detector parts from unexpected radiation caused by the loss map fill. The luminosity levelling procedure of LHCb (refer to Section 3.1.3) is disabled, and the Vertex Locator (VELO) is turned off and moved out. The subdetectors taking data are the Scintillating Fibre Tracker (SciFi Tracker), electromagnetic calorimeter (ECAL), hadron calorimeter (HCAL), and muon stations. Additionally, data is taken with the Probe for Luminosity Measurement (PLUME) detector and the Beam Conditions Monitor (BCM); however, they show no distinguishable signal from background and are therefore not further considered.

Table 8.1: The relevant bunches with their BxID and their bunch population at injection, as well as the number of protons lost.

BxID	Bunch type	Scenario	Bunch population	Protons lost
0891	nominal	$+dp/p$ (B2)	0.99×10^{11}	1.7×10^{8}
0891	nominal	$-dp/p$ (B2)	0.99×10^{11}	4.9×10^{9}
1111	pilot	B2H	0.99×10^{10}	9.8×10^{9}
1511	pilot	B2V	1.01×10^{10}	9.9×10^{9}
1785	nominal	$+dp/p$ (B1)	0.97×10^{11}	2.2×10^{8}
1785	nominal	$-dp/p$ (B1)	0.97×10^{11}	1.0×10^{9}
2085	pilot	B1H	1.01×10^{10}	9.9×10^{9}
2200	pilot	B1V	0.95×10^{10}	9.3×10^{9}

The collimator settings during the fill are given in Table 8.2. They are expressed in multiples of the local root mean square (RMS) of the beam [94, 20]

$$\sigma(s) = \sqrt{\frac{\beta(s)\epsilon_{\mathrm{n}}}{\beta_{\mathrm{rel}}\gamma_{\mathrm{rel}}}}, \tag{8.1}$$

where $\beta(s)$ is the optical Twiss function, describing the strength of the beam spread as a function of the location s along the beam; $\epsilon_{\mathrm{n}} = 3.5\,\mu\mathrm{m}$ is the normalised transverse emittance, describing the occupied area of the beam particles in a position-momentum phase space; and β_{rel} and γ_{rel} are the relativistic parameters.

8.1.3 Expectations

Simulations of MIB particles in 2011 at 7 TeV [95] show the fraction of protons lost at the LHCb TCTs from the various MIB sources (see Table 8.3). Combining the contributions of the Target Collimator Tertiary Vertical (TCTV) and Target Collimator Tertiary Horizontal (TCTH) collimators, the largest simulated loss contribution originates from elastic beam-gas interactions (13.5×10^{-3} protons lost for beam 1 and 18.2×10^{-3} for beam 2). The losses originating from the betatron collimation are an order lower, with a higher contribution from beam 1 (11×10^{-4} protons lost for beam 1 compared to 3.7×10^{-5} for beam 2). The losses originating from momentum collimation are of the same order as betatron losses, but show the opposite behaviour with a higher contribution from beam 2 (1.9×10^{-4} protons lost for beam 1 compared to 5.6×10^{-3} for beam 2).

Table 8.2: The relevant collimator settings for LHCb during the loss map fill [96]. The collimator settings are given as multiples of the RMS (Equation 8.1).

collimator	IR3 [σ]	IR7 [σ]	IR8 [σ]
TCP	15.0	5.0	—
TCSG	18.0	6.5	—
TCLA	20.0	10.0	—
TCT	—	—	11.5

Table 8.3: The simulated fraction of protons lost at the LHCb TCTs from different loss sources. For the TCTV, a purely vertical halo, and for the TCTH, a purely horizontal halo is assumed.

Source	Collimator	Beam 1	Beam 2
Betatron	TCTH	1.8×10^{-4}	3.1×10^{-5}
Betatron	TCTV	9.2×10^{-4}	6.4×10^{-6}
Momentum	TCTH	8.2×10^{-5}	5.4×10^{-3}
Momentum	TCTV	1.1×10^{-4}	2.2×10^{-4}
Beam-Gas	TCTH	7.8×10^{-3}	8.7×10^{-3}
Beam-Gas	TCTV	5.7×10^{-3}	9.5×10^{-3}

Examining the layout of the LHC, considering the position of the momentum and betatron cleaning sections at IR3 and IR7, the LHCb detector at IR8 is located directly next to the betatron cleaning section. This means for beam 1 the betatron losses directly reach the detector, while for beam 2 the losses traverse almost through the entire LHC ring before reaching the experiment. Therefore, a higher contribution to betatron losses is expected from beam 1. In addition, beam 2 losses are expected to be partially absorbed by the muon shielding of the LHCb detector.

For losses from the betatron source, the detector response is expected to show a short, elevated signal at the time of bunch excitation. For the momentum source, losses are expected during the momentum variation, adding a signal contribution to the proton-proton collisions of the semi-nominal bunches. In both cases, global and local beam-gas interactions with the residual gas in the beam pipe contribute to a background signal in the detector. Other effects, such as the intrinsic detector noise, also contribute to the background.

8.2 Analysis of losses

Interesting for the operation of the LHCb detector is the contribution of the MIB background to the physics signal, the temporal and spatial distribution and the fraction of particles reaching the detector when scattering at the experiments TCTs.

In this section, the detector response to the losses is studied over time to identify the time window of the loss signal. Afterwards, the number of hits in the subdetectors is given, depending on the losses, to quantify the signal distribution. Then, the hits from the losses are separated from the background, and the localisation of the losses within the detector is studied. Hitmaps are created for the subdetectors, and partial tracks are reconstructed with the SciFi Tracker and muon stations. Finally, the BLM values of the collimators during the loss maps are used to scale and compare the loss contributions to what is expected in physics operations.

8.2.1 Identification of losses

For the loss maps, only specific bunches of the LHC are filled. Therefore, the trigger is set to *passthrough* mode. This means the trigger is not removing any of the data. The information is not served per bunch crossing, but instead per orbit. The relevant BxIDs of the pilot and nominal bunches are known from the filling scheme. The rest of the bunches are taken randomly every other orbit to account for detector effects and possible spill-over. Spill-over refers to signals spilling over into neighbouring bunches due to imperfect time resolution and detector calibration.

The losses are identified by running a peak-detection algorithm over all the data, split by BxID. The signals from each subdetector are plotted against time. The peak detection is then calibrated so that it does not trigger on every background fluctuation, yet still filters out all significant peaks in the data. The time distribution of the different loss scenarios is given in Figure 8.3, exemplary for the ECAL. A 250 ms binning is used. The signal is time-aligned for beam 1, with no significant contribution distinguishable from the background in the neighbouring bunches. For beam 2, no signal can be separated from the background. The momentum variation differs from the betatron losses, as the bunch-crossing signal underlies the loss signal. This case is therefore handled separately at the end of this study in Section 8.2.4.

Only losses from the beam 1 betatron source are clearly distinguishable from the background. As such, the focus will be on the B1H and B1V scenarios from this point onward. The plots for the B1V scenario are presented in the main text, and the plots for the B1H scenario are attached to Appendix C. The time distribution plots for the B1V scenario for the ECAL, HCAL, SciFi Tracker, and muon systems are shown in Figure 8.4.

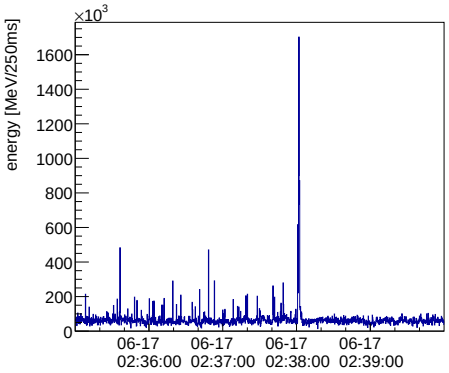
It is possible to distinguish three regions across all subdetectors. The first region is defined around the loss signal in the range of 2 s before the maximum of the loss peak to 8 s after the maximum. The second region is the region before the loss signal. It is called the pre-loss region. It includes elastic beam-gas interactions between the circulating bunches and the residual gas. This region is used for background subtraction. The last region is the region after the loss signal. The bunch is fully depleted, leaving only detector noise. This region is used for cross-checks with the pre-loss region. The choice of the borders of the regions is based on the shape of the loss signal, seen in Figure 8.5, where the signal region is zoomed in, and the average signal of the pre-loss region is subtracted.

8.2.2 Quantification of losses

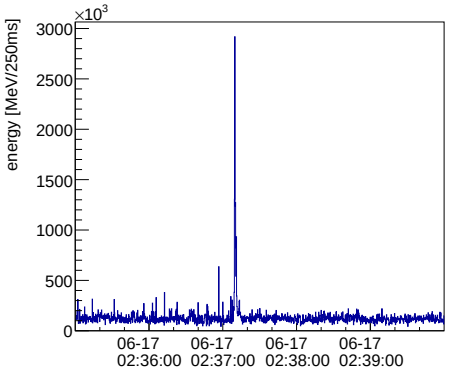
The signal strength is quantified in two ways. First, the maximum losses are compared between the different detectors. The values are summarised in Table 8.4. Horizontal losses are consistently higher by a factor of two to three compared to vertical losses. Additionally, by comparing the values to the average value of the semi-nominal bunch crossing, the relative contributions across the different detectors are inferred. The absolute value of the muon contribution over 100 % is not concerning, as the semi-nominal reference signal cannot be directly compared to the signal during physics operation due to different machine conditions. Therefore, the relative values between the subdetectors are of relevance.

The losses in the muon stations have the highest relative contribution, followed by the losses in the HCAL and the ECAL. This could be related to the cleaning efficiency of the collimators. Muons are the least affected by absorbers and thus most likely to reach the detector even when the source of the losses is far away. Hadrons and electromagnetic particles, on the other hand, interact via the strong and electromagnetic forces and are thus more affected by the cleaning process. Especially, the low-energy electrons and photons scatter more and are absorbed more easily than heavy, higher-energy hadrons.

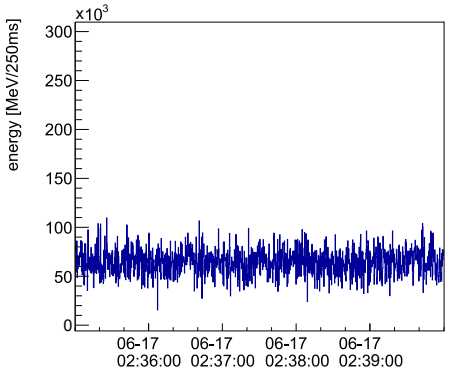
In a second step, the signal is integrated over the whole timespan of the signal region. This gives the total losses from excitation of the bunches and removes the dependency of the signal strength on the chosen binning. The total number of protons lost at the collimators at IR7 is known, as the entire pilot bunch is lost. This can be used to normalise the signal in the subdetectors to the total number of protons lost at the Target Collimator Primarys (TCPs). The integrated and normalised losses are given in Table 8.5. The normalised number of hits in the subdetectors is of the order $\mathcal{O}(1 \times 10^{-5})$ to $\mathcal{O}(1 \times 10^{-6})$, meaning around one MIB particle is registered for every 10 000 protons lost at IR7. Again, horizontal losses are twice as high as vertical losses.



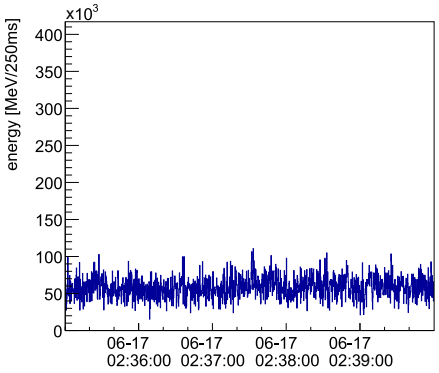
(a) B1V



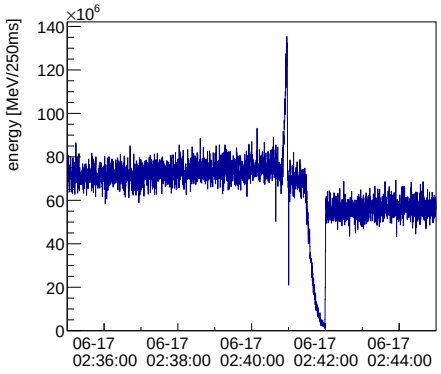
(b) B1H



(c) B2V

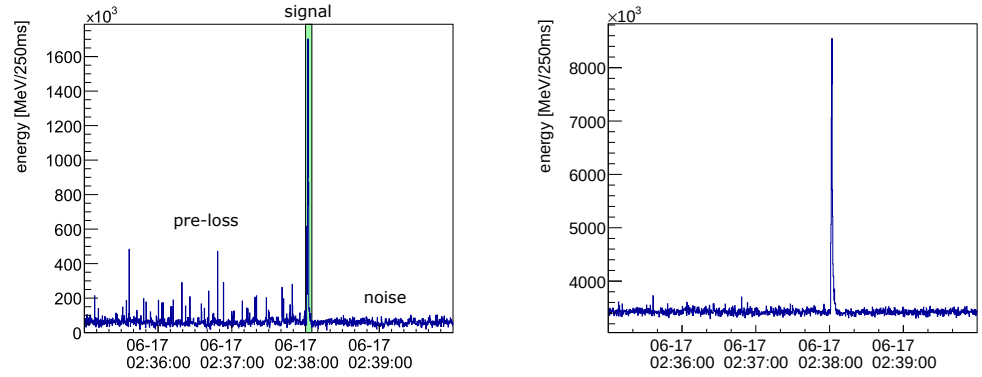


(d) B2H



(e) $\pm dp/p$

Figure 8.3: The loss scenarios, exemplary for the ECAL, for the beam 1 betatron source (top), for the beam 2 betatron source (middle) and the momentum source (bottom). Only noise is observed for the beam 2 scenarios.



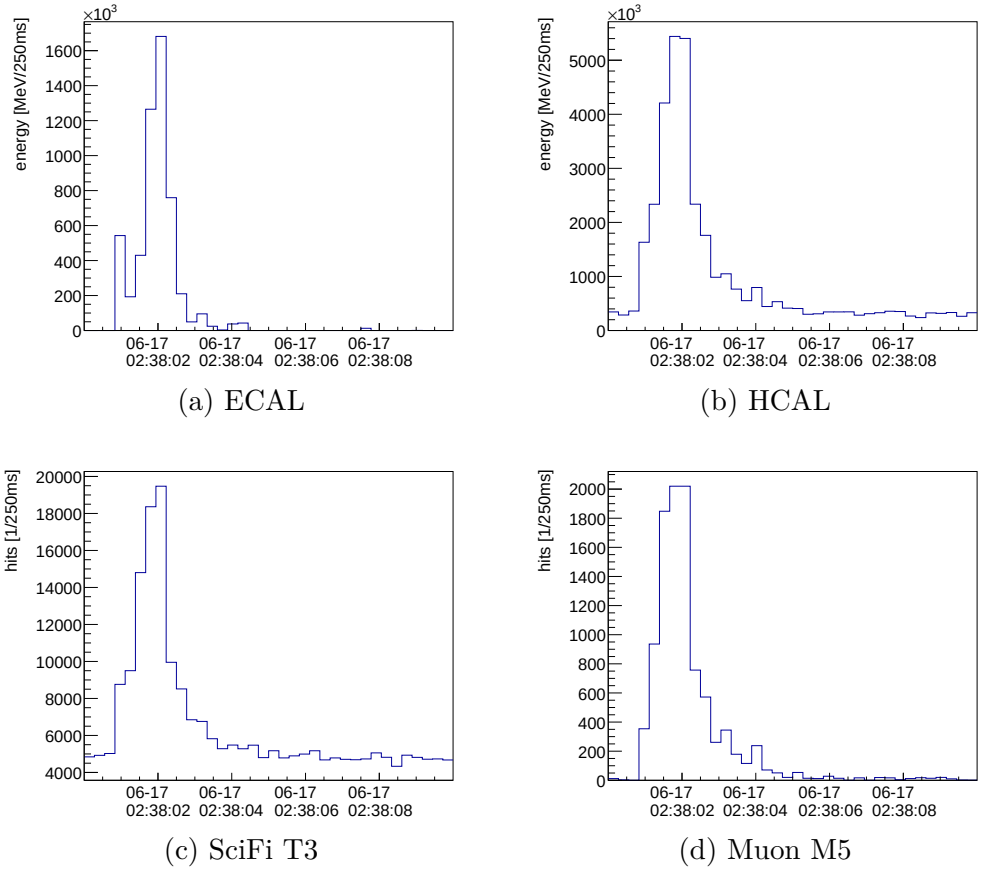
(a) ECAL (b) HCAL

(c) SciFi T3 (d) Muon M5

Figure 8.4: The B1V losses in the subdetectors. The SciFi Tracker station T3 and muon station M5 are exemplary for the other stations. For the ECAL, the three regions are visualised.

Table 8.4: Maximum loss signal in 250 ms, as well as relative maximum signal compared to the average signal from the semi-nominal reference bunch.

	B1H (max)	B1V (max)	B1H/B1V	B1H (%)	B1V (%)
ECal [MeV]	2.9×10^6	1.7×10^6	1.7	3.9	2.6
HCal [MeV]	1.4×10^7	5.4×10^6	2.6	22.6	8.1
SciFi hits	1.1×10^5	5.2×10^4	2.1	27.5	12.5
Muon hits	3.3×10^4	1.2×10^4	2.8	137.5	50.0



(a) ECAL (b) HCAL

(c) SciFi T3 (d) Muon M5

Figure 8.5: The signal region of the B1V losses for the subdetectors with subtracted average pre-loss background. The SciFi Tracker station T3 and muon station M5 are exemplary for the other stations.

Table 8.5: Integrated number of hits in the subdetectors normalised to the number of protons lost at IR7.

	B1H (integral)	B1V (integral)	B1H/B1V
ECal	4.33×10^{-6}	2.28×10^{-6}	1.9
HCal	3.46×10^{-6}	1.86×10^{-6}	1.9
SciFi	3.61×10^{-5}	1.74×10^{-5}	2.1
Muon	1.08×10^{-5}	5.11×10^{-6}	2.1

8.2.3 Localisation of losses

In the next step, the localisation of the losses in the LHCb coordinate system is studied. This is done to characterise the shape of losses and find asymmetries that contribute to the nominal signal during physics operations. The previously specified subdetectors that are operative during the loss maps are considered.

