Inaugural-Dissertation

zur
Erlangung der Doktorwürde
der
Naturwissenschaftlich-Mathematischen
Gesamtfakultät
der
Ruprecht-Karls-Universität
Heidelberg

vorgelegt von
Diplom-Physiker Marcus Gutfleisch
aus Heidelberg

Tag der mündlichen Prüfung:
28.03.2006

Local Signal Processing of the ALICE Transition Radiation Detector at LHC (CERN)

Gutachter: Prof. Dr. Volker Lindenstruth

Prof. Dr. Bernd Jähne

Lokale Signalverarbeitung des ALICE Übergangsstrahlungsdetektors am LHC (CERN)

Beim Schwerionenexperiment ALICE am LHC (CERN) werden innerhalb des Übergangsstrahlungsdetektors bereits Teile des Datennahme- und Triggersystems integriert. Hierfür wurde ein aus zwei Mikrochips bestehendes Modul entwickelt. Die Detektorsignale werden vorverstärkt und geformt (Preamplifier and Shaper Chip, PASA). Danach werden sie analog-digital-gewandelt und weiterverarbeitet (Tracklet Processor Chip, TRAP).

Diese Arbeit beschreibt die digitale Signalverarbeitung im TRAP-Chip. Die Eingangssignale werden digital gefiltert und durch einen Preprozessor sowie vier CPUs hinsichtlich Teilchenspurabschnitten untersucht. In der Arbeit wird der Bogen gespannt vom Hardware-Entwurf von Filter und Preprozessor über deren Kalibration, die Programmierung der CPUs bis hin zu ersten Anwendungsstudien an einem Prototypensystem.

Local Signal Processing of the ALICE Transition Radiation Detector at LHC (CERN)

The transition radiation detector of the heavy ion experiment ALICE at LHC (CERN) integrates parts of the data acquisition and trigger system. Therefore, a multi chip module has been developped which incorporates two microchips. Detector signals are preamplified and shaped (Preamplifier and Shaper Chip, PASA). Thereafter they are converted from analog to digital and are processed (Tracklet Processing Chip, TRAP).

This thesis describes the digital signal processing of the TRAP chip. The input signals are filtered digitally. Then, they are analyzed by a preprocessor and four CPUs with respect to segments of tracks. The thesis covers the complete development from hardware design of filter and preprocessor, their calibration, programming of the CPUs, up to first application studies on a prototype system.

Contents

List of Figures

List of Tables

1

Introduction

In collision experiments, the structure of matter is analyzed to gain a better understanding of nature's fundamental forces and particles. The apparatus for such experiments consists of an accelerator machine and a detector which traces and identifies particles resulting from the collision reaction.

The technology of both accelerators and detectors has developed rapidly during the last decades. By increasing collision energy and improving spatial resolution it has been possible to resolve several levels of the structure of matter. While the demand of higher energies leads to larger accelerator machines with several kilometers circumference, the need of higher accuracy at last causes a huge increment of the amount of raw sensor information, which has to be processed per collision event.

To achieve an optimum utilization of bandwidth to mass storage, a selection of events with interesting reactions has to be performed. The so-called trigger systems take these decisions by the analysis of quickly available sensor information from the detector. While in classical detector architectures there is a spacial separation of readout electronics and sensors from trigger and data acquisition system, parts of the trigger functionality migrate on the detector in experiments of the next generation.

A Large Ion Collider Experiment (ALICE) is one of the four new experiments on the Large Hadron Collider (LHC) which will be built at CERN, Geneva, until 2008. It is the goal to analyze the Quark-Gluon Plasma which will be created by the collision of two lead ions. An indication for an interesting reaction is the observation of electron pairs leaving reaction area with high transverse momentum. To identify those particles the Transition Radiation Detector (TRD) is integrated into the experiment. It makes possible separation of electrons from pion background by transition radiation photons and determination of transverse momentum by calculation of the curvature of the particle tracks.

The readout time of the TRD is about two microseconds. This is fast enough to use it as a primary part of the ALICE trigger system. Therefore particle identification and tracking has to be performed within a time budget of six mi-

croseconds. The processing of the TRD's 1.2 million channels within this time corresponds to a data rate of up to five terabytes per second. That challenge is handled by a hierarchical approach. Piecewise tracking and first estimate of particle identity are realized locally within the front-end electronics. On a global stage, track segments have to be combined to common tracks. Here, final particle classification and transverse momentum determination are accomplished. Thus the data flow from the detector to the Global Tracking Unit (GTU) is reduced to the percent level during the trigger phase.

To satisfy the requirements given above, front-end electronics with specialized components has been developed. The detector chambers are covered by readout boards which are carriers of the major building blocks. These are Multi Chip Modules (MCM) which incorporate one analog and one mixed signal chip. The analog Preamplifier and Shaper Chip (PASA) is connected to the detector chamber output channels and drives the analog inputs of the mixed signal chip. This Tracklet Processing Chip (TRAP) performs analog-to-digital conversion, digital filtering, event buffering and local tracking. Due to the well-defined task and tight boundary conditions like timing, analog noise generation and power consumption, the chip was realized as an Application Specific Integrated Circuit (ASIC).

This thesis describes the digital signal path of the TRAP chip which consists of a filter and a preprocessor preparing data for four on-chip CPUs. It is divided into a structural presentation, the elaboration of its functionality and the description of application aspects.

Chapter 2 gives an overview on the target application, the ALICE experiment. It describes how the TRAP chips are interated in the detector design. The general structure and the mayor components of the chip are presented in chapter 3. With the knowledge of the available devices it is possible to realize the mayor functionality of the chip. The data acquisition and readout is described in chapter 4, while the data processing is separated in digital filtering (chapter 5) and local tracking (chapter 6). The signal processing system has to be adapted to the actual boundary conditions. Therefore the system is tuned by a couple of parameters, their calibration is presented in chapter 7. To guarantee the functionality of the chip, a test environment has been developed, it is described in chapter 8. The power consumption of the chip is analyzed in chapter 9. The final proof of concept of the signal processing is given by the application in a prototype environment (chapter 10). The results of the thesis are discussed in chapter 11.

2

The Target Application

2.1 The Large Hadron Collider (LHC)

The Large Hadron Collider (LHC) is an accelerator ring, which will be built for the European Organization for Nuclear Research (CERN) until the end of 2007 (see fig. 2.1). It will be located in the tunnel of the former Large Electron Positron Collider (LEP) and will be operated in two modes. Proton-proton collisions will take place at energies of 14 TeV while collisions of two lead ions reach energies of 1150 TeV.

Particles are injected into the 27 kilometers long LHC ring from a system of accelerators consisting of the Proton Synchrotron (PS) and the Super Proton Synchrotron (SPS) (see fig. 2.2). They are accelerated in two beams in opposite direction and meet at four interaction points. Around these collision points the four experiments of LHC are built.

The experiments ATLAS (A Toroidal LHC Apparatus) and CMS (Compact Muon Solenoid) are based on proton-proton collisions and are intended to analyze nature of mass, especially to proof existence of Higgs Bosons. Furthermore theoretical models beyond the Standard Model are planned to be checked there. LHCb (LHC Beauty Experiment) is supposed to measure CP violation in b-meson systems. Thus it contributes to the understanding of imbalance of matter and antimatter. ALICE (A Large Ion Collider Experiment, see section 2.2) analyzes the Quark-Gluon Plasma which will be created by the collision of two lead ions.

With LHC a new generation of accelerator machines will be put into operation. Collision energies will be about 30 times higher than those of the Relativistic Heavy Ion Collider (RHIC) at the Brookhaven National Laboratory (BNL). Luminosity, a measure for the rate of events of a specific process, will be more than two times larger. In proton-proton mode it will exceed existing accelerators even by two orders of magnitude. Thereby its constituent energy will be seven times larger than that of the Tevatron at the Fermilab. Table 2.1 shows some parameters of LHC in Lead-Lead mode.

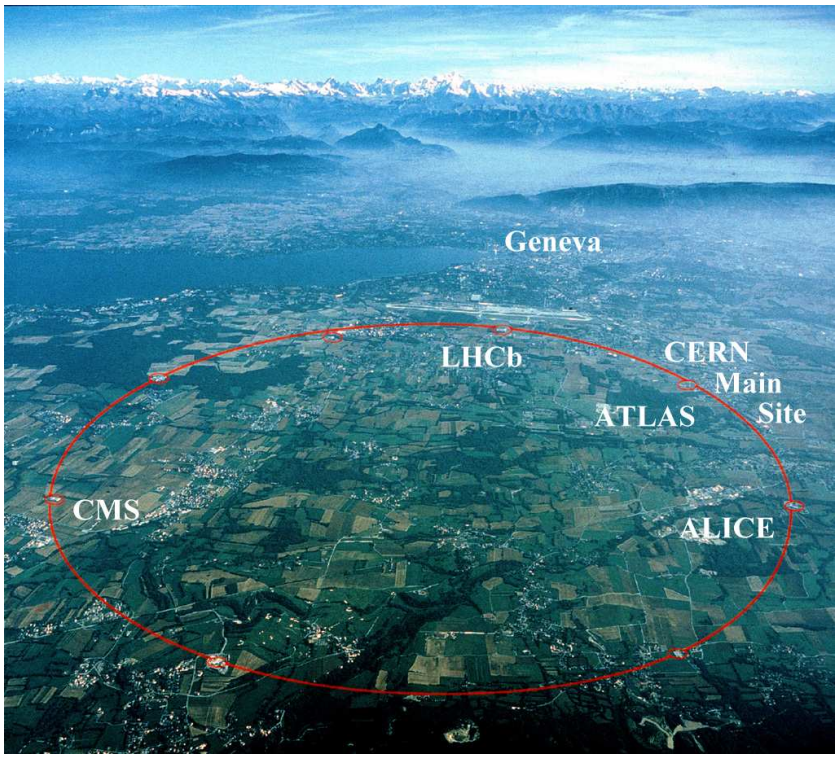

Figure 2.1: overview of the LHC ring (CERN Photo)

Maximum Beam Energy	2.76 TeV/u
Luminosity	$1 \cdot 10^{27}$ cm^{-2} s^{-1}
Time between Collisions	0.1 µs
Bunch Length	7.94 cm
Beam Radius	15.9 µm
Luminosity Lifetime	7.3 h
Filling Time (both beams)	20 min
Acceleration Period	1200 min
Injection Energy	0.1774 TeV/u
Particles per Bunch	$7 \cdot 10^{7}$
Bunches per Ring per Species	592
Circumference	26.659 km
Dipoles in Ring	1232 Main Dipoles
Quadrupoles in Ring	482 2-in-1, 24 1-in-1
Peak Magnetic Field	8.3 T

Table 2.1: some LHC parameters in Lead-Lead mode [pdg0]

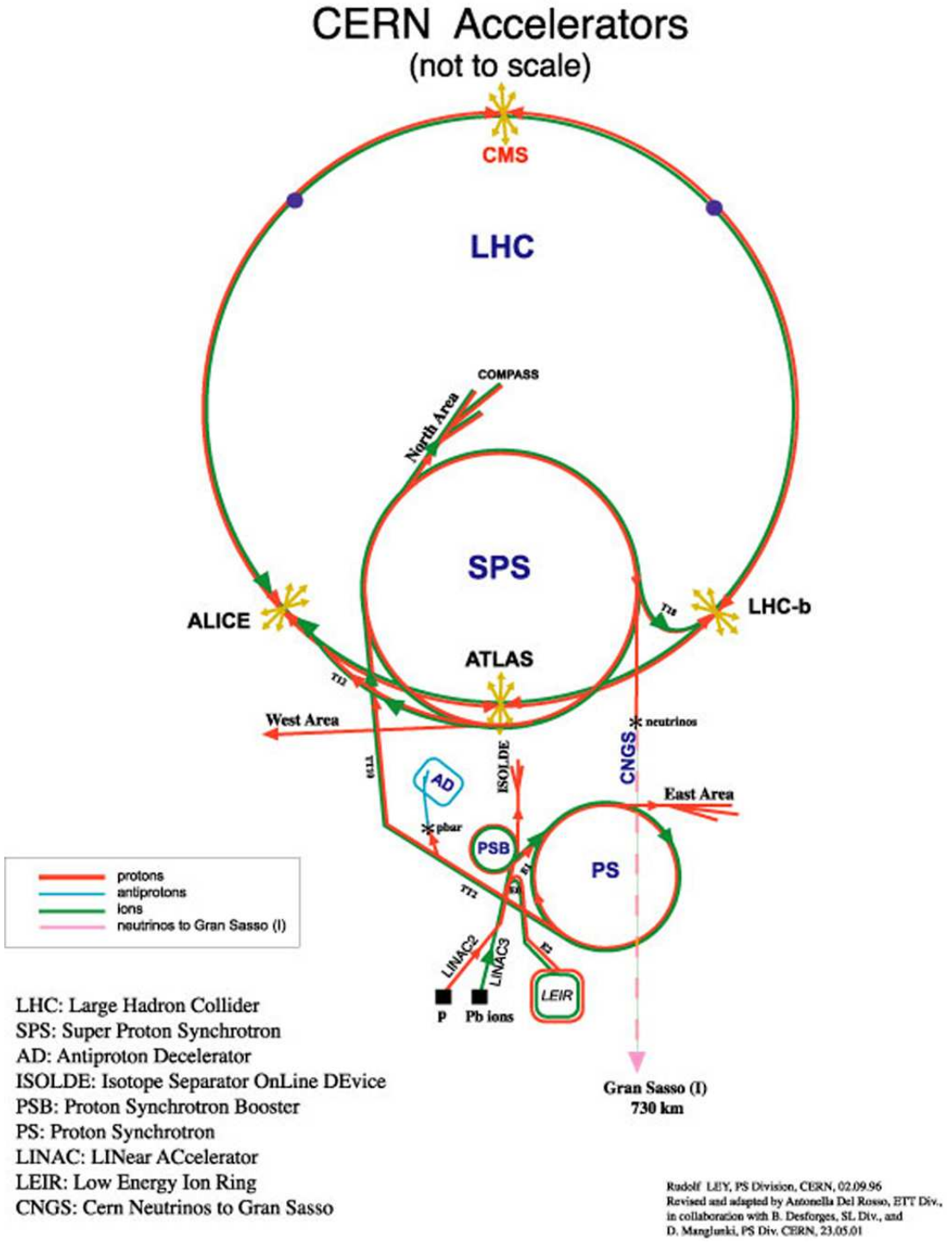

Figure 2.2: the CERN accelerator system (CERN Photo)

2.2 A Large Ion Collider Experiment (ALICE)

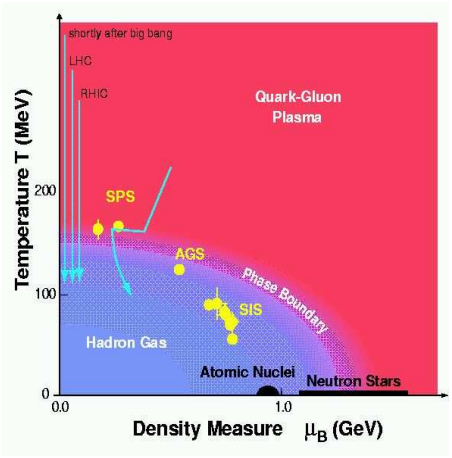

Figure 2.3: phase diagram of the Quark-Gluon Plasma (ALICE Collaboration)

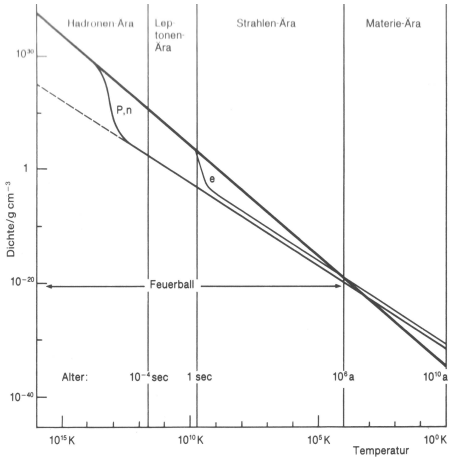

Figure 2.4: development of the universe after the electro-weak transition (from [mhw], p. 523)

In ALICE lead ions collide at a total center of mass energy of 1148.0 TeV. Thereby a Quark-Gluon Plasma is expected to be created. A plasma is a system incorporating a large number of particles which are excited by high temperature or pressure (see fig. 2.3). Thus, particles loose their individuality, the whole system corresponds to a sea of the constituents of the former particles. In case of an electromagnetic plasma ions and electrons move independently, only the system as a whole is neutral with respect to electric charge. A Quark-Gluon Plasma consists of the constituents of the nucleons and their interaction particles, the gluons. In this case strong interaction is dominant and only the system as a whole is neutral with respect to its charge, the color charge or flavor.

The universe passed through this state about 1 µs after the big bang. It is common belief that all particles, anti-particles and interaction particles were in thermodynamic equilibrium at the beginning of the universe. Because of the high energy density all particles could convert into each other and all forces had equal strength. About 10^{-35} seconds after the big bang strong force decoupled from electro-weak force. After this phase transition, almost all quarks were able to convert to quarks only and leptons to leptons respectively. Another 10^{-11} seconds later the temperature of the universe fell below 100 GeV (kT = 1 eV is equivalent to T = 11604 K) and weak force decoupled from electromagnetic force.

This Quark-Gluon Plasma existed until universe had expanded such that temperature had reached 100 MeV. About 10^{-6} seconds after the big bang quarks combined into hadrons. Figure 2.4 shows the further development of the universe.

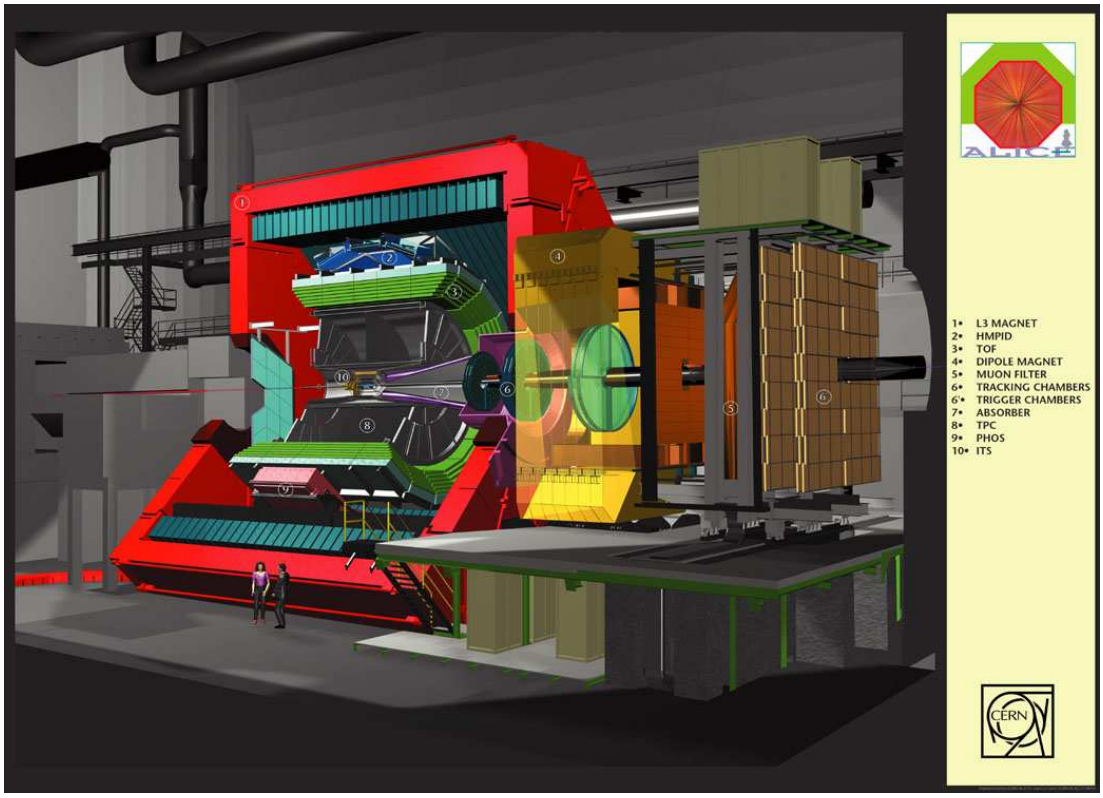

Figure 2.5: the ALICE detectors (CERN Photo)

Due to its expansion, temperature and density decreased continously. Thermal excitation was no longer sufficient for creation of new hadrons (hadron era) and of new leptons (lepton era) afterwards. Down to a temperature of about 3000 K radiation density exceeded density of matter (radiation era). At the current temperature of the universe (cosmic background radiation: 2.7 K) matter dominates radiation (matter era).

The Quark-Gluon Plasma created in ALICE is observed indirectly. Up to 20,000 particles which are leaving the collision area are traced and identified. This allows a reconstruction of reactions within the plasma. ALICE setup incorporates a variety of detectors focusing on different particle properties (see fig. 2.5).

There are three tracking detectors in ALICE. The Inner Tracking System (ITS) is located closest to the beam. It has a length of about one meter and an inner radius of about three centimeters. It consists of multiple layers of silicon pixel, silicon strip and silicon drift detectors. Next, the cylindrical Time Projection Chamber (TPC) covers space from about 57 cm to 278 cm in radial dimension. The within the chambers is ionized locally by charged particles. Due

to an electric field, charges drift to the end caps where they are detected. This takes less than 100 µs on an axial length of the TPC of about 5.1 m. The Transition Radiation Detector (TRD, see section 2.3) is about 295 cm apart from the beam pipe, has a thickness of ≈ 75 cm and an axial length of seven meters. It is structured in six layers. Each of them contains a radiator volume and a drift chamber. Charged particles traversing the gas volume can be detected as well as transition radiation photons, which are generated by them. .

All other detectors provide particle properties only. Next, there is a Time of Flight Detector (TOF). Its 160,000 parallel pad counters measure the time it takes for the particles to move away from the interaction point. The Photon Spectrometer (PHOS) is an electromagnetic calorimeter made of lead-tungsten crystals. The High Momentum Particle Identification Detector (HMPID) operates by measuring Cherenkov radiation rings.

All detectors above are part of the central area which is enclosed by a magnet generating a field of about 0.5 Tesla. It allows the determination of particle momentum by calculation of its track curvature. In addition there are a couple of forward detectors and a Muon Spectrometer (Forward Multiplicity Detector, FMD; Photon Multiplicity Detector, PMD; Centauro and Strange Object Research, CASTOR; Zero Degree Calorimeter, ZDC).

2.3 The Transition Radiation Detector (TRD)

The TRD is structured in a cylindric geometry (see fig. 2.6). In the angular direction (ϕ) it is separated into 18 super-modules. Each of them contains five stacks which are arranged in z direction. The TRD stacks are located at a radius position between 2.9 and 3.7 meters. Each of them incorporates six detector modules. These modules consist of detector chambers and front-end electronics. The chambers can be divided into radiator, drift region and amplification region which are about 4.5 cm, 3.0 cm and 0.7 cm thick, respectively. Front-end electronics is mounted on top of these modules.

The radiator consists of polypropylene fiber mats, embedded in Rohacell foam sheets. Both components are extraordinary inhomogeneous in terms of optical density. If a charged particle crosses the transition of media of different optical density, the so-called transition radiation will be emitted. It is concentrated in a forward cone of $\theta \leq \gamma^{-1}$ where $\gamma = \frac{1}{\sqrt{1-\beta^2}}$ is the relativistic factor ($\beta = \frac{v}{c}$). Thus transition radiation is a direct measure of particle velocity. At given momentum it can be used to determine the particle mass. In case of the ALICE TRD this radiation is in the range of soft X-rays. It is used to discriminate pions from electrons. Because pions are up to ≈ 273 times heavier,their production of transition radiation can be neglected.

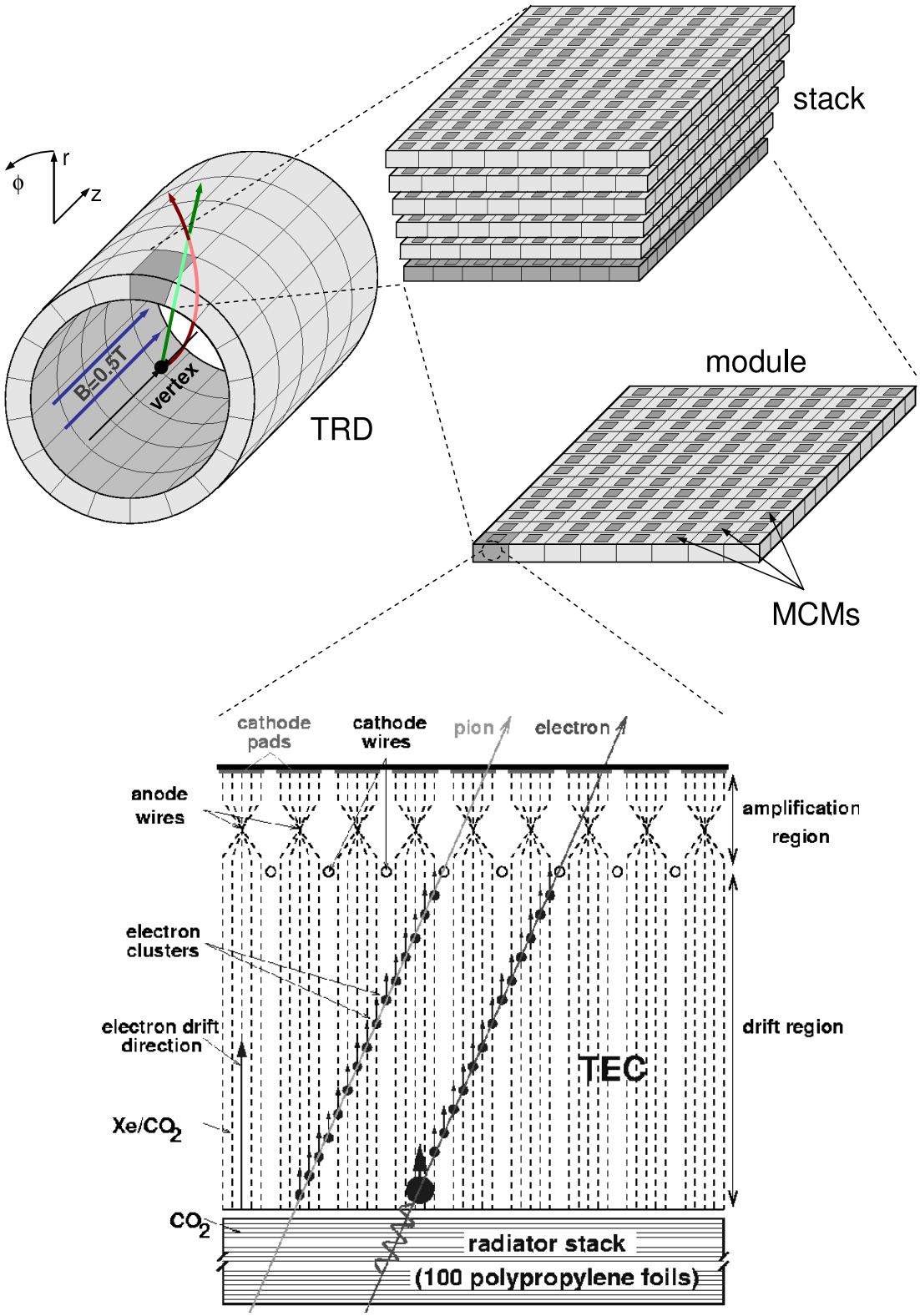

Figure 2.6: structure of the ALICE TRD

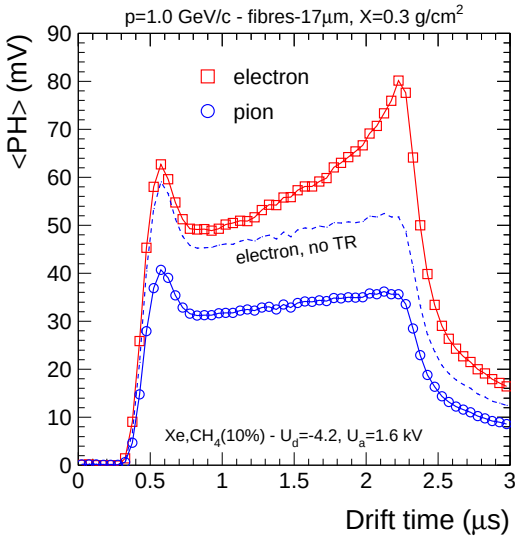

Figure 2.7: mean ionization of one detector chamber along a particle track

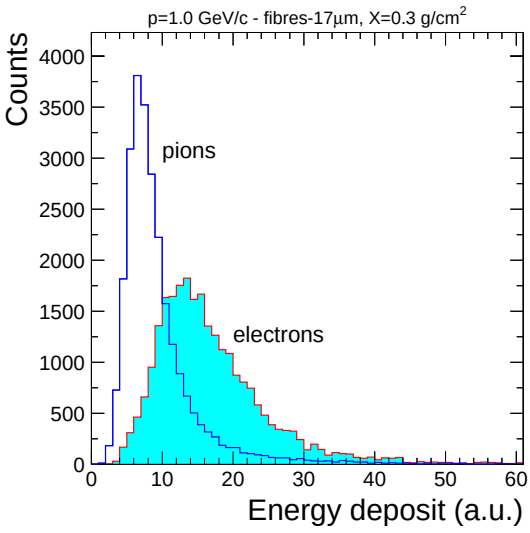

Figure 2.8: energy deposit of one particle track at a momentum of 1 GeV/c

The drift region is filled with a mixture of xenon and carbon dioxide. As both electrons and pions are charged particles, they induce ionization clusters along the track within the gas. Figure 2.7 shows the mean charge along the track. On average, ionization caused by electrons is larger than that of pions. Furthermore the electron's transition radiation photons convert into additional charge clusters of relatively high amplitudes. The exponential probability distribution of this conversion can also be seen here. The first peak of the signal is due to the transition of the particles through the amplification region. The overall ionization caused by one particle is shown in figure 2.8.

