

Development of the Control System for the Vacuum Operation and Validation of the MVD Prototype for the CBM Experiment

**Dissertation
zur Erlangung des Doktorgrades
der Naturwissenschaften**

**vorgelegt beim Fachbereich Physik
der Johann Wolfgang Goethe-Universität
in Frankfurt am Main**

von
Philipp Lothar Klaus
aus Starnberg

**Frankfurt 2021
(D 30)**

5. Oktober 2021

vom Fachbereich Physik der

Johann Wolfgang Goethe - Universität als Dissertation angenommen.

Dekan: Prof. Dr. Harald Appelshäuser

1. Gutachter: Prof. Dr. Joachim Stroth
2. Gutachter: PD Dr. Michael Deveaux

Datum der Disputation: 4. Oktober 2021

Mühsam nährt sich das Eichhörnchen.

The squirrel laboriously feeds itself.

Contents

Zusammenfassung (deutsch)	ix
List of Figures	xiii
1 Introduction	1
1.1 Matter under Extreme Conditions	2
1.2 The Need for High-Rate Experiments	3
1.3 Sensing the Physical World with Particle Detectors	4
1.4 The Importance of Stable Detector Operation	5
1.5 The Micro Vertex Detector of CBM and its Prototypes	6
1.6 Development of the Control System for the Prototype PRESTO	7
1.7 Validation of the Vacuum Operation of the Prototype PRESTO	7
2 The CBM Experiment: Matter at High Baryon Chemical Potentials	9
2.1 FAIR and the SIS100 Accelerator	9
2.2 The CBM Experiment	10
2.3 Tracking and Vertexing Detectors	13
2.4 Particle Identification Detectors	14
2.5 Event Characterising Detectors	14
2.6 Further Elements of CBM	14
3 The Micro Vertex Detector of CBM	17
3.1 Energy Loss of Charged Particles in the Detector	17
3.2 Radiation Hardness of Silicon Pixel Sensors	20
3.3 Material Budget Associated Detector Performance	21
3.3.1 Precision of Planar Detectors	21
3.3.2 Multiple Scattering	22
3.3.3 Radiation Length	22
3.4 Commissioning with Radioactive Sources and Cosmic Muons	22
3.4.1 Fe-55 Sources to Test Sensors	23
3.4.2 Muons from Cosmic Radiation	23
3.5 Desired Performance and Derived Requirements	25
3.6 The MIMOSIS Sensor Family	25
3.7 Mechanical Design	27

3.8	Evaluation of the Material Budget based on Simulation Geometries	29
4	The MVD Prototype PRESTO	33
4.1	Previous Prototypes of the MVD	33
4.2	The current Prototype: PRESTO	33
4.3	Sensor Used: MIMOSA-26	35
4.4	Prototype Assembly – Sensors and Carrier	36
4.5	Mechanical Setup of the Prototype	37
4.6	Major Devices for the Operation of the Prototype	39
4.7	Vacuum System for PRESTO	41
4.7.1	Vacuum Feedthrough for Flat Flexible Cables	41
4.8	Electronics and Data Acquisition	42
5	Development of the Control System for PRESTO	47
5.1	General Description of Detector Control Systems	48
5.2	System of Choice: EPICS	49
5.3	Evaluation of EPICS from Various Viewpoints	50
5.4	Hierarchy of DAQ and DCS	51
5.5	General Architecture of an EPICS Control System	51
5.6	Standard and Custom EPICS Components used for PRESTO	52
5.6.1	Standard Components used for PRESTO	52
5.6.2	Custom EPICS Components used for PRESTO	53
5.7	Packaging EPICS Control System Components with Container Virtualization	54
5.8	Testing the Control System with Simulated Hardware	57
5.9	Computer Architecture for DAQ and DCS	58
5.10	Implementation of I/O Controllers at the Example of the Cooling System	59
5.11	Graphical User Interfaces	60
5.11.1	Control System Studio and Phoebus	60
5.11.2	Dashboard Website	62
5.11.3	PyTrbNet IOC - A Bridge between DAQ and DCS	64
5.12	Channel Access Gateway	65
5.13	Architecture of the EPICS DCS for PRESTO	66
5.14	EPICS Archiver Solutions	67
5.15	Monitoring the Operational Stability	72
5.16	Commissioning of the Control System	73
6	Field Tests	75
6.1	Vacuum and Cooling Tests with PRESTO	75
6.1.1	Motivation	75
6.1.2	Setup	76

6.1.3	Implementation of the Measurements	76
6.1.4	Result of the Long-Term Vacuum Operation	77
6.1.5	Results: Vacuum Cycles	79
6.1.6	Result: Cooling Cycles	79
6.1.7	Discussion	81
6.2	Test: Full System Test with a ⁵⁵ Fe Source	84
6.2.1	Motivation	84
6.2.2	Experimental Setup	84
6.2.3	Measurement Result and Data Analysis	84
6.2.4	Interpretation	86
6.3	Measurement: Cosmic Rays	88
6.3.1	Motivation	88
6.3.2	Modelling of the Expected Cosmic Muons	89
6.3.3	Data Processing	93
6.3.4	Result	97
6.3.5	Interpretation	97
6.3.6	Implications for the Operation of the MVD	99
6.4	Opportunities for Improvement of the Control System and 24/7 Op- eration	106
7	Modernizing the EPICS Control System	109
7.1	Technological Enablers	110
7.1.1	Technology: VLAN (VXLAN)	110
7.1.2	Technology: Container Virtualization (e.g. Docker)	111
7.1.3	Technology: Kubernetes Cluster	113
7.1.4	Technology: Data Stream Management Systems	114
7.2	Modernization of EPICS Key Components	115
7.2.1	Core: EPICS Base	115
7.2.2	Core: synApps	116
7.2.3	Operator Interface: CS-Studio Phoebus	117
7.2.4	Operator Interface: Prometheus (r/o)	117
7.2.5	Gateways: Channel Access Gateway	118
7.2.6	Archivers: Archiver Appliance	119
7.2.7	Archivers: Cassandra PV Archiver	121
7.2.8	EPICS Operator Interface Implemented in the Browser	121
8	Summary	123
	Glossary	127
	Bibliography	135

Contents

Acknowledgement

155

Zusammenfassung (deutsch)

Die vorliegende Arbeit beschreibt die Entwicklung des Kontrollsystems für den Vakuumbetrieb und die Vakuumvalidierung des Prototypen des Mikro Vertex Detektors für das CBM Experiment. Als Doktorarbeit in der Physik enthält sie zusätzlich interdisziplinäre Aspekte aus der Elektrotechnik und der Informatik.

Der Mikro Vertex Detektor ist der vorderste in einer Reihe verschiedener Detektoren, die zusammen das Compressed Baryonic Matter (CBM) Experiment bilden. Das ist ein Schwerionenkollisionsexperiment, bei dem, bei moderaten Strahlenergien aber mit extrem hohen Kollisionsraten, Atomkerne zur Kollision gebracht werden. Im Rahmen der Kollisionen steht viel Energie zur Produktion neuer Teilchen zur Verfügung. Die Detektoren des Experiments ermöglichen es, diese Teilchen oder ihre Zerfallsprodukte nachzuweisen und so das Kollisionsgeschehen zu rekonstruieren. Das CBM Experiment soll einige Forschungsfelder abdecken: So sollen besonders seltene Teilchen und ihre Produktions-/Wechselwirkungsquerschnitte beobachtet und vermessen werden. Des Weiteren sollen Zerfälle von Hyperonen – Baryonen, die mindestens ein Strange-Quark enthalten – gemessen werden. Zudem ist die Di-Leptonen Spektroskopie ein weiteres spannendes Forschungsfeld, da sie als elektromagnetische Proben nicht der starken Wechselwirkung unterliegen. Zuletzt ist auch die Untersuchung des Phasenraums baryonischer Materie ein weiteres Ziel von CBM. Wenn bei einer Schwerionenkollision kurzzeitig extrem hohe Drücke und Temperaturen herrschen, können sich die Eigenschaften der Materie ändern und es kann, gemäß der derzeitigen Interpretation schon durchgeführter Experimente, ein Materiezustand erzeugt werden, der Quark-Gluon-Plasma (QGP) genannt wird. Das CBM Experiment wird mit den hohen erreichbaren Dichten und langen Lebensdauern hier voraussichtlich nur an die vermutete Grenze des Phasenübergangs kommen und kann beispielsweise zu der Suche nach einem kritischen Punkt im Phasendiagramm beitragen.

Diese Fragestellungen werden mit multidifferentiellen Analysen großer Datenmengen untersucht, die im Rahmen von Experimenten über mehrere Wochen aufgezeichnet werden. Zur Qualitätssicherung dieser Daten und zur Sicherstellung des kontrollierten und sicheren Betriebs der Detektorsysteme bedarf es spezialisierter Kontrollsysteme für den Betrieb. Einheitliche, detektorübergreifende Methoden ermöglichen eine effektive Aufsicht über das Experiment, und zwar während der Datennahme wie im Standby-Betrieb. Für die Minimierung systemati-

scher Fehler, müssen die Betriebsparameter so konstant wie möglich gehalten und aufgezeichnet werden oder – wo dies nicht möglich ist – so genau wie möglich vermessen werden, um Zeiten mit grenzwertigen Betriebsbedingungen anschließend aus den aufgezeichneten Daten ausschließen zu können.

Die dazu genannten Ziele werden auf Ebene des Gesamtexperiments mit der Entwicklung des Experiment-Kontrollsystems (ECS) verfolgt; auf Detektor-Ebene ist dafür die Bereitstellung eines Detektor-Kontrollsystems (DCS) notwendig. Für CBM wurde EPICS als Software-Framework für die Entwicklung der Detektor-Kontrollsysteme ausgewählt. Die Umsetzung eines solchen EPICS-Kontrollsystems für die Validierung eines sicheren Betrieb des aktuellen Prototypen des Mikro Vertex Detektors im Vakuum ist der hauptsächliche Gegenstand dieser Arbeit. Das Kontrollsystem kann zudem über den Kontext dieser Arbeit hinaus als “Blaupause” des Kontrollsystems für den zukünftigen CBM Mikro Vertex Detektor gelten.

Zunächst wurden die zu überwachenden und zu steuernden Prozessvariablen und Geräte identifiziert, wozu außer den Vakuumkomponenten, einem Temperiersystem und Niederspannungsnetzteilen auszugsweise auch Parameter aus dem Datenstrom des Datenerfassungs-Systems gehören. Für diese Geräte wurde EPICS Unterstützung (“device support”) bereitgestellt; teils musste dafür die Datenkommunikation nachkonstruiert werden, sofern hierfür keine Dokumentation zur Verfügung stand. Des Weiteren wurden interaktive graphische Benutzeroberflächen für das Detektor-System erstellt. Das beinhaltet eine mit der Anwendung Control System Studio/phoebus realisierte Oberfläche, die für Detektorexperten und Bediener bei Experiment-Schichten vorgesehen ist. Zusätzlich wurde eine Weboberfläche entwickelt, die eine Möglichkeit bietet, von jedem Gerät und verschiedensten Zugriffsorten aus den Status des Detektors einzusehen, ohne jedoch Änderungen vornehmen zu können. Eine sogenannte Archiver-Software ist ebenfalls Teil des Kontrollsystems. Sie zeichnet den Verlauf der Prozessvariablen auf und erlaubt es, deren historischen Verlauf für Fehlersuche und Korrelationen bei Artefakten heranzuziehen. Weiter sorgt ein Alarm-System dafür, dass Fehlerzustände leicht identifiziert und mittels hinterlegter Hinweise behoben werden können. Akustische und optische Signale zur Warnung der Benutzer wurden getestet.

Nach der erfolgreichen Inbetriebnahme des oben beschriebenen Kontrollsystems wurden verschiedene Tests durchgeführt, um dessen Zuverlässigkeit für den Betrieb des Prototyps zu testen. Neben automatisierten Testzyklen der Kühl- und Vakuumsysteme wurde die Auslese der Sensoren in verschiedenen Szenarien getestet. So wurde der Prototyp mit Photonen einer radioaktiven ^{55}Fe Quelle beleuchtet.

Die Verwendung von Container-Virtualisierung zur Bereitstellung und zum Betrieb des Systems ist naheliegend, da innerhalb eines Container-Abbildes alle Software-Abhängigkeiten enthalten sind. So entfallen Probleme, die etwa beim Einspielen

von Updates oder im Fall einer notwendigen Systemneuinstallation nach einem Systemausfall auftreten können. Zudem konnte so die Kompilierzeit für Teile des Kontrollsystems, die auf Einplatinencomputern ausgeführt werden, drastisch reduziert werden. Auf Grund der genannten Vorteile wurde erwogen, das Kontrollsystem insgesamt einer Software-Modernisierung zu unterziehen und die Ergebnisse dieser Analyse sind in Form eines Whitepapers in Kapitel 7 zu finden.

Der zuverlässige Betrieb des Auslese- und Steuerungs-Systems des MVD Prototypen PRESTO konnte erfolgreich ermöglicht werden. Das umgesetzte und beschriebene System wird dem künftigen Vollausbau des Mikro Vertex Detektors des CBM Experiments als zuverlässige Grundlage dienen. Im Rahmen dieser Arbeit gemessene und ausgewertete Daten aus dem (24/7) Betrieb des MVD Prototypen, der in vielen Aspekten dem späteren MVD bereits sehr nahe kommt, zeigt die hohe Sensitivität des Detektors auf, etwa bezüglich der Messung von Hintergrundaktivität und von Sekundärteilchen kosmischer Strahlung.

Abschließend ist festzustellen, dass beim Betrieb des MVD Prototypen im Vakuum auch nach langjährigem Betrieb und nach vielen Temperatur- und Vakuumzyklen – im Rahmen der zur Verfügung stehenden Anhaltspunkte – keine Leistungsmin- derung zu beobachten war.

List of Figures

2.1	Cave of the CBM experiment also containing the HADES experiment.	9
2.2	Overview of the GSI/FAIR research facility.	10
2.3	Maximum interaction rates for current and future heavy-ion experiments.	12
2.4	Hadron abundances and predictions of the statistical hadronization model with data from ALICE.	12
3.1	Mass stopping power for muons in copper.	18
3.2	Electronic energy deposit distribution for a 10 GeV muon traversing 1.7 mm of silicon.	19
3.3	Cluster size vs. energy deposition of carbon ions in a MIMOSA-26 sensor.	20
3.4	Standard cartesian & spherical coordinate system to describe cosmic rays.	23
3.5	Working principle of a simple CMOS MAPS sensor.	26
3.6	MVD Mechanical design of the MVD.	28
3.7	Material budget of the MVD detector geometry.	31
4.1	The prototype PRESTO.	35
4.2	Production of the front-side of PRESTO.	36
4.3	Planning and implementation of Prototype of the Second Station (PRESTO)'s operation in the vacuum chamber.	38
4.4	EPICS-controlled devices for PRESTO.	40
4.5	Feed-through vacuum flange production.	42
4.6	Read-out chain of the TRB3-based DAQ of PRESTO equipped with MIMOSA-26 sensors.	43
4.7	Revised ultra-thin FPC cables for MIMOSA-26/PRESTO and test of their signal integrity.	44
4.8	Photos of the DAQ set-up for PRESTO.	45
5.1	Deployment of EPICS components in containers.	56
5.2	EPICS controls network topology foreseen for CBM.	57
5.3	Computer architecture for the operation of the prototype PRESTO.	58

List of Figures

5.4	Cooling plant and hardware used to reverse-engineer its communication protocols.	59
5.5	Overview GUI for the control system implemented with CSS Phoebus.	61
5.7	EPICS Dashboard website.	63
5.8	Entities connected to the EPICS network for PRESTO.	66
5.9	The generalized sample processing chain to test for stability of conditions	73
6.1	Long-term high-vacuum test with three consecutive vacuum cycles.	77
6.2	Histogram of frame size of each sensor for each day during the early 2019 long-term vacuum campaign.	78
6.4	Pressure during 130 automated vacuum cycles.	79
6.5	Detailed vacuum pumping curve.	80
6.6	Surveillance of the PRESTO Cooling System 2017 to 2020.	80
6.7	Cooling cycle on 2018-12-11. The volume flow almost dropped to	81
6.8	Cooling cycle of PRESTO on 2019-12-01.	81
6.9	Illustration of the experimental setup for the ^{55}Fe measurements. .	85
6.10	Photo of the experimental setup for the ^{55}Fe measurements in vacuum.	85
6.11	Dimensions of the shielding mask for the Fe-55 test in vacuum. . . .	85
6.12	Hit pattern analysis of a masked ^{55}Fe source on a sensor of PRESTO in vacuum.	86
6.13	Close-up view of Fe-55 hits on a MIMOSA-26 sensor.	87
6.14	Typical setup to measure cosmics with multiple planes (left) vs. the single upright sensor approach used in this measurement (right). .	88
6.15	Geometric considerations of the grazing incidence angle needed to create a cluster of four consecutive pixels in MIMOSA-26 sensors. .	100
6.16	Verification of the MC sampling scheme for cosmics following a $\cos^2 \theta$ distribution.	101
6.17	Effect of the cuts on the number of cosmics surviving in the MC (Monte Carlo) data.	101
6.18	Monte-Carlo simulation of cosmic muons distributed according to $\cos^2 \theta$	102
6.19	Merging fired pixels to clusters: example hit patterns.	103
6.20	Rejection of clusters involving $\geq 50\%$ noisy pixels.	103
6.21	Visualization of the effectiveness of cluster classification (noise/signal).	103
6.22	Visualization of the discrete upscaling of a cluster shape.	104
6.23	An example for how upscaling a binary cluster image using nearest neighbour interpolation helps determining its moments of inertia precisely.	104

6.24	Examples for reconstructed projected particle directions from clusters encountered in the data.	105
6.25	Measured distribution of cluster angles.	105
7.1	Deployment of EPICS components on a Kubernetes Pod with the CNI plugin Multus.	116
7.2	The architecture of Prometheus and its typical ecosystem's components.	118
7.3	Architecture for data visualization of EPICS PVs with Grafana & Prometheus.	118
7.4	Setup for the EPICS Archiver Appliance scalability test.	120
7.5	EPICS Archiver Appliance scalability test with 250 000 PVs.	121

1 Introduction

The Compressed Baryonic Matter (CBM) experiment at Facility for Antiproton and Ion Research in Europe (FAIR) is going to probe matter under extreme conditions [1, 2]. The collision of heavy ions in a fixed target geometry will create extremely dense nuclear matter for a short time, the final states of which will be detected in a cascade of particle detectors measuring different aspects. In addition, also proton induced reactions (p-A) will be studied, which can provide reference measurements for the heavy-ion collisions (A-A). With center-of-mass energies of up to 11 AGeV for gold-gold collisions using the SIS100 accelerator, the collisions will create matter in an intermediate phase space region between high temperatures on the one hand and high pressure (i.e. baryon chemical potential μ_B) on the other hand, resulting in large interaction times for the baryons. What makes CBM unique is the extremely high collision rate capability (in the range of 100 kHz for A-A and 10 MHz for p-A), enabled by a trigger-less free-streaming data acquisition system directly feeding data to a compute cluster for on-line event reconstruction and selection. This enables the search for particles with low production cross-sections and allows multi-differential analyses with high statistics to be carried out. The unique data expected from CBM will allow to put effective theories and various models used to describe the physics at very high pressure and temperature to a test.

The Micro Vertex Detector (MVD) is the first detector in a cascade of detectors following the fixed-target of the Compressed Baryonic Matter (CBM) experiment, where the interactions of the nuclear collisions will take place. The Micro Vertex Detector (MVD) and the following Silicon Tracking System (STS) detector are placed inside the field created by the magnet of the experiment, allowing to measure the particles' momenta via their track curvature in the magnetic field by detecting space points of particle trajectories. Both, the MVD as well as the Silicon Tracking System (STS) consist of stations with position-sensitive detectors able to sense traversing electrically charged particles. High performance algorithms were developed to “connect the dots” and reconstruct tracks from the hit on the subsequent detector planes. While the MVD can contribute to this overall tracking task, its main task is high-precision track extrapolation towards the target. This way, the primary interaction point (primary vertex) can be identified. Secondly, it can help identifying tracks/particles not originating from the target (tracking de-

cay products of short-lived particles decaying outside the target (like multi-strange hyperons or open charm), or tagging background particles). Thirdly, the MVD helps assisting to increase S/B (signal over background ratio) in dielectron measurements, e.g. by suppressing closed pairs from conversion of photons.

In order to fulfil its role for the experiment, the MVD has to meet most diverse requirements, such as high detection efficiency, rate capability, ability to operate in vacuum, low material budget and last, but not least, stable and reliable operation during the operation of the experiment.

This thesis is a contribution towards the control and readout system of the Micro Vertex Detector (MVD) of the CBM Experiment. Specifically, the control system of a prototype called PRESTO was developed and the whole detector system transformed to a more robust, self-controlling unit.

1.1 Matter under Extreme Conditions

When subjected to extreme conditions (pressure or baryon density, temperature), nuclear matter changes its properties [3]. The quarks normally confined in nucleons, might form a new state of matter called a quark gluon plasma as chiral symmetry is restored under those conditions. The high energy available in the fireball of such a collision allows for quark-antiquark pair production as well as thermal production of particles. While electro-magnetic probes like electrons, muons or (virtual) photons, are expected to leave the fireball (i.e. the hot and dense medium) with a high probability of not interacting further, they are considered to give access even to the early stages of the collision, before statistical effects such as thermalization of the medium would modify the character of typical interactions. The bulk of the newly created hadrons typically emerges from the fireball following a thermal particle production function according to a Boltzmann distribution, with the abundance/multiplicity of different particles depending mostly on their mass [4]. Various models predicting the outcome of such collisions exist, which make different simplifications and assumptions. They may take different interactions into account, may use different degrees of freedom (i.e. sets of particles that can be produced), may or may not approximate certain phases with a macroscopic description such as hydrodynamics, may be suitable for higher or lower energies, etc. Examples for such models are UrQMD [5, 5, 6] and SMASH [7, 8]. The interplay between experimental results and refined theoretical models has advanced the understanding of matter under extreme conditions in the past.

1.2 The Need for High-Rate Experiments

The Compressed Baryonic Matter (CBM) experiment is particularly targeted to record heavy-ion collisions at unprecedented repetition rates. This will allow to explore the nature of strongly interacting matter with a level of precision and depth that was unheard of before. When particles collide in a physics experiment, new particles can be created, if sufficient energy is available in the collision system. If the elementary production threshold for a given particle is below the incident energy per nucleon, its direct production in a single binary collision is possible. Otherwise, subthreshold particle production might still occur but rarely, i.e. with a highly reduced probability resulting in very low yields. Another reason for rare particles are small cross-sections for processes with small coupling constants (e.g. particles created in electro-weak processes) [9, p.657]. Many of the particles produced in such an inelastic processes are unstable and decay after an average lifetime that is characteristic for the particle in question, thereby following the well-known exponential decay laws. To observe particles with a short life-time, it is beneficial if the particles have relativistic speeds: they appear to be longer lived in the laboratory frame, as the decay time is tied to their Eigenzeit (i.e. proper time) – an effect called relative velocity time dilation [10, p.19]. This leads to typical lifetimes in the order of $\text{cm } c^{-1}$ for strangeness and $\mu\text{m } c^{-1}$ for charm particles in a fixed target experiment with beams in the range of a few AGeV. By carefully observing and measuring the daughter particles, the decayed particle can be reconstructed and its properties determined. In fact, large parts of today’s knowledge about short-lived particles and their interactions stems from collision experiments reconstructing and analysing those decay processes. Which particles are being produced depends on the production cross-section for the specific process. Particles which have a small production cross-section and a very short lifetime are hard to find. The more seldomly a specific particle is being created, the more collisions (“events”) have to be recorded for discoveries with certain significance. In other words, a lower cross-section or production probability can be compensated by a larger number of events: $N_{\text{part}} = p \cdot N_{\text{event}}$. Increasing the number of events within a measurement campaign of fixed duration, the most evident possibility is to raise the interaction rate. Furthermore, the vertexing precision plays a role when reconstructing decaying particles. Due to the exponential decay of particles, the number of identified secondary vertices and thus reconstructable decayed mother particles is proportional to $\sim N_{\text{event}} e^{-\sigma_{\text{geo}}}$, where σ_{geo} is the geometric (secondary) vertex reconstruction precision. Again, the proportionality to N_{event} indicates that in addition to improving the vertexing precision, raising the interaction rate of the experiment will also improve statistics. Other ways to improve significance of results can be improved cuts, corrections or background removal techniques in the data-analysis stage. While those methods are improving significantly, for example

1 Introduction

by introducing artificial neural networks in selection processes, they can also introduce new unwanted biases and their effectiveness is limited for extremely rare events. In order to reach the highest (interaction) rates, experiments have to improve the performance of all their components, from the sensors and detectors, via the data acquisition systems to the data-preselection and reconstruction algorithms on the data-recording compute cluster.

There are also further reasons demanding high-rate, high-statistics experiments. Notably the analysis of exponential spectra demands high statistics to cover the high energy ends as precisely as possible. Another technique asking for highest statistics are multi-differential analyses, where the recorded data is subject to cuts on various dimensions, reducing the remaining statistics significantly, which can be compensated by higher initial statistics.

More information on how the CBM experiment aims to push the boundaries of rate capability to whole new limits with the hope to gain valuable new insights into nuclear matter is provided in chapter 2.

As was explained, high rates are favourable for statistically significant findings in many physics cases for the experiment. They come, however, with big challenges on the technological side of the experiment, as they translate into high occupancies, data rates, increased dead-times, radiation damage and ageing. The detection and handling of these effects asks for a close monitoring of the detector, an important task fulfilled by a Detector Control System (DCS), such as the one introduced in this thesis.

1.3 Sensing the Physical World with Particle Detectors

In today's well-established physics paradigms, particles are the constituents forming the matter in the universe, be it in gas, liquid or solid state. What started with the discovery of the photoelectric effect by A. Einstein in 1905 found its deep foundation in the development of QFT (quantum field theory): particles do not only represent matter but also the fundamental forces in the form of exchange particles, carrying momentum and information. Consequently, probing and understanding the sub-atomic physical world means: sensing particles. An important question in the physical science has always been: are there fundamental particles? Over the past century, the understanding of composite vs. fundamental particles was vastly improved. What was once considered a fundamental particle is shown today to be a composite state. After the proton was discovered 101 years ago by Rutherford [11], it took decades, until (valence) quarks were proposed as funda-

1.4 The Importance of Stable Detector Operation

mental constituents of mesons and baryons (like the proton) by Zweig [12] and Gell-Mann [13].

Sensing a particle with a particle detector means measuring its properties; this covers a wide range of detecting its sheer presence, measuring its velocity, momentum, mass, production cross section, decay channels, determining its quantum numbers (e.g. iso-spin, charge) and more. The ability to trace and identify particles is essential for discoveries like the aforementioned proton.

In order to detect a particle with a detector, an interaction of the particle with a medium is necessary. In the most typical case, electrically charged particles interact via the electro-magnetic force with matter in the detector. For detectors tracking the path of the particle, this interaction should be as small as possible in order to keep changes to the particle's momentum as low as possible. It must be large enough, however, to allow for reliable detection. In the case of detectors used for energy measurements (calorimeters) the interaction with the detector material is maximised with the aim to fully stop the particle in the detector material, usually in a cascade of collisions. The original kinetic energy of the particle can then be reconstructed from the measured signal.

The particle detectors used in experimental physics are getting ever more precise, fast, versatile, or – ideally – all of it at the same time. Their geometric acceptance, precision of time and space measurement, efficiency, radiation hardness and dead time are examples of the generic features that detector physicists are working hard to improve from generation to generation. All of these properties can extend the scenario the detector can be used for (e.g. tracking, momentum or velocity measurement, identification).

1.4 The Importance of Stable Detector Operation

Physics experiments are successful if they allow to challenge the assumptions about the world – what it consists of, and which laws the constituents obey to. For any such experiment, it is desirable to keep most parameters stable and vary only a single controlled parameter. If controlling (i.e. ‘preparing’) the varied parameter is impossible, it is more and more common to reconstruct it from data afterwards. An example is the impact parameter of a nuclear collision, which cannot be adjusted prior to the collision, but can be determined (‘reconstructed’) from the data once a collision event has been recorded.

Experience shows that it is of paramount importance for reliable operation and a subsequent, successful analysis of the data to keep any external parameters as constant as possible during the full length of experiment, typically multiple weeks.

1 Introduction

The path from recording raw data to finally answering physics question involves many intermediate steps. Special care needs to be taken when transforming raw data to calibrated data to guarantee high significance for the results of the physics analysis. To allow for a reliable calibration over the full experiment, e.g, environmental parameters have to be recorded, assessed, and kept constant. This requires their comparison to set values (ranges), notification, re-adjustments by algorithms or operators, and documentation. As an example, high-quality track reconstruction in a TPC (Time Projection Chamber) requires to keep the gas temperature and pressure constant. Any change beyond pre-defined limits has to be detected and recorded, operators have to be informed for a possible re-adjustment. However, by means of the recorded values and known detector physics, offline corrections and calibration would be still possible within a given range of parameters. That aspect is being discussed in the context of monitoring the stability of a particle detector's conditions in section 5.15.

This thesis documents an important maturation of the system used to operate the prototype of the MVD and should eventually enable stable operation of the MVD during the operation of the CBM experiment in order to record high quality data of the collisions with maximum uptime.

1.5 The Micro Vertex Detector of CBM and its Prototypes

The Micro Vertex Detector of CBM will be a detector consisting of four planar stations equipped with CMOS (Complementary Metal-Oxide-Semiconductor) pixel sensors. Within the CBM experiment, which will be introduced in chapter 2, it fulfils the task of tracking and vertexing. It helps reconstructing the primary vertex as well as secondary vertices, allowing to distinguish between particles originating from the primary collision and particles that decayed and were reconstructed with a (desired) vertex precision of 50 μm .

At the technical level, the Micro Vertex Detector (MVD) performs well, if its sensors reliably report any incident particles (high digital detection efficiency), while staying silent in the absence of incident particles (low fake hit rate). To achieve such a reliable mode of operation, the detector has to be kept at a suitable working point. This implies aspects such as stable supply voltages for the CMOS pixel sensors, tuning and monitoring thresholds, ensuring signal integrity of the sensor output links and more. Furthermore, the sensors need to be cooled to a target temperature of $\sim -20^\circ\text{C}$ to compensate for radiation damage over the lifetime of the sensor and to avoid thermal runaway in the absence of convective air cooling due to the operation in vacuum. Due to radiation damage and ageing effects, the

ideal working point of the individual sensors will certainly need to be readjusted during the lifetime of the detector and the operation in vacuum requires monitoring of the residual pressure. All of the aspects mentioned here are examples of vital operational parameters, that need to be controlled for a successful operation of the detector.

To validate the various concepts of the detector, like its mechanical and electrical integration, the read-out system and the control system, there have been multiple prototyping efforts for the Micro Vertex Detector project. A prototype called “MVD Demonstrator” was created in 2009 and another prototype simply called “MVD Prototype” in 2012. The following prototype was called **PRESTO** and was put to operation with a fully overhauled read-out system [14]. The prototype, its production and commissioning is discussed in chapter 4.

1.6 Development of the Control System for the Prototype PRESTO

This thesis completes the prototype Prototype of the Second Station (PRESTO) with a full-blown Detector Control System (DCS), that is necessary to operate the prototype safely with little human intervention at the desired overall working point. Beyond applying to the prototype, the system and the gained insights, should ultimately contribute to the final MVD of CBM. Being the system of choice for CBM, the Experimental Physics and Industrial Control System (EPICS) was selected as framework to implement that control system. The design choices, components, implementation details and architecture of the control system are described detail in chapter 5.

1.7 Validation of the Vacuum Operation of the Prototype PRESTO

After creating and setting-up the control system, PRESTO was operated 24/7 in the laboratory and subjected to various tests to validate its continuous and reliable operation (in vacuum). Chapter 6 presents the field tests performed for this purpose of validating the prototype in the foreseen operating conditions. Those tests comprise cooling and vacuum cycles, observations on electronic noise, operation with a radioactive ^{55}Fe source with cooling under vacuum, and measurements of the background activity including an indirect measurement of cosmics.

The final chapter, chapter 7, is a white paper discussing how the Experimental

1 Introduction

Physics and Industrial Control System (EPICS) control system can be subjected to an 'application modernization' in order to make it more maintainable and scalable.

2 The CBM Experiment: Matter at High Baryon Chemical Potentials

The Compressed Baryonic Matter (CBM) experiment is currently being under construction at the site of the Facility for Antiproton and Ion Research in Europe (FAIR). It is an experiment designed to investigate matter at high baryon chemical potential. Probes particularly interesting for the CBM experiment include collectivity, event-by-event fluctuations, strangeness, lepton pairs, charm, as well as hypernuclei and strange objects [1]. Figure 2.1 shows a three-dimensional rendering of the experiment placed in a cave for radiation protection. The accelerator beam enters the cave from the left side of the picture and passes the HADES (High Acceptance Di-Electron Spectrometer) experiment first. On the right side, the CBM experiment is visible in the so-called RICH-configuration (with the MUCH detector moved out to its parking position).

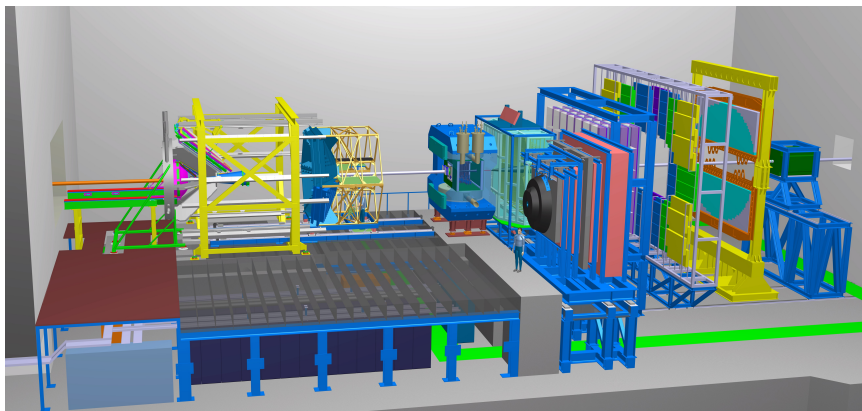


Figure 2.1: Cave of the CBM experiment also containing the HADES experiment (3D Model). Taken from [15].

2.1 FAIR and the SIS100 Accelerator

The Facility for Antiproton and Ion Research in Europe (FAIR) is a joint international effort to create a research facility for accelerator-based science by building

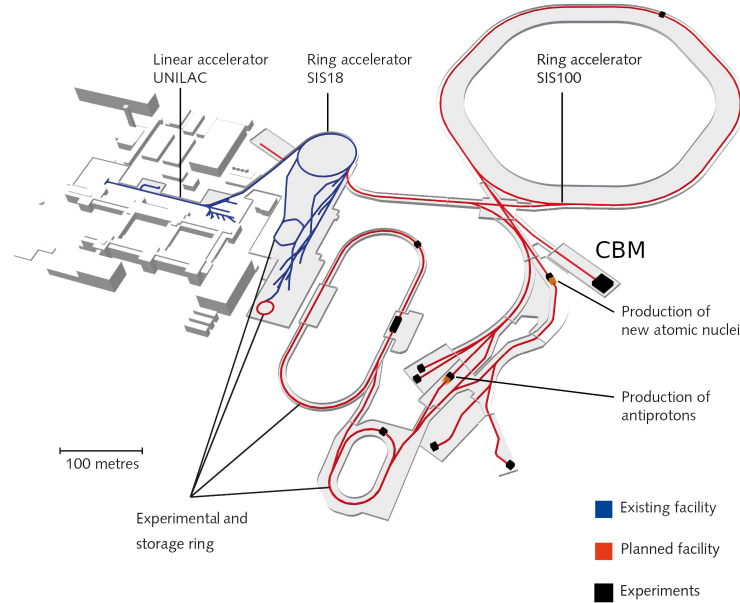


Figure 2.2: Overview of the GSI/FAIR research facility. Taken from [16].

upon the existing infrastructure of the already existing facility GSI Helmholtzzentrum für Schwerionenforschung (GSI). Figure 2.2 provides an insight into the infrastructure currently under construction for FAIR. Facility for Antiproton and Ion Research in Europe (FAIR) will extend GSI with a more capable accelerator, storage rings and dedicated experiments from different fields, namely Atomic Physics, Plasma physics and Applications (APPA), antiProton ANnihilation at DArmstadt (PANDA), Nuclear Structure, Astrophysics and Reactions (NUSTAR), and Compressed Baryonic Matter (CBM). The Schwerionensynchrotron 100 (SIS100) is the accelerator ring built for FAIR and its experiments. The status of SIS100 and its plans were recently described in Spiller et al. [17].

2.2 The CBM Experiment

The Compressed Baryonic Matter (CBM) Experiment is a fixed target heavy-ion experiment, suited to probe dense nuclear matter at high baryon densities μ_B and intermediate temperatures. It is well suited to find rare signals due to its high performance detectors and a particle beam with a high luminosity:

“The CBM experiment is designed for the detection of signals from the high-density phase. These signals, however, are produced very rarely, either because of the low cross section (charm) or because of the small branching ratios into lepton pairs (low-mass vector mesons). Therefore, the CBM experimental program can only be realized with

a combination of fast detector systems and high beam luminosity as provided by the FAIR accelerators.” ([9, p. 853])

(Note that the production of the mentioned charm particles seems unlikely with the beam energies available with the SIS100 accelerator currently under construction, and the suspension to realize the Schwerionensynchrotron 300 (SIS300) accelerator.)

The collision system will be subject to a strong Lorentz boost in the laboratory frame of reference due to the fixed-target geometry of the experiment. As a result, CBM can be a rather compact (and as a result relatively inexpensive) experiment, with its detectors placed in forward direction from the target at a polar angle of 2.5 to 25°. In turn, the high resulting track density requires detector stations with a high granularity and excellent radiation hardness.

CBM will measure heavy-ion collisions at center-of-mass energies also covered by prior experiments but it will do so with a several orders of magnitude higher interaction rate, see fig. 2.3. The figure shows the maximum rate current and future experiments can achieve when colliding heavy-ions in the 1 to 50 AGeV energy range. CBM stands out with an unprecedented rate capability, that gives access to rare probes impossible to measure otherwise [2, p.6][9].

Figure 2.4 shows a recent result of ALICE (A Large Ion Collider Experiment) where measured hadron abundances were compared to the predictions of the statistical hadronization model. The main parameter for the model is the temperature of the medium in the fireball of the collision after thermalisation. Such a measurement is foreseen with CBM too, and could provide additional insights at the rather low beam energies of CBM (when compared to ALICE, for example). As such a measurement might cover 10 orders of magnitude in particle production yields, the high rate capability of CBM will prove very useful to obtain statistically significant results. They may complement previous measurements done by HADES [4], potentially reducing the statistical uncertainties and adding further particles with very low production cross-sections to the analysis.

The CBM project is structured into the following detectors and other sub-projects: Beam Counter, DAQ, FLES, Infrastructure and Engineering, MUCH, MVD, Magnet, PSD, RICH, STS, TOF and TRD [20, 21].

The experiment can be configured for different setups optimized for specific physics objectives, called ‘electron’, ‘hadron’ or ‘muon’ setup. Depending on the setup, different detectors are planned to be part of the experiment. For example, either the RICH (ring-imaging Cherenkov detector) or the MUCH (Muon Chamber) detector might be part of the experiment. The RICH detector is foreseen to perform lepton identification together with the TRD (Transition Radiation Detector) in the case of the electron setup, while the MUCH will be part of the muon setup(s),

2 The CBM Experiment: Matter at High Baryon Chemical Potentials

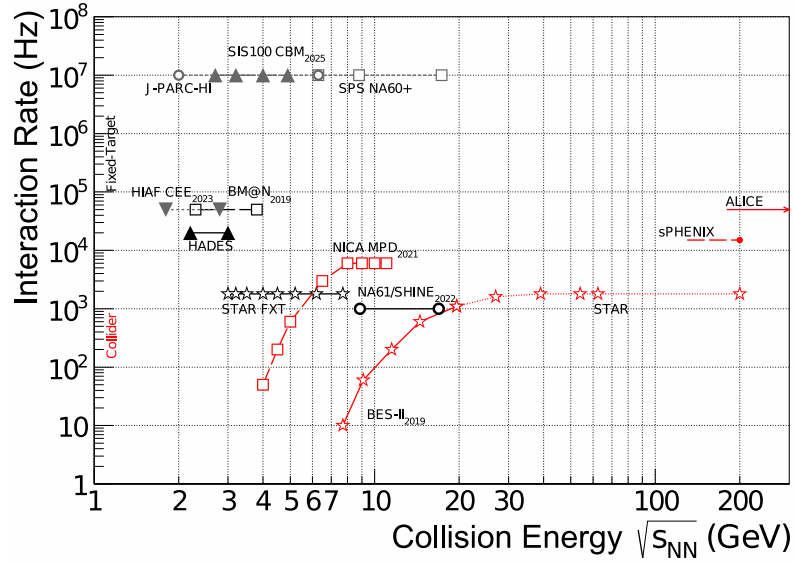


Figure 2.3: Maximum interaction rates for current and future heavy-ion experiments as a function of the CMS energy $\sqrt{s_{NN}}$. Taken from [18].

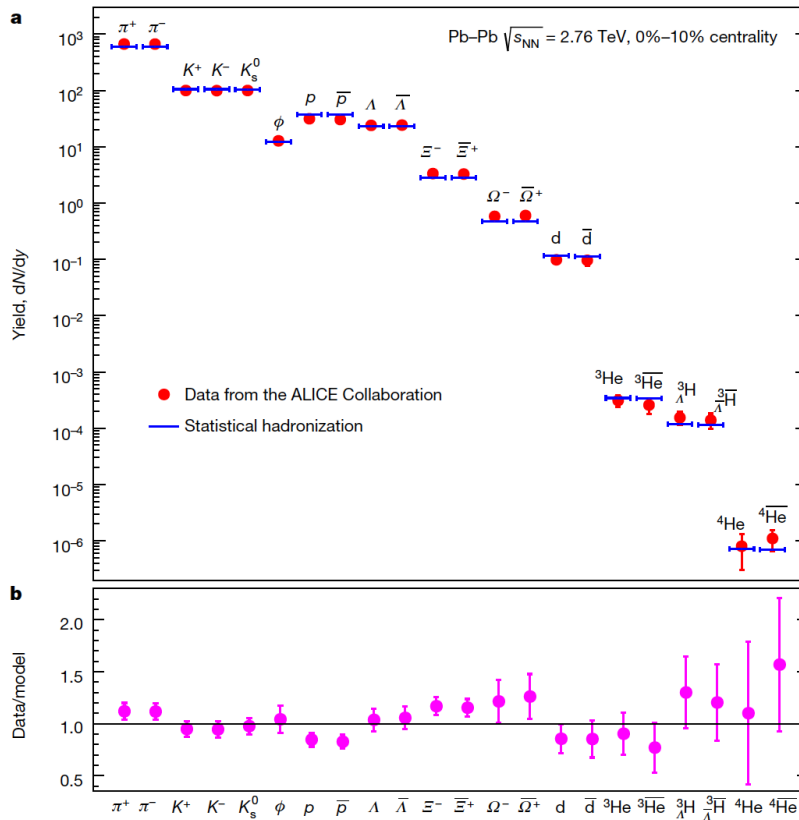


Figure 2.4: Hadron abundances and predictions of the statistical hadronization model with data from ALICE. Taken from [19, p. 324].

replacing the RICH in that case. The participation of the MVD is foreseen in the case of the ‘electron’ and ‘hadron’ setups, but not for the ‘muon’ setup. The details foreseen for each setup can be looked up in the CbmRoot setup files, see [22] and [23, slide 5]. Table 2.1 lists the detectors to be used for the different setups.

Configuration	MVD	STS	RICH	MUCH	TRD	TOF	PSD
sis100_electron{,_sts_long}	✓	✓	✓		✓	✓	✓
sis100_hadron{,_sts_long}	✓	✓			✓	✓	✓
sis100_muon_{jpsi,lmvm}{,_sts_long}		✓		✓	✓	✓	
sis300_electron		✓	✓		✓	✓	✓

Table 2.1: CBM setups and detectors intended to be used. Data source: [22].

2.3 Tracking and Vertexing Detectors

Tracking and vertexing detectors in physics experiments are the ones identifying tracks and thereby reconstructing the particles’ momenta via their bending radius in an externally created magnetic field. The vertexing aspect means that they allow to measure the primary vertex and to identify particles from secondary vertices. Silicon-based detectors of this category typically consist of position sensitive detector stations (planes), which detect a hit when charged particles traverse them. Algorithms can then ‘connect the dots’ and reconstruct the particles’ tracks.

In the case of the CBM experiment, the detectors assigned to this category are the MVD and the Silicon Tracking System (STS) [24]. They need to be positioned close to the interaction point for a precise track extrapolation towards the vertex. A low material budget is required to minimize the conversion of the crossing particles in the detector material and to minimize the deviation they experience due to multiple scattering while traversing the detector. In addition, a good knowledge of the magnetic field is essential for precise track fits, i.e. momentum measurements. Both, the MVD and the STS, will use silicon based semiconductor sensors; the MVD will be equipped with MAPS (Monolithic Active Pixel Sensor) while the STS will use silicon microstrip sensors. The use of the CMOS based MAPS for the MVD is necessary, as silicon microstrip sensors yield ambiguous results in the case of concurrent double-hits, which are much more likely at the distance the MVD is placed from the reaction (i.e., the target). While the MVD is currently planned to be composed of four consecutive stations between 5 and 20 cm from the target and placed inside the target vacuum chamber, the STS will be composed of eight tracking layers beginning at 30 cm and ending at 103.5 cm [25].

2.4 Particle Identification Detectors

Detectors dedicated to PID (Particle Identification) are designed to identify the type or class of a given particle traversing it. Multiple ways of PID exist, with one of the most popular methods being cuts based on the energy loss in the detector's material dE/dx . The CBM RICH (ring-imaging Cherenkov detector) detector will help identifying particles by measuring their velocity via a reconstruction of their Cherenkov radiation emission angle [26]. The emission of Cherenkov light is subject to a threshold effect, as it is only being emitted if the particles' velocity β exceeds the speed of light in the emitting medium. This threshold is around $0.4 \text{ GeV } c^{-1}$ for e^{\pm} , which allows to separate them nicely from pions at around $4.7 \text{ GeV } c^{-1}$ for the CO_2 gas mixture of the RICH [26, p. 29]. The MUCH (Muon Chamber) detector, in turn, can identify muon pairs. It achieves this by a staggered setup of hadron absorber plates and tracking stations [27]. CBM can be configured to include either the RICH or the MUCH, depending on the physics to be investigated by a specific measurement campaign. The TRD (Transition Radiation Detector) is used for electron identification and dE/dx measurements. It is composed of 4 layers, each consisting of a radiator and a MWPC read-out chamber [28]. A TOF (Time of Flight) detector allows precise measurements of the velocity β [29] based on the time measurements by an array of MRPCs (Multi-Gap Resistive-Plate Chambers).

2.5 Event Characterising Detectors

The PSD (Projectile Spectator Detector) is specifically designed for CBM and will allow to determine the number of participants, the collision centrality, as well as the reaction plane orientation [30]. It is a lead-scintillator calorimeter with a sandwich structure of 12×9 modules, "each consisting of 60 lead/scintillator layers [...] read out via wavelength shifting (WLS) fibers by MAPD" (Multi-Avalanche Photo-Diodes) [9, p. 872]. By measuring the energy of spectator nucleons, the number of participating nucleons can be determined. This helps to improve the centrality determination of each event from the particle multiplicities with the help of MC (Monte Carlo) Glauber or DCM-QGSM models.