So-called hitmaps are created, where the reconstructed hit position from the particles is projected onto the x-y, z-x or z-y plane of the LHCb coordinate system. The z-axis of the histograms shows the detector signal strength normalised per event. For normalisation, it is assumed that there is only one associated loss event per LHC orbit due to the filling scheme. The time window identified in the previous section is used to integrate the loss signal. The same is done for the pre-loss and the noise region. The histogram of the pre-loss region is subtracted from the histogram of the loss signal region to remove the background.

For the calorimeters, the x-y plane is relevant. Hitmaps are created for both the number of hits and the reconstructed energy deposition in the detectors, given on the z-axis. For the ECAL they are given in Figure 8.6 (a) and (b), respectively. The hole in the middle of the plots is the detector opening for the beam pipe. The other white spaces are due to low event counts for the loss event or dead links in the detector. The hitmaps for the calorimeters show a step structure going from the inside out. This structure is a direct result of the density of the detector readout in the different regions of the calorimeters, as laid out in Section 3.1.2.

The distinction between the number of hits and energy is interesting, as it shows the different contributions from higher- and lower-energy particles. The particles are showering outwards, but the energy plots do not show this behaviour with the same intensity. This means that most of the outward particles are low-energy. In contrast, the higher-energy particles, which have a greater impact on the energy plot, are more centrally distributed, especially around the y-axis.

To compare the shape of the loss signal to nominal operations, the same hitmaps are created from the detector response created by the bunch crossings of the semi-nominal reference bunches. As there is no loss signal peak for the reference bunch crossings, the signal is integrated over 30 seconds and then normalised per event. The hitmaps from the nominal reference are given in Figure 8.6 (c) and (d).

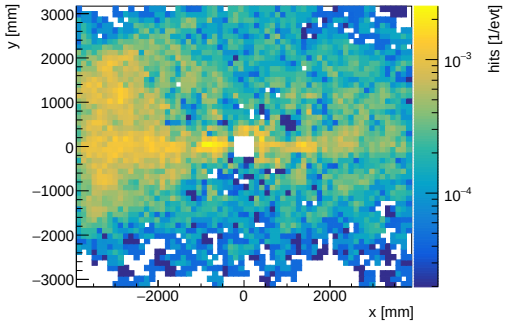
Comparing the hitmaps, there is an asymmetry with higher contribution at negative x, towards the centre of the LHC ring. The nominal signal is centred in a Gaussian distribution around the beam pipe, unlike the spread seen from the losses, as discussed before. All the plots show a higher signal contribution around $y = 0$, resulting in an eye shape. As this is seen also for the nominal reference, this behaviour is concluded to be a detector effect independent of the losses.

In addition, the ECAL losses are projected onto the x-axis. This is shown in Figure 8.6 (e) and (f). The asymmetry is clearly visible, especially when looking at the number of hits in the ECAL. The lower values near zero again originate from the hole in the detector for the beam pipe. Interesting is the peak around $-800\,\text{mm}$ in x. This peak is observed throughout all the subdetectors as a region with a high loss contribution.

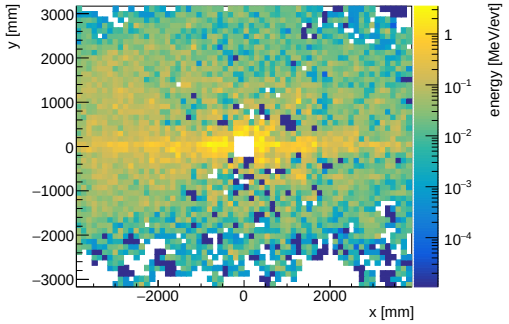
The same hitmaps are also studied for the HCAL in Figure 8.7 and for one of the muon stations in Figure 8.8. For the muon stations, no energy deposition is given; instead, the number of hits are additionally shown in the z-x plane to visualise the evolution of the losses in the z direction. The HCAL and muon stations show the same behaviour as the ECAL, except the particles are less spread out and more condensed around the beam pipe. The asymmetry towards negative x with a peak at $-800\,\text{mm}$ is confirmed. The intensity of the losses is halved between the first and the last muon station. The pixels with increased signal in the plots are due to hot links in the detector.

For the SciFi Tracker, there is no precise information on the y position of the hits before the track reconstruction. Therefore, the tracks are reconstructed using the t-track algorithm [49]. The x-y position of the tracks is given in Figure 8.9 (a) at the interface plane at the centre of the SciFi Tracker station T1. The tracking allows obtaining information on the slope of the tracks and thus on the direction of the traversing particles. The slope of the tracks at the interface plane is presented in Figure 8.9 (b). The x-axis shows the ratio x over z, and the y-axis shows the ratio y over z. This means that a value of zero indicates the particles traverse straight, and a value of one indicates the particles traverse at a 45° angle through the detector. The track information is obtained for the nominal reference bunch for comparison and is shown in Figure 8.9 (c) and (d), respectively. The projections onto the x-axis are given in Figure 8.9 (e) and (f).

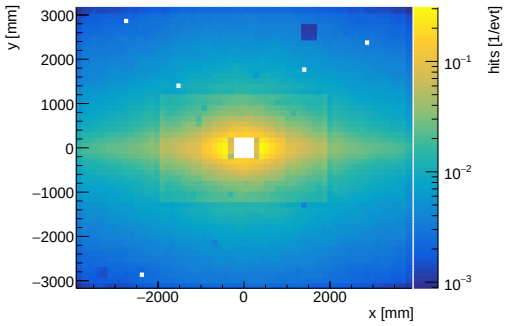
For the SciFi Tracker, the asymmetry in the losses is very pronounced with a clear maximum in x at $-800\,\text{mm}$ while remaining symmetric around y. The slope of the tracks is close to zero, especially in the y direction, compared to the nominal reference, where most of the particles have a slight angle to their track. Muon tracks are reconstructed and matched with the t-tracks of the SciFi Tracker, confirming the observations. The plots are given in Figure 8.10, reconstructed at the centre of muon station M2. The larger spread and the observed asymmetry in the x direction could be caused by the LHCb magnet, considering the magnet is bending the particles in the z-x plane. To validate whether the magnet causes the asymmetry, studies with inverted magnet polarity are planned in the future.



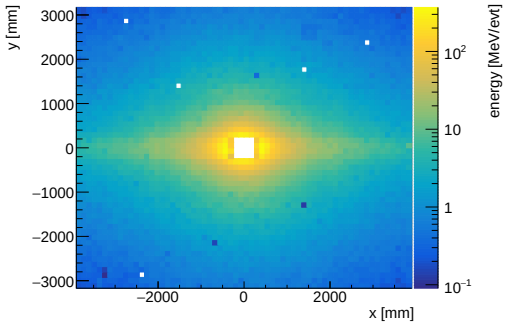
(a) ECAL loss hits x-y



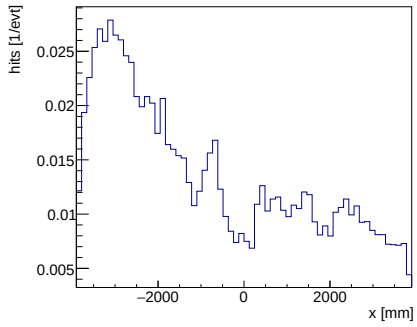
(b) ECAL loss energy x-y



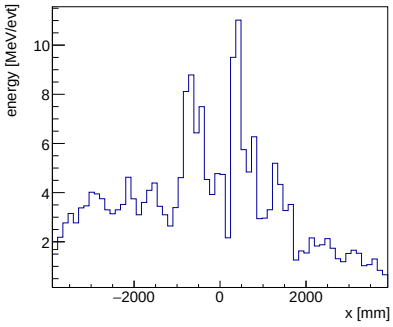
(c) ECAL nominal hits x-y



(d) ECAL nominal energy x-y

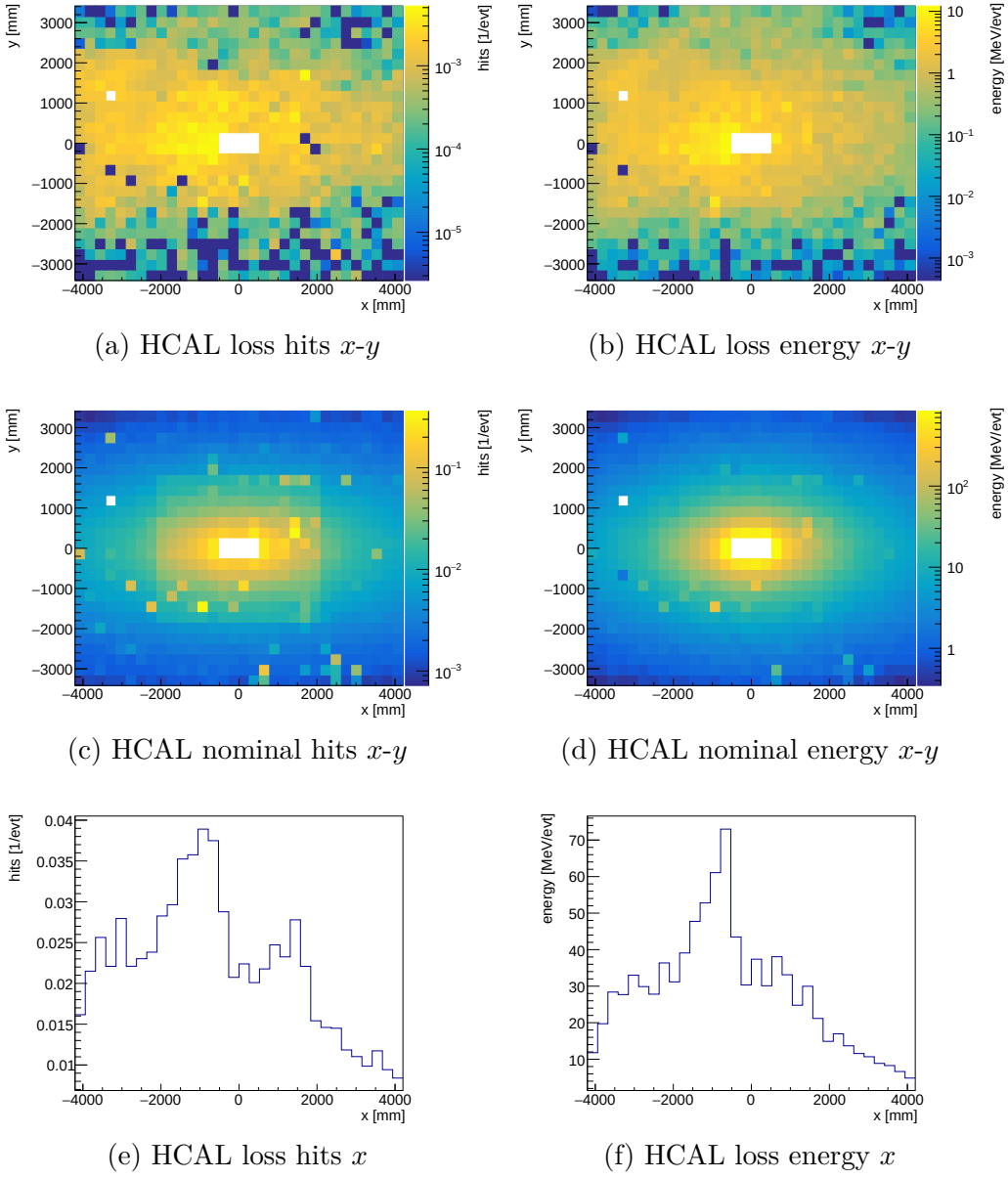


(e) ECAL loss hits x



(f) ECAL loss energy x

Figure 8.6: The number of hits (left) and energy deposition (right) in the ECAL. The signal region of the B1V losses with subtracted pre-loss background projected onto the x-y plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis (bottom).



(a) HCAL loss hits x-y

(b) HCAL loss energy x-y

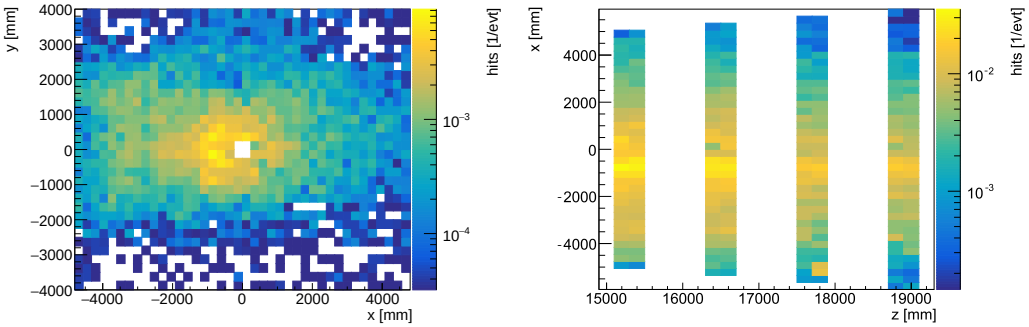
(c) HCAL nominal hits x-y

(d) HCAL nominal energy x-y

(e) HCAL loss hits x

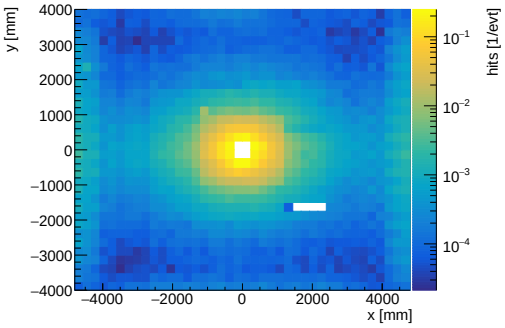
(f) HCAL loss energy x

Figure 8.7: The number of hits (left) and energy deposition (right) in the HCAL. The signal region of the B1V losses with subtracted pre-loss background projected onto the x-y plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis (bottom).

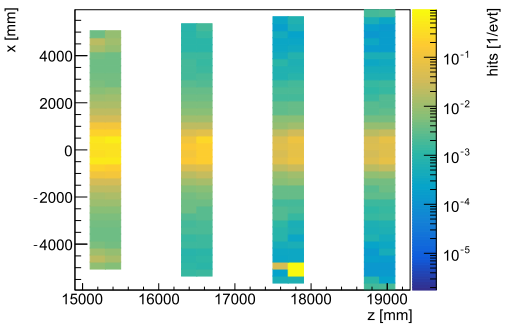


(a) Muon M2 loss hits x-y

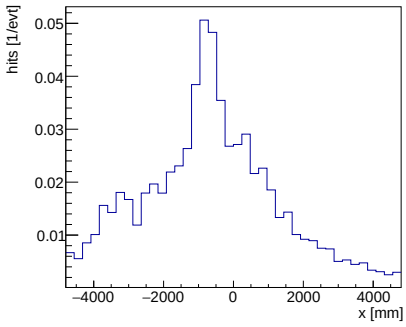
(b) Muon M2 loss hits z-x



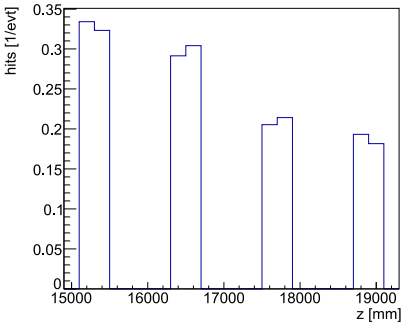
(c) Muon M2 nominal hits x-y



(d) Muon M2 nominal hits z-x

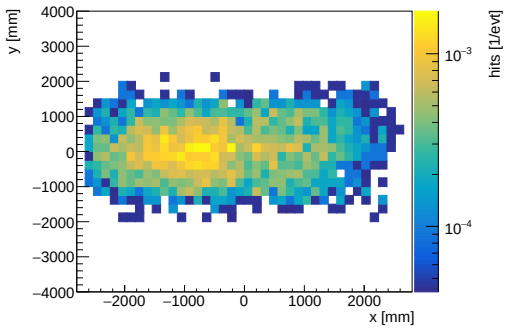


(e) Muon M2 loss hits x

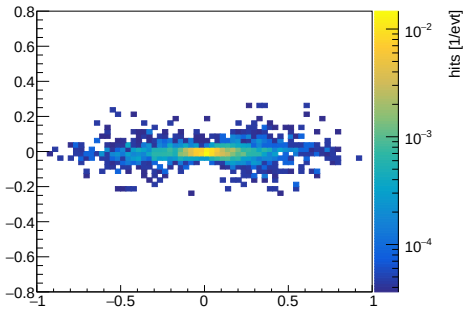


(f) Muon M2 loss hits z

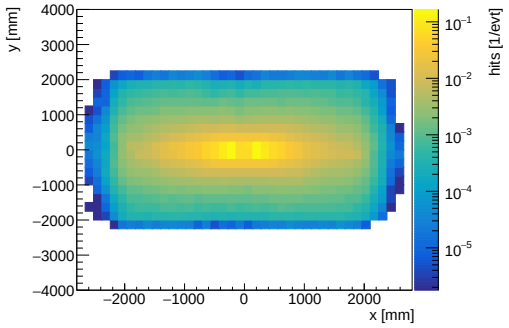
Figure 8.8: The number of hits in the muon station M2. The signal region of the B1V losses with subtracted pre-loss background projected onto the x-y plane and z-x plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis and z-axis (bottom).