Ionization within the gas volume is detected by net charges which are induced on the cathode pads next to the amplification region (see fig. 2.6). Due to an electric field gas electrons within the drift region move into the amplification region within 2 µs. There they initiate cascades of ionization which are caused by the cylindrical electric field around the anode wire. While the electrons are caught by the anode wire, the ions slowly drift out of the amplification region.

2.4 The ALICE Trigger System

The event rate is about 10^4 collisions per second in lead-lead mode. Only about 100 events per second are considered to contain interesting physics. They are characterized by a couple of criteria like small impact parameters. To optimize

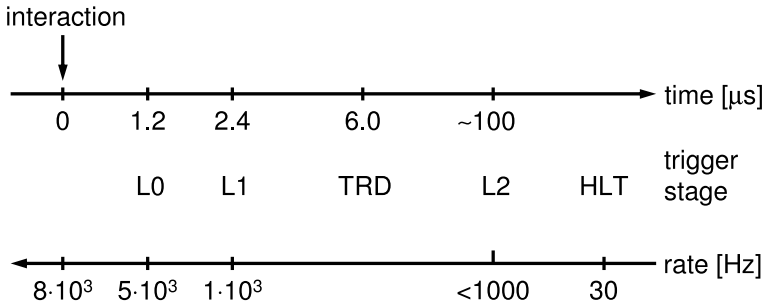

Figure 2.9: latencies and rates of the several trigger levels

the usage of detector data bandwidth a selection of events has to be performed by the so-called trigger system [vl0].

The ALICE trigger system is separated into several levels. They differ by the amount of data on which the decision is based and by the complexity of the data analysis. Trigger levels of longer latency process data of detectors with longer readout time (see fig. 2.9).

The Level-0 Trigger (L0) uses the data of the Forward Multiplicity Detector (FMD). It will accept an event after 1.2 µs, if its multiplicity is above some threshold, the interaction point is close to the nominal position and the forward-backward distribution is that of a beam-beam interaction.

In parallel to the generation of the Level-0 Trigger, the TRD Trigger searches for electron positron pairs with high transverse momenta (see section 2.5). For this purpose, the time budget is 6.5 µs.

The Level-1 Trigger (L1) uses information of the Zero Degree Calorimeter (ZDC) and of the Muon Spectrometer to perform further analysis of centrality and multiplicity of an event. Furthermore, it is looking for muon pairs with high momenta. This decision is taken 6.5 µs after the event. The acceptance of an event by this trigger initiates readout of the TPC.

The Level-2 Trigger (L2) uses TPC readout time for a more detailed analysis like cuts on the mass of muon pairs, search for J/Ψ particles or inspection of clusters of the FMD. In addition, data from more detectors like the Photon Spectrometer is available.

Finally, the High-Level Trigger (HLT) processes the tracking information of the TRD Trigger in combination with the TPC data. Is performs more sophisticated particle tracking for particle identification and momentum determination. The input event rate of this trigger level is about 200 Hz, while the event rate of the succeeding data acquisition system is about 30 Hz.

2.5 The TRD Trigger

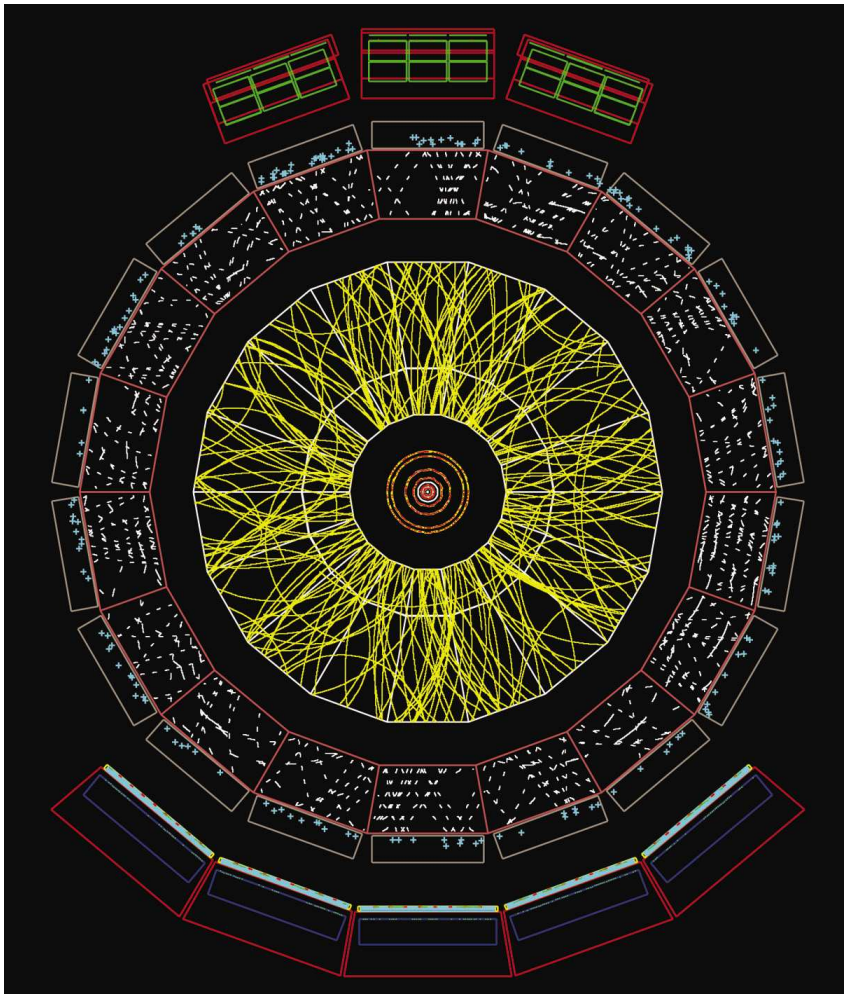

Figure 2.10: display of a simulated ALICE event with cut in $60° \leq \theta \leq 62°$ (CERN Photo)

It is the main task of the TRD trigger to select events with a high probability of the production of so-called heavy vector mesons like J/Ψ or Υ. They can be detected via their decay channel into an electron-positron pair. Due to the energy released by the decay they have a high total momentum and a high transverse momentum, consequently. Accordingly, the TRD Trigger is supposed to search for electron-positron pairs with transverse momenta of typically more than 3 GeV/c. The TRD provides the data for the described trigger decision, because it allows to sufficiently estimate transverse momenta by particle tracking, is capable of identifying electrons and its readout time is short enough due to the multiple flat chamber design.

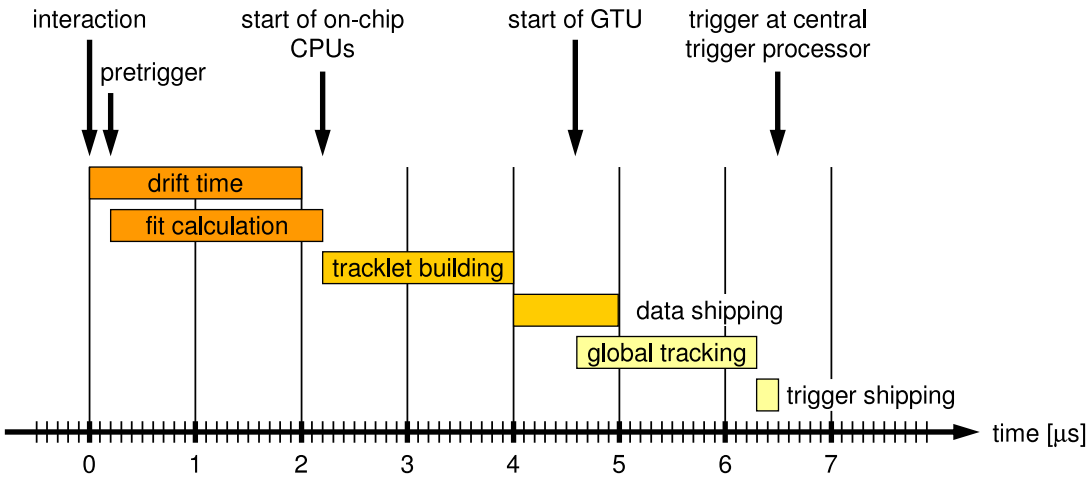

Figure 2.11: timing of the TRD trigger

Electron identification is performed by the separation from pions which are the dominant particles (see section 2.3). Particle tracks have to be reconstructed through the six layers of the TRD. Since there are 20 samples per event of ten bit each within about 1.2 million analog channels this leads to 29 MB of raw data. This has to be processed within a time budget of 6.5 µs. To handle such an amount of data a hierarchical design has been implemented. Because the segments of the tracks (the so-called *tracklets*) in several layers can be detected within a few adjacent channels, their parameterization is performed locally and in parallel. As the bandwidth of the detector is limited by cabling, which is determined by geometrical boundary conditions, it is inefficient to ship the raw data for any processing out of the detector in the given time. Hence, this computation has to be done by the front-end electronics on the detector. To meet these functional requirements as well as tight power and electric noise restrictions, an Application Specific Integrated Circuit (ASIC) has been developed, the Tracklet Processing Chip (TRAP, see chapter 3). On a global stage, only information of detected tracklets is necessary to combine them to common tracks and thus to estimate their transverse momentum (Global Tracking Unit, GTU, see section 2.6). The timing of the TRD trigger is shown in figure 2.11. Detector data is acquired during drift time, which is 2 µs long and starts immediately after the interaction. Meanwhile, data is buffered, digitally filtered and furthermore preprocessed to parameterize the tracklet content. Thereafter, tracking information for the GTU is built locally by four CPUs on each of the TRAP chips. About 4 µs after the interaction TRAP chips are starting to send tracklet data to the GTU. The GTU is starting in parallel and has to make a trigger decision until 6.5 µs after the interaction. About 0.1 µs later the trigger decision will arrive at the Central Trigger Processor (CTP).

2.6 The Global Tracking Unit (GTU)

The Global Tracking Unit (GTU) matches tracklets to tracks within the full TRD. This makes it possible to calculate certain track parameters, especially transverse momentum and particle identity.

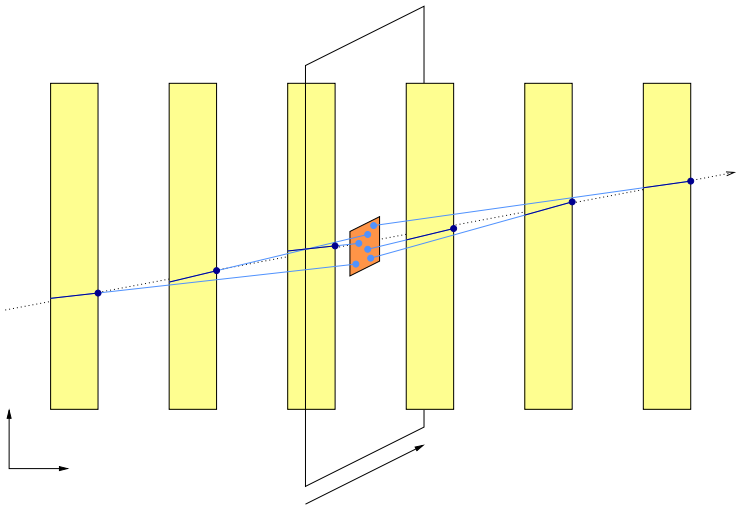

Figure 2.12: tracklets matching within the GTU [jc0, p. 29]

Because the detector stacks are arranged in a nearly projective geometry, especially the stiff tracks are located in one of them exclusively. Hence, the problem of global track reconstruction can be reduced to matching in each of the detector stacks. This is performed by 90 Track Matching Units (TMU), corresponding to the number of detector stacks. Each of them receives data from one stack by 12 optical lines. They provide the data which is collected by readout networks in the detector modules.

Inside the GTU, all tracklets of a detector stack are projected to a central plane. If a minimum of four tracklets out of six detector chambers with similar slope can be found in a local neighborhood within this plane, these tracklets will be combined to a global track (see fig. 2.12). This criterion is implemented as a moving window algorithm.

Subsequently, the track parameters can be estimated by using the position and slope information of the incorporated tracklets. A final decision on particle identity is performed by combination of the estimates which were calculated locally and are contained in the tracklet data as well. Structured like this, only little information has to be processed by a common trigger unit within the GTU. It receives all relevant results of the TMUs. At present, a prototype of the GTU is developed, its design is proposed and analyzed in [jc0].

2.7 ALICE TRD Front-end Electronics

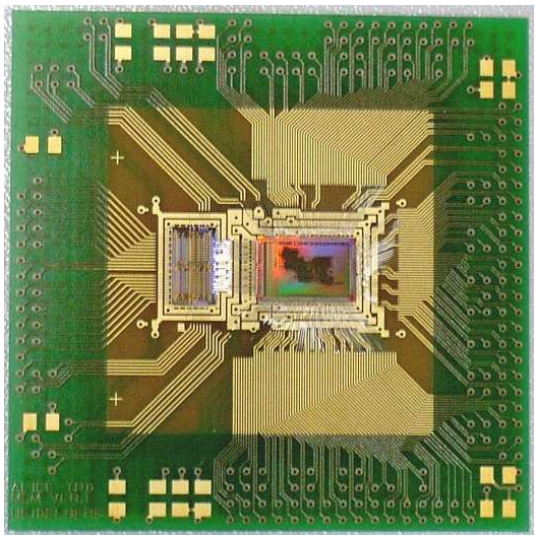

Figure 2.13: one Multi Chip Module (MCM) incorporating one PASA on the left hand side and one TRAP on the right hand side

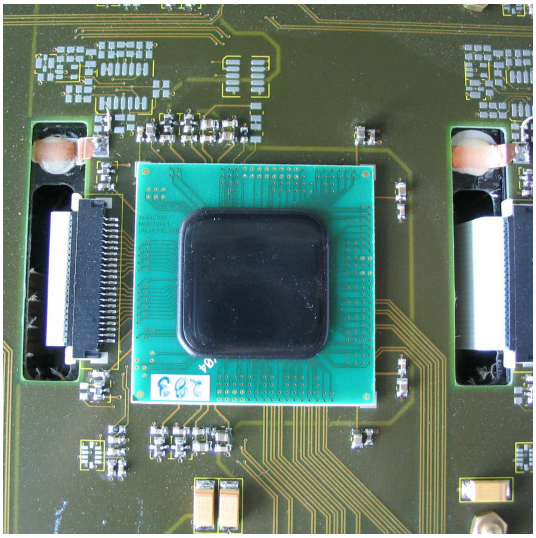

Figure 2.14: one MCM coated with glob top, integrated on a readout board and connected to a detector chamber via a ribbon cable on the left

There are two main tasks, front-end electronics has to accomplish. First, it has to acquire and to buffer detector data. In addition, it has to perform local online tracking. All the functionality is implemented on two chips, which are combined to a Multi Chip Module (MCM, fig. 2.13). Each of them is connected to 18 cathode pads of the detector (fig. 2.14).

On the MCMs, pads inputs are connected to the Preamplifier and Shaper Chip (PASA). It amplifies the charge fluctuations on its input ports by a factor of 12 mV/fC. Furthermore, it removes long signal components by a first order pole/zero circuit. Finally the signals are shaped by a fourth order filter. This leads to a Gaussian-like point spread function with a peaking time of 70 ns and a FWHM of 120 ns. To prevent artifacts in data processing due to tracklets crossing pads, which are associated to two adjacent TRAPs, three output channels are distributed to the TRAP chips of neighboring MCMs. At last PASA drives 18+3 differential voltage output ports.

The TRAP chip receives signals by 21 differential voltage input ports. Being a mixed mode chip it contains the Analog-to-Digital Converters (ADCs) as well as digital circuits for event buffering and processing (see section 3.1).

The MCMs are integrated on the so-called readout boards (fig. 2.15). Six up to eight of them are used to cover one detector chamber. They provide the

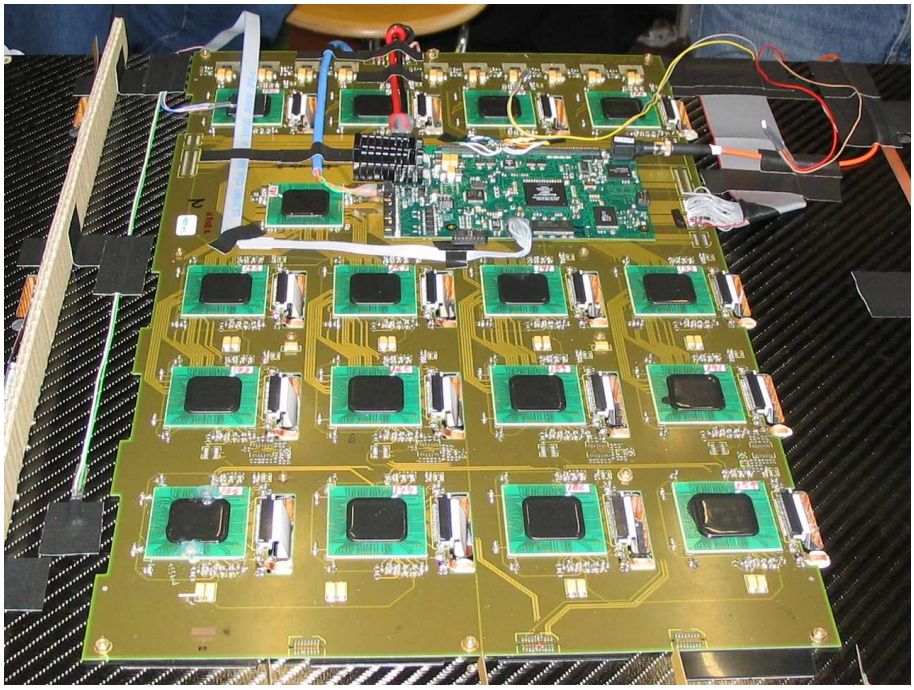

Figure 2.15: one readout board mounted on a detector chamber carrying 17 MCMs and one DCS board

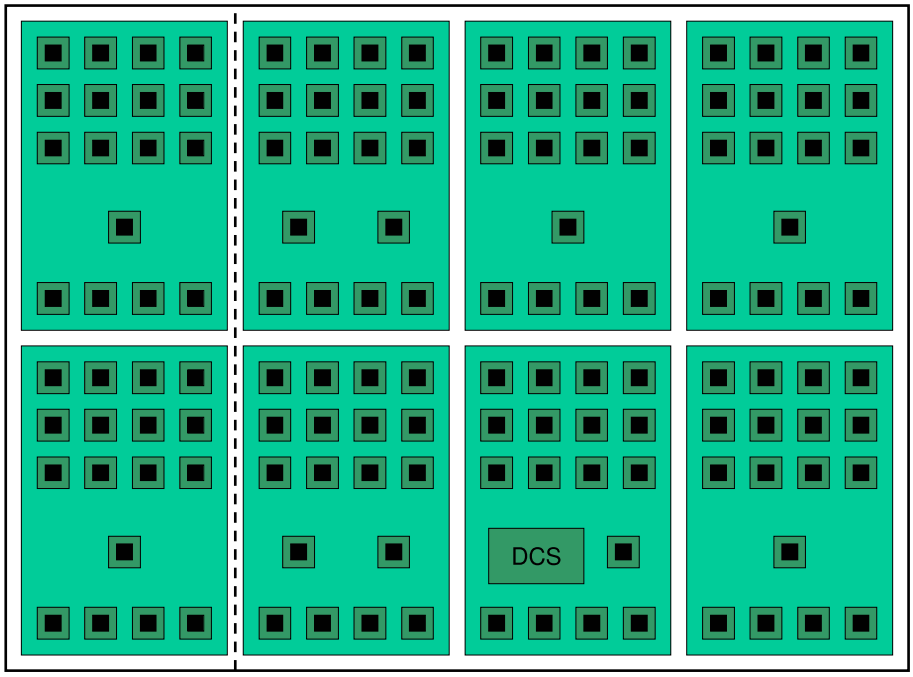

Figure 2.16: coverage of the detector chambers by various readout boards. There are half chamber mergers on two of the boards and one DCs board.

configuration network of the chips (Slow Control Serial Network, SCSN, see [rg0]), the fast readout network for tracklet and raw data transmission (see [rs0]), power supply and the clock and pre-trigger distribution.

Each readout board incorporates 16 MCMs which are connected to the chamber to acquire data. In addition there is one MCM which is used for merging of the boards readout stream only (Board Merger). On two of the readout boards another MCM is mounted to merge the data of one half of the chamber (Half Chamber Merger). Here, the final data stream is built. It is sent via an optical transmitter, which is mounted on a mezzanine board, which is next to each of the Half Chamber Mergers.

One readout board per chamber carries a board of the Detector Control System (DCS). It controls the power of a chamber, checks runtime parameters like supply voltage, humidity or temperature, distributes clock and pre-trigger signals and provides the configuration interface for all TRAP chips of one chamber (SCSN). A detailed discussion of readout board components, structure and routing is given in [ir0].

3

The Tracklet Processing Chip (TRAP)

3.1 General Structure

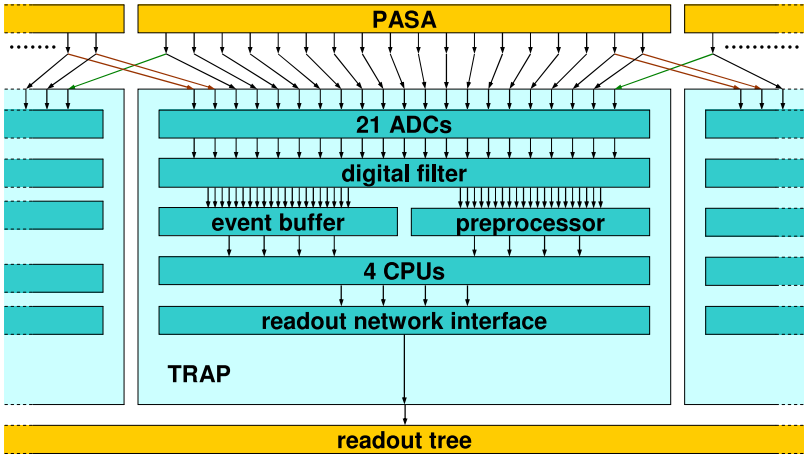

Figure 3.1: main signal path of the TRAP chip

The Tracklet Processing Chip (TRAP) is the core component of the ALICE TRD front-end electronics. It is a mixed mode chip receiving analog signals from the PASA chip (see section 2.7) and providing digital signal processing capabilities. Figure 3.1 shows the main signal path of the TRAP chip.

First, analog signals are converted to digital signals (see section 3.5). Then, they are filtered and stored in the event buffer (see section 3.7). There they are accessible for the on-chip CPUs (see section 3.9) to be read, compressed and sent to the readout network interface. That device collects data of all CPUs and feeds it into the readout network which is organized in a tree structure (see section 3.10).

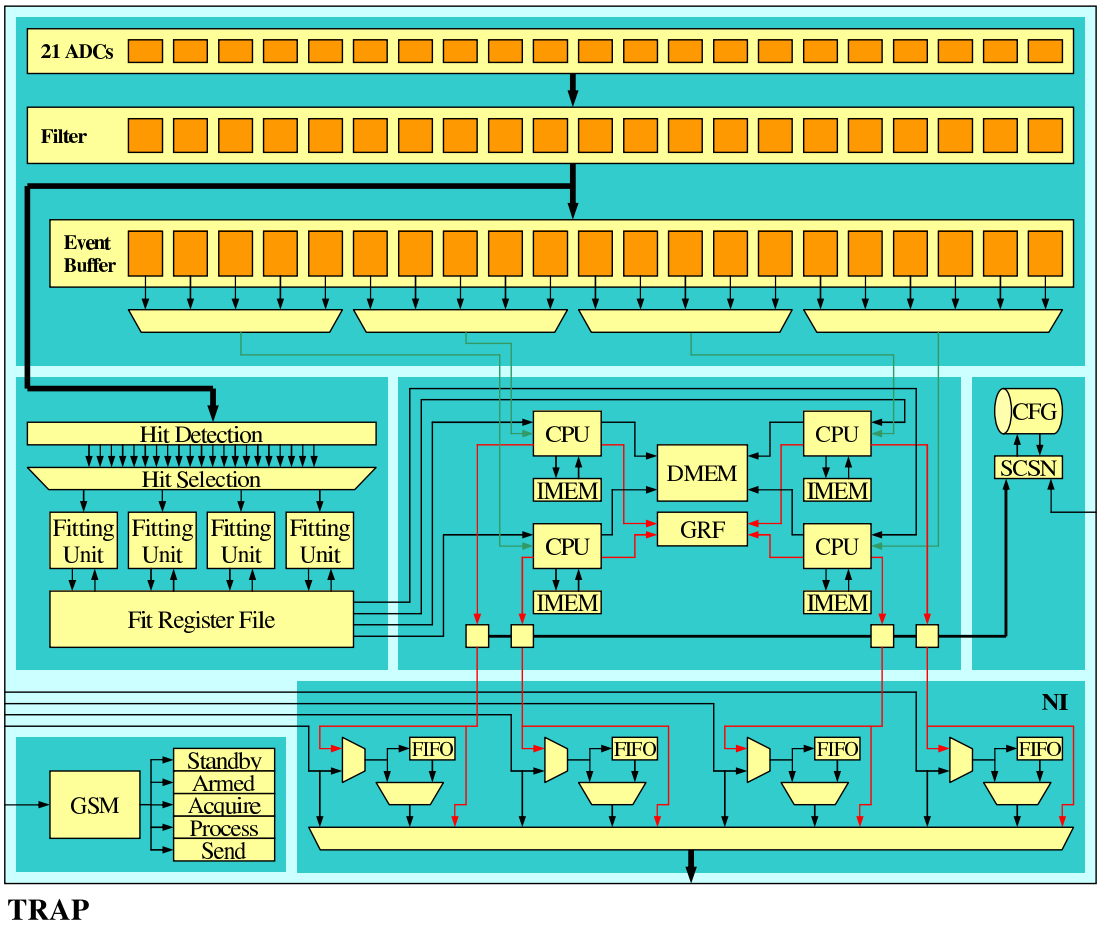

Figure 3.2: structure of the TRAP chip

The signal processing devices of the chip are specialized for local track finding (see chapter 6). It is the first step to detect and fit sub-tracklets within each of the covered signal channels. This is performed by a preprocessor which is working on the basis of the filtered signals (see section 3.8). The CPUs are started to finally process the fit information of the preprocessor. Tracklets with signal in more than one channel are combined and the tracking information for the GTU is built. It is transmitted by the same network as the raw data.

Figure 3.2 shows the detailed architecture of the TRAP chip. It includes the inner structure of the major components, which are discussed in the following sections. There are two additional components of the chip which are not a part of the actual data path. The global state machine controls all actions on the chip and especially synchronizes it to external trigger commands (see section 3.3). The configuration interface is discussed in section 3.4.

3.2 Chip Development and Design Flow

The TRAP chip is produced in the UMC 0.18 µm mixed mode process. A first chip incorporating prototypes of all major components was designed and submitted in May 2002 (TRAP1). A chip prototype providing full functionality has been developed until May 2003 (TRAP2). The final chip was submitted in March 2004 (TRAP3) and is available for testing since July 2004. This chip is used for integration tests, beam measurements at CERN in fall 2004 and continuous measurements of cosmic radiation at the Kirchhoff-Institute of Physics (KIP) at the University of Heidelberg (see chapter 10).

The analog components of the TRAP chip have been developed at the Chair of Microelectronics, Prof. Dr. Tielert, University of Kaiserslautern. The digital components are designed in the VHDL hardware description language and are mapped on a standard cell library provided by Virtual Silicon Technology Inc. The design synthesis has been performed by Synopsis Design Analyzer. All major building blocks like filter channels, preprocessor, network interface, CPUs, etc. were synthesized separately and merged only on higher design levels. The result of the synthesis was exported in a Verilog net-list file.

On the basis of the result of the synthesis and geometrical information of the analog components and standard cells (Library Exchange Format, LEF) the layout of the chip can be designed. For this purpose Cadence First Encounter has been used. It provides all necessary design steps such as floor planning, placement, power planning, clock tree routing and optimization, signal routing and optimization and finally the timing analysis. The latter can be used for back annotation of simulation timing models. Thus, the target speed of 120 MHz for the CPUs, and the readout network interface as well as 10 MHz for digital filter, preprocessor and event buffer was verified. The layout is exported to another intermediate file (Gridded Data Set Format 2, GDS2).

Finally, the chip layout and the layout of the analog components are combined using Calibre by Mentor Graphics Corp. A last consistency check is included by comparing chip layout and schematic net-list (Layout versus Schematic Check, LVS). To ensure producibility the design has to satisfy various process dependent design rules like minimum distance of adjacent signal traces or minimum width of design structures. Thus a Design Rule Check (DRC) has to be performed. Figure 3.3 shows the final layout of the TRAP chip. It is bordered by various input/output pads providing signal connectivity and power supply. Some easily recognizable structures are labeled next to the figure.

A priori, functionality has been verified by simulation on functional level and on gate level using ModelSim by Mentor Graphics Corp. On gate level, timing of each used standard cell and timing of signal propagation between those cells (back annotation) is included into the model.