2.6 Further Elements of CBM

In addition to the aforementioned detectors, a superconducting dipole magnet will be part of CBM to provide a vertical magnetic field with a bending power of

1 T m over the length of 1 m from the target in a lateral opening of 1.4 m [31]. Liquid Helium (LHe) will be used to cool the coil windings. The magnet will store ~ 5 MJ at its nominal current, which requires an elaborate quench protection for the case of a sudden loss of its superconductivity.

Furthermore, detectors to determine the beam position and quality will be implemented.

An elaborate online data processing is necessary to reconstruct the events in real-time for on-line event selection, requiring precise calibration of the detector during measurement campaigns. Therefore, the online event reconstruction performed by the FLES computing cluster are also important infrastructure elements of the CBM experiment.

3 The Micro Vertex Detector of CBM

The Micro Vertex Detector is the particle detector closest to the target of the CBM experiment. Being intended for vertexing as well as tracking of low momentum particles, it needs to provide a high granularity/resolution while adding as little material budget as possible.

The idea to add a micro vertex detector based on monolithic active pixel sensors to the CBM experiment was first reported in 2004 [32, p. 236][33, p. 11]. At that time, it was planned to be part of the STS detector and the FAIR Baseline Technical Report published in 2006, still details the plans under the name “inner tracking station (ITS)” of the STS detector [34, sec. 2.2]. Several publications since that time were elaborating on the requirements and design of this detector now commonly referred to as MVD or CBM-MVD. Specifically, [35] documents the mechanical design of the micro-vertex detector in great detail.

3.1 Energy Loss of Charged Particles in the Detector

For a particle traversing a silicon detector to be detected, its energy loss has to be big enough to create a sufficient amount of charges in the material to discriminate the resulting signal from noise. For rather slow and heavy particles, such as heavy-ion fragments, the deposited energy might be very high, causing a large electronic signal in the sensor. In both those cases, it is desirable to know the energy loss for a given particle species and momentum, be it to estimate efficiency or to determine the expected number of firing pixels.

In the case of heavy particles, the electronic energy loss with a velocity of in the region of $0.1 < \beta\gamma < 1000$, the semi-empirical Bethe equation can describe the “mean rate of energy loss by moderately relativistic charged heavy particles” [36, p. 536]:

$$\left\langle -\frac{dE}{dx} \right\rangle = Kz^2 \frac{Z}{A} \frac{1}{\beta^2} \left[\frac{1}{2} \ln \frac{2m_e c^2 \beta^2 \gamma^2 W_{\max}}{I^2} - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right]. \quad (3.1)$$

For MIPs (Minimum Ionizing Particles), however, the most probable value (MPV) for the energy loss when traversing a silicon sensor can be approximated as: [37,

p. 13]:

$$\left(\frac{dE}{dx}\right)_{\text{MPV}}^{\text{silicon}} \approx 0.3 \text{ keV } \mu\text{m}^{-1}. \quad (3.2)$$

Together with the mean energy needed for ionization in the silicon semi-conductor crystal of $E_{e-h} = 3.6 \text{ eV}$ [38, p. 31], a typical number of ~ 80 electron-hole pairs are generated per traversed micrometre.

In the case of a $15 \mu\text{m}$ thick epitaxial layer [39, Fig. 20 p. 48], this results in ~ 1200 charge carriers for a MIP (Minimum Ionizing Particle) traversing the sensor perpendicularly. While this value is the most probable value for the energy loss, large fluctuations are possible. They are described by the Landau distribution or the Bichsel straggling function introduced below for the case of muons.

For other cases (e.g. lighter particles, very low momenta, etc.), measured data is essential to determine energy loss or stopping power. The following paragraphs provide such data relevant for the measurements in this thesis.

Muons

The mass stopping power muons experience in Copper is shown in fig. 3.1. The range where the Bethe equation 3.1 is valid is found between $10 \text{ MeV } c^{-1}$ and $60 \text{ GeV } c^{-1}$.

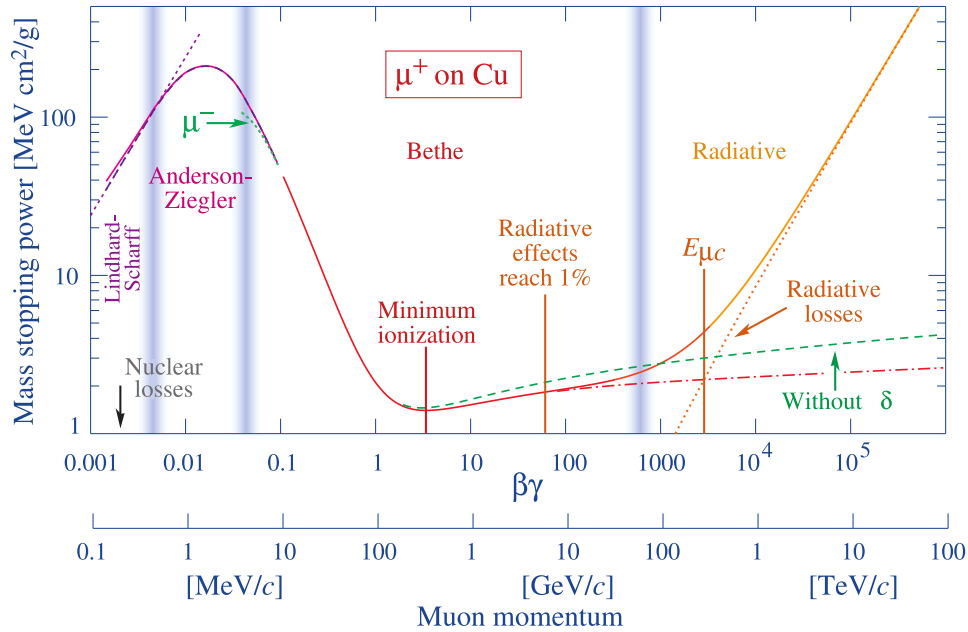


Figure 3.1: Mass stopping power for muons in copper. Taken from [36, p. 536].

While the mean energy loss is shown in fig. 3.1, the actual energy deposition of individual particles can fluctuate quite a lot and the distribution of the energy deposition can be described by the Landau distribution [36, p. 539] or more precisely

by a Bichsel straggling function [40]. Figure 3.2 shows the energy loss distribution for Muons traversing 1.7 mm of silicon.

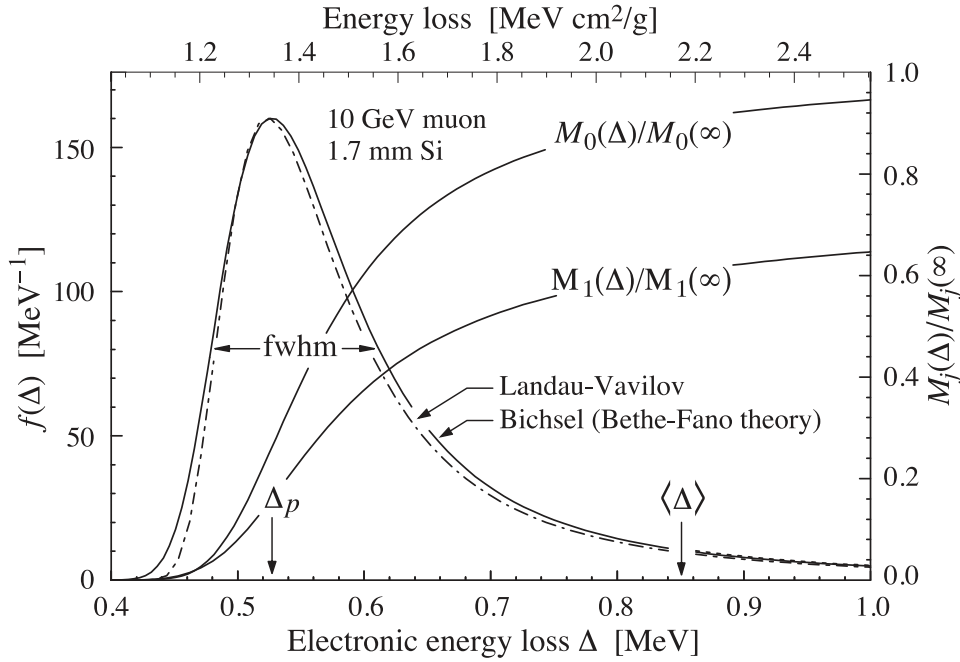


Figure 3.2: Electronic energy deposit distribution for a 10 GeV muon traversing 1.7 mm of silicon. Taken from [36, p. 539].

Heavy-Ion Fragments

Slow fragments of heavy ions can deposit huge amounts of energy in a silicon sensor compared to lighter particles, such as MIPs. Figure 3.3 shows the cluster size depending on the energy deposition from carbon ions (3.6 to 52.8 MeV per nucleon) in a MIMOSA-26 sensor. The rather high energy deposition from light ions creates a huge amount of electrons diffusing in the sensor, which leads to the large number of neighbouring pixels firing in coincidence (i.e. clusters). The data was extracted from the publication in order to fit it with the function $A \cdot E^n$. For the MIMOSIS sensor of CBM, clusters from heavy-ion hits are expected to be slightly smaller due to the improved depletion and other measures potentially leading to a more local charge collection. In addition to creating very large clusters (effectively creating a dead-time for that area in the specific frame), heavy-ion hits can also affect the lifetime of the sensors, as they also cause high radiation damage [39, sec. 8.3].

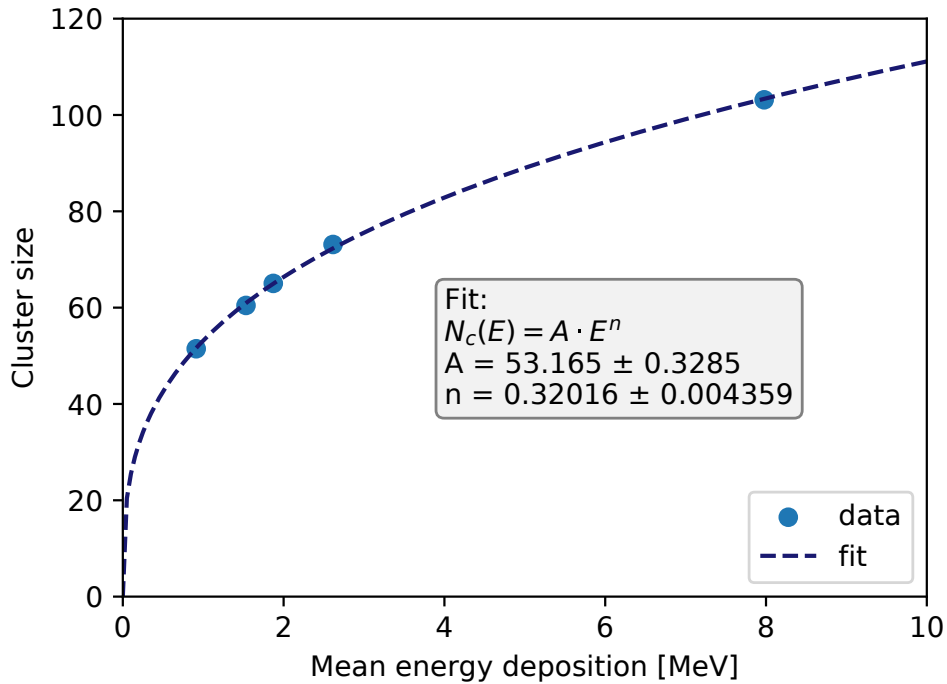


Figure 3.3: Cluster size vs. energy deposition of carbon ions in a MIMOSA-26 sensor. Data extracted and refitted from [41, p. 135].

3.2 Radiation Hardness of Silicon Pixel Sensors

While semi-conductor electronics in general are quite sensitive to ionizing and non-ionizing radiation damage, this is typically not an important aspect when used in non-radiation environments (e.g. consumer electronics). If used as a particle detector in a physics experiment, however, the particles to be measured represent a source of ‘radiation’ that cannot be avoided. The radiation hardness of a particle detector and its sensors is therefore a critical issue determining the reliability and life-time of the detector.

In the case of silicon pixel sensors, specifically MAPS (Monolithic Active Pixel Sensor) or CMOS pixel sensors, efforts over the past decades have improved their radiation hardness for use in particle detectors significantly [42, sec. 2.4.2] [43, sec. 3].

In general, damage from radiation is caused by ionizing and non-ionizing radiation. Ionizing radiation can result in broken chemical bonds at the interfaces between different materials (such as SiO_2 and Si) causing so-called “surface damage” [44, p. 80]. Non-ionizing radiation, however, can modify the crystal structure (lattice) of the silicon semi-conductor resulting in so-called bulk-damage, which can increase leakage currents.

To overcome issues with RTS (random telegraph signal) noise and leakage currents, which are increasing with the total radiation dose, the sensors can be actively cooled, ideally to sub-zero temperatures (in °C) [44, p. 163f]. In addition, low temperatures also reduce the thermal electronic noise present in the pixels, which is just as desirable.

Many different ways to improve radiation hardness are being explored. They range from using high-resistivity epitaxial layers for the charge-collection layer [43, ch. 5], via CMOS processes with smaller feature sizes [43, ch. 6], to mitigation strategies such as cooling to sub-zero temperatures during operation, redundant/triple-logic for digital circuits, thermal annealing after expose and many more.

In the context of this thesis, the cooling aspect dominates, as it is desirable to operate the prototype PRESTO reliably at temperatures in the range of -20°C to 0°C and the task of its detector control system to make sure this is the case during operation.

3.3 Material Budget Associated Detector Performance

For tracking and vertexing detectors, the matter making up the detector inside their acceptance can have a huge impact on the overall detector performance (for example on its secondary vertex precision). High energy charged particles scatter elastically with the nuclei of the detector material resulting in a slightly bent path, an effect commonly known as “multiple scattering”. Particles with lower momentum are generally more affected with larger resulting scattering angles. The matter making up a detector – expressed in a number called material budget x/X_0 – is a driving factor for the multiple scattering and thereby its overall performance.

This section introduces the detector precision, multiple scattering and radiation length in details and provides the equations to understand how they are related and to be able to calculate the effect on the detector performance.

3.3.1 Precision of Planar Detectors

Charged particles traversing matter are subject to elastic Coulomb scattering. For a particle detector, this is an unpleasant effect, often denoted as “multiple scattering”, where the detector’s material alters the path of the particle – i.e. the measurement has an influence on the observable. For a better understanding of this effect, the following equation relates the secondary vertex precision of a simple particle detector consisting of two planar position sensitive detector stations to its

geometry, material budget and position precision [35, p.25], [45, p.3]:

$$\sigma_{SV}^2 = \frac{\sigma_1^2 r_2^2 + \sigma_2^2 r_1^2}{(r_2 - r_1)^2} + \frac{\theta_{ms}^2 r_1^2}{\cos^4 \theta} \quad (3.3)$$

In this equation, the r_1 and r_2 denote the respective distance of the first and second station from the vertex, the term θ_{ms} denotes the average angular deviation resulting from multiple scattering after traversing the first detector station.

3.3.2 Multiple Scattering

The mean angular deviation of due to multiple scattering $\theta_{ms,rms}$ can be expressed as a function of the traversing particle's velocity βc , momentum p and electric charge z as well as the traversed matter's material budget x/X_0 , x being the traversed path length and X_0 being the radiation length [38, p. 33]:

$$\theta_{ms,rms} = \frac{13.6 \text{ MeV}}{\beta p c} z \sqrt{x/X_0} [1 + 0.038 \ln(x/X_0)] \quad (3.4)$$

3.3.3 Radiation Length

The radiation length X_0 is a material property related to the energy loss of charged (i.e. electromagnetically charged) high energy particles. It can be calculated using Tsai's semi-empirical formula [36, p. 541] and it is commonly approximated [46, p. 82] using:

$$X_0 = \frac{716.4 \text{ g cm}^{-2} A}{Z(Z + 1) \ln(287/\sqrt{Z})}, \quad (3.5)$$

where A and Z are the mass number and the atomic number, respectively. For many materials, their (mostly measured) radiation length can be looked up in [47].

For composite materials, the radiation length can be calculated from the mass ratio of its constituents.

3.4 Commissioning with Radioactive Sources and Cosmic Muons

This section briefly discusses the use of non-accelerator particles and radiation to commission sensors and detectors. A very typical example for those are radioactive

^{55}Fe sources and the use of cosmic muons. These methods are very popular due to the easy availability in smaller laboratories or during a shutdown or maintenance windows of accelerator facilities. For cosmic rays, it has even been reported that they were used for detector alignment [48].

3.4.1 Fe-55 Sources to Test Sensors

A radioactive source of ^{55}Fe is frequently used by detector physicists to test the response of a sensor/detector to photons. ^{55}Fe has a half-life of 2.747 a and emits photons in the range of 5.8 keV ($K\alpha_1$ & $K\alpha_2$) and 6.5 keV ($K\beta_3$ & $K\beta_5''$) [49]. Auger electrons with a kinetic energy of 5.2 keV are also emitted by ^{55}Fe [50, p. 26]. In general, it is the photons that are converted into an electronic signal in a sensor, the electrons lose their energy already in the outermost metal layer.

3.4.2 Muons from Cosmic Radiation

Cosmic radiation is a ubiquitous phenomenon discovered and unravelled during the 20th century [51]. Cosmic particles reach the earth's atmosphere in large numbers with exponentially falling energy spectra. Converting high energy particles to showers of lower energy particles, the earth's atmosphere serves as a protective shield against most of the cosmic radiation. Consequently, only a fraction of the original cosmic rays reach the earth's atmosphere, with most of the charged particles on ground level being secondary particles from decay showers, especially high-energy, minimum-ionizing muons.

To describe the angular distribution of cosmic muons, the laboratory reference frame (cartesian and spherical) is defined as shown in fig. 3.4.

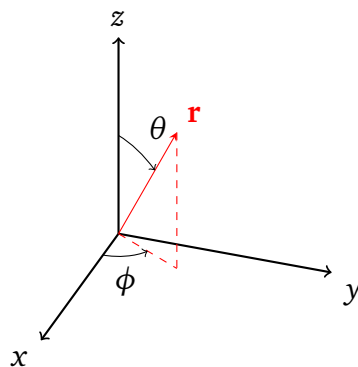


Figure 3.4: Standard cartesian & spherical coordinate system to describe cosmic rays.

3 The Micro Vertex Detector of CBM

The spherical coordinates are defined by the tuple (r, θ, ϕ) representing the radial distance r , the zenith angle θ and the azimuthal angle ϕ . This is the definition most commonly used in physics and standardized in [52]. The origin of the coordinate system is pinned to the observer. z is defined to point upwards from the local observer, i.e. ‘skywards’.

Peak Flux of Secondary Muons

The flux of secondary muons from cosmic radiation expected at earth’s ground level peaks at

$$I_\nu = j(\theta = 0) \approx 70 \text{ m}^{-2} \text{ s}^{-1} \text{ sr}^{-1} \quad \text{cf. [36, p. 512]}. \quad (3.6)$$

Angular Distribution and Mean Energy

For muons with an energy of $E_\mu \approx 3 \text{ GeV}$, the angular dependence of the muon flux is found to be proportional to $\cos^2(\theta)$ and the “mean energy of muons at the ground is $\sim 4 \text{ GeV}$ ” [36, p. 512]. As MIPs, those muons are expected to ionize $\sim 80 \text{ e}^-/\mu\text{m}$, when they hit a silicon sensor, cf. eq. (3.2).

Particle Current Expected for a Single Sensor Depending on its Orientation

In a typical case of a single-layer, horizontal particle sensor, integrating j over the full hemisphere, taking into account the projected sensor area $\sim \cos \theta$, yields a particle current density of:

$$J_h = \int_{\theta \leq \frac{\pi}{2}} I_\nu \cos^2(\theta) |\cos \theta| d\Omega = I_\nu \cdot \frac{\pi}{2} \text{sr} \quad [\text{with } d\Omega = \sin(\theta) d\theta d\phi] \quad (3.7)$$

resulting in a muon count rate of $\dot{N} \sim 0.66 \text{ min}^{-1}$ for a horizontal detector plane sized $1 \times 1 \text{ cm}^2$, which is in accordance with the experimentalists’ rule of thumb of $\dot{N} \sim 1 \text{ min}^{-1}$ cosmic events to be expected in such a setup [36, p. 512].

Now let’s assume the sensor is placed *upright* instead. Then the equation becomes:

$$J_\nu = \int_{\theta \leq \frac{\pi}{2}} I_\nu \cos^2(\theta) |\sin \theta \sin \phi| d\Omega = I_\nu \cdot \frac{\pi}{4} \text{sr} = \frac{1}{2} J_h \quad (3.8)$$

where $|\sin \theta \sin \phi|$ corresponds to the reduced projected area of the sensor when seen from the direction (θ, ϕ) , approximating the sensor to be a two dimensional object. Hence, such a detector will only be subjected to half the particle flow compared to a detector with a horizontal sensor plane.

3.5 Desired Performance and Derived Requirements

In order for the Micro Vertex Detector (MVD) to reach the desired performance for the intended physics cases, a secondary vertex precision along the z-axis of $\sigma Sv_z \approx 70 \mu\text{m}$ (for particles with $p > 1 \text{ GeV} c^{-1}$) is required [53, sec. 7.4.2] [54, sec. 4.1.1].

Most of the subsequent requirements result from this figure of merit and were determined in simulations. Upper limits for the material budget of the first station, for example, should not exceed $0.3\% X_0$, which is in accordance with eq. (3.3). For the following stations $0.5\% X_0$ is the desired upper limit [54, sec.4.2.5.1]. The low material budget is also important for a high precision and accuracy of the measurements carried out by the subsequent detectors.

A quick overview over the most important requirements for the MVD and its sensor is provided below:

- Collision rate capability of: 100 kHz for A-A; 10 MHz for p-A
- Position resolution (sensor): $5 \mu\text{m}$ [55]
- Hit rate capability (sensor): 7×10^5 (peak); 1.5×10^5 (average) [$\text{mm}^{-2} \text{s}^{-1}$] [55]
- Material budget: $0.3\% X_0$ (1st station); $0.5\% X_0$ (following stations) [54]
- Time resolution: $10 \mu\text{s}$

While some requirements are direct results of the requested detection performance, others choices were made due to constraints imposed by technology available at that point in time, e.g. the time resolution.

3.6 The MIMOSIS Sensor Family

The operation of the Micro Vertex Detector requires very specialised sensors. The unique combination of required hit rates, radiation tolerance, material budget and more cannot be fulfilled by any other sensor currently used by other physics experiments. Therefore, a custom MIMOSIS chip is being developed to specifically meet the detailed sensor specifications for the MVD listed in the (internal) specifications document [55] as well in the TDR (Technical Design Report) of the MVD [54, sec. 4.1.1].

Compared to sensors specifically designed for collider experiments, such as ALPIDE (for ALICE), the MIMOSIS sensors are designed to have particularly outstanding properties that are specific for use in a high-rate, fixed-target experiment. In short, these are the ability to cope with a highly inhomogeneous hit density across its

sensitive area, a very high radiation tolerance with respect to ionizing ($\sim 5 \text{ MRad}$) and non-ionizing ($\sim 7 \times 10^{13} \text{ n}_{\text{eq}}/\text{cm}^2$) radiation [54].

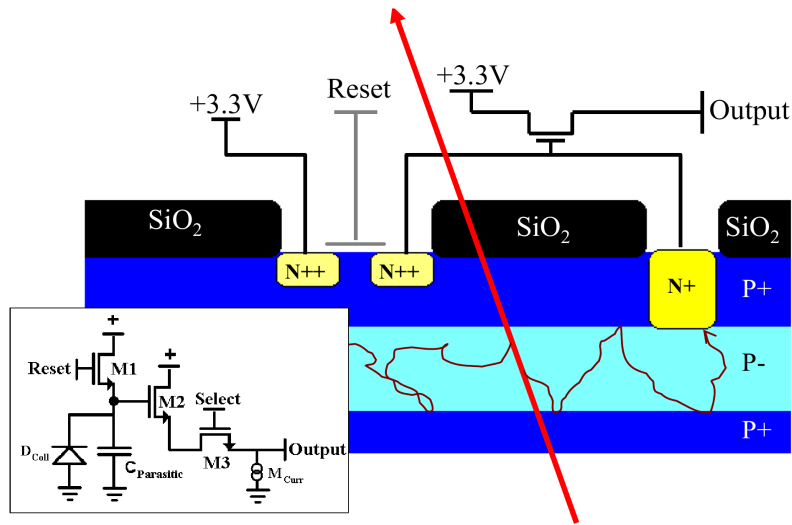


Figure 3.5: Working principle of a simple CMOS MAPS sensor. Taken from [54].

As illustrated in fig. 3.5, a silicon pixel sensor implemented in CMOS technology collects the charges freed by traversing particles on their way through the epitaxial layer of the semi-conductor silicon crystal. This is the general working principle of the MIMOSIS pixels and also its MIMOSA predecessors. While earlier sensors relied on the (statistical/thermal) diffusion of the charges to reach the charge-collecting diode, the MIMOSIS sensors aim to (fully) deplete the collection volume, thereby letting the charges drift to the diode in the resulting electric field. While such depleted pixels will likely improve the radiation tolerance of the sensor, they could result in lower amounts of charge sharing between pixels, thus reducing the effective spatial resolution of the sensor [54].

Electrical connections to the sensor are made with wire-bonding to pads at the edges of the die. Essentially, the sensors need stable supply voltages (1.8V for MIMOSIS) for the analogue and digital parts of their on-chip circuits. In addition, an external clock signal drives the clocked internal components and keeps all sensors in synchronization. The sensor needs to be configured using the I2C (Inter-Integrated Circuit) communication bus, which provides access to many essential parameters like threshold values or features such as injecting a defined charge into the pixels for noise scans. Finally, up to 8 differential data links of 320 Mbits^{-1} send the data to the DAQ (Data Acquisition System) system downstream. The number of data links is configurable as only the sensors closest to the target and in a particular position with respect to the magnetic field are expected to need the full bandwidth to transmit all hits. The other sensors can be operated with a smaller amount of data links, which in turn reduces cost and engineering

efforts, as the DAQ network is less complex then and as the lower power consumption on-chip will ease the thermal management of the detector. The external connectivity of the sensors (i.e. the detector readout) is further presented in [54, sec. 4.4].

The MIMOSIS sensors will comprise at least three subsequent sensors developments. The first one named *MIMOSIS-0* is a smaller scale sensor allowing to evaluate the pixel design and exploring different coupling schemes (DC vs. AC). The following *MIMOSIS-1* chip is a full-size chip with I2C control instead of JTAG (Joint Test Action Group) control used on all previous designs. In addition, it contains the large-scale digital processing part such as an elastic buffer to cope with occupancy fluctuations. The *MIMOSIS-2* sensor chip is planned to be the final chip to equip the MVD.

The development of the custom MIMOSIS series was preceded by several chips of the MIMOSA series. Those chips were also developed by IPHC (Institut Pluridisciplinaire Hubert Curien) and many of them with tested and refined with contributions from IKF (Institut für Kernphysik). One particular chip of that series was used on a prototype of the MVD and for the work carried out within this thesis: MIMOSA-26, described in detail in section 4.3.

3.7 Mechanical Design

Essentially, the MVD – as of today – is planned to include four detector planes, each equipped with a number of pixel sensors arranged around the beam axis on the front and back sides of a carrier material. The entire detector has to run in vacuum to avoid an entry window in front of the first station and to allow for placement close to the beam axis, a requirement coming from the phase space the detector needs to cover in its foreseen physics cases.

The mechanical design of the MVD was to large extents done and described by [35]. Figure 3.6 shows a quick overview of the mechanical design planning at that state.

The pixel sensors to be used belong to a sensor family named MIMOSIS specifically developed for the CBM experiment and is introduced in section 3.6. To compensate for radiation damage occurring during operation, the sensors must be kept at a low temperature to maintain their performance with increasing damage. The intended sensor temperature during operation is $-20\text{ }^{\circ}\text{C}$ to $0\text{ }^{\circ}\text{C}$, depending on further measurements of their radiation hardness and on their individual radiation dose/position in the detector.

The sensors on the station closest to the target will be glued to pCVD diamond

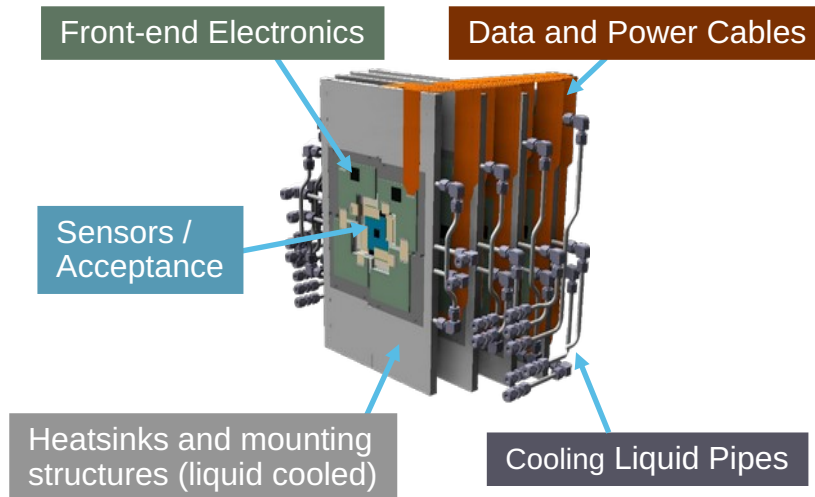


Figure 3.6: MVD Mechanical design of the MVD. Adapted from [35].

carrier plates. While diamond provides an excellent thermal conductivity allowing to cool the sensors via heat conduction, they are also good electric insulators, thus avoiding the danger of introducing undesirable electrically connecting connections between the sensors and the detector's components via the carrier plates. They also provide enough mechanical stability for a $150\ \mu\text{m}$ thin layer to safely carry the sensors while contributing only $0.12\% X_0$ to the material budget (equivalent to $116\ \mu\text{m}$ of Si).

As industrial diamond is very expensive and in addition can only be produced up to a limited size, the sensor on the larger stations will be mounted to TPG (Thermal Pyrolytic Graphite) carrier plates instead. Just like diamond, this material consists only of carbon. In the case of TPG, the carbon is arranged in graphene-like layers. It is more fragile than diamond, which is compensated in the design of the MVD by an increased designated thickness of $\sim 380\ \mu\text{m}$. Contrary to pCVD diamond, TPG is electrically conductive, asking for strategies to insulate it against any potential of the sensors glued on top. A possible solution to this problem might be coating of the TPG carriers with an insulation layer of parylene before gluing the sensors [54, p. 59].

Outside the geometrical acceptance of the detector, the carrier plates are fixed by solid aluminium heatsinks with embedded channels for liquid cooling. The heatsinks serve multiple purposes like firmly mounting the carrier plates in a defined location, removing the waste heat of the sensors and carrying the PCBs (printed circuit boards) of any front-end electronics required close to the sensors.

All heatsinks are mounted to other aluminium structures, forming two stiff detector halves of the MVD. Being mounted on linear stages, the detector halves can

each be moved 5 cm away from the beam to a parking position during commissioning and beam tuning.

Further mechanical designs of the MVD, which were created subsequently for various purposes, are described and in a CBM Technical Note [56]. An additional note was published detailing the keep-out volumes within the CBM experiment required to realize the MVD [58, 57].

Further details about the planned mechanical and electrical integration of the MVD can be found in [54, sec. 4.3.1].

Comparison to similar sensors used for other experiments:

The ITS Detector for the ALICE Experiment at CERN — The ALICE ITS upgrade [60, 59] to be installed in 2021 uses a sensor called ALPIDE, developed by a collaboration of CERN and IPHC. The sensor currently in development for the MVD – MIMOSIS – will in parts be based on ALPIDE’s internal architecture. Among the shared features are in pixel discrimination and a priority encoder circuit to determine the address of hit pixels.

The PXL Detector of the STAR Experiment at LBNL — The PXL detector [61] of the collider experiment STAR (Solenoidal Tracker at RHIC)¹ is part of its HFT (Heavy Flavor Tracker) and uses a MAPS sensors based on the MIMOSA family called “ULTIMATE-2” since 2014. It shares many properties with MIMOSA-26, the sensor used for the MVD prototype PRESTO, introduced in the next chapter.

3.8 Evaluation of the Material Budget based on Simulation Geometries

As mentioned in section 3.5, the MVD is highly optimized to exhibit a very low material budget, with the aim a) to minimize multiple scattering and b) to minimize secondary particle production in the detector material, both of which are a function of the radiation length X_0 . As multiple scattering changes the path of traversing particles, it has a detrimental effect on the secondary vertex precision, cf. eqs. (3.3) and (3.4). To achieve a small overall material budget of the detector, materials with a large radiation length X_0 (i.e. low atomic number) eq. (3.5) need to be used and they must be kept as thin as possible.

In order to run Monte-Carlo-Simulations of the CBM experiment, models of the individual detector geometries are necessary, describing the detector in a file format understood by the CbmRoot software framework [62]. For meaningful results, the

¹While STAR is mostly a collider experiment, there is also a fixed-target mode of operation, named ‘STAR FXT’.

particle transport through the detector material in CbmRoot needs to be both, fast and realistic. It is implemented with the GEANT framework and complemented by a digitizer software component, modelling the response to incident particles specifically for the MVD and its sensor type [63]. For meaningful simulation results, the detector geometries representing the actual composition and structure of the detector's materials need to be modelled accurately while keeping its complexity as low as possible (e.g. small number of components/volumes) for fast computational results.

For this thesis, the MVD geometries were generated using a ROOT macro taking advantage of the many geometrical symmetries the current MVD design exhibits. A CBM Computing Note was published documenting those new simulation geometries [64]. Over the years, there were multiple revisions of the MVD geometry. New revisions mostly refined implementation details of the MVD geometry. Furthermore, in 2017 the MVD geometry was split into two distinct configurations: "TR" optimized for tracking of low momentum particles and "VX" optimized for (secondary) vertexing. The tracking configuration has a station-to-station distance of only 4 cm (compared to 5 cm for the vertexing configuration). In addition, the first station will be located further downstream of the target (8 cm instead of 5 cm for the VX configuration). At the time of concluding this thesis, three revisions of the simulation geometries can currently be used within CbmRoot and are explained further in the Technical Note: v17a/b, v20a/c and v20b. Each of the revision exists in the mentioned TR and VX configuration.

In order to evaluate the expected material budget of the MVD for each detector station, massless test particles called ROOTinos (PDG – ID = 0) were propagated through the simulation geometries. By filling a two-dimensional histogram with the material budget for each station encountered, a map of the material as defined in the geometry can be created. The result is shown for the TR configuration of the MVD in fig. 3.7 comparing the three revisions of the simulation geometries (v17a/b, v20a/c and v20b), which are part of CbmRoot. The centre column shows the current baseline model v20{a,c}, relying on 380 μm TPG for all four stations. The lower total material budget of the left column (v17{a,b}) is the result of assumptions that pCVD diamond will be available and affordable in the size needed for the first two stations, while 300 μm thick TPG would suffice to provide enough material strength for the subsequent two stations. The geometry version v20b (right column) shows the resulting material budget for a fallback solution using stiffer 500 μm thick TPG carrier plates for all four stations. (Further details about the technical differences between those versions are documented in [64].) Each plot shows a two-dimensional representation of the material budget for the specific station. The dark red outer area represents the thick aluminium heat sinks, while the white hole in the centre is the opening for the beam. In the acceptance

3.8 Evaluation of the Material Budget based on Simulation Geometries

of the detector, a lot of structure can be seen from the rows of sensors on the front and back sides of the detector as well as from their FPC (Flexible Printed Circuit) cables. It can be concluded that for station 0/1 (module type b), the material budget target of $0.3\%X_0$ needs to be adjusted if TPG is to be used instead of diamond. For station 2/3 (module type c) the typical material budget will stay well below the targeted value of $0.5\%X_0$ and will exceed that only locally in small areas with overlapping components.

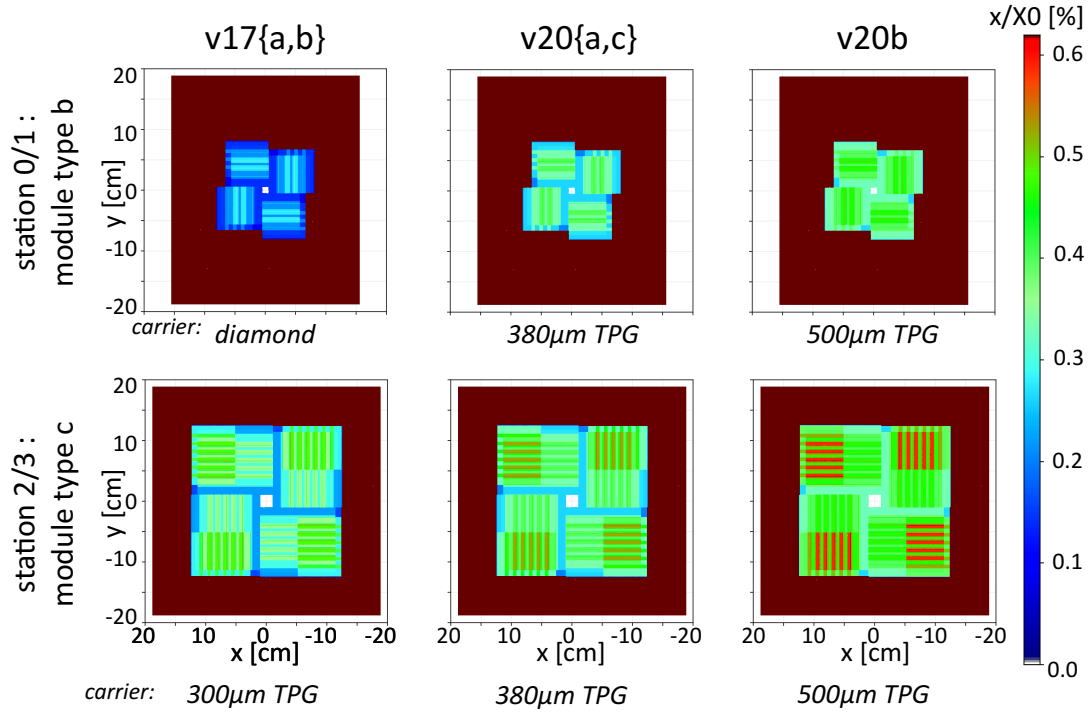


Figure 3.7: Material budget of the MVD detector geometry in its TR (tracking) configuration for the module types 'b' and 'c' of the three current geometry revisions. Taken from [64].

4 The MVD Prototype PRESTO

This chapter describes the prototype Prototype of the Second Station (PRESTO) of the Micro Vertex Detector (MVD). First, previous prototyping efforts are being briefly mentioned, followed by a detailed description of the current prototype PRESTO in section 4.2. Subsequently, further aspects of the prototype are discussed, such as its MAPS sensors, its assembly and mechanical setup, the external devices required to operate it as well as its electronics and data acquisition system. The setup created for its long-term operation in vacuum is discussed in section 4.5.

4.1 Previous Prototypes of the MVD

The development of a detector is a complex undertaking which usually requires several prototypes to demonstrate feasibility, test key aspects of new techniques and provide a platform for the control and readout systems to be developed. In the course of the development of the MVD, two prototypes were previously built to investigate various aspects. They were successfully tested in beam tests at SPS (Super Proton Synchrotron) at CERN (European Organization for Nuclear Research), the results were published and their evaluation was concluded:

- A first prototype called **MVD Demonstrator** (beam tested in 2009) [37, ch. 3] [65, 66, 67, 68]. It employed a total of 4 750 μm thick MIMOSA-20 sensors [37, p.23].
- A second one simply called **MVD Prototype** (beam tested in 2012) [54, sec. 5.2.1] [74, 69, 70, 71, 72, 73]. It hosted a total of 4 50 μm thin MIMOSA-26 sensors [35, pp.85].

In addition to those previous prototypes, the current prototype is called “PRESTO” hosting a total of 15 sensors. It is described in detail in the subsequent section 4.2.

4.2 The current Prototype: PRESTO

The research conducted for this thesis was carried out on a prototype of the MVD called Prototype of the Second Station (PRESTO). Reports of its purpose, produc-

4 The MVD Prototype PRESTO

tion and various results can be found in [75, 76, 77, 78]. It is the first prototype hosting more than four sensors on a single carrier plate. With 9 sensors on the front side and 6 sensors on the back side, it resembles a quadrant belonging to the second MVD station. PRESTO was built to verify that the mechanical integration could successfully be done with such a large number of sensors attached to both sides of a very thin carrier plate. The thin carrier and the double sided assembly result in an excellent overall material budget figure for the prototype. It shares key aspects with the foreseen final MVD design, such as its cooling concept, designated operation in vacuum, and the double sided integration of the sensors. There are also aspects where PRESTO differs from the current plans for the final MVD. This concerns the sensor used (type, dimension, frames per second, data interface, slow control), the sensor arrangement (number of rows and columns of sensors per station quadrant, total number of sensors), the DAQ (readout board), the carrier material (TPG thickness) and the FPC cables (which do have a coverlay in the case of PRESTO). In terms of the required control aspects it shares a lot of similarities with the foreseen final MVD.

PRESTO itself consists of the TPG carrier plate with sensors and cables glued to both sides. The sensors are wire-bonded to custom FPC cables [79]. Figure 4.1 shows the object integrated into an aluminium mounting frame which also serves as liquid-cooled heatsink. At the centre of the figure, the front-side of PRESTO is seen, with a 3x3 arrangement of MIMOSA-26 sensors; the two lower rows of sensors are hidden behind the orange FPC cables. The sensors and cables are glued to a 500 μm TPG plate of which only a small grey portion is visible just right of the top right sensor. Further six sensors are glued to the back side of the object so that PRESTO is equipped with a total of 15 sensors. The carrier plate is mounted to an aluminium heatsink (silver) with hidden cooling channels. A custom 3d-printed bracket (bottom black element) allows to mount the heat sink to the base table. The black pole on the left is a strain relief for cables going to the left. The large black clamp on the heat sink is a strain relief for cables going to the right. Two black clips on the heat sink help as mounting structures for Pt100 temperature sensors. At the bottom, an optical table with a 25 mm used to move all components into the vacuum chamber at once.

The read-out of PRESTO is realized with the TRB3 FPGA platform initially developed for the HADES experiment [80, p.108], [81]. The board contains five FPGAs from *Lattice*¹. In the case of PRESTO, it was adapted for the readout of the MIMOSA-26 sensors. The read-out chain used for PRESTO is described in detail in [14] and also introduced in section 4.8.

¹<https://latticesemi.com>

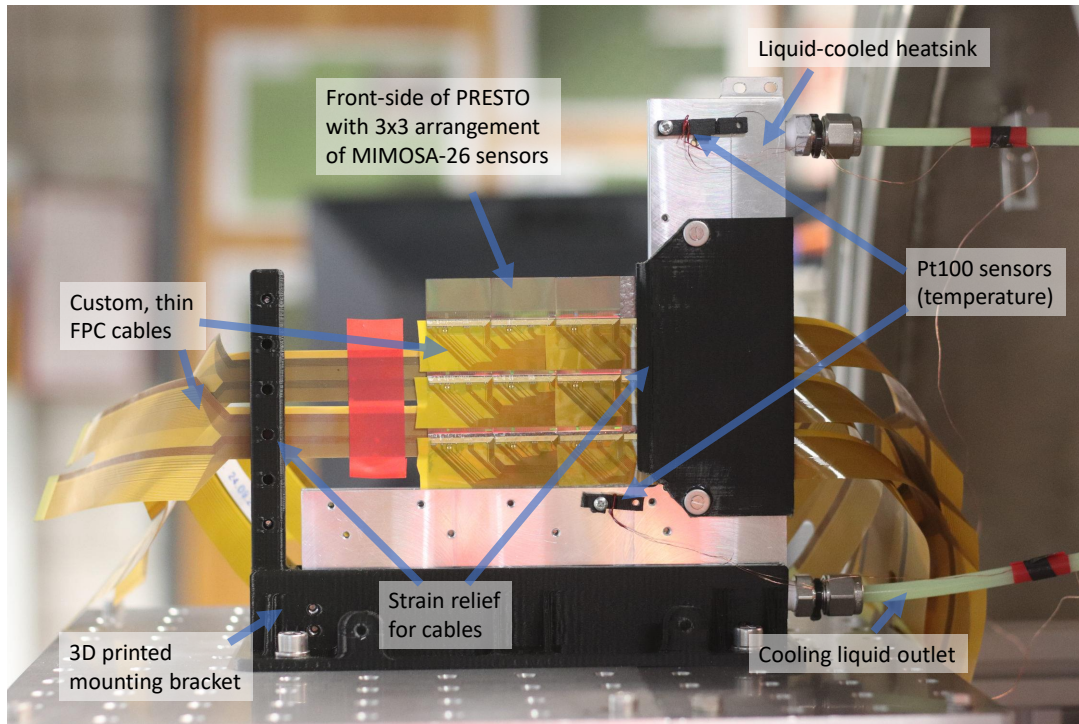


Figure 4.1: The prototype PRESTO.

4.3 Sensor Used: MIMOSA-26

As presented in section 3.6, the final sensor planned for the MVD will be a custom sensor belonging to the MIMOSIS sensor family. As those sensors are still in development and weren't available when the prototype PRESTO was conceived, it was equipped with an earlier MAPS sensor named MIMOSA-26. It is part of a long series of MIMOSA sensors developed at IPHC and was introduced in 2008 [82, 83]. Various research activities employed the chip since then. Most prominently, it is in continued use for the EUDET beam telescope at DESY (Deutsches Elektronen-Synchrotron) [42, p. 14] [85, 86, 84]. A variant of that sensor called Ultimate-2 was used to equip the PXL detector of the HFT (Heavy Flavor Tracker) of the STAR experiment; its in-chip reference voltage generation reduced the susceptibility for external noise compared to MIMOSA-26. The process used to manufacture the MIMOSA-26 sensors is AMS 0.35 OPTO². The chip is described in detail in [35, p.90], [87] and [88, p.12].

As MIMOSA-26 was already used for the previous prototype, its features were well understood due to the multi-year experience running the sensor, including its precision, fake hit rate and SNR (signal to noise ratio). In addition, the required data acquisition chain was already existing [87] but still received major improvements

²<http://www.iphc.cnrs.fr/List-of-MIMOSA-chips.html>

in [88]. MIMOSA-26 was also available in the amounts necessary to equip a larger prototype hosting 15 sensors.

While the performance of the final MIMOSIS sensor will mark a significant upgrade over MIMOSA-26, key aspects tested in this thesis – the ability to control and monitor the performance of a detector employing this sensor type in vacuum over a longer period of time – is independent of those changes.

4.4 Prototype Assembly – Sensors and Carrier

The production of the back side (1st) and front side (2nd) of PRESTO and subsequent testing is described in [75, 76, 77, 78]. Figure 4.2 shows the assembly of the sensors and FPC cables on the front side, after the back side was already equipped and fully tested. As a result of the mechanical prototyping effort, experience was gained on how to place and glue the sensors and cables reliably, how to wire-bond from the thinned sensors to the FPC cables and how to rework the module, if a sensor is not fully functional after the integration. In the latter case, a replacement sensor had been successfully glued on top of the defective one after removing all bonds.