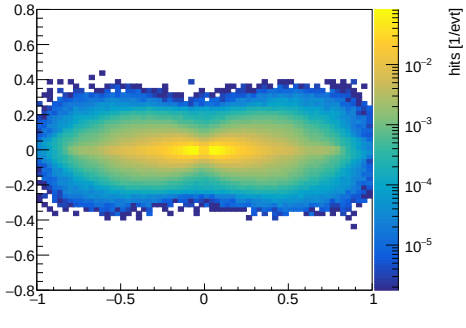
(a) SciFi T1 loss track hits x-y



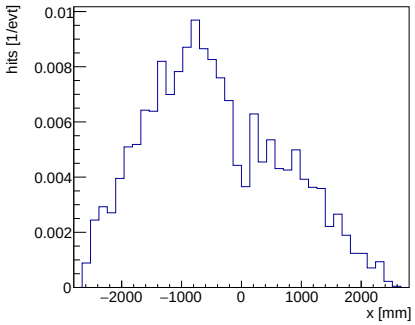
(b) SciFi T1 loss track hits x-y



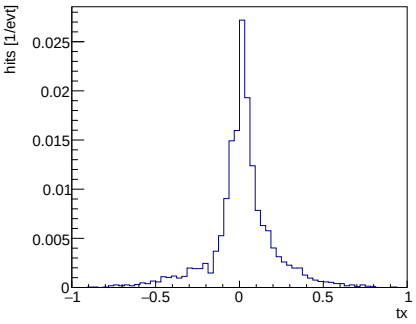
(c) SciFi T1 nominal track hits x-y



(d) SciFi T1 nominal track hits x-y

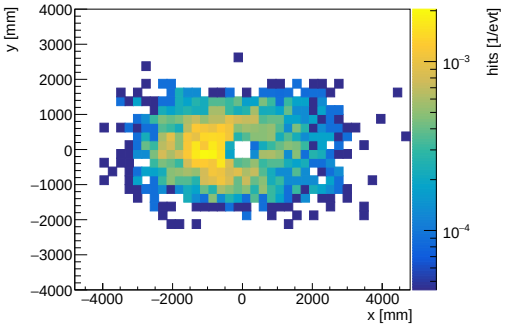


(e) SciFi T1 loss track hits x

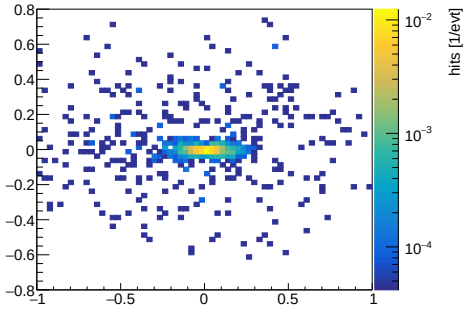


(f) SciFi T1 loss track slopes x

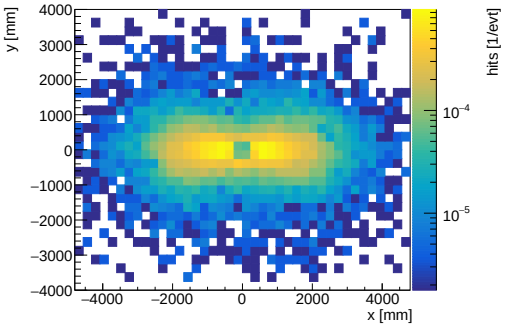
Figure 8.9: The reconstructed track hits (left) and track slopes (right) in the SciFi Tracker. The signal region of the B1V losses with subtracted pre-loss background projected onto the x-y plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis (bottom).



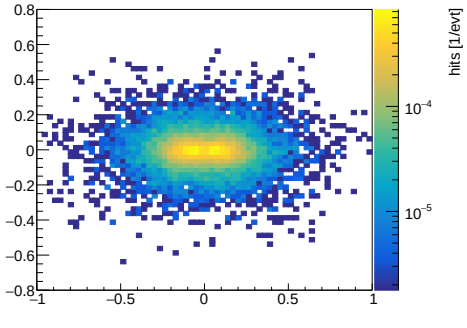
(a) Muon M2 loss track hits x-y



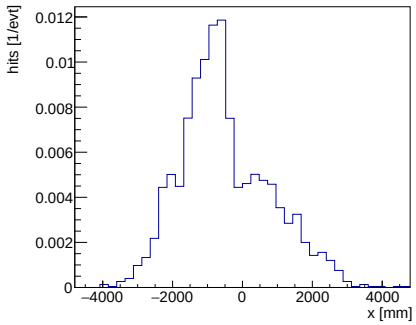
(b) Muon M2 loss track hits x-y



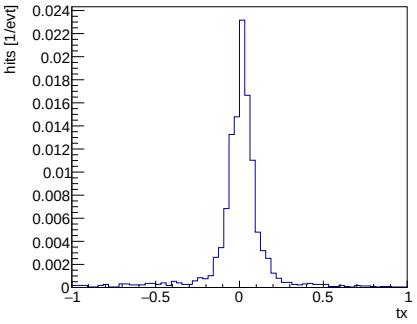
(c) Muon M2 nominal track hits x-y



(d) Muon M2 nominal track hits x-y

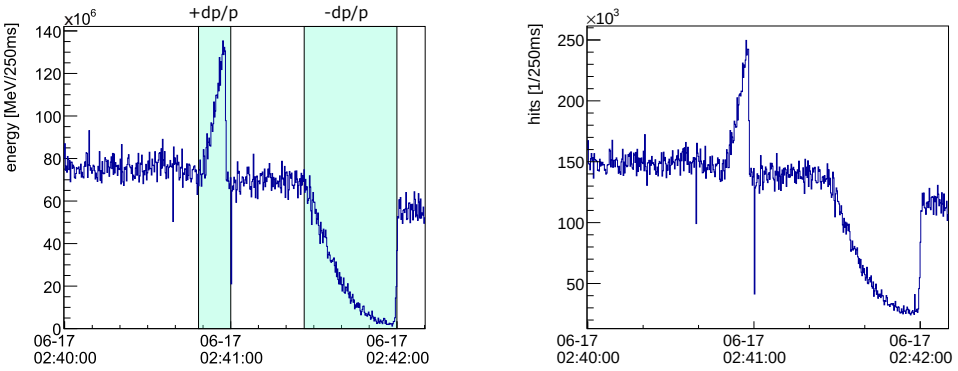


(e) Muon M2 loss track hits x



(f) Muon M2 loss track slopes x

Figure 8.10: The reconstructed track hits (left) and track slopes (right) in the muon stations. The signal region of the B1V losses with subtracted pre-loss background projected onto the x-y plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis (bottom).



(a) ECAL (b) SciFi T1

Figure 8.11: The signal response during the momentum variation, exemplary for the ECAL and the SciFi Tracker station T1. For the ECAL, the variation regions are visualised.

8.2.4 Special case momentum variation

As mentioned before, no loss signal can be observed for the $\pm \mathrm{d}p/p$ cases. As seen from the number of protons lost, the losses due to momentum variation are small compared to the betatron losses. Moreover, the losses are not isolated; there is an underlying contribution from semi-nominal collisions.

Exemplary, the detector response of the ECAL and the SciFi Tracker station T1 is given in Figure 8.11 over the time period of the momentum variation. For the ECAL, the $\pm \mathrm{d}p/p$ regions are highlighted. The same trend is observed for all other subdetectors. In the beginning, there is the reference signal from the semi-nominal bunch crossing. During positive momentum variation, the signal increases in strength; during negative momentum variation, the signal decreases. After the momentum variation, the nominal signal is reduced, corresponding to the protons lost during the momentum variation.

The increased and decreased signals are a direct result of the changed beam orbit, causing the beam to drift, resulting in a changed separation of the two beams at the interaction point (IP) (refer to Section 2.2.3). This is confirmed over the beam parameters during the momentum variation saved in the LHC online database TIMBER [97]. It logs accelerator data during machine operation. From the data, beam 1 and beam 2 are moving in different directions during the loss map, removing around 6 µm separation during the $+\mathrm{d}p/p$ variation and introducing around 25 µm during the $-\mathrm{d}p/p$ variation. This, in turn, leads to more and less head-on collisions. As a result, no losses can be distinguished in the detector for losses from the momentum source.

8.2.5 Impact on physics operations

For physics operations, it is relevant to understand the impact of losses on the nominal collision. The artificial losses created during the loss maps are not directly comparable to the losses during physics operations. The purpose of the loss map studies is to characterise and quantify the losses in an isolated environment with the dedicated fill. To estimate the impact of the losses, the BLM readings from the collimators are compared between the loss map fill and a nominal physics fill. The losses observed during the loss map fill are then scaled to those during physics operations.

The more relevant losses for the LHCb experiment are from beam 1 betatron cleaning. Therefore, the BLMs considered for this part of the analysis are the Target Collimator Primary Horizontal (TCPH), Target Collimator Primary Vertical (TCPV) and Target Collimator Primary Skew (TCPS) at IR7 behind the primary collimators of the betatron cleaning and the BLMs behind the TCTH and TCTV tertiary collimators of the LHCb experiment. The BLMs are referenced by their respective collimator. The BLM readings are obtained again from TIMBER, the LHC online database.

The naive way to determine the physics impact by comparing the BLM readings of the TCTs at P8 is not applicable; during nominal physics operations, the TCT BLM readings are dominated by the particles originating from the proton-proton collision. Therefore, the losses are measured at IR7, where they originate, and the factor $P_{7,\mathrm{ref}}/P_{7,\mathrm{H(V)},\mathrm{loss}}$ is calculated, where $P_{7,\mathrm{H(V)},\mathrm{loss}}$ are the losses at IR7 of the B1H (B1V) case during the loss map fill and $P_{7,\mathrm{ref}}$ are the losses at IR7 during a reference physics fill. A linear dependency of the losses between IR7 and IR8 is assumed, and the factor is applied to the BLM readings at IR8 $P_{8,\mathrm{H(V)},\mathrm{loss}}$ from the loss maps fill. After normalising to the BLM readings at IR8 during the reference physics fill $P_{8,\mathrm{ref}}$, this gives the relative losses $P_{8,\mathrm{H(V)},\mathrm{rel}}$ at IR8:

$$P_{8,\mathrm{H(V)},\mathrm{rel}} = \frac{P_{7,\mathrm{ref}}}{P_{7,\mathrm{H(V)},\mathrm{loss}}} \frac{P_{8,\mathrm{H(V)},\mathrm{loss}}}{P_{8,\mathrm{ref}}}. \tag{8.2}$$

The following reference fills are considered to get an idea of the impact on physics operations: fill 9846, 9848 and 9849 with an average number of visible proton-proton interactions of $\mu_{\mathrm{vis}} = 4.21$, $\mu_{\mathrm{vis}} = 4.39$, and $\mu_{\mathrm{vis}} = 4.36$, respectively. These are the first three fills of July, within half a month of the loss map fill, so the detector and machine are in a comparable state.

Table 8.6: The BLM readings in $\mathrm{Gy\,s^{-1}}$ for the B1H and B1V scenario, as well as the average of the reference runs mean and max values, at IR7 and IR8.

IR	Collimator	B1H	B1V	Ref mean	Ref max
IR7	TCPV	3.91×10^{-4}	1.28×10^{-2}	$(2.7 \pm 1.5) \times 10^{-4}$	$(1.2 \pm 0.7) \times 10^{-2}$
IR7	TCPH	2.77×10^{-2}	4.06×10^{-2}	$(2.9 \pm 0.8) \times 10^{-3}$	$(6.0 \pm 1.8) \times 10^{-2}$
IR7	TCPS	3.63×10^{-2}	3.16×10^{-2}	$(3.6 \pm 1.1) \times 10^{-3}$	$(5.7 \pm 1.3) \times 10^{-2}$
IR7	ALL	6.44×10^{-2}	8.50×10^{-2}	$(6.7 \pm 1.4) \times 10^{-3}$	$(1.3 \pm 0.3) \times 10^{-1}$
IR8	TCTH	5.90×10^{-6}	1.65×10^{-6}	$(3.45 \pm 0.06) \times 10^{-4}$	$(3.90 \pm 0.06) \times 10^{-4}$
IR8	TCTV	1.54×10^{-6}	9.53×10^{-7}	$(1.56 \pm 0.02) \times 10^{-3}$	$(1.75 \pm 0.03) \times 10^{-3}$
IR8	ALL	7.44×10^{-6}	2.60×10^{-6}	$(1.91 \pm 0.02) \times 10^{-3}$	$(2.14 \pm 0.03) \times 10^{-3}$

Table 8.7: The mean and maximum relative impact of the losses at IR8 on nominal physics operations for horizontal (H) and vertical (V) losses.

H/V	mean	max
H	$(4.1 \pm 0.8) \times 10^{-4}$	$(7.0 \pm 1.3) \times 10^{-3}$
V	$(1.1 \pm 0.3) \times 10^{-4}$	$(1.9 \pm 0.4) \times 10^{-3}$

The losses are not constant or Gaussian distributed; they degrade over time and fluctuate. Therefore, the mean and maximum of the distributions are calculated from the loss readings during the stable-beam period of the fill. To make the values at IR8 independent of μ_{vis}, they are divided by the average value of μ_{vis} during the fill and multiplied by the design value of five. This allows combining the fills. The error is given purely as the deviation between the different fills, since the standard error is not meaningful for non-Gaussian distributions.

The values for the loss readings for the B1H and B1V scenario, as well as for the reference fills, are given in Table 8.6. The total readings from all collimators at the interaction regions are also provided. The combined value of all collimators is used to calculate the relative loss impact by applying equation 8.2 for both the B1H and B1V scenarios. An alternative would be to introduce a decomposition matrix to disentangle horizontal from vertical losses; however, the chosen method is deemed sufficient. The results are given in Table 8.7.

The time integral and normalisation of the mean value cancel out; therefore, it also provides the total relative impact on physics during the fill. The losses have a maximum relative contribution of a few permille to the nominal physics collision. The total contribution is another order of magnitude smaller at $\mathcal{O}(10^{-4})$.

8.3 Conclusion for beam losses

For the first time, the contribution of beam losses to MIB is measured at the LHCb detector, using data collected during dedicated loss map fills. The confirmed losses impacting the detector are the beam 1 losses from betatron cleaning at IR7. No losses can be observed for beam 2 or from momentum cleaning.

The relative maximum loss signal in the detector, compared to the semi-nominal reference signal, shows that the muon stations are the most sensitive to losses, followed by the HCAL, the ECAL and then the SciFi Tracker. Further, the horizontal loss signal is consistently observed to be a factor of two larger than the vertical loss signal, indicating a lower cleaning performance of the horizontal collimators at IR7. The integrated number of hits normalised to the number of protons lost at IR7 is in the order below $\mathcal{O}(10^{-5})$. Around one hit is seen in the experiment per 10 000 protons lost at IR7. This is an order of magnitude smaller than the fraction of protons simulated to reach the TCTs of the experiment.

An asymmetry is observed towards negative x in the LHCb coordinate system, towards the centre of the LHC ring. The reconstructed tracks show straight tracks parallel to the beam. This information is relevant to potentially separate loss tracks from nominal collision tracks. Comparing the contribution of the losses to physics operations, the relative maximum contribution at a given time is a few per mille. In contrast, the total contribution over the timespan of an average fill in July 2024 is smaller $\mathcal{O}(10^{-4})$.

To check whether the asymmetry is caused by the magnet, for future studies, the polarity of the magnet can be inverted. The asymmetry can then be quantified and cross-validated against the observed asymmetry in detector ageing. This analysis studies losses from proton-proton collisions, as this is the particle species injected into the LHC for most of the year. Other losses from ion-ion collision have not been considered, and it is of interest to validate the results with additional loss map fills in ions. Moreover, the statistical sample for this analysis is constrained by the 2024 loss map data, which is limited to one pilot bunch per scenario. This can be increased by including data from additional loss map fills.

9 Final conclusion

Beam monitoring is a vital part of the Large Hadron Collider (LHC) and its experiments. If the beam gets out of control, crucial parts of the machine and the detector can be damaged, resulting in exceeding costs in time, physics data and money. Therefore, dedicated detector systems are constructed to prevent critical damage to the machine components. One of the safety systems is the Beam Conditions Monitor (BCM) of the Large Hadron Collider beauty (LHCb) experiment. Its task is to protect sensitive detector parts by initiating a beam abort if it detects critical beam conditions for the experiment.

The BCM was upgraded with a new readout system: the Machine Interface Beam Abort Decision (MIBAD) board. It is responsible for making the dump decision based on the BCM sensor information. The BCM is controlled by an Experiment Control System (ECS), which is adapted to the MIBAD system. Complete monitoring and operational control are provided, together with continuous health monitoring of the system. A post-mortem system is implemented to store the most recent sensor data in transient storage. It saves the data to disk after a beam dump to provide post-mortem analysis of the dump cause. The BCM ECS is fully integrated into the global LHCb ECS and is successfully operating since the commissioning of the MIBAD in June 2023.

Further upgrades are planned to enhance beam-monitoring capabilities at the LHCb experiment. To start development of the future ECS already, the ECS of the Timepix4 telescope is fully implemented. The telescope is considered a baseline and prototype detector for the beam monitoring upgrade at LHCb. Monitoring and control are provided for the power supplies and the motion stages in dedicated components. The environmental conditions are monitored, implemented as a plug-and-play system for easy setup with the hardware. The ECS components are designed to be versatile and transferable to related detector systems, saving development time and resources.

To study the beam behaviour at the LHCb experiment, the effect of beam losses on the detector is investigated. An asymmetry is observed in the horizontal plane of the detector, with a bias towards the centre of the LHC ring. A small, but not negligible, impact on physics operations is observed. These effects will be analysed further in the future. The effects of ion losses, compared to the proton-proton losses considered in this work, are an interesting aspect to be considered.

As a final remark, the LHCb experiment is in an excellent state regarding safety. The BCM upgrade proved its worth and reliability through multiple justified beam aborts over the last two years. Further upgrades with even more enhanced beam-monitoring capabilities are underway, and the Timepix4 telescope ECS has been used extensively by the Timepix4 group to operate the system at various test beams. The loss studies reveal an interesting asymmetry that warrants further research, and helps better understand the impact of losses on the LHCb experiment.

Bibliography

[1] H. Hilscher. *Elementare Teilchenphysik*. Facetten. Vieweg+Teubner Verlag, 2013. ISBN: 9783322850034.

[2] Q. R. Ahmad et al. 'Direct Evidence for Neutrino Flavor Transformation from Neutral-Current Interactions in the Sudbury Neutrino Observatory'. In: *Phys. Rev. Lett.* 89 (2002), p. 011301. DOI: 10.1103/PhysRevLett.89.011301.

[3] A. Knochel. 'Offene Fragen'. In: *Neustart des LHC: neue Physik. essentials* (2016). Springer Spektrum, Wiesbaden.