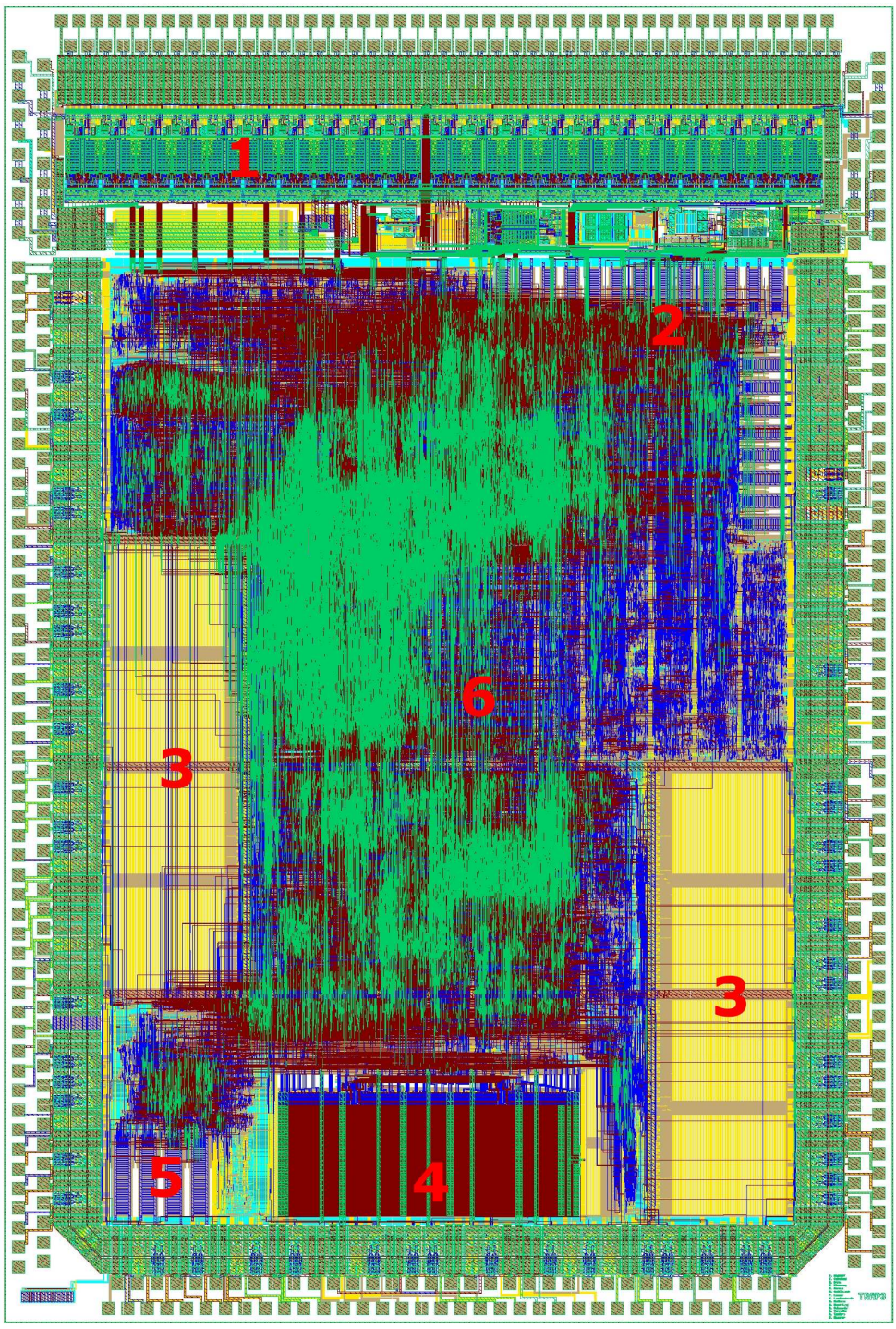

Figure 3.3: layout of the TRAP chip: analog digital converters 1, event buffer 2, instruction memory 3, data memory 4, readout network interface 5, standard cell area ('sea of gates') 6

3.3 Global State Machine

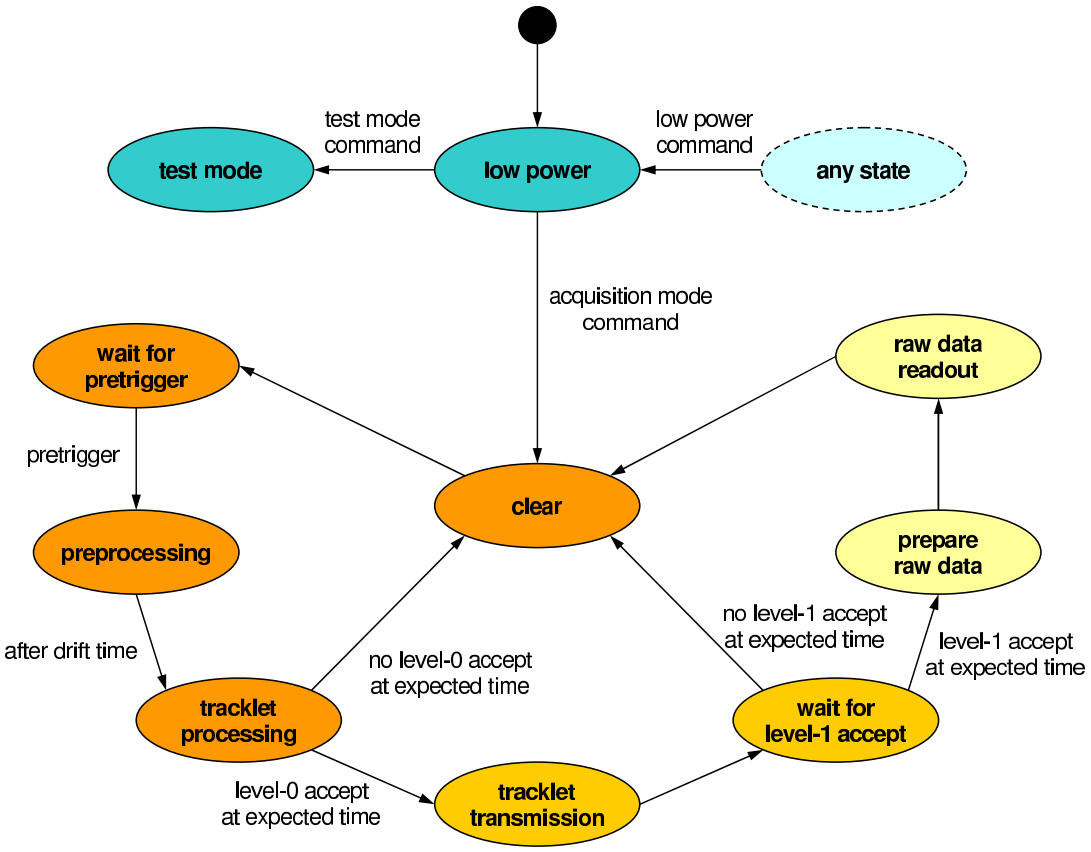

Figure 3.4: state diagram of the global state machine

All functionality of the chip is controlled by a central device, the Global State Machine (GSM). The transitions between its ten states can be initiated by external trigger signals, by internal timers or by commands in the command register which is accessible for configuration and the CPUs via a global input/output bus (see section 3.4). The resulting state diagram is shown in figure 3.4.

After a reset the chip starts up in low power state. All but the configuration interface is disabled and not clocked. The chip can always return to this state by a low power command. It can be left for test mode or clear state by the test mode or acquisition mode command. Leaving the low power state the analog digital converters are enabled and the digital filter is clocked.

The clear state is the entry point to typical operation. The preprocessor is reset and the CPUs are started and clocked to prepare for potential tracklet processing later on. This state is left after the CPUs have finished their clearing procedure. The GSM stays in the following state until a pre-trigger signal occurs.

Subsequently, the GSM timers are started and the chip changes to preprocessing state. The event data is acquired and the fit parameters of the preprocessor are calculated. The preprocessor is clocked only in this state. After the drift time of the detector chamber, the acquisition control machine signalizes to the GSM to have finished. This causes the GSM to change to tracklet processing state.

In this state the CPUs are clocked to process the tracklet candidates presented by the preprocessor. This is completed by building of tracklet words which are sent to the network interface. The external level-0 accept signal is expected at a given time after the pre-trigger, which is checked by one of the timers, which has been started at the transition to preprocessing state. The present acquisition sequence will be ended by changing to clear state if no level-0 accept has occurred. If so, the GSM will change to tracklet transmission state during which the network interface transmits the tracklet information. Leaving the tracklet processing state the CPUs are switched off.

The state after tracklet transmission is a wait state for level-1 accept. The accept signal is expected at a time, which is defined by another timer, which has also been started at the transition to preprocessing state. If the signal does not occur, the acquisition sequence is ended by the transition to clear state. If it does, the GSM changes to the raw data preparation state. The CPUs are enabled again to read data from the event buffer and prepare data for transmission. After changing to the raw data readout state, the CPUs fill all data, which is to be sent, into the network interface. It handles the handshaking and communication with the network. After completion the chip returns to clear state and is prepared for another acquisition sequence.

3.4 Configuration

To put a TRAP chip into operation, internal parameters have to be set, the CPUs' data and instruction memory have to be initialized, some commands have to be stated via control registers and the data bank memory has to be filled eventually. All those resources are accessible by a chip-global input/output bus (see fig. 3.5). There is one bus controller which drives the write enable signal, the 16 bits wide address bus and the 32 bits wide input data bus. The addressed device is multiplexed to the output data bus.

Both write and read statements on the global input/output bus can be initiated by any of the four on-chip CPUs or by commands received via the external configuration network. To handle concurrent bus accesses a priority driven arbitration is performed by the bus controller [tm, section 8.1].

The configuration network (Slow Control Serial Network, SCSN) is organized in two serial rings of opposite direction. If a TRAP chip within the SCSN network fails, the neighboring chips can close one remaining ring by the assignment of the input of one ring to the output of the other one and vice versa [rg0]. The SCSN

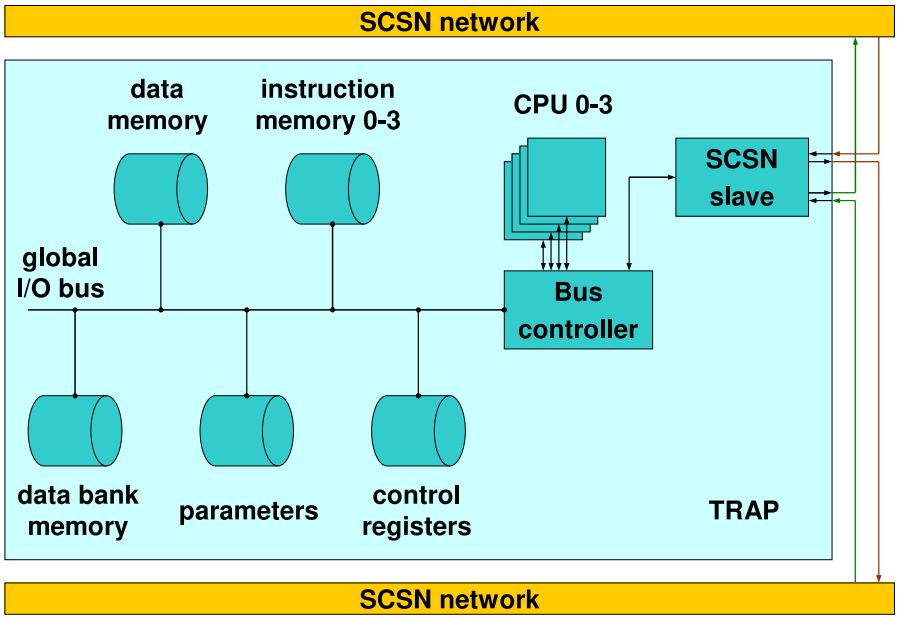

Figure 3.5: configuration path of the TRAP chip via SCSN network and global input/output bus

communication is organized in packets, which are generated by the SCSN master of the corresponding ring, and modified or passed over by each of the SCSN slaves in the ring. Thus, a configuration of a TRAP chip corresponds to a series of write packets, which are received by its SCSN slave. It translates them to accesses on the global input/output bus.

The global input/output bus operates with parallel busses and a strobe signal with a 120 MHz timing. The asynchronous and serial SCSN network incorporates one signal only. The packet has a net width of 86 bits including a 16-bit check sum to ensure data integrity. It is enlarged by a variable number of padding bits to guarantee a minimum frequency of signal changes for synchronization.

3.5 Analog Digital Converters

The Analog Digital Converter (ADC) used in the TRAP chip has been developed for the special purposes of the target experiment [dm0]. It performs a cyclic successive approximation of its differential analog input signal. Some typical parameters are give in table 3.1. As the internal reference voltage can be adjusted by about 40 % the conversion gain and dynamic range are variable by the same order of magnitude. The phase of the sampling time of the ADC relative to the internal 10 MHz clock is adjustable in order to minimize disturbances of the analog measurements due to digital activity.

Conversion Gain	2.0-2.8 mV/ADC Count
Dynamic Range	2.0-2.8 V
Sampling Rate	10.0 MHz
Number of Bits	10
Effective Number of Bits	9.3
Differential Nonlinearity	<0.5 ADC Counts
Integral Nonlinearity	<0.5 ADC Counts
Power Consumption/Channel	12,0 mW

Table 3.1: operating parameters of the TRAP ADCs

One of the major demands on the ADCs was to combine high quality signal sampling with extraordinary low power consumption. The ADC is designed such that the power consumption can be adjusted downward but this leads to a corruption of faster signals since the dynamics of the circuits is reduced. The ADC is instantiated 21 times for all analog inputs of the TRAP chip plus another one for internal run time parameter checks like temperature and voltages [tm, section 5.2].

3.6 Digital Filter

The digital filter of the TRAP chip receives data of 21 ADCs and distributes it to the event buffer and the preprocessor. For test purposes it is possible to change filter input to the event buffer (see fig. 3.8). The filter is structured channel-wise because the signals are processed almost independently. Only Data of adjacent channels has to be exchanged (see fig. 3.6).

The internal structure of each of the filter channels corresponds to its functionality and is discussed in chapter 5. The filter is operated at the sampling frequency of the ADCs (10 MHz).

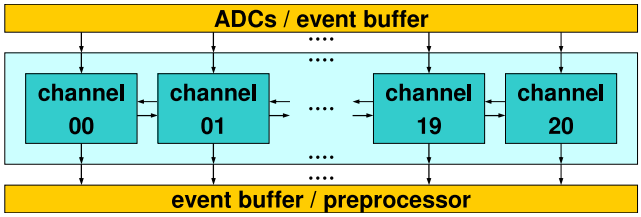

Figure 3.6: overall structure of the digital filter

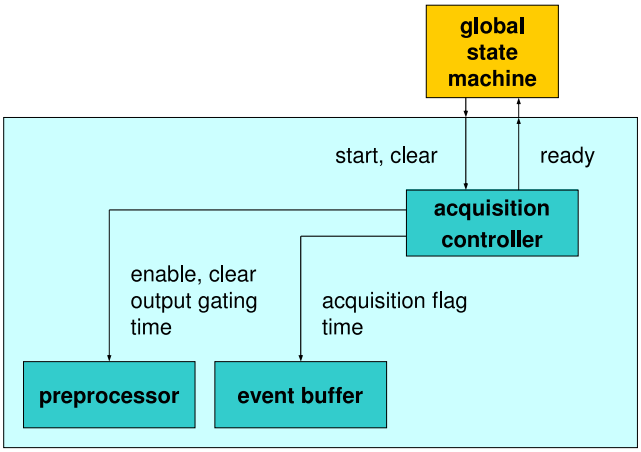

Figure 3.7: acquisition control of event buffer and preprocessor

3.7 Event Buffer

The event buffer stores the data of the events. It is steered by a common control module for event acquisition and event preprocessing (see fig. 3.7). It is cleared and started by the GSM and provides a time reference relative to the pre-trigger signal. After a programmable time it indicates readiness to the GSM which proceeds to its next state subsequently (see section 3.3).

The event buffer is organized in 21 independent memory blocks of 64×11 bits each. The memory blocks offer two independent ports for reading and for writing. The write port is operated at the ADC's sampling frequency of 10 MHz. The write address is given by the acquisition controller's time tag plus a given offset in acquisition mode. In the other states, it is connected to the global input/output bus. The write data is determined by the selection logic shown in fig. 3.8. If the acquisition controller indicates no acquisition the data can be taken from the configuration interface. For acquisition phase it can be selected whether filtered data or data from the input delay chain is stored. The delay can be set to one up to six samples. The input delay chain is filled by the ADCs or by data taken from the event buffer itself (see below).

The read port is operated at the CPUs' operation frequency of 120 MHz. Typically the read address is set by the CPUs while they are accessing the event buffer via their local input/output bus. Five event buffer channels are assigned to CPU 0 to 2 while CPU 3 is assigned to the uppermost six channels. For test purposes the chip can be set in simulation mode to provide data out of the event buffer as input for the input delay chain (fig. 3.8). Therefore the read address is given by the acquisition controller's time tag plus a given offset. The read data bus is connected to the input multiplexer of the data delay chain and to the data multiplexer of the CPUs' local input/output bus.

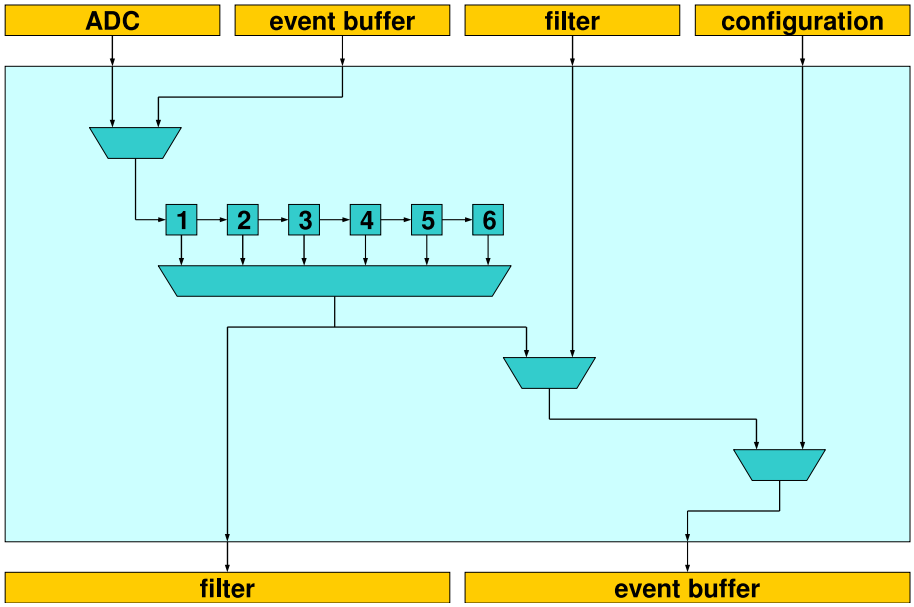

Figure 3.8: data path to filter and event buffer

In order to handle write access to event buffer memory an interface between those two clock domains is provided. Thus, address and data of an event buffer write access are buffered and a synchronization logic controls the data flow from the 120 MHz strobe domain into the 10 MHz clock domain. This reduces data bandwidth down to half a word per 10 MHz clock cycle. As the event buffer access during configuration is much slower and write access of the CPUs are for test purposes only the core functionality of the chip is not affected.

3.8 Preprocessor

The preprocessor parameterizes track segments in 19 channels by processing triplets of the 21 filter channels. It is steered by and gets timing information for segment fitting from the acquisition controller (see fig. 3.9).

The concept of track segment finding and fitting as well as the corresponding hardware design are discussed in chapter 6. Up to four track segments can be processed in parallel. The fit parameters are stored for each channel. Therefore a memory instance has been implemented, which provides four independent read-modify-write ports with a size of 19×111 bits, the Fit Register File (FRF). The memory entries are cleared by the acquisition controller during the GSM's clear state (see section 3.3). They are enabled for writing during a given period of time after the pre-trigger while the GSM is in preprocessing state and the acquisition controller takes care of the precise timing. The time information is

also distributed to the segment finding and fitting unit.

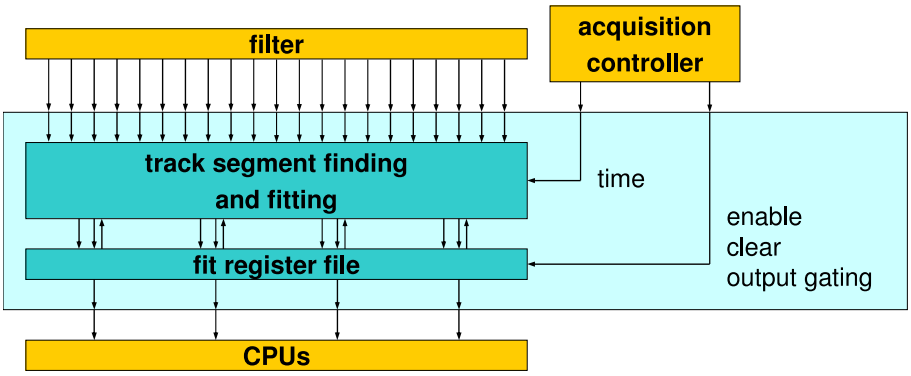

Figure 3.9: preprocessor control and data flow

For further processing by the CPUs up to four tracklet candidates are selected. A tracklet candidate is a pair of adjacent channels with a given minimum number of hits in both channels and another minimum number of hits in its lower channel. It is indicated by its lower channel number. If there are more than four potential candidates, those with the maximum number of hits in both channels are selected.

The sorting is performed in parallel to the fitting arithmetics. The corresponding hardware design is optimized for cell area and not for timing because the cell area increases rapidly with respect to tighter timing constraints and timing is not critical here. After a given time the sorting logic provides a stable result which is signaled by the acquisition controller's output gating flag.

After the fitting period, the addressing of the FRF is switched from the segment finding and fitting unit to the sorting unit in order to distribute the tracklet candidate parameters to the CPUs. Thus, fit parameters of the lower channels of the tracklet candidates are read from the FRF. They are stored in four latches which are gated by the acquisition controller's output gating flag. Subsequently, the address to the FRF is incremented. So the parameters of the upper channels of the tracklet candidates are available in parallel to those stored in the latches. Each CPU incorporates a multiplexer to select the parameters out of its tracklet candidate. For test purposes the sorting unit can be bypassed and an arbitrary address can be defined to read the FRF entry.

3.9 CPUs

The four TRAP CPUs are designed in a Harvard architecture with separated data and instruction memory. There is an individual instruction memory block for each CPU while the data memory is implemented as a common device with four independent ports (see fig. 3.10).

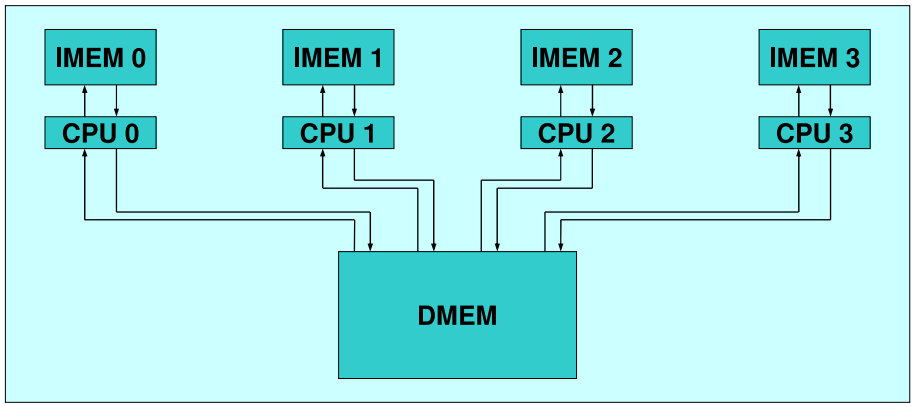

Figure 3.10: CPUs' data and control flow

Since both the data and the instruction flows of the CPUs are independent, the four CPUs represent a Multiple Instruction Multiple Data (MIMD) processor [fl0]. For the instruction memories (IMEM) a single ported 4096×30 bits design by Virtual Silicon Technology Inc. has been chosen. Each instruction word contains 24 functional bits plus six redundancy bits. The quad-ported data memory (DMEM) has been developed as a microelectronic full custom design of the size of 1024×39 bits. Each data word contains 32 functional bits plus seven redundancy bits. Therefore the hamming distance of the instruction and the data code set is four. Because this allows to correct one bit errors and to detect up to two bit errors, the radiation hardness of the CPUs is improved.

The CPUs are designed with a RISC instruction set, which affords to use a two stage pipeline at an operation frequency of 120 MHz. The small pipeline depth avoids data dependencies in the instruction code, which is useful as the programs, which have to be executed, are quite short.

Each CPU can read the fit parameter registers of the tracklet candidate it has been assigned to. It possesses 16 private registers and accesses 16 additional global registers, which can be used for a fast data exchange in between the CPUs. For communication with other devices on the chip the CPUs are connected to a private input/output bus (e.g. event buffer read access or transactions to readout network interface) and the global input/output bus (see section 3.4). In order to start the CPUs with instruction code which is suitable for a given state of the chip (e.g. tracklet processing, raw data readout) there is an interrupt handler which sets the program counter of the CPUs to an appropriate start address which is defined in the interrupt vector table.

3.10 Readout Network Interface

The readout network interface of the TRAP chip is designed to merge data streams of up to four external input sources and those from the four internal CPUs [rs0]. The resulting data stream is forwarded to one external output destination (see fig. 3.11).

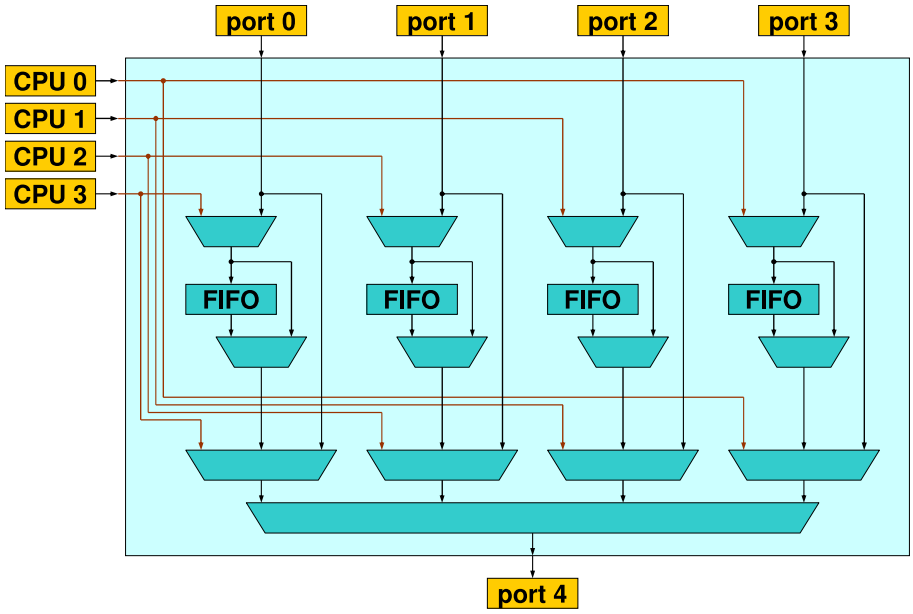

Figure 3.11: structure of the Readout Network Interface

The network interface operates in two modi. For tracklet transmission it is important to ship data as fast as possible. Thus, the tracklet words are sent directly to the output multiplexers of the interface module. All incoming data is buffered in the First-In First-Out Memories (FIFOs). In order to speed up data forwarding a FIFO is bypassed if it is empty and no data is contributed from the CPUs (Merger Functionality). The buffering is necessary because there is no handshaking procedure between the ports of two chips in tracklet mode.

In raw data readout mode the amount of data is larger and timing is not that tight. Consequently, some handshaking protocol is performed between the ports of connected chips. In this mode the FIFOs are used to buffer the data coming from the on-chip CPUs. In order to save power, this makes it possible to switch off the CPUs during major parts of the readout while the FIFOs of the network interfaces of all TRAP chips in readout tree are flushed (see section 2.7).

While the CPUs' data word width is 32 bits, the network interface works with words of the width of 16 bits only. Therefore buffers are placed at the input ports of the CPUs to the interface module. This makes it possible to split each incoming word in two successive half words. To improve utilization of the

TRAP's input/output pins the internal 16-bit words of the interface module are split again in two 8-bit words which are transmitted on each of the clock edges. At last, the external communication between two network interfaces is performed on a data bus of the width of 8 bits plus a parity bit and a spare bit, one control and one clock/strobe signal at a frequency of 120 MHz Double Data Rate (DDR).

4

Event Acquisition and Readout

4.1 Functional Overview

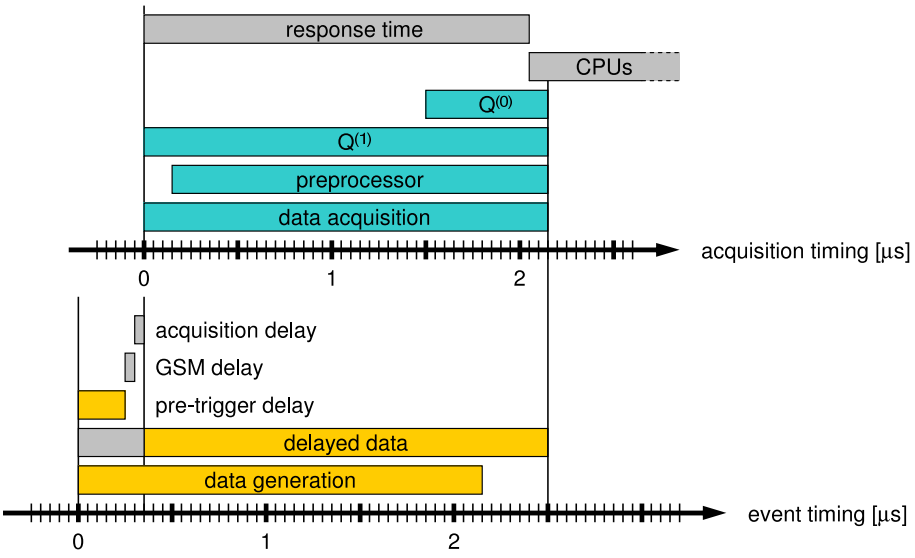

Figure 4.1: timing of the data acquisition

While analog-to-digital conversion and signal filtering within the TRAP chip are continuously active, data acquisition and readout are event oriented. The event data is stored in the event buffer, which incorporates one 11×64 bits memory for each of the 21 input channels. One data word contains 10-bit ADC data and one parity bit. It can be accessed by the CPUs for test purposes. In addition it is checked continuously in parallel to each read access. The results are monitored in parity violation counters.