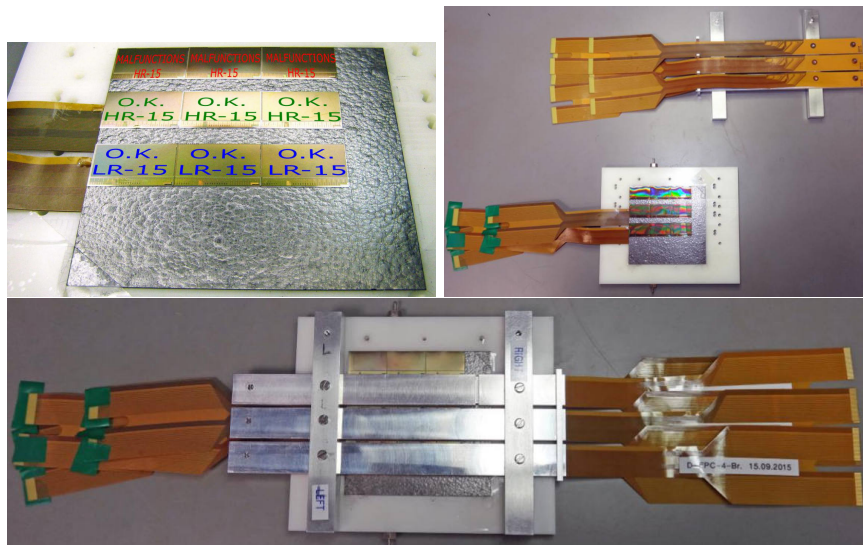


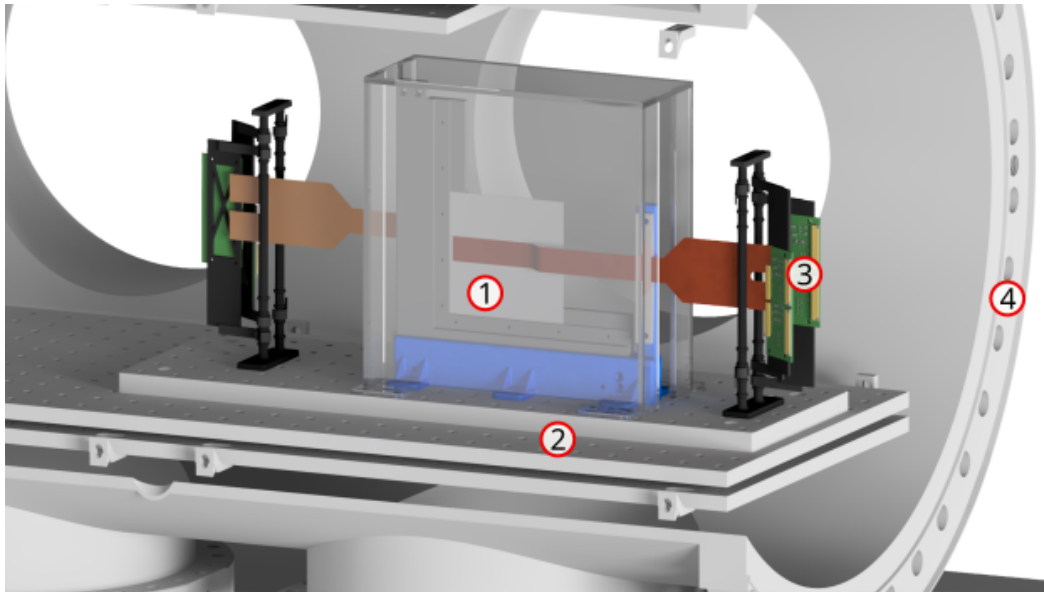
Figure 4.2: Production of the front-side of PRESTO. Taken from [77].

The assembly took place on a 500 μm thick TPG carrier plate for the thermal management and the mechanical stability. The temperature difference between the hottest and the coldest point of the carrier plate during operation of the sensors in vacuum will be in the range of 5 $^{\circ}\text{C}$ and 10 $^{\circ}\text{C}$, depending on the coolant mass flow [35, 89, 90]

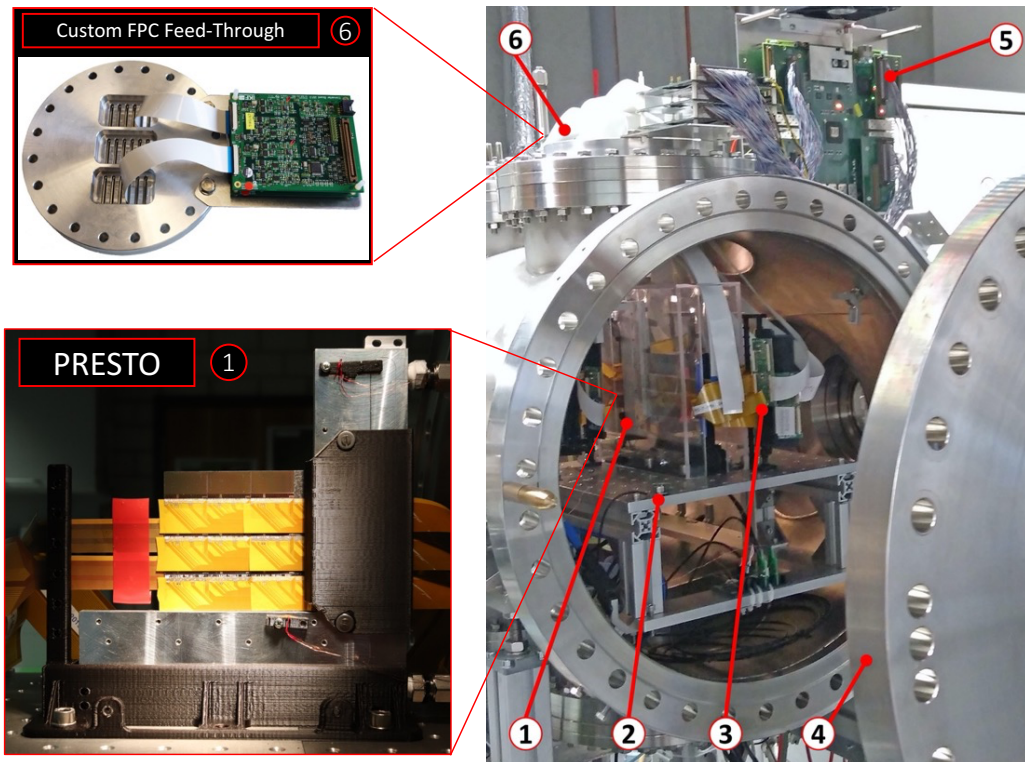
4.5 Mechanical Setup of the Prototype

To run PRESTO continuously (24/7) in the laboratory, its mechanical design had to be planned carefully to be able to mount the prototype safely in the vacuum chamber, avoid vibrations and allow for cable strain relief. The entire setup was first designed in a mechanical CAD (Computer Aided Design) system³ to validate the planning. A very large and flexible tubular vacuum vessel with flanges for access to the inside from all directions was tailored to the needs of MVD prototype tests. The prototype itself is mounted to an optical table, which can be slid into the chamber on a structural frame, allowing to commission the prototype setup with ease. Figure 4.3a shows the CAD design of the vacuum chamber and mechanical structures hosting the prototype tested in this thesis. The black stands are 3d-printed holders for the front-end boards (green), the transparent case in the centre is an acrylic glass to protect the prototype during installation and while pumping. Inside the acrylic case, the heatsink (grey) is mounted to the table using a 3d-printed structure (blue). Figure 4.3b shows annotated photos of all components inside the vacuum chamber with all supply lines connected.

³Autodesk Inventor Professional



(a) Rendering of the CAD planning to operate PRESTO in the vacuum vessel.



(b) Photos showing PRESTO itself, the custom feedthrough for FPC cables and how all the components are mounted in/on the vacuum chamber.

Figure 4.3: Planning and implementation of PRESTO's operation in the vacuum chamber. ① Prototype PRESTO inside of a protective box, ② movable table supporting PRESTO, ③ front-end-boards, ④ main flange, ⑤ Version 3 of the TRB platform (TRB3) based readout, ⑥ feed-through flange for FPCs

4.6 Major Devices for the Operation of the Prototype

Many devices across multiple categories need to work together seamlessly in order to provide reliable detector operation. To have access to vital parameters of these components and to changes their operational modes or settings, a digital interface is needed for each of them to connect them to the overall consolidated detector system. Some devices are equipped with suitable digital communication interfaces while others have analogue outputs providing access to required readings. Table 4.1 lists the device necessary for the operation of PRESTO. For all of these devices, support in the control system EPICS had to be created within the context of this thesis. (For some devices, such as the R&S HMC 8043 and the Agilent 34411A, EPICS support was added but the devices are not in use to operate the prototype in the current state of the system.) A challenge was the diversity of all these devices. Besides the different types of interfaces available on each devices, the communication protocol of each device was also mostly unique and needed to be investigated. An example for the effort necessary to add EPICS-support to a specific device, see section 5.10. Figure 4.4 shows a photos of each of the devices currently in use.

Category	Vendor & Product	Type of Interface
Bench Power Supply	R&S HMP 4030 / 4040	USB (FTDI)
Bench Power Supply	R&S HMC 8043	USBTMC / VXI-11
LV Power Distribution	2-ch 5V/5A "Power5" (IKF development)	USB Serial
Cooling System	Huber CC-405	RS-232
Vacuum Gauge Controller	Vacom MVC-3	RS-232
Vacuum Gauge Controller	Balzers PKG 020	Analog Vout
Vacuum Gauge	MKS 910 DualTrans	RS232
General Purpose ADC	ADCuC Board (IKF development)	USB Serial
Cooling: Flow Meter	Kobold DPM (ATmega 32u4 uC)	USB Serial
Temperature Sensors	Pt100 IKF 8ch board (ATmega 328p)	USB Serial
Humidity Sensor	Lufft OPUS 20	TCP/IP
6½ Digit Multimeter	Agilent 34411A	VXI-11
DAQ bridge	TRB3 / trbnet registers	Trbnet

Table 4.1: Devices added to the EPICS control system.

4 The MVD Prototype PRESTO

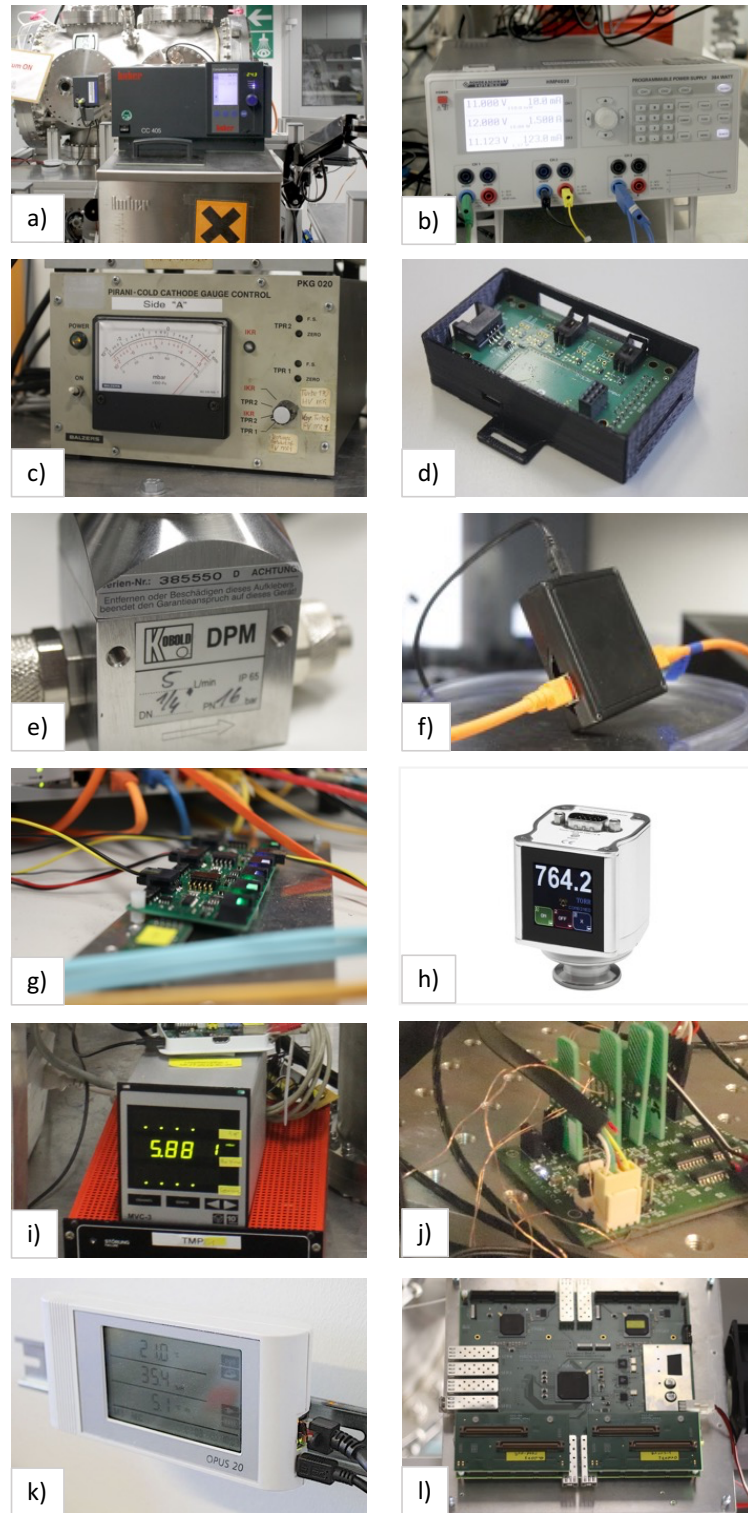


Figure 4.4: EPICS-controlled devices in setup: a) Huber CC-405 thermostat, b) R&S HMP4030 power supply, c) Leybold vacuum gauge controller, d) custom 16bit 4-channel ADC board, e) Kobold DPM flow meter, f) custom pulse frequency measurement used for the flow meter (ATmega 32u4), g) custom low voltage power distribution board, h) MKS 910 DualTrans vacuum gauge, i) Vacom MVC-3 vacuum gauge controller, j) custom 8-channel Pt100 temperature measurement board for use in vacuum, k) Lufft OPUS20 indoor climate logger l) TRB3 DAQ platform.

4.7 Vacuum System for PRESTO

The vacuum system of PRESTO consists of a large, mostly cylindrical chamber with a diameter of 450 mm and a length of 838 mm, i.e. a volume of $\sim 0.133 \text{ m}^3$. At both ends, DN 450 COF flanges allow for full access to the chamber. Eight further DN 250 CF flanges facilitate placing of viewports and electrical feed-throughs to top, bottom, left and right. Another four DN 40 CF flanges allow further equipment to be connected such as venting valves and vacuum gauges.

The chamber is being pumped by a classical rotary vane pump together with a Leybold turbomolecular pump. A gate valve between the turbomolecular pump and the chamber allows to decouple the pumping stage from the chamber volume for maintenance access or simple check for the system's tightness.

The vacuum typically reached with the chamber is in the order of 1×10^{-6} mbar if empty and pumped for a few hours. If equipment is placed inside with many virtual leaks (such as 3d printed mounts for example), the vacuum might deteriorate in the order of about two magnitudes. As the chamber has a large surface and cannot be baked, the vacuum is limited to those values but this is in accordance with the vacuum foreseen for the MVD [54].

4.7.1 Vacuum Feedthrough for Flat Flexible Cables

It was decided to mount the FEBs (Front-End Boards) inside the vacuum chamber and the CBs (Converter Boards) just outside the vacuum chamber. In consequence, the FFCs (Flat Flexible Cables) connecting the two boards had to be fed through a vacuum flange. Figure 4.5 shows the flange designed within this thesis to host 3×9 flat flex cables. After feeding all cables through and fixing them with the help of a 3d-printed holding structure from below, they were glued to the flange with Torr Seal, an EPOXY two-component glue known for its small outgassing rates. Once commissioned, the converter boards are mounted directly to that flange using a custom laser-cut mounting brackets to provide the necessary strain relief for the FFCs (Flat Flexible Cables).

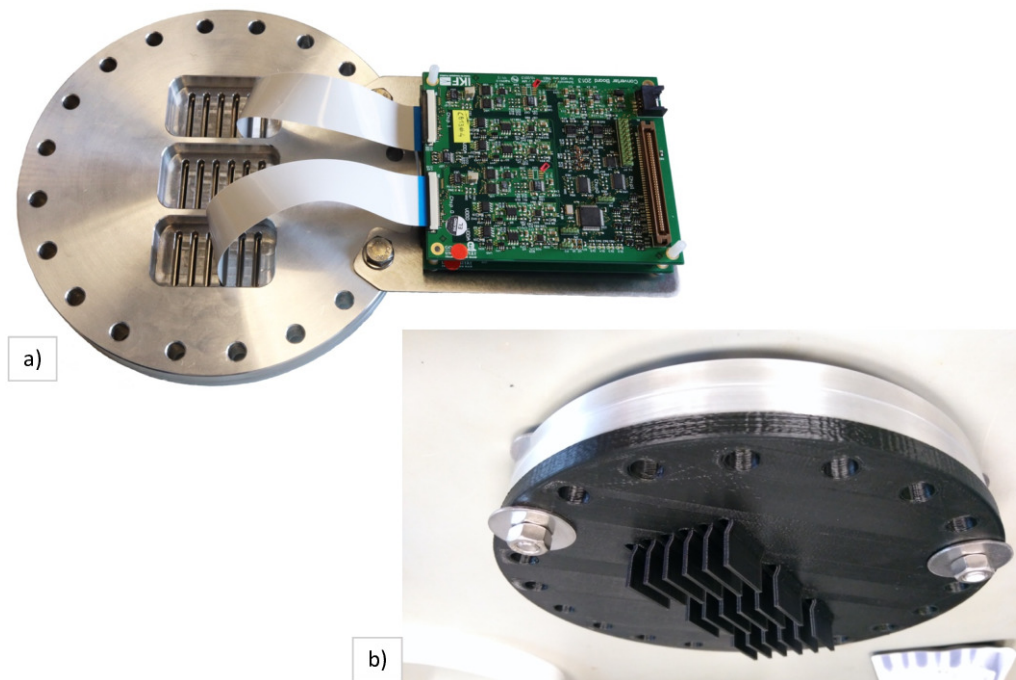
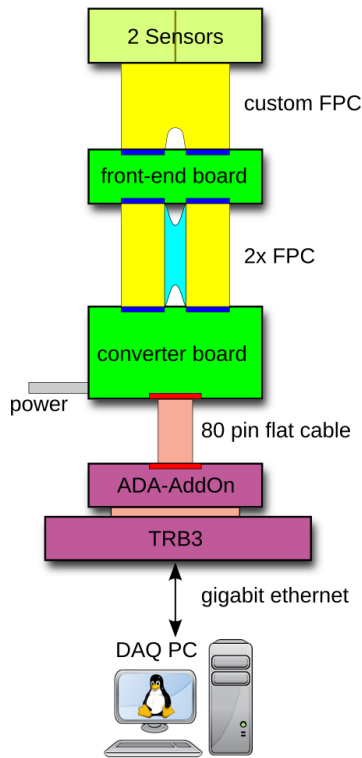


Figure 4.5: Flange production a) feedthrough flange with a converter board mounted to it. The image shows the flange before flat flex cables were glued into all 18 slots of the flange, b) 3d printed preparation tool to hold the flat flex cables in place while gluing them into slots machined into the vacuum feedthrough flange.

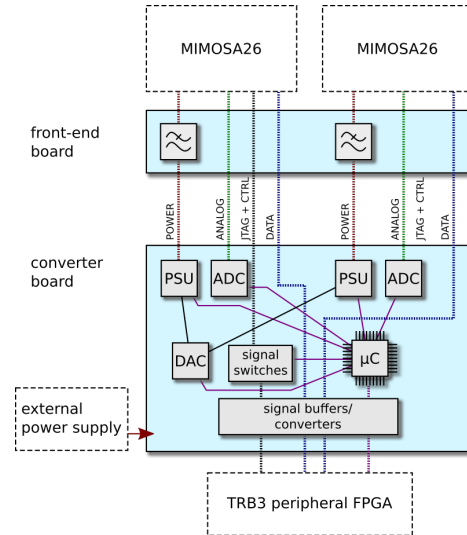
4.8 Electronics and Data Acquisition

The read-out of PRESTO is currently implemented with FPGAs (Field-Programmable Gate Arrays) on the basis of Version 3 of the TRB platform (TRB3) [81] as described in [14]. The FPGA code was ported from the previous TRB2-based read-out [87] to the newer TRB3 including a substantial redesign of the front-end electronics [88]. While the previous generation had a much more complex ring-based topology regarding the JTAG based configuration of the sensors, the new JTAG scheme closely follows a star like topology, largely reducing the cabling complexity. In addition, the debugging and monitoring capabilities built into the intermediate “converter boards” – cf. fig. 4.6b – are put to a whole new level with the abilities to monitor voltages and current consumption of the sensor, testing different schemes for reference voltage generation, reacting to over-currents (latch-ups) etc. [88].

Besides effective and reliable electronics, the connectivity plays a major role to operate the sensors, specifically the custom FPC cable as shown in fig. 4.6a. As the required FPCs will be located in the acceptance for all but the first detector stations, a custom cable with minimalistic layer thickness was designed, fabricated



(a) Schematic connection diagram of sensors with the TRB3 DAQ. Taken from [88, fig.6, p. 11].



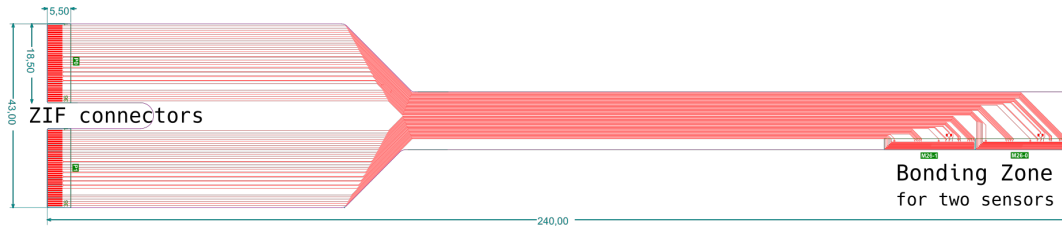
(b) Simplified block diagram of the front-end and convert boards used. Taken from [88, fig.10, p. 15].

Figure 4.6: Read-out chain of the TRB3-based DAQ of PRESTO equipped with MIMOSA-26 sensors.

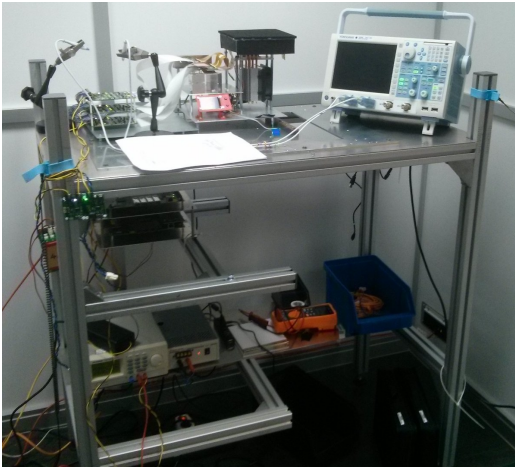
and its performance verified. The relevant scientific question for the verification was the resulting signal integrity for data transmission over the cable, as it was designed as a single layer circuit without a ground layer below the differential data lines. The signal integrity was verified for the longer version of two fabricated cable lengths by measuring an eye-diagram of the signal at the receiving end, see fig. 4.7c. The measurement allows for a realistic estimate of the expected signal integrity of the signal coming from the LVDS (Low-Voltage Differential Signaling) drivers of MIMOSA-26 after being transmitted through the long FPC cables. The resulting eye opening was measured to (11 ns, 470 mV). The rise and fall times were measured as $t_r = 1.7\text{ ns}$ and $t_f = 1.6\text{ ns}$. The measured parameters are satisfying and the identical cable is expected to meet the requirements for the twice as fast 320 Mbit/s (160 MHz) signals planned for the final MIMOSA sensors [55]. Further details regarding the ultra-thin FPC cables designed within the context of this thesis are to be found in [79, 91, 93, 92].

Figure 4.8 shows photos of the described laboratory setup focusing on three main regions: inside the vacuum chamber, converter boards and cables mounted to

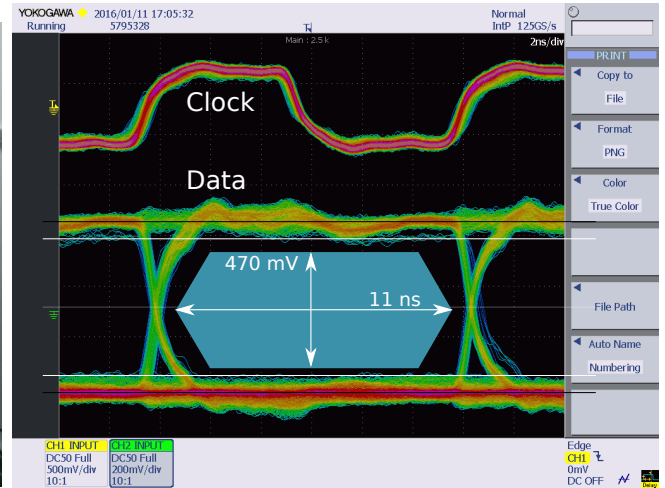
4 The MVD Prototype PRESTO



(a) Design of the new FPC cables. A photo showing the produced cable in the experiment is shown in fig. 4.3. The delicate question is the signal integrity of the unshielded single-layer cable over the distance of 240 mm.



(b) Test stand used to perform the measurement.



(c) Eye diagram of the MIMOSIS-26 data out D0 measured at the input of the LVDS buffer on the converter boards.

Figure 4.7: Revised ultra-thin FPC cables for MIMOSA-26 and PRESTO and verification of the signal integrity for differential 160 Mbit/s data transmission.

the top of the chamber and the TRB3 board used as read-out platform. PRESTO and the FEBs (Front-End Boards) are placed inside the vacuum chamber. The FEBs adapt the custom ultra-thin FPC cables to long and more robust standard FFCs, which are then connected through the feed-through flange on the top of the chamber to the converter boards. From there, the data finally reaches the TRB3 FPGA board mounted vertically next to the chamber.

A full overview of the DAQ system is provided in the subsequent chapter on the control system, see fig. 5.5. It contains a schematic overview of all major components necessary to operate PRESTO inside the vacuum chamber including the DAQ read-out chain.

A major contribution of this thesis is a full system integration of the prototype PRESTO into an operation framework allowing for 24/7 operation. This is a necessary transition going from a singular prototype object built and tested for for a short amount of time under laboratory conditions to a detector system that allows

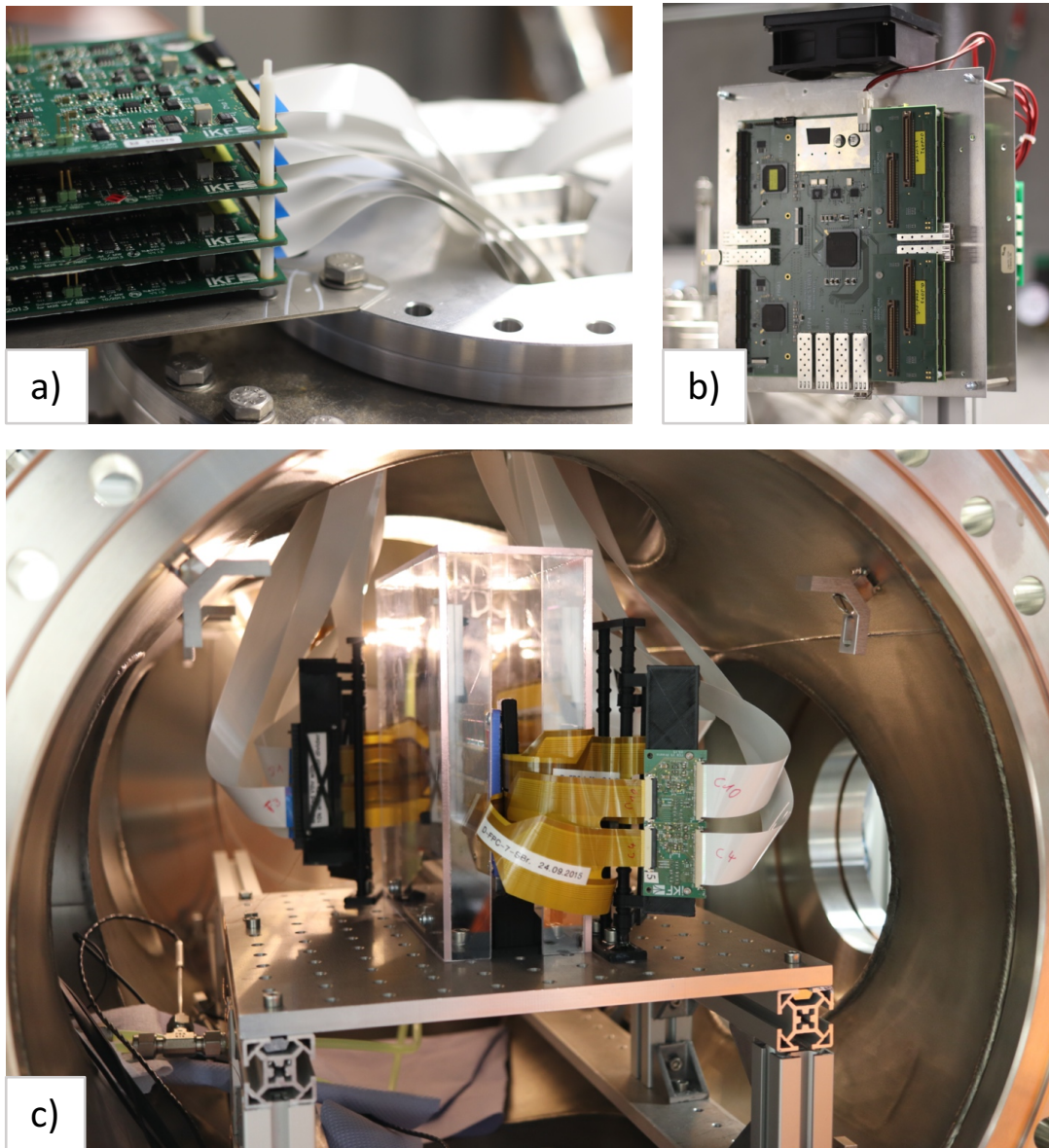


Figure 4.8: Photos of the setup with a focus on the visibility of cables and DAQ components: a) converter boards mounted on the feedthrough-flange connected to the FFC cables (white) going to the front-end boards, b) TRB3 boards used for the data acquisition, mounted next to the vacuum chamber, c) PRESTO inside the vacuum chamber, mounted onto an “optics table”. The sensors are fully connected with data, power and control lines on the custom ultra-thin custom FPC cables (orange). They are plugged into the FEBs and connect to the (white) industrial FFCs. Thermal sensors are connected and the cooling system is operational. A plexi-glass box protects the prototype during commissioning.

it to be used for reliable data taking over weeks in the context of the larger physics experiment. The missing link was a control system which provides consolidated

4 The MVD Prototype PRESTO

access to all major components necessary to operate the detector. The design and architecture of this control system is described in the following chapter, chapter 5.

5 Development of the Control System for PRESTO

A particle detector such as the MVD is a complex system of interacting devices, which need to be orchestrated for performant and reliable operation. This task is fulfilled by a so-called control system, similar to the ones required to run an industrial plant. Besides providing the required OPIs (Operator Interfaces) to operate the detector, a Detector Control System (DCS) needs to restrict the allowed operational parameter space for safe operation, e.g. refuse to power the sensors when they are not properly cooled.

The control system developed in the context of this thesis for PRESTO is based on the Experimental Physics and Industrial Control System (EPICS) and described in this section. The control system for PRESTO was first introduced in [94].

Apart from being used to operate the prototype PRESTO, the control system presented here and implemented in the context of this doctoral thesis shall also serve as a blueprint for the later MVD detector as it will be used in CBM. This is possible because the comprehensive operation of the prototype already requires controlling (i.e. monitoring, documenting, automating and regulating) the vast majority of technical aspects that will also be necessary in the operation of the later detector.

There are only a few aspects that will distinguish the operation of the full detector from the current prototype: PRESTO integrates 15 sensors, whereas the later MVD will use up to 288 sensors depending on the configuration, which will also have a significantly revised architecture (MIMOSIS instead of MIMOSA-26). The sensors will be operated with a lower voltage supply (1.8V instead of 3.3V) and a different configuration bus (I2C instead of JTAG). The data acquisition (DAQ) needed for operation will also be built with data concentrator ASICs (GBTx [95, 96]) instead of a FPGA network (TRB3).

On the other hand, there are many aspects that are identical or very similar in the prototype as in the final MVD: Operation is envisaged in the transitional range between fine and high vacuum. For this, the mechanical integration must allow to withstand operation in vacuum. In addition to the detection of the residual gas pressure in the vacuum with measuring cells, the monitoring of vacuum pumps (pre- and turbomolecular pumps) and valves is necessary for this. The sensors

are cooled to a target range of -20°C to 0°C with a liquid cooling medium that is tempered by an external cooling bath and whose cooling lines must be fed into the vacuum chamber without leaks. In the chamber, the coolant flows through specially manufactured aluminium heat sinks to dissipate the waste heat from the sensors, the ribbon cables and any front-end electronics. The parameters of the cooling circuit must be monitored, such as the volume flow, pressure and temperature, each for the supply and return. As a result, it must also be ensured that the sensors are actually cooled and do not overheat. If reliable calibratable temperature measuring diodes are available in the sensors, these are preferred, otherwise indirect measurements on the heat sinks must be used together with an exact recording of the electrically dissipated power.

For all of those aspects, the control system developed for the prototype PRESTO for this thesis and presented in this chapter can be transferred in large parts to the final MVD detector.

5.1 General Description of Detector Control Systems

Detector Control Systems (DCSs) are soft- and hardware systems that allow to operate complex equipment or infrastructure, such as industrial production plants or large-scale scientific experiments. They fulfil multiple purposes such as visualizing, monitoring, controlling (automated or manual), protocolling changes, archiving historic values and more.

Important aspects for reliable distributed detector control systems are:

- **Encapsulation and Decentralization**
Small, coherent units with limited purposes work together with other units collaboratively. This combines a separation of concerns (reducing complexity) with a mostly autonomous mode of operation (ensuring safe and deterministic system states in the case of partial outages or network partitions).
- **Fail-safe Operation**
Ensuring systems go to sane defaults, if any subcomponent fails. This is also a necessary prerequisite for the decentral/distributed units introduced just above.

Classification of Detector Control Systems vs. Distributed Control Systems In control systems and physics experiments, the abbreviation *DCS* is often in two different contexts. Sometimes it stands for ‘Distributed Control Systems’, a term long established for industrial control systems. It stands for control systems that are decentralized where each site has its own logic and an operational mode that allows to run independently in the case the other sites are inaccessible.

In this thesis, the acronym denotes Detector Control Systems (DCSs), although the system described is also implemented as a distributed system. While Detector Control Systems may be implemented in a distributed fashion, it is not a requirement in the definition of the term. After all, the difference between Distributed Control Systems and Detector Control Systems is mainly to be found in the areas of usage. While Distributed Control Systems often refers to the systems implemented to operate industrial plants (production, power, etc.), Detector Control Systems rather denotes the systems used to operate large physics experiments.

5.2 System of Choice: EPICS

The Experimental Physics and Industrial Control System (EPICS) is an open-source software with a custom free software licence [97] and was first described around 1991 in [98, 99, 100, 101]. A good source of information on the early days of EPICS can also be found in [102].

The system has a lot of advantages such as widespread adoption and a helpful community helping with issues.

Alternatives to implement such a control system include:

- Siemens Win CC (former PVSS)
- Tango,
- a custom software (e.g. based on MQTT¹ or Kafka²)

Table 5.1 lists a few different institutes and experiments together with their choice of control system. While LabVIEW [103] can also be used for control purposes, it is typically not used for larger experiments or facilities [104, p. 199].

Site	Experiment/System	Control System
SOLEIL, France	synchrotron	TANGO [106, 105]
SNS, USA	neutron source	EPICS [107]
GSI/FAIR, Germany	HADES	EPICS [108, 109]
CERN, Switzerland	ALICE	WinCC OA [110, 111]

Table 5.1: Table listing different research institutes (sites) together with the control system used in the different experiments (or site-globally)

¹MQTT is a publish/subscribe protocol.

²Apache Kafka is a stream-processing software for low-latency processing of real-time data.

5.3 Evaluation of EPICS from Various Viewpoints

Code Maturity is a criterion of huge importance for a critical application such as a Detector Control System (DCS). As the EPICS base software is now almost 30 years old and in widespread use, many programming bugs were found and the code can now be considered very stable. The big experience after adoption for various experiments, detectors and even industry has a huge impact on reliability, fail safety, robustness, fault tolerance and high availability.

Community Support is excellent, with exhaustive online documentation [112, 114, 113, 116, 115], the helpful tech-talk mailing list [117], conference proceedings (for example of the biennial International Conference on Accelerator and Large Experimental Physics Control Systems – ICALEPCS [118]) to the collaboration meetings twice a year [119]. Many contributions are keeping the software up-to date with newer developments. While the community support for EPICS is definitely good, the total userbase (in research and industry) is certainly smaller than that of the leading commercial competitor, Siemens Win CC.

Total Cost of Ownership is difficult to evaluate as the cost structure of free software is very different to commercial software. A system like Siemens Win CC would require the licensing of the software and its development could potentially be outsourced to an IT consulting firm. In the case of EPICS, there are no immediate license costs but the system needs to be maintained by highly trained personnel with the associated risk of knowledge drain. As CBM might run for decades, the long-term support is an important aspect too. Commercial solutions, however, could over time become more and more costly (as software might be deprecated or experts become scarce) and – in the worst case – stop working altogether. Here, free software such as EPICS marks a clear advantage with the availability of the source code and the permission to adapt it to future needs. If support for a given system is offered by the host institution this can also lower total cost of ownership significantly; this is the case for EPICS, where GSI/FAIR is also the control system of choice for the current HADES experiment.

Ease of Use: The ease of use depends largely on the implementation of the user interfaces for the operators. The influence of the choice of software has a rather small influence.

Ease of Development: While EPICS has many advantages, the first steps in the system are not always as intuitive as in the case of other solutions, especially commercial ones. This might be due to the fragmentation of the system (typical for large free software projects) with multiple (partially competing) choices to implement the same functionality. After the initial steep learning curve to understand the fundamentals of an EPICS based control system, the various openly available

resources provide fast help for most problems.

Ease of Deployment: An EPICS software can have multiple dependencies on (sometimes specific versions of) optional EPICS modules. Installing them is not always evident and might take quite some time to get right. While this can slow down the development, it is also an issue in deployment, when setting up the software on a new node/machine. For this reason (as well as others), a considerable amount of effort went into the containerization of most EPICS software components. It is described in detail in section 5.7 and significantly eases deployment.

5.4 Hierarchy of DAQ and DCS

An important design decision when creating a detector is the question of the hierarchy of DAQ and DCS.

In an extreme scenario, the DCS controls the DAQ. In that case, it might provide the possibility to start and stop the data acquisition and the vital parameters such as occupancy, data rates etc. are all aggregated in the DCS.

In a different scenario, the two systems (DAQ and DCS) are fully independent and have to be started and operated individually. While this mode can be more accommodating for typical detector developers and operators, it may also lack the high integration desired for a complex measurement device.

In a third scenario, both, the DCS and the DAQ are controlled by yet another entity, the ECS (Experiment Control System). This seems to be the scenario the CBM experiment is currently preferring [120, fig. 2].

5.5 General Architecture of an EPICS Control System

An EPICS control system operates on IPv4 network (i.e. a broadcast domain) in the form of client and server applications. The **network protocol** is EPICS-specific and is either the CA (Channel Access) protocol, the newer PVA (PV Access) protocol or both in coexistence. Support for PVA is natively part of the newer EPICS7 release branch, although it can also be used with EPICS base 3.15 as an optional module. While all relevant functionality provided by the older CA protocol is still available in the new PVA protocol, some key features of the new PVA protocol break backwards compatibility. Most notably, any data using new structured (or composite) data types available with PVA isn't available via the older CA.

Therefore, an important decision for the implementation of an EPICS system is the *choice of network protocol*: CA (Channel Access), PVA (PV Access) or a coexistence

mode of both (where it's advisable to only use the backwards compatible feature set of PVA).

CA/PVA servers are the EPICS entities that hold the state of the system in PVs (Process Variables). The servers are called IOCs (Input/Output Controllers) if they are implemented with the so-called “database records” provided by EPICS. A typical IOC would be an application connecting the control system to physical devices (in event-driven or time-based approaches). Such server applications without hardware access are called SoftIOCs.

CA/PVA clients process and consume the information provided by the servers. They can fetch (“get”) and change (“put”) the current state or subscribe to (“monitor”) changes of the state. Clients can take various roles, such as providing OPIs (Operator Interfaces) GUIs (Graphical User Interfaces), archiving the system state and evolution, alarm handling, system automation based on live status information and more.

5.6 Standard and Custom EPICS Components used for PRESTO

When a control system is designed and developed based on EPICS, many components can be used “off-the-shelf”. As everything is open source and free software, alternative solutions for most of the components exist, too.

Therefore, an important task is to determine for each task if the off-the-shelf standard solution is the best choice or if alternative solutions should be preferred. For some components, such as archiving software, no standard or default EPICS solution exists so that the choice has to be made between multiple 3rd party solutions.

5.6.1 Standard Components used for PRESTO

This section lists all software components used for the EPICS control system which follow a MOTS (Modifiable Off-The-Shelf) approach. They require some modifications to fit them to the needs of the specific control system, but they are ready-made solutions overall.

Standard EPICS components are understood as those implemented classically using the EPICS base framework. EPICS base comprises the necessary tools and shared libraries to implement CA/PVA clients and servers.

Standard EPICS components used for PRESTO, either in ‘production’ (continued use for the prototype) or for performance & suitability ‘evaluation’:

- EPICS base 3.15 (production) used for many IOCs,
- EPICS base 7.0.4 (evaluation) to evaluate new PVA features,
- tools **caget** and **caput**,
- EPICS extensions: StreamDevice2, async, calc, sncseq.

Further standard components used for the system include:

- CSS RDB Archiver (production),
- Cassandra PV Archiver (evaluation),
- Archiver EPICS Archiver Appliance (evaluation),
- CSS (Control System Studio) ‘classic’ 4.5.0 (production)
- CSS ‘phoebus’ 4.6.3 (evaluation)

5.6.2 Custom EPICS Components used for PRESTO

In addition to the standard MOTS (Modifiable Off-The-Shelf) components, quite a few additional components were custom developed for the control system and they are introduced in this section.

Custom software components specifically developed for use in the context of the control system for PRESTO include:

- PyTrbNet - a (IOC-like) CA server providing a bridge between the TRB3-based DAQ and the DCS, see section 5.11.3,
- Alarms - using a dedicated IOC aggregating data from other IOCs via CA,
- Archiver Software - a POC (Proof of Concept) for archiving data as EPICS-native CA protocol datagrams (using caproto),
- Container based deployments - IOCs and archiver packaged in application containers with all necessary dependencies,
- Online Web Interface - a web GUI allowing to view the system status from just any web browser,
- Use of VLAN (Virtual Local Area Network) - allows to connect to multiple networks (DAQ and DCS) over a single network cable,
- spec-plot - creates plots of archived data based on plot configuration files in YAML format,

- carecplay - allows to record and play back (i.e. ‘mock’) the evolution of PVs (Process Variables).

Custom hardware developed for the DCS for PRESTO is listed below. Each of those hardware components is in addition also accompanied by a companion software component, an EPICS IOCs:

- 4-channel 16-bit ADC board,
- 8-channel Pt100 temperature sensor board,

In the following section, the contributions of this thesis to package EPICS components in containers are being presented.

5.7 Packaging EPICS Control System Components with Container Virtualization

Prior Work

The first mention of Container Virtualization in the context of the EPICS control system appeared in a Japanese contribution for the PASJ2015 meeting³ [121] and shortly afterwards in an email on the EPICS Tech-Talk mailing list [122].

Thereafter, publications such as [124, 123] more intensely studied the feasibility of using EPICS within (Docker) containers. Finally, [125] shows how a modern controls environment is embedded into container environments tying together all necessary components such as the backend, GUI, archiving, and more.

During the most recent EPICS collaboration meetings, more and more groups working on EPICS showed that they actively work towards modern software paradigms such as DevOps, agile project management, and container virtualization. See [126] at ICALEPCS2019.

In 2019 a dedicated workshop on containers in control applications took place [127] followed by a contribution at the EPICS collaboration meeting [128].

Porting EPICS to a Containerized Environment for the MVD

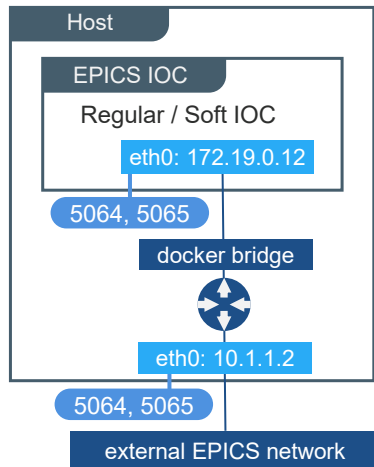
In the context of this thesis, multiple EPICS components were packaged and used in a containerized environment with Docker [129]. Those include the IOCs implemented with EPICS base, the CSS RDB Archiver and the CA Gateway. As containers are usually running in a virtual isolated network namespace, care has to be taken to connect them properly to the external EPICS network. A few options for this external connectivity are possible: Adding a custom IP address to

³The 12th Annual Meeting of the Japanese Accelerator Society

5.7 Packaging EPICS Control System Components with Container Virtualization

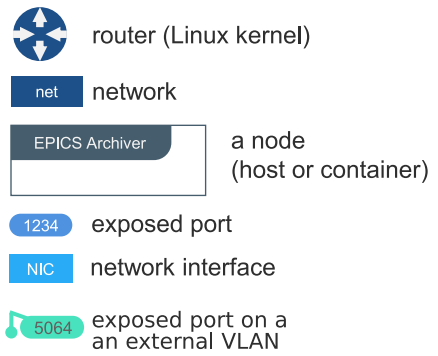
the container using 'macvlan', operating the container in the 'host' namespace or forwarding only the required ports from the host computer. The latter method was chosen to deploy most of the EPICS architecture for the MVD and PRESTO. The implementation details are depicted in fig. 5.1. Most IOCs were deployed as containers as shown in fig. 5.1a. The EPICS CSS Archiver was deployed together with the necessary database server on a separate host as shown in fig. 5.1c. The CA gateway application as shown in fig. 5.1d is not strictly necessary to operate PRESTO but it may be used when integrating the MVD DCS into the overall EPICS control system of the CBM experiment, if they should run on separate network segments, as depicted in fig. 5.2. All MVD related devices are in a single field network, a CA Gateway is the bridge towards the supervisory layer containing GUIs in the control room, access security, archiver software etc. In this context, the individual networks (10.0.0.0/24 etc.) would all be dedicated EPICS networks, running separately but being interconnected by the CA gateway instances. This network partitioning allows for an improved system reliability, where network issues such as for example so called broadcast storms would only affect a small and clearly separated part of the whole system.

5 Development of the Control System for PRESTO

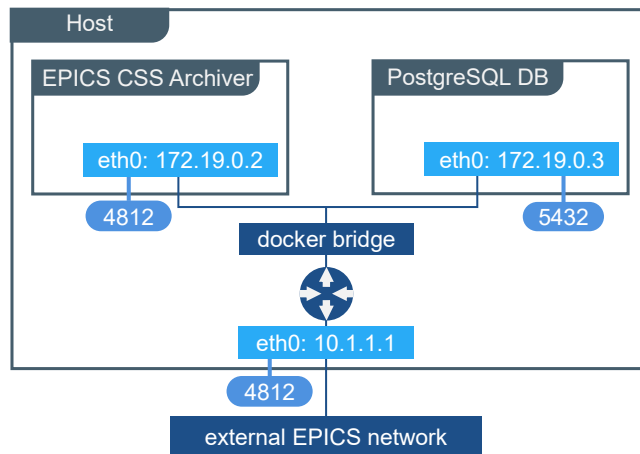


(a) A regular EPICS IOC deployed as a Docker container.