[4] O. S. Brüning et al. *LHC Design Report*. CERN Yellow Reports: Monographs. Geneva: CERN, 2004. DOI: 10.5170/CERN-2004-003-V-1.

[5] G. Aad et al. 'Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC'. In: *Physics Letters B* 716.1 (2012), pp. 1–29. ISSN: 0370-2693. DOI: 10.1016/j.physletb.2012.08.020.

[6] S. Chatrchyan et al. 'Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC'. In: *Physics Letters B* 716.1 (2012), pp. 30–61. ISSN: 0370-2693. DOI: 10.1016/j.physletb.2012.08.021.

[7] G. Aad et al. 'The ATLAS Experiment at the CERN Large Hadron Collider'. In: *JINST* 3 (2008), S08003. DOI: 10.1088/1748-0221/3/08/S08003.

[8] K. Aamodt et al. 'The ALICE experiment at the CERN LHC'. In: *JINST* 3 (2008), S08002. DOI: 10.1088/1748-0221/3/08/S08002.

[9] S. Chatrchyan et al. 'The CMS Experiment at the CERN LHC'. In: *JINST* 3 (2008), S08004. DOI: 10.1088/1748-0221/3/08/S08004.

[10] A. Augusto Alves Jr. et al. 'The LHCb Detector at the LHC'. In: *JINST* 3 (2008), S08005. DOI: 10.1088/1748-0221/3/08/S08005.

[11] G. Arduini et al. 'LHC Upgrades in preparation of Run 3'. In: *Journal of Instrumentation* 19.05 (2024), P05061. DOI: 10.1088/1748-0221/19/05/P05061.

[12] C. Ilgner et al. *The Beam Conditions Monitor of the LHCb Experiment*. 2010. arXiv: 1001.2487 [physics.ins-det].

[13] M. S. Bieker. 'MIBAD - An FPGA-based readout system for the LHCb Beam Conditions Monitor'. Dissertation. TU Dortmund university, 2024. DOI: 10.17877/DE290R-24517.

[14] K. Akiba et al. 'The Timepix4 beam telescope'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1084 (2026), p. 171234. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2025.171234.

[15] K. Akiba et al. 'Reconstruction of charged tracks with Timepix4 ASICs'. In: *Journal of Instrumentation* 18.02 (2023), P02011. DOI: 10.1088/1748-0221/18/02/P02011.

[16] The LHCb collaboration. 'The LHCb Upgrade I'. In: *Journal of Instrumentation* 19.05 (2024), P05065. ISSN: 1748-0221. DOI: 10.1088/1748-0221/19/05/p05065.

[17] A. I. Drozhdin, N. V. Mokhov and S. I. Striganov. 'Beam Losses and Background Loads on Collider Detectors Due to Beam-Gas Interactions in the LHC'. In: *Particle Accelerator Conference (PAC 09)*. 2010, WE6PFP027. URL: https://cds.cern.ch/record/1370157.

[18] R. Bruce et al. 'Simulations and measurements of beam loss patterns at the CERN Large Hadron Collider'. In: *Phys. Rev. ST Accel. Beams* 17 (8 2014), p. 081004. DOI: 10.1103/PhysRevSTAB.17.081004.

[19] D. Rolf. 'Simulation of the Beam Conditions Monitor for the Run III upgrade of the LHCb detector and development of a control system for the Timepix4 telescope'. MA thesis. TU Dortmund university, 2021. DOI: 10.17877/DE290R-24483.

[20] S. Lee. *Accelerator Physics (Fourth Edition)*. World Scientific Publishing Company, 2018. ISBN: 978-981-327-468-6, 978-981-327-467-9. DOI: 10.1142/11111.

[21] R. Talman. 'Scaling behavior of circular colliders dominated by synchrotron radiation'. In: *International Journal of Modern Physics A* 30.23 (2015), p. 1544003. DOI: 10.1142/S0217751X15440030.

[22] S. W. Ellingson. *Electromagnetics, Vol. 2*. Blacksburg, VA: Virginia Tech Publishing, 2020. DOI: 10.21061/electromagnetics-vol-2.

[23] *The Cambridge Companion to Newton*. 2nd ed. Cambridge Companions to Philosophy. Cambridge University Press, 2016.

[24] S. P. Thompson. *The Electromagnet and Electromagnetic Mechanism*. Cambridge Library Collection - Technology. Cambridge University Press, 2011.

[25] B.V. Svistunov, E.S. Babaev and N.V. Prokof'ev. *Superfluid States of Matter*. 2015. DOI: 10.1201/b18346.

[26] T. P. Wangler. *RF Acceleration in Linacs*. John Wiley & Sons, Ltd, 2008. ISBN: 9783527623426. DOI: 10.1002/9783527623426.

[27] S. Baird. *ACCELERATORS FOR PEDESTRIANS*. AB-Note-2007-014 OP. 2007. URL: https://cds.cern.ch/record/1017689/files/ab-note-2007-014.pdf.

[28] F. Marcastel. *CERN's Accelerator Complex*. Poster-2013-377. 2013. URL: http://cds.cern.ch/record/1621586.

[29] C. Wiesner. *Summary of LHC Beam Operation 2024*. 215th TE-TM meeting. 2025. URL: https://indico.cern.ch/event/1488564/#22-summary-of-lhc-beam-operati.

[30] R. Alemany, M. Lamont and S. Page. *LHC MODES*. LHC-OP-ES-0005 rev 1.0. 2007. URL: https://lhc-commissioning.web.cern.ch/systems/data-exchange/doc/LHC-OP-ES-0005-10-00.pdf.

[31] P. Hermes et al. 'Measured and simulated heavy-ion beam loss patterns at the CERN Large Hadron Collider'. In: *Nuclear Instruments and Methods in Physics Research Section A Accelerators Spectrometers Detectors and Associated Equipment* 819 (2016), pp. 73–83. DOI: 10.1016/j.nima.2016.02.050.

[32] J. Bernhard et al. 'The New CERN Low-energy Facilities for Neutrino Detector Tests'. In: *Journal of Physics: Conference Series* 1468 (2020). 012127. DOI: 10.1088/1742-6596/1468/1/012127.

[33] J. Bernhard. *The Test Beam Facilities of CERN*. ICHEP 2024. 2024. URL: https://indico.cern.ch/event/1291157/contributions/5888461/.

[34] R. Assmann et al. 'Equilibrium beam distribution and halo in the LHC'. In: *8th European Particle Accelerator Conference (EPAC)*. 2002, pp. 1326–1328. URL: https://cds.cern.ch/record/569470.

[35] R. Bruce et al. 'Collimation-induced experimental background studies at the CERN Large Hadron Collider'. In: *Phys. Rev. Accel. Beams* 22 (2 2019), p. 021004. DOI: 10.1103/PhysRevAccelBeams.22.021004.

[36] E. B. Holzer et al. 'Beam Loss Monitoring for LHC Machine Protection'. In: *Physics Procedia* 37 (2012), pp. 2055–2062. ISSN: 1875-3892. DOI: 10.1016/j.phpro.2012.04.110.

[37] F. Meot. 'Perturbation of the periodic dispersion under beam crossing optics in LHC'. In: *Conf. Proc. C* 970512 (1997), pp. 121–123. URL: https://cds.cern.ch/record/337370.

[38] L. H. A. Leunissen. 'Influence of vertical dispersion and crossing angle on the performance of the LHC'. In: *IEEE Particle Accelerator Conference (PAC 99)*. 1999. URL: https://cds.cern.ch/record/386689.

[39] B. J. Holzer. 'Introduction to Transverse Beam Dynamics'. In: *CERN Accelerator School: Course on Superconductivity for Accelerators*. 2013, pp. 27–45. DOI: 10.5170/CERN-2013-007.27. arXiv: 1404.0923 [physics.acc-ph].

[40] S. Redaelli et al. 'Chapter 5: Collimation system'. In: *CERN Yellow Rep. Monogr.* vol. 10 (2020), pp. 87–114. DOI: 10.23731/CYRM-2020-0010.87.

[41] S. Redaelli. 'Beam Cleaning and Collimation Systems'. In: *CERN Accelerator School course: Advanced Accelerator Physics*. 2025. arXiv: 2511.03511 [physics.acc-ph].

[42] N. Fuster-Martínez et al. 'Simulations of heavy-ion halo collimation at the CERN Large Hadron Collider: benchmark with measurements and cleaning performance evaluation'. In: *Phys. Rev. Accel. Beams* 23 (2020). DOI: 10.1103/PhysRevAccelBeams.23.111002. arXiv: 2008.03234.

[43] J. Wenninger. 'Machine Protection and Operation for LHC'. In: *CERN Yellow Reports* vol. 2 (2016). DOI: 10.5170/CERN-2016-002.377.

[44] B. Todd et al. *THE ARCHITECTURE, DESIGN AND REALISATION OF THE LHC BEAM INTERLOCK SYSTEM*. 2005. URL: https://api.semanticscholar.org/CorpusID:110431179.

[45] B. Todd. 'A Beam Interlock System for CERN High Energy Accelerators'. Dissertation. Brunel University, 2006. URL: https://www.researchgate.net/publication/41217456_A_Beam_Interlock_System_for_CERN_High_Energy_Accelerators.

[46] W. Baldini et al. 'Overview of LHCb alignment'. In: *1st LHC Detection Alignment Workshop*. 2006, pp. 197–222. DOI: 10.5170/CERN-2007-004.197.

[47] E. Buchanan. 'The LHCb Vertex Locator (VELO) Pixel Detector Upgrade'. In: *Journal of Instrumentation* 12.01 (2017), pp. C01013–C01013. DOI: 10.1088/1748-0221/12/01/c01013.

[48] The LHCb Collaboration. *LHCb VELO Upgrade Technical Design Report*. Tech. rep. 2013. URL: https://cds.cern.ch/record/1624070.

[49] The LHCb Collaboration. *LHCb Tracker Upgrade Technical Design Report*. Tech. rep. CERN-LHCC-2014-001. LHCB-TDR-015. 2014. URL: http://cds.cern.ch/record/1647400.

[50] P. Hopchev et al. *SciFi: A large Scintillating Fibre Tracker for LHCb*. 2017. arXiv: 1710.08325 [physics.ins-det]. URL: https://arxiv.org/abs/1710.08325.

[51] K. Hennessy. 'LHCb VELO upgrade'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 845 (2017), pp. 97–100. ISSN: 0168-9002. DOI: 10.1016/j.nima.2016.04.077.

[52] The LHCb Collaboration. *LHCb PID Upgrade Technical Design Report*. Tech. rep. 2013. DOI: 10.17181/CERN.BD4G.I3VP.

[53] The LHCb Collaboration. *LHCb RICH Technical Design Report*. Tech. rep. Geneva, 2000. URL: https://cds.cern.ch/record/494263.

[54] The LHCb Collaboration. *LHCb calorimeters Technical Design Report*. Tech. rep. Geneva, 2000. URL: https://cds.cern.ch/record/494264.

[55] The LHCb Collaboration. *LHCb muon system Technical Design Report*. Tech. rep. Geneva, 2001. URL: https://cds.cern.ch/record/504326.

[56] The LHCb Collaboration. *LHCb PLUME: Probe for LUminosity MEasurement*. Tech. rep. Geneva: CERN, 2021. DOI: 10.17181/CERN.WLU0.M37F.

[57] B. Jost et al. 'The LHCb front-end electronics and data acquisition system'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 453.1 (2000). Proc. 7th Int. Conf on Instrumentation for colliding Beam Physics, pp. 377–381. ISSN: 0168-9002. DOI: 10.1016/S0168-9002(00)00660-4.

[58] The LHCb Collaboration. *LHCb Trigger and Online Upgrade Technical Design Report*. Tech. rep. CERN-LHCC-2014-016. LHCB-TDR-016. 2014. DOI: 10.17181/CERN.5F5X.FDJM.

[59] B. Franek and C. Gaspar. 'SMI++ object oriented framework for designing and implementing distributed control systems'. In: *IEEE Transactions on Nuclear Science* 45.4 (1998), pp. 1946–1950. DOI: 10.1109/23.710969.

[60] A. Ciccotelli et al. *Energy deposition studies in the LHCb insertion region from the validation to a step into the Hilumi challenge*. 2022. arXiv: 2205.15645 [physics.acc-ph]. URL: https://arxiv.org/abs/2205.15645.

[61] BCM group. Internal communication. 2025.

[62] E Effinger et al. 'The LHC beam loss monitoring system's data acquisition card'. In: *Proceedings of the Twelfth Workshop on Electronics for LHC and Future Experiments* (2007), pp. 108–112. DOI: 10.5170/CERN-2007-001.108.

[63] S. Schleich. 'FPGA based Data Acquisition and Beam Dump Decision System for the LHCb Beam Conditions Monitor'. Diploma thesis. TU Dortmund, 2008.

[64] CAEN SpA. *Power Supply*. URL: https://www.caen.it/products/dt1415et/ (visited on 01/01/2026).

[65] K. Akiba et al. 'LHCb VELO Timepix3 telescope'. In: *Journal of Instrumentation* 14.05 (2019), P05026–P05026. ISSN: 1748-0221. DOI: 10.1088/1748-0221/14/05/p05026.

[66] ETM professional control GmbH. *SIMATIC WinCC Open Architecture Portal*. URL: https://www.winccoa.com/ (visited on 01/01/2026).

[67] S Schmeling. 'Common tools for large experiment controls'. In: *IEEE Trans. Nucl. Sci.* 53 (2006), pp. 970–973. DOI: 10.1109/TNS.2006.873706.

[68] P. Burkimsher et al. *WinCC OA and the JCOP Framework*. Presentations as part of the Joint WinCCOA JCOP Framework Course. 2020. URL: https://edms.cern.ch/ui/#!master/navigator/project?P:1601072490:1601072490:subDocs.

[69] O. Holme et al. 'The JCOP framework'. In: *Conf. Proc. C051010:WE* 2 (2005). URL: https://www.researchgate.net/publication/44216554_The_JCOP_framework.

[70] OPC Foundation. *Unified Architecture*. URL: https://opcfoundation.org/about/opc-technologies/opc-ua/ (visited on 01/01/2026).

[71] C. Gaspar, M. Dönszelmann and Ph. Charpentier. 'DIM, a portable, light weight package for information publishing, data transfer and inter-process communication'. In: *Computer Physics Communications* 140.1 (2001). CHEP2000, pp. 102–109. ISSN: 0010-4655. DOI: https://doi.org/10.1016/S0010-4655(01)00260-0.

[72] Python Software Foundation. *Python*. URL: https://www.python.org/ (visited on 01/01/2026).

[73] R. Stoica and CERN LHCb Online Group. *PyDIM documentation*. Python package. URL: http://lhcbdoc.web.cern.ch/lhcbdoc/pydim/guide/index.html (visited on 01/01/2026).

[74] G. van Rossum and Python Software Foundation. *asyncio - Asynchronous I/O*. Python package. URL: https://docs.python.org/3/library/asyncio.html (visited on 01/01/2026).

[75] V. Sajip and Python Software Foundation. *logging - Logging facility for Python*. Python package. URL: https://docs.python.org/3/library/logging.html (visited on 01/01/2026).

[76] Python Software Foundation. *configparser - Configuration file parser*. Python package. URL: https://docs.python.org/3/library/configparser.html (visited on 01/01/2026).

[77] Python Software Foundation. *argparse - Parser for command-line options, arguments and subcommands*. Python package. URL: https://docs.python.org/3/library/argparse.html (visited on 01/01/2026).

[78] G. Vouters et al. *LHCb Upgrade FE and BE data format*. Tech. rep. 2019. URL: https://edms.cern.ch/ui/file/2114571/1/frond_end_and_back_end_data_format_of_the_lhcb_upgrade.pdf.

[79] Keithley Instruments Inc. *Series 2400 SourceMeter*. URL: https://de.tek.com/keithley-source-measure-units/keithley-smu-2400-series-sourcemeter-manual/series-2400-sourcemeter (visited on 01/01/2026).

[80] CAEN SpA. *HiVolta (DT1415ET)*. URL: https://www.caen.it/products/dt1415et/ (visited on 01/01/2026).

[81] Rohde & Schwarz GmbH & Co. KG. *R&S HMP4000*. URL: https://www.rohde-schwarz.com/de/handbuch/hmp4000/ (visited on 01/01/2026).

[82] vddvss. *linux-gpib-packaging*. GitHub repository. URL: https://github.com/vddvss/linux-gpib-packaging (visited on 01/01/2026).

[83] GitHub Inc. *GitHub*. URL: https://github.com/ (visited on 01/01/2026).

[84] PyVISA Authors. *PyVISA: Control your instruments with Python*. Python package. URL: https://pyvisa.readthedocs.io/en/latest/ (visited on 01/01/2026).

[85] PyVISA Authors. *PyVISA-py: Pure Python backend for PyVISA*. Python package. URL: https://pyvisa.readthedocs.io/projects/pyvisa-py/en/latest/ (visited on 01/01/2026).

[86] Honeywell International Inc. *Honeywell Industrial Automation*. URL: https://automation.honeywell.com/us/en (visited on 01/01/2026).

[87] H. Boterenbrood and B. I. Hallgren. *The Development of Embedded Local Monitor Board (ELMB)*. 2003. DOI: 10.5170/CERN-2003-006.331.

[88] H. Boterenbrood et al. *The Embedded Local Monitor Board (ELMB) in the LHC Front-end I/O Control System*. 2001. DOI: 10.5170/CERN-2001-005.325.

[89] SYS TEC electronic AG. *SYS TEC electronic*. URL: https://www.systec-electronic.com/en/ (visited on 01/01/2026).

[90] Physik Instrumente (PI) SE & Co. KG. *Physik Instrumente (PI) – Solution for Precision Motion and Positioning*. URL: https://www.physikinstrumente.com/en/ (visited on 01/01/2026).

[91] Festo Inc. *Automation technology and technical education solutions*. URL: https://www.festo.com/ch/en/ (visited on 01/01/2026).

[92] Phytron GmbH. *Your partner for stepper motors*. URL: https://www.phytron.eu/ (visited on 01/01/2026).

[93] National Instruments Corp. *LabVIEW*. URL: https://www.ni.com/en/shop/labview.html (visited on 01/01/2026).