For the purpose of zero suppression, the event buffer has been amended by a set of indicator bits. They are used to mark those data words, which are supposed

to be transmitted in case of a raw data readout (see section 4.3). The indicators as well as the event buffer memories are accessible for reading by the CPU's private input/output bus. The CPUs compile the data to the actual readout stream, which is transmitted via the network interface (see section 4.2). For test purposes it is copied to the local data memory, where it is visible for the SCSN slave as well.

The timing of the data acquisition is controlled by various configuration settings. In Figure 4.1, there are two time axes. Event timing refers to the primary physical reaction, acquisition timing to the corresponding activity of the TRAP chip. The shift between event and acquisition timing is caused by the delay of the pre-trigger signal and some delay of the GSM which is due to the decoding of the pre-trigger signal and the synchronization of the acquisition to the sampling clock of the chip. The additional acquisition delay parameter is typically set to zero. In order not to loose data, the data is delayed by the front-end electronics to balance the delay of the acquisition machinery (see table 4.1).

PASA signal rise time	70 ns
ADC pipeline delay	130 to 140 ns
programmable input shift register	100 to 600 ns
crosstalk filter shift register	0 or 200 ns

Table 4.1: data delay contributions

sample content	start [ns]	duration [ns]
baseline (pre-sample)	0	100
amplification region peak	100	200
drift time	300	2000
signal decay (post-samples)	2300	0 to 700

Table 4.2: typical data content

The duration of the data taking phase is adjusted according to the data content (see table 4.2) as well as the activity of the larger charge accumulator ($Q^{(1)}$). The start and end of the preprocessor operation are aligned to the drift time. The second charge accumulator ($Q^{(0)}$) can be adjusted to that part of the drift time, where the signal is dominated by transition radiation photons in case of electrons to improve particle identification. At last, the response time to the GSM is set one cycle in advance to the end of preprocessor activity to avoid loss of time in tracklet processing between preprocessor and CPU operation.

4.2 Data Transmission

The data transmission consists of several steps. At first, the data, which has to be transferred, is determined by the usage of the data indicator bits (see 4.3). Then, data is read from the event buffer, combined to a series of 32 bit words and finally sent to the readout network interface.

The readout is performed by all four CPUs in parallel. There is a unique assignment of the CPUs to the 21 channels of the event buffer (see table 4.3). CPU 0 to 2 compile data out of five channels while CPU 3 takes care of the data of six channels. The streams from the four CPUs are buffered in the FIFOs of the network interface and merged afterwards. The order corresponds to the CPU number.

CPU number	channels
0	0 to 4
1	5 to 9
2	10 to 14
3	15 to 20

Table 4.3: assignment of CPUs to event buffer channels

The indicator bits $d_{i,t}$ are grouped in two 32-bit data words per channel. Typically, only one of them is used, since there are not more than 30 samples per event which can easily be stored in one half of the event buffer memories:

$$D^i = d^i_{31}...d^i_2 d^i_1 d^i_0 \tag{4.1}$$

Because the data is structured channel-wise, a first reduction of the data is achieved by the selection of the channels c^i which are actually transmitted:

$$c^i = \bigoplus_{t=0}^{31} d^i_t \quad \Longleftrightarrow \quad c^i = \begin{cases} 1 = \text{TRUE} & \text{if } D^i \neq 0 \\ 0 = \text{FALSE} & \text{otherwise} \end{cases} \tag{4.2}$$

The CPUs determine their channel indicators and store it to global registers. Hence, they are accessible to CPU 0, which combines them to the channel transmission mask:

$$C^i = c^{20}...c^2 c^1 c^0 \tag{4.3}$$

The first 32-bit word, which is transmitted by CPU 0 is the channel transmission mask plus some extra bits to prevent it to be equal to the end marker. Next, the CPUs send the information of the marked channels to the network interface. Each channel block starts with the sample transmission mask (eqn. 4.1), which is also slightly modified to be unequal to the end marker. Then, the marked data is shipped in the order of the time bins. One 32-bit word within the data stream contains three ADC values plus two extra bits. Finally, the raw data end marker is sent. The whole data sequence out of one TRAP chip is shown in table 4.4.

data word	format	annotation
once per chip		
chip header	$\{c^i\}_i$ 1110 $\{I^i\}_i$	$\{c^i\}_{20\leq i\leq 0}$ channel mask $\{I^i\}_{6\leq i\leq 0}$ chip identification
for all channel of a chip		
channel header	$\{d^i_j\}_j$ 11	$\{d^i_j\}_{29\leq j\leq 0}$ sample mask of channel i
for all samples of a channel		
sample data	$\{e^i_{n_{j+2},l}\}_l\{e^i_{n_{j+1},l}\}_l\{e^i_{n_j,l}\}_l$ p_1p_0	$\{e^i_{n_j,l}\}_{9\leq l\leq 0}$ value in time bin j of channel i p_1p_0 alternating = 01 or = 10, starting with 01
once per stream		
end marker	$\{0\}$	

Table 4.4: raw data stream format

4.3 Zero Suppression

To optimize the usage of mass storage and to keep the readout time low, the data is compressed. The main idea is not to transmit data which contains variations around the baseline only ('zero').

The compression is based on a set of indicator bits (see section 4.2), one for each sample. These indicators of channel i in time bin j are set due to various criteria:

$$\text{maximality} \qquad \overline{Z}^{0,i}_j = \begin{cases} 0 & \text{if } e^i_j \geq e^{i-1}_j \text{ and } e^i_j \geq e^{i+1}_j \\ 1 & \text{otherwise} \end{cases} \tag{4.4}$$

$$\text{triplet threshold} \qquad \overline{Z}^{1,i}_j = \begin{cases} 0 & \text{if } e^{i-1}_j + e^i_j + e^{i+1}_j \geq T^{\text{triplet}}_{\text{I}} \\ 1 & \text{otherwise} \end{cases} \tag{4.5}$$

$$\text{single threshold} \qquad \overline{Z}^{2,i}_j = \begin{cases} 0 & \text{if } e^i_j \geq T^{\text{single}}_{\text{I}} \\ 1 & \text{otherwise} \end{cases} \tag{4.6}$$

The actual indicator is a configurable combination of those criteria:

$$\tilde{d}_j^i = LUT_I\left(\{\overline{Z}_j^{b,i}\}_{2\geq b\geq 0}\right) \tag{4.7}$$

Finally, it is possible to take the evaluation of neighbor channels into account:

$$d_j^i = \begin{cases} \tilde{d}_j^{i-1} \wedge \tilde{d}_j^i \wedge \tilde{d}_j^{i+1} & \text{with neighbor sensitivity} \\ \tilde{d}_j^i & \text{otherwise} \end{cases} \tag{4.8}$$

This provides a lot of flexibility for future optimization after some experience with the real system. At present, two compression strategies have been investigated, single sample threshold zero suppression and cluster transmission.

The compression ratio of the classical zero suppression as a function of the single sample threshold is shown in figure 4.2. The threshold is corrected for the baseline. The compression estimation is based on simulated events (ALIROOT) with different levels of detector occupancy between 25 % ($\frac{dN}{dy} = 2000$) and 100 % ($\frac{dN}{dy} = 8000$). The data is disturbed artificially with noise with a RMS of 1.0 ADC Counts. One can see, that the transmitted data volume decreases dramatically around the baseline. At thresholds beyond 3 ADC Counts relative to the baseline, the main benefit of this compression strategy is reached. This is far away from the most propable value of minimum ionizing particles, which is about 40 ADC Counts, summed over all channels affected.

At threshold values below the baseline ("black event"), the coding overhead due to the indicator word transmission is visible by a compression ratio of about 110 % in comparison to the transmission of the raw data only.

Alternatively, the transmission of data, which satisfies cluster criteria has been analyzed (fig. 4.3). This means, that samples at local maxima and their neighbors will be transmitted, if their sum is above the give triplet threshold. Since the threshold is compared with sample triplets, the decrement of the compression ratio is slower than in the example above. The main benefit of this compression strategy is expected at threshold values beyond 4 ADC Counts above the baseline.

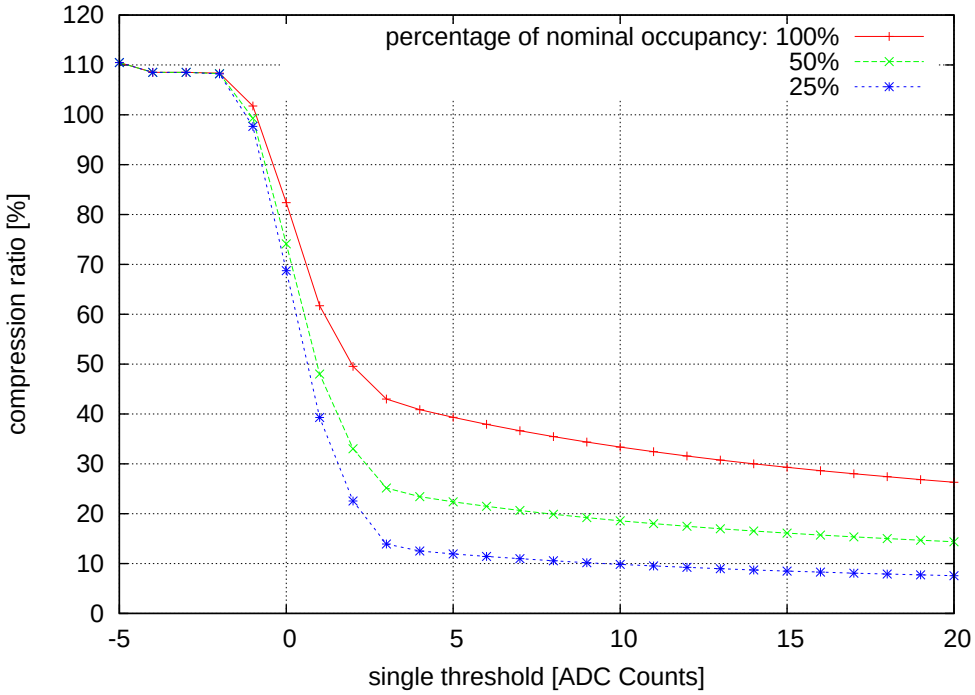

Figure 4.2: compression ratio in case of a single threshold criterion

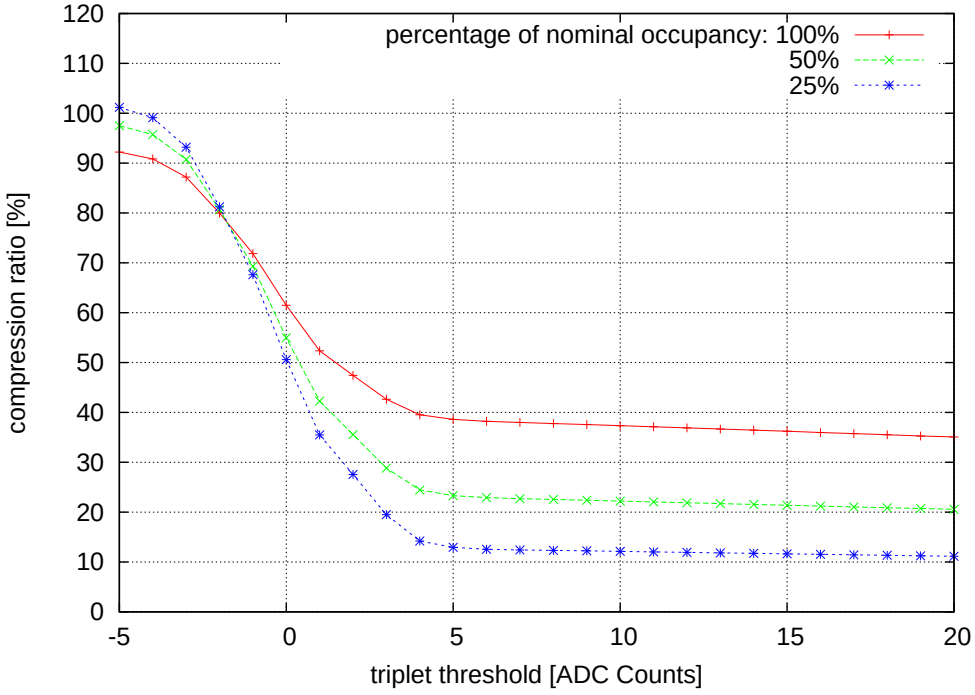

Figure 4.3: compression ratio in case of cluster transmission

5

Digital Filter

5.1 Functional Overview

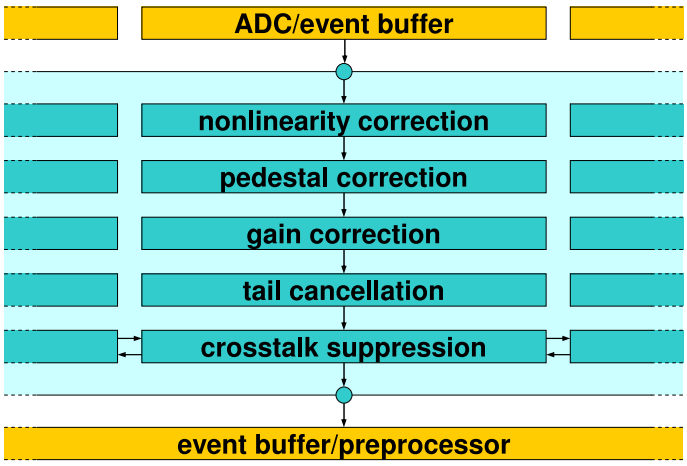

Figure 5.1: data flow through the five stages of the TRAP's digital filter

The digital filter of the TRAP chip is organized channel-wise (see section 3.6). Within each channel there are five different functional blocks (see fig. 5.1). The input data is provided by the TRAP ADCs or by the event buffer in case of internal filter testing. The data word width is enlarged from 10 bits to 12 bits within the first filter stage. The two bits are added behind the decimal point in order to keep the filter's rounding errors below one ADC Count in worst case. The filtered data is distributed to the preprocessor for particle tracking and to the event buffer for event data acquisition.

The filter is designed according to the signal generation process of the TRD [mg0, p. 23]. The gas of a chamber is ionized by either a charged particle which is passing by or by the absorption of a transition radiation photon. Due to a

homogeneous electric field inside the drift region of a chamber, the ionization electrons drift into the amplification region with a constant velocity. The drift distance is mapped to the arrival time of the cluster.

The amplification region incorporates a series of cylindric electric fields along the anode wires. While the electrons drift towards that wire, they are accelerated by the increasing electric field. Due to the energy increment this leads to a cascade of electrons encountering the anode wire. This effect amplifies the moving charge by a factor of ~ 4000. The electrons are leaving the amplification region fast via the anode wire while the remaining gas ions drift slowly. The signal is derived from the charges which are induced on cathode pads next to the anode wire. The response of the system is strongly asymmetric with respect to the temporal behavior. The signal increment, which is in the range of some nanoseconds, is determined by the drift velocity of the electrons. The signal decay is dominated by the velocity of the ions. It can be approximated by the sum of several exponentials. The temporal behavior of the amplification process is discussed in section 5.5.

The cathode pads have been designed such that a maximum of three pads per pad row is involved in the detection of a cluster. The charge sharing of the pads is described by the pad response function. Since the length of a pad ($\approx$ 8 cm) is relatively large compared to its width ($\approx$ 7 mm) or the thickness of the drift region (30 mm), the z dimension can be neglected and the discussion can be reduced to that of a two-dimensional data matrix per event. The remaining dimensions are the drift time, which corresponds to the distance of the original cluster to the amplification region, and the pad numbering within a pad row.

The combination of the cathode pads with the ribbon cable, which connects 18 of them to a PASA chip (see section 2.7), leads to a capacitive coupling of adjacent detector channels. The crosstalk is described in section 5.6.

The charge signals are measured by the PASA chip in combination with the TRAP ADCs. For analog signal processing, the signal is biased to some pedestal value. Since the pedestal varies from channel to channel, its correction is foreseen (see section 5.3). The linear amplification of the signal differs from channel to channel, too. To achieve a locally constant gain factor, another correction has to be performed (see section 5.4). Finally, there is a small common nonlinear contribution to the amplification. Its correction is discussed in section 5.2.

The complete signal path from a charge cluster to the digital front-end electronics is described by the operation A:

$$A = F_{\mathrm{Nonlin}} \circ F_{\mathrm{Gain}} \circ F_{\mathrm{Ped}} \circ F_{\mathrm{Xtalk}} \circ G_{\mathrm{PRF}} \circ F_{\mathrm{Tail}} \tag{5.1}$$

All operators but F_{Nonlin} are linear and translation invariant with respect to the sampling time. They can be exchanged arbitrary with one boundary condition: F_{Gain} and F_{Ped} cannot be exchanged with F_{Xtalk} or G_{PRF} because they are not translation invariant with respect to the pad dimension while F_{Xtalk} and

G_{PRF} cover more than one pad. Thus, equation 5.1 can be modified to:

$$A = F_{\mathrm{Nonlin}} \circ F_{\mathrm{Ped}} \circ F_{\mathrm{Gain}} \circ F_{\mathrm{Tail}} \circ F_{\mathrm{Xtalk}} \circ G_{\mathrm{PRF}} \tag{5.2}$$

The only contribution to A which is of interest is G_{PRF}: Since the detector occupancy is low, the clusters along the tracks of different particles can be assumed to be isolated in pad direction. Accordingly the charge sharing of adjacent pads is used to determine the position of clusters with a sub-pad resolution.

All other operators represent undesirable correlations of the signal in time (F_{Tail}, Xtalk), inhomogeneities within the signal path (F_{Ped}, F_{Gain}) or deviations of the common system from the aimed linear behavior (F_{Nonlin}). It is the task of the digital filter D to keep those influences as low as possible:

$$D \circ A = G_{\mathrm{PRF}} \tag{5.3}$$

$$\Longleftrightarrow D = F_{\mathrm{Xtalk}}^{-1} \circ F_{\mathrm{Tail}}^{-1} \circ F_{\mathrm{Gain}}^{-1} \circ F_{\mathrm{Ped}}^{-1} \circ F_{\mathrm{Nonlin}}^{-1} \tag{5.4}$$

The five inverse operators occurring in equation 5.4 are represented by the five separate filter stages.

5.2 Nonlinearity Correction

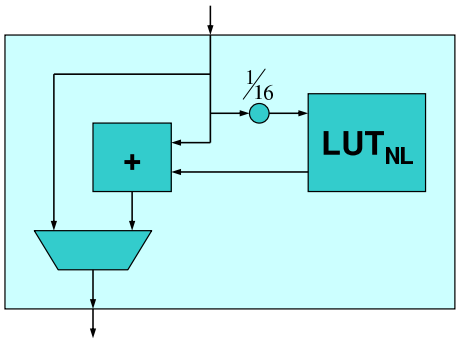

Figure 5.2: structure of the nonlinearity correction filter

Each part of the signal chain contributes to an overall nonlinearity. These are especially the gas amplification, the amplification of the PASA and the conversion within the ADCs. Because the contribution of the electronics is systematic and dominates the total nonlinearity, it can be corrected by a common correction look-up table:

$$O(t) = I(t) + \frac{1}{4} \cdot LUT_{\mathrm{NL}} \left(\left\lfloor \frac{I(t)}{16} \right\rfloor \right) \tag{5.5}$$

This implies, that there is a unique mapping between ideally linear distributed signal amplitudes and the disturbed signal amplitudes. Since the correction is

expected to change slowly with respect to the signal amplitude, a common deviation is hold for groups of 16 ADC values only. The filter input data is 10 bits wide while the width of the output data is 12 bits. The two bits are added behind the decimal point. They correspond to the lower two bits of the entries of the correction look-up table. The width of the entries of the look-up table is restricted to 6 bits. This defines the maximum correction:

$$LUT_{\mathrm{NL}}\left(\frac{I(t)}{16}\right) \in \left[0, \frac{2^6-1}{4}\right] \equiv [0, 15.75] \qquad \forall\ I(t) \tag{5.6}$$

First upper estimates of the nonlinearity were in the range of four to five ADC Counts, which is easily covered by the present implementation of the look-up table. Figure 5.2 shows the structure of the nonlinearity correction filter.

The filter output is connected to a multiplexer in order to have the opportunity to bypass its calculus. The calibration of the correction look-up table is discussed in section 7.1.

5.3 Pedestal Correction

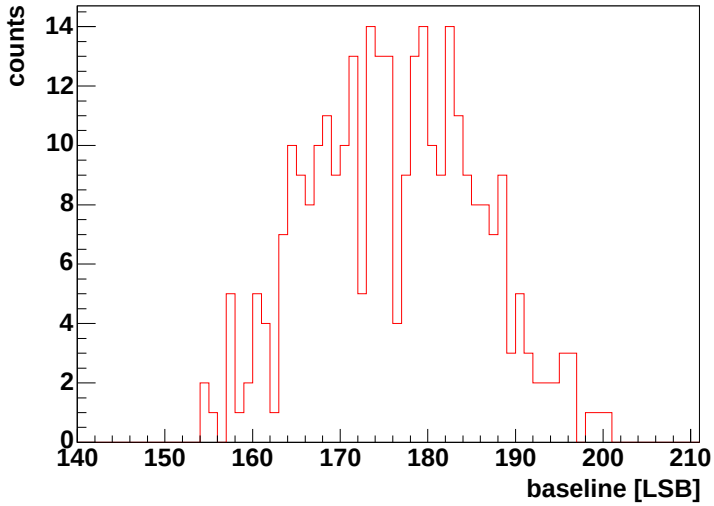

Figure 5.3: baseline distribution of one readout board

The amplification process of the PASA shifts the signal to a pedestal value, which is changing slowly in time and varies strongly from channel to channel (see fig. 5.3). To prevent undershoots it is desirable to have a small and well defined baseline value. To reach that a configurable pedestal value has to be added and the actual baseline needs to be determined and subtracted. A recursive filter of first order has been implemented to fulfill this task. On the right hand

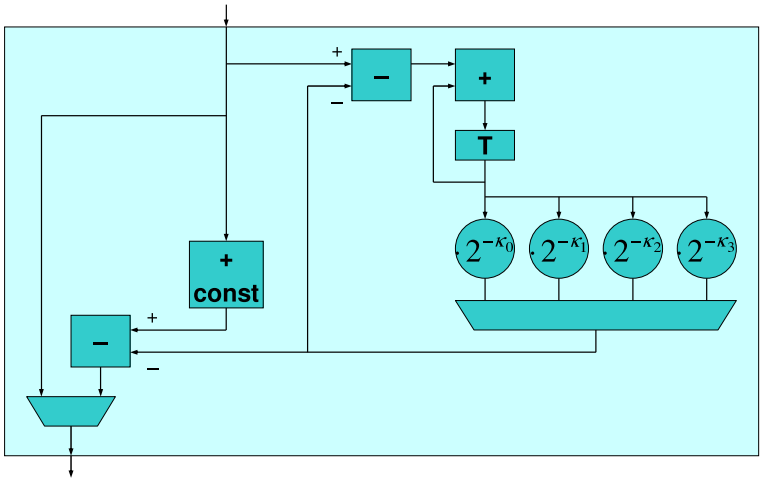

Figure 5.4: structure of the pedestal correction filter

side in figure 5.4, the circuits for the pedestal determination are outlined. The register contains the estimated pedestal P_{est} scaled by a factor of 2^{κ_i}. Its recursion equation can be read from the structure:

$$P_{\text{est}}(t + \Delta t) = 2^{-\kappa_i} \cdot I(t) + \left(1 - 2^{-\kappa_i}\right) \cdot P_{\text{est}}(t) \tag{5.7}$$

The sampling time is given by Δt. The difference of the relaxation constant $(1 - 2^{-\kappa_i})$ from one is given by the same number as the ratio of the estimated pedestal and the corresponding register entry, $2^{-\kappa_i}$. Thus, the corresponding multiplier can be implemented as a bit shifter, which reduces the number of gates involved to zero. The filter can be operated using one out of four time constants. This corresponds to four different scaling units $2^{-\kappa_i}$ whose output can be selected by a configurable multiplexer.

A given constant (P_0, see section 7.2) is added to the input data, the intermediate result is represented with an additional bit to prevent overflows. Subsequently, the estimated pedestal is subtracted. Finally, there is an output multiplexer which can be used to bypass the pedestal correction filter.

The impulse response of the pedestal estimation is given by:

$$P'_{\text{est}}(t) = 2^{-\kappa_i} \cdot \left(1 - 2^{-\kappa_i}\right)^{t-\Delta t} \cdot 1_{t>\Delta t} \tag{5.8}$$

Neglecting the addition of an arbitrary constant value, the impulse response of the filter can be described like:

$$f_{\text{Ped}}(t) = \delta_t - P'_{\text{est}}(t) \tag{5.9}$$

With $\xi = \frac{-\log\left(1-2^{-\kappa_i}\right)}{\Delta t}$, its frequency response has the following form:

$$\tilde{f}_{\text{Ped}}(\omega) = 1 - e^{-i\omega\Delta t} \frac{2^{-\kappa_i}}{\xi + i\omega} \tag{5.10}$$

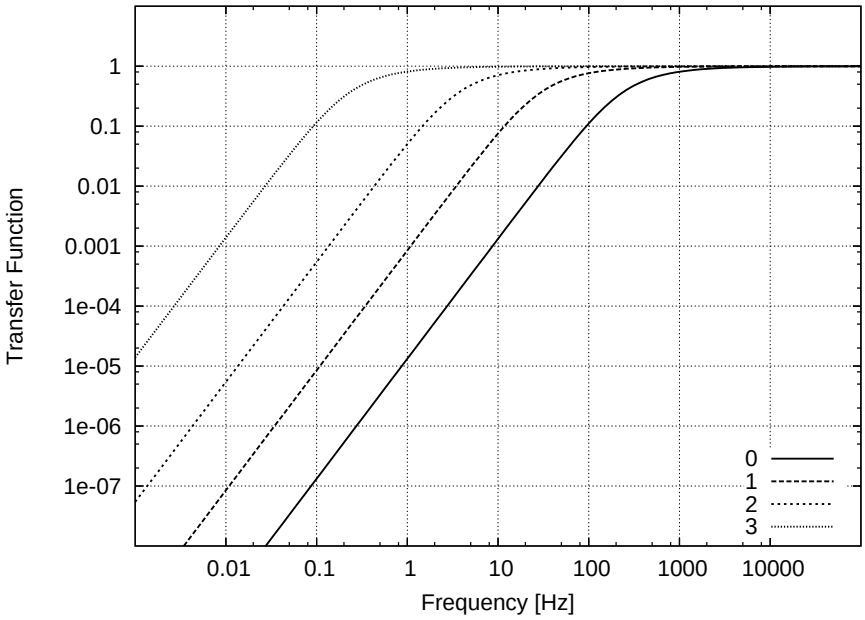

Figure 5.5: absolute value of the transfer function of the pedestal correction filter at the four available settings

Since $2^{-\kappa_i} \ll 1$, it is possible to approximate $\xi \approx \frac{2^{-\kappa_i}}{\Delta t}$. Introducing the sampling frequency $\nu_0 = \frac{1}{\Delta t}$, equation 5.10 can be modified to:

$$\tilde{f}_{\mathrm{Ped}}(\omega) = 1 - e^{-i\omega\Delta t}\frac{1}{1+\frac{i\omega}{2^{-\kappa_i}\cdot\nu_0}} \tag{5.11}$$

The phase factor $e^{-i\omega\Delta t}$ is close to one for lower frequencies $\omega \ll 2\pi\nu_0$. It can be neglected in general, since the second term vanishes for higher frequencies $\omega \gg 2\pi\nu_0 \cdot 2^{-\kappa_i}$. The final form of the frequency response is given by:

$$\hat{f}_{\mathrm{Ped}}(\nu) = 1 - \frac{1}{1+2\pi i\frac{\nu}{2^{-\kappa_i}\cdot\nu_0}} \tag{5.12}$$

It is shown in figure 5.5. This is a low-pass filter which can be characterized by the cut-off frequency $\bar{\nu}_i$ at which the attenuation is $|\hat{f}_{\mathrm{Ped}}(\bar{\nu}_i)| = \frac{1}{2}$:

$$\bar{\nu}_i = \nu_0 \cdot \frac{1}{2\pi\sqrt{3}} \cdot 2^{-\kappa_i} \tag{5.13}$$

The four available settings of the filter dynamics are implemented according to table 5.1.

i	κ_i	$\bar{\nu}_i$[Hz]	$\tau_i = \frac{1}{\bar{\nu}_i}$[ms]
0	13	112.2	8.9
1	16	14.02	71.3
2	19	1.75	570.6
3	23	0.11	9129.2

Table 5.1: pedestal correction filter characteristics

5.4 Gain Correction

The gain of the system is determined by the gas amplification, the amplification of the PASA chip and the conversion gain of the ADCs. Each of those steps shows some channel to channel variation: The gas gain changes with various parameters like temperature, pressure, distance from anode wire to cathode pads and the anode wire radius. The gain variation of the electronics is caused by variations related to the production process. Because the ratio of the signal in adjacent channels is used for position and track calculations, the total gain needs to be constant for all channel of a TRAP chip. Therefore, it is sufficient to balance the gain locally. The spread within the domain of one MCM is expected to be below ±10%. The equalization is performed by digitally scaling of each individual channel (see fig. 5.6).