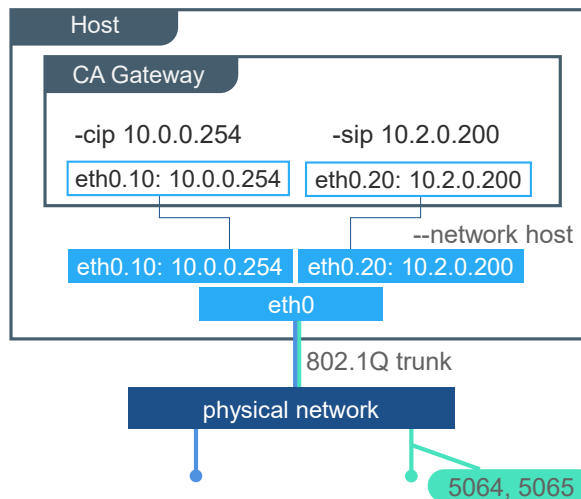
Legend



(b) Legend for the Docker architecture diagrams.



(c) Archiver together with its backend database PostgreSQL in two Docker containers tied together by virtualized networking.



(d) CA gateway in a Docker container. Here, the use of a 802.1Q VLAN trunk port allows for consolidated external network connectivity.

Figure 5.1: Architecture for the use of container virtualization with Docker [129] in the EPICS control system's software stack as implemented for the MVD prototype PRESTO.

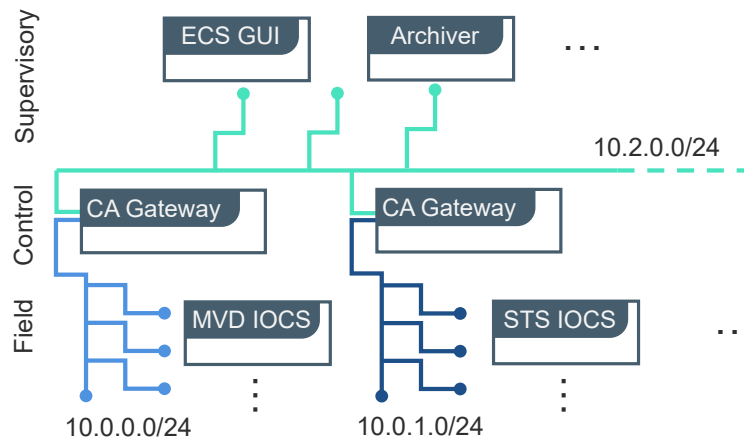


Figure 5.2: The network topology foreseen for the CBM experiment.

5.8 Testing the Control System with Simulated Hardware

Testing a control system is an important but delicate task. A major feature of a control system is preventing degenerate or critical states of operation. It is vital to test and validate the proper reaction of the control system to those critical states while at the same time not putting the actual hardware and detector at risk. Using software mock-ups of critical systems (instead of testing against the actual hardware) has proven very helpful during the development of those tests and validations.

In general, an EPICS system can be tested by starting mock-up IOCs that serve PVs (Process Variables) with names identical to the ones present in the real control system. In many cases, only a few such PVs representing the alarm state may be sufficient to replace the physical system. In the best case, all PVs in the control system can be simulated with a mockup IOC in order to test the full system without any hardware dependencies.

Ways to mock up PVs with virtual/soft IOCs used for this thesis include:

- **carecplay**, a software developed for this thesis [130],
- **softIoc**.

The approach using carecplay has proven very powerful, as it allowed to record live controls data from physical hardware, e.g. a cooling system currently performing a cooling cycle. The tool allows then to play back the series of recorded PVs later on without needing to power on the actual hardware (e.g. cooling system).

A simple softIoc, however, together with the Phoebus tool “PV Table” is very effective for interactive work, where the control system developer might need to test the reaction of the control system to a specific error state [131].

5.9 Computer Architecture for DAQ and DCS

Figure 5.3 shows the essential nodes (i.e. computers) involved in running the EPICS-based control system for PRESTO and its TRB3-based data acquisition system. Most IOCs run in containers on a Raspberry Pi 3 SBC (Single Board Computer) with ARM architecture, while all other tasks, such as archiving, and the DAQ tasks are performed on regular x86-64 PCs (Personal Computers).

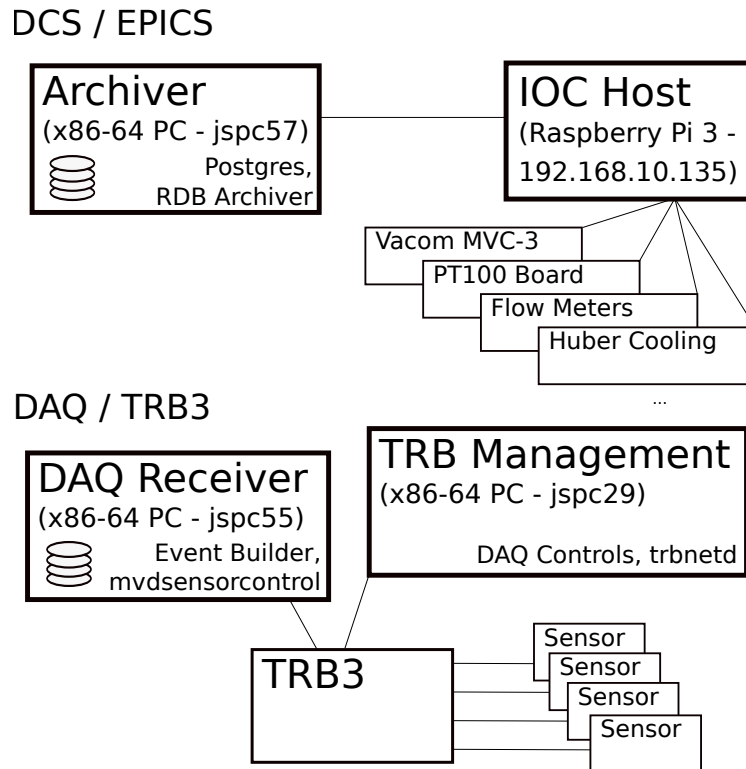


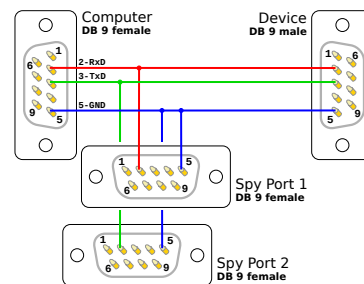
Figure 5.3: Computer architecture for the operation of the prototype PRESTO.

There are little interdependencies between the DAQ and the DCS in this architecture. Each of the two systems can continue to run on its own the case of incidents (like a power failure of one machine). There is one aspect, where the DCS depends on data provided by the DAQ though: Some EPICS PVs are requested from the TRB3 DAQ network, as introduced later in section 5.11.3. Those values are PVs are read only (r/o) within the EPICS DCS and those PVs shouldn't be used as necessary inputs for other EPICS records, e.g. to determine the overall detector state, to avoid a factual interdependence between the DAQ and the DCS. PRESTO does not feature a top-level overview layer (or ECS, see section 5.4) that would fuse the DAQ and DCS into a common system (e.g. to power the detector and start a new run of the data acquisition). Instead, two independent systems were used, one for each system (DAQ / DCS).

5.10 Implementation of Input/Output Controllers at the Example of the Huber CC-405 Cooling System



(a) Huber CC-405 cooling thermostat.



(b) A custom RS-232 cable allowing full-duplex sniffing of the host/device communication.

Figure 5.4: Cooling plant (Huber CC-405) and hardware used to reverse-engineer its communication protocols.

The cooling system used for PRESTO is a CC-405 thermostat with an internal heat bath by Peter Huber Kältemaschinenbau AG, see fig. 5.4a. It has an RS-232 connector allowing to monitor all vital parameters of the device and also providing control of the circulation pumps and temperature conditioning including the set-point temperature. A software called SpyControl and SpyLight is provided by the manufacturer to communicate with the device [132]. While a manual for the data communication via its custom “LAI” and “PP” protocols is provided by the vendor [133], some of the more elaborate commands of the LAI protocol are undocumented. In order to compensate for those shortcomings, reverse engineering techniques were employed in order to discover the commands missing in the documentation. To do so, a dedicated sniffing cable was placed in between the regular RS-232 communication path, see fig. 5.4b. The connectors in the figure are shown from the side of their solder tails. The communication between the software and the cooling device was recorded using `jpneulator.py` [134] using two additional RS-232 ports connected to a separate computer. While recording, the actions taken and the values displayed in the vendor’s software were noted down. This way, further datagrams could finally be reverse-engineered. To validate the findings, a small microcontroller board was programmed to be a drop-in mock-up device (imitating the cooling system) allowing to send forged messages to the vendor software. Using this mock-up device, the reverse-engineered know-

ledge has been validated by comparing the interpretation of the forged datagrams in the vendor's software to the assumed output.

This reverse-engineering effort was specific for the Huber CC-405 with its RS-232 serial port. For many other devices, the vendor's documentation is good enough to avoid the rather high effort of reverse-engineering the devices' communication protocols. A similar reverse engineering effort was only needed in the case of the laboratory temperature/humidity logger Lufft OPUS20, where the communication protocol built on top of the TCP (Transmission Control Protocol) was disentangled by sniffing the communication with the vendor software with the network packet dissecting software Wireshark [135].

5.11 Graphical User Interfaces

Graphical representations of quantitative information can potentially have much higher data density than textual representations [136, pp. 161]. That is a major reason why GUIs (Graphical User Interfaces) have such a central role in control systems. The data density in such GUIs can be very high, but the human input bandwidth for visual information is limited [137]. Therefore, a very important aspect when creating control systems is the thoughtful design which guides the users to any details that need attention or which show irregular behaviour. This is typically done with colours (e.g. green - OK, yellow - warning, red - alert). Care has to be taken to keep the balance between providing all necessary information without overloading the user.

5.11.1 Control System Studio and Phoebus

Control System Studio (also commonly called CS-Studio or simply CSS) is a Java-based software that provides a wide variety of GUI tools for the user-facing side of the control system [138, 139, 140]. It allows to create OPIs (Operator Interfaces) with ease and was selected as primary GUI to operate PRESTO.

The OPI GUIs were first set-up using CS-Studio 'classic' v4.5.0 and later transferred to the newer version CS-Studio 'Phoebus', an effort to modernize the CSS source code [141, 142].

The installation and configuration of the 'classic' variant can be found in [143, p. 36], including the device support for several basic devices used to operate PRESTO.

Figure 5.5 shows the overall system status view realized with Phoebus v4.6.3. The prototype – PRESTO – is placed inside the vacuum chamber and all connections to FEBs, cooling, temperature sensors, etc. are indicated. Buttons with an 'external

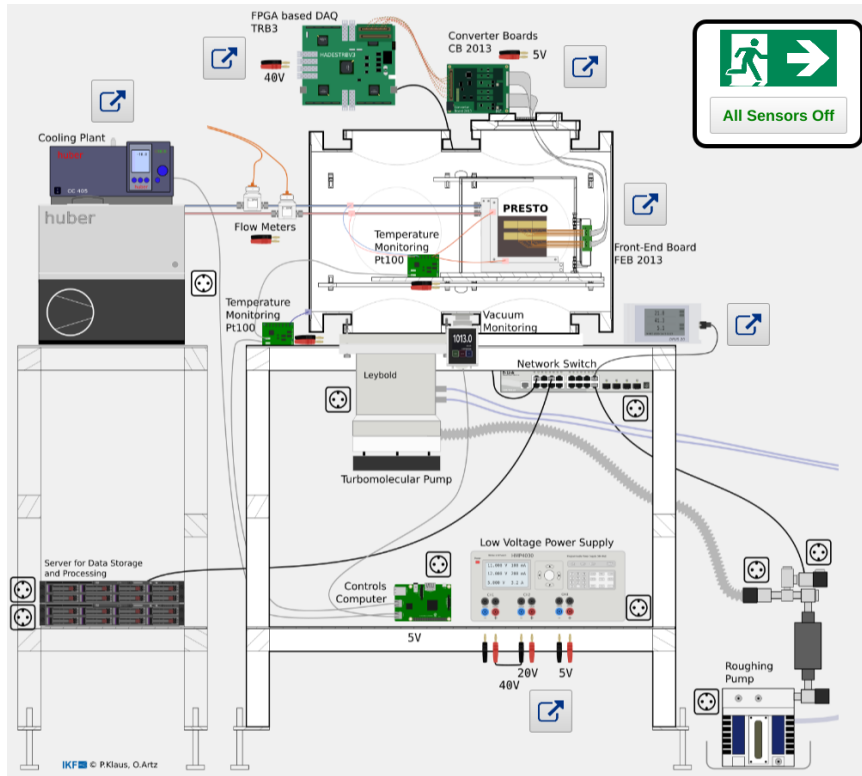


Figure 5.5: Overview GUI for the control system implemented with CSS Phoebus.

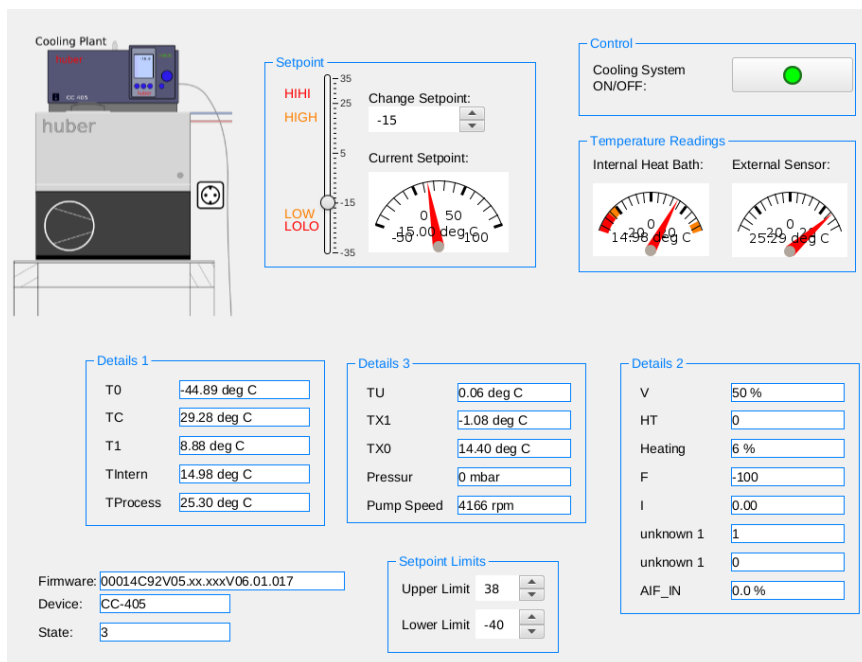


Figure 5.6: Details GUI for the Huber CC-405 cooling system implemented with CSS Phoebus.

link' icon lead to separate, detailed subsystem views, such as the one shown in fig. 5.6. The 230V Schuko icons symbolize devices requiring 230 V mains power and indicate the power status of respective devices. Toggling the power outlet allows hard resets of the essential devices in the rare case of a malfunction. Red and black banana plugs are used to indicate devices providing/requiring low voltage power. An emergency button allows to quickly switch off the sensors.

The detailed GUI for the cooling system as shown in fig. 5.6 allows to monitor and control most aspects of the cooling plant. The current temperature in the heat bath can is shown as well as the set-point temperature. Temperature control and the circulation pumps can be switched on and off. Three detail boxes grouped as 'Details 1/2/3' contain information available due to the reverse engineering of the device protocol, see section 5.10. This includes further internal temperature readings of the cooling plant (T0, TC, T1, TU, TX1, TX0), the pump speed, heating and cooling power. Most of these detailed parameters are not directly necessary to operate the plant, but archived data of those will prove useful for the technical support staff of the cooling system in case of malfunction.

5.11.2 Dashboard Website

In order to access the most important PVs from anywhere via a web browser, an interactive dashboard was created and published on [144]. Figure 5.7 shows a screenshot of the dashboard in action. Each box contains a single PV and its latest value. It also contains an interactive sparkline of the past 30 minutes: data points and their time stamps are shown when hovering the sparklines with the mouse pointer. The sparklines are inspired by [136, p.93] and [145, pp. 46], where they were first introduced as "data-intense, design-simple, word-sized graphics".

The principles guiding its technical design were to create a mobile-first, cross-platform, and cross-browser dashboard to be accessed from anywhere. Due to security concerns and threat mitigation, the dashboard was designed as a r/o system with its server running in separate network partition connected to the main EPICS system only via a Channel Access Gateway with restricted access (i.e. allowing r/o access only for the desired process variables).

For the implementation of the dashboard website, widespread standard web technology was used:

- Python 3 (programming language)
- Bottle v12 (micro web framework)
- JSON (configuration file)
- Twitter Bootstrap v3 (responsive GUI framework for websites)

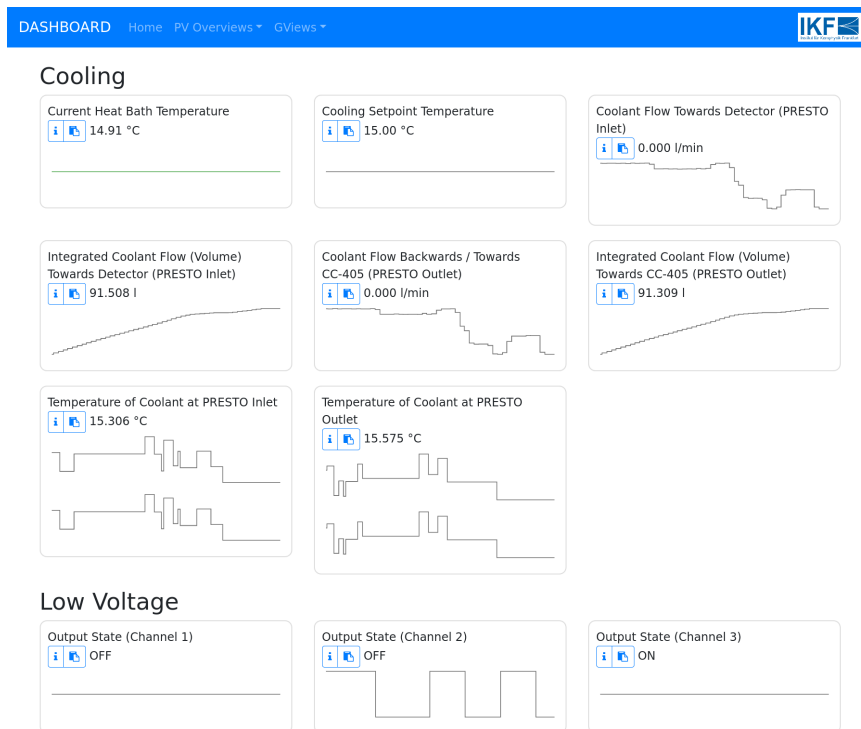


Figure 5.7: Dashboard website with an overview of essential PVs of the EPICS control system.

- d3.js v3 (visualization and plotting toolkit)
- SVG graphics (live updating illustrations of the experiment)

In some use cases, the web dashboard should have an even higher degree of system isolation for security reasons. One way to achieve this is to deprive the web dashboard application from live-access to the production EPICS system; instead, the web dashboard could read its data from a r/o copy of the archived data. This would come at the cost of an additional delay of data updates. Depending on the implementation, a delay of 0.5 s to 20 s seems realistic for such a scenario. While such a delay is too high for expert activities such as detector tuning, it could be acceptable if the site should rather serve as an Internet facing information service, for example in the context of a PR (public relations) campaign. The required bandwidth and processing power would roughly stay identical, as only the continuous flow of updated values would have to be appended to the archive data. Ideally, the web dashboard would run on a separate node connected to the internet. This node could be connected to EPICS via an optical fibre connection, where the direction from the web dashboard node back to the EPICS network would be physically disconnected/interrupted. This way, an unidirectional continuous data transfer from EPICS to the web dashboard would be possible – and the web dashboard node wouldn't have any access to the EPICS system if compromised.

5.11.3 PyTrbNet IOC - A Bridge between Data Acquisition and Control System

In nowadays' nuclear physics experiments, the data acquisition gets more and more complex. A big factor are increasing channel numbers⁴ and read-out rates over the course of the past decades. In addition, the integration of circuits is advancing; in the case of the CMOS pixel sensors used for the MVD, more and more electronics classically described as front-end electronics are not discrete external components anymore and instead embedded on-chip. As a consequence, a lot of metrics (voltages, currents, hit rates) and control settings (thresholds, frame integration time) are only accessible via the regular data acquisition data stream (and no longer via classical dedicated controls data paths). An important aspect of this thesis was therefore to connect values from the data acquisition system, currently based on the TRB3 platform to the EPICS-based control system.

Prior research and development was done on an EPICS IOC for TrbNet called devTrbnet (with an even earlier predecessor called trbnetIoc) [146]. This IOC can aggregate the register values of all responding TRB (Trigger and Readout Board) boards into a single EPICS waveform database record. This results in a low bandwidth, high performance transmission for applications requiring fast access to individual registers across large amounts of TRB boards. Therefore, the focus of devTrbnet is to provide access to individual registers for a large number of concurrent TRB boards. The shortcoming of the devTrbnet approach is that it a) cannot easily provide access to *all* registers provided by TrbNet and b) the waveform records complicate keeping track of individual TRBs and the time-evolution of its register values.

For this purpose, a software bridge from TRB3's TrbNet to EPICS was created. It is implemented in Python and is part of the software package PyTrbNet⁵ [147], a Python binding for TrbNet. PyTrbNet (including the bridge IOC to EPICS) was created as part of this thesis for the prototype PRESTO. When compared to the prior work on devTrbnet, the focus of the PyTrbNet bridge to EPICS is different, as it aims to make *all* register entities of each TRB board available to EPICS with a 1:1 mapping to a single PV name for each combination of TRB board address and entity name. Due to those design choices, much less configuration effort is required to set-up the IOC with the PyTrbNet version. At the same time, it comes at the cost of slightly less performance, being implemented in Python (i.e. a scripting language) and as multiple PVs have to be queried for a full set of data across all

⁴An example for increasing channel numbers is found in the slow shift from silicon-microstrip sensors towards pixel sensors, almost squaring the number of channels present in a system with comparable granularity.

⁵Registered on the Python Package Index at <https://pypi.org/project/trbnet/> and on Github at <https://github.com/pklaus/pytrbnet>.

TRBs.

The TrbNet to EPICS bridge IOC is to be found in the module `trbnet.epics.pcaspy_ioc`. While it was designed and implemented for PRESTO, it can be used with any TRB-based system. The specific implementation for the TRB3-based readout of PRESTO⁶ can serve as an example. The module allows to implement an EPICS IOC that provides access to TrbNet registers. For the latter, XML (Extensible Markup Language) database files exist, which describe ‘entities’ of the FPGA registers, comprising addresses, data format, name, description, possible repetitions, membership to a group of entities, etc. As the bridge IOC was implemented to parse those XML files, very little configuration is required to start it up: except specifying the XML database file to use and the desired update rate, the TrbNet addresses to send the requests to need to be provided, together with the addresses of the TRBs expected to respond⁷. Once started, the IOC will provide PVs for each of the desired entities from the XML file and for each of the specified TRBs.

For PRESTO (and previous MVD prototypes), specific XML database files exist describing the registers of the MVD-specific FPGA implementation, for example `Mvd.xml`⁸, which lists the registers of the *MVD Read-out Controller* FPGA block. Once, the bridge IOC is started, it serves PVs with names mapping to those entities. An example for such a PV name in the case of `Mvd.xml` is `CBM:MVD:TRB:Mvd-0xd010-DataLength.0`, where `CBM:MVD:TRB:` is a static prefix, the subsequent `Mvd` identifies the XML file, `0xd010` represents the hexadecimal address of the TRB3 board answering the request and `DataLength.0` identifies the first slice of the entity named ‘DataLength’.

5.12 Channel Access Gateway

The Channel Access Gateway is an application that serves as a unidirectional router for EPICS networks. It binds to two network interfaces (with different networks/broadcast domains) and allows EPICS clients on the one network to access PVs served by IOCs on the other network via CA (Channel Access). A quick introduction by the FAIR/PANDA experiment is given in [148]. Please refer to fig. 5.1d for a possible Docker deployment of the CA Gateway application. If bi-directional routing is required, two instances of the CA Gateway can be started

⁶Found at https://jspc29.x-matter.uni-frankfurt.de/git/?p=mvd_epics.git;a=blob;f=MVD_TRBNET/mvd_trbnet_ioc.py;hb=HEAD.

⁷As TrbNet has a concept similar to Multicast, a request can be sent to a group address and multiple responses from all TRB boards belonging to that group will be received with their individual TrbNet address as sender. I.e., responses to requests can come from addresses different from the one the request had been sent to.

⁸<http://jspc29.x-matter.uni-frankfurt.de/git/?p=daqtools.git;a=blob;f=xml-db/database/Mvd.xml;hb=HEAD>

with complementary client/server network settings.

5.13 Architecture of the EPICS DCS for PRESTO

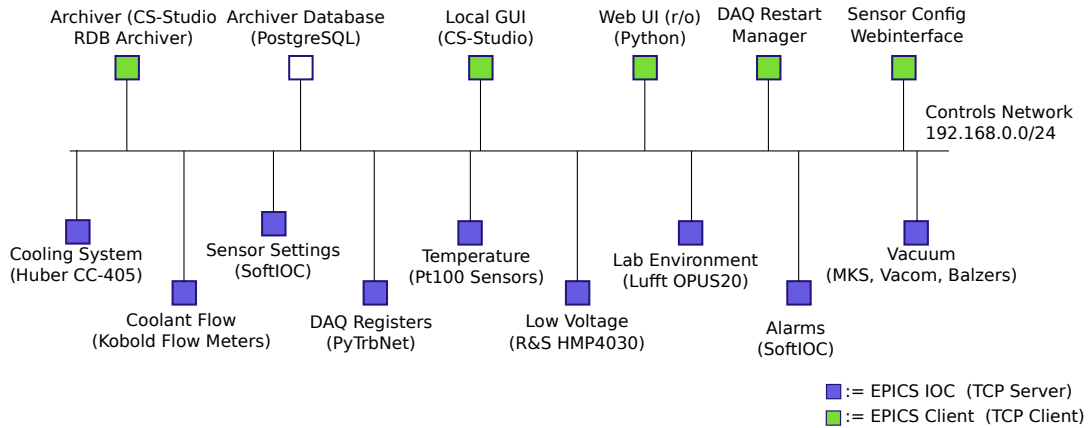


Figure 5.8: Entities connected to the EPICS network for PRESTO.

This section analyses the EPICS DCS implemented for PRESTO with respect from two perspectives: (a) its categorization into a server/client architecture and (b) its categorization into backend and frontend services.

EPICS follows a client-server paradigm, see section 5.5. Client and server refer to the connect and bind roles in the TCP/UDP protocols. In the EPICS ecosystem, the IOCs are the ‘servers’, providing access to their process variables (PVs) on the network and thereby allowing to interface the hardware connected to them.

Figure 5.8 shows the architecture of the PRESTO control system with all essential network nodes.

Another possible perspective of analysing the EPICS control system is splitting the system into logical backend and front-end components. Front-end would denote all interfaces exposed to operators (from detector specialists to shifters) and mostly means graphical user interfaces. Backend, in turn, denotes any system running in the background, ensuring the proper operation of the system (but hidden to regular users).

The front-end of PRESTO’s control system comprises:

- Operator Interface GUIs (CS-Studio / Phoebus)
- Dashboard Website (user-facing side)

The backend comprises:

- EPICS IOCs connected to actors/sensors

- EPICS SoftIOCs (state determination, alarming)
- EPICS clients for autonomous actions
- Archiver
- Dashboard Website (server side)

In the following sections, the different components making up the EPICS control system set up for PRESTO are described in more detail.

5.14 EPICS Archiver Solutions

The so-called archiver software is an essential part of any DCS. Its major task is twofold. First of all, it needs to be connected to the EPICS control system, subscribe to any PV updates and archive (i.e. save) any (significant) changes in a time-series data store. Secondly it needs to provide the archived time-series data to any operator or other software requesting access to the time-evolution of given PVs. For the former, it is very important that the archiver software runs as reliably as possible, while the latter the main challenge is response time, as time-series covering a long period are often large data sets.

The archiver mainly used for the PRESTO EPICS system is the CSS/RDB Archiver, sometimes called “Best Ever Archive Toolset, yet”, or BEAUTY [149]. It stores metadata and archived values in a JDBC compatible relational database (commonly MySQL or PostgreSQL). This archiver is being used by the HADES experiment at GSI [150] in a very similar setup and the installation for PRESTO implemented a PostgreSQL database setup with table partitioning scheme identical to theirs. During the operation of PRESTO, the archiver was running reliably, but the performance of requesting archived time-series data was rather low, if long periods of time were queried (e.g. a full year of time-evolution for a given PV). It became clear that the choice of the archiver is critical for various reasons: (a) the retrieval and visualization of process variables should be as responsive as possible, (b) the archiver should scale appropriately to support the continuous archival of a large amount of process variables, (c) the storage backend should be as straight forward to manage (backup and transfer).

Therefore, a more thorough evaluation of alternative – possibly better suited – archiver solutions was done and is presented in the following paragraphs.

Criteria for the Evaluation of Archiver Software

In the course of this thesis, different software solutions to archive EPICS data were evaluated based on the following criteria:

- Storage efficiency (typical byte size required per stored PV value update),

5 Development of the Control System for PRESTO

- Retrieval performance (responsiveness when retrieving historic values),
- Data ageing capability (e.g. ability to thin out older data),
- GUI integration (into typical EPICS OPI software),
- Architecture (e.g. modular and light-weight vs. fully featured and monolithic)
- Integrated data viewer (e.g possibility to view PVs on a web page),
- Data accessibility (e.g. for use in ROOT [151] analyses, export to file etc.),
- Scalability (capability to archive a large number of PVs concurrently),
- Ease of deployment (configuration complexity, number of processes/containers/databases/load balancers)
- Use of standard software (e.g. a well established backend database),
- Code maturity and stability (with a broad user and developer base),
- Extensibility (e.g. custom requirements for plotting, aggregated data queries, etc.).

The functionality (i.e. set of features) and stability offered by the software should be the primary consideration when deciding on the most suitable archiver software. While the performance should not be neglected on the other hand, it is not always straight forward to evaluate, as it can vary depending on many external factors. To achieve optimum performance for both, storage of new PV updates and retrieval of stored time-series data, a given archiver solution can typically profit from fine tuning its configuration and from **scaling up** (vertical scalability). By switching to more capable hardware, scaling up can provide the easiest path to rapid performance gains, even without configuration changes. Specifically, the archiver storage backend might profit significantly from switching to low latency storage technology, e.g. Intel 3D XPoint drives, increasing TPS (transactions per second) in the case of Relational Database Management Systems (RDBMSs) or file IOPS (input/output operations per second) in general for other persistency backends. Further fine-tuning can then help to achieve additional performance gains. In the case of an RDBMS storage backend an example for such a fine-tuning would be the addition of indices for tables affected by performance issues. After all, the performance of chosen archiver software should be benchmarked with a realistic scenario (number of PVs to be archived, update rate) and then be improved iteratively – as suggested before, by scaling up first (often the fastest and thus cheapest solution) and, secondly, by fine-tuning the configuration parameters of the archiver software.

Some archiver solutions also allow **scaling out** (horizontal scalability). In this scheme, the task of the software is spread across multiple nodes (computers).

Those nodes can then be composed of cheaper standard commodity hardware. The overall system performance is then dominated by the number of running nodes, not their individual performance. Only few existing archiver solutions support such an architecture. Examples include the EPICS Archiver Appliance [152, 153] and the Cassandra PV Archiver [154, 155].

In addition to those existing archivers, a POC solution was investigated to implement a new, custom archiver with a binary file backend storing its data in the format of CA protocol datagrams.

While the following paragraphs state evaluation results for the individual archivers, the table 5.2 provides an aggregated overview of how those archivers solutions compare based on the aforementioned criteria. It includes the CSS RDB Archiver ‘BEAUTY’, the Archiver Appliance, the Cassandra PV Archiver and the caproto-based archiver developed as a POC for this thesis.

CSS/RDB Archiver

The CSS/RDB Archiver was chosen in 2017 to be used as primary archiver solution for the 24/7 operation of PRESTO. The software dates back to as early as 2007 [156].

When operating the prototype PRESTO, accessing archived data of PVs with high update rates for long time spans (e.g. 1 year) revealed performance issues of the CSS/RDB Archiver with the PostgreSQL backend in use, which resulted in an inadequate user experience (response time) for the operator. The CSS/RDB Archiver is claimed to be less performant than its predecessor, the ‘Channel Archiver’ [157].

EPICS Archiver Appliance

By EPICS standards, the *EPICS Archiver Appliance* is a rather recent application, with its first contribution to the (now outdated) Mercury/hg repository documented in 2011 [158].

The archiver appliance is a software application written in Java and it is typically split into four Java servlet processes:

- **Engine** (engine.war) – subscribing to process variable value/state changes
- **ETL** (etl.war) – moving data from short to medium or long term store
- **Retrieval** (retrieval.war) – serving requests to retrieve archived data
- **Management** (mgmt.war) – managing the other servlets and keeping configuration state

Experience with the Archiver Appliance:

Registering process variables can be done on the management web interface for smaller numbers of PVs. For the load testing, this was done with a Python script. It sent request to add PVs in blocks of 200. When the archiver has already 120 000

5 Development of the Control System for PRESTO

	CSS/RDB Archiver	EPICS Archiver Appliance	Cassandra PV Archiver	Caproto POC (Custom Proof of Concept)
Written in	Java	Java	Java	Python
Data stored in	Relational Database (RDB)	Binary Files, one per PV	Distributed NoSQL Database	Binary Files, one per PV
Data store specifics / formats	MySQL / PostgreSQL / InfluxDB	Protocol Buffers (pb)	Apache Cassandra	Binary Channel Access Datagrams
Approach	Slim Java application from same code base as CSS	Fully featured Java web appliance with high extensibility	Modern distributed architecture with multi-instance	Very slim and light weight, data stored as native Channel Access messages
Integrated web interface	Yes: status of archiver and each PV including latest value	Rich web interface to add PVs, manage settings, view archived data	Login-protected web interface to add and configure PVs, NO built-in data viewer	n/a
Way of adding PVs	Edit XML configuration file and apply to DB; restart application	Via web interface or scripting	Via web interface or scripting	Edit configuration file; restart application
First public appearance	2008 or earlier [1]	2011 [2]	2016 [3]	2020 [4]
Phasing out old data	Support using partitioning of the DB tables / deletion of old partitions	Storage segmented into short/mid/long term storage (by default based on hour/day/year)	Configurable for advanced (cascaded) decimation of samples.	Data is split according to calendar year. Easy deletion on file level.
Retrieval responsiveness	Rather slow (depending on DB storage IOPS)	Fast	Fast	Fast
User base	Large	Large	Limited	None (POC)
Integration into typical GUIs	CSS ✓ phoebus ✓ (built-in)	CSS ✓ phoebus ✓ (built-in)	CSS ✓ phoebus (?) (additional plugin)	No (not yet)
Scalability	Limited (vertical)	Good (horizontal and vertical)	Excellent (horizontal)	Possible (horizontal, not yet implemented)
Built-in ability to export data	Indirect (via regular DB tools)	Good (different formats possible)	Basic (available as JSON documents)	No (not yet)
Code maturity / maintenance	Very mature / well maintained	Very mature / solid maintenance	Mature / average maintenance	Alpha Software / no maintenance

[1] <https://github.com/ControlSystemStudio/cs-studio/commit/116c074> accessed 2020-06-18

[2] <https://sourceforge.net/p/epicsarchiverap/code/ci/86b4b7c16733705561fb22d501d64f55d67c52b5/> accessed 2021-06-13

[3] <https://oss.aquenos.com/cassandra-pv-archiver/#download> accessed 2021-06-02

[4] <https://github.com/pklaus/caproto-archiver> accessed 2021-06-02

Table 5.2: A comparison between three major archiver solutions for EPICS and a custom POC (Proof of Concept) archiver using caproto.

PVs in archiving state and connected, the script was blocking for about 7 seconds for each block of 200 additional PVs. When added this way, they enter an "initial sampling" state. It takes further time in the range of minutes to hours until they are processed by internally to transition into the final "being archived" state.

Cassandra PV Archiver

The Cassandra PV Archiver [154] is another archiver application for EPICS that attempts to be horizontally scalable built on top of a high performance Apache Cassandra distributed database, thus the name. The project started under the name *Cassandra Archiver for CSS* in 2011 [160, 159]. To evaluate this archiver software, an example deployment in a container image was created in the context of this thesis for the Cassandra PV Archiver [161]. Further details on this archiver and an analysis of its potential for an application modernization of EPICS can be found in section 7.2.7. The Cassandra PV Archiver is an option considered by the PANDA experiment [162, sec. 3.6.3.3]. Beyond implementing an evaluation deployment as mentioned before, this archiver was investigated further due to the requirement of deploying a Cassandra database [163], which adds a complexity that was deemed undesirable.

Caproto Archiver

To overcome retrieval performance bottlenecks (i.e. when accessing archived data) that occurred with the CSS/RDB Archiver, a POC (Proof of Concept) for an archiver application storing the PV updates in the format of appendable binary files on the filesystem was drafted. As it relies on the Python package caproto [164] it was named **caproto-archiver** [165]. That Python package **caproto** serves as an EPICS client library with the special feature that it provides access to the binary representation of the CA datagrams it receives, whenever it is being notified about a PV update. This binary data, specifically the **CA_PROTO_EVENT_ADD** response packets [166], is well suited as archiving format and is being stored without converting it to a different format. The datagram provides all necessary information in a tightly packed binary form and loss of information due to format conversions is also excluded. As an example, the severity metadata associated with an updated value of a PV is then directly embedded in the recorded data stream.

The archived data is written to files with the file name extension **.ca**, named after the CA protocol as the data it contains are CA commands. A high speed parser written in C would easily allow to reach the unpacking speed desired to process millions of updates in fractions of a second, which would result in a responsive user experience for operators browsing time-series data of archived PVs.

5.15 Monitoring the Operational Stability

Closely monitoring and controlling key parameters of the detector is crucial to ensure stable conditions during the data taking. This is an important requirement in order to keep systematic errors of the subsequent physics analysis as small as possible to increase significance⁹. A way to measure the stability, is to monitor the incoming stream of controls data for changes. Those changes can be manifold – e.g. abrupt changes, slow drifts, noise, periodic fluctuations, depending on the type of information and the system the data is representing. In the simplest implementation, watching for a stable parameter might be as simple making sure its raw value doesn't leave a defined deadband (tolerance range). An example would be the volumetric flow of a coolant. In more complex cases, the sampled raw values are varying over time. An example would be a signal subjected to electronic noise – instead of the raw value of the process variable, the power spectral density of the noise might be monitored and expected to stay within certain limits.

In more general terms, a processing step on a number of subsequent raw values shall lead to a metric that's supposed to stay constant over time. In this context, 'constant' means constant within certain limits, for example 5% of a foreseen value. The exact tolerance that seems acceptable can be different for each PV. Besides thresholds to be set by the operator, data mining can be used to automatically determine the baseline of PV values and the typical variance during normal operation.

In machine learning, the entire process of transforming the raw input data to information that allows for straight-forward categorization is called dimensionality reduction [167, p. 354–361] and can be applied to this stability testing as well.

Section 5.15 shows a generalized processing chain for this kind of stability analysis. Incoming values updates ("samples") are first being buffered¹⁰. One or multiple processing steps are then triggered on the buffered samples. Finally, a single derived figure of merit is compared (= subtracted) either to its previous value or to an external reference value. If the difference exceeds a configurable threshold, the discrimination stage triggers an alarm to indicate the unstable condition.

To monitor such a property for stability, the following approach is proposed:

1. Group the incoming stream of raw values according to their timestamp.
This grouping can be rather simple according to the absolute minute, hour, day etc. or more complex according to time of day, day of week and so on.
2. Fill histograms with the values (with fixed bin number and limits) for each

⁹An example for conditions affecting the systematic errors are time dependent calibration or alignment parameters, e.g. the stability of thresholds, which could the tracking efficiency.

¹⁰typically with a ring-buffer for a configurable number of maximum sample entries

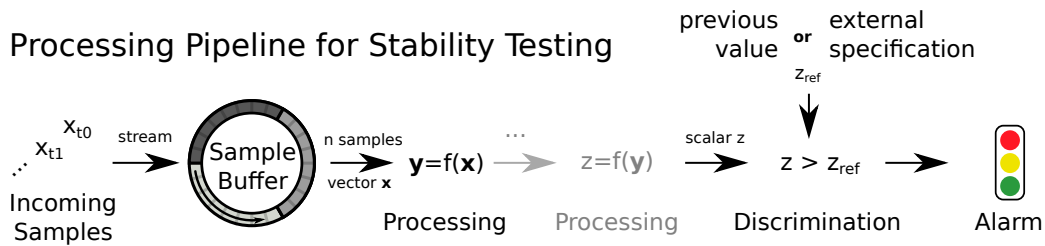


Figure 5.9: The generalized sample processing chain to test for stability of conditions

group

3. Compare the histograms to each other using
 - a) a similarity function or
 - b) machine learning.

and take the output of the chosen approach as the measure of 'stability' or 'changing conditions'.

At the end of the stability testing processing pipeline, the outcome is an overall state, which indicates possible alarm states. It is then up to the control system to react appropriately. This might be a visible / audible indication to the operator, a push message delivered to the responsible expert, an automated shutdown, or similar.

5.16 Commissioning of the Control System

The system as described above was successfully put into operation during the year 2017 and successively extended with more capable IOCs, automations, the PyTrb-Net bridge between DAQ and DCS in the subsequent years. Historic values for essential operational data, such as the PVs of the temperature measurements of heat-sink and bath temperature of the cooling system date back to as early as mid 2017.

After commissioning the prototype together with all devices required to operate it and the EPICS control system, field tests were required to validate the reliable vacuum operation of the prototype. The next chapter, chapter 6, describes those field tests in detail: multiple tests of vacuum operation and cooling cycles (section 6.1), in-vacuum test with an ^{55}Fe source (section 6.2) and a measurement of cosmic rays (section 6.3).

6 Field Tests

After the installation & commissioning of PRESTO in its foreseen vacuum setup, and after adding support for the required devices to be controlled for its operation, PRESTO was started to be operated 24/7. Various field tests were performed with the MVD prototype PRESTO to validate its vacuum operation and to test the capability of the detector control system.

6.1 Vacuum and Cooling Tests with PRESTO

First, a long-term vacuum operation was scheduled in order to test the full prototype under continuous vacuum conditions. Subsequently, many automated vacuum cycles were performed in an automated way. Finally, dedicated cooling cycles were done.

6.1.1 Motivation

Validating essential concepts of the MVD with the prototype PRESTO is the primary goal the tests (i.e. measurements) described in this chapter. The vacuum quality required for the MVD is located at the transition from medium to high vacuum, i.e. 1×10^{-3} mbar [168]. It is therefore the upper limit acceptable for PRESTO. The ability of the prototype to sustain extended periods of operating under constant vacuum was put to test. The aim is to evaluate whether the prototype performs well and is operated safely 24/7 under those conditions, as they are very challenging for both, the integration concept (including the cooling concept) as well as the electrical integration (including the data acquisition system and the control system. For this test, it is essential that key parameters are properly controlled in well-defined boundaries during the entire campaign, including process variables related to vacuum, thermal management, data acquisition, powering, and more. Besides testing the prototype itself for its ability to sustain the prolonged operation at the foreseen working point without failure, the measurements also serve as perfect field test for the Detector Control System (DCS) developed in the context of this work to operate the prototype. This required the control system to be fully commissioned at the start of the field test.

6 Field Tests

The vacuum and temperature cycles put further stress on all components involved, as material fatigue may occur due to outgassing or expansion. In the CBM experiment, the detector will have to endure numerous pumping and cooling cycles without damage. Vacuum cycles introduce mechanical stress to the components and pose a risk if air would still be contained in pockets or bubbles, which might expand – a scenario hopefully avoided with careful integration. In addition, vibrations caused by the gas flow in or out of the chamber or caused by valves and pumps might cause additional mechanical stress, for example on the bonding wires connecting the sensors with the FPC cables.

Cooling cycles are another important tool to test system stability under realistic conditions. Mismatching coefficients of thermal expansion (CTE mismatch) can cause bending, coolant leaks, or even breaking components. Cooling cycles are essential to validate coolant line tightness at the desired connections and can help to estimate the number of times, a system can safely be subjected to thermal cycles in the final installation in the experiment.

A further motivation to perform these tests is to validate the cooling and stability of PRESTO, at an intended coolant temperature of -30°C .

6.1.2 Setup

The full setup to operate PRESTO as described in chapter 4 was used for this test. Specifically, the vacuum system used is described in section 4.7. Figure 5.5 shows a drawing of PRESTO and all devices which need to work together to run the prototype continuously as used for these measurements.

6.1.3 Implementation of the Measurements

The initial long-term vacuum test was designed to consist of three vacuum cycles with rising durations of 2, 4, and 96 subsequent days. The results for this measurement are shown in section 6.1.4.

After verifying long-term operation of the prototype PRESTO under constant vacuum, the device has been subjected to intensive vacuum cycles. A total of 130 vacuum cycles were carried out fully automatically. For this purpose, the roughing pump was switched on and off remotely via a power socket connected to the network. After stopping the pump at the end of each cycle, the chamber was vented in a controlled manner by means of an electromagnetic valve. For this purpose, a binary valve (open/closed), which was also connected via the network, was controlled. A manual flow limiting valve connected in series allowed to tune the time

needed to vent the full chamber. The result of those measurement is shown in section 6.1.5.

6.1.4 Result of the Long-Term Vacuum Operation

During the long-term vacuum test, a final vacuum of $\sim 8 \times 10^{-5}$ mbar was reached. While the cooling system was constantly set to 15 °C the heat sink of PRESTO was constantly kept at ~ 17.8 °C which results in an estimated sensor temperature of ~ 28 °C at the coolant flow of 0.8 l min^{-1} [169].

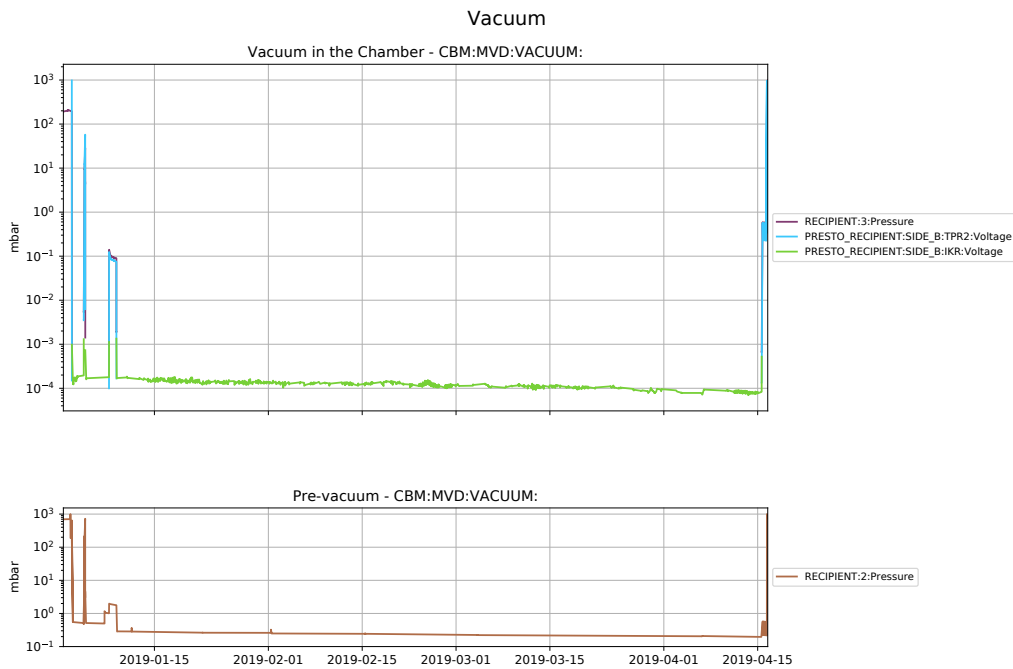


Figure 6.1: Long-term high-vacuum test with three consecutive vacuum cycles started on 2019-01-02 and finished on 2019-04-15.