[94] R. Bruce, R. W. Assmann and S. Redaelli. 'Calculations of safe collimator settings and β^* at the CERN Large Hadron Collider'. In: *Phys. Rev. ST Accel. Beams* 18 (6 2015), p. 061001. DOI: 10.1103/PhysRevSTAB.18.061001.

[95] M. Lieng. 'Studies of the Machine Induced Background, simulations for the design of the Beam Condition Monitor and implementation of the Inclusive ϕ Trigger at the LHCb experiment at CERN'. Dissertation. TU Dortmund, 2011. DOI: 10.17877/DE290R-735.

[96] F. Van Der Veken and LBS group. Internal communication. 2024.

[97] CERN. *TIMBER*. URL: https://cern.service-now.com/service-portal?id=functional_element&name=TIMBER (visited on 01/01/2026).

Appendices

A Beam Conditions Monitor operator panels

In this appendix, additional panels of the Beam Conditions Monitor (BCM) control system are shown. First, the CAEN panels are given, including the overview panel and the panels for the individual channels. Second, the overview panel is given for the special cases of maintenance and override mode. Lastly, the panels displaying the currents are shown.

Overview panel special cases

The BCM can be set in maintenance mode. This is indicated on the overview panel by a flag as shown in Figure 1. The override mode of the BCM is visually indicated by a yellow grid, which shows that the system permits are not based on the actual hardware state. This is demonstrated in Figure 2.

Current panels

The panel tab of the overview panel, which provides an overview of the BCM currents, is shown in Figure 3. It displays the minimum and maximum RS2-SUM and RS32-SUM values, along with a random sample value from the last readout period. The detailed RS1, RS2 and RS32 values of each sensor, as well as the average flux within the last readout period of 1.3 s, are provided in separate panels as shown in Figure 4. The values are also provided in the form of a table as seen in Figure 5.

CAEN panels

Figure 6 shows the CAEN overview panel. A trend displays the current and voltage values for both BCM stations. Figure 7 shows the CAEN channel panel. The channel status and details are provided, and the voltages and currents are monitored. Figure 8 shows the CAEN channel detail panel. The detailed parameters read back from the CAEN are presented in two tables.

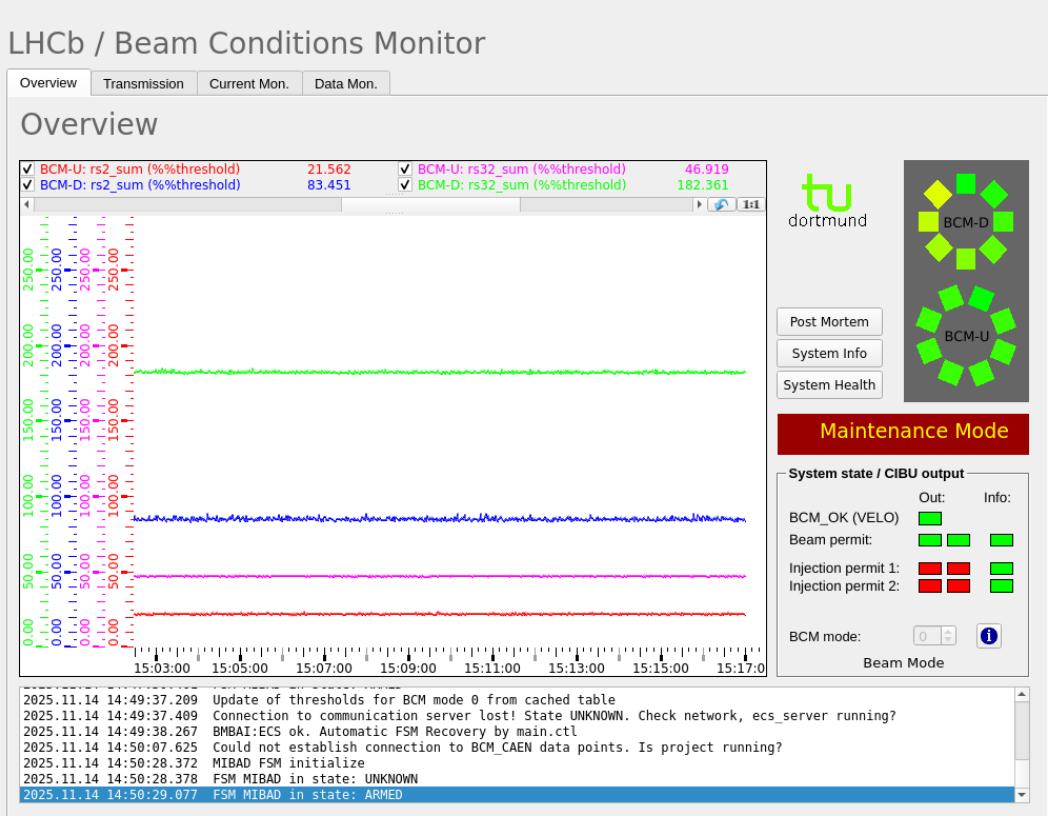


Figure 1: The BCM overview panel in maintenance mode.

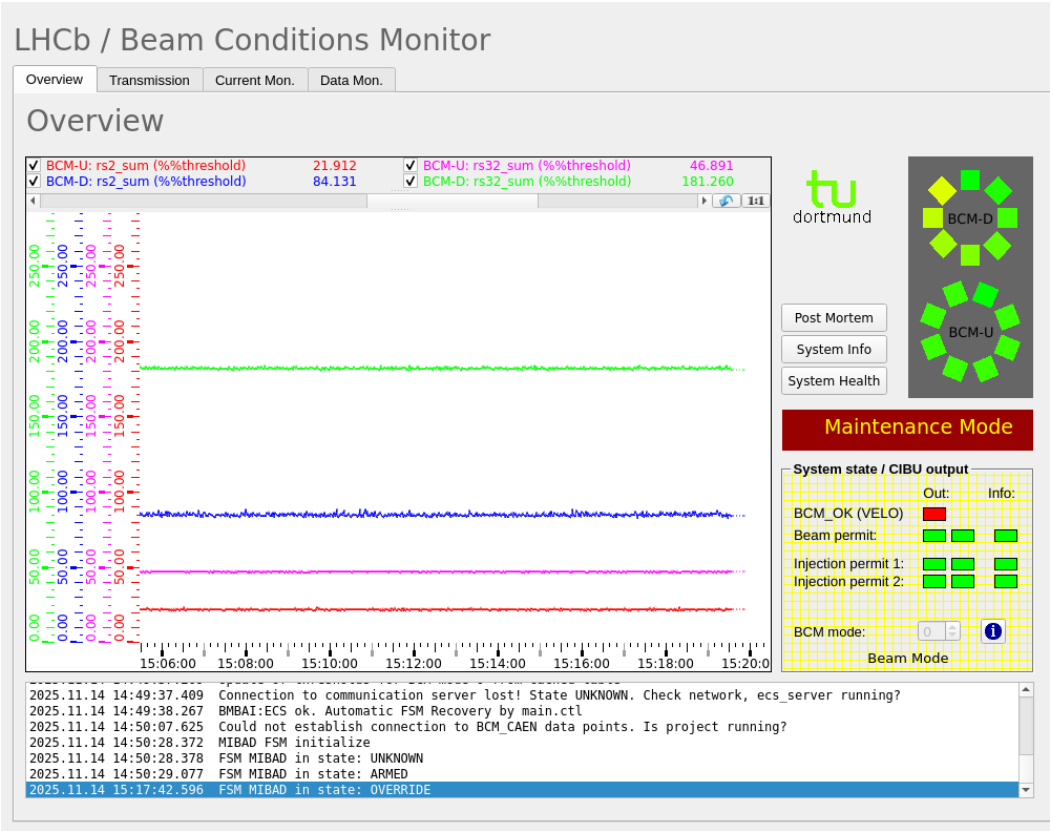


Figure 2: The BCM overview panel in override mode.

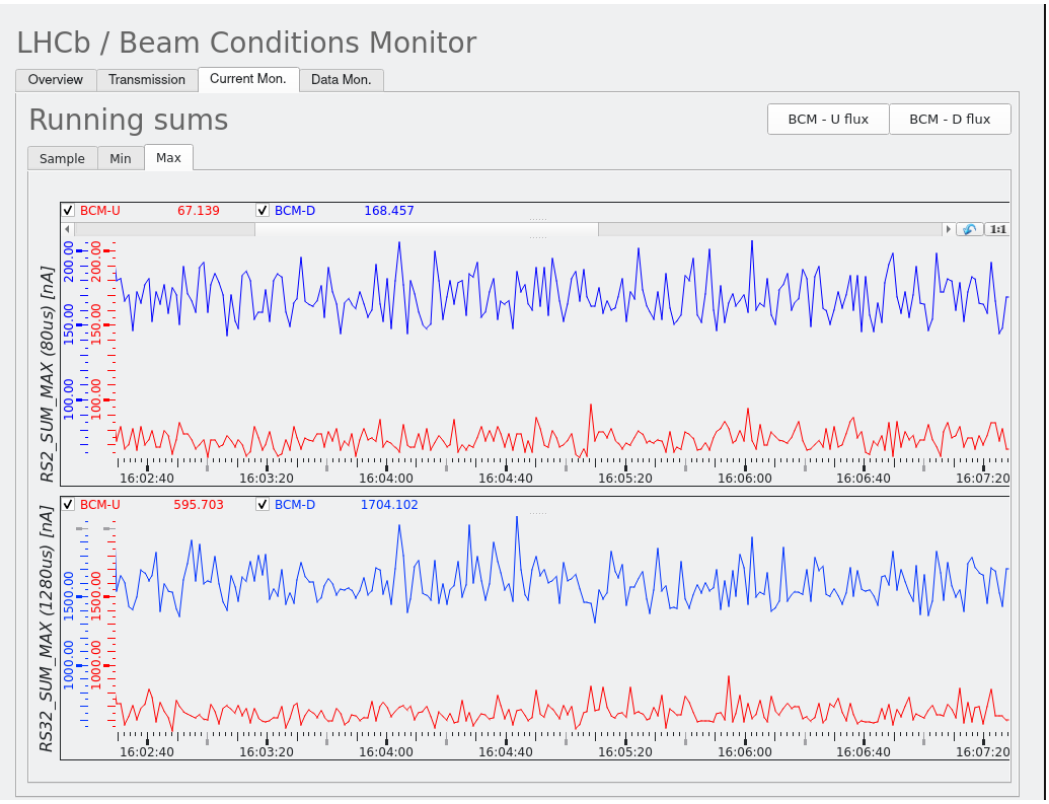
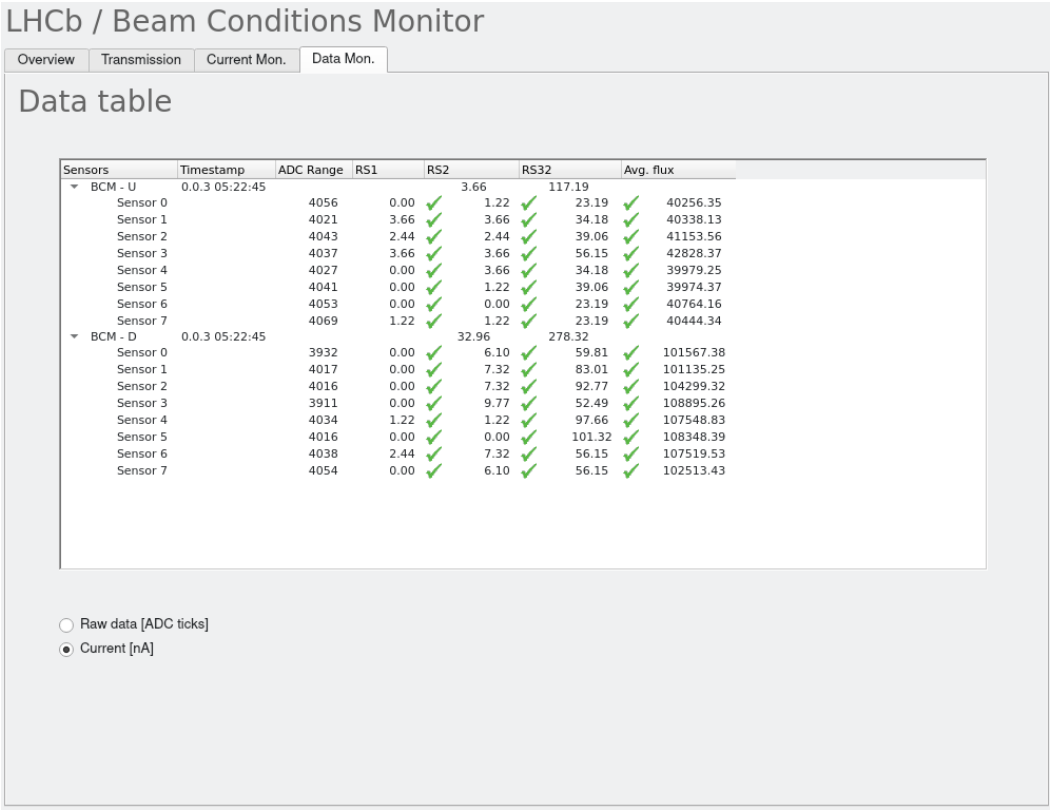


Figure 3: The current overview panel showing the RS2-SUM and RS32-SUM values for both BCM stations.

Figure 4: The detailed currents of the BCM-U station showing the values for RS1, RS2, RS32 and the average flux.



Sensors	Timestamp	ADC Range	RS1	RS2	RS32	Avg. flux
BCM - U	0.0.3 05:22:45			3.66	117.19	
Sensor 0		4056	0.00	1.22	23.19	40256.35
Sensor 1		4021	3.66	3.66	34.18	40338.13
Sensor 2		4043	2.44	2.44	39.06	41153.56
Sensor 3		4037	3.66	3.66	56.15	42828.37
Sensor 4		4027	0.00	3.66	34.18	39979.25
Sensor 5		4041	0.00	1.22	39.06	39974.37
Sensor 6		4053	0.00	0.00	23.19	40764.16
Sensor 7		4069	1.22	1.22	23.19	40444.34
BCM - D	0.0.3 05:22:45			32.96	278.32	
Sensor 0		3932	0.00	6.10	59.81	101567.38
Sensor 1		4017	0.00	7.32	83.01	101135.25
Sensor 2		4016	0.00	7.32	92.77	104299.32
Sensor 3		3911	0.00	9.77	52.49	108895.26
Sensor 4		4034	1.22	1.22	97.66	107548.83
Sensor 5		4016	0.00	0.00	101.32	108348.39
Sensor 6		4038	2.44	7.32	56.15	107519.53
Sensor 7		4054	0.00	6.10	56.15	102513.43

Figure 5: The values for RS1, RS2, RS32 and average flux, as well as the ADC range of both BCM stations, in a table format.

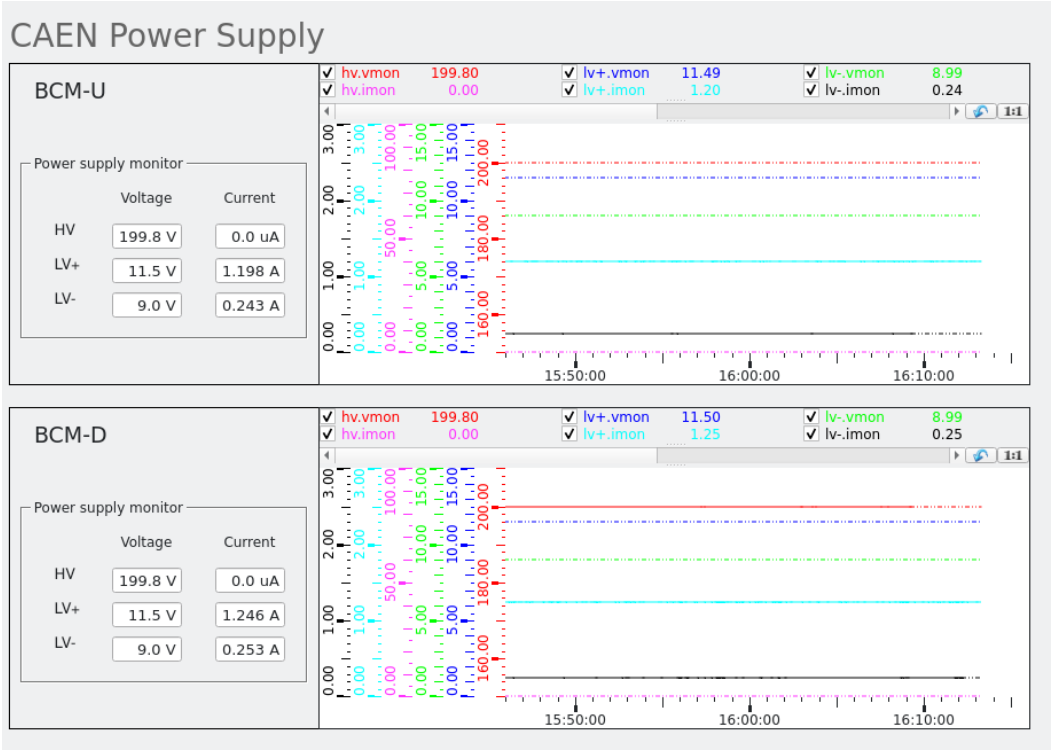


Figure 6: The CAEN overview panel.

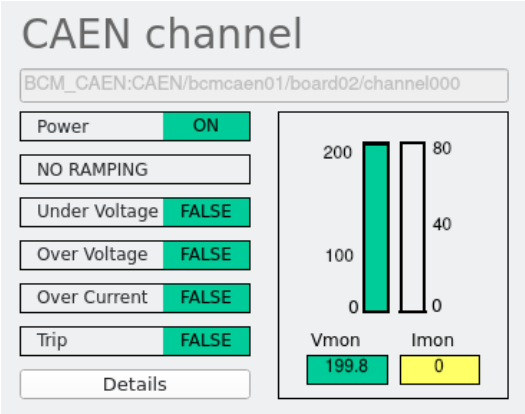


Figure 7: The CAEN channel panel for channel 0.