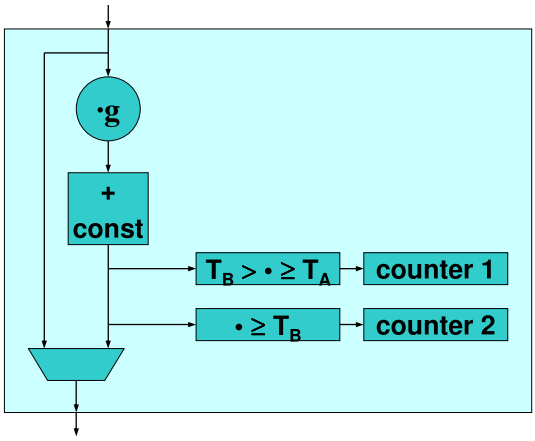

Figure 5.6: structure of the gain correction filter

The scaling factor is given as a fixed point number with 11 bits after the decimal point. Its dynamic range covers values of $g \in [1-2^{-3}, 1+2^{-3})$. Because the gain correction is performed after the pedestal correction, the given pedestal is multiplied by the correction factor, too. To compensate that, another adder has been implemented. The output of the gain correction filter can be bypassed

by another multiplexer.

The gain correction filter is located after the pedestal correction filter, because it is calibrated by observation of the amplitude spectrum and therefore the signals have to refer to a common baseline. For the iterative calibration procedure, the corrected signal values are compared with two thresholds T_A and T_B. At last, the number of occurances of values in the intervals $[T_A, T_B)$ and $[T_B, 2^{10})$ is counted to estimate the amplitude distribution. The calibration algorithm is executed offline. It is discussed in section 7.3.

5.5 Tail Cancellation

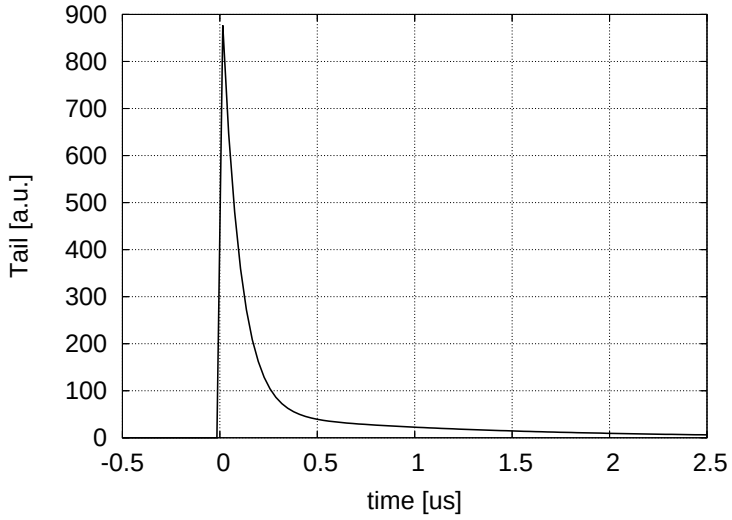

Figure 5.7: visible temporal response of the gas amplification

The first amplification step of the signal is gas amplification (see 5.1) with a quite high gain factor in the range of 4000. Unfortunately the amplification process shows a temporal behavior: While the rise time is in the nanoseconds range, the fall time is in the range of microseconds. It can be described by the sum of exponential functions, whose slowest is already suppressed by a first order pole/zero circuit within the PASA (see 2.7). The remaining contribution to the tail is in the lower microseconds range, it is shown in figure 5.7. It is known from previous analysis [mg0, p. 37], that it can be sufficiently approximated by an attempt using two exponential functions. In general the impulse response $g(t)$ looks like:

$$g(t) = 1_{t \geq 0} \cdot \sum_{n=1}^{N} \alpha_n e^{-\eta_n t} + R_N(t) \tag{5.14}$$

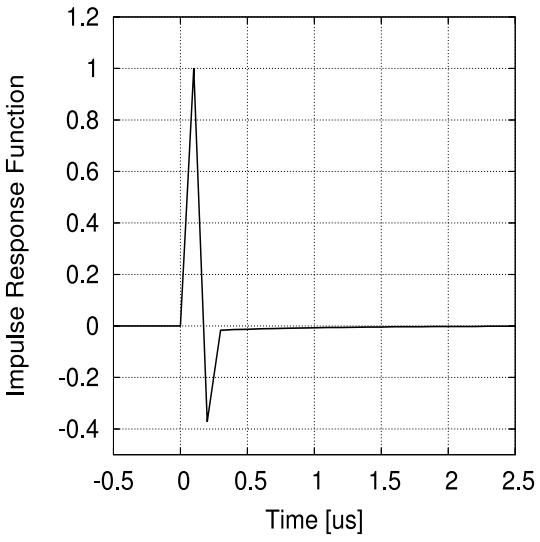

Figure 5.8: impulse response function of the tail cancellation filter

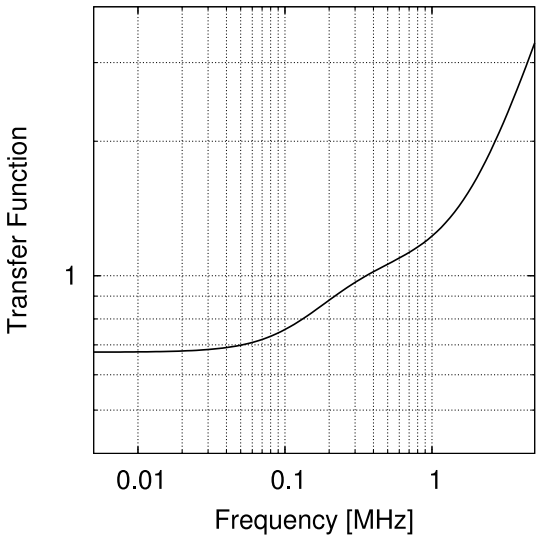

Figure 5.9: absolute value of the transfer function of the tail cancellation filter

Neglecting $R_N(t)$, a discrete, recursive representation of the tail generation process has the following form:

$$O_{\text{gen}}(t) = \sum_{n=1}^{N} r_n(t) \tag{5.15}$$

$$\text{with} \qquad r_n(t) = \alpha_n \cdot I_{\text{gen}}(t) + \lambda_n \cdot r_n(t - \Delta t) \qquad \forall\ n \tag{5.16}$$

Here, the decay factors $\lambda_n = e^{-\eta_n \Delta t}$ at the sampling time Δt are used. Since the dynamic range before and after the isolated tail generation is considered to by the same, a normalization of the weight parameters is chosen such that the amplitude of the primary peak is conserved:

$$\sum_{n=1}^{N} \alpha_n = 1 \tag{5.17}$$

To construct the tail cancellation filter, the tail generation process is inverted. Inserting equation 5.16 in 5.15 and resolving for $i_{\text{gen}}(t)$ leads to:

$$O_{\text{gen}}(t) = \sum_{n=1}^{N} \Big(\alpha_n \cdot I_{\text{gen}}(t) + \lambda_n \cdot r_n(t - \Delta t) \Big) \tag{5.18}$$

$$\iff \qquad O_{\text{gen}}(t) = I_{\text{gen}}(t) \cdot \sum_{n=1}^{N} \alpha_n + \sum_{n=1}^{N} \Big(\lambda_n \cdot r_n(t - \Delta t) \Big) \tag{5.19}$$

$$\iff \qquad I_{\text{gen}}(t) \cdot \sum_{n=1}^{N} \alpha_n = O_{\text{gen}}(t) - \sum_{n=1}^{N} \Big(\lambda_n \cdot r_n(t - \Delta t) \Big) \tag{5.20}$$

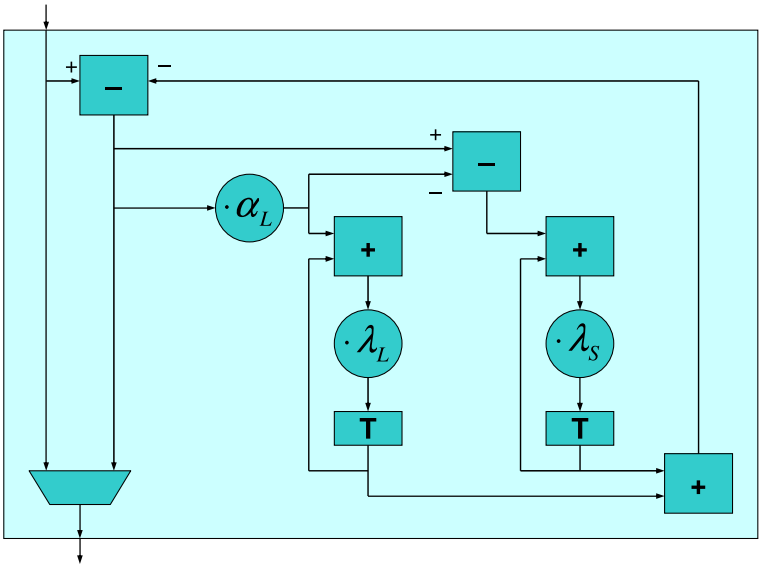

Figure 5.10: structure of the tail cancellation filter

With equation 5.17 and the substitutions $I_{\text{gen}}(t) \rightarrow O_{\text{Tail}}(t)$, $O_{\text{gen}}(t) \rightarrow I_{\text{Tail}}(t)$ the final description of the tail cancellation filter is given by:

$$O_{\text{Tail}}(t) = I_{\text{Tail}}(t) - \sum_{n=1}^{N} \Big(\lambda_n \cdot r_n(t - \Delta t) \Big) \tag{5.21}$$

The construction shows that it is the inverse of the tail generation filter. The output is given by the input minus the tail. It is reconstructed from the signal history, which is stored in a series of memory elements $r_n(t)$. Thus, it is a recursive filter, too, whose number of memory elements corresponds to the number of exponential functions, which are needed to approximate the visible tail. In the present implementation, an attempt with two exponential functions has been used. The impulse response function is shown in figure 5.8, the absolute value of the corresponding transfer function in figure 5.9. The two time constants of the visible tail can easily be seen in the double logarithmic display.

The actual implementation of the filter is shown in figure 5.10. In the upper left corner the subtracter can be seen, which subtracts the expected tail from the input value. The result is distributed to the filter's output multiplexer which enables bypass functionality. According to equation 5.21, the expected tail is equal to the sum of the two memory elements $r_n(t)$. The multipliers λ_n have been moved before the actual registers because this does not change the calculus and the register entries are smaller. This relaxes timing and decreases the probability of saturation of the registers since they are protected against overflows.

The central parts of the filter are the two recursive units as described by equation 5.16. Each of them consists of a register $r_n(t)$, a multiplier for the signal decay, an adder and a multiplier for its weighted stimulation. Since the order of magnitude of the two decay constants is known [mg0, p. 42], the dynamic range, in which they can be chosen, is reduced. So, a couple of bits of the decay constants are fixed. This lowers the number of gates, which are spent for the corresponding multipliers. The decay constant of the slower exponential function can be chosen in the range of $\lambda_L \in [0.75, 1)$, that of the faster one in the range of $\lambda_S \in [0.25, 0.5)$ with a granularity of 2^{-11}. Because of the same arguments, the weight of the stimulus of the slower recursive unit is restricted to $\alpha_L \in [0, 0.5)$. Using equation 5.17 it follows, that

$$I_{\text{gen}}(t) \cdot \alpha_S = I_{\text{gen}}(t) \cdot (1 - \alpha_L) = I_{\text{gen}}(t) - I_{\text{gen}}(t) \cdot \alpha_L \tag{5.22}$$

Consequently, the multiplier for the weighted stimulation of the faster unit has been replaced by a subtracter with a relatively small expense of gates.

Figures 5.7, 5.8 and 5.9 are based on an example parameter set to illustrate the basic behavior of the tail cancellation filter. Since the dynamic behavior of the tail generation varies with environmental influences as well as geometrical inhomogeneities from chamber to chamber, a calibration process has been established. It is discussed in section 7.4.

5.6 Crosstalk Suppression

The signal of adjacent cathode pads are not independent from each other. First, there is some position-dependent correlation due to charge sharing which is used for position calculation with sub-pad resolution. Second, there is capacitative coupling of neighboring pads including their connectivity to the PASA chip. The fluctuations in the coupled pads reach amplitudes in the percent range. Their temporal behavior is shown in figure 5.11. It can be described as the slow derivative of the initial signal in the neighboring pads [xt0]. Since the frequency components above the Nyquist frequency cannot be neglected, the discrete crosstalk generation operator is not well defined. Therefore, an average of the signal with respect to the phase between sampling and the initial pulse is taken into account.

The average crosstalk generation is described by a two-dimensional filter matrix. Since only next neighbors are taken into account, the width in pad direction can be restricted to three. The length in time direction depends on the typical signal shape which is being taken into account for the averaging process. It has been shown, that in case of the TRD signals, a length of five samples is sufficient [mg0, p. 50].

The crosstalk suppression is based on the application of another two-dimensional filter matrix on the incoming signals. It is an approximation of the inverse matrix to that of the crosstalk generation process. The calibration process for its

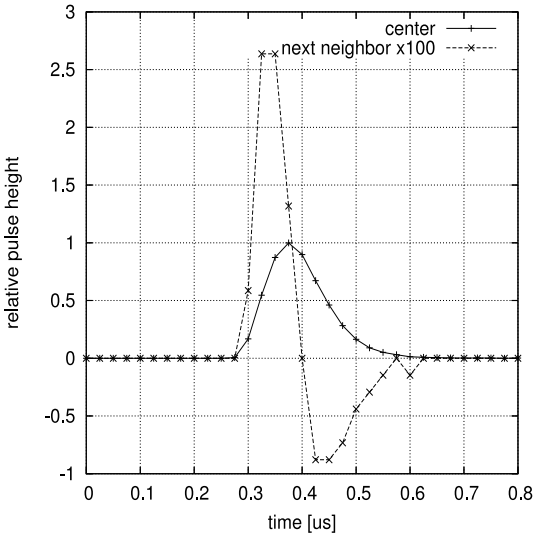

Figure 5.11: signal shape in a pulsed channel and induced signal shape in an adjacent channel

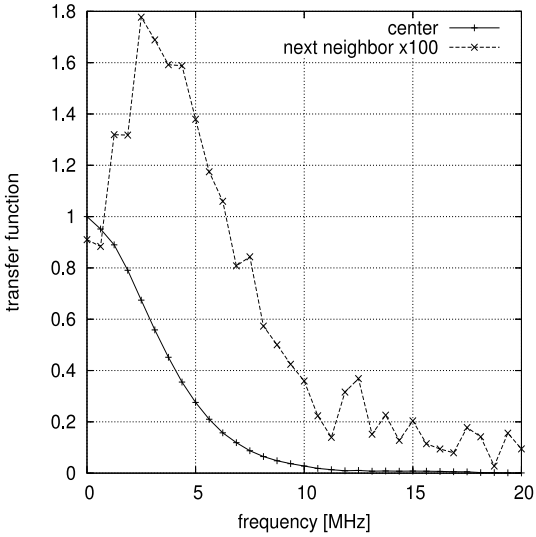

Figure 5.12: frequency spectrum of the signal induced by a pulse to a neighboring channel

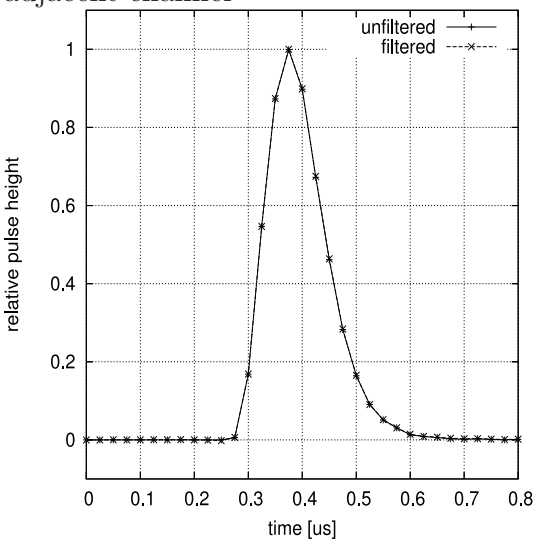

Figure 5.13: signal shape in a pulsed channel with crosstalk before and after the crosstalk suppression filter

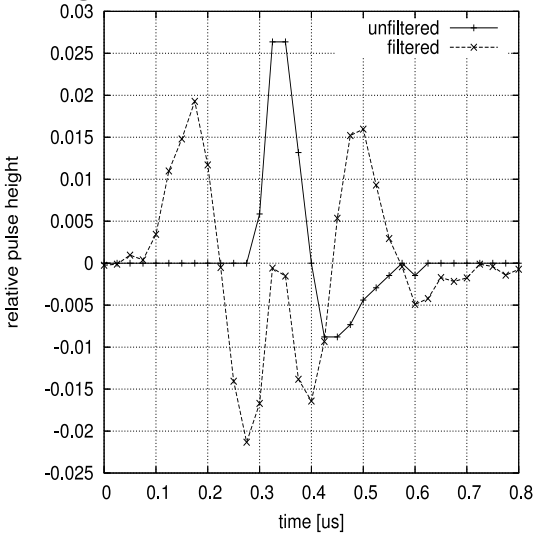

Figure 5.14: signal shape in a neighbor channel with crosstalk before and after the crosstalk suppression filter

entries is described in section 7.5. Though it is impossible to correct for crosstalk exactly, it can be suppressed slightly in amplitude [mg0, p. 51].

After the application of the filter to a series of input signals with various phases between initial pulse and sampling clock, the average response of the system of crosstalk generation, sampling and crosstalk suppression filter can be constructed. The comparison of the central peak before and after the filter is shown in figure 5.13, the signal shape in the neighbor channel in figure 5.14. The second order disturbances in the channels which are more than one apart are below 0.6 % with respect to the peak amplitude and can be neglected.

M_{-2}	0	M_{-2}
M_{-1}	0	M_{-1}
M_0	1	M_0
M_{+1}	0	M_{+1}
M_{+2}	0	M_{+2}

Table 5.2: entries of the crosstalk suppression filter matrix

In order to save gate resources the choice of the entries of the filter matrix is restricted (see tab. 5.2).

$$O(\rho,t) = (I * M)\ (\rho,t) = \sum_{\xi\in[-1,1]} \sum_{s\in[-2,2]} I(\rho-\xi, t-s)\cdot M(\xi,s) \tag{5.23}$$

The convolution of the signal with the filter matrix can be separated into three contributions:

$$\begin{aligned} O(\rho,t) \quad = \quad & \sum_s I(\rho+1, t-s)\cdot M(-1,s) \\ + \quad & \sum_s I(\rho+0, t-s)\cdot M(0,s) \\ + \quad & \sum_s I(\rho-1, t-s)\cdot M(+1,s) \end{aligned} \tag{5.24}$$

The weight parameters of the central column are set to constant values. The center of this column is set to one, all other entries to zero. This represents the identity operation:

$$\begin{aligned} O(\rho,t) = I(\rho,t) \quad + \quad & \sum_s I(\rho+1, t-s)\cdot M(-1,s) \\ + \quad & \sum_s I(\rho-1, t-s)\cdot M(+1,s) \end{aligned} \tag{5.25}$$

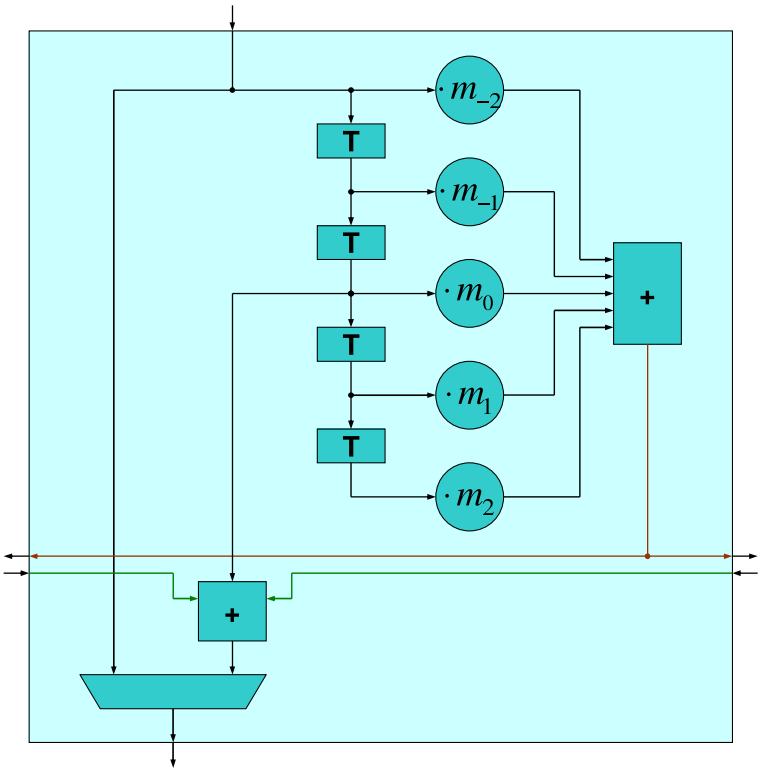

Figure 5.15: structure of the crosstalk suppression filter

Since the deviation is relatively small, it is sufficient to support first order corrections only. Thus the only free parameters are the weights of the marginal columns. Due to the symmetry of the matrix there are only five of them:

$$M(\xi, s) = M(-\xi, s) \qquad \forall\ s \tag{5.26}$$

Figure 5.15 shows the implementation of the convolution with the filter matrix. In order to access the incoming signal of the last five time bins, a shift register is used. Four values are taken from its memory units, while the first one is derived from the input directly. Because a real filter has to be causal and there are non-zero coefficients assigned to the 'future' of the signal, the time reference is shifted by two clock cycles. Therefore, the central signal values which are weighted with one are derived from the output of the second memory unit. This is the central input to the lower adder.

In order to save further gate resources, equation 5.25 can be transformed to:

$$O(\rho, t) = I(\rho, t) + C(\rho - 1, t) + C(\rho + 1, t) \tag{5.27}$$

The correction additives

$$C(\rho, t) = \sum_s I(\rho, t - s) \cdot M(+1, s) \tag{5.28}$$

can be used for both neighbor channels $O(\rho \pm 1, t)$. These are the two other inputs from the neighbor modules of the filter to the adder next to the output port. Its own contribution to the correction of the adjacent channels is computed according to the part on the right hand side of figure 5.15. The outputs of the memory units as well as the filter's input are weighted by the matrix entries $M(\pm 1, s) = M_s$ and added to gain $C(\rho, t)$ and distribute it to the neighbors. The weights can be adjusted in the range of $M_i \in [-0.125, 0.125)$ with a granularity of 2^{-10}: On the output port there is another multiplexer to have the possibility to bypass the filter.

6

Local Tracking

6.1 Overview

It is one of the main tasks of the TRAP chip to provide a local parameterization of particle tracks. Figure 6.1 shows a typical data matrix which is acquired during an event. The horizontal axis correspond to 21 adjacent detector channels whose charge signals have been amplified by several PASA chips. They are sampled at frequency of 10 MHz. Due to the charge drift within the detector the temporal behavior of the signal corresponds to the distance of the charge cluster to the amplification region (see section 2.3). This is represented by the vertical axis. The signal amplitudes are color encoded.

Locally, the track segments (tracklets) can be sufficiently approximated by a straight line model. It is determined by two parameters:

$$y = \vartheta \cdot x + \hat{y} \tag{6.1}$$

Here, the tuple (x, y) represents the position of a cluster on the tracklet. x is the time coordinate while y corresponds to the pad position. Since the slope is expected to be quite low, it is sufficient to consider the following error measure of the fit:

$$\chi^2 = \frac{1}{\sigma_y^2} \cdot \overline{\Big(y - (\vartheta \cdot x + \hat{y})\Big)^2} \tag{6.2}$$

Here, σ_y is the mean error of the pad position. A minimization of equation 6.2 with respect to ϑ and $\hat{y}$ leads to:

$$\text{slope} \qquad \vartheta = \frac{\overline{xy} - \overline{x} \cdot \overline{y}}{\overline{x^2} - \overline{x}^2} \tag{6.3}$$

$$\text{offset} \qquad \hat{y} = \frac{\overline{x^2} \cdot \overline{y} - \overline{x} \cdot \overline{xy}}{\overline{x^2} - \overline{x}^2} \tag{6.4}$$

Finally, to estimate the fit error, equation 6.2 is transformed to:

$$\sigma_y^2 \chi^2 = \overline{(y - \overline{y})^2} - \frac{\Big(\overline{(x - \overline{x})(y - \overline{y})}\Big)^2}{\overline{(x - \overline{x})^2}} \tag{6.5}$$

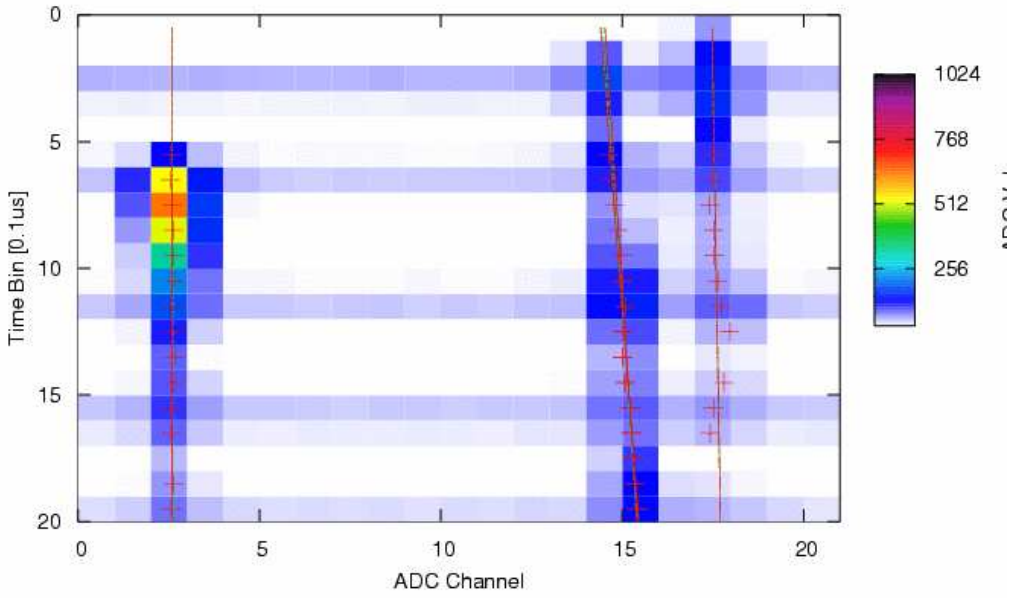

Figure 6.1: example event

σ_y^2 is unknown, but it is independent from the measurement. Thus, $\sigma_y^2\chi^2$ is used instead of χ^2 as a non-normalized error measure. The mean is calculate over all detected clusters N. In order to calculate the quantities introduced above, the following sums have to be built:

$$X = \sum_{n=1}^{N} x_n,\ Y = \sum_{n=1}^{N} y_n,\ XY = \sum_{n=1}^{N} x_n y_n$$

$$X^2 = \sum_{n=1}^{N} x_n^2,\ Y^2 = \sum_{n=1}^{N} y_n^2 \tag{6.6}$$

Hence, equations 6.3 to 6.5 change to:

$$\vartheta = \frac{N \cdot XY - X \cdot Y}{N \cdot X^2 - X \cdot X} \tag{6.7}$$

$$\hat{y} = \frac{X^2 \cdot Y - X \cdot XY}{N \cdot X^2 - X \cdot X} \tag{6.8}$$

$$\sigma_y^2\chi^2 = N \cdot Y^2 - Y \cdot Y - \frac{(N \cdot XY - X \cdot Y)^2}{N \cdot X^2 - X \cdot X} \tag{6.9}$$

It is the task of the TRAP chip to calculate and ship the fit parameters ϑ and $\hat{y}$. In order to keep the time to the trigger decision as low as possible, the error measure $\sigma_y^2\chi^2$ is calculated for test purposes only.

Because the trigger is based on the search for high energetic electron-positron pairs, those particles have to be separated from pion background. A criterion is the mean charge per cluster, which contains direct ionization of the gas by the particle and the charges due to the conversion of transition radiation photons in case of electrons (see section 2.3). Therefore, an additional quantity is calculated by the TRAP chips:

$$\frac{Q}{N} = \frac{1}{N} \cdot \sum_{n=1}^{N} q_n \tag{6.10}$$

q_n is the charge of the nth cluster, which is calculated by accumulating the charges of adjacent detector channels around the hits. With the input signals $i(\rho, t)$, this is described by:

$$q_n = I(\lfloor y_n \rfloor - 1, x_n) + I(\lfloor y_n \rfloor, x_n) + I(\lfloor y_n \rfloor + 1, x_n) \tag{6.11}$$

The TRAP chip is able to fit up to four tracklet, which corresponds to the maximum number of tracklets, that can be detected without interfering each other in the observed part of the detector. The fitting process is separated into two steps:

- **Accumulation of the Sums** (Eqn. 6.6, 6.10)
  This is performed by a specialized module, the tracklet preprocessor. According to the readout of the detector, the clusters are detected sequentially by the electronics and the accumulation takes place sequentially, too. It is discussed in section 6.2.

- **Calculation of the Fit Parameters** (Eqn. 6.7, 6.8, 6.10)
  This is accomplish by the four CPUs. They are started after the data acquisition in order to save power and not to disturb the measurement by coupling of higher digital noise due to their higher operation frequency (section 6.3).