Figure 6.1 shows the vacuum obtained during the campaign. The upper plot shows the residual pressure in PRESTO’s vacuum chamber. Multiple vacuum gauges are combined to give a good measurement of the vacuum present in the chamber, including a Pirani-style sensor (“TPR2”) and a cold-cathode gauge (“IKR”) connected to a Balzers PKG020 controller. In addition, a newer Vacom MVC-3 controller with combined Pirani + hot-cathode gauge (“Atmion”) is plotted there. The pre-vacuum system (as displayed in the lower plot) was measured with a Pirani vacuum gauge (connected to channel 2 of the Vacom MVC-3 controller). As another Balzers PKG020 controller was connected to the same chamber (Side-A, not plotted) was providing noisy data, the other gauges all functioned regularly. As they were all connected to the same chamber, they provide redundant access to

6 Field Tests

the same measurement, which proved useful to analyse the data month or even years later.

During the long-term test in fig. 6.1, the frame size of all eight sensors of PRESTO, that could be read-out concurrently, was monitored. For each day of the measurement campaign, histograms of the frame size values (sampled once per second) are displayed in fig. 6.2.

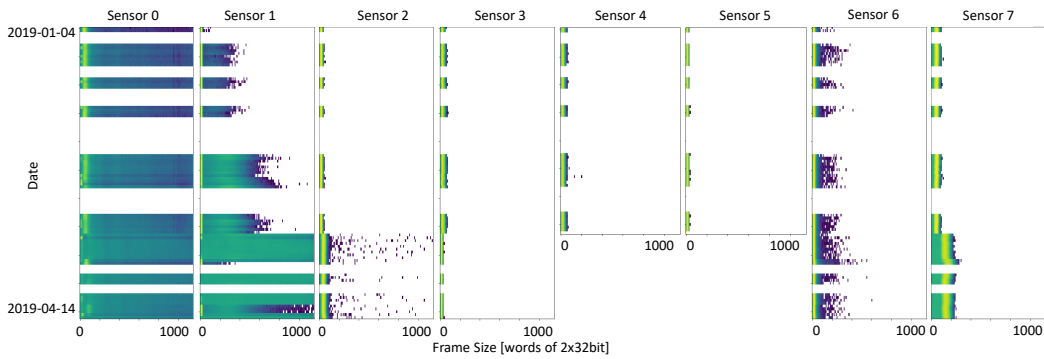


Figure 6.2: Histogram of frame size of each sensor for each day during the early 2019 long-term vacuum campaign.

Finally, the last measurement campaign under vacuum was in 2020, see fig. 6.3. In this long-term vacuum measurement, a newer vacuum gauge was employed (MKS 910 DualTrans, with the PV name **CBM:MVD:VACUUM:PRESTO:Pressure**), replacing one of the older Balzers PKG020 controllers as well as the Vacom MVC-3 controller used before. During this campaign, PRESTO was first tested with a vacuum compatible radioactive ^{55}Fe source, see section 6.2 and in the second vacuum cycle a measurement of cosmic rays and natural background activity was performed, see section 6.3.

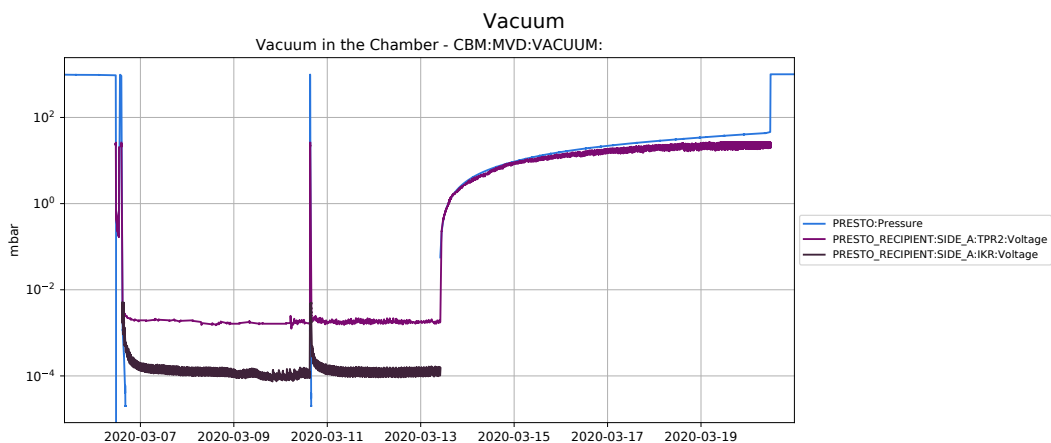


Figure 6.3: Long-term vacuum test from 2020-03-06 to 2020-03-13.

6.1.5 Results: Vacuum Cycles

Figure 6.4 shows the vacuum as recorded during the cycles by the CSS RDB Archiver. During the initial couple of vacuum cycles, the system wasn't vented entirely. After increasing the venting gas flow rate, the system was fully vented after each pumping cycle. A single cycle was missed on 2019-04-25 at 14:05 due to a network outage. Figure 6.5 shows the last vacuum cycle of this series in detail.

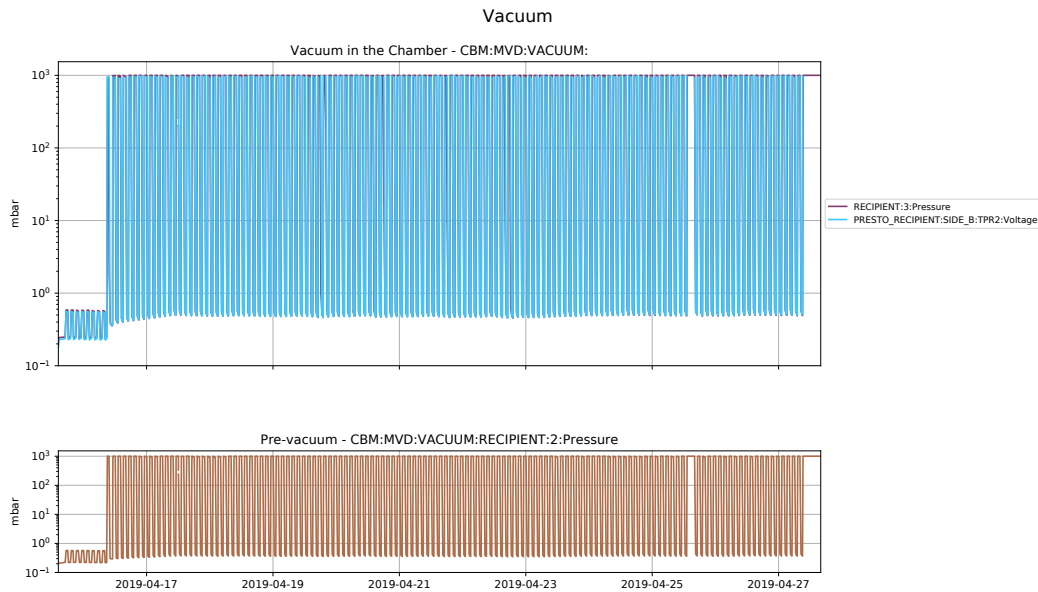


Figure 6.4: Pressure during 130 automated vacuum cycles from 2019-04-15 to 2019-04-27.

6.1.6 Result: Cooling Cycles

Figure 6.6 shows the operation of the cooling system from 2017 until 2020. The upper plot shows the temperatures measured with the Pt100 measurement system¹ as well as the setpoint and actual temperature for the cooling system, while the lower plot shows the measured coolant flow to the heat sink and back². Continuous operation started mid-2018. In 2019-05, a major reduction of the noise in the measured flow data was achieved, by fixing an inconsistency in the algorithm that calculates the flow based on the digital pulses from the flow meters.

¹The Pt100 measurement system consists of the custom PT100 PCB shown in fig. 4.4 j) and an accompanying EPICS IOC.

²The coolant flow was measured by the two Kobold DPM flow meters fig. 4.4 e).

6 Field Tests

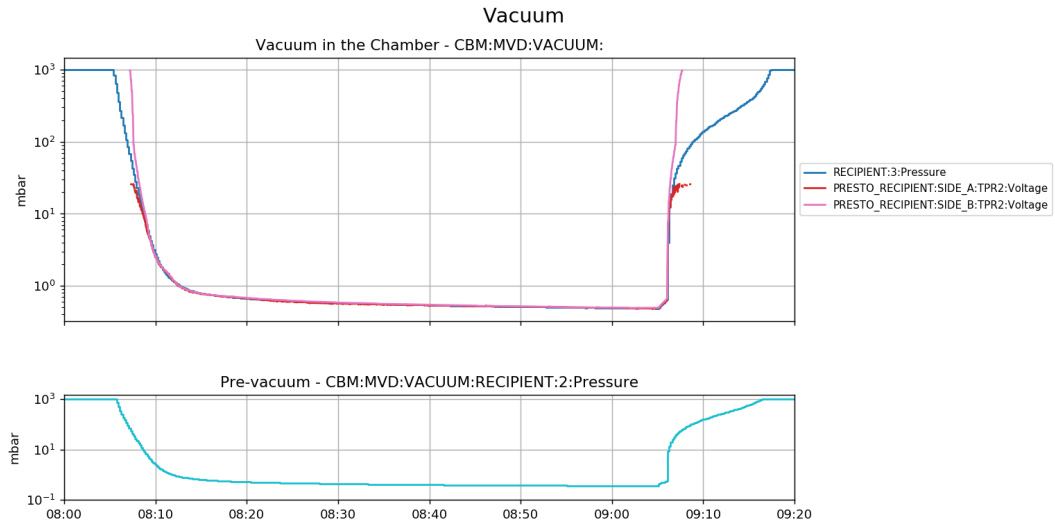


Figure 6.5: Detailed pumping curve for the final vacuum cycle of the 130-cycle series shown in fig. 6.4. Measured on 2019-04-27.

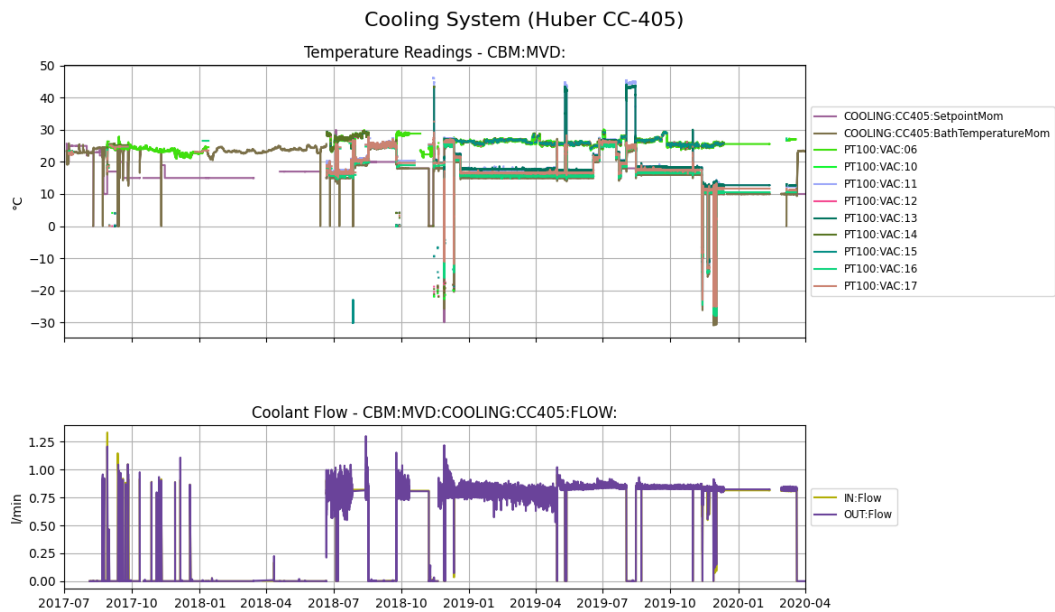


Figure 6.6: Surveillance of the PRESTO Cooling System 2017 to 2020.

Figure 6.7 shows a cooling cycle measured on 2018-12-11, which was aborted as the coolant flow dropped to almost zero.

Figure 6.8 shows a later cooling cycle performed with coolant mixture of 50% MEG (monoethylene glycol) and 50% deionized water.

6.1 Vacuum and Cooling Tests with PRESTO

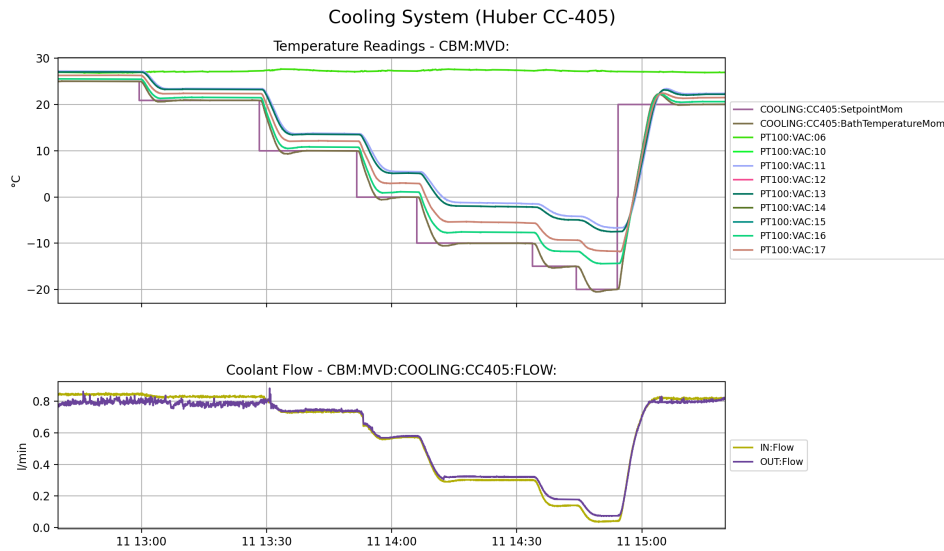


Figure 6.7: Cooling cycle on 2018-12-11. The volume flow almost dropped to

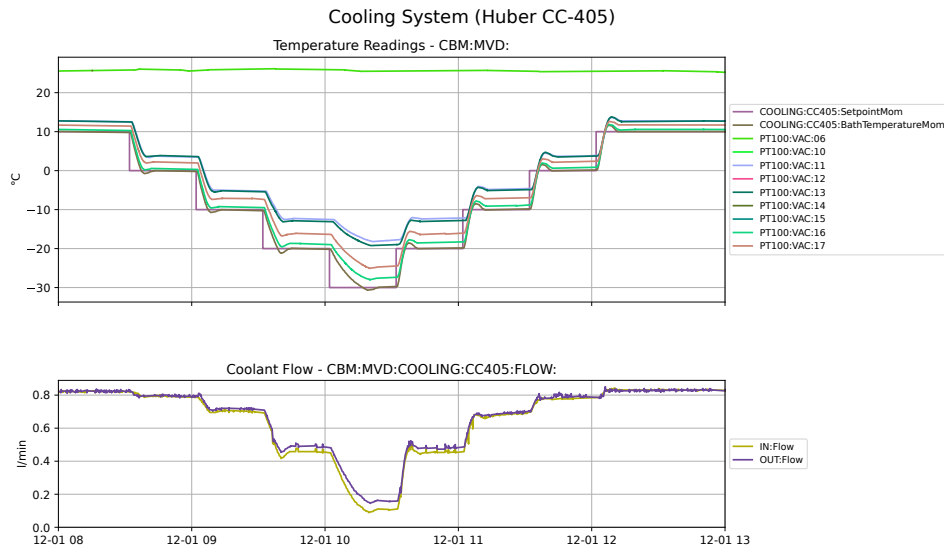


Figure 6.8: Cooling cycle of PRESTO on 2019-12-01.

6.1.7 Discussion

A first conclusion of the long-term vacuum test in section 6.1.4 is the insight that the vacuum chamber was tight with PRESTO inside and was able to reach a continuous vacuum of better than 10×10^{-3} mbar. Furthermore, it can be concluded that the control system was properly, continuously archiving the readings of all vacuum gauges. Another important result of the long-term vacuum campaign is the insight that redundant measurements with separate communication links allow to reconstruct experimental conditions in the case of device failures, connectivity

issues and network outages.

The frame size distribution for each day during the long-term vacuum campaign as shown in fig. 6.2 serves as a good observable to assess the stability of the system's DAQ. It was readily available in EPICS via the PyTrbNet-based IOC described in section 5.11.3. In the absence of physical hits (e.g. accelerator beam on target, radioactive source, pulsed light), the fake-hit rate can well be estimated from the frame size. The DAQ stability analysis in fig. 6.2 reveals that (a) that the data stream was interrupted several times, (b) that the level of noise varied from time to time, although no thresholds were actively changed and (c) that two sensors (4 and 5) stopped sending data altogether by 2019-03-15. The problem of the readout halting every couple of days was identified to be caused by the specific DAQ, which came to a halt after having received 2^{31} frames after being started. This is equivalent to 2 days, 20 hours and 43 minutes of data taking, as the MIMOSA-26 sensors provide a frame rate of 8680 fps. A workaround was implemented later implemented to restart the data acquisition automatically every other day, effectively solving this issue. As a consequence of manual DAQ restarts in the presented data, the typical frame size (i.e. the noise level) sometimes changed inadvertently, although the threshold settings were not changed. This is undesirable behaviour that needs monitoring for future detector and sensor generations. After all, using noise to test the stability of the detector operation in terms of fake hit rate, sensor currents, etc., as done in fig. 6.2, is challenging. The reason for this is (1) bad reproducibility, as noise often changes without obvious reasons if too many external factors play a role, which can result in (2) a lack of meaningfulness, when the sources of the noise and its overall role for the system performance might be unclear. This calls for well-defined and robust grounding scheme and full control of the run parameters. The vacuum chamber for the MVD prototype PRESTO (and later the MVD itself) helps reducing external influence as it serves as a faraday cage shielding many undesirable electro-magnetic sources of noise. High frequency signals, however, might still enter the chamber via viewports or cables through feedthroughs.

Nevertheless, the measurement was important, as it showed that the DAQ came to a halt after a certain amount of time and that a way to automatically restart the data acquisition was needed, which was in deed implemented consecutively.

From the vacuum cycles presented in section 6.1.5, it can stated that running fully automated cooling cycles was possible and successfull with the system. The total amount of 130 vacuum cycles performed is expected to contain a sufficient safety provision for an estimated 40 to 60 vacuum cycles during the lifetime of the Micro Vertex Detector within the CBM experiment. The last vacuum cycle presented in fig. 6.5 indicates that the vacuum chamber reached a vacuum of 1 mbar after ~ 8 min, which would have been enough for a typical turbomolecular pump to be

switched on. This is a very good result, given the presence of 3d-printed components in the vacuum chamber, which are inherently prone to representing virtual leaks in the vacuum chamber.

Section 6.1.6 presents the measurements of the overall successful cooling campaign.

A first attempt to perform a low-temperature cooling cycle at the end of 2018, depicted in fig. 6.7, revealed that it was impossible to go to temperatures lower than -10°C on the heat sink (PVs: `PT100:VAC:11` and `PT100:VAC:13`) even if the heat bath in the Huber CC-405 cooling plant was reaching -20°C (`COOLING:CC405:Bath TemperatureMom`). The volume flow of the coolant (`IN:Flow`) dropped to almost zero, which is visible in the second graph of the figure. It became evident, that the cooling system was accidentally filled with a 100% MEG mixture (Coracon KS 6 [170]), resulting in a steep rise of its dynamic viscosity at sub-zero temperatures [171, fig. 7]. For laminar flow through a long cylindrical pipe, Hagen-Poiseuille's law shows the inverse proportionality between the volumetric flow of a fluid and its dynamic viscosity for a given pressure difference. In consequence, the internal pumps of the Huber CC-405 cooling plant were effectively unable to pump the coolant at -20°C . The freezing point of a 100% MEG solution, however, is -13°C [171, fig. 2]. The observation that the cooling system was effectively still able to cool down the fluid to almost -20°C without any ice building up in the internal heat bath gives an indication that the coolant had already drawn water from the laboratory humidity in the order of several percent by weight.

After changing the coolant to a fresh mix of 50% Coracon KS 6 and 50% deionized water, the dynamic viscosity at low temperatures was greatly improved. Figure 6.8 shows a cooling cycle with the new coolant mixture. With each step lowering the setpoint temperature by 10°C , it took longer to achieve the steady state. This is partly because of the reduced cooling power of the Huber CC-405 heat bath thermostat at lower temperatures [172]. Another factor in the less steep slope is the reduced flow with lower temperatures, leading to slower turn-around times.

While the cooling system adjusts to the updated setpoint, there is a small overshooting in the reached value. It is an interesting observation that this directly translates to a lower coolant flow. The resulting volumetric flow at the setpoint temperature of -20°C was more than 4 times higher compared to the cooling cycle with the measurement on 2018-12-11, where the coolant was (almost) pure MEG (fig. 6.7).

The findings highlight the importance to fully understand the dynamic viscosity of the cooling liquid (both, at the target temperature and room temperature)

Most importantly, no sensors were lost, and no degradation of the prototype was found upon visual inspection after the vacuum and cooling cycles.

6.2 Test: Full System Test with a ^{55}Fe Source

Subsequently, the prototype was tested with a a radioactive ^{55}Fe source in vacuum.

6.2.1 Motivation

To validate the prototype under realistic operating conditions in the laboratory (in vacuum, with particles – not just noise), a radioactive ^{55}Fe source was employed. The photons it emitted were partially shielded by a mask, yielding a characteristic and recognizable illumination pattern on the sensor. This test can help to assess the ability of the sensor to sense physical particles and the validity of the DAQ and the data being collected over an extended period of time.

6.2.2 Experimental Setup

Measurements with ^{55}Fe sources are briefly introduced in section 3.4.1. A schematic closeup of the setup placed inside the vacuum vessel is shown in fig. 6.9. To the left, a nickel-coated vacuum-compatible radioactive source of ^{55}Fe is located.³ The radiation is being partially blocked by a milled Aluminium mask with strip and box-like cut-outs, see fig. 6.11. Any radiation passing the mask illuminates the sensors of PRESTO (back-side) to the right. A photo of the setup is provided in fig. 6.10, showing the source to the left and the mask and the sensors on the right. The sensors are hardly visible behind the mask but the orange FPC cables from the sensors to the FEBs are visible in the centre of the photo. The dimensions of the mask and its cut-outs are indicated in fig. 6.11. The aluminium plate has a thickness of 0.8 mm and the mask shape has been CNC-machined at the in-house workshop.

Operating conditions

The prototype was operating in vacuum of approximately 10^{-4} mbar, see fig. 6.3 (first cycle). The sensors were operated at $\approx 28^\circ\text{C}$.

6.2.3 Measurement Result and Data Analysis

The measurement was carried out with PRESTO operating in vacuum and a continuous data taking was scheduled for a duration of 25200 seconds (7 hours).

³The source used for this study, with the inventory number 55.26.08, has a nominal activity of 1.85 GBq as of its reference date 2008-07-01. On the day of measurement, 2020-03-07, the remaining activity was 97 MBq.

6.2 Test: Full System Test with a ^{55}Fe Source

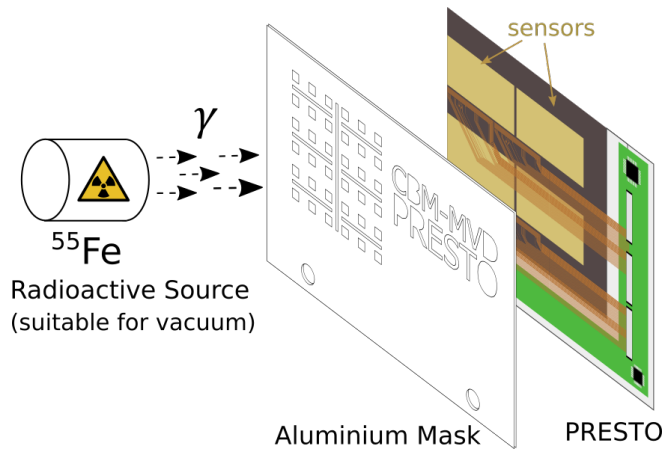


Figure 6.9: Illustration of the experimental setup for the ^{55}Fe measurements.

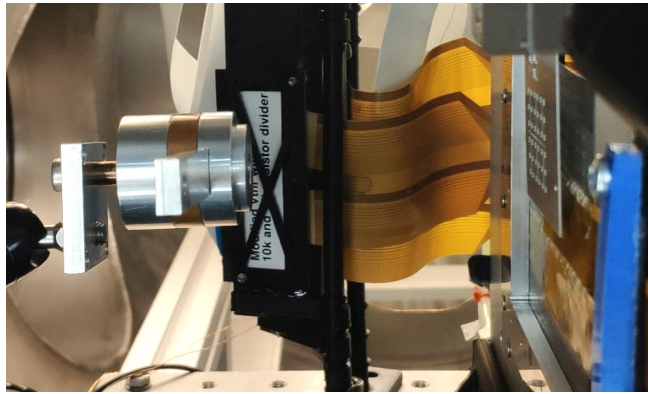


Figure 6.10: Photo of the experimental setup for the ^{55}Fe measurements in vacuum.

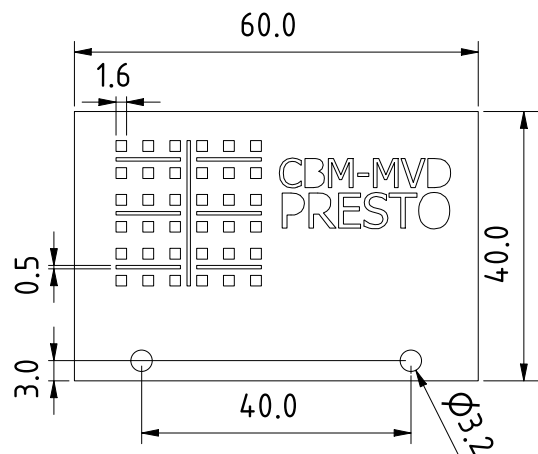


Figure 6.11: Dimensions of the shielding mask for the ^{55}Fe test in vacuum. Dimensions in mm.

Figure 6.12 shows the matrix of accumulated hits on the most illuminated sensor and the cut-out regions of the used mask are clearly visible in the hit pattern. A region of interest was selected containing a slice of one of the illuminated patches. A vertical integration (i.e. sum) of the hits in the ROI (Region of Interest) along the column direction of the sensor is shown at the bottom of the figure. In addition, the steepness of the edge of the cut-out can be used to reproduce the thickness of the metal mask used. Due to uncertainties about the internal geometry of the vacuum-compatible radioactive source, this is not done here. The rise in integrated hits covers over a length of ~ 20 pixels, which translates to a distance of $\sim 370 \mu\text{m}$ (due to the pixel pitch of MIMOSA-26 of $18.4 \mu\text{m}$).

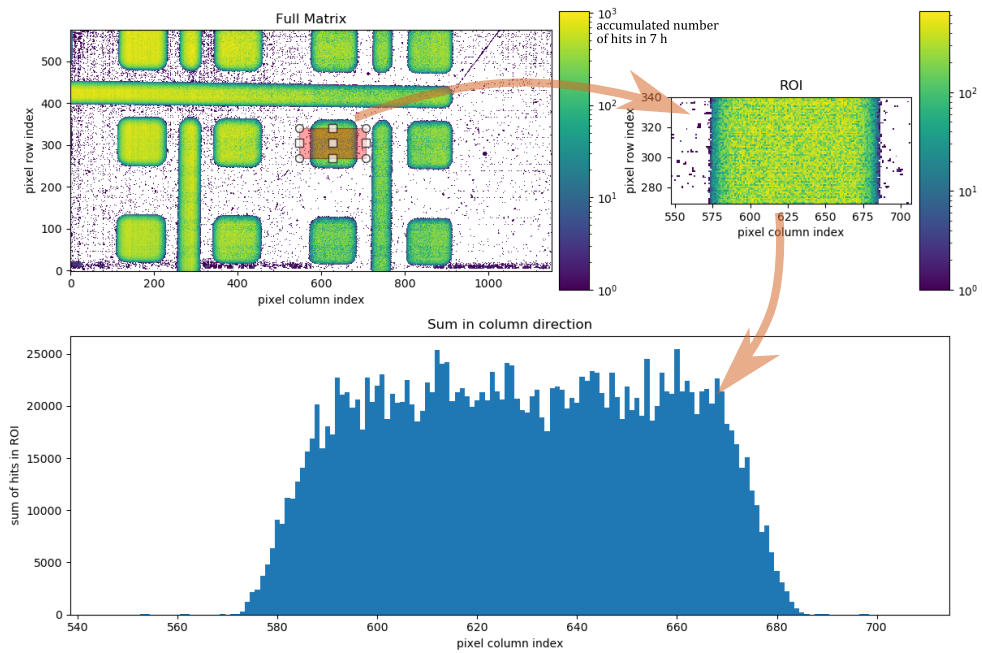


Figure 6.12: Hit pattern analysis of a masked ^{55}Fe source on a sensor of PRESTO in vacuum.

6.2.4 Interpretation

Both, the electrons and photons from the ^{55}Fe source are expected to be almost entirely stopped by the thickness of the aluminium plate.

For *electrons* of 5 keV the total stopping power of $2 \times 10^1 \text{ MeV cm}^2 \text{ g}^{-1}$ [173] leads to a total energy loss within approximately $0.9 \mu\text{m}$ of aluminium and $1.1 \mu\text{m}$ of silicon. Those electrons will therefore either be stopped by the aluminium mask or lose their energy in the upper (metal) layers of the MAPS pixel sensor without creating a measurable signal in the epitaxial layer of the sensors.

For *photons*, the intensity remaining after traversal of thickness t (in mass/unit area) is given by $I = I_0 e^{-t/\lambda}$ [36, p. 544]. Here, λ is the photon mass attenuation length, which is the inverse of the mass attenuation coefficient μ/ρ . λ is quite constant for high energy photons where pair production dominates ($> \text{ca. } 10 \text{ MeV}$), but for lower energy photons in the range, where the photoelectric effect dominates, it is highly energy dependent [36, p. 544]. For aluminium and photons in the relevant range $E_\gamma \sim 6 \text{ keV}$ the mass attenuation coefficient is stated as $\mu/\rho = 1.2 \times 10^2 \text{ cm}^2 \text{ g}^{-1}$ [174] and therefore $\lambda = 1/(\mu/\rho) = 8.673 \times 10^{-2} \text{ g cm}^{-2}$. In the case of the 0.8 mm thick aluminium mask with a density of $\rho = 2.7 \text{ g cm}^{-3}$ and a resulting thickness of $t = 0.22 \text{ g cm}^{-2}$, the intensity of photons traversing the entire material is therefore $I = I_0 \exp(-\frac{0.22}{0.0087}) = 1 \times 10^{-11}$, so practically no photons will make it through the aluminium.

Indeed, those expectations are met, as in the areas where the radiation of the ^{55}Fe source was shielded by the aluminium mask, there were only very rare and uniformly distributed background events.

The histogram in the lower part of fig. 6.12 shows a rather high number of fluctuations, which are not expected from a uniform distribution as the standard error on the mean value σ_N , which should scale with the number of samples as $\sigma_N \sim 1/\sqrt{N}$. Instead, a non-uniform efficiency is assumed for the pixels. This interpretation is supported by the patterns visible if the centre left spot of the same measurement is selected as ROI with a linear colour map as shown in fig. 6.13. There, hints to efficiency variations in the column and row direction are visible in the linear colour scale applied to that region of interest. Those regular patterns are not expected to be present in newer sensors of the MIMOSIS generation. Quite likely, an improved tuning of thresholds might be able to produce a more uniform efficiency across the full matrix of the sensor.

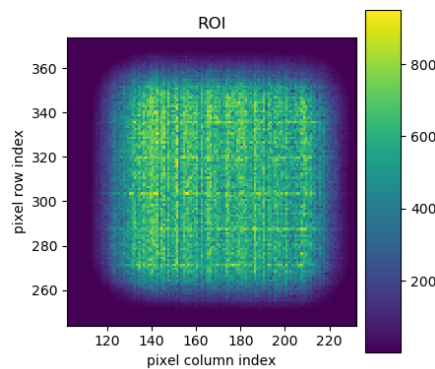


Figure 6.13: Close-up view of ^{55}Fe hits on a MIMOSA-26 sensor with linear colour scale. (It shows a different ROI of fig. 6.12 at col=175, row=310.).

6.3 Measurement: Cosmic Rays

This section describes a measurement performed with PRESTO to identify ‘cosmic muons’ – muons created in the atmosphere from the showers of cosmic rays – as introduced in section 3.4.2. The aim of this measurement is to measure the (projected) angular distribution of particles hitting a sensor of PRESTO during a background scan of 28 hours. Comparing the distribution to and the total number of hits to modelled expectations based on literature data of cosmic muons is done to evaluate the presence of the cosmic in the measured data.

6.3.1 Motivation

This measurement of cosmic is done for the following reasons: It can demonstrate the capability of PRESTO and its control system to be fully operational in data-taking mode and to record data reliably and continuously over the course of hours. Furthermore, the measurement allows to estimate the possibilities to use cosmic for commissioning of the final MVD, as its sensors will all be placed upright in the CBM experiment, due to the fixed-target geometry of the experiment.

Typically, a measurement of cosmic is done with multiple stacked detector planes. Instead, this measurement performed with PRESTO was carried out with a single MAPS sensor placed vertically (i.e. with its sensor plane normal oriented horizontally), as visualized in fig. 6.14 . The coordinate system (as defined in fig. 3.4) is oriented such, that the sensor plane coincides with the x - z plane of the coordinate system. Specifically, x points to the south-west, y points to the south-east and z points towards the sky.

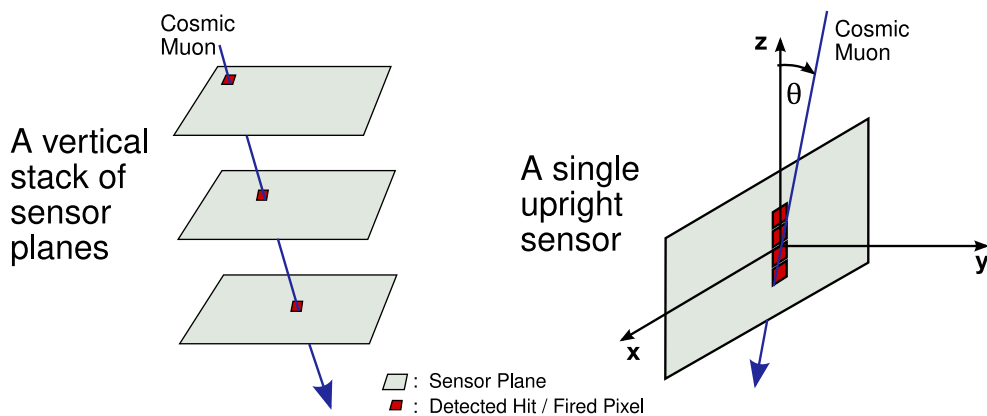


Figure 6.14: Typical setup to measure cosmic with multiple planes (left) vs. the single upright sensor approach used in this measurement (right).

This is an unusual approach to measure comics, that uses the coincidence of neigh-

bouring pixels firing in coincidence in a single frame. As cosmic rays follow a \cos^2 distribution and therefore are coming predominantly ‘from straight above’, they’ll leave a trace of subsequently hit pixels, provided they hit the very thin sensor at all.

As cosmic rays occur only relatively rarely, the main task of a cosmic ray measurement is to distinguish signals of cosmic rays from background (which might originate from electronic noise or natural radioactivity). Here, it helps that the background present in the laboratory is considered to have an isotropic angular distribution.

The measurement and data analysis (noise-rejection, cluster finding and directionality reconstruction) is accompanied by a theoretical modelling of the expected projected angular distribution of cosmic rays to allow fitting the data.

6.3.2 Modelling of the Expected Cosmic Muons

6.3.2.1 Expected Current of Cosmic Muons for a Single Upright MIMOSA-26 Sensor

In the case of the prototype “PRESTO”, the MIMOSA-26 sensors are in fact mounted in that vertical fashion inside the vacuum chamber. They have a sensitive area of $S_{m26} = 2.25 \text{ cm}^2$ each and can be approximated as two-dimensional objects as the sensitive volume of the sensor, the epitaxial layer, is only $\sim 15 \mu\text{m}$ thick [39, Fig. 20 p. 48], [175, p. 149]. (Modifying the effective surface term in the integral eq. (3.8) to represent a very slim 3d cuboid was tested but changed the final count rate only by about a percent as expected and was therefore considered irrelevant.)

Together with the peak flux of cosmic muons of $I_\nu = 70 \text{ m}^{-2} \text{ s}^{-1} \text{ sr}^{-1}$, cf. eq. (3.6), and the particle current density J_ν , cf. eq. (3.8), this yields a total particle rate of $\dot{N} = S_{m26} \cdot J_\nu = 0.74 \text{ min}^{-1}$ hitting a single MIMOSA-26 sensor, when it is oriented as depicted on the right side of fig. 6.14.

6.3.2.2 Expected Count Rate for Cosmic Rays with a Grazing Incidence

In the measured data, cuts on elongated clusters – representing particles hitting the sensor at grazing incidence – are used to improve signal over noise in the analysis, as introduced later in section 6.3.3.4. As a consequence, only cosmic rays with a grazing incidence angle onto the detector are accepted in the final data set. The incidence angle denotes the angle between the incident cosmic particle (muon) \vec{p}_c and the normal of the sensor surface \vec{n}_s :

$$\alpha_{\text{inc}}(\theta, \phi) = \angle(\vec{p}_c, \vec{n}_s) = \arccos\left(\frac{\vec{p}_c \cdot \vec{n}_s}{|\vec{p}_c| |\vec{n}_s|}\right) \quad (6.1)$$

with $\vec{n}_s \stackrel{\text{def}}{=} (0, 1, 0)_{xyz} = (1, \pi/2, \pi/2)_{r\theta\phi}$. With the known pixel pitch of $p = 18.4 \mu\text{m}$ and the assumed thickness of the epitaxial layer, which collects the charges, of $15 \mu\text{m}$ (see [39, Fig. 20 p. 48], [175, p. 149]), the minimum incidence angle creating elongated clusters of at least four firing pixels can be approximated. This is visualized in fig. 6.15. It shows the geometric details for this grazing incidence scenario. The incidence angle was calculated to be $\alpha = \arccos(t_e/(N \cdot p)) = \arccos(15/(4 \cdot 18.4)) \sim 78^\circ$, where p is the pixel pitch, N the number of firing pixels and t_e the thickness of the epitaxial layer. The number of four consecutive pixels was chosen somehow arbitrarily as producing cluster shapes with such an elongated shape, that allow for quite precise determination of the projected angle. Those considerations assume a simplified sensor model without diffusion of charge carriers to neighbouring pixels, instead all charges created in the epitaxial layer of the sensors are assumed to create a signal only in the pixel (collection diode) directly above. If a particle would hit the sensor at such an angle and traverse only pixels belong to a single column, the typical expected cluster shape would be 1x4. Diffusion of charge carriers might slightly dilute the cluster shape to e.g. 2x6 or similar, while still preserving the deformation (very elongated shape) of the cluster. Therefore, the simplification of ignoring diffusion of charge carrier in the sensor is valid in this case, where the main objective of the consideration is the analysis how a given incidence angle would translate to a noticeable cluster deformation.

This condition of $\alpha_{\text{inc}} \geq 78^\circ$ derived above is a further constraint on the geometric acceptance and can be expressed as the following filtering condition:

$$f_{\alpha_{\text{inc}}}(\theta, \phi) = \begin{cases} 1 & 78^\circ < \alpha_{\text{inc}} < 102^\circ \quad [\text{with } 102^\circ = 180^\circ - 78^\circ] \\ 0 & \text{otherwise} \end{cases} \quad (6.2)$$

This filter reduces the number of cosmic muons further and transforms 3.8 therefore to:

$$J'_v = \int_{\theta \leq \frac{\pi}{2}} I_v \cos^2(\theta) f_{\alpha_{\text{inc}}}(\theta, \phi) |\sin \theta \sin \phi| d\Omega \approx I_v \cdot \frac{\pi}{46} \text{sr} \approx \frac{1}{12} J_v \quad (6.3)$$

The integration above was done numerically using the Lebedev quadrature scheme⁴ of the 131st order with 5810 points distributed almost uniformly over the sphere's surface [176, p.2683] [178, 177]. (As the integration was done only over the hemisphere $\theta \leq \frac{\pi}{2}$ the effective number of numeric grid points was reduced to 2905.)

⁴The Lebedev quadrature scheme approximates surface integrals of a function f over the unit sphere $I[f] = \int f(\Omega) d\Omega = \int_{\theta=0}^{\pi} \int_{\varphi=0}^{2\pi} f(\theta, \varphi) \sin(\theta) d\theta d\varphi$ numerically by $\tilde{I}[f] = 4\pi \sum_{i=1}^N w_i f(\theta_i, \varphi_i)$.

After this cut acceptance/efficiency correction, **an event rate of accepted hits for identified cosmics of $\sim 3.7 \text{ h}^{-1}$ is expected.** This estimate on the expected rate of detected and accepted tracks (from muons of cosmic origin) is essential to

1. estimate the amount of time a measurement needs to run until a significantly high amount of cosmics is found in the data to justify the hypothesis, that cosmics are present in the data sample (e.g. by reproducing the known \cos^2 distribution),
2. determine whether a measured rate seems plausible,
3. (re-)scale simulations done with the MC method (see next section).

6.3.2.3 Simulating the Cosmics Muons with the Monte Carlo Method

A simulation using the MC (Monte Carlo) method was done to allow comparisons between simulated and measured particle incidence angle histograms. As only the angles of the muons play a role in the measurement (and not where the particle hits on the sensor), only the muon's track direction (θ, ϕ) had to be sampled in spherical coordinates.

- ϕ is drawn from the uniform function $f(\phi) = 1$ for $0 \leq \phi \leq 2\pi$.
- θ is drawn from $f(\theta) = \cos^2(\theta) \sin(\theta)$ for $0 \leq \theta \leq \pi/2$.
The sin term comes from the differential solid angle when integrating the desired $j(\theta, \phi) \sim \cos^2(\theta)$.

After establishing the sampling scheme, it is worthwhile to check whether the sampling actually reproduces the expected angular flux density distribution.

This is done as shown in fig. 6.16 with a series of conical search windows along a series of directions from $(90^\circ, 0^\circ)_{\theta\phi}$ to $(90^\circ, 90^\circ)_{\theta\phi}$ with cone half angles of $\alpha = 5^\circ$ and therefore solid angles of $\Omega = 2\pi(1 - \cos \alpha) = 0.024 \text{ sr}$. This is presented in fig. 6.16a. The chosen cone half angle α for this validation is a compromise: a smaller value would help to reproduce the sampled function more precisely but would reduce the number of samples that match the search window, thus adding noise to the verification. fig. 6.16b shows the result with data points at every 5° of θ , revealing that indeed, the $\cos^2(\theta)$ distribution for $j(\theta, \phi)$ has been properly sampled.

In the next section, cuts are applied to the sampled cosmics to mimick the acceptance of the measurement.

6.3.2.4 Expected Distribution of Projected Angles from Monte Carlo Modelling

This section details how the MC sampled cosmics from the previous section are subjected to cuts mimicking the acceptance of the detector, to model the angular distribution of cosmic muons (projected onto the sensor plane).

Two cuts were performed on the sampled cosmic muons. One cut, here called cut ‘cut **S**’, excludes any cosmic muons which will not hit the sensor surface at all, effectively mimicking the geometric acceptance of the detector; the other ‘cut **I**’ excludes muons which hit the sensor at an incidence angle which won’t create a characteristic pattern of neighbouring pixels allowing to reconstruct their direction. Figure 6.17 shows how those cuts reduce the statistics starting with a total number of randomly sampled cosmics of $N = 1.00 \times 10^8$. The numbers show that the two cuts do not cut on fully independent categories but rather remove complementary parts of the full distribution as the removed percentages are much higher, when the other cut was already applied.

The results of the Monte Carlo simulation can be found in fig. 6.18. Figure 6.18a shows all 10^8 cosmic muons, which were randomly sampled as (θ, ϕ) tuples from the pdf (probability density function) $\propto \cos^2(\theta) \sin(\theta)$ between 0 and $\frac{\pi}{2}$ and a uniform distribution between 0 and 2π , respectively. The following plot, fig. 6.18b, has the cut applied that accounts for the geometric acceptance of the sensor being mounted vertically with an almost 2d surface projection. Finally, fig. 6.18c has both cuts included, the sensor surface cut accounting for the sensors geometric acceptance as well as the cut on the incidence angle requiring particles to hit the sensor at grazing incidence (min 78° to sensor surface normal, see eq. (6.2) and fig. 6.15). In addition to the main histogram θ vs. θ_{xz} , the upper right corner in each plot shows a small additional polar plot where θ is plotted against ϕ . This provides further helpful insights, which directions in the sky / northern hemisphere the cosmic particles can come from to survive the cuts.

The θ_{xz} distribution that can be found on top of each plot is the relevant observable in the measured data. The shape of this distribution is invariant to the cuts applied.

The event rate of accepted and identified cosmics in a single upright MIMOSA-26 detector was already estimated before in section 6.3.2.2 to be $\sim 3.7 \text{ h}^{-1}$. This section, however, is focusing on the expected shape for the distribution of the θ angles onto the sensor surface: While the $\cos^2(\theta)$ angular distribution of cosmics was introduced in section 3.4.2, quite surprisingly, it still holds for the distribution for the projection of the particles’ trajectory onto the sensor plane, which seems to follow $\cos^2(\theta_{xz})$. And this still holds after applying the cut, that accepts only particles with a grazing incidence, without affecting the shape of the distribution. This insight is used later when fitting the data in section 6.3.4.

Assuming that a good reproduction of the \cos^2 distribution requires a number of 100 cosmics, this would require a measurement campaign of ~ 27 hours.

6.3.3 Data Processing

Following the modelling of the expected count rate of cosmics and their angular distribution in the previous section, this section now details the data processing steps performed on the measured data.

6.3.3.1 Discarding Fake Hits to Improve Signal-over-Noise

An important way to reduce background in the measurement is to only consider clusters of firing pixels, not individual firing pixels. This is needed as the expected flux of cosmic particles is rather low compared to the number of noisy pixels firing continuously or randomly. Sources of such noise can be external sources such pickup from external electromagnetic contamination (e.g. vacuum pumps, power supplies etc.), as well as intrinsic sources such as thermal electronic noise in the pixel amplification circuits or RTS (random telegraph signal) noise commonly found in CMOS sensors [43, p. 47]. Typically on a MIMOSA-26 sensor, this noise turns into a fake hit rate of $\approx 1 \times 10^{-4}$ per frame. For its $1152 \cdot 576 = 663552$ pixels, this results in roughly 6×10^5 fake hits per second (at 8680 frames per second), where the total particle rate calculated in section 6.3.2 was used as signal. The SNR is therefore estimated to be $0.74 \text{ min}^{-1} / 6 \times 10^5 \text{ s}^{-1} = 2 \times 10^{-8}$. For an analysis with statistically significant results, it is therefore vital to improve the SNR.