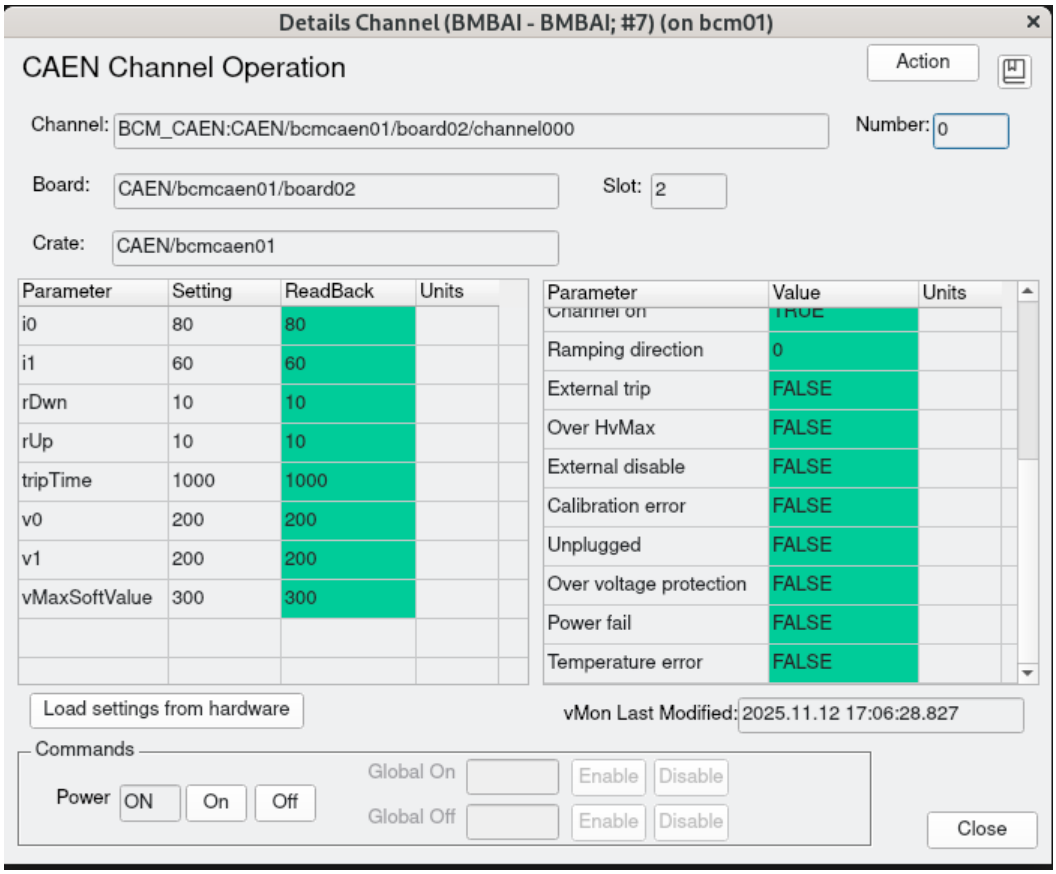


Figure 8: The CAEN channel detail panel for channel 0.

B Timepix4 telescope operator panels

In this appendix, additional panels of the Timepix4 telescope control system are shown. First, the panels provided for the power supplies are given. Second, the motion control panels implemented are presented.

Voltage panels

The voltage panels are created analogously for all power supplies. Figure 9 and Figure 10 show the main and the settings panel for the low voltage power supply, respectively. They are colour-coded in yellow for the low voltage. The main panel displays the read currents and voltages. They can be changed in the settings panel. The settings panel additionally displays the limits and alarms set on the channels. Figure 11 and Figure 12 show the same panels for the high-voltage power supplies. The high-voltage panels are colour-coded blue. Additionally, Figure 13 shows the expert settings used to set the limits and alarms on the channels.

Motion panels

Figure 14 shows the main panel of the motion control. The centre stage is selected. The other stages are controlled analogously. Information on all stages is provided to the right of the panel. A navigation panel can be extended, depicted in Figure 15. It shows the selected motion stage of the telescope.

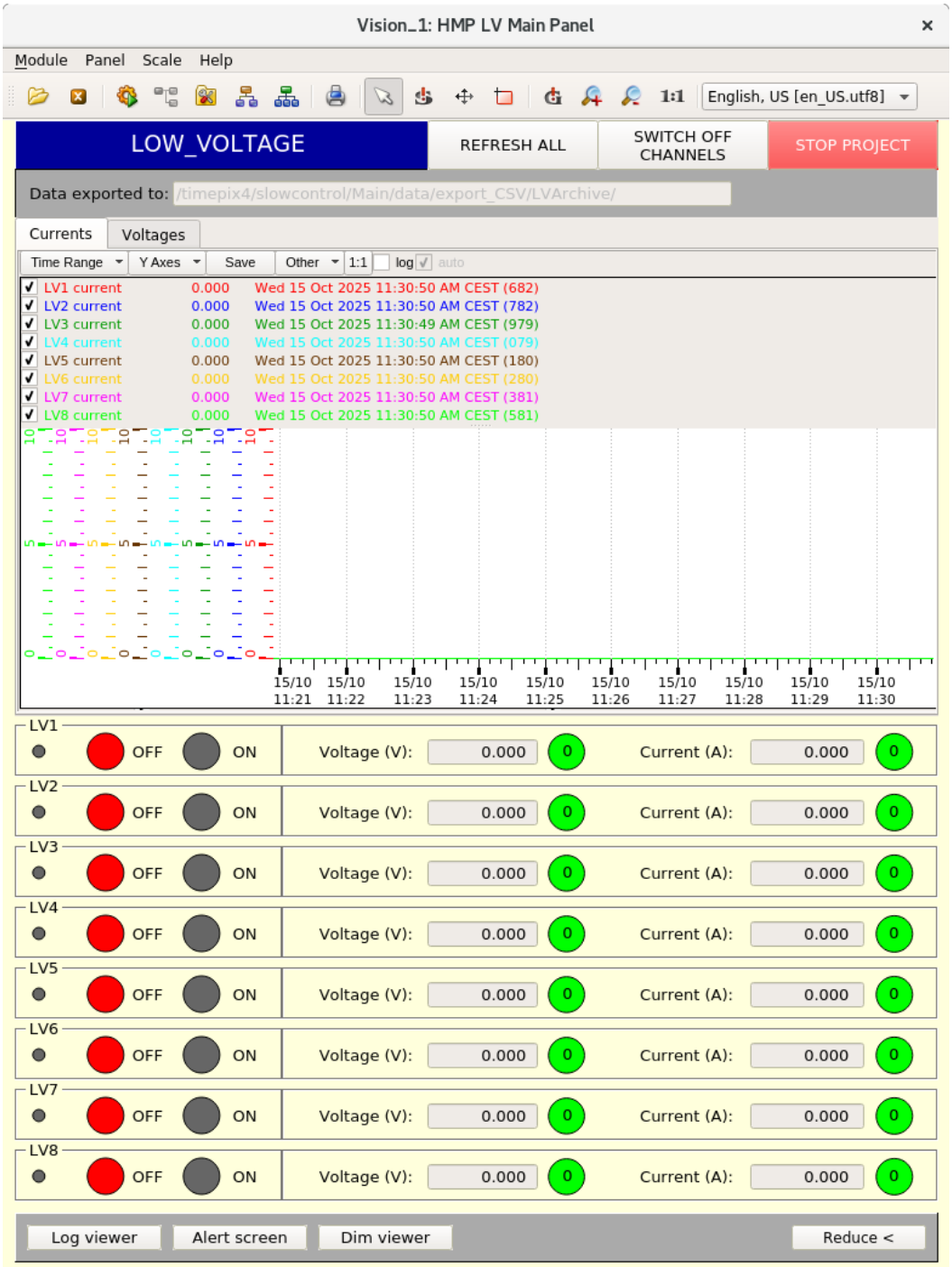


Figure 9: The main panel of the low-voltage power supply.

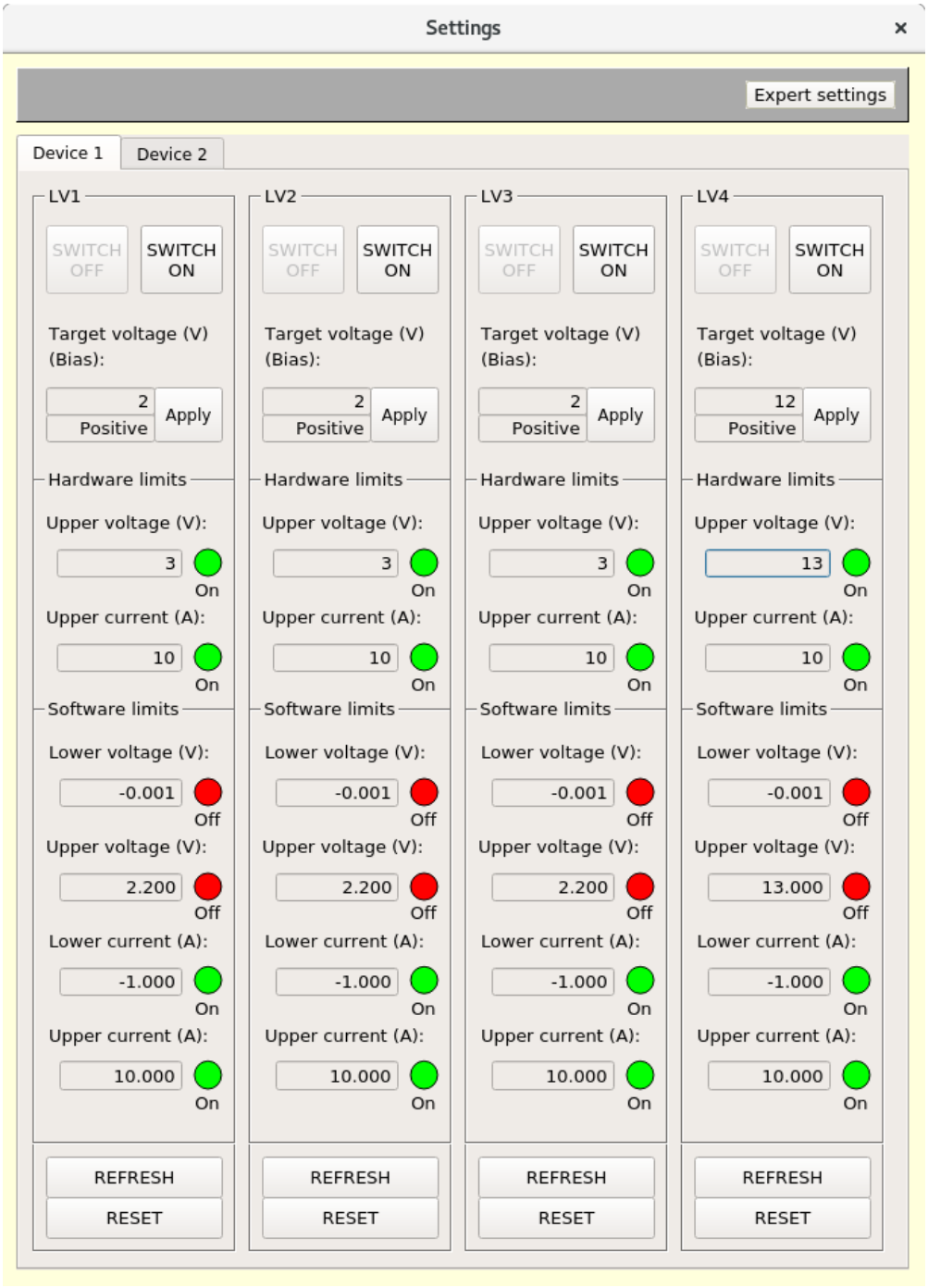


Figure 10: The settings panel of the low-voltage power supply.

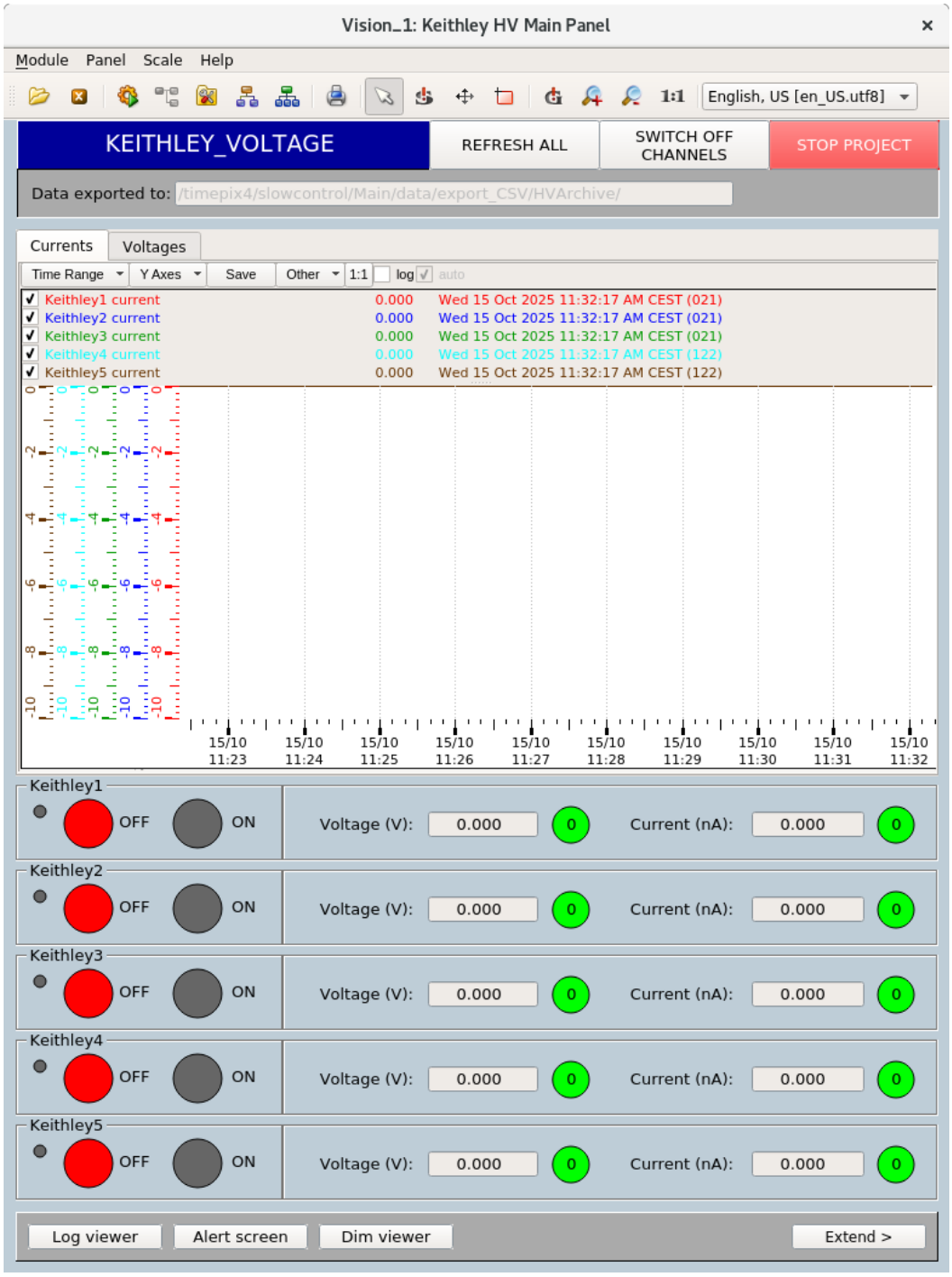


Figure 11: The main panel of the Keithley power supply.

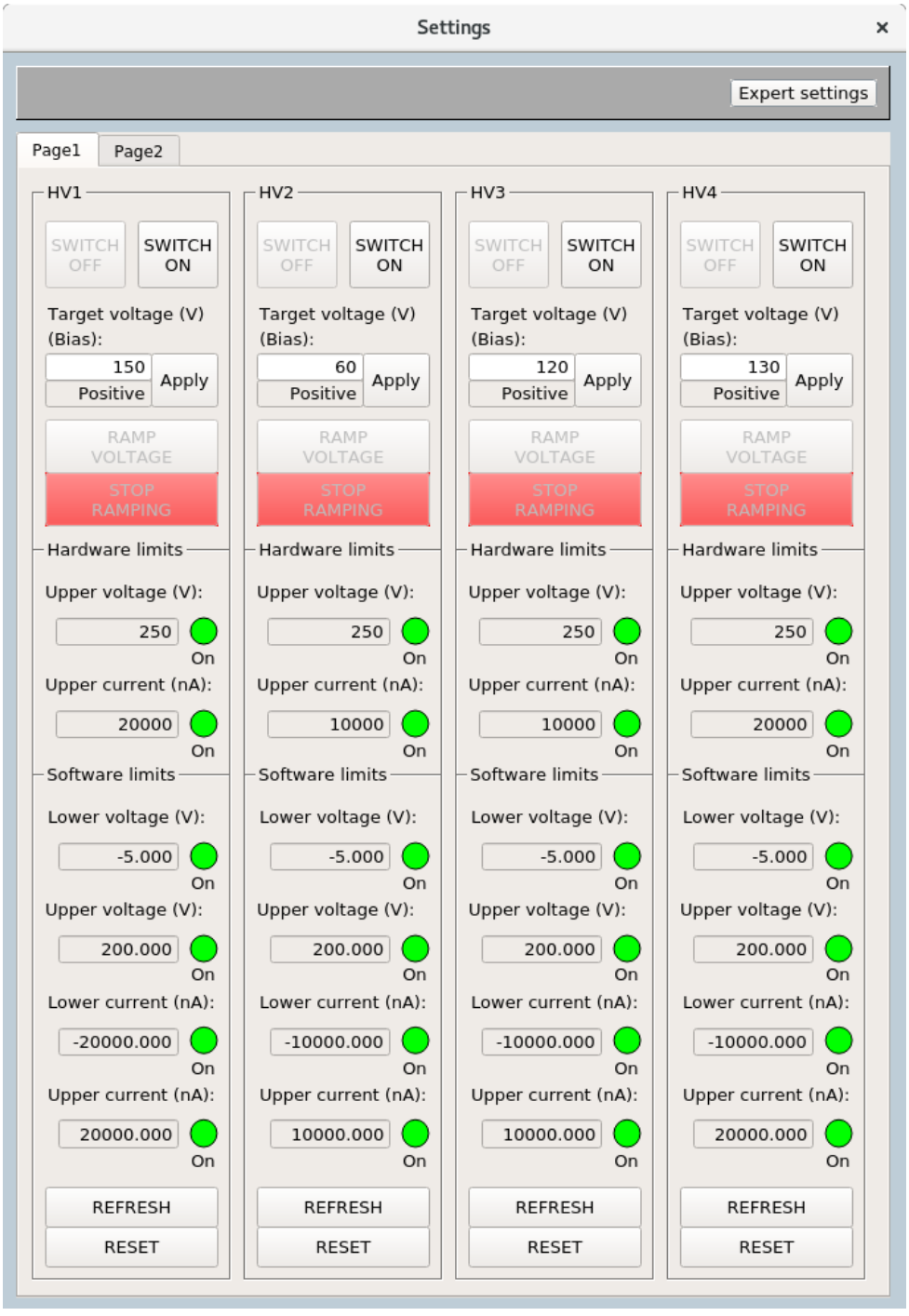


Figure 12: The settings panel of the high-voltage power supply.