6.2 Channel-wise Track Segment Fitting

The first step of tracklet fitting is the accumulation of the fit sums (eqn. 6.6, 6.10) and is performed by the tracklet preprocessor. Because the trigger is based on high energetic particles only, the tracklet fitting can be optimized for stiff tracks, such that the hits of each tracklet are distributed over a maximum of two adjacent channels. Consequently, the output of the TPP provides the hit sums of those channel pairs to the CPUs, which contain tracklet information. A priori, it

is unknown, to which channel pair the hits within a channel have to be assigned. Thus, the fit sums are accumulated for each channel independently. It is the task of the CPUs to combine them for the final parameter calculation.

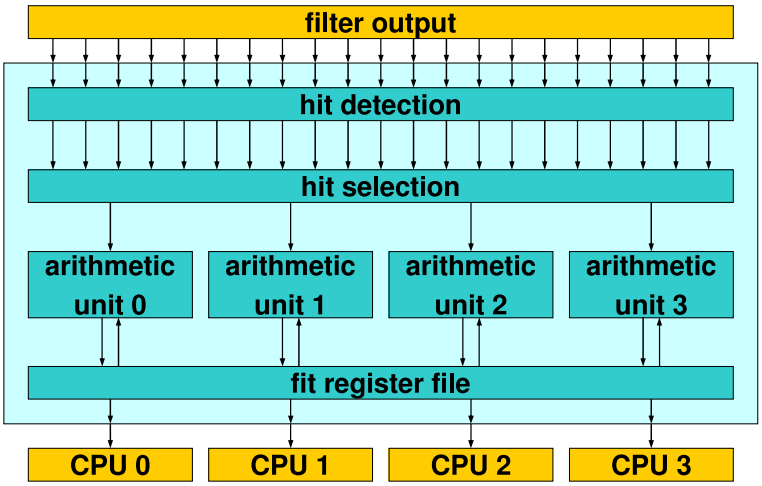

Figure 6.2: structure of the tracklet preprocessor

The implementation of the TPP is outlined in figure 6.2. The 21 output channels of the filter are processed by a hit detection unit first. A hit is a sample of a cluster. Using the total charge of a hit candidate

$$h(\rho,t) = i(\rho-1,t) + i(\rho,t) + i(\rho+1,t) \tag{6.12}$$

a channel ρ will be recognized as the central channel of a hit, if two conditions are fulfilled:

$$\begin{aligned} \text{maximality} \quad & i(\rho,t) \geq i(\rho-1,t) \\ & i(\rho,t) > i(\rho+1,t) \\ \text{minimum charge} \quad & h(\rho,t) \geq T_{\text{Hit}} \end{aligned}$$

The next step is to select up to four channels, which satisfy the hit requirements. The hit selection unit sorts the indicated channels with respect to their hit charge (eqn. 6.12). The hit information of the four hit channels with the largest hit charge is distributed to the four output ports of this unit. If there are less than four hits per chip at a given clock cycle or time bin, some ports are indicated to remain unused. The hit information of a channel incorporates its channel number ρ, total hit charge $h(\rho,t)$, its own value $i(\rho,t)$ and that of its neighbors $i(\rho \pm 1,t)$.

The hit information is processed by four parallel arithmetic units. They update the fit sums, which are stored in the Fit Register File (FRF). This is the

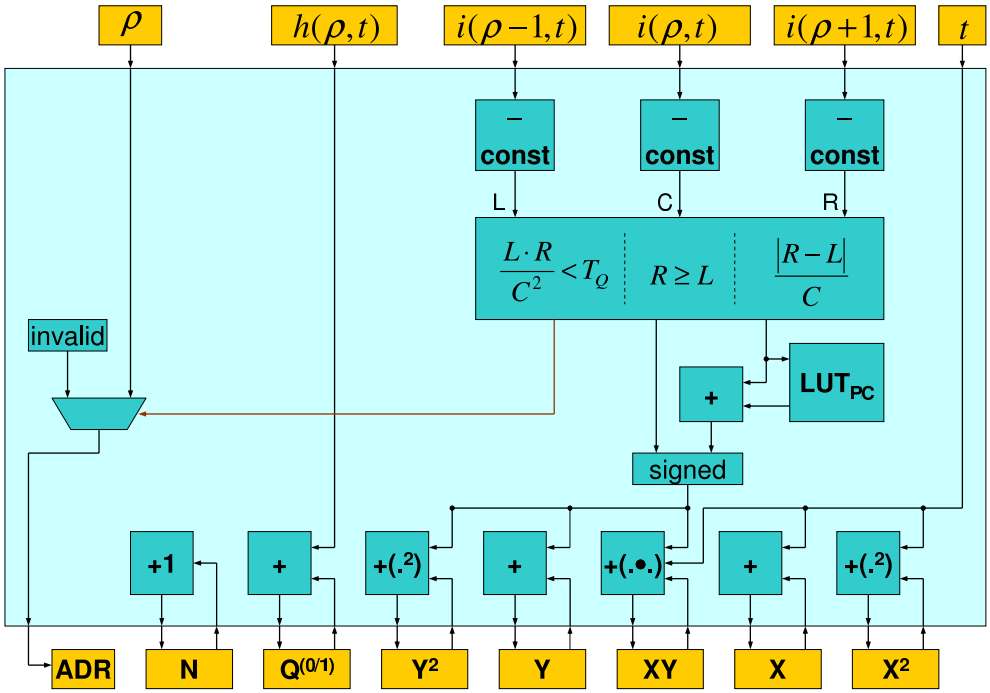

Figure 6.3: structure of an arithmetic unit of the tracklet preprocessor

interface to the CPUs. Here, the channels of the tracklet candidates can be accessed after the fitting process (see section 3.8).

An arithmetic unit transforms hit information to a local hit position (x_n, y_n) and updates the fit sums according to equations 6.6 and 6.10. The data flow is shown in figure 6.3. The channel number ρ is mapped to the FRF address. If the regarded hit does not satisfy some quality criterion (eqn. 6.30), the address will be set to an invalid value. In that case, the FRF entries are not affected by any of the other ports of this arithmetic unit. Otherwise, the remaining ports are used for a joint read-modify-write access on the given address.

The first port increases the number of hits of the addressed channel N by one, the second port increases the accumulated tracklet charges $Q^{(0/1)}$ by the total charge of the current hit. Two charges are accumulated in order to be able to determine $\frac{Q}{N}$ in different segments of the tracklets. The remaining ports increase the fit sums of X, X^2, XY, Y and Y^2. For their increment, the local hit position (x_n, y_n) is calculated. The first coordinate can be derived directly from the time information, which is provided by the acquisition controller (see sections 3.7, 3.8). It is implemented as a five bit value:

$$x_n = t \tag{6.13}$$

The second coordinate represents the shift of the hit relative to its central pad. It is possible to calculate sub-pad resolution, because the hits are considered to

be isolated and to show a well defined charge sharing between the involved pads. As a first step, the baseline, which is defined by the filter functionality, has to be subtracted from the values of the central channel and its neighbors:

$$\begin{aligned} L &= i(\rho-1,t) - C_{\text{Baseline}} \\ C &= i(\rho,t) - C_{\text{Baseline}} \\ R &= i(\rho+1,t) - C_{\text{Baseline}} \end{aligned} \tag{6.14}$$

The relation between charge distribution and cluster positioning is given by the Pad Response Function (PRF). It determines the relations between L, C and R. A first estimate of the inverse operation is given by:

$$y_n^{est} = \frac{1}{2} \cdot \frac{R-L}{C} \tag{6.15}$$

The relationship between real position and that quantity is unique, so there exists a one-to-one transformation C_{Pos} between them:

$$C_{\text{Pos}}: \; y_n^{est} \rightarrow y_n \tag{6.16}$$

Since the PRF is symmetric, it can be simplified to:

$$C_{\text{Pos}}: \; |y_n^{est}| \rightarrow |y_n| \tag{6.17}$$

The deviation $|y_n^{est}| - |y_n|$ is always positive. Its calibration procedure is discussed in section 7.6. Thus the second step is to calculate the quantity:

$$2 \cdot |y_n^{est}| = \cdot \frac{|R-L|}{C} \tag{6.18}$$

It is implemented using a fixed point representation of one bit before and seven bits after the decimal point. Its correction is given by additives out of a look-up table with 128 entries of five bits each:

$$|y_n| = \frac{1}{2} \cdot 2 \cdot |y_n^{est}| + LUT_{\text{PC}}(|y_n^{est}|) \tag{6.19}$$

The absolute value of the position is represented by a fixed point number with eight bits after the decimal point. Finally, the positive values $|y_n|$ are transformed to signed numbers, which adds another bit to their binary representation:

$$y_n = \begin{cases} +|y_n| & \text{if } R \geq L \\ -|y_n| & \text{otherwise} \end{cases} \tag{6.20}$$

From this, the additives to the fit sums are derived. The width of the additives and the corresponding sums is given in brackets:

$$X_N \;[9] = X_{N-1} + x_N \;[5] \tag{6.21}$$

$$X_N^2 \;[14] = X_{N-1}^2 + (x_N \cdot x_N) \;[10] \tag{6.22}$$

$$Y_N \;[14] = Y_{N-1} + y_N \;[9] \tag{6.23}$$

$$Y_N^2 \;[21] = Y_{N-1}^2 + (y_N \cdot y_N) \;[16] \tag{6.24}$$

$$XY_N \;[17] = XY_{N-1} + (x_N \cdot y_N) \;[14] \tag{6.25}$$

Together with the previous quantities and $h_N = L + C + R$

$$Q_N^{(0)}\ [15] = Q_{N-1}^{(1)} + \frac{1}{4} \cdot h_N\ [12] \tag{6.26}$$

$$Q_N^{(1)}\ [16] = Q_{N-1}^{(2)} + \frac{1}{4} \cdot h_N\ [12] \tag{6.27}$$

$$N\ [5] = N - 1 + 1\ [1] \tag{6.28}$$

$$\text{ADR}\ [5] = \rho\ [5] \tag{6.29}$$

they form the components of the FRF entry of one channel. They are update in up to four channels per clock cycle. In order to recognize overlapping hits of different tracklets, a quality criterion can be used to reject hits:

$$\text{accept} \Longleftrightarrow \frac{L \cdot R}{C^2} < T_Q \tag{6.30}$$

Its calibration is discussed in section 7.7. Because the expense of gates of multipliers is much less than that of dividers, the implementation corresponds to the modified equation:

$$\text{accept} \Longleftrightarrow L \cdot R < T_Q \cdot C \cdot C \tag{6.31}$$

The check can be bypassed, the threshold is adjustable in the range $[0, 2^{-4})$ with a granularity of 2^{-10}.

6.3 Tracklet Assembly

After drift time, a selection of promising channel pairs has to be inspected by the CPUs. The selection is performed by a specialized hardware which is included into the preprocessors'FRF. Because there are 21 ADC channels per chip and charge sharing is used for position determination, there are 19 channels left for fitting within the preprocessor. Out of these 19 channels 18 pairs of adjacent channels can be selected (see fig. 6.4).

The decision is taken on the basis of the number of hits in the channels of such a pair. There is a minimum number of hits in the left channel as well as a minimum number for the joint number of hits:

$$N_i \geq T_{N_{\text{Left}}} \quad \text{and} \quad N_i + N_{i+1} \geq T_{N_{\text{Total}}} \quad \forall\ i \in [0, 17] \tag{6.32}$$

Typically, the thresholds are set to $N_{\text{Left}} = 1$ and $N_{\text{Total}} = 8$, which corresponds to a tracklet involving at least the left channel and having hits in 40 % of the samples of both channels.

The channel pairs are sorted by their total number of hits. The four pairs with the maximum number of hits which fulfill the criterion 6.32 are assigned as tracklet candidates to the CPUs.

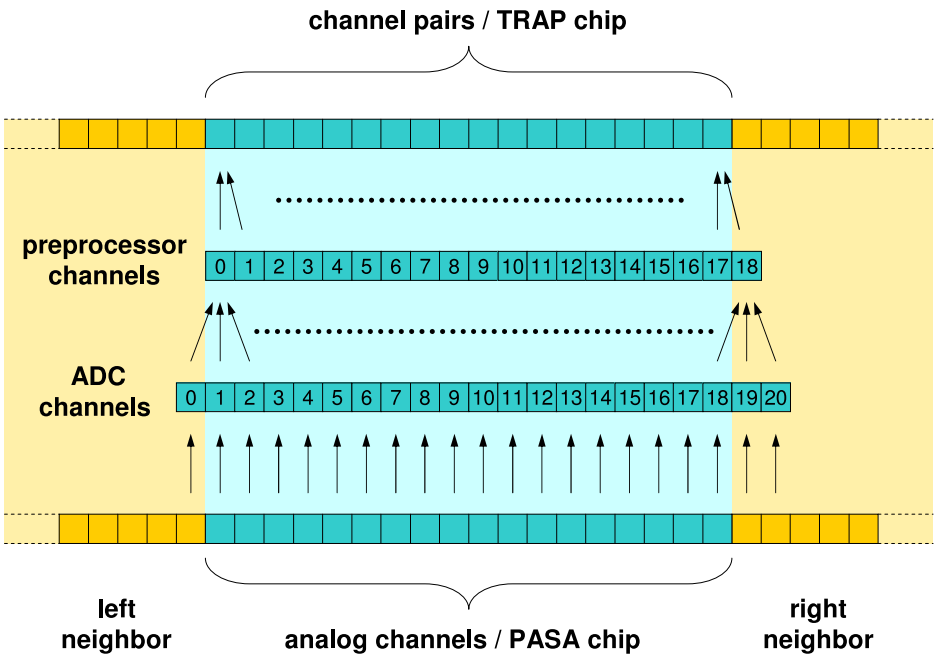

Figure 6.4: channel assignment within the ALICE TRD front-end electronics

According to table 6.1, each CPU has direct register-like access to the fit sums of its tracklet candidates. Because some of the quantities are stored as fixed point values and the CPUs support integer operations only, the values are represented by integers which have to be considered on a certain scale. All but the accumulated charges allocate their own register address.

The contribution of CPUs to the tracking is separated into the following steps:

- avoidance of the submission of double tracklets
- merging of the fit sums of the channel pair
- calculation of the parameters offset, slope and electron probability
- transformation of the parameters to the coordinate system of the GTU
- application of a local cut on the slope
- composition of the parameters to a tracklet word and its transmission

Because the tracking of the TRAP chips is optimized for high energetic particles, the tracklets can be considered to be distributed over a maximum of two channels. If a tracklet is distributed over more than one channel such that the number of hits in the right channel are enough to build a stand-alone tracklet, two

quantity	scale	CPU access	width
i	channels	FRF[0]	5
N_i	hits	FRF[1]	5
$Q_i^{(0)}$	ADC Counts	FRF[2].Bit[14..0]	15
$Q_i^{(1)}$	ADC Counts	FRF[2].Bit[32..16]	16
X_i	Δt	FRF[3]	9
X_i^2	Δt^2	FRF[4]	14
Y_i	$2^{-8} \cdot W_{\text{Pad}}$	FRF[5]	14
XY_i	$2^{-8} \cdot \Delta t W_{\text{Pad}}$	FRF[6]	17
Y_i^2	$2^{-16} \cdot W_{\text{Pad}}^2$	FRF[7]	21
$i+1$	channels	FRF[8]	5
N_{i+1}	hits	FRF[9]	5
$Q_{i+1}^{(0)}$	ADC Counts	FRF[10].Bit[14..0]	15
$Q_{i+1}^{(1)}$	ADC Counts	FRF[10].Bit[32..16]	16
X_{i+1}	Δt	FRF[11]	9
X_{i+1}^2	Δt^2	FRF[12]	14
Y_{i+1}	$2^{-8} \cdot W_{\text{Pad}}$	FRF[13]	14
XY_{i+1}	$2^{-8} \cdot \Delta t W_{\text{Pad}}$	FRF[14]	17
Y_{i+1}^2	$2^{-16} \cdot W_{\text{Pad}}^2$	FRF[15]	21

Table 6.1: Fit Register File access of the CPUs

channel pairs will be assigned to the CPUs: One covering left and right channel and the other one covering the right channel and its right neighbor. To handle this, some communication between the CPUs is performed via their global registers. The numbers of the right channels are stored there, such that each CPU compares these numbers from the other CPUs with its own left channel. In case of matching channels, the tracklet candidate with the lower number of hits is rejected.

CPUs with higher identification numbers are assigned to tracklet candidates with lower number of hits. Since for the most part the rejected tracklet candidates are assumed to be based on a subset of the hits of another candidate, the procedure can be reduced. Each CPU compares its own tracklet candidate with those of the CPUs with lower identification number and rejects its hit, if a match occurs.

It is the next step to merge the fit sums of the two channels of a tracklet candidate. Because the preprocessor calculates the hit position relative to the involved channels, the coordinates of the right channel are transformed to coordinates relative to the left channel:

$$(x, y)_{\text{Channel i+1}} \longrightarrow (x, y + W_{\text{Pad}})_{\text{Channel i}} \tag{6.33}$$

Accordingly, all sums based on the y coordinate are transformed:

$$Y_{\text{Channel i+1}} \longrightarrow \left(Y + W_{\text{Pad}}\right)_{\text{Channel i}} \tag{6.34}$$

$$XY_{\text{Channel i+1}} \longrightarrow \left(XY + X \cdot W_{\text{Pad}}\right)_{\text{Channel i}} \tag{6.35}$$

$$Y^2_{\text{Channel i+1}} \longrightarrow \left(Y^2 + 2 \cdot Y \cdot W_{\text{Pad}} + W^2_{\text{Pad}}\right)_{\text{Channel i}} \tag{6.36}$$

So, the complete set of arithmetics for the fit sum merging is given by:

$$N = N_i + N_{i+1} \tag{6.37}$$

$$Q^{(0/1)} = Q_i^{(0/1)} + Q_{i+1}^{(0/1)} \tag{6.38}$$

$$X = X_i + X_{i+1} \tag{6.39}$$

$$X^2 = X_i^2 + X_{i+1}^2 \tag{6.40}$$

$$Y = Y_i + Y_{i+1} + W_{\text{Pad}} \tag{6.41}$$

$$XY = XY_i + XY_{i+1} + X_{i+1} \cdot W_{\text{Pad}} \tag{6.42}$$

$$Y^2 = Y_i^2 + Y_{i+1}^2 + 2 \cdot Y_{i+1} \cdot W_{\text{Pad}} + W_{\text{Pad}}^2 \tag{6.43}$$

The next step is the calculation of offset and slope based on the merged fit sums according to the equations 6.7, 6.8 and 6.10. In order to save processing time, it is useful to calculate the division first:

$$\Gamma = \frac{1}{N \cdot X^2 - X \cdot X} \tag{6.44}$$

Next, the nominators are determined:

$$\Theta = N \cdot XY - X \cdot Y \tag{6.45}$$

$$\hat{Y} = X^2 \cdot Y - X \cdot XY \tag{6.46}$$

Then, the final fit parameters are calculated:

$$\vartheta = \Theta \cdot \Gamma \tag{6.47}$$

$$\hat{y} = \hat{Y} \cdot \Gamma + (i + 1) \tag{6.48}$$

The additive i to the offset transforms from the coordinate system relative to the left channel of a tracklet candidate to a coordinate system relative to the chip. Another 1 is added to avoid negative values. The electron probability is calculated as a function of $\frac{Q}{N}$, which is estimated by using one of the accumulated charges:

$$P_{\text{electron}} = P_{\text{electron}}\left(\frac{Q^{(1)}}{N}\right) \tag{6.49}$$

Now, the parameters are transformed to the local coordinate system of a stack, which is used by the GTU:

$$\tilde{\vartheta} = m^{(\vartheta)} \cdot \vartheta + c^{(\vartheta)} \quad (6.50)$$

$$\tilde{\hat{y}} = m^{(\hat{y})} \cdot \hat{y} + c^{(\hat{y})} \quad (6.51)$$

The slope is scaled to the deflection length, which is the difference of the y coordinates of the starting and end points of a tracklet. Further, it is matched to the metric system of the GTU. The additive $c^{(\vartheta)}$ contains the correction of the Lorentz deflection within the chamber and a correction which is due to a slight tilt of the pads relative to the orthogonal to the pad rows. The offset is matched to the metric system of the GTU, too. The additive $c^{(\hat{y})}$ represents the position of a chip relative to the stack on which it is integrated.

At last, the submission of tracklets which are not stiff enough is suppressed. Therefor the slope is compared to limit values which depend on the channel number:

$$\text{accept} \Longleftrightarrow T_i^{\min} < \vartheta < T_i^{\max} \quad (6.52)$$

The parameters of the CPU program (eqn. 6.49, 6.50, 6.51 and 6.52) are discussed in more detail in section 7.8. The final track information is combined to one 32 bit word:

parameter	symbol	granularity	range	bits
pad position	$\tilde{\hat{y}}$	160 µm	[-643.2 mm, 643.2 mm]	13
deflection length	$\tilde{\vartheta}$	140 µm	[-8.8 mm, 8.8 mm]	7
pad row		1	[0, 15]	4
electron probability	P_{electron}	0.39 %	[0, 1]	8

Table 6.2: Bit content of a tracklet word [jc0, p. 41]

It is sent to the readout network interface. In case of no tracklet transmission, the tracklet end marker, a void tracklet word, is sent. Figure 6.5 illustrates the implementation of the tracklet assembly in an actual assembler program, which running on each of the four TRAP CPUs. The operations are mixed slightly in order to reduce the run time.

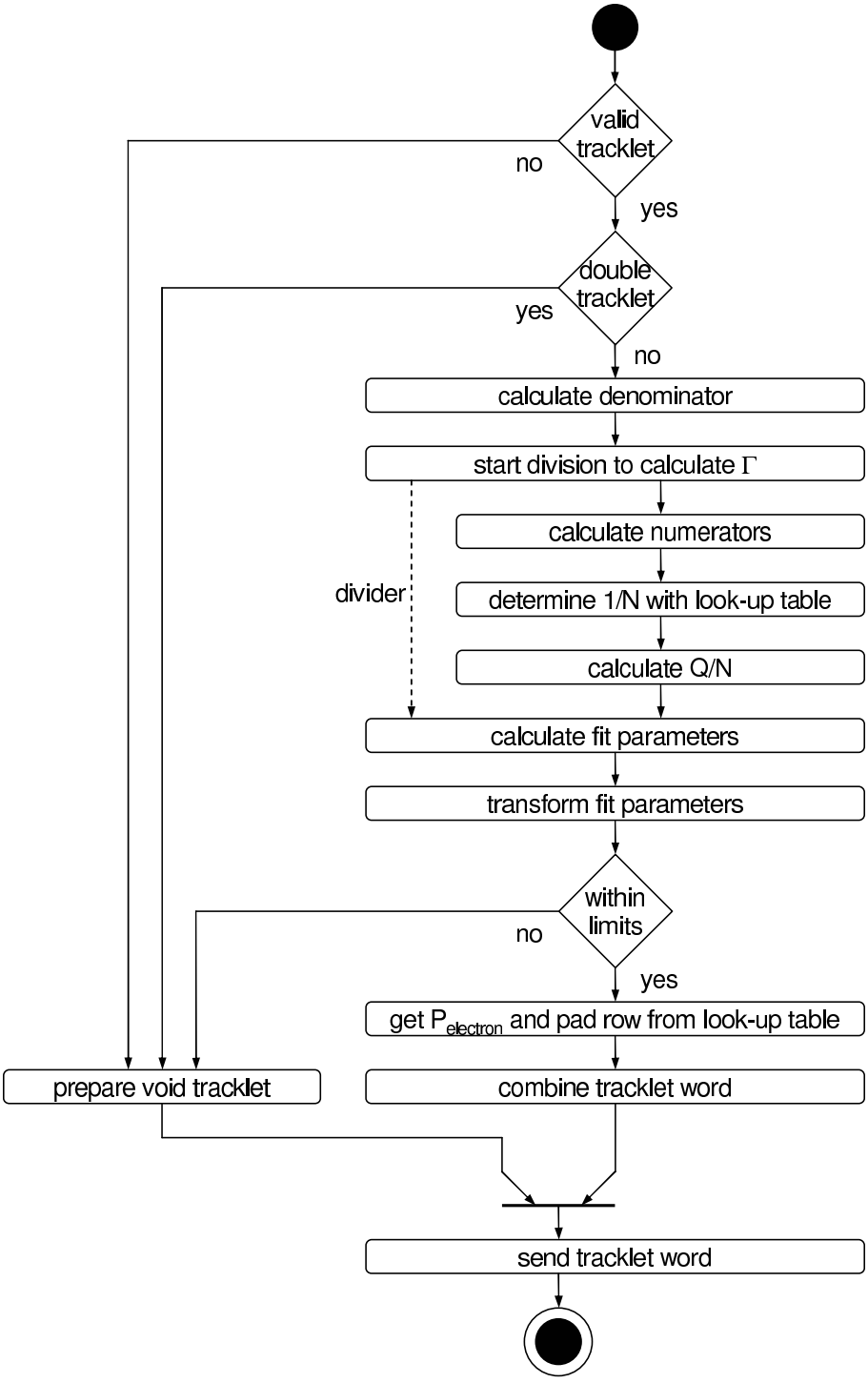

Figure 6.5: flow chart of the assembler implementation of the tracklet assembly

7

Parameter Calibration

7.1 Nonlinearity Correction Filter

It is the task of the nonlinearity correction filter to correct common deviations of the front-end electronics from an ideal linear behavior (section 5.2). The transfer function t_{NL} can be measured by the stimulation (s) in the well defined environment of the MCM tester (see chapter 8):

$$t_{\mathrm{NL}} = t_{\mathrm{NL}}(s) \tag{7.1}$$

The deviation from a linear behavior is rather low such that there exists a global inverse function:

$$s = t_{\mathrm{NL}}^{-1}\Big(\; t_{\mathrm{NL}}(s) \; \Big) \tag{7.2}$$

It is approximated by a straight line:

$$t_{\mathrm{NL}}^{-1}(i) = m_{\mathrm{NL}} \cdot i + c_{\mathrm{NL}} + r_{\mathrm{NL}}(i) \tag{7.3}$$

While m_{NL} is fixed to the result of the straight line fit, the offset c_{NL} has to be shifted such that:

$$r_{\mathrm{NL}}(i) \geq 0 \quad \forall \; i \tag{7.4}$$

This is necessary, because the arithmetics of the nonlinearity correction filter is restricted to positive numbers.

Since the correction $r_{\mathrm{NL}}(i)$ is slightly changing with respect to the input values i, the filter is designed to work on the basis of a mean corrective for groups of 16 input values:

$$r_{\mathrm{NL}}(i) \;\approx\; \frac{1}{4} \cdot LUT_{\mathrm{NL}} \left(\left\lfloor \frac{i}{16} \right\rfloor \right) \tag{7.5}$$

$$\text{with} \qquad LUT_{\mathrm{NL}} \left(\left\lfloor \frac{i}{16} \right\rfloor \right) \;=\; 4 \cdot \Big\langle r_{\mathrm{NL}}(j) \Big\rangle_{\left\lfloor \frac{i}{16} \right\rfloor \leq j < \left\lfloor \frac{i}{16} \right\rfloor + 16} \tag{7.6}$$

The entries of the look-up table are scaled by four because they contain two bits behind the decimal point.

7.2 Pedestal Correction Filter

The pedestal correction filter subtracts the individual pedestal in each of its input channels and adds a common baseline value P_0 (section 5.3). It has to be set such that the effective pedestal P_0^{eff} after the subsequent filter stages is equal to some given number, which is necessary to prevent underflows due to baseline variations:

$$P_0^{\text{eff}} = F_{\text{Xtalk}}^{-1} \circ F_{\text{Tail}}^{-1} \underbrace{\circ F_{\text{Gain}}^{-1} \;\Big|\; P_0}_{P_0^G} \tag{7.7}$$

The gain correction filter contributes an additional constant component while the effect of the last two filter stages corresponds to some scaling only:

$$P_0^{\text{eff}} = \tilde{f}_{\text{Xtalk}}^{-1}(0) \cdot \tilde{f}_{\text{Tail}}^{-1}(0) \cdot P_0^G \tag{7.8}$$

$$\Longleftrightarrow \qquad P_0^G = \tilde{f}_{\text{Xtalk}}(0) \cdot \tilde{f}_{\text{Tail}}(0) \cdot P_0^{\text{eff}} \tag{7.9}$$

with the filters' Fourier components at $\omega = 0$:

$$\tilde{f}_{\text{Tail}}(0) = \frac{1}{\tilde{f}_{\text{Tail}}^{-1}(0)} \;, \qquad \tilde{f}_{\text{Xtalk}}(0) = \frac{1}{\tilde{f}_{\text{Xtalk}}^{-1}(0)} \tag{7.10}$$

The model of the tail (section 5.5) is given by:

$$g(t) = 1_{t \geq 0} \cdot \left(\alpha_L \cdot \lambda_L^t + (1 - \alpha_L) \cdot \lambda_S^t \right) \tag{7.11}$$

Integration over time yields:

$$\tilde{f}_{\text{Tail}}(0) = \int g(t)\; dt = \frac{-\alpha_L}{\ln \lambda_L} - \frac{1 - \alpha_L}{\ln \lambda_S} \tag{7.12}$$

The procedure to obtain the tail parameters α_L, λ_L and λ_S is discussed in section 7.4. The crosstalk suppression filter is represented by a finite filter mask (section 5.6). Thus, the integration of the filter masks leads to:

$$\tilde{f}_{\text{Xtalk}}^{-1}(0) = 1 + 2 \cdot \sum_{i=1}^{5} M_i \tag{7.13}$$

$$\Longleftrightarrow \qquad \tilde{f}_{\text{Xtalk}}(0) = \left(1 + 2 \cdot \sum_{i=1}^{5} M_i \right)^{-1} \tag{7.14}$$

The determination of the crosstalk suppression filter parameters M_i is covered by section 7.5. In order to be able to use the full dynamic range of the gain correction filter $(1 \pm \frac{1}{8})$, the actual baseline value of the pedestal correction filter is set to:

$$P_0 = \frac{8}{9} \cdot P_0^G \tag{7.15}$$

If one of the filter stages, that have been taken into account above, is bypassed, the corresponding factor has to be set to one. Especially if the gain correction filter is inactive, equation 7.15 changes to $P_0 = P_0^G$.