The most important approach to reduce the electronic noise in the subsequent data analysis is a cluster size analysis: For electronic noise, typically only individual pixels are firing, while hits from MIPs have a rather high probability to create clusters of more than one firing pixel. The average cluster size increases with grazing incidence angles and with lower threshold voltages [179, Fig. 4]. With the sensor orientation as shown in fig. 6.14, most of the particles which come from cosmic background will hit the sensor at incidence angles $> 45^\circ$, thus typically creating a cluster of hit pixels instead of a single firing pixel. Thus, by excluding any hit on the sensor where only a single pixel is firing (i.e. requiring a cluster size of $N_p x > 1$), the SNR can be greatly improved.

6.3.3.2 Clustering Algorithm

A cluster is the set of neighbouring pixels firing at roughly the same time. Clusters typically appear in the data when physical hits occur, but sometimes also they only

represent pure noise. In the case of physical hits, a cluster can occur if charge is shared between neighbouring pixels due to a) electron diffusion in the epitaxial layer or b) particles hitting the sensor at an angle so that the charges are already created in different pixels. In addition, particles with very high energy deposition (such as slow heavy ions) can create a huge amount of electrons, resulting in very large clusters, cf. fig. 3.3.

In the following sections, a cluster is defined as the set of at least two pixels where all pixels are within a defined distance (i.e. clustering threshold) from their nearest neighbour.

The agglomerative algorithm implemented for this analysis is very similar to DB-SCAN[180] and works as follows for each frame of data:

1. **pairing:** determine all combinations of two fired pixels in the frame,
2. **distance calculation:** calculate the Euclidean distance between the centres of the two pixels for each pair,
3. **pixel pool determination:** place all pixels from pairs whose distance is smaller than the chosen clustering threshold into a (single) pool of pixels to be distributed to clusters,
4. **cluster forming:** while there are still pixel in the pool:
 - a) **start new cluster:** begin a new cluster by taking out a random pixel from the pool,
 - b) **add pixels belonging to cluster:** determine any pixel from the pool that's close to the pixels in the cluster; add it to the cluster and repeat checking until no more pixels in the pool (if any) are close to any of the cluster's pixels.

This algorithm is straight forward to implement. When running at high luminosities/when the frames are rather full, its time complexity of $\sim O(n^2)$ would require optimizations⁵.

There are a couple of well-established alternatives to implementing a custom clustering algorithm:

⁵The time complexity is largely dominated by the pairing process which is $O(\binom{n}{2}) \sim O(n^2)$. A large optimization of the algorithm's runtime would be achieved if the pairing was done in a "divide and conquer" scheme, were the pixel matrix is divided into (slightly overlapping) sub-areas and the pairing would only have to be done for each sub-area. In addition, repeated distance calculations for pixel pairs can be cached with a look-up table.

- DBSCAN⁶ [180] and
- OPTICS[182].

In previous research, a FPGA based cluster finder was implemented for the MIMOSA-26 sensors [179].

Figure 6.19 shows which hit patterns are accepted as clusters with the definition above. In this example, any set of two pixels with a centre to centre distance of less than 1.9 times the pixel pitch is considered “close” (regarding step 4. b) of the algorithm described above). This means that not only pixels with adjacent sides are “close” but also diagonal ones, where only the corners coincide.

6.3.3.3 Further Background Reduction using Noisy Cluster Identification

The cut presented in [[sec:snr-improvement-using-cluster-analysis](#)] to only consider clusters for the final analysis, not single firing pixels, does not yield enough reduction of background. Rare coincidences of noise in neighbouring pixels can create fake or ‘noise’ clusters. This section discusses how those clusters were identified.

Definitions:

- **noise cluster:** every cluster consisting of 50% or more pixels tagged as ‘noisy’.
- **signal cluster:** every cluster consisting of less than 50% pixels tagged as ‘noisy’.

For this purpose, any pixel is assigned the tag ‘noisy’ if it fired more 3 times during 7 hours of background measurements. As a result, approximately 0.7% of all pixels were tagged as ‘noisy’. Figure 6.20 shows examples for clusters being accepted or rejected based on the definition above.

Figure 6.21 shows how effective this method is. The upper central plot in the figure is the result of a background and noise measurement over the course of 7 hours. Only clusters reconstructed as described in the previous section are shown there. In the second step of removing any clusters with $\geq 50\%$ noisy pixels, entire areas containing obvious noise (noisy rows at the bottom, clusters due to hot pixels, thin periodic structures every 16 rows) were removed. The remaining sample of clusters classified as “signal” (lower right in the figure) mostly contains large (and often elongated) clusters.

⁶Especially the DBSCAN algorithm could prove useful, as its implementations are well-established and proven, for example in the machine learning toolkit scikit-learn under the name `sklearn.cluster.DBSCAN` [181]. A good comparison of clustering algorithms in scikit-learn can be found at <https://scikit-learn.org/stable/modules/clustering.html#overview-of-clustering-methods>.

6.3.3.4 Reconstructing the Projected Particle Direction from the Cluster Shape

When multiple neighbouring pixels are firing due to a particle hitting the sensor at an angle, shape of the produced cluster of pixels can allow to reconstruct the direction of the particle as projected onto the sensor. In such a case, an elongated cluster will be produced – the longer the more grazing the incidence of the particle will be. Identifying elongated clusters and determining their orientation can be done by calculating the second moment of the (two-dimensional) area the pixels of each cluster are covering. (This is similar to the calculation of the moment of inertia, except that no mass is involved.)

The second moment tensor of an area of arbitrary shape is given by

$$\mathbf{I} = \begin{bmatrix} I_{11} & I_{12} \\ I_{21} & I_{22} \end{bmatrix} \quad (6.4)$$

with its components being defined as:

$$I_{ij} \stackrel{\text{def}}{=} \int_A w (\|\mathbf{r}\|^2 \delta_{ij} - x_i x_j) dA, \quad (6.5)$$

where $w(\mathbf{r})$ is a weight or density (analogously to a mass when calculating to moment of inertia), \mathbf{r} is the location vector, δ_{ij} is the Kronecker delta, x_i is the i -component of the location vector, and A is the area over which to perform the integral. The density w is defined so that $\int_A w dA = 1$ for a pixel that was hit, as binary pixels do not provide access to continuous (analogue) values. This integral may naively be approximated using the following discrete form:

$$I_{ij} \approx \sum_{k=1}^N w_k (\|\mathbf{r}\|^2 \delta_{ij} - x_i x_j) \quad (6.6)$$

where k is the index for the pixels which are part of the cluster, and w_k is a weight assigned to each pixel. For each cluster, the moment of inertia tensor is calculated by summing over its fired pixels by regarding them to have a “weight” w_k of 1 centered at the pixels midpoint. The use of this discrete formulation allows it to be computed efficiently with vectorized algebra. This numeric approximation of a pixel as a point like object is not very realistic, however. In order for the discrete approximation to be as equivalent to the integral form as possible, the discretisation grid needs to be much finer than the pixel width. Technically, this can be achieved, by scaling up the binary image representation of a cluster with nearest neighbour interpolation. Figure 6.22 shows the idea of this upscaling from the original cluster (left) to a 2×2 scaling and a 4×4 scaling of the image representation

of the cluster. It shows how the numeric accuracy can be improved when using image processing algorithms for numerical/discretised approximations. This is further demonstrated in Figure 6.23, which that how the up-scaling improves the precision of the numeric approximation of the second moment calculation with increasing scaling factors. Starting at about an increase of 10×10 the value is within 3σ of its asymptotical/exact value.

Examples for the reconstruction of projected particle directions via the analysis of the cluster deformation / directionality are shown in fig. 6.24. The determined matrix elements I'_{row} and I'_{col} in the figure are identical to I_{11} and I_{22} in the preceding paragraphs, respectively.

6.3.4 Result

Figure 6.25 shows the distribution of projected particle trajectories, i.e. the reconstructed impact angles projected to the sensor plane θ_{xz} ('cluster angle'). The number of clusters is plotted on the left axis as blue bars for each of the 10° wide bins. A total of 377 clusters survived the data selection process, which required cluster of at least 5 pixels and a deformation (I'_{22}/I'_{11}) of at least 4. A cluster angle of 0° means the cluster is oriented skywards in the laboratory frame. In addition to the measured data, the dashed line shows a fit with a function modelling a constant background (parameter B) independent of the cluster angle and a \cos^2 dependent component of signal (parameter S).

6.3.5 Interpretation

The measured distribution presented in fig. 6.25 shows a clear clustering of entries around 0° . The data is highly consistent with a \cos^2 distribution centred around 0° , which is subject to statistical fluctuations due to the limited statistics available. The data as a whole appears to be composed of an angle-independent baseline component and the $\cos^2(\theta_{xz})$ dependent component. The measured distribution was fitted to separate those components. The constant (angle-independent) background was found to contribute $\approx 203 \pm 49$ counts to the data, while the $\cos^2(\theta)$ dependent component contributes $\approx 174 \pm 40$ entries. The latter hits (clusters) are attributed to a cosmic origin.

Due to the expected event rate of accepted hits for identified cosmics of $\sim 3.7 \text{ h}^{-1}$, see section 6.3.2.2, a total of 104 cosmic muons is expected for the measurement. The fitted number of particles that follow the $\cos^2(\theta)$ exceeds the estimated rate of cosmic muons slightly. The measurement seems to be in good overall agreement with the hypothesis that the remaining sample of clusters do indeed come from

cosmic muons.

The overall shape of the distribution seems to be in good agreement with the Monte-Carlo simulation in section 6.3.2.4, which showed that the measured projected angles also follow a \cos^2 distribution, which seems to be present in the data. Where the $\cos^2(\theta)$ distribution does not match perfectly are at the bins centered at around 0° , which are slightly overpronounced. A possible explanation for such a modification of the angular distribution could be the concrete roof/ceiling of the laboratory. Depending on their polar angle, muons have to traverse a different distances of concrete (proportional to $1/\cos(\theta)$ in the model of a fixed thickness, large and uniform ceiling layer).

After all, the hypothesis that of the 377 particle hits (clusters) in the final data set, the \cos^2 part of 174 ± 40 entries can be attributed to cosmic muons is being accepted. The overall conclusion is that the measurement results are consistent with the expected angular distribution. While a higher number of secondary muons was measured than estimated using MC sampling, the mismatch can likely be explained with the slightly hand-waving criterion of requiring four consecutive pixels to be hit. If this would be relaxed to ~ 3.5 consecutive pixels, the experimental value could be reproduced better.

Consideration of Other Sources of Clusters in the Background Measurement

While approximately 46% of the clusters in the final data set are attributed to cosmic muons, the nature of the remaining clusters present in the data leaves room for further interpretation. They are most probably from natural radioactive background, which is why they don't show an angular dependence. Possible sources for radioactive radiation might be:

- Contaminants might be present in the semi-conductor die of the MIMOSA-26 sensor.
- Traces of radioactive isotopes might be present in the carrier for the sensor, a TPG-plate. It contains near 100% carbon atoms. However, a large amount of ^{14}C seems unlikely, as it is claimed to come from "hydrocarbon gas, such as natural gas" [183].
- Radon atoms might still be present on the surface of the vacuum vessel.

Telling apart those possible sources of background radiation in the data seems impossible with the current experimental setup. However, they provide a plausible explanation for the measured background data overall.

6.3.6 Implications for the Operation of the MVD

While it's certainly impressive that the long-term data recorded with a single sensor is capable of (statistically) finding muons from cosmic events, the rate of such events (or background events in general) is extremely small compared to the occupancies expected when measuring particle collision at the foreseen interaction rate. Therefore, no relevant contamination of the measured physics data is expected from cosmics / natural background.

The geometric arrangement of the acceptance of the four MVD stations complicate the use of cosmic muons for alignment purposes very much. The measurement in the section above was relying on coincidence between neighbouring pixels in individual sensors. Coincidences between different sensors are very unlikely, unless the detector's beam axis will be pointed skywards, due to the \cos^2 distribution of cosmic muons. If the entire detector assembly could be positioned in such a way, however, cosmic rays could well be used for alignment of the sensors with stiff tracks from the higher energy part of (secondary) cosmic radiation.

When the detector is placed in its regular orientation, directional clusters are expected in individual sensors, however. When commissioning the detector, those clusters lead to a small stream of data in the DAQ so that clustering algorithms can be monitored/tested even when no beam is available.

A measurement of the background radiation with the MVD might be used to monitor electronic noise and the sensitivity of the sensors periodically, e.g. every 30 minutes. However, depending on the level of activation of CBM detector components directly after a run, a background measurement might prove useful for radiation monitoring, rather than monitoring electronic noise of the sensors.

In the light of CBM being a high-rate experiment with high amounts of activation in the detector material expected during measurement campaigns, any natural background radiation will only represent a tiny fraction though, and can also safely be neglected. Therefore, investigating the origin of natural background radiation further is not recommended.

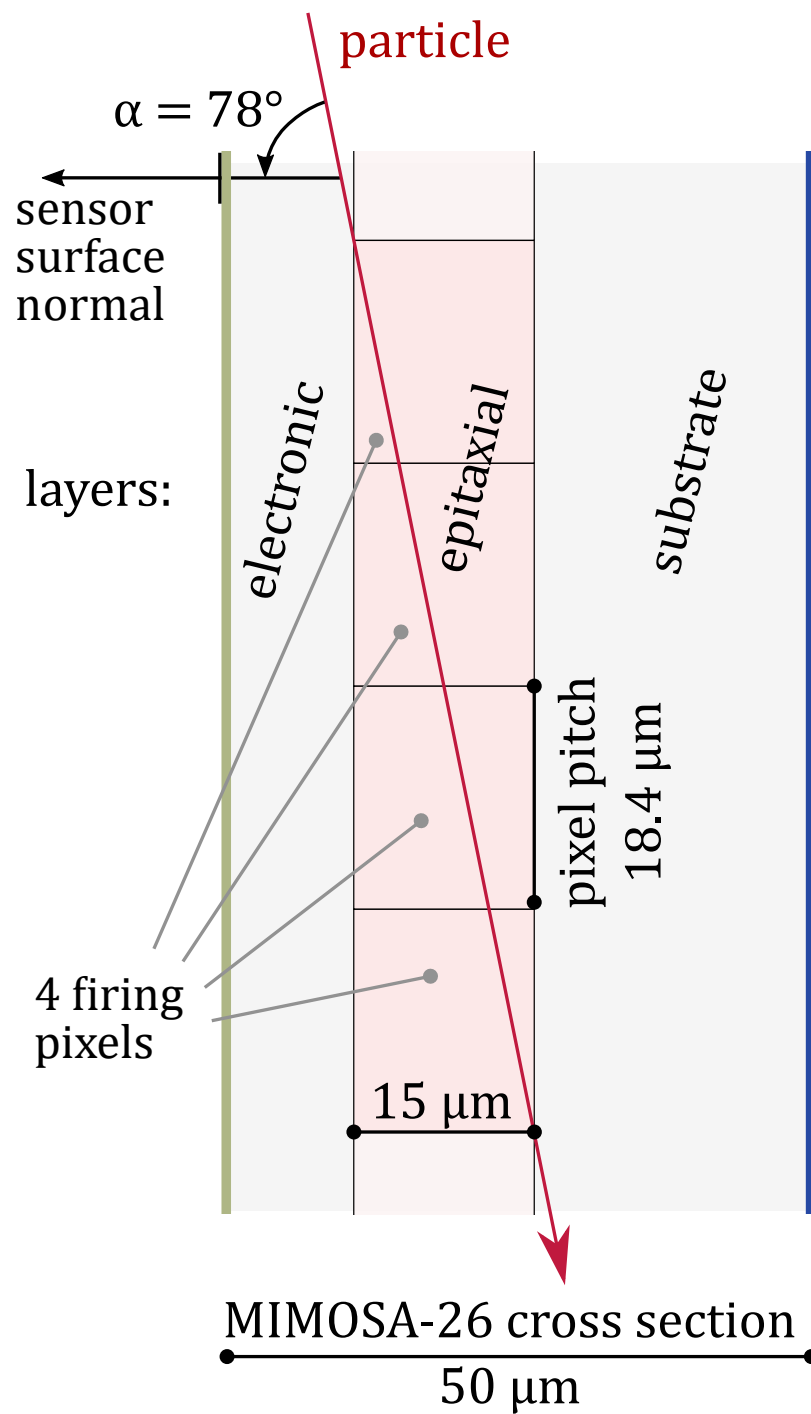


Figure 6.15: Geometric considerations of the grazing incidence angle needed to create a cluster of four consecutive pixels in MIMOSA-26 sensors.

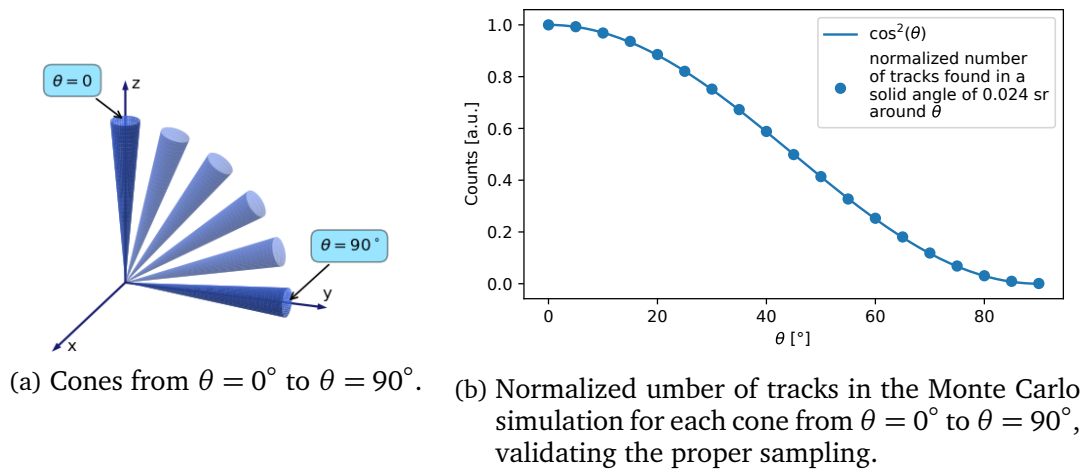


Figure 6.16: Verification of the MC sampling scheme for cos² θ distribution.

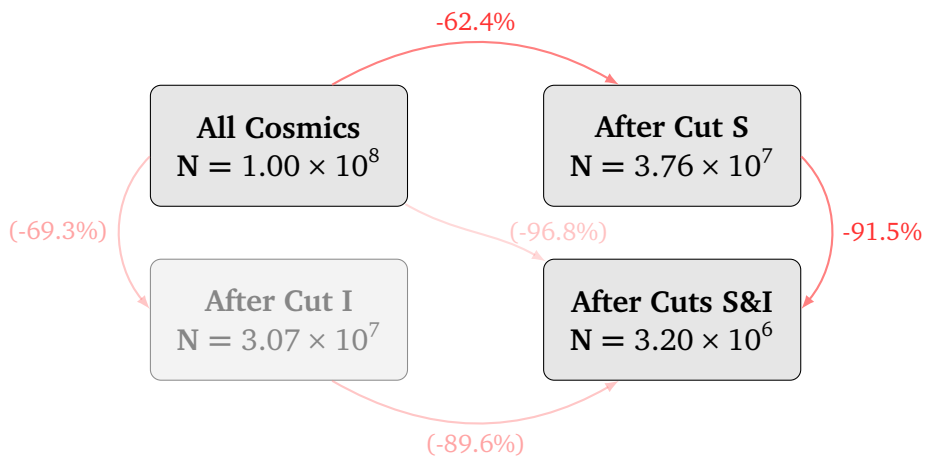
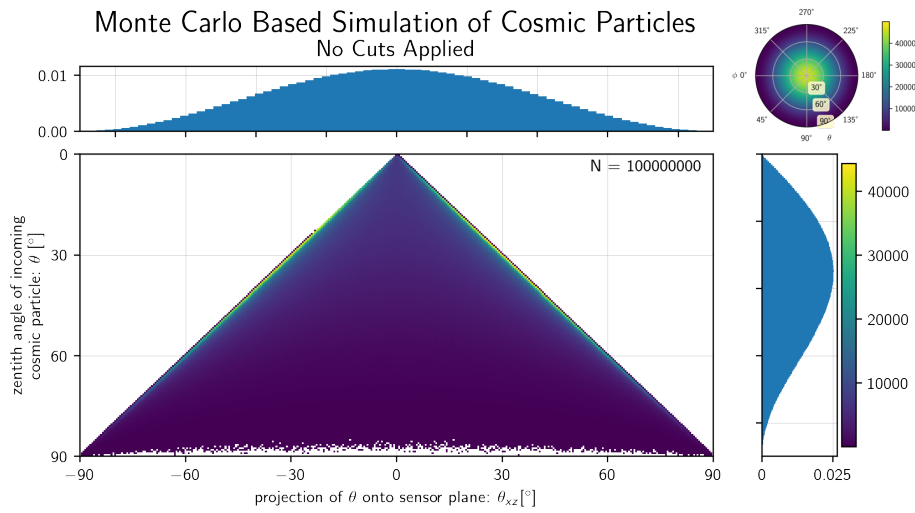
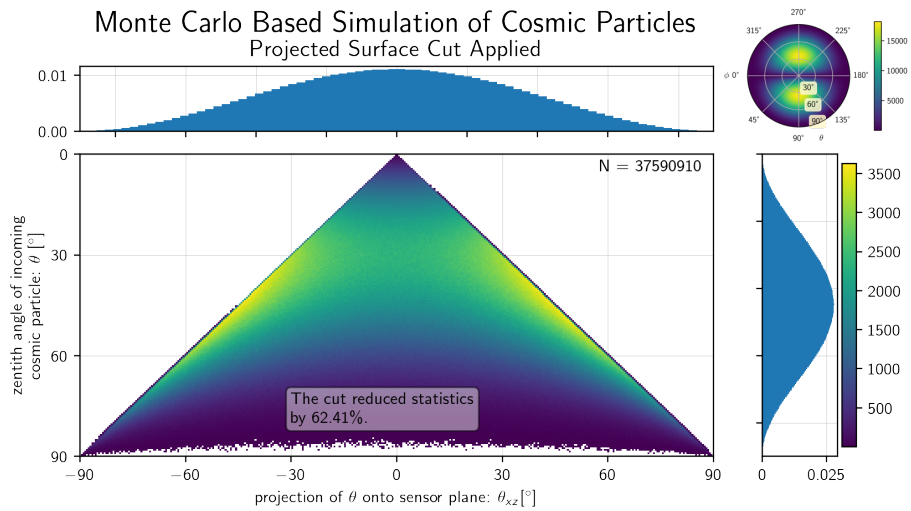


Figure 6.17: Effect of the cuts on the number of cosmic surviving in the MC data.

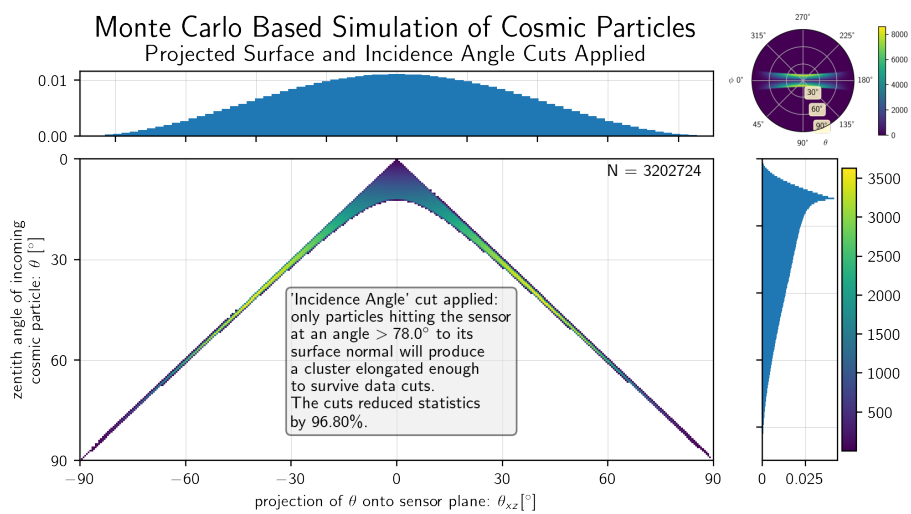
6 Field Tests



(a) Without any cuts applied.



(b) After cut **S**.



(c) After cut **S** and **I**.

Figure 6.18: Monte-Carlo simulation of cosmic muons distributed according to $\cos^2 \theta$ with cuts modelled to mimic geometric acceptance (**S**) and the ability to reconstruct cluster directionality dependent on the incidence angle (**I**).

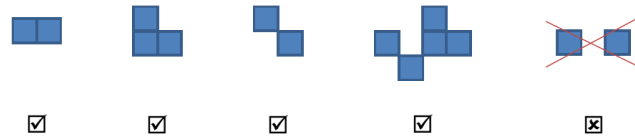


Figure 6.19: Merging fired pixels to clusters: example hit patterns (based the pixels' centre-to-centre constraint $d_{\max} < 1.9$ the pixel pitch).

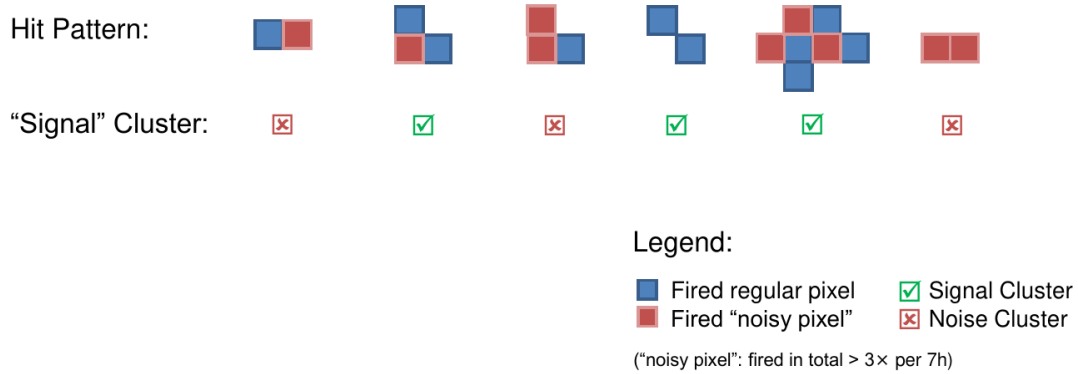


Figure 6.20: Rejection of clusters involving $\geq 50\%$ noisy pixels.

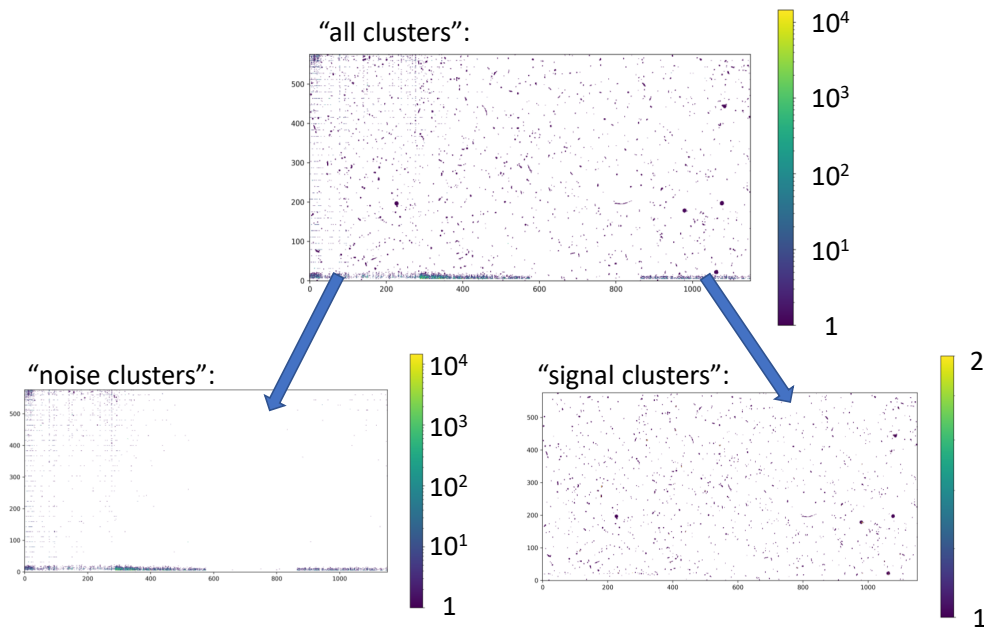


Figure 6.21: Visualization of the effectiveness of cluster classification (noise/signal).

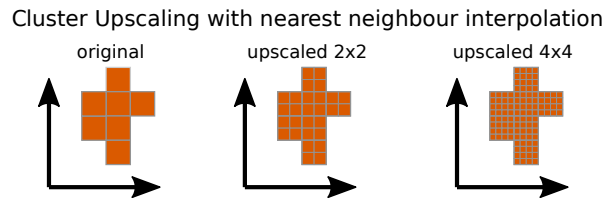


Figure 6.22: Visualization of the discrete upscaling of a cluster shape.

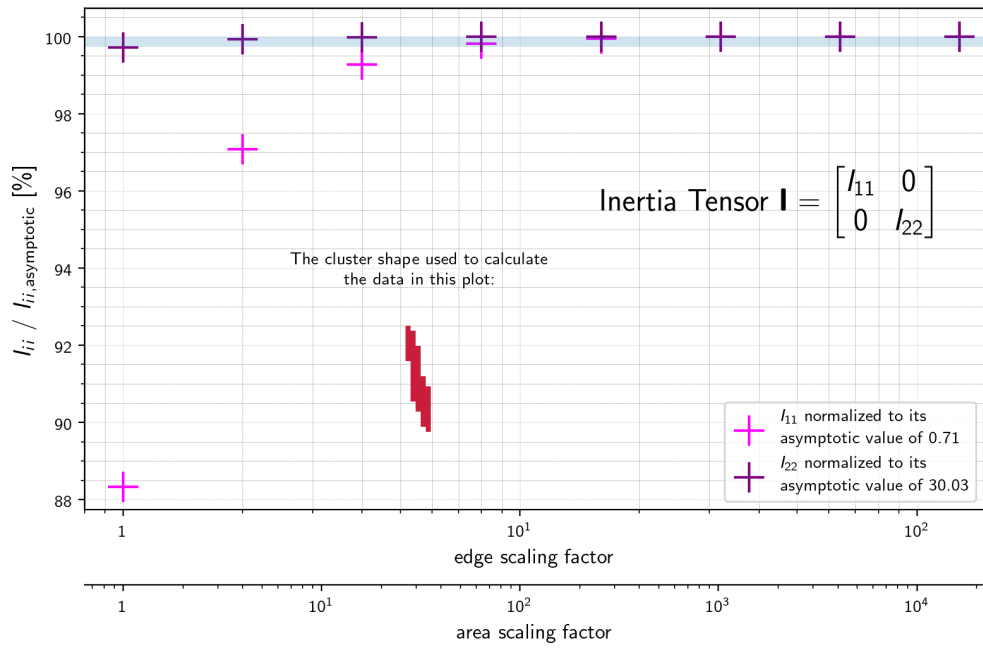


Figure 6.23: An example for how upscaling a binary cluster image using nearest neighbour interpolation helps determining its moments of inertia precisely.

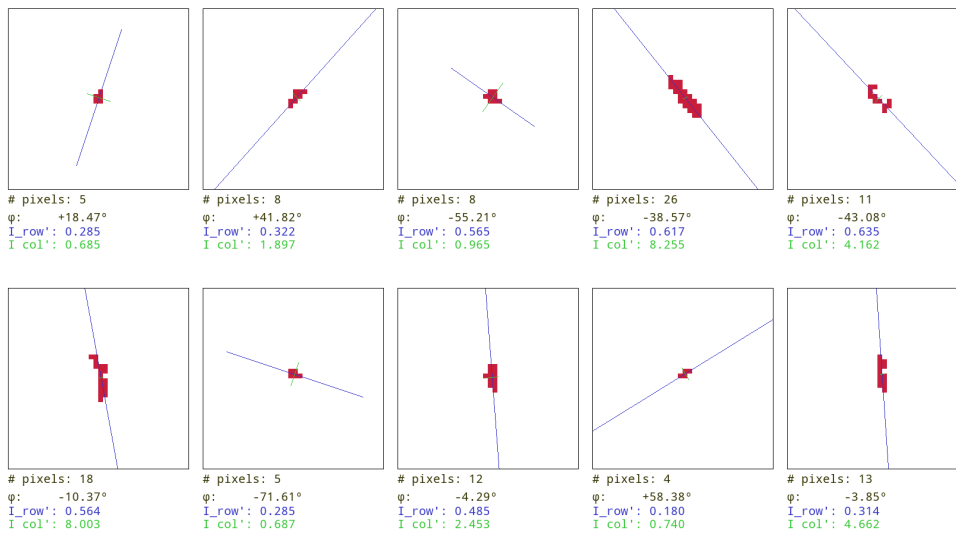


Figure 6.24: Examples for reconstructed projected particle directions from clusters encountered in the data.

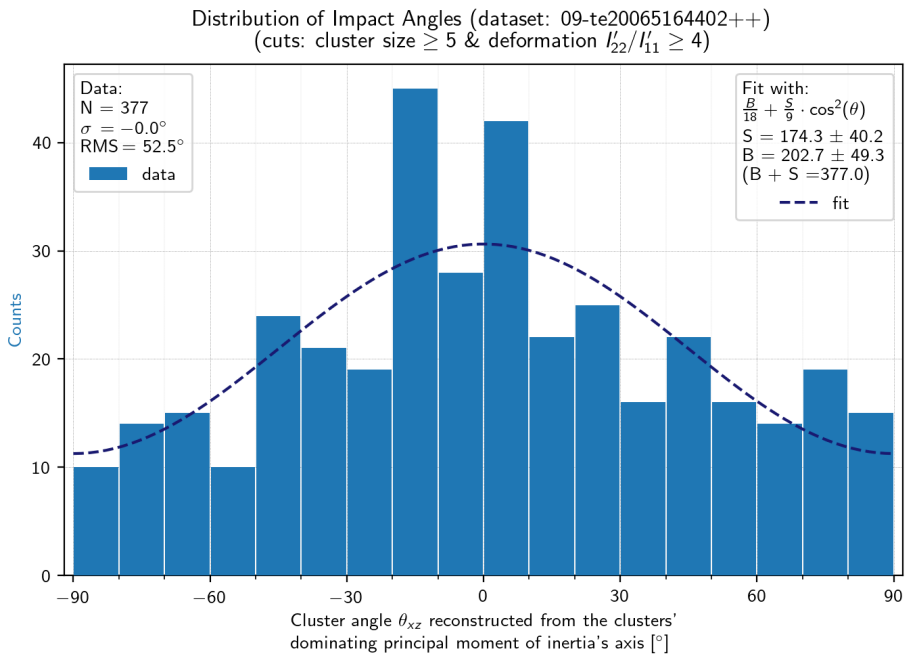


Figure 6.25: Measured distribution of cluster angles.

6.4 Opportunities for Improvement of the Control System and Continuous Operation

Summing up the above field tests, the prototype PRESTO was successfully validated in continuous vacuum operation with all vital parameters monitored and the overall system controlled with a dedicated EPICS control system, that proved to be reliable and responsive. Nevertheless, various aspects regarding the operation of the detector prototype and its control system were encountered that call for improvements. While many of these were addressed and solved in this thesis (e.g. evaluating higher performance options for archiving & retrieval, ability to archive DAQ related PVs through the PyTrbNet EPICS bridge), other aspects remain to be addressed and should be considered for the future Micro Vertex Detector.

The following aspects provide a few very specific ideas concluded from the experience with the PRESTO control system, which could improve a future control system for the MVD:

- When the work on PRESTO's control system was started, EPICS 3.15 was the latest stable release. In the meantime, EPICS 7 was published [184] expanding its capabilities in many ways. It is being considered for upgrades of the IOCs.
- Failure detection through vibration sensors
Mounting a vibration sensor to the structural frame of the detector. This might enable the detection of a failing coolant pump or vacuum pump. In addition, having access to the typical vibration amplitude and frequencies might help understanding alignment parameters and their precision.
- Increased safety with mechanical interlocks
Mechanical interlocks should be added to all electric interlock systems in order to also work in the case of a power outage / network partition / software failure.

Furthermore, ways of improving the future control system based on slightly more abstract concepts are presented in more detail below.

Machine Learning in the Context of Control Systems

Depending on the amount of autonomous operation, control systems have traditionally implemented more and more complex finite state machines. As an alternative to those increasingly complex finite state machines with a large number of states and transition rules, machine learning approaches could help to control the machine if good training data is available. First attempts of using machine learning in the context of control systems for accelerators and experiments have been published cf. [188, 190, 186, 187, 185, 189]. Specifically, the structured

6.4 Opportunities for Improvement of the Control System and 24/7 Operation

datatypes of EPICS 7 are intended to assert data consistency when correlating different process parameters via machine learning [184, p. 1217]. “The size and scale of hardware involved and telescope high availability requirements suggest that machine learning and other automated methods will be required for fault finding and fault prediction of hardware components.”[190] Another recent publication mentioning machine learning in the context of EPICS is [191] (section on machine learning methods).

Message Passing

If data from the EPICS-based DCS is needed in other subsystems of the experiment (e.g. DAQ or ECS), message passing protocols and software might be a good option. The ESS (European Spallation Source), for example uses Apache Kafka as a distributed event streaming stream platform with Google FlatBuffers being the selected message serialization format for the EPICS events [192, 193, 194, 195]

Logging Parameter Changes (*caput*)

Parameter changes – or *caput* operations in the EPICS / CA terminology – are requests for the control system to change DCS parameters. This could be a small event such as a threshold being adjusted or a larger event, such as a request to transition to a different detector state (e.g. OFF to IDLE). Logging any requests to change is mandatory. It can provide information about who requested the change and from which network host. Keeping a log of this data is essential to assure traceability of issues in case of detector or experiment malfunction.

This data could be logged outside EPICS to software commonly used to fulfil such tasks in the context of data-centre operations. An example for such a scheme would be sending the relevant data to `fluentd`⁷ which would then forward the processed information to `Elasticsearch`⁸ for easy retrieval and searchability.

The field tests documented in this chapter were able to demonstrate the capability of PRESTO to be operated continuously in vacuum, thereby validating both, the integration concept of PRESTO and the ability of the control system to guarantee the safe operation of the prototype.

In the last section of this chapter, important insights to advance the control system gained during the operation of the prototype were stated. The next and final chapter extends upon this topic and discusses a further and more fundamental ‘modernization’ of the EPICS control system framework.

⁷`fluentd` is a data collection software that analyses event logs and data streams.

⁸`Elasticsearch` is a distributed search engine that provides an HTTP web interface and schema-free JSON documents.

7 Modernizing the EPICS Control System

What started with the containerization of multiple EPICS components as described in section 5.7 quickly emerged as an important topic with a wider scope. As further options of modernizing the EPICS software components were evaluated, it was evident that a more general overview on those perspectives would be desirable, based on the experience gained with the control system for the prototype PRESTO (introduced in section 4.2). This chapter sums up those findings.

Application modernization is a term that commonly refers to the process of transforming legacy software to the current state-of-the-art. The term “modernizing” was introduced in [196] as one of three activities in the field of software system evolution and later refined in [197]. With the rise of several trends in IT – like DevOps, cloud computing and containerization – the meaning of the term was slightly redefined and now refers to the transformation of a monolithic legacy application to a (modern) scalable platform and architecture [199, 200, 201, 203, 198, 202]. In this chapter, such a modern definition is used, where application modernization aims for:

- Modularization – breaking down a monolithic application into components with separate concerns – microservices – that can run independently with only few inter-dependencies
- Implementation of DevOps Techniques – This may include topics such as Infrastructure as Code, Continuous Integration & Deployment
- High Availability (HA) – providing reliable operation, desirably even during upgrades
- Virtualization – abstracting the software from hardware (dependencies)
- Containerization – building applications in a reproducible way and shipping them in a self-contained environment

It is a major task to achieve these goals with reasonable effort, which typically requires changes from adjusting existing components to entirely rewriting and restructuring them. In the specific case of a complex EPICS control system, some of these objectives are easy to achieve, because they are already a key concept of

the system: Already in the current form, EPICS is highly suited to create a distributed software system, thus covering the modularization aspect already very well. Implementing HA (high availability) of critical components can help achieving exceptionally high uptimes where outages would be costly or even put the experiment and scientific discoveries at risk. It is very important when operating a control system, but potentially overlooked for many physics experiments due to the high effort required to achieve it. In specific deployment schemes, HA can already be achieved with today aspect might be covered depending on the deployment, too. The other goals of application modernization require more effort.

The following sections describe and evaluate a possible application modernization strategy for the EPICS control system.

7.1 Technological Enablers

Various new developments in the last decade(s) play an important role for the EPICS modernization. With the huge pace of today's network and web technology innovation, a lot of technologies that seemed futuristic and prototypical a few years back are now ubiquitous in data-centres around the world and play a key role as technological enablers for this study. They are mostly industry standards in the world of computer science.

7.1.1 Technology: VLAN (VXLAN)

VLAN – or more specifically IEEE 802.1Q – is a network virtualization method allowing to send packets belonging to different logical networks (i.e. broadcast domains) over a shared physical connection. This is done by wrapping the original packet with the 4-byte IEEE 802.1Q header (also called ‘VLAN tag’). A network port intended to send and receive those wrapped packets from multiple logical networks is commonly called “trunk port”. Unwrapping those packets can be done by managed switches or on the software layer of endpoints (e.g. the Linux kernel).

In the case of the newer VXLAN (Virtual Extensible Local Area Network), which is commonly used in cloud computing environments, the original packets are encapsulated at OSI layer 2 within UDP packets (OSI layer 4). While this technology can potentially provide even more flexibility (for example for networks distributed across multiple sites), it is not yet commonly found in commodity hardware switches and thus wasn't considered further.

Common use cases for VLAN comprise:

- Connecting a router to the multiple networks with a single ethernet cable.

- Bundling the DAQ data network and the controls network in a single ethernet cable in a temporary setup for a beam time.

Use Case for EPICS

In the case of the EPICS control system, careful network segmentation can help to keep mostly self-contained systems isolated. Specifically all subsystems (like detectors, interlock-system, magnet, beam-dump, etc.) can have their own controls network (called the “field layer”). The “supervisory layer” combines the state from all subsystems into a global system state for the full experiment. Each of those different networks (i.e. broadcast domains) should be assigned its own VLAN. This allows to use a smaller number of ethernet cables as managed switches can exchange tagged packets from all these VLANs through a single link, a concept also known as trunking. Furthermore, this can drastically simplify the debugging workflow for control system specialists in cases where they need direct access to the individual field networks. In that case, the trunk link of a single Ethernet cable can provide access to all individual controls networks present in the experiment.

7.1.2 Technology: Container Virtualization (e.g. Docker)

Container virtualization has a long history in the Unix/BSD world. The introduction of **chroot** (in 1979) was a first step towards process isolation. Later notable implementations include OpenVZ, FreeBSD Jails and LXC (Linux Containers). Today, the de-facto standard for container virtualization are Docker containers or its generalizations OCI (Open Container Initiative) Image and Runtime.

The engine running docker containers makes strong use of a Linux kernel feature called cgroups, allowing to restrict the resources available to a process. This is combined with namespace isolation, allowing to restrict visibility of various system aspects for sandboxed processes.

The reasons for using container virtualization can be manifold. They might comprise sandboxing a process, documenting the steps necessary to get to a valid software environment, running software depending other library versions than available on the host machine, and many more (list not exhaustive).

Typical use cases are:

- Distributing ready-to-run applications to other hosts,
- documenting all required steps to set up an application in a ‘Dockerfile’,
- leverage horizontal scalability by running multiple container instances,
- reducing application development lead times by deriving from predefined 3rd party environments (e.g. Java OpenJDK),

- running specific software versions that are otherwise not available via the system's package manager,
- rebuilding the entire application daily and whenever changes to the source code are made, which helps to identify issues with new source code or updated library components early,
- simplify and/or automate the deployment and rollback of applications,
- deploying cross-compiled applications on less powerful hardware with different CPU architectures (e.g. single board computers).

Use Case for EPICS

Providing the EPICS base software (including frequently used extensions) in a container has a lot of advantages. The container image contains all software dependencies the EPICS system depends on in the required version, which allows to deploy the container image even years later, when a specific version of the dependency might not be available anymore. Furthermore, having a general EPICS base image available allows to create different EPICS applications which simply derive from that base image, reducing set-up times and boilerplate code.

In the case of related software such as archivers, alarm handling and operator interfaces, a containerized version also proved to be very helpful. For many of these applications, a specific version of Java is required, which can be provided by specifying a suitable base image. Furthermore, complex applications such as the EPICS Archiver Appliance consisting of several separate components with common dependencies profit hugely from the shared layered structure of container images.

An extremely valuable use case of containers for an EPICS control system is the ability to quickly restore the system under various circumstances. This might be the case, if an update to a newer version of an application fails, where a roll back of the system to a working state (i.e. the previous container image) can be performed in a matter of minutes¹. Another use case for the fast system restore feature of applications shipped as container images, is the case where hardware fails. In that case, a new system would only have to be set-up to the point where the container runtime engine is ready to deploy². In the next step, the images for the containers in use on the failed machine would have to be pulled to the new computer and

¹This applies as long as the previous version of the container is still accessible; ideally all images every deployed to the system should be stored on an in-house container registry for this purpose.

²For this purpose, it would be desirable to have spare computers ready with such a minimal setup to reduce down-times.

the containers started with identical parameters, which is a matter of minutes³.

7.1.3 Technology: Kubernetes Cluster

Kubernetes is the de-facto standard for container orchestration, which means automated coordination, management and configuration of OCI application containers. A Kubernetes cluster consists of one or multiple nodes, which are typically virtual machines but can also be bare-metal computers. Kubernetes clusters can run on-premises or in the cloud via the offerings of various hyperscalers (such as Amazon AWS, Google GCP, or Microsoft Azure).

The nodes in a Kubernetes cluster are typically divided into control-plane and worker nodes, while in more custom configurations a node may also be given both these labels and fulfil both roles at once. The worker nodes run the actual workloads or Kubernetes Deployments, all managed by one or more control-plane nodes. High availability of the cluster can be achieved by scaling the number of control-plane nodes up to an uneven number (1 → 3, 5, ...) in order to make use of majority-based agreement. OCI compliant container images are used in Kubernetes and can be configured for different deployment schemes. For example, a container deployment can be set up with a minimum and maximum number of identical instances of a particular container in order to scale a stateless application up or down on demand. Multiple containers can run on an abstraction called Kubernetes Pod, which represents a common network namespace. The state of Kubernetes Deployments is continuously monitored and the system can automatically restart failed services or scale the number of workers depending on the (varying) load.

Typical use cases of Kubernetes clusters comprise:

- Running scalable web applications in a redundant, high-availability mode,
- deploying microservice based applications

Use Case for EPICS

Kubernetes potentially has a lot to offer for essential EPICS applications, like

- centralizing logging and monitoring of the epics processes (IOCs, gateways)
- automated restarting of failed applications,
- auto scaling (e.g. scale the number of running channel-access gateways up or down depending on the number of connected clients),

³It should not be implicit knowledge, what containers are needed and how they need to be started. For this very reason, DevOps techniques should be employed, documenting the infrastructure of the control system as code/configuration. Docker Compose [204] has proven to be very useful for this purpose.