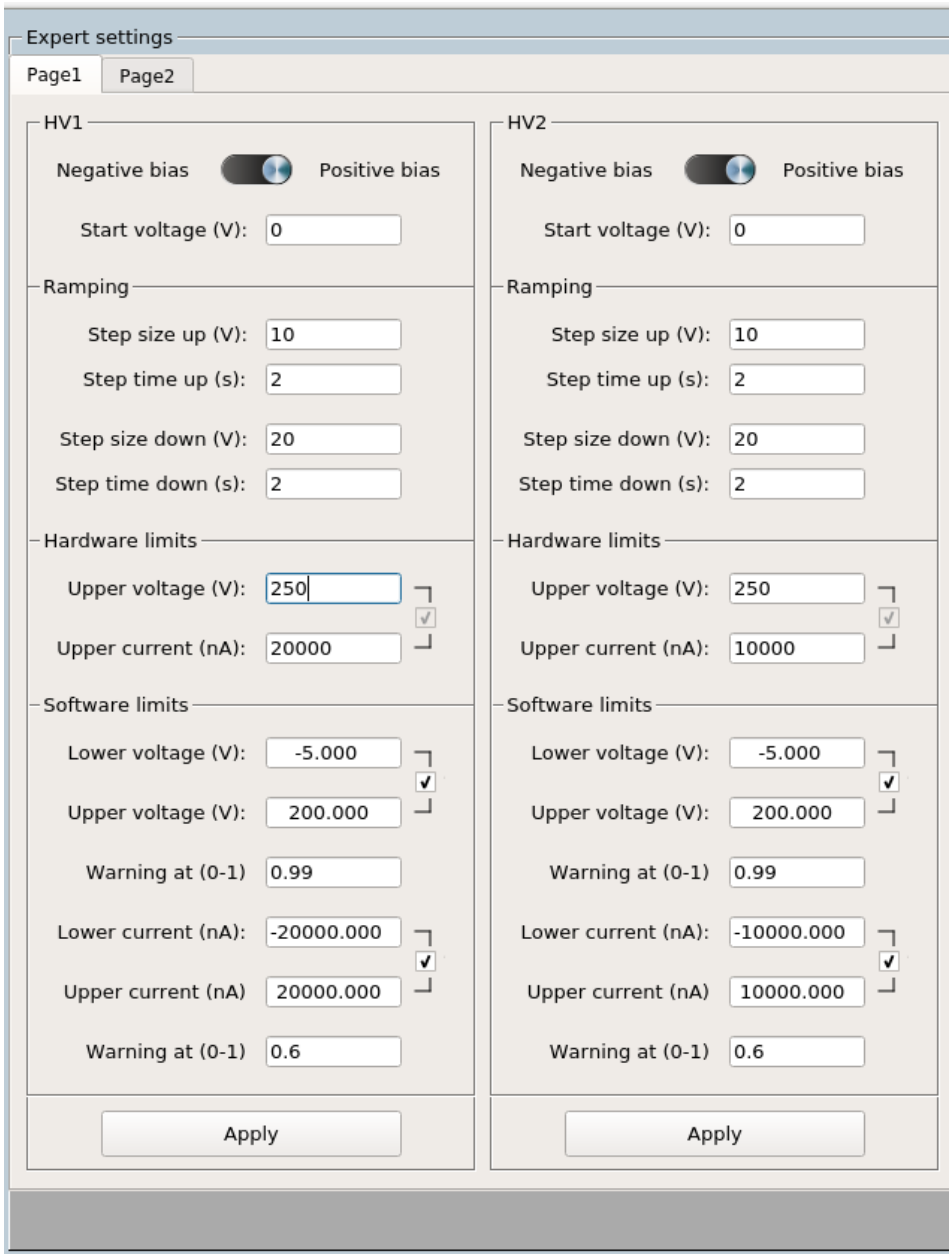


Figure 13: The expert settings panel of the high-voltage power supply.

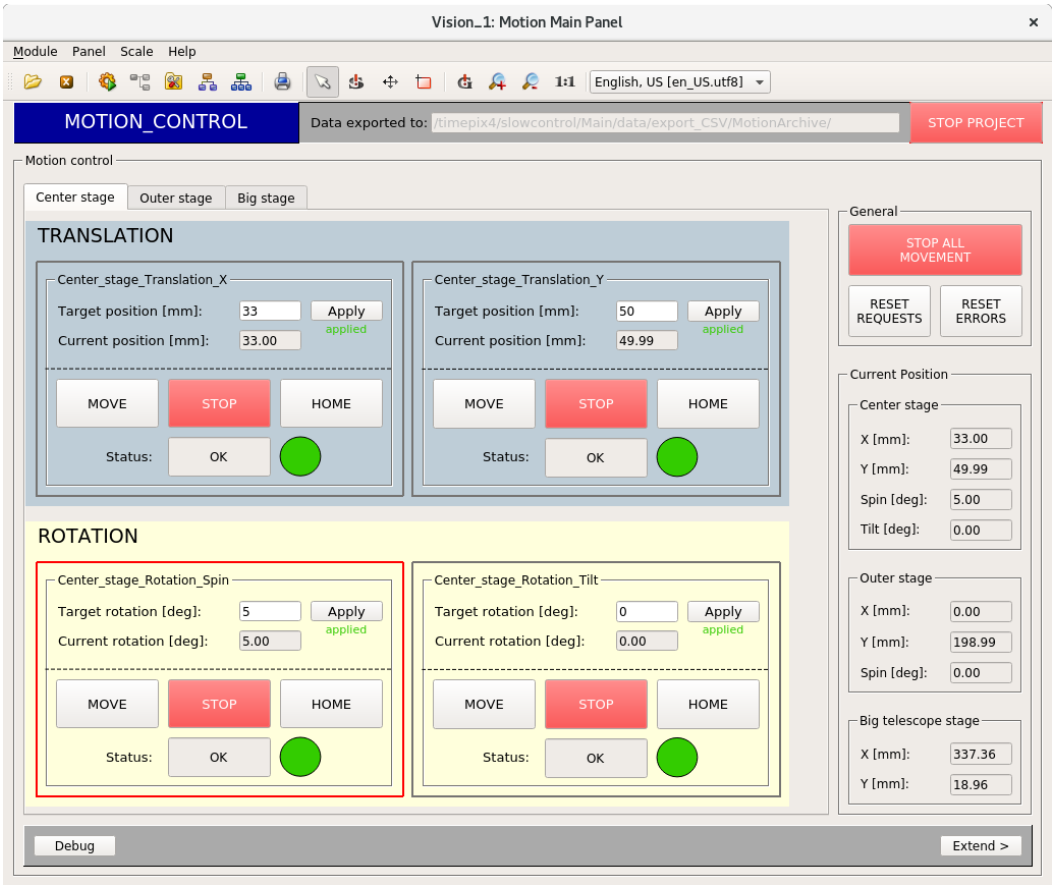


Figure 14: The main motion panel of the motion control.

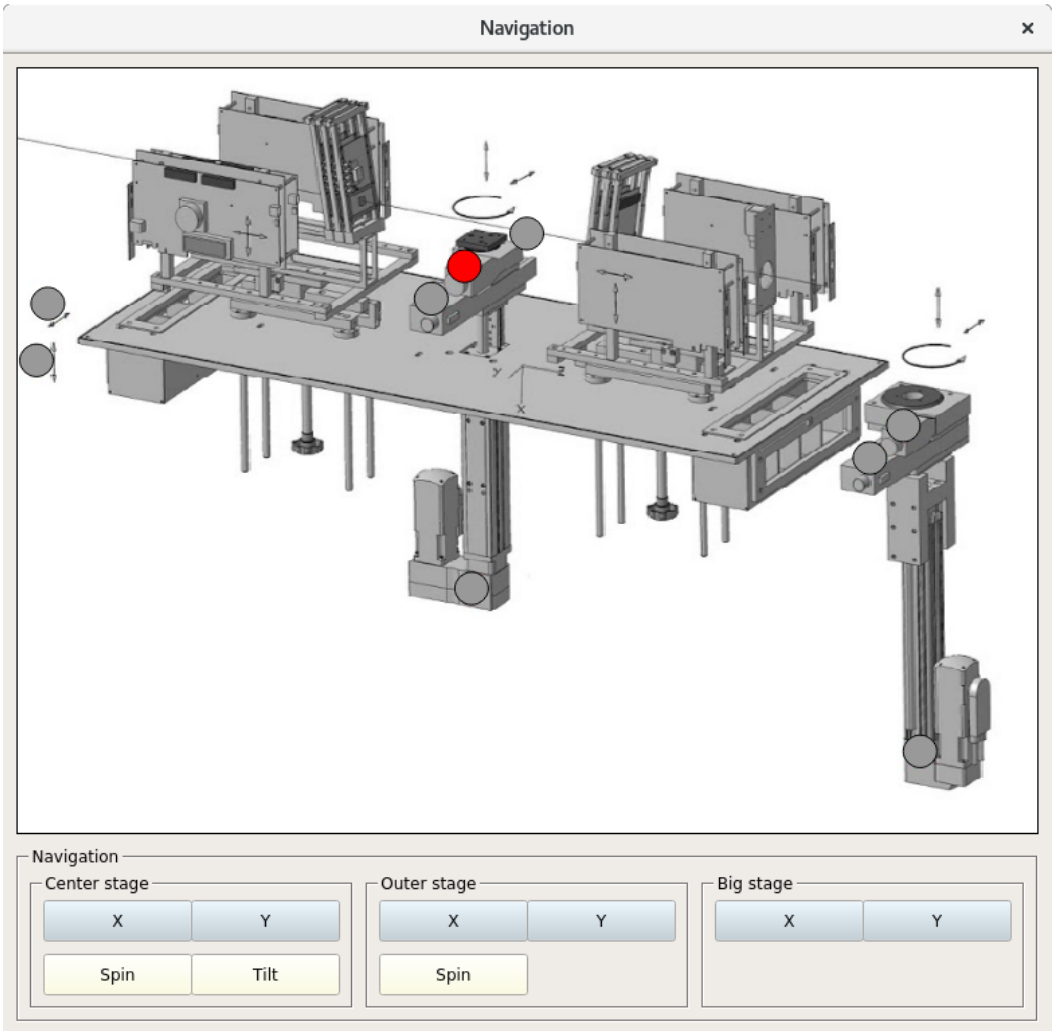


Figure 15: The navigation panel of the motion control.

C Additional loss study plots

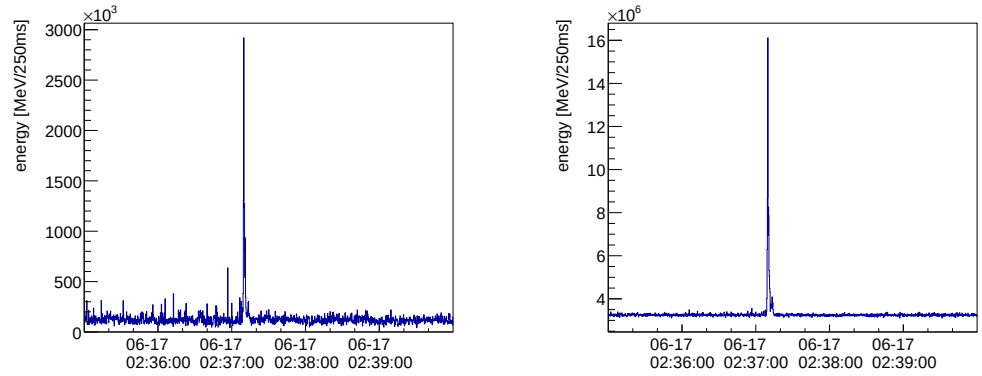
In this appendix, additional plots of the loss studies are shown. This includes plots for the time distribution and localisation of the losses for the B1H scenario, not shown in the main text.

Time distribution plots

Figure 16 shows the losses in the Large Hadron Collider beauty (LHCb) subdetectors over time for the B1H scenario. Figure 17 shows the signal region of the losses with subtracted pre-loss background. The behaviour is the same as for the B1V scenario covered in the main text.

Localisation plots

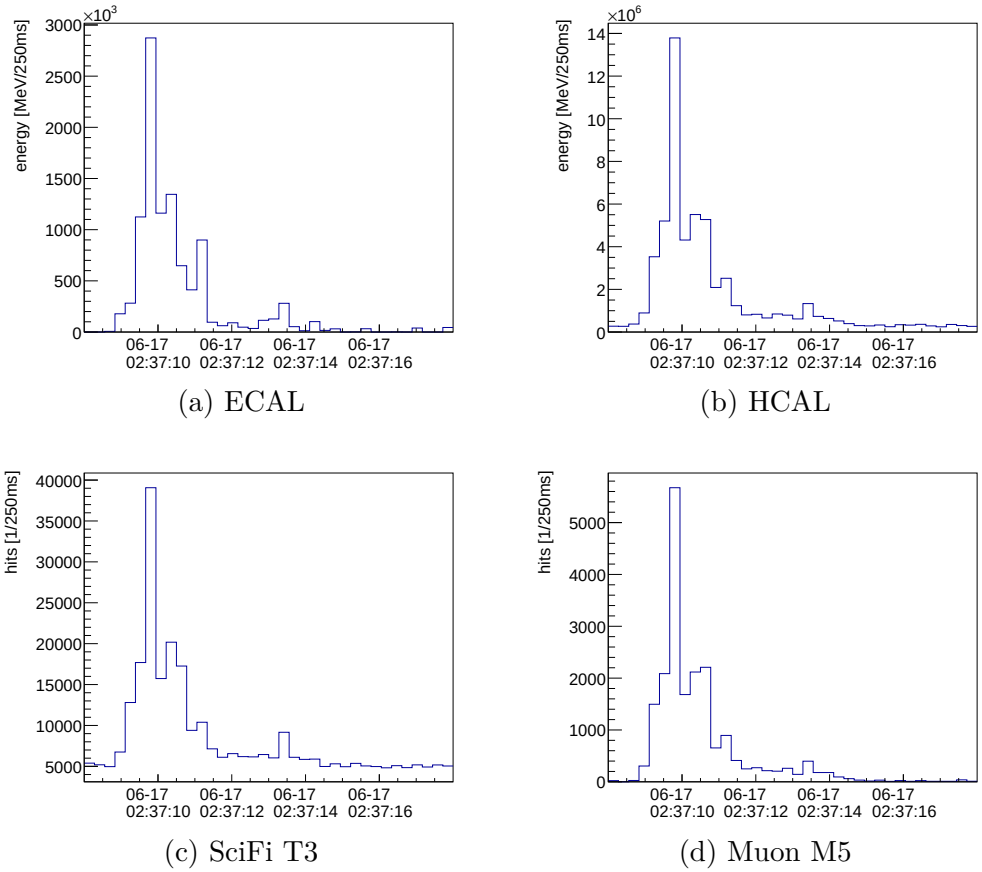
The localisation plots for the B1H scenario are given. Figure 18 shows the hits and energy deposition in the x-y plane of the electromagnetic calorimeter (ECAL) compared to the nominal reference signal and projected onto the x-axis. The same is shown for the hadron calorimeter (HCAL) in Figure 19. For the muon station, additionally, the hits in the z-x plane are given in Figure 20. The SciFi tracks and slopes of the tracks are given in Figure 21. The same is done for the matched muon tracks in Figure 22. The observations for all subdetectors are the same as for the B1V case covered in the main text.