7.3 Gain Correction Filter

It is the goal of the gain correction filter to locally balance gain variations of the sensor system. Therefore the input signals are scaled by some factor $\overline{\beta}$ which is individual for each ADC channel. Its determination is described below. In order to put the signal on the desired baseline, a matching corrective is added. In terms of the symbols in section 7.2, the additive is calculated like:

$$P_0^{\text{Gain}} = P_0^G - \overline{\beta} \cdot P_0 \tag{7.16}$$

The total gain of a channel is the combination of the sensor's gain and the gain correction factor:

$$\beta = \beta^{\text{Sensor}} \cdot \overline{\beta} \tag{7.17}$$

To balance the gain locally, a gain measure is defined. It is based on the scaling of the signal-related part of the amplitude spectrum by the gain.

Figure 7.1 shows five spectra which have been measured on the test stack (see section 10) and include signals from cosmic radiation. Here, the detector gain is approximately constant, while the signals are multiplied by the gain correction filter by factors in the range of $\overline{\beta} \in [0.875,\ 1.125]$. For these measurement, only the pedestal and the gain correction filters were active. The baseline was set to 30, which corresponds to the dominant peak of the spectra. The scaling of the spectra of the actual signals is strongly visible at values above 50. In that range, the spectra can be approximated by a series of exponential functions. In the following, only one exponential function has been considered for model building:

$$A(\beta, x) = 1_{x \geq x_0} \cdot \beta \cdot e^{-\beta \cdot (x - x_0)} \tag{7.18}$$

It is valid for amplitudes larger than x_0, which is sufficiently separated from the baseline. Within the test measurements, a value of 50 has been chosen. This is 20 ADC Counts $< 14\ \sigma$ apart from the baseline, which was set to 30. Above this threshold, even at low occupancy of the detector, the baseline fluctuations can be neglected compared to the actual signal amplitudes. The parameter β corresponds to the total system gain. The spectra are normalized to $\|A(\beta, .)\|_1 = 1$.

For the purpose of gain balancing, spectra like these have to be analyzed for each of the 1.4 million ADC channels of the detector. Therefore, each spectrum is estimated by the proportions of two parts of it:

$$n_1(\beta) = \int_{x_0}^{\tau} A(\beta, x)dx = 1 - e^{-\beta(\tau - x_0)} \tag{7.19}$$

$$n_2(\beta) = \int_{\tau}^{\infty} A(\beta, x)dx = e^{-\beta(\tau - x_0)} \tag{7.20}$$

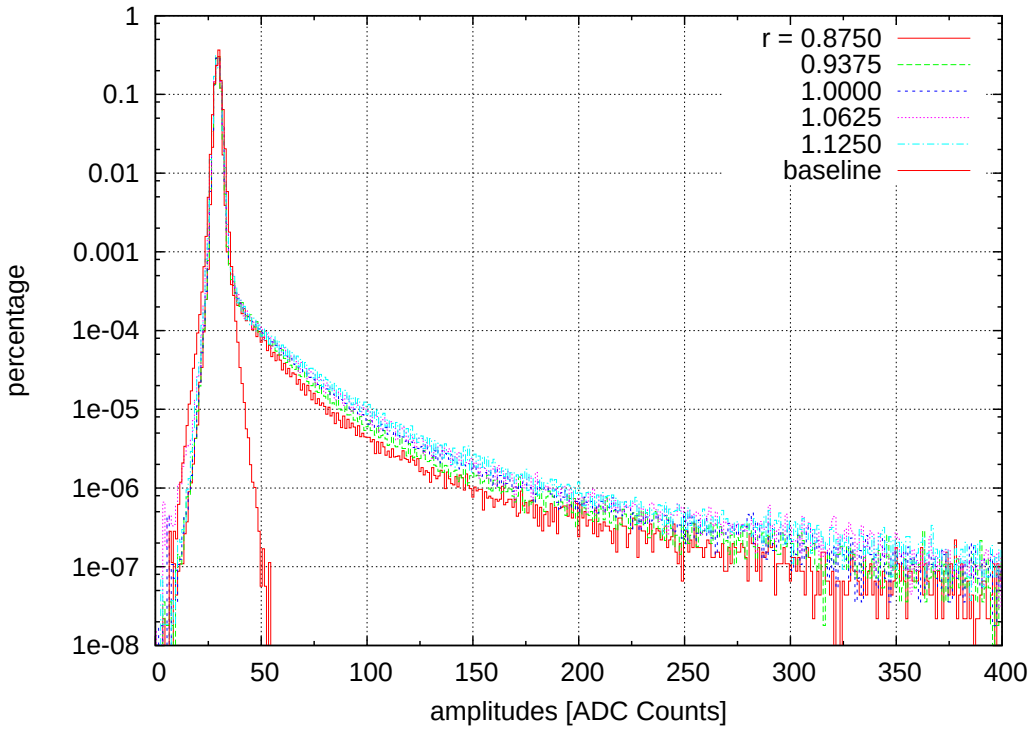

Figure 7.1: amplitude spectra with various gain settings

The corresponding counters are implemented in hardware within each of the filter channels (section 5.4). These two numbers are combined to a gain measure:

$$G(\beta) = n_1(\beta) - n_2(\beta) = 1 - 2 \cdot e^{-\beta(\tau - x_0)} \tag{7.21}$$

Next, the second threshold τ is chosen such that G is mapped to the center of its range at the nominal gain value β_0:

$$G(\beta_0) = 0 \tag{7.22}$$

$$\iff \quad \tau = x_0 + \frac{\log 2}{\beta_0} \tag{7.23}$$

This means, that each choice of the threshold τ corresponds to a choice of a specific nominal gain β_0. Introducing the relative gain $r = \frac{\beta}{\beta_0}$, equation 7.21 changes to:

$$G_0(r) = 1 - 2 \cdot 2^{-r} \tag{7.24}$$

For each channel within a TRAP chip, the gain can be estimated like this. To reduce the complexity of the calculations and to handle variations which are due to the error of the model, an incremental attempt is preferable. Since only

small variations of the gain ($|.| < 12.5\ \%$) are expected locally, equation 7.24 can be linearized around some proper gain reference r_0:

$$\Delta G_0 \approx \left.\frac{\partial G_0}{\partial r}\right|_{r_0} \cdot \Delta r \tag{7.25}$$

$$\text{with} \quad \left.\frac{\partial G_0}{\partial r}\right|_{r_0} = 2 \cdot \log 2 \cdot 2^{-r_0} \tag{7.26}$$

From chip to chip, the variation is in the 30 to 50 % range in addition to a potential slow variation of the whole detector with respect to time. Thus, a local adaption of the gain reference is quite difficult. Fortunately, the choice of G as gain measure allows to fix $r_0 = 1$ and still to have a stable and converging calibration procedure:

With the local gain reference r'_0, the abbreviation $m' = \frac{\partial G_0}{\partial r}|_{r'_0}$ for calibration step i the following equation holds in good approximation:

$$\Delta G_0^{(i)} = m' \cdot \Delta r^{(i)} \tag{7.27}$$

With $m = \frac{\partial G_0}{\partial r}|_{r_0}$, the estimated gain difference is determined by:

$$\Delta \tilde{r}^{(i)} = \frac{1}{m} \Delta G_0^{(i)} = \frac{m'}{m} \Delta r^{(i)} \tag{7.28}$$

The iterative correction strategy yields to a recursion equation:

$$\Delta r^{(i+1)} = \Delta r^{(i)} - \Delta \tilde{r}^{(i)} = \Delta r^{(i)} \left(1 - \frac{m'}{m}\right) \tag{7.29}$$

The corresponding geometric series

$$\Delta r^{(i)} = \Delta r^{(0)} \left(1 - \frac{m'}{m}\right)^i \tag{7.30}$$

converges for all m' with $\left|1 - \frac{m'}{m}\right| < 1$. This is equivalent to $2 > \frac{m'}{m} > 0$. Regarding equation 7.26, it is obvious, that:

$$\frac{\partial G_0}{\partial r} : \ \Re_{>0} \to (0,\ 2 \log 2) \tag{7.31}$$

This means, that choosing $m = \frac{\partial G_0}{\partial r}|_1 = \log 2$, the calibration process always converges. Assuming global variations in the range of $r'_0 \in [1-\epsilon, 1+\epsilon]$, the worst case convergence speed can be estimated by:

$$\frac{\Delta r^{(i+1)}}{\Delta r^{(i)}} \leq 2^{\epsilon} - 1 \tag{7.32}$$

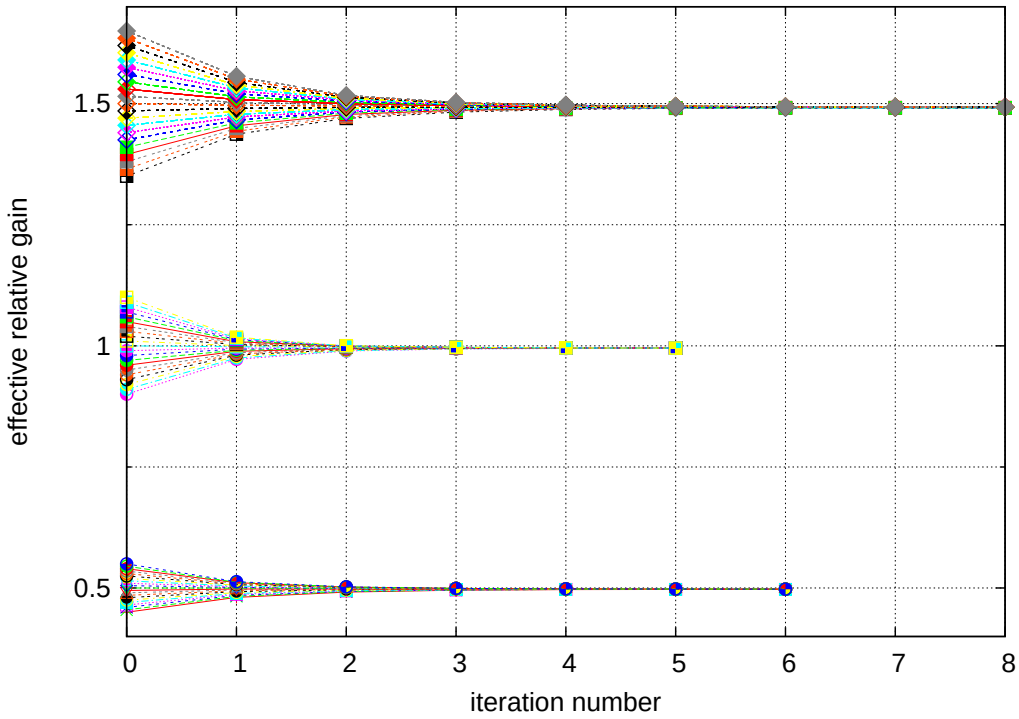

Figure 7.2: examples of convergence of local gain balancing

This corresponds to $\frac{\Delta r^{(i+1)}}{\Delta r^{(i)}} \leq 0.29$ in case of $\epsilon = 0.5$. Figure 7.2 shows three examples of the iterative calibration procedure. Within each of those, the relative gain numbers of the 21 channels are set equidistant in the range of $\pm$ 10 % around a specific local mean value. The three examples represent a balancing process for mean values of the gain of 0.5, 1.0 and 1.5. Obviously, the balancing converges after five iterations in case of a mean relative gain of 1.0 and slightly slower in case of the other mean settings. Here, convergence means, that the estimated local gain variation is below the precision of the TRAP's gain arithmetics, which is $2^{-11} \approx 0.5$ ‰. The process has been simulated by scaling of a representative spectrum and the application of a user program according to the considerations above. With the present test devices in combination to cosmic radiation, the statistics is too poor since the measurement is limited by global gain variations in the order of hours.

In this context, the numerical demand is estimated. The hardware counters of the gain filter N_1, N_2 provide the pure number of counts in the specific part of the spectrum. To get the proportion numbers of the equations 7.19 and 7.20, they are normalized:

$$n_1(\beta) = \frac{N_1}{N_1 + N_2} \ , \quad n_2(\beta) = \frac{N_2}{N_1 + N_2} \tag{7.33}$$

Thus, the gain measure is calculated like:

$$G_0 = \frac{N_1 - N_2}{N_1 + N_2} \tag{7.34}$$

Assuming the two counters N_1, N_2 to be independent and to have a normal distribution, its numerical error is:

$$\Delta G_0 = \frac{2\sqrt{N_1 \cdot N_2}}{(N_1 + N_2)^{\frac{3}{2}}} \tag{7.35}$$

Setting $N_1 = N(\frac{1}{2} - \xi)$ and $N_2 = N(\frac{1}{2} + \xi)$ with the total count N and the deviation from the nominal proportions ξ, the final form is:

$$\Delta G_0 = \frac{\sqrt{1 - 4\xi^2}}{\sqrt{N}} = \frac{\sqrt{1 - G_0^2}}{\sqrt{N}} \tag{7.36}$$

Assuming the relative gain to vary in the range $[1 - \epsilon, 1 + \epsilon]$, this leads to the following worst case estimate of the error of the reconstructed gain spread:

$$\Delta r \leq \frac{\sqrt{2^{1+\epsilon} - 1}}{\log 2} \cdot \sqrt{\frac{2}{N}} \tag{7.37}$$

Or, in terms of required precision:

$$N \leq \frac{2^{1+\epsilon} - 1}{\log^2 2} \cdot \frac{2}{\Delta r^2} \tag{7.38}$$

This means, that e.g. with $\epsilon = 0.5$ and $\Delta r = 0.01$, not more than $N = 76113$ total counts are needed.

7.4 Tail Cancellation Filter

The tail cancellation filter is based on a second order tail model (section 5.5):

$$g(t) = 1_{t \geq 0} \cdot \left(\alpha_L \cdot \lambda_L^t + (1 - \alpha_L) \cdot \lambda_S^t \right) \tag{7.39}$$

This is the impulse response of the tail generating system. Unfortunately, no test devices are integrated in the detector which would allow to inject isolated charge peaks and thereby to determine the parameters of $g(t)$ directly. Typical tracks contain a series of overlapping clusters with a temporal random distribution of the clusters.

To gain deterministic signal shapes, the average of a couple of events has to be taken into account. Figure 7.3 shows the expected average signals along the tracks with and without tail. The primary peak is due to the amplification

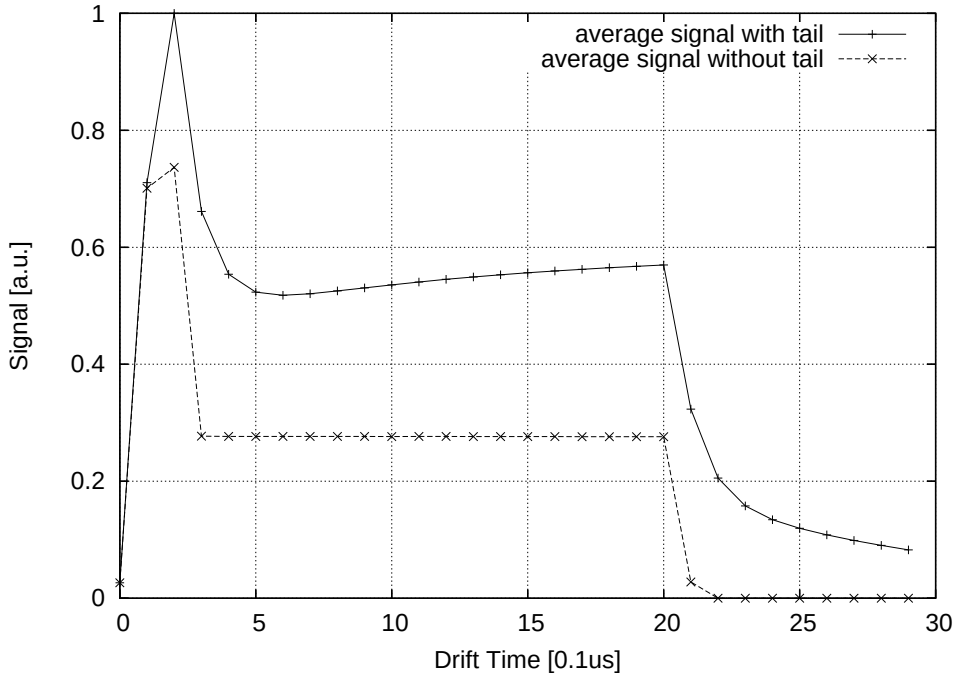

Figure 7.3: available signals from the detector

region, while the plateau corresponds to the drift region with its approximately uniform conversion gain. The drift time depends on detector settings like gas or drift voltage. In addition to the constant mean signal out of the drift region some contributions from the electron's transition radiation photons are expected, but their influence to the mean signal is rather small since the pion background is dominant. This model is the basis of the calibration procedure. Since the tail is a property of the detector chambers, the gained parameters are valid for all kind of clusters.

The parameters are determined by an optimization process. With the mean detector signal $A(t)$ and some guessed parameter set $(\alpha_L, \lambda_L, \lambda_S)$, the filter's mean response is given by:

$$\overline{A}_{\alpha_L,\lambda_L,\lambda_S}(t) = \left(F^{-1}_{\mathrm{Tail}(\alpha_L,\lambda_L,\lambda_S)}|A\right)(t) \tag{7.40}$$

The error measure $M(\alpha_L, \lambda_L, \lambda_S)$ is defined as the ratio between remaining signal components after the drift time T_D and the mean signal before the drift time:

$$M(\alpha_L, \lambda_L, \lambda_S) = \frac{\sum_{t=T_D}^{T_E} \overline{A}^2_{\alpha_L,\lambda_L,\lambda_S}(t)}{\frac{1}{T_D}\sum_{t=0}^{T_D-1} \overline{A}_{\alpha_L,\lambda_L,\lambda_S}(t)} \tag{7.41}$$

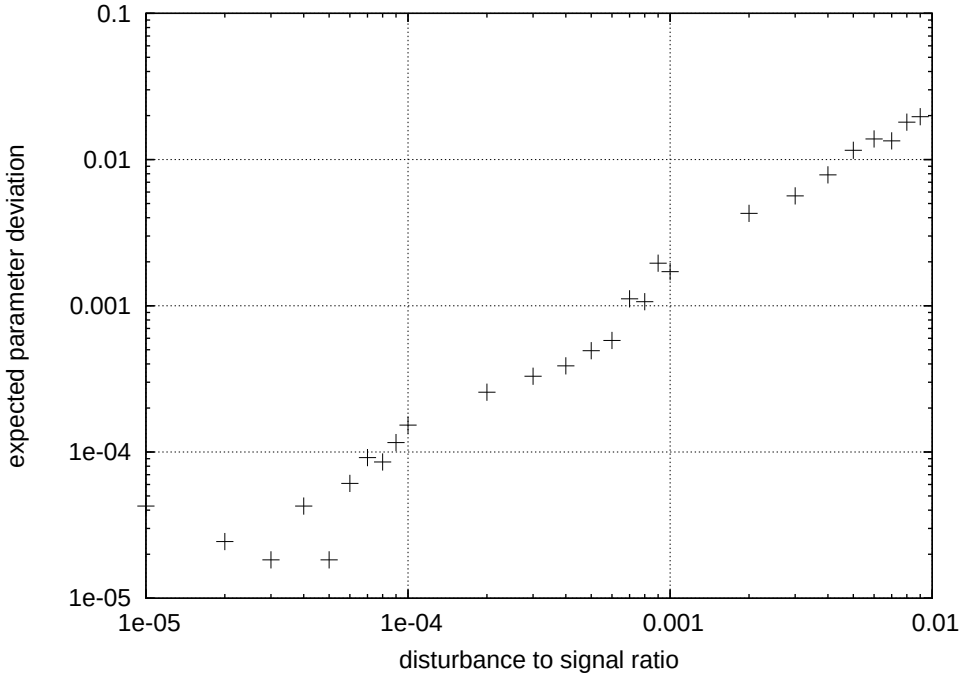

Figure 7.4: stability analysis of the tail parameter reconstruction

The recorded signal is assumed to start at $t = 0$ and to end at $t = T_E$. The idea of the error measure is to minimize the course of the signal after drift time in relation to the wanted signal.

The tail parameters are determined by the minimization of the error measure with respect to the tail parameters. Since the phase space is finite, the support is an integer grid and a couple of local minima are expected, the minimization has been implemented as an adaptive phase space scan. This means, that the phase space is scanned on a coarse grid. Next, it is analyzed on a refined grid in a subspace around the best candidate, that has been found on the coarse grid. This procedure is iterated until the best parameter set is found on the finest grid.

To estimate the error of the present parameter calibration algorithm, a test environment has been developed: The expected mean signal without tail is convoluted with a tail generating filter of known parameters. Next, the resulting signal is disturbed by Gaussian noise of given noise-to-signal ratio. Then, it is processed by the calibration program. Finally, the parameters, which have been determined by the calibration tool, are compared to those, that have been used for the initial convolution.

The test parameters have been taken out of the neighborhood of a typical

parameter set as it has been observed during test operation. Figure 7.4 shows the mean deviation of initial and reconstructed parameters for such a set of parameters. In order to get rid of random effects due to the signal disturbance, the average of ten runs has been taken. The x axis represents the range, in which the disturbance to signal ratio has been varied while the y axis shows the observed mean parameter deviation. Since the functional relation between expected deviation and signal disturbance is of first order, the method is stable. Further, the requirement on the mean signal is a function of the desired precision of the parameters. For example, if one expects signal amplitudes in the order of 100 and wants to use the filter's full precision of $\frac{1}{4}$, a parameter precision of $\frac{1}{400}$ will be sufficient. This corresponds to a disturbance to signal ratio of about 0.001. The disturbance itself is related to the error of the mean of the signal. Thus, the calculation needs to be based on 1000 tracks at least.

7.5 Crosstalk Suppression Filter

The crosstalk generating process is described by a two-dimensional filter matrix C. Because next-neighbor interaction is significant only, its support is limited to the range of $[-1, +1]$ in the direction of the pad channels. In time direction, it is sufficient to limit the parameterization to the range of $[-2, +2]$. The inversion process of the crosstalk generation to gain the filter matrix M is discussed in [mg0, p. 51]:

Due to the too low sampling frequency, the potential suppression of crosstalk effects by the filter is rather low. Here, the corresponding filter entries are in the order of less than $\frac{1}{10}$. Thus, it is sufficient to perform a first order inversion only:

$$M = \underline{1} - (C - \underline{1}) = 2 \cdot \underline{1} - C \tag{7.42}$$

The crosstalk matrix C can be measured by the usage of a test pulser within the PASA input ports. Hereby, charge pulses with selectable phase relative to the data acquisition system can be injected directly on the pad. Figure 7.5 shows an example. The crosstalk amplitude is about 3 % of the height of the central peak. The coupling to the overnext neighbor is on the per mill level and can be neglected. To locate the crosstalk coupling in the system, the measurement has been repeated with disconnected input pads of the PASA (fig. 7.6). The relative amplitude of the crosstalk signal is reduced to 0.2 %. This is less than the noise level at typical signal height.

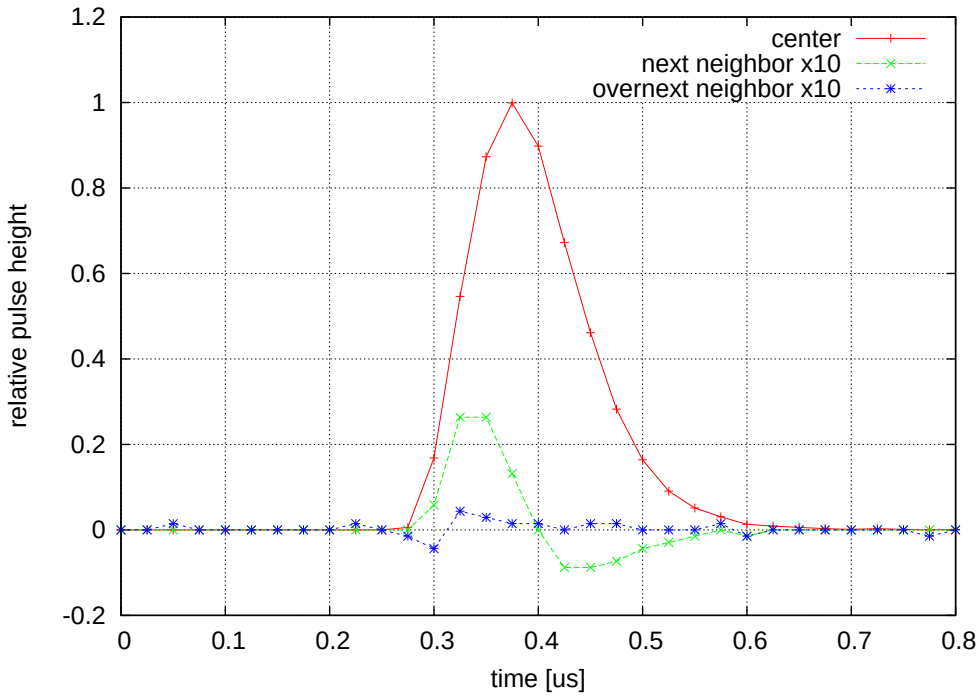

Figure 7.5: crosstalk signal, measured with PASA pulser on TRD chamber

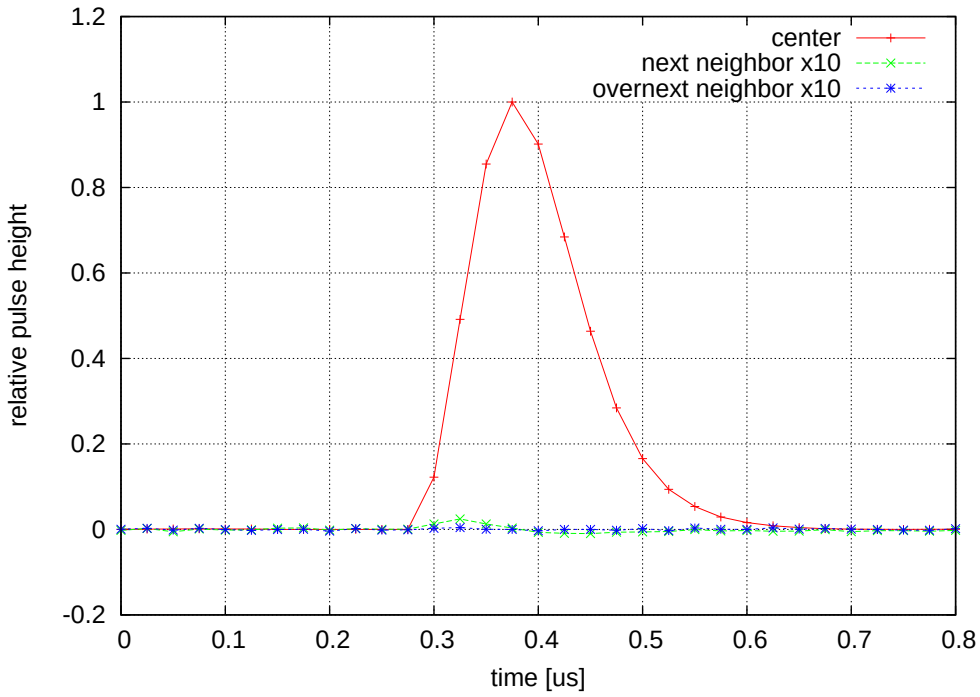

Figure 7.6: crosstalk signal, measured with PASA pulser with disconnected input

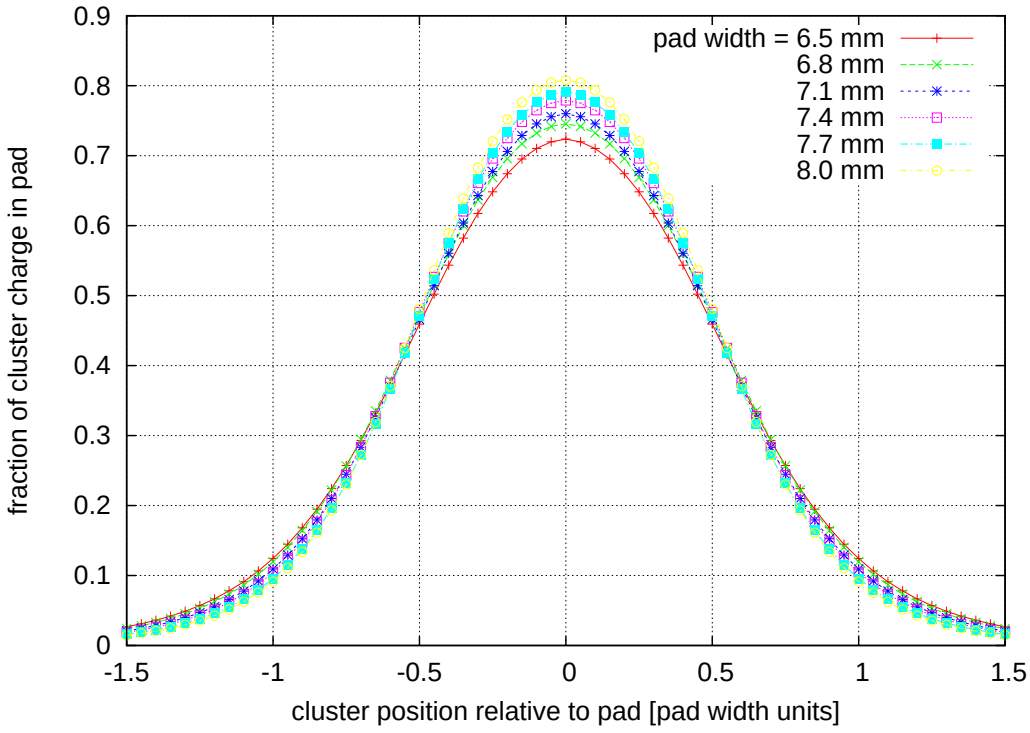

Figure 7.7: Pad Response Function of the six pad geometries of the TRD [ac0]

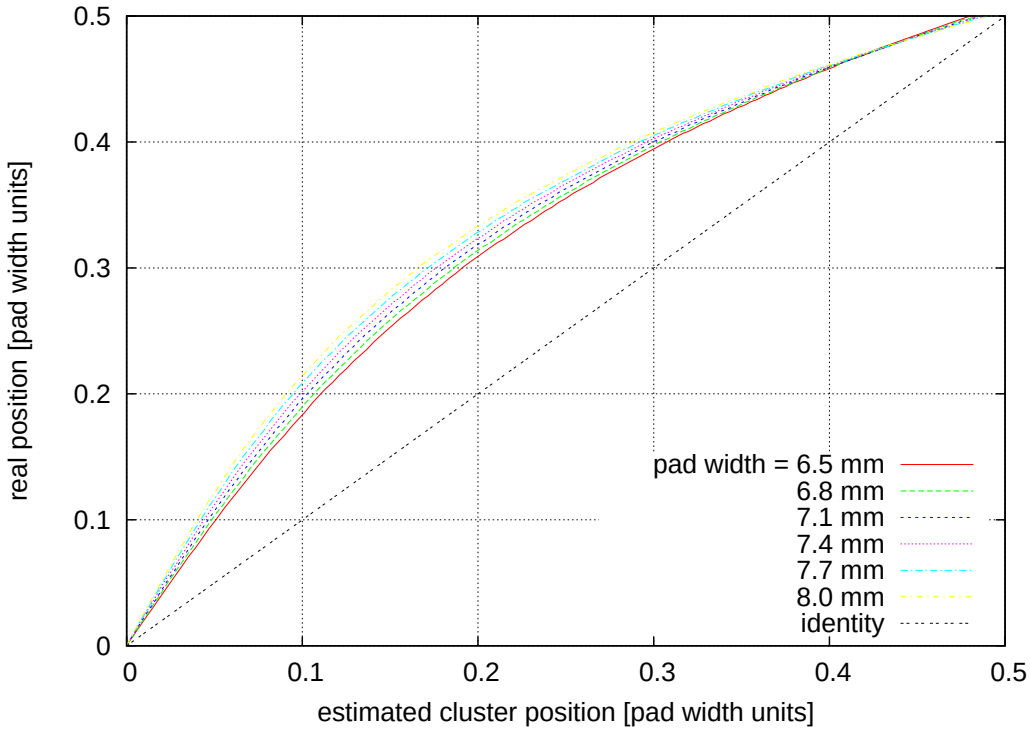

Figure 7.8: real cluster position as a function of the estimation

7.6 Position Correction Look-up Table of the Preprocessor

Within the detector chambers, charge clusters are amplified around the anode wire and detected via charge fluctuations on the nearby cathode pads. The amount of charge, that is collected by a certain pad, is a function of the position of the cluster relative to the pad, the so-called pad response function PRF(y) (fig. 7.7). It is measured in test chambers with well-defined localization the clusters. The values are normalized to the total charge which is visible within one pad row. Because of the symmetry of PRF(y), the following considerations are restricted to positive cluster displacements y.