- commodity web interfaces available to check system health of EPICS (Prometheus, Grafana),
- high availability for SoftIOCs (relocating it to a different node in case of a node failure),
- and potentially many more.

To realise these benefits, the Kubernetes system must be configured in a certain way, however. Most importantly, EPICS networking relies on UDP (User Datagram Protocol) broadcasts for PV name search requests⁴, so that secondary network configuration might be required (e.g. with Multus & a CNI plugin such as macvlan [205, 206]). Furthermore, the data backend of EPICS Archivers isn't stateless and needs persistent volumes to keep its data, meaning it cannot as easily be scaled up/down.

7.1.4 Technology: Data Stream Management Systems

The rise of Data Stream Management Systems (DSMSs) represents a huge shift of paradigms with respect to the much older concept of Database Management Systems (DBMSs). Both concepts describe systems to store and process information. While a Database Management System (DBMS) stores data (and its relations) persistently, queries for the data are transient. In a Data Stream Management System (DSMS), however, a query is a persistent entity stored in the system and is continuously being executed against the (transient) incoming data stream [208, 207, 209].

Table 7.1 highlights the major differences between DBMSs and DSMSs.

Typical use cases for DSMSs include:

- Global system state monitoring and surveillance, (e.g. classification into states such as 'regular operation', 'peak load', 'system outage', 'anomaly detected'),
- fast identification of trends (e.g. real-time data mining instead of periodic batch processing),
- message passing between distributed systems.

Use Case for EPICS

By their very nature, EPICS-based control systems are specialized data stream management systems inherently. Compared to the more general purpose DSMSs that were developed in the last years – such as Kafka + KSQL for example – an

⁴Unicast can be configured as an alternative if the relevant hosts to be queried are known and configured beforehand. While this is possible, does not scale well scalable and is thus recommended against.

DBMS	DSMS
Persistent relations	Transient streams
One-time queries	Continuous queries
Random access	Sequential access
‘Unbounded’ disk store	Bounded main memory
Only current state matters	History/arrival-order is critical
Passive repository	Active stores
Relatively low update rate	Possibly multi-GB arrival rate
No real-time services	Real-time requirements
Assume exact data	Data stale/imprecise
Access plan determined by query processor, physical DB design	Unpredictable/variable data arrival and characteristics

Table 7.1: Comparison table of DBMS and DSMS. Taken from [210].

EPICS system is much more dedicated to its specific task resulting in disadvantages in terms of high availability, scope of their use case, scalability, performance and the required, specialized expert knowledge to operate them.

In that sense, EPICS can profit a lot from recent developments in DSMSs. Either by adopting concepts and architectural ideas from those new general purpose solutions or by simply deploying those systems in addition to their existing architecture.

One example of the latter approach is the use of Kafka and Elasticsearch as backends for the EPICS alarm server and logger implemented in Phoebus suite after it emerged from the previous CS-Studio code base.

7.2 Modernization of EPICS Key Components

While the previous section introduced the general technological enablers allowing to move forward with the application modernisation, this section will now discuss the individual components of EPICS and their potential to undergo a modernisation process.

7.2.1 Core: EPICS Base

EPICS base is at its very heart a software development kit (SDK) consisting mainly of several shared libraries, a couple of executables, and to tools to create boilerplate code to start a project using EPICS base. The typical EPICS architecture of

CA (Channel Access) servers called IOCs (keeping track of process variable state and logic in their built-in “EPICS database”) and CA clients (consuming the state information or commanding state transitions) is already a good basis for a loosely coupled light-weight ecosystem in the “modernized” sense. Concerning EPICS base, essentially creating base container images of the software are the most important modernization step. The images for this thesis were created using the Dockerfiles to be found at <https://github.com/pklau/docker-epics>.

EPICS Base components such as SoftIOCs are also well suited to run in a Kubernetes cluster, if the connectivity to the EPICS network can be provided. This can be implemented using the Kubernetes CNI plugins Multus [205] and macvlan [206] adding a secondary network connection to the Kubernetes Pods for low-level access to the (external) EPICS network. This is visualized in fig. 7.1.

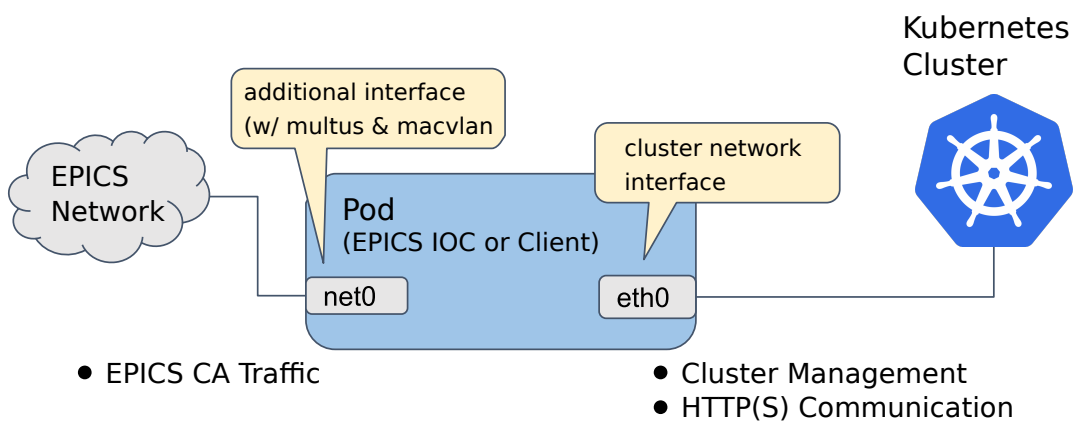


Figure 7.1: Deployment of EPICS components on a Kubernetes Pod with the CNI plugin Multus. Image adopted from [205] under the Apache 2.0 license.

7.2.2 Core: synApps

In addition, many modules are available to extend EPICS base with optional functionality. Some of them add new record types to the EPICS database (e.g. `calc`), others add the ability to write state machines inside EPICS (`sncseq`), add the device support to connect to specific hardware buses (e.g. `asyn`, `modbus`) or the ability to describe the communication protocol used by devices sending (and receiving) a data stream `StreamDevice`. A distribution of the most commonly used modules is published by APS/ANL under the name `synApps` [211, 212]. Unfortunately, the current `synApps` distribution is not well suited for compilation in a container build: If all modules are enabled, the compilation is a single step taking a lot of time to complete; if anything fails during the compilation, the whole step

has to be repeated without making use of the container build cache.

In a modernized scenario of synApps, the collection of optional EPICS modules are fetched and compiled individually in the context of the aforementioned EPICS base container image. This has been implemented in the contApps container image: <https://github.com/pklaus/docker-contapps>.

7.2.3 Operator Interface: CS-Studio Phoebus

Phoebus is a rich GUI to control and operate the EPICS system. It comes with multiple built-in applications such a runtime environment for interactive operator interfaces, a data viewer for past values, alarming features, and much more. The application is written in Java. After its predecessor was implemented for the RCP (Eclipse Rich Client Platform), phoebus is now a plain Java application mostly built on top of the JavaFX GUI UI framework. As a fully-fledged desktop application, phoebus might be considered a monolith in the terminology of “application modernization”. It will, however, in the mid-term future persist in the EPICS system landscape. It was therefore decided to simply package the application in a container: <https://github.com/pklaus/docker-phoebus>. Running a GUI application distributed as a container can comfortably be achieved with the x11docker wrapper script. x11docker can be configured to run the application in a fashion indistinguishable from a native application or started in a separate desktop environment, optionally setting up remote access via VNC.

7.2.4 Operator Interface: Prometheus (r/o)

In the recent years, cloud computing and ever bigger server farms were driving the need to monitor large amounts of metrics from many different services. In this context, various solutions for time series data storage and plotting (such as Graphite, Influx DB and Prometheus) were developed. Prometheus [213] is one of the more recent developments and is frequently used nowadays to monitor the system health of a Kubernetes cluster and the workloads that are deployed to the cluster. Some of the mentioned solutions include built-in plotting, others only act as specialized time series databases. While Prometheus comes with a minimal web UI, it is best used with a separate plotting tool such as Grafana [214].

For EPICS, it makes sense to evaluate and reuse technology that already exists and in order to save the scarce resource EPICS expert developer time. So if the need arises for a read-only web dashboard to plot trends, a solution such Prometheus can save a lot effort while providing users with the huge flexibility to create derived metrics using its built-in query language ‘PromQL’. In order to export data

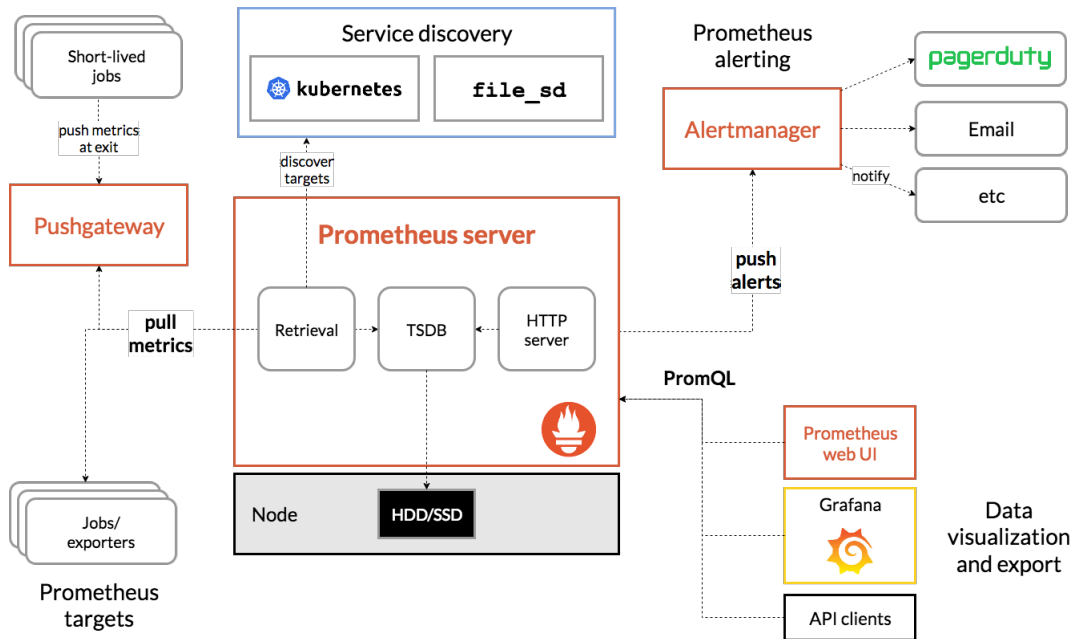


Figure 7.2: The architecture of Prometheus and its typical ecosystem’s components. From [215] (CC-BY-4.0).

to Prometheus, an intermediate layer between EPICS and Prometheus is required. Figure 7.3 shows the proposed architecture with a chain of communication from the EPICS IOC to the operator’s web browser, which shows the visualizations of the PV data provided by Grafana. All services could be on the same physical network (bottom line), the logical communication paths would form a linear chain from layer to layer as indicated with the arrows. The intermediate layer between EPICS and Prometheus would be implemented with a dedicated EPICS client exposing the PVs via HTTP (Hypertext Transfer Protocol) as metrics that Prometheus can fetch.

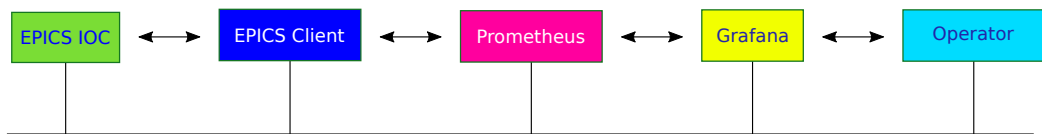


Figure 7.3: Architecture for data visualization of EPICS PVs with Grafana & Prometheus.

7.2.5 Gateways: Channel Access Gateway

The Channel Access Gateway [216] is a software allowing traffic of the CA protocol to traverse network boundaries. It is further explained in section 5.12. (An

alternative for the successor protocol PVA is the tool `pvagw`, see below.) The gateway process starts a server listening on one network where it provides access to process variables it tries to resolve and connect to on one or more other networks where it – in turn – acts as a client.

The Channel Access Gateway can be started multiple times on different machines, which have access to the relevant networks. This allows to scale the number of gateway containers up in case the load on the gateway process is increasing.

Newer alternative: PVAccess Gateway The PVA Gateway `pvagw` (from the Python package `p4p`) [217] is an equivalent of the Channel Access Gateway for the newer PVAccess Protocol. It replaces an earlier attempt for such a gateway made by the `pva2pva` software under the name `p2p` [218].

7.2.6 Archivers: Archiver Appliance

The EPICS archiver appliance is a very interesting target for “application modernization” as it comes with out of the box support to run in a distributed fashion on multiple nodes with flexible storage options. Each instance is split into four separate components – ‘management’, archiving ‘engine’, ‘etl’ and ‘retrieval’, which are required to run on the same hostname. The management component is the coordinating service in each instance, communicating with the other components via TCP. The ‘etl’ component (ETL (Extract Transform Load)) is managing the data storage tiering, which allows to keep short/mid/long term storage on different types of storage media for access speed, reliability and cost reasons. This concept is perfectly suited for running an instance of the appliance in a Kubernetes Pod as it allows to run the four mentioned components as multiple containers in a single network namespace (thus fulfilling the requirement of a common hostname). The EPICS Archiver Appliance can then be configured to run multiple instances, which share the burden of archiving incoming data (‘engine’ component). For each registered PV to be archived, a responsible instance is assigned, which allows to obtain great horizontal scalability. The same is true for retrieval of the data: When an operator GUI requests recorded values for time-series plots, the horizontal scaling of the ‘retrieval’ component provides an excellent user experience when many PVs are requested at the same time. The binary storage model used by the Archiver Appliance by default, provides highly optimized response times compared to the user experience offered by other archivers that rely on relational databases as persistent storage.

To demonstrate the capability of the EPICS Archiver Appliance to archive a large number of PVs, a scalability test was performed. The tools used for this purpose can be found on [219]. A single instance of the archiver appliance was employed

for the scalability test. All in all, 25 SoftIOCs were started on a separate computer as shown in fig. 7.4, each serving 10 000 so called ‘`calc`’ records with a simulated random walk of a float value updating at 1 Hz. In total, 250 000 PVs were being registered for archival, with each of them consisting of 7 channels due to the nature of the EPICS `calc` record. The value of the main channel representing the value of the record is updating at said 1 Hz, a typical update rate for many of the control system’s process variables.

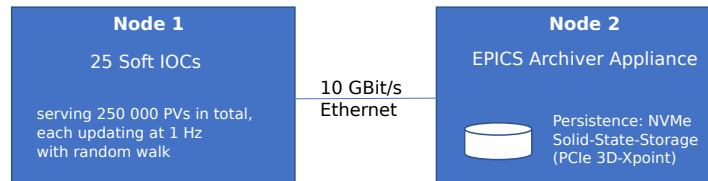


Figure 7.4: Setup for the EPICS Archiver Appliance scalability test.

To perform the scalability test, all process variables must first be registered with the archiver. For the large amount of 250 000 PVs, this cannot be done in a single step. Instead, a Python script automates adding the PVs in chunks of 200, which takes quite some time (~ a few hours), as the Archiver Appliance determines each PV’s characteristic during an initial sampling phase when adding it to the system, see the user guide [153]. After all PVs were successfully added to the archiver, the system was restarted and during its start-up phase, another script was continuously logging all important status parameters of the Archiver Appliance. Most notably, the amount of time the writer thread spent to write the incoming stream of PV-updates to disk was recorded. When this measured value is normalized to the time window available (a configuration parameter of the archiver), the result is the busy time ratio of the writer thread. If this ratio approaches 1.0 (100%), the buffers for PV updates would overflow and discard the oldest updates. Therefore, this value needs to be well below 100% and represents an important indication of reliable operation of the archiver.

The result of the measurement during the start-up phase of the Archiver Appliance is shown in fig. 7.5. The graph shows the very linear behaviour of busy time and data rate with respect to the number of connected PVs. At the maximum of 250 000 of concurrently connected PVs, the busy time of the thread writing the data to the storage reaches ~ 50%, which is a safe value resulting in reliable operation. As a result, a single instance of the EPICS Archiver Appliance was shown to be able to scale to 250 000 PVs without data loss due to buffer overflows.

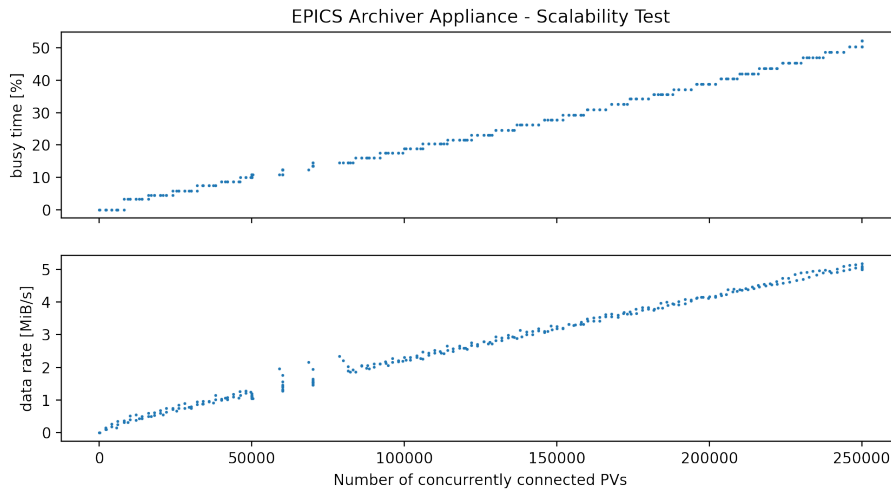


Figure 7.5: EPICS Archiver Appliance scalability test with 250 000 concurrently connected PVs.

7.2.7 Archivers: Cassandra PV Archiver

The Cassandra PV Archiver is another archiver software that needs to be mentioned in the context of application modernization due to its unique architecture with scalability in mind. It is a software published under the terms of the Eclipse Public License v1.0 by aquenos GmbH [154] and is the successor of the earlier ‘Cassandra Archiver for CSS’ [160].

The Cassandra PV Archiver is able to archive EPICS PVs to a an Apache Cassandra database [163]. Cassandra is a distributed and highly scalable database system that is well suited to be deployed as container images or in a Kubernetes cluster. As “the Cassandra PV Archiver is designed to work [...] for [...] very large-scale setups (with tens or even hundreds of nodes)” [155], it might be a valuable option to replace older archiver components if scalability requirements would rise further. This makes it a perfect example of a component that was rewritten from scratch in order to fulfil the goals of application modernization. As a downside, the excellent scalability in this case comes at the cost of having to manage the distributed Cassandra database, which adds complexity to the system.

7.2.8 EPICS Operator Interface Implemented in the Browser

A further option to evaluate a modernization of EPICS is the idea to move the operator interface (GUI) from a dedicated desktop application to an application running in the web browser.

A first attempt in this direction was the status dashboard for the EPICS control system of the prototype “PRESTO” in the context of this thesis, see section 5.11.2. Beyond this r/o dashboard, it might be valuable to have the entire operator interface running in the browser.

In general, moving the control system to the browser has the following advantages:

- The operator interface could be used practically any hardware and operating system, as a browser is almost always available.

Furthermore, the following advantages specifically related to application modernization would apply:

- Modern web technology provides excellent possibilities for **horizontal scalability and load balancing**. This might be achieved by scaling up the number of web servers and using a reverse proxy to distribute incoming requests to all of them, for example in a round-robin scheme.
- Using a web application would allow to break the aspects covered by current operator interface software (i.e. Phoebus) into many small **microservices**. As an example, access to archived data for plotting could be one independent microservice, while another microservice could serve the graphical representation of the data to render it in the browser in the form of interactive plots. Yet another microservice could provide the real-time updates of EPICS PVs via a websocket connection.
- When running in a browser environment, the control system could easily be protected with individual user accounts. **Authentication** would increase the security of the control system and allow to log the responsible account name for any operator action.

Operators logging-in via personalized accounts and having assigned roles with custom permissions would be a great improvement, possible with a full overhaul of a web based operator interface.

As a conclusion, transforming the operator interface into a native browser application would have many benefits and would allow to port the control systems to today's user interface standards and best practices.

8 Summary

The Compressed Baryonic Matter (CBM) Experiment will investigate heavy ion collisions and reactions at interaction rates of 100 kHz in a targeted energy range of up to 11 AGeV for systems such as gold-gold or lead-lead. It will be one of the major scientific experiments of the Facility for Antiproton and Ion Research in Europe (FAIR) currently under construction at the site of the GSI Helmholtzzentrum für Schwerionenforschung (GSI) in Darmstadt, Germany. CBM is going to be a fixed target experiment consisting of a superconducting magnet, multiple detectors of various types, and high-performance computing for online event reconstruction and selection. The detector closest to the interaction point of the experiment will be the Micro Vertex Detector (MVD). Consisting of four planar stations equipped with custom CMOS pixel sensors, it will allow to reconstruct the primary vertex with high precision and will help to reconstruct secondary vertices and identify particles originating from conversion in the detector material.

Challenges Addressed in this Thesis

Due to the high interaction rates foreseen for CBM, understanding and minimising *systematic errors* due to the detectors' operating conditions will become all the more important to obtain significant measurement results, as statistical errors in the measurements of many observables are diminishing due to the enormous amount of data available.

Furthermore, the MVD will be the *first* detector based on CMOS pixel sensors used in a large physics experiment, that will be *operated in vacuum*. As a result, many aspects of the mechanical and electrical integration of the detector require careful testing and validation.

This thesis addresses both those challenges specifically for the Micro Vertex Detector with the development of a control system for the operation and validation of the MVD prototype "PRESTO" in vacuum. The prototype was selected as device under test as the final MVD is not yet built.

The developed *control system* helps a) to operate the prototype safely and keep it at the desired working point and b) to record important time-series data of the state of the detector prototype. Those two aspects allow the control system (which might later serve as a 'blueprint' for the final detector) to minimize the mentioned systematic errors as much as possible and to contribute to the understanding of

8 Summary

remaining systematic errors using correlations with the time-series data. The controlled *operation of the prototype in vacuum* allowed to validate the integration concepts from a wide range of mechanical and electrical aspects in an endurance test for more than a year with 24/7 operation.

Contributions of this Thesis to the small-scale CBM-MVD Prototype “PRESTO”

The prototype for this study itself was named “PRESTO” (standing for ‘PREcursor of the Second sTatiOn of the CBM-MVD’). It represents one quadrant of an MVD detector plane, equipped with a total of 15 MIMOSA-26 sensors on the front and back side of a carrier plate. Within this thesis, major parts of the prototype itself were designed. Custom ultra-thin flat flexible cables for data and power were designed and validated. Furthermore, the CNC-machined Aluminium heatsink to mount and cool the prototype design was refined to increase thermal performance. A custom vacuum feedthrough for a total of 21 flat ribbon cables was designed and fabricated. The read-out chain for MIMOSA-26 was extended to cover a total of 8 sensors with a single and newer TRB-3 FPGA board and was set-up with the prototype. Vacuum equipment including chambers, hoses, pumps, valves and gauges were integrated to form a large vacuum testing system. A cooling circuit for the prototype was assembled comprising an external chiller, hoses, vacuum feedthroughs, as well as temperature, flow and pressure sensors.

The Control System for the small-scale CBM-MVD Prototype “PRESTO”

The control system was developed to serve the needs of the prototype, while taking the requirements of the final MVD already into account:

The main *design goals of the control system* are:

- compatibility with the other detectors and the overall CBM experiment,
- access to real-time measurements of all necessary parameters (‘process values’),
- reliable, fail-safe operation of the detector,
- recording of all time-series data (‘archiving’),
- cost efficiency and acceptance within the physics community,
- good usability for the users (‘operators’),
- long-term maintainability

The recorded time-series data of the process variables (i.e. sensor readings) allow a post-measurement analysis of variations in the detector performance. The long-term archiving of all relevant system parameters is therefore of outstanding importance, which is why the software intended for this purpose – called “archiver” – was given special attention in this thesis.

For this reason in particular, it is necessary to implement a comprehensive control

system that allows the detector to be operated safely under these conditions and cooled effectively. Before the start of this doctoral thesis, vigilant and extensively trained operators were always necessary for this. The control system that has been developed makes it possible that, after basic training, the detector can also be operated by a less specialised shift supervisor during measurement campaigns. This involves the integration of a variety of sensors and actuators, all of which are necessary to ensure effective control of the system. State machines allow the system to operate autonomously to a large extent and provide the operator with a summarised overview of the system state, which significantly reduces the complexity in operation. A clear graphical user interface provides the overview, which allows the state of the detector to be assessed in the shortest possible time, monitored effectively and desired operating states to be reached in a controlled manner.

As a technical basis for the control system, a proven system was selected that is intended for use by the entire CBM experiment (and further FAIR large-scale experiments): Experimental Physics and Industrial Control System (EPICS). This system combines decisive advantages, such as the permanently free use under a "free software licence", the strong roots in the physics community, and the possibility to easily modify or even replace individual components.

With the help of the aforementioned EPICS software framework, it was possible to build a control system that records all the important parameters for the operation of the detector in PVs (Process Variables). Starting with general parameters such as the ambient temperature and humidity in the laboratory, the measured values of the residual pressure in the vacuum chamber, the temperatures, pressures and flow rates of the cooling system, the resulting temperatures inside the vacuum chamber, the power supply with the low-voltage power supplies and the threshold values of the individual sensors and their data rates, the system status was comprehensively measured, controlled (monitored) and recorded. The PVs are displayed in a graphical user interface that, depending on the view, allows a broad system overview or details all the details of a selected subsystem such as the cooling system or the low-voltage power supply. The user interface can also be used to initiate changes to system parameters by the operator of the detector. State machines make it possible to categorise the state of the system on the multitude of PVs available and to enable controlled transitions between the differentiated states such as 'switched off', 'ready for measurement' and 'fully operational'.

Technically, when implementing the control system, emphasis was placed on benefiting from newer technologies such as container virtualization. Specifically, the so-called IOC instances of the EPICS system were containerised, each communicating with different hardware components. The archivers used and the necessary database systems were also set up accordingly. As a result, a corresponding solution can be put into operation within a very short time (for example, for a test beam

period), since ready-made container images are available that already contain all the necessary dependencies. A further modernisation of the complete EPICS software stack is discussed in the chapter “Modernising the EPICS Control System” in the form of a white paper.

Validation of the Prototype using Field Tests and Thermal/Pressure Cycling

The operation in vacuum allowed to validate many aspects of the mechanical and electrical integration. Running the prototype through many vacuum and temperature pumping cycles led to the conclusion that with the present mechanical integration concept, no damage due to pressure and temperature differences is likely to be expected. No physical damage to the prototype was observed and no degradation of the sensors’ sensitivity was found with the help of the available data.

After the control system was successively built up, the MVD prototype PRESTO was exposed to various field tests. To test for pressure changes, the prototype was operated in a vacuum with cooling for a long period of time and continuously read out. In addition, fully automated vacuum and cooling cycles were also performed for this purpose, which can simulate the higher material loads and hazards with changing environmental parameters. The field tests were completed by measurements with an ^{55}Fe source in vacuum and measurements of the background activity. Within the framework of the evaluation of the background measurements, strong indications were found for the recording of muons, which occur as secondary particles of cosmic rays. The expected angular distribution in the measurement data was confirmed.

As a result, it could be confirmed that the detector can be successfully cooled and operated over extended periods of time in a vacuum without constant human interaction and that no abnormalities occurred in the sensor operation. The EPICS-based control system (DCS) proved effective and well suited for the task, given its distributed nature allowing to tune, extend or replace parts of the system as needed. In addition to the result that the control system was successfully implemented in the process, the work also confirms in this respect the sophisticated integration concept of the detector, in which pixel sensors thinned on $50\ \mu\text{m}$ are applied to carbon-based carrier plates for positioning and cooling.

Conclusion

To conclude, the thesis was able to prove that the concepts for the mechanical and electrical integration to operate the future Micro Vertex Detector of CBM for operation in vacuum are valid. This was done successfully with the help of a custom control system developed specifically for the MVD prototype PRESTO and can serve as a blueprint for the control system of the future MVD.

Glossary

24/7 Continuous, uninterrupted operation 24 h a day, 7 d a week. 7, 37, 44, 69, 75, 124

ALICE A Large Ion Collider Experiment 11, 12, 25, 49, 127

ALPIDE A family of monolithic active pixel sensors developed bfor the ALICE experiment in a joint effort between CERN and IPHC. 25, 29

APPA Atomic Physics, Plasma physics and Applications 10

BEAUTY Best Ever Archive Toolset, yet – an alias for the CSS RDB Archiver. 67, 69

CA Channel Access 51–56, 65, 69, 71, 107, 116, 118, 131, 132

CAD Computer Aided Design 37

CB Converter Board 41, *further details:* Converter Board

CBM Compressed Baryonic Matter 1–4, 7, 9–11, 15, 19, 29, 30, 51, 55, 88, 99

CbmRoot A software framework for the CBM experiment, facilitating the simulation and online operation of the experiment. 13, 29, 30

CERN European Organization for Nuclear Research 33, 127

Channel Access Gateway An application forwarding CA messages thus allowing EPICS applications to cross network boundaries. 62, 65, 118

CMOS Complementary Metal-Oxide-Semiconductor 6, 13, 20, 21, 26, 130, *further details:* Complementary Metal-Oxide-Semiconductor

CMS Center-of-Momentum System 12

Complementary Metal-Oxide-Semiconductor A method of chip production resulting in logic elements with low power operation well established in commercial production. 6, 13, 20, 21, 26, 127, 130,

Control System Studio Control System Studio is a graphical application (GUI) providing OPIs for EPICS written in Java. 53, 54, 60, 61, 128, 131,

Converter Board The Converter Boards in the revision 2013 [88, 14]. 41, 127,

- CSS** Control System Studio 53, 54, 60, 61, 131, *further details*: Control System Studio
- DAQ** Data Acquisition System 11, 26, 27, 43–45, 47, 51, 53, 58, 73, 82, 84, 99, 106, 107
- DBMS** Database Management System 114, 115
- DCS** Detector Control System 4, 7, 47–51, 53–55, 58, 66, 67, 73, 75, 107
- DESY** Deutsches Elektronen-Synchrotron 35
- DevOps** Development and Operations – a set of practices that combines software development (Dev) and IT operations (Ops). 109, 113
- Docker** A very popular solution for container virtualization. 56
- DSMS** Data Stream Management System 114, 115
- ECS** Experiment Control System 51, 58, 107
- EPICS** Experimental Physics and Industrial Control System 7, 39, 47, 49, 50, 53–58, 63–71, 73, 106, 107, 112–116, 125, 126, 131
- EPICS base** The fundamental software distribution of the EPICS framework, including header files, shared libraries, scripts for boilerplate code generation and some executable tools. 52, 53
- ESS** European Spallation Source 107
- ETL** Extract Transform Load 119
- EUDET** A research project for detector R&D towards the International Linear Collider between 2007 and 2010 funded in the in the 6th Framework Programme of the EU. 35
- FAIR** Facility for Antiproton and Ion Research in Europe 1, 9, 10, 50
- FEB** Front-End Board 41, 44, 45, 60, 84, *further details*: Front-End Board
- FFC** Flat Flexible Cable 41, 44, 45
- FLES** First Level Event Selector 11
- FPC** Flexible Printed Circuit 31, 34, 36, 38, 42–45, 76, 84
- FPGA** Field-Programmable Gate Array 42, 47, 65, 132
- Front-End Board** The Front-End Boards in the revision XX [88, 14]. 41, 44, 45, 60, 84, 128,
- GEANT** GEometry ANd Tracking – a platform for the simulation of the passage of particles through matter using Monte Carlo methods. 30

- Grafana** A commodity open source analytics, monitoring and interactive visualization web application software. 117, 118
- GSI** GSI Helmholtzzentrum für Schwerionenforschung 10, 50, 67
- GUI** Graphical User Interface 52, 53, 60, 66, 68, 119, 121, 127
- HA** high availability 110
- HADES** High Acceptance Di-Electron Spectrometer 9, 11, 34, 49, 50, 67
- HFT** Heavy Flavor Tracker 29, 35, 131
- HTTP** Hypertext Transfer Protocol 118
- I2C** Inter-Integrated Circuit 26, 47
- IC** Integrated Circuit 129
- IKF** Institut für Kernphysik 27
- IOC** Input/Output Controller 52–58, 64–66, 73, 79, 82, 113, 118, 125, 131, 132
- IOPS** input/output operations per second 68
- IPHC** Institut Pluridisciplinaire Hubert Curien 27, 35, 127, 130
- IPv4** The 4th (and currently still the most common) version of the Internet Protocol. 51
- Joint Test Action Group** Common industry standard IEEE 1149.1 for testing and debugging ICs (Integrated Circuits). 27, 42, 47, 129,
- JTAG** Joint Test Action Group 27, 42, 47, *further details:* Joint Test Action Group
- Kubernetes** A container orchestration software initially developed by Google, now managed by the Cloud Native Computing Foundation (CNCF). 113, 114, 121, 129
- Kubernetes Deployment** A Deployment is a declarative description of Kubernetes workloads, typically a Kubernetes Pods. 113
- Kubernetes Pod** A Pod allows to run one or more containers in a single network namespace in Kubernetes. 113, 116, 119, 129
- LVDS** Low-Voltage Differential Signaling 43, 44
- LXC** Linux Containers 111
- MAPS** Monolithic Active Pixel Sensor 13, 20, 26, 29, 33, 35, 86, 88, 130, *further details:* Monolithic Active Pixel Sensor

- MC** Monte Carlo xiv, 14, 91, 92, 98, 101
- MEG** monoethylene glycol 80, 83
- MIMOSA** A series of MAPS (Monolithic Active Pixel Sensor) developed by IPHC. 26, 27, 29, 130
- MIMOSA-26** A particular sensor belonging to the MIMOSA series of MAPS developed by IPHC. It was used for the prototype PRESTO used in this thesis. xiii, 19, 20, 29, 35, 36, 44, 47, 82, 86, 87, 89, 92, 93, 95, 98, 100
- MIMOSIS** A family of monolithic active pixel sensors developed by IPHC specifically for the MVD. 19, 25–27, 29, 35, 36, 47, 87
- Minimum Ionizing Particle** Particles within a certain specific momentum of $\beta\gamma \approx 3 \text{ MeV}$ where their energy loss in matter is at a minimum. 17–19, 24, 93, 130,
- MIP** Minimum Ionizing Particle 17–19, 24, 93, *further details*: Minimum Ionizing Particle
- Monolithic Active Pixel Sensor** A type of pixel sensor combining front-end electronics and active sensor volume in a single CMOS die. 13, 20, 26, 29, 33, 35, 86, 88, 129, 130,
- MOTS** Modifiable Off-The-Shelf 52, 53
- MRPC** Multi-Gap Resistive-Plate Chamber 14
- MUCH** Muon Chamber 11, 14
- MVD** Micro Vertex Detector 1, 2, 6, 7, 11, 13, 17, 25, 27, 29, 30, 33–35, 37, 47, 48, 55, 64, 65, 75, 82, 88, 99, 106, 130
- MWPC** Multi-Wire Proportional Chamber 14
- NUSTAR** Nuclear Structure, Astrophysics and Reactions 10
- OCI** Open Container Initiative 111, 113
- OPI** Operator Interface 47, 52, 60, 68, 127
- PANDA** antiProton ANnihilation at DArmstadt 10
- Particle Identification** Act of telling particle classes apart with the help of dedicated detectors. 14, 131,
- PC** Personal Computer 58
- PCB** printed circuit board 28
- pCVD diamond** A polycrystalline diamond created using chemical vapour deposition. 27, 28, 30

- pdf** probability density function 92
- Phoebus** Renewed version of CSS 60, 61, 122
- PID** Particle Identification 14, *further details:* Particle Identification
- POC** Proof of Concept 53, 69–71
- PR** public relations 63
- PRESTO** Prototype of the Second Station xiii, xiv, 2, 7, 29, 33–36, 38, 39, 41, 42, 44, 45, 47, 48, 53–56, 58, 60, 64–67, 69, 75–78, 81, 82, 84, 86, 88, 106, 107, 109, 126, 130
- Process Variable** An identifier for a specific EPICS control record served via CA or PVA. 52, 54, 57, 125, 131,
- Prometheus** A commodity free software application used for event monitoring and alerting, that records real-time metrics in a time series database. 117, 118
- PSD** Projectile Spectator Detector 11, 14
- PV** Process Variable 52, 54, 57, 58, 62–65, 67–69, 71–73, 78, 83, 106, 114, 118–122, 125, 131, *Glossary:* Process Variable
- PV Access** A network protocol designed as a more capable successor of the CA protocol. 51–53, 119, 131, 132,
- PVA** PV Access 51–53, 119, 131, 132, *further details:* PV Access
- PXL** The innermost particle detector of the HFT featuring MAPS PiXeL sensors. 29, 35
- PyTrbNet** Python bindings for TrbNet, which also provides the ability to run an IOC serving TRB register contents as EPICS PVs. 64, 73, 82, 106
- QFT** quantum field theory 4
- r/o** read only 58, 62, 63, 122
- RCP** Eclipse Rich Client Platform 117
- RDBMS** Relational Database Management System 68
- RHIC** Relativistic Heavy Ion Collider 29, 132
- RICH** ring-imaging Cherenkov detector 11, 13, 14
- ROI** Region of Interest 86, 87
- RTS** random telegraph signal 21, 93
- S/B** signal over background ratio 2

- SBC** Single Board Computer 58
- SchuKo** AC power plugs and sockets standardized as ‘CEE 7/3’ and ‘CEE 7/4’ for sockets and plugs respectively. 62
- SIS100** Schwerionensynchrotron 100 10, 11
- SIS300** Schwerionensynchrotron 300 11
- SMASH** Simulating Many Accelerated Strongly-interacting Hadrons 2
- SNR** signal to noise ratio 35, 93
- SoftIOC** An IOC that does not communicate with additional hardware. All communication is done via CA or PVA. 52, 67, 114, 116, 120
- SPS** Super Proton Synchrotron 33
- Standard for the Exchange of Product model data** A very common exchange format for 3D models standardized in ISO 10303. 132
- STAR** Solenoidal Tracker at RHIC (Relativistic Heavy Ion Collider) 29, 35
- STEP** Standard for the Exchange of Product model data *further details*: Standard for the Exchange of Product model data
- STS** Silicon Tracking System 1, 11, 13, 17
- TCP** Transmission Control Protocol 60, 119
- TDR** Technical Design Report 25
- Time of Flight** A method to measure the velocity of a particle by measuring the time they need to travel between displaced detector planes. 11, 14, 132
- TOF** Time of Flight 11, 14, *further details*: Time of Flight
- TPC** Time Projection Chamber 6
- TPG** Thermal Pyrolytic Graphite 28, 30, 34, 36, 98
- TPS** ransactions per second 68
- TRB** Trigger and Readout Board 38, 42, 64, 65, 131, 132
- TRB3** Version 3 of the TRB platform 38, 42, 43, 45, 47, 53, 58, 64, 65
- TrbNet** Network implemented on FPGA-based TRB DAQ systems allowing to read and write custom registers, uploading new design files etc. 64, 65, 131
- TRD** Transition Radiation Detector 11, 14
- UDP** User Datagram Protocol 114
- UrQMD** Ultra-relativistic Quantum Molecular Dynamics 2

VLAN Virtual Local Area Network 53, 110, 111

VXLAN Virtual Extensible Local Area Network 110

XML Extensible Markup Language 65

YAML A human-readable data-serialization language. 53

Bibliography

- [1] T. Ablyazimov et al. ‘Challenges in QCD matter physics – The scientific programme of the Compressed Baryonic Matter experiment at FAIR’. In: *The European Physical Journal A* 53.3 (Mar. 2017), p. 60. DOI: 10.1140/epja/i2017-12248-y. arXiv: 1607.01487 [nucl-ex] (cit. on pp. 1, 9).
- [2] P. Senger. ‘QCD Matter Physics at the Future FAIR Facility in Germany’. In: *Proceedings of 13th International Conference on Nucleus-Nucleus Collisions*. Journal of the Physical Society of Japan, July 2020. DOI: 10.7566/jpscp.32.010092 (cit. on pp. 1, 11).
- [3] T. Galatyuk. ‘Recent Results from HADES’. In: *Proceedings of 13th International Conference on Nucleus-Nucleus Collisions*. Journal of the Physical Society of Japan, July 2020. DOI: 10.7566/jpscp.32.010079 (cit. on p. 2).
- [4] G. Agakishiev et al. ‘Statistical hadronization model analysis of hadron yields in p + Nb and Ar + KCl at SIS18 energies’. In: *The European Physical Journal A* 52.6 (June 2016). DOI: 10.1140/epja/i2016-16178-x. arXiv: 1512.07070 [nucl-ex] (cit. on pp. 2, 11).
- [5] S. A. Bass et al. ‘Microscopic models for ultrarelativistic heavy ion collisions’. In: *Progress in Particle and Nuclear Physics* 41 (1998), pp. 255–369 (cit. on p. 2).
- [6] H. Petersen et al. ‘Fully integrated transport approach to heavy ion reactions with an intermediate hydrodynamic stage’. In: *Physical Review C* 78.4 (Oct. 2008). DOI: 10.1103/physrevc.78.044901 (cit. on p. 2).
- [7] J. Weil et al. ‘Particle production and equilibrium properties within a new hadron transport approach for heavy-ion collisions’. In: *Physical Review C* 94.5 (Nov. 2016). DOI: 10.1103/physrevc.94.054905. URL: <https://doi.org/10.1103/physrevc.94.054905> (cit. on p. 2).
- [8] D. Oliinychenko et al. *smash-transport/smash: SMASH-2.0*. 2020. DOI: 10.5281/ZENODO.4336358 (cit. on p. 2).
- [9] B. Friman et al., eds. *The CBM Physics Book*. Springer Berlin Heidelberg, 2011. DOI: 10.1007/978-3-642-13293-3 (cit. on pp. 3, 11, 14).

Bibliography

- [10] B. Sonne. *Spezielle Relativitätstheorie für jedermann*. Springer Fachmedien Wiesbaden, 2016. DOI: 10.1007/978-3-658-12777-0 (cit. on p. 3).
- [11] E. Rutherford. ‘LIV. Collision of α particles with light atoms. IV. An anomalous effect in nitrogen’. In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 37.222 (June 1919), pp. 581–587. DOI: 10.1080/14786440608635919 (cit. on p. 4).
- [12] G. Zweig. ‘An SU(3) model for strong interaction symmetry and its breaking. Version 1’. In: (Jan. 1964) (cit. on p. 5).
- [13] M. Gell-Mann. ‘A schematic model of baryons and mesons’. In: *Physics Letters* 8.3 (Feb. 1964), pp. 214–215. DOI: 10.1016/s0031-9163(64)92001-3 (cit. on p. 5).
- [14] P. Klaus et al. ‘Prototyping the read-out chain of the CBM Microvertex Detector’. In: *Journal of Instrumentation* 11.03 (Mar. 2016), p. C03046. DOI: 10.1088/1748-0221/11/03/c03046 (cit. on pp. 7, 34, 42, 127, 128).
- [15] FAIR Website. <https://fair-center.eu/for-users/experiments/nuclear-matter-physics/cbm/introduction.html>. Accessed: 2020-10-16. 2020 (cit. on p. 9).
- [16] GSI. *FAIR – The Universe in the Lab*. <https://www.gsi.de/en/researchaccelerators/fair.htm>. Accessed: 2020-10-16. 2020 (cit. on p. 10).
- [17] P. Spiller et al. ‘Status of the FAIR Project’. In: *Proc. 9th International Particle Accelerator Conference (IPAC’18), Vancouver, BC, Canada, April 29-May 4, 2018* (Vancouver, BC, Canada). 9. Geneva, Switzerland: JACoW Publishing, June 2018, pp. 63–68. ISBN: 978-3-954-50184-7. DOI: 10.18429/JACoW-IPAC2018-MOZGBF2 (cit. on p. 10).
- [18] T. Galatyuk. ‘Future facilities for high μ_B physics’. In: *Nuclear Physics A* 982 (Feb. 2019), pp. 163–169. DOI: 10.1016/j.nuclphysa.2018.11.025. URL: <https://doi.org/10.1016/j.nuclphysa.2018.11.025> (cit. on p. 12).
- [19] A. Andronic et al. ‘Decoding the phase structure of QCD via particle production at high energy’. In: *Nature* 561.7723 (Sept. 2018), pp. 321–330. DOI: 10.1038/s41586-018-0491-6 (cit. on p. 12).
- [20] FAIR Website. *The CBM detector and data processing system*. <https://fair-center.eu/for-users/experiments/nuclear-matter-physics/cbm/projects.html>. Accessed: 2020-10-16. 2020 (cit. on p. 11).
- [21] CBM Website. *The detector and data processing system | CBM experiment at FAIR*. <https://www.cbm.gsi.de/detectors>. Accessed: 2021-06-11. 2021 (cit. on p. 11).

- [22] *CbmRoot Geometry Repository / Setups*. https://git.cbm.gsi.de/CbmSoft/cbmroot_geometry/-/tree/master/setup. Accessed: 2021-06-07 (cit. on p. 13).
- [23] A. Senger. *Hadron Identification with the Muon Setup of CBM*. https://git.cbm.gsi.de/CbmSoft/cbmroot_geometry/-/tree/master/setup. Accessed: 2021-06-07 (cit. on p. 13).
- [24] J. Heuser et al. *Technical design report for the CBM: Silicon Tracking System (STS)*. GSI Report. Collaboration FAIR: CBM, 2013 (cit. on p. 13).
- [25] *create_stsgeo_v19w.C – cbmroot Gitlab repository*. https://git.cbm.gsi.de/computing/cbmroot/-/blob/master/macro/sts/geometry/create_stsgeo_v19w.C. Accessed: 2021-06-11 (cit. on p. 13).
- [26] C. Höhne et al. *Technical design report for the CBM: Ring Imaging Cherenkov (RICH) Detector*. Technical Design Report. Collaboration FAIR: CBM, 2013 (cit. on p. 14).
- [27] S. Chattopadhyay et al. *Technical design report for the CBM: Muon Chambers (MuCh)*. Technical Design Report. Collaboration FAIR: CBM, 2015 (cit. on p. 14).
- [28] C. Blume et al. *Technical design report for the CBM: Transition Radiation Detector (TRD)*. Technical Design Report. Collaboration FAIR: CBM, 2018 (cit. on p. 14).
- [29] N. Herrmann et al. *Technical design report for the CBM: Time-of-Flight System (TOF)*. Technical Design Report. Collaboration FAIR: CBM, 2014 (cit. on p. 14).
- [30] F. Guber, I. Selyuzhenkov, CBM Collaboration et al. *Technical design report for the CBM: Projectile Spectator Detector (PSD)*. Technical Design Report. Collaboration FAIR: CBM, 2015 (cit. on p. 14).
- [31] A. Malakhov, A. Shabunov et al. *Technical design report for the CBM: Superconducting Dipole Magnet*. GSI Report. Collaboration FAIR: CBM, 2013 (cit. on p. 15).
- [32] M. Deveaux et al. ‘R&D on Monolithic Active Pixel Sensors for the CBM Vertex Detector’. In: *GSI Scientific Report 2003* (2004) (cit. on p. 17).
- [33] C. collaboration et al. ‘Letter of Intent for the Compressed Baryonic Matter Experiment at the Future Accelerator Facility in Darmstadt’. In: *Darmstadt* (Jan. 2004). Accessed: 2021-05-23 (cit. on p. 17).
- [34] ‘FAIR Baseline Technical Report – Accelerator and Scientific Infrastructure (Volume 2)’. In: (Mar. 2006) (cit. on p. 17).