(a) ECAL (b) HCAL

(c) SciFi T3 (d) Muon M5

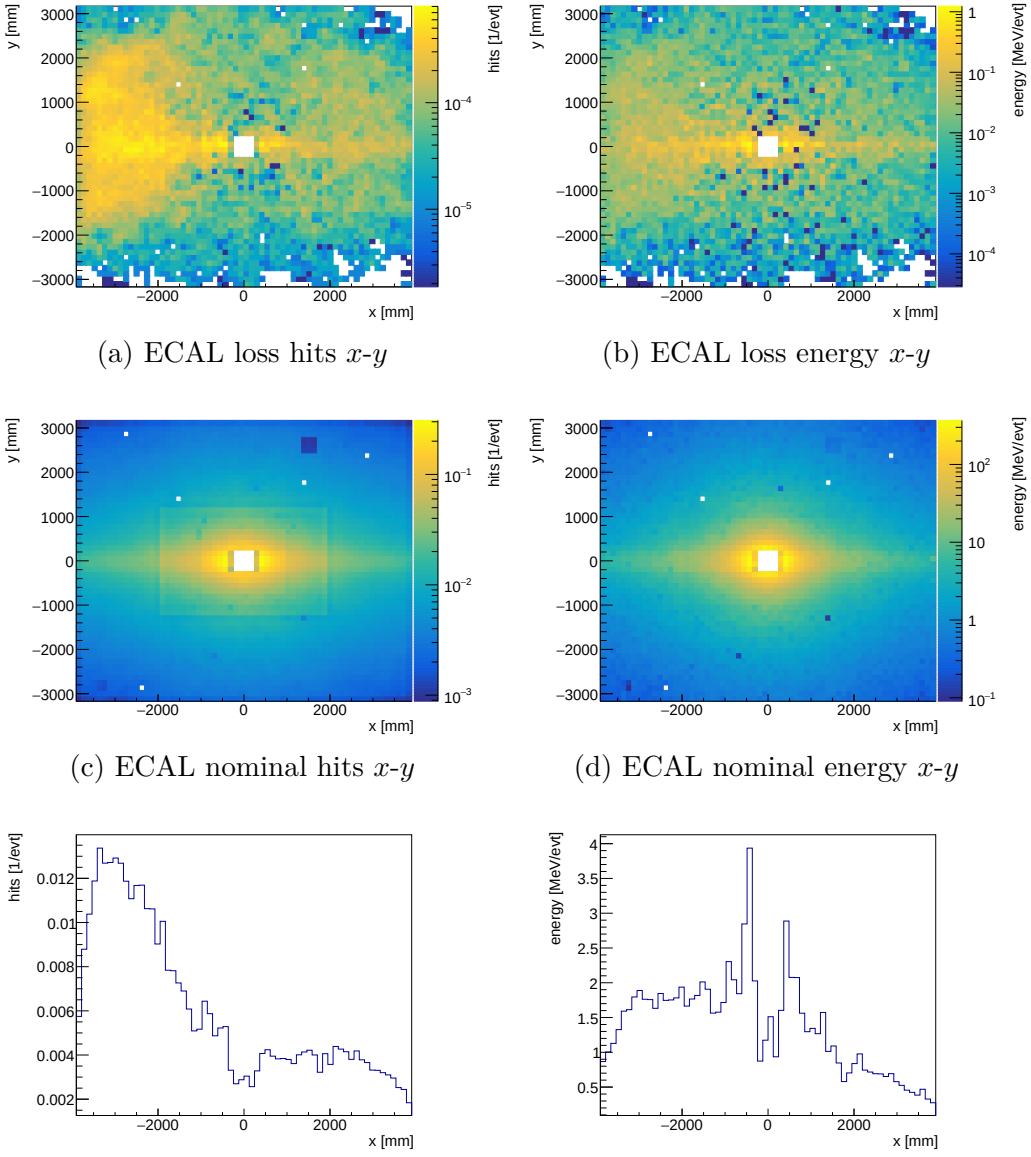
Figure 16: The beam1 losses vor B1H in the subdetectors. The SciFi Tracker station T3 and muon station M5 are exemplary for the other stations.



(a) ECAL (b) HCAL

(c) SciFi T3 (d) Muon M5

Figure 17: The signal region of the B1H losses for the subdetectors with subtracted average pre-loss background. The SciFi Tracker station T3 and muon station M5 are exemplary for the other stations.



(a) ECAL loss hits x-y

(b) ECAL loss energy x-y

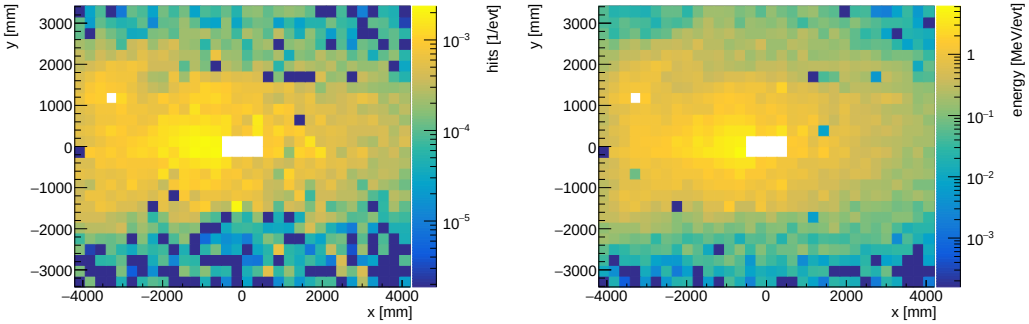
(c) ECAL nominal hits x-y

(d) ECAL nominal energy x-y

(e) ECAL loss hits x

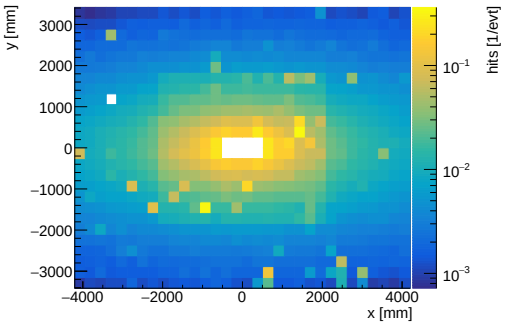
(f) ECAL loss energy x

Figure 18: The number of hits (left) and energy deposition (right) in the ECAL. The signal region of the B1H losses with subtracted pre-loss background projected onto the x-y plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis (bottom).

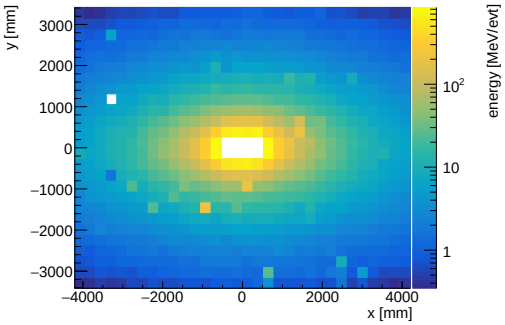


(a) HCAL loss hits x-y

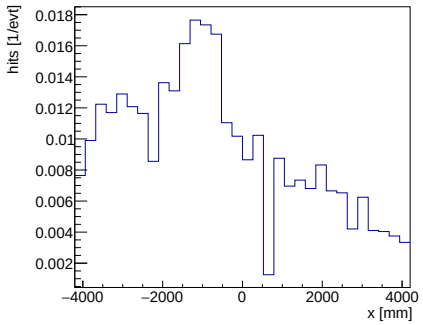
(b) HCAL loss energy x-y



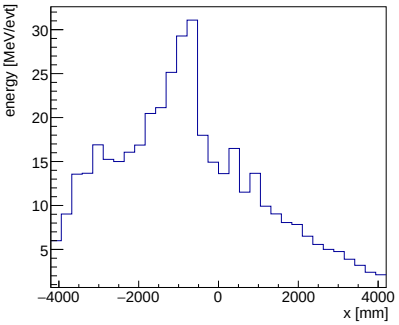
(c) HCAL nominal hits x-y



(d) HCAL nominal energy x-y

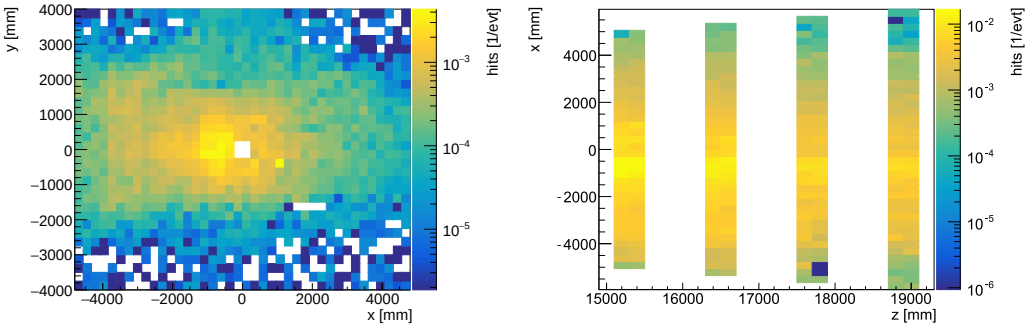


(e) HCAL loss hits x



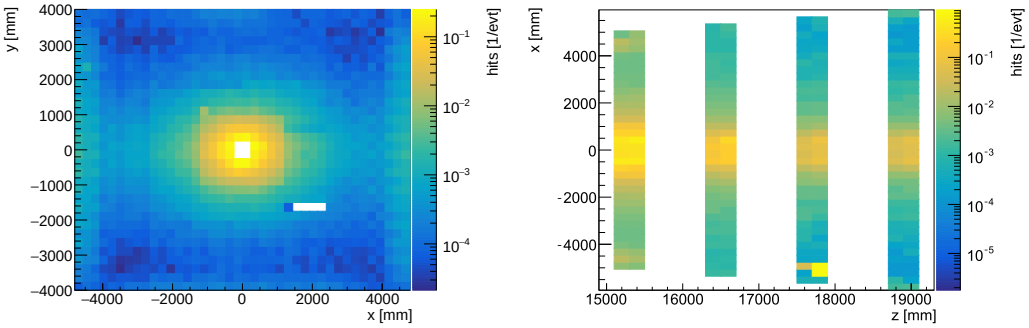
(f) HCAL loss energy x

Figure 19: The number of hits (left) and energy deposition (right) in the HCAL. The signal region of the B1H losses with subtracted pre-loss background projected onto the x-y plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis (bottom).



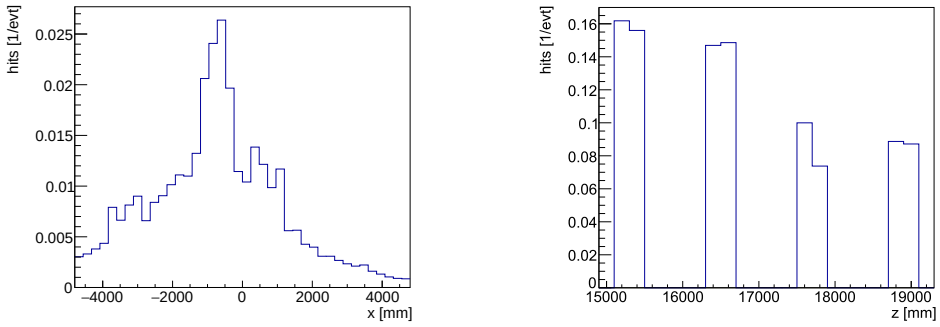
(a) Muon M2 loss hits x-y

(b) Muon M2 loss hits z-x



(c) Muon M2 nominal hits x-y

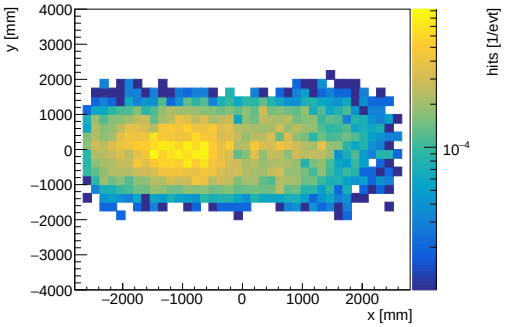
(d) Muon M2 nominal hits z-x



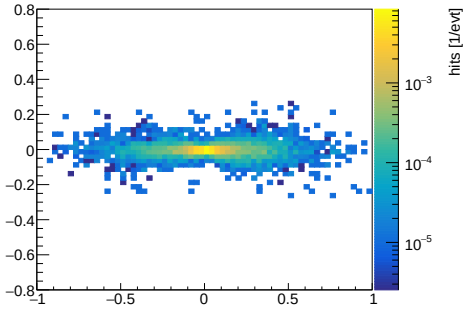
(e) Muon M2 loss hits x

(f) Muon M2 loss hits z

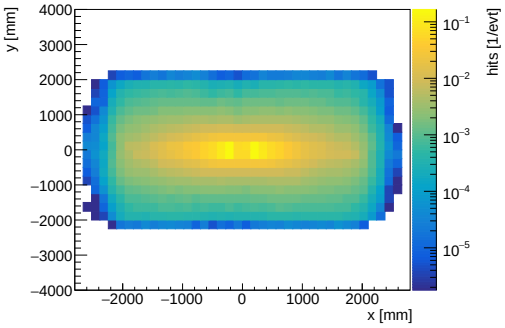
Figure 20: The number of hits in the muon station M2. The signal region of the B1H losses with subtracted pre-loss background projected onto the x-y plane and z-x plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis and z-axis (bottom).



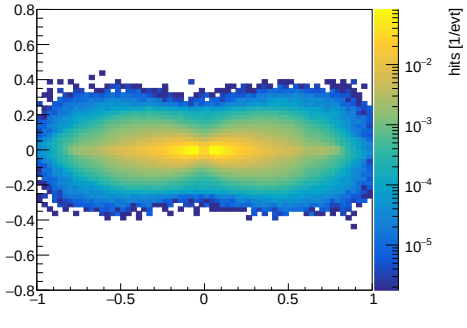
(a) SciFi T1 loss track hits x-y



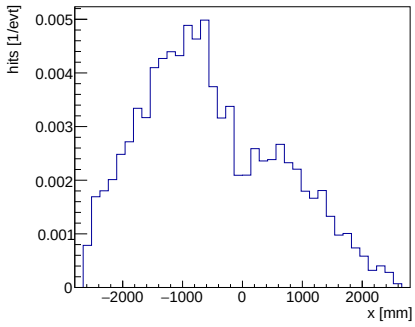
(b) SciFi T1 loss track hits x-y



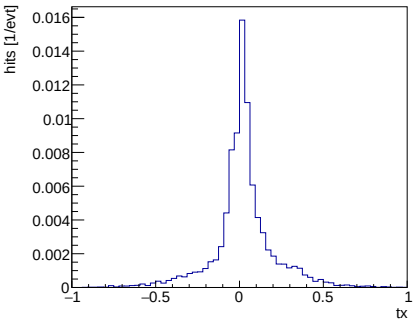
(c) SciFi T1 nominal track hits x-y



(d) SciFi T1 nominal track hits x-y

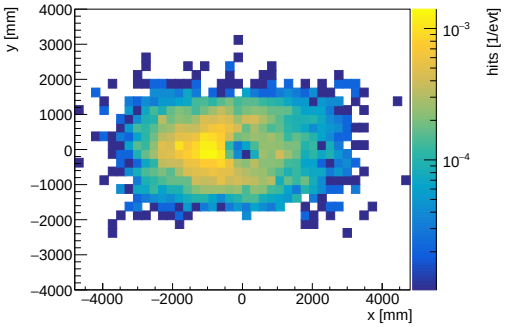


(e) SciFi T1 loss track hits x

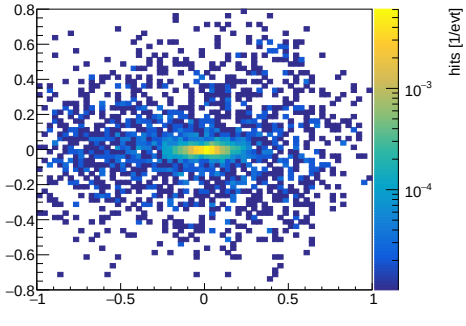


(f) SciFi T1 loss track slopes x

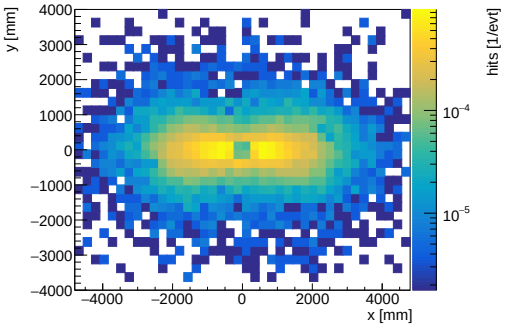
Figure 21: The reconstructed track hits (left) and track slopes (right) in the SciFi Tracker. The signal region of the B1H losses with subtracted pre-loss background projected onto the x-y plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis (bottom).



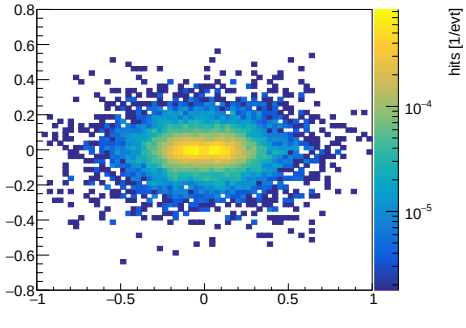
(a) Muon M2 loss track hits x-y



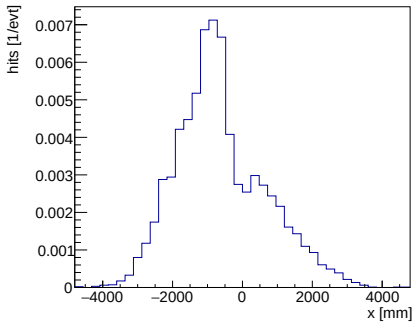
(b) Muon M2 loss track hits x-y



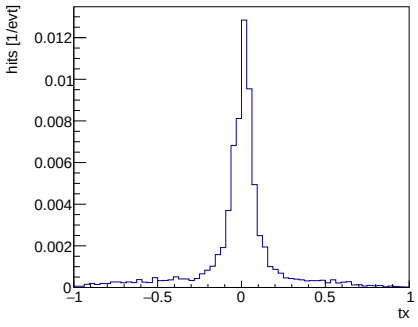
(c) Muon M2 nominal track hits x-y



(d) Muon M2 nominal track hits x-y



(e) Muon M2 loss track hits x



(f) Muon M2 loss track slopes x

Figure 22: The reconstructed track hits (left) and track slopes (right) in the muon stations. The signal region of the B1H losses with subtracted pre-loss background projected onto the x-y plane (top) compared to the signal from the semi-nominal reference bunch (middle). Additionally, the projection of the loss signal onto the x-axis (bottom).

Acknowledgement

I sincerely thank everyone who supported me in writing this dissertation. Without their help, it would not have been possible to deliver the work in this form. My first thanks goes to Prof. Dr. Johannes Albrecht, who welcomed me into his working group and allowed me to work on the presented topic. I also thank PD Dr. Dominik Elsässer as the second corrector of the dissertation.

My special thanks go to Elena and Federico for supervising and supporting me during my stay at CERN. In addition, I give many thanks to Clara for her expertise on the LHCb online system and to Martin, who worked with me on the BCM upgrade. I thank Dirk and Kai for their expertise on the hardware components, as well as the other members of the BCM and Timepix4 group for their contributions. I also thank everybody for proofreading my text.

Special thanks go to Biljana and Donata for being open and welcoming during a rather difficult time. My final thanks go to my family and my friend Yannik for their continued support during my studies. Their encouragement helped me a lot in writing up this dissertation.

This work has been sponsored by the Wolfgang Gentner Programme of the German Federal Ministry of Education and Research (grant no. 13E18CHA). Grammarly was used for spell checking, grammar and syntax.

Thank you!