- [35] T. Tischler. ‘Mechanical Integration of the Micro Vertex Detector for the CBM Experiment’. PhD thesis. Goethe-Universität Frankfurt, 2015, p. 262 (cit. on pp. 17, 22, 27, 28, 33, 35, 36).
- [36] P. Zyla et al. ‘Review of Particle Physics’. In: *Progress of Theoretical and Experimental Physics* 2020.8 (Aug. 2020). DOI: 10.1093/ptep/ptaa104 (cit. on pp. 17–19, 22, 24, 87).
- [37] S. Amar-Youcef. ‘Design and performance studies of the Micro-Vertex-Detector for the CBM experiment at FAIR’. Dissertation. Johann Wolfgang Goethe Universität Frankfurt am Main, 2011, VIII, 144 S. : Ill., graph. Darst., 30 cm. URL: <https://repository.gsi.de/record/49348> (cit. on pp. 17, 33).
- [38] L. Rossi et al. *Pixel Detectors – From Fundamentals to Applications*. Springer Berlin Heidelberg, 2006, p. 304. ISBN: 978-3-540-28332-4, 978-3-642-06652-8, 978-3-540-28333-1. DOI: 10.1007/3-540-28333-1 (cit. on pp. 18, 22).
- [39] M. Deveaux. ‘Progress on the radiation tolerance of CMOS Monolithic Active Pixel Sensors’. In: *Journal of Instrumentation* 14.11 (Nov. 2019), R11001. DOI: 10.1088/1748-0221/14/11/r11001. arXiv: 1909.05715 [physics.ins-det] (cit. on pp. 18, 19, 89, 90).
- [40] H. Bichsel. ‘A method to improve tracking and particle identification in TPCs and silicon detectors’. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 562.1 (June 2006), pp. 154–197. DOI: 10.1016/j.nima.2006.03.009 (cit. on p. 19).
- [41] R. Pleskac et al. ‘The FIRST experiment at GSI’. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 678 (June 2012), pp. 130–138. DOI: 10.1016/j.nima.2012.02.020 (cit. on p. 20).
- [42] M. Koziel. ‘Development of radiation hardened pixel sensors for charged particle detection’. PhD thesis. Strasbourg, IPHC, 2011 (cit. on pp. 20, 35).
- [43] D. Doering. ‘Untersuchungen zur Verbesserung der Strahlenhärte von CMOS-Sensoren zum Einsatz in Vertexdetektoren von Schwerionenexperimenten’. Gutachter: J. Stroth; Johann Wolfgang Goethe-Universität Frankfurt, Diss., 2015. Dr. Johann Wolfgang Goethe-Universität Frankfurt, 2015, 179 p. URL: <https://repository.gsi.de/record/184572> (cit. on pp. 20, 21, 93).

- [44] M. Deveaux. ‘Development of fast and radiation hard Monolithic Active Pixel Sensors (MAPS) optimized for open charm meson detection with the CBM - vertex detector’. PhD thesis. Johann Wolfgang Goethe-Universität Frankfurt, 2007 (cit. on pp. 20, 21).
- [45] N. Wermes. ‘Pixel Vertex Detectors’. In: *lectures notes SLAC Summer Institute* (2006). arXiv: physics/0611075 [physics.ins-det] (cit. on p. 22).
- [46] W. Allison. *Fundamental Physics for Probing and Imaging*. Oxford University Press, 2006 (cit. on p. 22).
- [47] D. Groom et al. *Atomic and Nuclear Properties of Materials*. <http://pdg.lbl.gov/2019/AtomicNuclearProperties/index.html>. Accessed: 2020-05-01 (cit. on p. 22).
- [48] A. collaboration. ‘Alignment of the ALICE Inner Tracking System with cosmic-ray tracks’. In: *Journal of Instrumentation* 5.03 (Mar. 2010), P03003. ISSN: 1748-0221. DOI: 10.1088/1748-0221/5/03/p03003. arXiv: 1001.0502 [physics.ins-det] (cit. on p. 23).
- [49] *Table de Radionucléides: ⁵⁵Fe*. http://www.nucleide.org/DDEP_WG/Nuclides/Fe-55_tables.pdf. Accessed: 2021-04-16 (cit. on p. 23).
- [50] E. Hussein. *Handbook on Radiation Probing, Gauging, Imaging and Analysis: Volume I Basics and Techniques*. Vol. 1. Springer Science & Business Media, 2003. ISBN: 1-4020-1294-2 (cit. on p. 23).
- [51] A. D. Angelis and M. Pimenta. *Introduction to Particle and Astroparticle Physics*. Springer International Publishing, 2018. DOI: 10.1007/978-3-319-78181-5 (cit. on p. 23).
- [52] ISO 80000-2:2019. *Quantities and units – Part 2: Mathematics*. Standard. Geneva, CH: International Organization for Standardization, Aug. 2019 (cit. on p. 24).
- [53] M. Deveaux. ‘Development of fast and radiation hard Monolithic Active Pixel Sensors (MAPS) optimized for open charm meson detection >with the CBM - vertex detector’. Theses. ULP Strasbourg - Goethe University Frankfurt, Mar. 2008. URL: <https://tel.archives-ouvertes.fr/tel-00392111> (cit. on p. 25).
- [54] M. Deveaux et al. *Technical design report for the: CBM Micro Vertex Detector (in preparation)*. Tech. rep. Draft TDR_CBM_MVD_ECE-FAIR_200521.pdf as of May 20. 2021 (cit. on pp. 25–29, 33, 41).
- [55] M. Deveaux. *Specifications of the MIMOSIS-1 sensor (draft)*. (internal). Mai 2016 (cit. on pp. 25, 43).
- [56] P. Klaus et al. *The MVD CAD Geometry Models*. CBM-TN-20005, Aug. 2020 (cit. on p. 29).

Bibliography

- [57] P. Klaus et al. ‘MVD keep-out volumes’. In: *CBM Progress Report 2018* (2019), p. 7. DOI: 10.15120/GSI-2019-01018 (cit. on p. 29).
- [58] P. Klaus et al. *MVD Keep-Out Volumes*. CBM-TN-19004, Mar. 2019 (cit. on p. 29).
- [59] G. C. and. ‘The MAPS-based ITS Upgrade for ALICE’. In: *Proceedings of The 28th International Workshop on Vertex Detectors — PoS(Vertex2019)*. Sissa Medialab, Feb. 2020. DOI: 10.22323/1.373.0003. arXiv: 2001.03042 [physics.ins-det] (cit. on p. 29).
- [60] B. A. et al and. ‘Technical Design Report for the Upgrade of the ALICE Inner Tracking System’. In: *Journal of Physics G: Nuclear and Particle Physics* 41.8 (July 2014), p. 087002. DOI: 10.1088/0954-3899/41/8/087002 (cit. on p. 29).
- [61] G. Contin. ‘The STAR PXL detector’. In: *Journal of Instrumentation* 11.12 (Dec. 2016), p. C12068. DOI: 10.1088/1748-0221/11/12/c12068 (cit. on p. 29).
- [62] *CbmRoot Gitlab Repository*. <https://git.cbm.gsi.de/computing/cbmroot>. Accessed: 2020-10-15 (cit. on p. 29).
- [63] P. Klaus et al. *The Digitizer for the MVD Detector*. CBM-CN-20005, June 2020 (cit. on p. 30).
- [64] P. Klaus et al. *The MVD simulation geometry models*. CBM-CN-20007, Aug. 2020 (cit. on pp. 30, 31).
- [65] S. Amar-Youcef et al. ‘R&D for the demonstrator of the CBM-Micro-Vertex Detector (MVD)’. In: *CBM Progress Report 2007* (2008), p. 31 (cit. on p. 33).
- [66] C. Schrader et al. ‘A Readout System for the CBM-MVD Demonstrator’. In: *CBM Progress Report 2007* (2008), p. 54 (cit. on p. 33).
- [67] C. Schrader. ‘A demonstrator for the Micro-Vertex Detector of the CBM experiment’. In: *Proceedings of XLVIII International Winter Meeting on Nuclear Physics — PoS(BORMIO2010)*. Sissa Medialab, Dec. 2010. DOI: 10.22323/1.103.0017 (cit. on p. 33).
- [68] C. Schrader. ‘A Readout System for the Micro-Vertex-Detector Demonstrator for the CBM experiment at FAIR’. Frankfurt (Main), Univ., Diss., 2011; JWG Universität Frankfurt, Diss., 2011. Dr. JWG Universität Frankfurt, 2011, p. 198. URL: <https://repository.gsi.de/record/49367> (cit. on p. 33).
- [69] M. Koziel et al. ‘Prototyping the CBM Micro-Vertex Detector’. In: *CBM Progress Report 2012* (2013), p. 3 (cit. on p. 33).

- [70] T. Tischler et al. ‘Mechanical Integration of the CBM MVD Prototype’. In: *CBM Progress Report 2012* (2013), p. 4 (cit. on p. 33).
- [71] B. Milanović et al. ‘Beamtime results of the MVD prototype DAQ network’. In: *CBM Progress Report 2012* (2013), p. 5 (cit. on p. 33).
- [72] Q. Li et al. ‘Online data processing with the CBM-MVD prototype’. In: *CBM Progress Report 2012* (2013), p. 6 (cit. on p. 33).
- [73] S. Amar-Youcef et al. ‘Results from the MVD prototype test CERN at the CERN-SPS’. In: *CBM Progress Report 2013* (2013), p. 16 (cit. on p. 33).
- [74] M. Deveaux et al. ‘A first generation prototype for the CBM micro vertex detector’. In: *Nuclear Instruments and Methods in Physics Research Section A – Proceedings of the 12th Pisa Meeting on Advanced Detectors*. Vol. 718. Elsevier BV, Aug. 2013, pp. 305–306. DOI: 10.1016/j.nima.2012.10.013 (cit. on p. 33).
- [75] M. Koziel et al. ‘PRESTO: PREcursor of the Second sTatiON of the CBM-MVD’. In: *CBM Progress Report 2014* (2015), p. 19 (cit. on pp. 34, 36).
- [76] M. Koziel et al. ‘Status of the CBM-MVD PRESTO project’. In: *CBM Progress Report 2015* (2016), p. 16 (cit. on pp. 34, 36).
- [77] M. Koziel et al. ‘Status of the CBM-MVD station precursor PRESTO’. In: *CBM Progress Report 2016* (2017), p. 13 (cit. on pp. 34, 36).
- [78] M. Koziel et al. ‘Update on the CBM-MVD PRESTO Module: Yield Studies’. In: *CBM Progress Report 2017* (2018), p. 14. DOI: 10.15120/GSI-2018-00485 (cit. on pp. 34, 36).
- [79] P. Klaus et al. ‘An ultra-low material budget Cu-based flexible cable for the CBM-MVD’. In: *CBM Progress Report 2014*. CBM, 2015, p. 55. DOI: 10.15120/GR-2015-1-MU-NQM-CBM-19 (cit. on pp. 34, 43).
- [80] J. Michel. ‘Development and Implementation of a New Trigger and Data Acquisition System for the HADES Detector’. PhD thesis. Goethe-Universität Frankfurt, 2012 (cit. on p. 34).
- [81] A. Neiser et al. ‘TRB3: a 264 channel high precision TDC platform and its applications’. In: *Journal of Instrumentation* 8.12 (2013), p. C12043 (cit. on pp. 34, 42).
- [82] A. Himmi et al. *MIMOSA26 User Manual v1.0*. http://www-rnc.lbl.gov/hft/hardware/docs/Phase1/M26_UserManual.pdf. 2008 (cit. on p. 35).
- [83] A. Himmi et al. *MIMOSA26 User Manual v1.5*. http://www.iphc.cnrs.fr/IMG/pdf/M26_UserManual_light.pdf. Mar. 2011 (cit. on p. 35).

Bibliography

- [84] J. Dreyling-Eschweiler and K. Hjansen. *DESY Wiki of the EUDET-type beam telescopes*. https://telescopes.desy.de/Main_Page. Accessed: 2020-12-18 (cit. on p. 35).
- [85] J. Behr. ‘Test beam measurements with the EUDET pixel telescope’. In: *EUDET-Report-2010-01 2* (2010) (cit. on p. 35).
- [86] C. Hu-Guo et al. ‘First reticule size MAPS with digital output and integrated zero suppression for the EUDET-JRA1 beam telescope’. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 623.1 (Nov. 2010), pp. 480–482. DOI: 10.1016/j.nima.2010.03.043 (cit. on p. 35).
- [87] B. Milanović. ‘Development of the readout controller for the CBM Micro-Vertex Detector’. PhD thesis. 2015 (cit. on pp. 35, 42).
- [88] M. Wiebusch. ‘Master Thesis - Development and Commissioning of the Next Generation CBM MVD Front-End Electronics’. Master’s Thesis. Goethe-Universität Frankfurt, 2014 (cit. on pp. 35, 36, 42, 43, 127, 128).
- [89] P. Klaus et al. ‘CBM-MVD: A Test Stand for the Thermal Characterization of MVD Stations’. In: *CBM Progress Report 2019* (2020), p. 11. DOI: 10.15120/GSI-2020-00904 (cit. on p. 36).
- [90] P. Klaus et al. ‘CBM-MVD: Measurements and Simulations on Thermal Performance’. In: *CBM Progress Report 2019* (2020), p. 12. DOI: 10.15120/GSI-2020-00904 (cit. on p. 36).
- [91] P. Klaus, J. Michel and J. Stroth. ‘The contribution of cables to the material budget of the CBM-MVD’. In: *Verhandlungen der Deutschen Physikalischen Gesellschaft* (2015) (cit. on p. 43).
- [92] M. Koziel et al. ‘Vacuum-compatible, ultra-low material budget Micro-Vertex Detector of the compressed baryonic matter experiment at FAIR’. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 845 (2017). Proceedings of the Vienna Conference on Instrumentation 2016, pp. 110–113. ISSN: 0168-9002. DOI: 10.1016/j.nima.2016.05.093 (cit. on p. 43).
- [93] M. Wiebusch et al. ‘The next generation CBM MVD front-end electronics’. In: *Verhandlungen der Deutschen Physikalischen Gesellschaft* (2015) (cit. on p. 43).
- [94] P. Klaus et al. ‘The Detector Control System for the MVD Prototype PRESTO’. In: *CBM Progress Report 2017* (2018), p. 12. DOI: 10.15120/GSI-2018-00485 (cit. on p. 47).
- [95] P. Moreira, A. Marchioro et al. ‘The GBT: a proposed architecture for multi-Gb/s data transmission in high energy physics’. In: (2007) (cit. on p. 47).

- [96] P. Moreira et al. ‘The GBT project’. In: (2009) (cit. on p. 47).
- [97] *EPICS License*. <https://epics.anl.gov/license/open.php>. Accessed: 2021-04-15 (cit. on p. 49).
- [98] L. R. Dalesio, A. J. Kozubal and M. R. Kraimer. ‘EPICS architecture’. In: (Jan. 1991) (cit. on p. 49).
- [99] J. Hill. ‘Channel access: A software bus for the LAACS’. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 293.1-2 (Aug. 1990), pp. 352–355. DOI: 10.1016/0168-9002(90)91459-o (cit. on p. 49).
- [100] L. R. Dalesio et al. ‘The experimental physics and industrial control system architecture: past, present, and future’. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 352.1-2 (Dec. 1994), pp. 179–184. DOI: 10.1016/0168-9002(94)91493-1 (cit. on p. 49).
- [101] M. R. Kraimer, B. K. Cha and M. Anderson. ‘Alarm Handler for the Advanced Photon Source Control System’. In: (Jan. 1991) (cit. on p. 49).
- [102] M. Knott et al. *EPICS: A Control System Software Co-Development Success Story*. https://epics.anl.gov/EpicsDocumentation/EpicsGeneral/epics_success.html. Accessed: 2020-10-06. Sept. 1993 (cit. on p. 49).
- [103] *LabVIEW*. <https://www.ni.com/en-us/shop/labview.html>. Accessed: 2021-04-15 (cit. on p. 49).
- [104] U. Grundinger, ed. *GSI Scientific Report 2000 [GSI Report 2001-1]*. Vol. 2001-1. GSI Report. Wissenschaftlicher Ergebnisbericht der GSI, GSI Annual Report. Darmstadt: GSI, 2001, 282 p. URL: <https://repository.gsi.de/record/53531> (cit. on p. 49).
- [105] J. Chaize et al. ‘TANGO: An Object Oriented Control System Based on CORBA’. en. In: (1999). <http://www.tango-controls.org/> and <http://cds.cern.ch/record/532662/files/wa2i01.pdf> (cit. on p. 49).
- [106] A. Buteau, P. Betinelli and B. Gagey. ‘Status of Soleil control systems’. In: *EPAC*. Vol. 6. 2006, pp. 3053–3055 (cit. on p. 49).
- [107] S. Hartman et al. ‘SNS Instrument Data Acquisition And Controls’. In: *14th International Conference on Accelerator & Large Experimental Physics Control Systems (ICALEPCS2013), San Francisco, Calif., October 6-11, 2013 Proceedings publ.online Paper TUPPC076 JACoW, 2013*. 2013, pp. 755–758. ISBN: 978-3-954-50139-7. URL: <http://accelconf.web.cern.ch/ICALEPCS2013/papers/tuppc076.pdf> (cit. on p. 49).

Bibliography

- [108] K. Rosenkranz. *Konzeption und Umsetzung eines Slow-Control-Systems für die HADES-Driftkammern*. 2002. URL: http://www.uni-frankfurt.de/fb/fb13/ikf/Arbeitsgruppen/Highenergy/Dateien/diplom/diploma%5C_klaus%5C_rosenkranz%5C_pdf.pdf (cit. on p. 49).
- [109] A. Gil et al. ‘Control and Monitoring System for the HADES RPC Detector’. In: *2009 16th IEEE-NPSS Real Time Conference*. 2009, pp. 351–353 (cit. on p. 49).
- [110] A. Augustinus et al. ‘Computing architecture of the ALICE detector control system’. In: *Proc. of Int. Conf. on Accelerator and Large Experimental Physics Control Systems*. 2011 (cit. on p. 49).
- [111] M. Tkáčik et al. ‘FRED—Flexible Framework for Frontend Electronics Control in ALICE Experiment at CERN’. In: *Processes* 8.5 (May 2020), p. 565. ISSN: 2227-9717. DOI: 10.3390/pr8050565 (cit. on p. 49).
- [112] M. Kraimer et al. *EPICS Application Developer’s Guide – EPICS Base Release 3.16.2*. <https://epics.anl.gov/base/R3-16/2-docs/AppDevGuide/AppDevGuide.html>. Accessed: 2020-07-05. 2018 (cit. on p. 50).
- [113] J. O. Hill and R. Lange. *EPICS Channel Access 4.13.1 Reference Manual*. <https://epics.anl.gov/base/R7-0/4-docs/CAref.html>. Accessed: 2020-07-05. 2018 (cit. on p. 50).
- [114] *Record Reference Documentation*. <https://epics.anl.gov/base/R7-0/4-docs/RecordReference.html>. Accessed: 2020-07-05 (cit. on p. 50).
- [115] *pvDataCPP documentation – PVData C++ 8.0.1*. <https://epics-base.github.io/pvDataCPP/>. Accessed: 2020-07-05. 2019 (cit. on p. 50).
- [116] M. Sekoranja et al. *pvAccess Protocol Specification (Revision bc10643)*. <https://github.com/epics-base/pvAccessCPP/wiki/protocol>. Accessed: 2020-07-05. 2019 (cit. on p. 50).
- [117] *EPICS Tech-talk Mailing List*. <https://epics.anl.gov/tech-talk/>. Accessed: 2020-12-19 (cit. on p. 50).
- [118] *ICALEPCS - Conferences*. <https://www.icalepcs.org/conferences.html>. Accessed: 2020-12-19 (cit. on p. 50).
- [119] *EPICS Collaboration Meetings*. <https://epics.anl.gov/meetings.php>. Accessed: 2020-12-19 (cit. on p. 50).
- [120] J. de Cuveland et al. *Definition of the CBM Computing Projects*. CBM-GN-18003, Sept. 2018 (cit. on p. 51).
- [121] A. Uchiyama, M. Komiyama and M. Nishimura. ‘An attempt to implement private cloud for sharing of operator interfaces in RIBF control system’. In: *PASJ2015*. 2015, pp. 1274–1277 (cit. on p. 54).

- [122] C. Pascual. *Email: Epics Docker Container*. <https://epics.anl.gov/tech-talk/2015/msg01969.php>. Accessed: 2020-05-23. 16 Dec 2015 (cit. on p. 54).
- [123] I. Arredondo and J. Jugo. ‘Containerized Control Structure for Accelerators’. In: *16th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS’17), Barcelona/Spain, 8-13 October 2017*. JACoW, Geneva, Switzerland. Jan. 2018, p. 816. DOI: 10.18429/JACoW-ICALEPCS2017-TUPHA170 (cit. on p. 54).
- [124] L. Fernández et al. ‘How to Build and Maintain a Development Environment for the Development of Controls Software Applications: An Example of "Infrastructure as Code" within the Physics Accelerator Community.’ In: *Proc. of International Particle Accelerator Conference (IPAC’16), May 8-13, 2016 (Busan, Korea)*. 7. Geneva, Switzerland: JACoW, June 2016, pp. 2781–2783. ISBN: 978-3-954-50147-2. DOI: 10.18429/JACoW-IPAC2016-WEPOR050 (cit. on p. 54).
- [125] X. Resende et al. ‘High-Level Applications for the Sirius Accelerator Control System’. In: *Proc. 10th International Particle Accelerator Conference (IPAC’19), 19-24 May 2019 (Melbourne, Australia)*. 10. Geneva, Switzerland: JACoW Publishing, June 2019, pp. 2462–2465. ISBN: 978-3-954-50208-0. DOI: 10.18429/JACoW-IPAC2019-WEPGW003 (cit. on p. 54).
- [126] A. A. Derbenev. ‘Improvement of EPICS Software Deployment at NSLS-II’. In: (cit. on p. 54).
- [127] *Containers in Controls Workshop – 6 October 2019*. <https://indico.cern.ch/event/823284/>. Accessed: 2020-12-20. Oct. 2019 (cit. on p. 54).
- [128] F. Feldbauer. *Container virtualization for EPICS*. <https://indico.gsi.de/event/9524/contributions/41177/attachments/29562/36756/epics-tut.pdf>. Accessed: 2020-12-20. Nov. 2019 (cit. on p. 54).
- [129] *Docker – Container Virtualization Software*. <https://www.docker.com/>. Accessed: 2021-04-15 (cit. on pp. 54, 56).
- [130] P. Klaus. *CaRecPlay – Channel Access Recorder and Player*. <https://github.com/pklaus/carecplay>. Accessed: 2020-09-15 (cit. on p. 57).
- [131] *epics-controls.org: What PV Save and Restore Tools are available*. <https://epics-controls.org/resources-and-support/documents/howto-documents/pv-save-restore-tools/>. Accessed: 2020-06-30 (cit. on p. 57).

Bibliography

- [132] P. H. K. GmbH. *Huber SpyLight and SpyControl – Software for device control, visualisation and documentation – Version 2.0*. https://www.huber-online.com/en/download_software.aspx. Accessed: 2021-04-14. June 2020 (cit. on p. 59).
- [133] P. H. K. GmbH. *Handbook Data Communication – Version 1.3/04.08*. https://www.huber-online.com/download/manuals/archive/Manual_Data_Communication_EN.pdf. Accessed: 2020-10-10. 2012 (cit. on p. 59).
- [134] P. Klaus. *jpnevulator.py – An implementation of jpnevulator in Python*. <https://github.com/pklaus/jpnevulator.py>. Accessed: 2020-10-10. 2016 (cit. on p. 59).
- [135] *Wireshark – a free and open-source packet analyzer*. <https://www.wireshark.org/>. Accessed: 2021-05-27 (cit. on p. 60).
- [136] E. R. Tufte. *The Visual Display of Quantitative Information*. eng. 2nd. Cheshire, Conn.: Graphics Press, 2001, p. 197. ISBN: 978-0-961-39214-7 (cit. on pp. 60, 62).
- [137] M. A. Cohen, D. C. Dennett and N. Kanwisher. ‘What is the Bandwidth of Perceptual Experience?’ In: *Trends in Cognitive Sciences* 20.5 (May 2016), pp. 324–335. DOI: 10.1016/j.tics.2016.03.006 (cit. on p. 60).
- [138] K. Kasemir. *Control System Studio Guide*. <http://cs-studio.sourceforge.net/docbook/>. Accessed: 2020-08-30. 2018 (cit. on p. 60).
- [139] G. C. Tony Stone Eric Berryman. *Control System Studio Website*. <http://controlsystemstudio.org/>. Accessed: 2020-08-30. 2020 (cit. on p. 60).
- [140] K. Kasemir et al. *Control System Studio Guide*. <https://github.com/ControlSystemStudio/cs-studio>. Accessed: 2020-08-30. 2020 (cit. on p. 60).
- [141] C. Rosati, K.-U. Kasemir and K. Shroff. ‘JavaFX and CS-Studio: Benefits and Disadvantages in Developing the Next Generation of Control System Software’. In: *16th International Conference on Accelerator and Large Experimental Physics Control Systems*. 2018, TUPHA154. DOI: 10.18429/JACoW-ICALEPCS2017-TUPHA154 (cit. on p. 60).
- [142] K. Shroff and K. Kasemir. ‘Phoebus Architecture’. In: *EPICS Collaboration Meeting*. <https://epics.anl.gov/meetings/2018-06/talks/06-14/AM/4.5-Phoebus-Architecture.pdf>. Aug. 2018 (cit. on p. 60).
- [143] O. J. Artz. ‘Bachelor Thesis - Development of Components of the Detector Control System for the Prototype of the Second Station of the Micro Vertex Detector of the Compressed Baryonic Matter Experiment’. Bachelor’s Thesis. Goethe-Universität Frankfurt, 2018 (cit. on p. 60).

- [144] P. Klaus. *Web Dashboard for Process Variables of an EPICS-based Control System*. https://github.com/pklaus/epics_dash. Accessed: 2020-09-06 (cit. on p. 62).
- [145] E. R. Tufte. *Beautiful Evidence*. 1st ed. Graphics Press, 2006. ISBN: 978-0-961-39217-8 (cit. on p. 62).
- [146] A. Weber et al. *devTrbnet*. <http://jspc29.x-matter.uni-frankfurt.de/git/?p=epics.git;a=shortlog;h=refs/heads/devtrbnet2>. Accessed: 2021-06-02 (cit. on p. 64).
- [147] P. Klaus and T. Weber. *PyTrbNet – a Python package wrapping libtrbnet.so from trbnettools*. <https://github.com/pklaus/pytrbnet>. Accessed: 2021-06-01 (cit. on p. 64).
- [148] F. Feldbauer. *Channel Access Gateway and Access Security, PANDA IV. Collaboration Meeting*. https://indico.gsi.de/event/4324/contributions/19856/attachments/14681/18635/feldbauer_151130_cagateway.pdf. Accessed: 2020-06-19 (cit. on p. 65).
- [149] K. Kasemir and G. Carcassi. *Chapter 11. Archive System, Control System Studio Guide - For installers and maintainers of CS-Studio*. <http://cs-studio.sourceforge.net/docbook/ch11.html>. Accessed: 2020-05-19. 2018 (cit. on p. 67).
- [150] P. Zumbach. *RDB Archiver based on CS-Studio – Hades DAQ Documentation – Hades Wiki*. <https://hades-wiki.gsi.de/foswiki/bin/view/DaqSlowControl/SCSArchiverCSSstudio>. Accessed: 2021-06-02 (cit. on p. 67).
- [151] R. Brun and F. Rademakers. ‘ROOT – An object oriented data analysis framework’. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Assoc. Equipment* 389.1-2 (Apr. 1997), pp. 81–86. DOI: 10.1016/S0168-9002(97)00048-X (cit. on p. 68).
- [152] M. Shankar et al. ‘The EPICS Archiver Appliance’. In: (2015). DOI: 10.18429/JACoW-ICALPCS2015-WEPGF030 (cit. on p. 69).
- [153] M. Shankar. *The EPICS Archiver Appliance Website*. https://slacmshankar.github.io/epicsarchiver_docs/. Accessed: 2020-06-13 (cit. on pp. 69, 120).
- [154] S. Marsching. *The Cassandra Archiver – A Scalable Solution for Archiving EPICS Channels*. <https://oss.aquenos.com/cassandra-pv-archiver/>. Accessed: 2020-06-19 (cit. on pp. 69, 71, 121).
- [155] S. Marsching. *Cassandra PV Archiver Reference Manual – Version 3.2.6*. <https://oss.aquenos.com/cassandra-pv-archiver/docs/3.2.6/manual/htmlsingle/>. Accessed: 2021-05-28 (cit. on pp. 69, 121).

Bibliography

- [156] K. Kasemir. *Earliest Changeset mentioning CSS/RDB Archiver in CS-Studio Source Code on 2007-11-19*. <http://cs-studio.hg.sourceforge.net/hgweb/cs-studio/cs-studio/rev/86024be21afb>. Accessed: 2020-06-18. 2007 (cit. on p. 69).
- [157] M. Hirose et al. 'Present status of CSS Archiver and Channel Archiver at SuperKEKB'. In: (TUP091). PASJ2016. https://www.pasj.jp/web_publish/pasj2016/proceedings/PDF/TUP0/TUP091.pdf accessed 2020-06-19. PASJ2016 TUP091, 2016 (cit. on p. 69).
- [158] M. Shankar. *EPICS Archiver Appliance – Initial version from the benchmarks project for the archiver appliance product*. <https://sourceforge.net/p/epicsarchiverap/code/ci/86b4b7c16733705561fb22d501d64f55d67c52b5/>. Accessed: 2020-06-13. Apr. 2011 (cit. on p. 69).
- [159] S. Marsching et al. 'Scalable Archiving with the Cassandra Archiver for CSS'. In: *Proc. of International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'13), October 2013* (San Francisco, California). 16. Geneva, Switzerland: JACoW, Mar. 2014. ISBN: 978-3-954-50139-7. URL: <https://accelconf.web.cern.ch/ICALEPCS2013/papers/tuppc004.pdf> (cit. on p. 71).
- [160] S. Marsching. *The Cassandra Archiver – A Scalable Solution for Archiving EPICS Channels*. https://epics.anl.gov/meetings/2013-05/dms/EPICS2013/presentations/thursday/Session_2/2013-05-02_EPICS_Meeting_Cassandra_Archiver-ppt/2013-05-02_EPICS_Meeting_Cassandra_Archiver.ppt.pdf. Accessed: 2020-06-19. 2013 (cit. on pp. 71, 121).
- [161] P. Klaus. *Docker Image for Cassandra PV Archiver*. . Accessed: 2021-06-02 (cit. on p. 71).
- [162] T. Triffterer et al. *Technical Design Report for the: PANDA Detector Control System*. PANDA Collaboration, Oct. 2020 (cit. on p. 71).
- [163] *Apache Cassandra – Open Source NoSQL Database*. <https://cassandra.apache.org/>. Accessed: 2021-05-28 (cit. on pp. 71, 121).
- [164] D. Allan, K. Lauer, T. A. Caswell et al. *Caproto – A bring-your-own-IO implementation of the EPICS Channel Access protocol*. <https://caproto.github.io/caproto/v0.8.0>. Accessed: 2021-06-02 (cit. on p. 71).
- [165] P. Klaus. *caproto-archiver – An EPICS Archiver based on caproto (Proof of Concept)*. <https://github.com/pklaus/caproto-archiver>. Accessed: 2021-06-02 (cit. on p. 71).
- [166] *Channel Access Protocol Specification (Revision fea83e08)*. https://docs.epics-controls.org/en/latest/specs/ca_protocol.html. Accessed: 2020-07-05. 2019 (cit. on p. 71).

- [167] C. Sammut and G. I. Webb. *Encyclopedia of Machine Learning and Data Mining*. Springer, 2017 (cit. on p. 72).
- [168] *AVS Reference Guide – Glossary*. <http://www.aip.org/avsguide/refguide/glossary.html>. Accessed: 2013-09-03 (cit. on p. 75).
- [169] C. Müntz et al. *MVD: Assessing the Thermal Performance*. CBM-TN-19007, Aug. 2019 (cit. on p. 77).
- [170] *coracon KS 6 - Ökologische Wärmeträgerflüssigkeit für Geothermie*. https://www.aqua-concept.pl/fileadmin/content/PDFs/Waermetraegerfluuids/Waermetraegerfluessigkeiten/Geothermie/PI_coracon_KS_6_170104_DE.pdf. Accessed: 2021-04-16 (cit. on p. 83).
- [171] *Ethylene Glycol – Product Guide*. https://www.meglobal.biz/wp-content/uploads/2019/10/MEG-0002_MEG_Guide_Rev_Sep_2019.pdf. Accessed: 2021-04-16 (cit. on p. 83).
- [172] *Huber CC-405 Datasheet – Refrigerated Heating Circulator Bath with air-cooled cooling machine*. https://www.huber-online.com/download/pdf_datasheets/englisch/2017.0001.01.PDF. Accessed: 2021-04-16 (cit. on p. 83).
- [173] *ESTAR : Stopping Power and Range Tables for Electrons*. <https://physics.nist.gov/PhysRefData/Star/Text/ESTAR.html>. Accessed: 2021-05-22 (cit. on p. 86).
- [174] *NIST: X-Ray Mass Attenuation Coefficients - Aluminum*. <https://physics.nist.gov/PhysRefData/XrayMassCoef/ElemTab/z13.html>. Accessed: 2021-05-22 (cit. on p. 87).
- [175] B. Boitrelle. ‘Development of a double-sided ladder for tracking in high-energy physics’. PhD thesis. 2017 (cit. on pp. 89, 90).
- [176] K. Hesse, I. H. Sloan and R. S. Womersley. ‘Numerical integration on the sphere’. In: *Handbook of Geomathematics*. 2010 (cit. on p. 90).
- [177] V. I. Lebedev and D. Laikov. ‘A quadrature formula for the sphere of the 131st algebraic order of accuracy’. In: *Doklady Mathematics*. Vol. 59. 3. Pleiades Publishing, Ltd. 1999, pp. 477–481 (cit. on p. 90).
- [178] V. Lebedev. ‘Quadratures on a sphere’. In: *USSR Comput. Methods Math. Phys* 16 (2 1976), pp. 10–24 (cit. on p. 90).
- [179] Q. Li et al. ‘A FPGA-based Cluster Finder for CMOS Monolithic Active Pixel Sensors of the MIMOSA-26 Family’. In: *Journal of Physics: Conference Series* 513.2 (June 2014), p. 022021. DOI: 10.1088/1742-6596/513/2/022021 (cit. on pp. 93, 95).

Bibliography

- [180] M. Ester et al. ‘A density-based algorithm for discovering clusters in large spatial databases with noise.’ In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. Vol. 96. 34. AAAI Press, 1996, pp. 226–231 (cit. on pp. 94, 95).
- [181] F. Pedregosa et al. ‘Scikit-learn: Machine Learning in Python’. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (cit. on p. 95).
- [182] M. Ankerst et al. ‘OPTICS: Ordering Points To Identify the Clustering Structure’. In: ACM Press, 1999, pp. 49–60 (cit. on p. 95).
- [183] X. Liu. private communication. 3rd Feb. 2013 (cit. on p. 98).
- [184] G. White et al. ‘The EPICS Software Framework Moves from Controls to Physics’. In: *Proc. of the 10th International Particle Accelerator Conference (IPAC’19), 19-24 May 2019* (Melbourne, Australia). 10. Geneva, Switzerland: JACoW Publishing, June 2019, pp. 1216–1218. ISBN: 978-3-954-50208-0. DOI: 10.18429/JACoW-IPAC2019-TUZZPLM3 (cit. on pp. 106, 107).
- [185] A. Phatak et al. ‘Multi-input Multi-output Self-learning-Based Control System’. In: *International Conference on Advanced Computing Networking and Informatics*. Springer Singapore, Nov. 2018, pp. 329–337. DOI: 10.1007/978-981-13-2673-8_35 (cit. on p. 106).
- [186] K. Brown et al. ‘Experience with Machine Learning in Accelerator Controls’. In: *Proc. of International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS’17), 8-13 October 2017* (Barcelona, Spain). 16. Geneva, Switzerland: JACoW, Jan. 2018, pp. 258–264. ISBN: 978-3-954-50193-9. DOI: 10.18429/JACoW-ICALEPCS2017-TUCPA03 (cit. on p. 106).
- [187] C. Chu et al. ‘HEPS Controls Status Update’. In: *Proc. 12th International Workshop on Emerging Technologies and Scientific Facilities Controls (PCaPAC’18), Hsinchu, Taiwan, 16-19 October 2018* (Hsinchu, Taiwan). 12. Geneva, Switzerland: JACoW Publishing, Jan. 2019, pp. 14–16. ISBN: 978-3-954-50200-4. DOI: 10.18429/JACoW-PCaPAC2018-WEC4 (cit. on p. 106).
- [188] A. L. Edelen et al. ‘Neural Networks for Modeling and Control of Particle Accelerators’. In: *IEEE Transactions on Nuclear Science* 63.2 (2016), pp. 878–897 (cit. on p. 106).
- [189] J. Hanten et al. ‘Enhancement of the S-DALINAC Control System with Machine Learning Methods’. In: *Proc. IBIC’19* (Malmö, Sweden). International Beam Instrumentation Conference 8. JACoW Publishing, Geneva, Switzerland, Nov. 2019, pp. 475–478. ISBN: 978-3-954-50204-2. DOI: 10.18429/JACoW-IBIC2019-WEBO04 (cit. on p. 106).

- [190] C. Haskins. ‘EPICS and Open Source Data Analytics Platforms’. In: *Proc. of International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS’17), 8-13 October 2017* (Barcelona, Spain). 16. Geneva, Switzerland: JACoW, Jan. 2018, pp. 1420–1422. ISBN: 978-3-954-50193-9. DOI: 10.18429/JACoW-ICALEPCS2017-THPHA032 (cit. on pp. 106, 107).
- [191] R. Müller et al. ‘Modernization of Experimental Data Taking at BESSY II’. In: *International Conference on Accelerator and Large Experimental Physics Control Systems 17* (Aug. 2020), pp. 65–69. ISSN: 2226-0358. DOI: 10.18429/JACoW-ICALEPCS2019-MOCPL02 (cit. on p. 107).
- [192] M. D. Jones et al. *ess-dmsc/streaming-data-types: BrightnESS-Software-Deliverable-MS35*. Version v1.0.0. June 2018. DOI: 10.5281/zenodo.1298378 (cit. on p. 107).
- [193] D. Werder et al. ‘EPICS Data Streaming and HDF File Writing for ESS Benchmarked Using the Virtual AMOR Instrument’. In: *Proc. of International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS’17), 8-13 October 2017* (Barcelona, Spain). 16. Geneva, Switzerland: JACoW, Jan. 2018, pp. 1815–1819. ISBN: 978-3-954-50193-9. DOI: 10.18429/JACoW-ICALEPCS2017-THPHA167 (cit. on p. 107).
- [194] M. Christensen et al. ‘Software-based data acquisition and processing for neutron detectors at European Spallation Source—early experience from four detector designs’. In: *Journal of Instrumentation* 13.11 (Nov. 2018), T11002. DOI: 10.1088/1748-0221/13/11/t11002 (cit. on p. 107).
- [195] M. Clarke et al. ‘Live Visualisation of Experiment Data at ISIS and the ESS’. In: *Proc. of International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS’17), 8-13 October 2017* (Barcelona, Spain). 16. Geneva, Switzerland: JACoW, Jan. 2018, pp. 431–434. ISBN: 978-3-954-50193-9. DOI: 10.18429/JACoW-ICALEPCS2017-TUPHA029 (cit. on p. 107).
- [196] N. H. Weiderman et al. *Approaches to Legacy System Evolution*. Tech. rep. Carnegie-Mellon Univ Pittsburgh PA Software Engineering Inst, 1997 (cit. on p. 109).
- [197] R. C. Seacord et al. *Modernizing Legacy Systems*. Addison-Wesley Professional, 2003. ISBN: 978-0-321-11884-4 (cit. on p. 109).
- [198] S. Lenga. ‘Modernization of Monolithic Legacy Applications towards a Microservice Architecture with ExplorViz’. Master thesis. Kiel University, June 2019. URL: <http://eprints.uni-kiel.de/48007/> (cit. on p. 109).

Bibliography

- [199] L. Bass, I. Weber and L. Zhu. *DevOps: A Software Architect's Perspective*. 1st ed. SEI Series in Software Engineering. Addison-Wesley Professional, 2015. ISBN: 978-0-134-04984-7 (cit. on p. 109).
- [200] H. Knoche and W. Hasselbring. 'Using Microservices for Legacy Software Modernization'. In: *IEEE Software* 35.3 (May 2018), pp. 44–49. DOI: 10.1109/ms.2018.2141035 (cit. on p. 109).
- [201] *Application Modernization*. <https://www.ibm.com/cloud/learn/application-modernization>. Accessed: 2020-11-05. Oct. 2019 (cit. on p. 109).
- [202] H. Knoche and W. Hasselbring. 'Experience with Microservices for Legacy Software Modernization'. en. In: (2019). DOI: 10.18420/SE2019-29 (cit. on p. 109).
- [203] W. Hasselbring. 'Software Architecture: Past, Present, Future'. In: *The Essence of Software Engineering*. Springer International Publishing, 2018, pp. 169–184. DOI: 10.1007/978-3-319-73897-0_10 (cit. on p. 109).
- [204] *Docker Compose – A tool for defining and running multi-container Docker applications*. <https://docs.docker.com/compose/>. Accessed: 2021-05-28 (cit. on p. 113).
- [205] Intel. *Multus-CNI Github Repository – Readme*. <https://github.com/intel/multus-cni#readme>. Accessed: 2020-07-29. 2020 (cit. on pp. 114, 116).
- [206] *macvlan plugin (Github Repository)*. <https://github.com/container networking/plugins/tree/master/plugins/main/macvlan>. Accessed: 2020-07-29. 2020 (cit. on pp. 114, 116).
- [207] N. Chaudhry et al. *Stream data management*. Vol. 30. Springer Science & Business Media, 2005. ISBN: 978-0-387-25229-2 (cit. on p. 114).
- [208] A. Arasu, M. Cherniack, E. Galvez et al. 'Linear road: a stream data management benchmark'. In: *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*. 2004, pp. 480–491 (cit. on p. 114).
- [209] W. Wingerath, N. Ritter and F. Gessert. *Real-Time & Stream Data Management: Push-Based Data in Research & Practice*. Springer, 2019 (cit. on p. 114).
- [210] B. Babcock et al. 'Models and Issues in Data Stream Systems'. In: *Proceedings of the Twenty-First ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. PODS '02. Madison, Wisconsin: Association for Computing Machinery, 2002, pp. 1–16. ISBN: 1-5811-3507-6. DOI: 10.1145/543613.543615 (cit. on p. 115).
- [211] T. Mooney. 'synApps: EPICS Application Software for Synchrotron Beamlines and Laboratories'. In: *PcaPAC2010* (2010) (cit. on p. 116).

- [212] T. Mooney et al. *synApps Website*. <https://www.aps.anl.gov/BCDA/synApps>. Accessed: 2020-08-18 (cit. on p. 116).
- [213] *Prometheus – Monitoring system and time series database*. <https://prometheus.io/>. Accessed: 2021-05-28 (cit. on p. 117).
- [214] *Grafana: The open observability platform*. <https://grafana.com>. Accessed: 2021-05-28 (cit. on p. 117).
- [215] *Prometheus Documentation – What is Prometheus?* <https://prometheus.io/docs/introduction/overview/>. Accessed: 2020-08-12 (cit. on p. 118).
- [216] R. Lange, D. Zimoch et al. *Channel Access Gateway*. <https://github.com/epics-extensions/ca-gateway> and <https://epics.anl.gov/extensions/gateway>. Accessed: 2020-08-30 (cit. on p. 118).
- [217] M. Davidsaver et al. *PVA Gateway v3.4.2*. <https://mdavidsaver.github.io/p4p/gw.html>. Accessed: 2020-08-30. 2020 (cit. on p. 119).
- [218] *pva2pva v1.2.2 – Home of QSRV and pvAccess 2 pvAccess gateway*. <https://epics-base.github.io/pva2pva/> and <https://github.com/epics-base/pva2pva>. Accessed: 2020-07-05. 2019 (cit. on p. 119).
- [219] P. Klaus. *EPICS Archiver Benchmark – Scaling Test Tools*. <https://github.com/pklaus/epics-archiver-benchmark>. Accessed: 2021-05-28 (cit. on p. 119).

Acknowledgement

At this point I would like to express my sincere thanks to all those who stood by me during my doctorate, be it as role models or with professional advice or other support. First and foremost, my supervisor **Prof. Dr. Joachim Stroth**. He gave me the fantastic opportunity to work on the CBM micro vertex detector and helped me to keep the focus of my work. Furthermore, I would like to thank **Dr. Christian Müntz**, from whom I was able to learn a lot about the operation and necessities of an already running experiment – HADES. He always provided me with valuable advice and his positive view of things kept me from despairing more than once. I would also like to thank my office colleague **Dr. Michal Koziel**, from whom I could learn a lot about the mechanical integration of detectors and from whom I would be happy if we could go on a mountain bike ride together again. I would like to thank **Dr. Jan Michel** for his dry humour, the great support in circuit design and the help with the TRB-3 readout platform, which hardly anyone understands as well as he does. I have had countless in-depth scientific discussions with **PD Dr. Michael Deveaux**, and I admire his ability to highlight important aspects with pointed statements. I am very grateful to **Dr. Peter Zumbruch** from GSI for his start-up help with EPICS and the great technical exchange with him and Marcel Bajdel on EPICS in container-virtualized environments. I benefited greatly from the previous work by **Dr. Tobias Tischler**, **Dr. Borislav Milanovic**, **Dr. Michael Wiebusch** and **Dr. Dennis Döring** on the MVD and have many fond memories of scientific exchanges and conferences. I would also like to thank my students whom I had the privilege of supervising myself during my time as a PhD student, **Daniela Mijatovic** and **Ole Artz**. I have always found it an honour to be able to pass on my knowledge and have also been able to develop personally in the process. I would also like to thank the postdocs **Dr. Manuel Lorenz**, **Dr. Samir Amar-Youcef** and **Dr. Ingo Fröhlich**, who were also able to help me with specific questions. Furthermore, I had fruitful discussions about the MVD and its technologies (and good times) with the following fellow students (list not exhaustive): **Adrian Weber**, **Florian Marx**, **Philipp Sitzmann**, **Benjamin Linnik**, **Qiyang Li**, **Gisa Kretschmar**, **Benedict Arnoldi-Meadows**.

I would also like to thank the entire community of the CBM Experiment for allowing me to be part of this exciting large-scale research project. In particular, I would like to thank my mentor at HGS-HiRE, **Volker Friese**, for his advice.

Acknowledgement

In addition, I'm also grateful to **Prof. Dr. Henner Büsching**, the coordinator and managing director of the graduate schools H-QM and HGS-HIRe, respectively, of which I was a member, for his tireless commitment to the students, e.g. in the form of lecture weeks and soft-skill courses. Furthermore, my thanks also go to the workshops at the IKF, the locksmith's shop and the precision mechanics. I would also like to thank **Norbert Bialas**, **Roland Weirich** and **Stefan Schreiber** from the electronics department for their support with electronics projects and 3D printing rapid prototyping. Besides, I would also like to take this opportunity to thank the administration at the IKF, especially **Marianne Frey** and **Claudia Freudenberger** for their kind-hearted manner and extensive knowledge of the work processes at the Institute.

Last but not least, I would also like to thank **my family**, first and foremost **my wife**. I have received a lot of guidance and support from them over the years.