Using the pad response function, the unsigned position estimate y^{est} of the preprocessor can be expressed as a function of the actual position:

$$|y^{\text{est}}|(y) = \frac{1}{2} \cdot \frac{\text{PRF}(y-1) - \text{PRF}(y+1)}{\text{PRF}(y)} \tag{7.43}$$

Due to the shape of the PRF, the mapping is one-to-one and thus, the function is invertible (see fig. 7.8). The correction values correspond to the difference between the estimated position and the real one:

$$LUT_{\text{PC}}(|y^{est}|) = y(|y^{\text{est}}|) - |y^{\text{est}}| \tag{7.44}$$

7.7 Cluster Quality Measure Threshold of the Preprocessor

To separate isolated clusters from shared clusters, a quality measure QM is foreseen to be used (see eqn. 6.30):

$$\text{QM} = \frac{L \cdot R}{C^2} \tag{7.45}$$

If the PRF had a Gaussian shape, this would result in a mapping to a single value. Figure 7.9 shows QM as a function of the cluster position. One can see, that positions, which are closer to the center of the pads, lead to lower values of QM, while the maximum values are reached at the border between two pads. In order not to suppress valid clusters, the threshold for shared cluster discrimination should be chosen there. The maximum values for all six pad plane types QM_{max} are presented in table 7.1. In addition, the error of QM rises to the border of two pads, too. There, it is dominated by the relative error of the contribution of the minimum of L and R. This leads to a relative error of QM of:

$$\frac{\Delta\text{QM}}{\text{QM}} = \frac{\sigma}{A \cdot \min(L, R)} \tag{7.46}$$

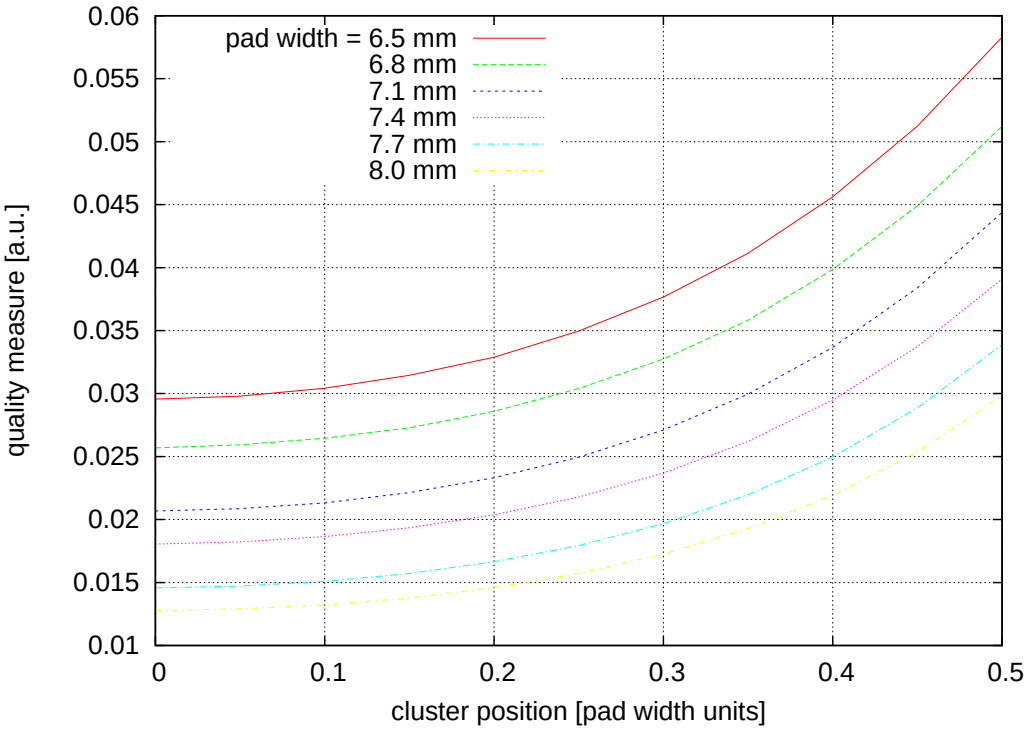

Figure 7.9: quality measure for ideal clusters as a function of position

Here, A represents the minimum total cluster charge and σ the noise level. In order to accept cluster with one sigma fluctuation, the threshold T_Q has to be incremented due to the expected error:

$$T_Q = \mathrm{QM}_{\max} \cdot \left(1 + \frac{\sigma}{A \cdot \min(L, R)}\right) \tag{7.47}$$

Table 7.1 shows example calculations with A=40 ADC Counts and σ=1.0 ADC Counts.

pad width [mm]	ideal $\mathrm{QM}_{\max}$	$\min(L, R)$	threshold
6.5	0.058	0.027	0.113
6.8	0.051	0.024	0.104
7.1	0.044	0.021	0.098
7.4	0.039	0.019	0.092
7.7	0.034	0.016	0.087
8.0	0.030	0.014	0.082

Table 7.1: ideal quality measure maxima and thresholds

7.8 Tracklet Program Parameters

The tracklet program uses various parameters to transform and to analyze the results of the tracklet merging process. This incorporates a mapping of mean cluster charge of a tracklet to electron probability, coordinate transformations of offset and slope and cuts due to the tracklet stiffness (see section 6.3).

The electron probability determination by partial charge accumulation along the tracklets ($Q^{(0)}$, $Q^{(1)}$) in combination to the hit count (N) is still under investigation. A classical solution is based on the mean charge per hit $q = \frac{Q^{(1)}}{N}$ (see p. 10, fig. 2.8). With the probability distribution of ionization by electrons $p_e(q)$, by pions $p_\pi(q)$ and the production rates of those particles within the experiment p_e^{Prod}, p_π^{Prod}, the probability that a tracklet of a definite mean charge per hit q has been caused by an electron, is given by:

$$P_{\text{electron}}(q) = \frac{p_e^{\text{Prod}} \cdot p_e(q)}{p_e^{\text{Prod}} \cdot p_e(q) + p_\pi^{\text{Prod}} \cdot p_\pi(q)} \tag{7.48}$$

This equation corresponds to a best guess on the particle identity.

The calibration of the other tracklet program parameters has been discussed in detail in [jc0, p. 37]. Thus, only a short summary of the basic ideas and results is given below.

The tracklet offset is measured in pad width units, while the GTU expects the position to be measured in multiples of a granularity unit $[\tilde{\hat{y}}] = 160$ µm. The scaling factor of the corresponding transformation (eqn. 6.51) is given by the ration of the pad width and this granularity unit:

$$m^{(\hat{y})} = \frac{\Delta y_{\text{Pad}}}{[\tilde{\hat{y}}]} \tag{7.49}$$

The additive within the same transformation is determined by the position of the MCM on the chamber. If the regarded MCM is the nth one, counted from left to right and from zero to seven on a chamber (upwards in fig. 2.16), the additive is given by:

$$c^{(\hat{y})} = \Big((n-4) \cdot 18 - 1\Big) \frac{\Delta y_{\text{Pad}}}{[\tilde{\hat{y}}]} \tag{7.50}$$

The 18 is due to the fact, that three out of the 21 channels within one TRAP chip are shared with the neighbor MCM. The subtraction of one corresponds to the extra one of equation 6.48.

After the tracklet merging process, the slope is available in units of pad width per sample. This needs to be transformed to the deflection with respect to

the whole chamber in multiples of a granularity unit $[\tilde{\vartheta}] = 140$ µm. With the maximum number of samples per drift time $\hat{N}$, the corresponding scaling factor (eqn. 6.50) is given by:

$$m^{(\vartheta)} = \frac{\hat{N} \cdot \Delta y_{\mathrm{Pad}}}{[\tilde{\vartheta}]} \tag{7.51}$$

The additive within the transformation consists of two components:

$$c^{(\vartheta)} = \left(\tan(\Psi_L) \cdot \hat{x} + \frac{z_{\mathrm{row}} \cdot \hat{x}}{x_{\mathrm{row}}} \cdot \tan(\beta_{\mathrm{tilt}}) \right) [\tilde{\vartheta}]^{-1} \tag{7.52}$$

Here, $\hat{x} = 3$ cm is the drift length of the detector chambers, while x_{row}, z_{row} are the coordinates of the treated pad row with respect to the GTU's coordinate system of the affected detector stack.

The first term corresponds to the correction of the Lorentz deflection $\tan(\Psi_L) = \frac{e\tau B}{m}$ with electron charge e and mass m, the local magnetic field B and the mean time between two collisions of gas molecules τ. This is due to the movement of the electrons within the drift volume which is not exactly perpendicular to the pad plane but deflected by the local magnetic field. At present, a deflection of $\Psi_L = 7^\circ$ is expected.

The second term represents a correction which is necessary because the pads are slightly tilted with respect to the pad direction, $\beta_{tilt} = \pm 2^\circ$. The correction implies the assumption, that the tracks are originated at the primary vertex. The tilting with alternating orientation from layer to layer improves z resolution for offline analysis.

It is the task of the TRD trigger to search for high momentum electrons which corresponds to sufficiently stiff tracks. With the position of the tracklet $(x_{\mathrm{tl}}, y_{\mathrm{tl}}^{(i)})$ and the minimum transverse momentum p_T^{min}, the maximum deflection from the direction to the primary vertex is given by:

$$\alpha_{\mathrm{max}}^{(i)} = \arcsin \left(\frac{e \cdot B \cdot \sqrt{x_{\mathrm{tl}}^2 + \left(y_{\mathrm{tl}}^{(i)} \right)^2}}{2 \cdot p_T^{\mathrm{min}}} \right) \tag{7.53}$$

The direction to the vertex is given by $(x_{\mathrm{tl}}, y_{\mathrm{tl}}^{(i)})$ directly. Finally, the local limits for the deflection are given by:

$$T_i^{\mathrm{min}} = \hat{x} \cdot \tan \left(\arctan \left(\frac{y_{\mathrm{tl}}^{(i)}}{x_{\mathrm{tl}}} \right) - \alpha_{\mathrm{max}}^{(i)} \right) \tag{7.54}$$

$$T_i^{\mathrm{max}} = \hat{x} \cdot \tan \left(\arctan \left(\frac{y_{\mathrm{tl}}^{(i)}}{x_{\mathrm{tl}}} \right) + \alpha_{\mathrm{max}}^{(i)} \right) \tag{7.55}$$

While the coordinate x_{tl} is fixed for a given pad row, the position within a pad row $y_{\mathrm{tl}}^{(i)}$ depends on the location of the tracklet within that row, even within a TRAP chip. Fortunately, it is sufficient to provide the threshold at the granularity of channel numbers i or pads, which were affected by a tracklet.

8

Functional Tests

8.1 Setup

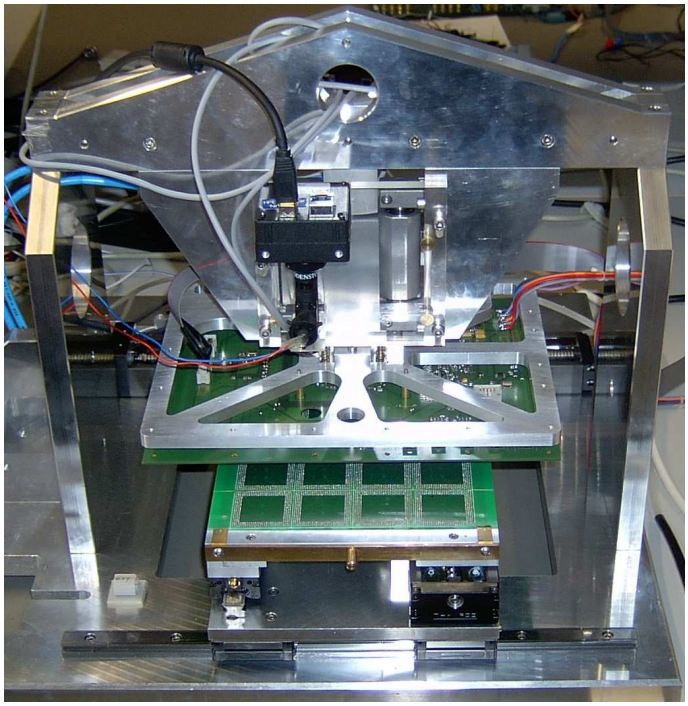

Figure 8.1: MCM tester

The TRD will incorporate more that 65,000 MCMs. To guarantee the functionality of the system, a test procedure of all MCM's features has been developed. The test consists of various modules focusing on internal devices like instruction memory, data memory, event buffer or the configuration memory and on separable functional parts like arithmetics within the CPUs or measurement of analog

test signals. The test is applied by an automatic apparatus, the MCM tester (fig. 8.1). The MCM are classified due to the test results which are stored in a data base. A detailed discussion of the test environment is given in [ff0].

The tests are separated in internal and external tests. The external tests check basic functionality like pin connectivity, short circuits or input resistance of the input ports. The internal tests are based on the functionality of the CPUs. They are executing several test programs and storing the test results. Afterwards, they are read externally. Using the global configuration bus of the TRAP chip, all internal devices of the chip are accessible by the CPUs. The author has focussed on the development of several internal tests, especially testing of the filter functionality.

8.2 Overall Method

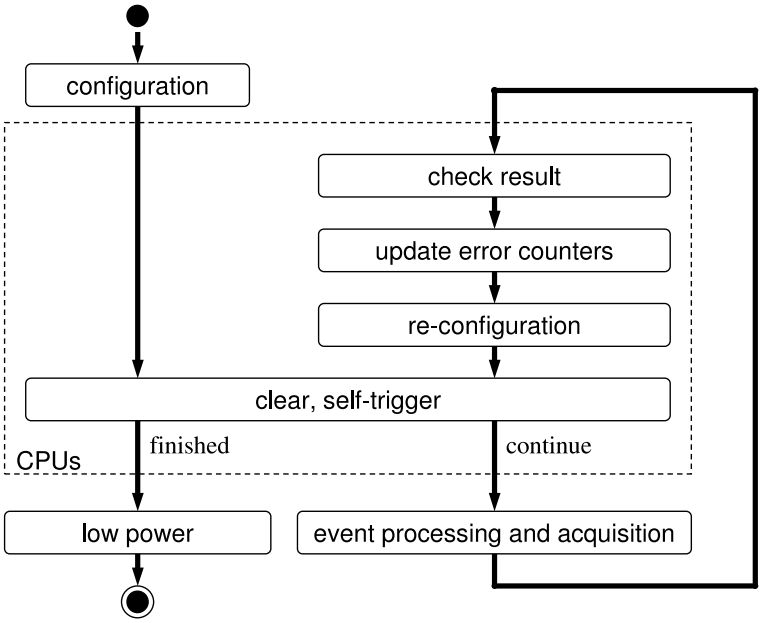

Figure 8.2: test procedure of the internal filter tests

The data processing and acquisition system of the TRAP chip is designed event-based. To digitally test its behavior, it has been foreseen in the hardware design to change the input device from the set of ADCs to the event buffer. Therefore, a predefined series of input values is stored there. The sequence is controlled by the acquisition state machine (section 3.7). Based on the input values as well as the settings of the filter, acquisition and processing system, the CPUs calculate the expected results.

The detailed test procedure is shown in figure 8.2. After the configuration of the chip, the CPUs are started in the clear state (see section 3.3) in order to prepare event acquisition. There, a trigger is generated internally such that event acquisition is started. Subsequently, the CPUs are started again, checking the filtered event buffer entries and the state registers. After updating the error counter, the configuration of the chip is changed such that a maximum coverage of valid input values and filter settings can be achieved. Afterwards, the next test cycle is starting by traversing the clear state.

The whole procedure ends after all desired combinations of input values and settings have been applied. In that case, the chip changes from clear state to low power state without any further pre-trigger. This state transition indicates the end of operation of the current test module.

8.3 Test Modules

The testing of the digital filter is divided into 18 modules. Since the different filter stages can be bypassed, the test of the whole filter can be separated into various tests of its components (section 5.1):

The input delay chain is tested in combination with the shift register of the crosstalk filter. All possible delay settings are tried on one set of input values.

The nonlinearity correction filter contains an adder and a look-up table (LUT) leading to two independent test modules. The addressing of the LUT is tested by input values covering the whole input range of the ADCs in combination with unique entries of the LUT. The entries of the LUT as well as the adder are tested by scanning the whole input range in combination with the whole range of possible corrections. Here, all LUT entries are set to common values.

The pedestal correction filter determines the actual pedestal value of each channel. The correction includes the subtraction of that value and an addition of a programmable target pedestal. The test is divided into three parts. The adder is tested in a first test module by covering the full input range and the full range of the additive in equilibrium state of the filter. The functionality of the pedestal follower circuit is checked by four test modules, one for each time constant of the filter. They are tracing the entries of control registers containing the actual reconstructed pedestal value. The subtracter is tested via another four test modules by observation of the output as a function of time in non-equilibrium state of the filter.

The gain correction filter incorporates a set of counters, a multiplier and an adder in each channel to estimate and to correct channel-to-channel gain variation, leading to three independent test modules. The adder is tested by keeping the multiplication factor constant, covering the whole input range and the whole range of the programmable additive. The multiplier is tested by keeping the additive constant and scanning the whole range of input values and multiplication

factors. The functionality of the counters is checked by scanning the whole input range of the filter in combination with selected threshold values of the counters.

The test of the tail cancellation filter is separated in three modules, according to the number of free parameters of the filter. Theses are the relative weight of the exponentials as well as the two time constants. Within each of those modules, four different test patterns are applied to the filter and one parameter is scanning its whole range of possible settings. The other two parameters are kept constant.

9

Power Consumption

9.1 Overview

One of the important arguments to design an ASIC is to accomplish a given task with a very efficient ratio between computational effort and power consumption of the considered device. The TRAP chip is integrated within the detector in great number. Here, the power supply is quite difficult due to very strict geometrical conditions and material requirements. This conditions the second problem which is influenced by the power consumption: A cooling system has to handle that the consumed power is dissipated as heat in its environment.

To consume as little power as possible, various strategies have been applied to the design of the TRAP chip. A target technology with a relatively small feature size has been chosen (UMC 0.18µm) and the clock signals of almost all major components are gated individually.

The main benefit of the gated clocks is due to the switching of the CPUs' operation. While the continuous operation of the global state machine, filter and event buffer consumes a maximum of 141 mA at the core voltage of 1.8 V, all four CPUs increase that current by up to 315 mA in total. Because the CPUs are clocked for tracklet merging and raw data readout only, the average power consumption is increased proportionally to the trigger and readout rate (section 9.3).

The MCMs are supplied with four voltages. The analog devices are supplied independent from the digital circuits. While the power consumption of the PASA chip is almost constant, the consumption of the ADCs within the TRAP chip can be adjusted by choosing different internal buffer strengths. Here, the settings have been chosen at minimum power consumption without significant loss of signal quality [dm0]. Table 9.1 summarizes the measurements of the average power consumption of un-triggered MCMs.

A major part of the power consumption at the core voltage is caused by those devices, which have to be always operational like the global state machine or the

device	voltage	usage	current [mA]	power [mW]
PASA	3.3 V	analog	91.9	303
TRAP ADC & I/O	3.3 V	analog/digital	23.8	78
TRAP ADC	1.8 V	analog	99.2	179
TRAP core	1.8 V	digital	85.8	154

Table 9.1: average power consumption of an un-triggered MCM

configuration network. Their contribution of 49.9 mA or 89.8 mW is quite constant, while the second important contribution, which is due to the digital filter, strongly depends on environmental conditions and parameter settings. Table 9.2 shows the power consumption of the filter with a variation of active components, measured with input signals with a disturbance of 1.2 ADC Counts. The present parameter settings are based on a systematic variation of those parameters, which influence significantly the power consumption of the filter (see section 9.2).

active filter circuits	current [mA]		power [mW]	
	per MCM	per channel	per MCM	per channel
clock tree	6.4	0.30	11.5	0.55
standby	11.5	0.55	20.7	0.99
P	11.8	0.56	21.2	1.01
P T	26.7	1.27	48.1	2.29
P G T	34.1	1.63	61.6	2.93
N P G T	36.4	1.73	65.5	3.12
P T C	77.1	3.67	138.8	6.61
P G T C	90.1	4.29	162.2	7.72
N P G T C	91.1	4.34	164.0	7.81

Table 9.2: power consumption of the digital filter at the core voltage depending on the usage of different filter components (nonlinearity correction N, pedestal correction P, gain correction G, tail cancellation T, crosstalk suppression C)

All filter components as well as the preprocessor are connected asynchronously to each other. Thus, the consumption of the whole filter is bigger than the sum of the consumptions of each component. This is because the power consumption of a gate is determined by the output capacity, which it has to drive, and the toggle rate of its output value. The number of switching operations of a gate increases with the number of gates, that are located between it and the previous registers.

Non-pipelined multipliers are the largest basic components of the filter. So, the power consumption is dominated by the filter modules, which are using multipliers. The gain correction filter incorporates one multiplier. The power increment due to its usage is significantly larger than the increment, which is due

to the usage of the pedestal or the nonlinearity correction filter. The tail cancellation filter incorporates three multipliers, which leads to the correspondingly higher current increment. Their output is connected synchronously to further parts of the filter and preprocessor, which lowers the effect. The crosstalk suppression filter contains five multipliers with an asynchronous connection to the output.

In addition to the average power consumption due to signal variations, there are short power increments, which are caused by the events and the corresponding processing (section 9.3).

9.2 Variation of the Crucial Parameters

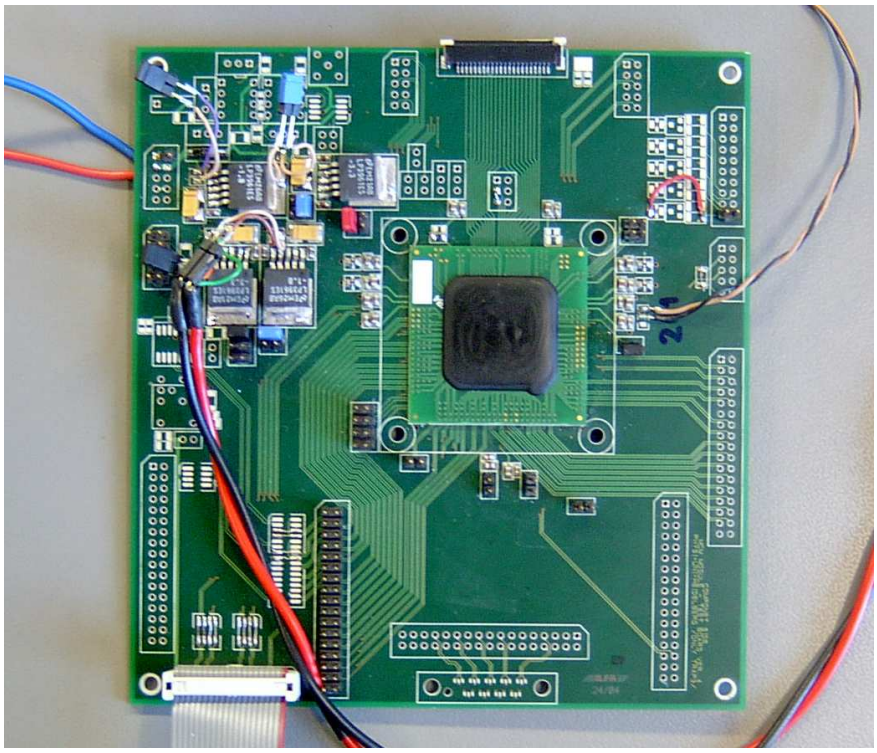

Figure 9.1: setup for parameter optimization

To optimize the power consumption of the TRAP chip, a setup is used, which allows to control all influencing quantities. Therefore, the chip is soldered on a printed circuit board (PCB) as it is used in the MCM tester, too (fig. 9.1). Its SCSN interface, trigger and clock input are accessible by a ribbon cable which is connected to an ACEX card [acex] within a personal computer (PC). This board contains a Field Programmable Gate Array (FPGA) which is connected to the

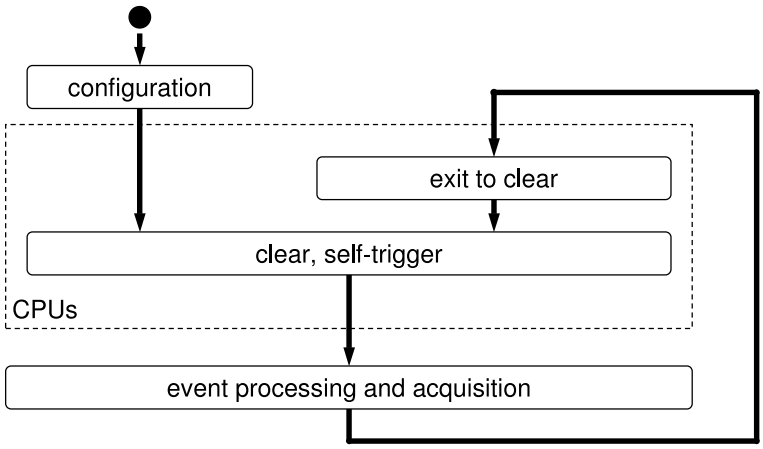

Figure 9.2: event loop for filter and preprocessor stimulation

computer's PCI bus. The other connectors are used for power supply and for the measurement of the current at each individual voltage of the examined MCM. A series of configurations is applied by a shell script which logs the current settings and the measured current. The current is available through the General Purpose Interface Bus (GPIB, IEEE 488) of the multimeter.

The circuits are stimulated by data from the event buffer. The data has been taken on one of the first TRD chambers. It is disturbed with an RMS of 2.7 ADC Counts. In order to measure average currents, the stimulation has to take place continuously. This is achieved by a loop, which includes minor CPU activity (fig. 9.2). The CPUs are started immediately after the event processing and acquisition. Via the GSM's clear state a trigger is initiated by one of the CPUs. Subsequently, the next event is started. The time between two events is about one sample while the duration of an event is 64 samples. The gained power data has to be calibrated for the use case. Especially it has to be corrected for the power consumption which is due to the CPU activity.

The digital filter and the preprocessor are connected asynchronously to each other. Therefore, they have to be seen as a common device in terms of power. The first component of the preprocessor is the hit detection unit. At proper settings of the hit threshold, which correspond to the use case, the preprocessor can be isolated from the filter. If none of the channels are selected for tracking, a constant zero is multiplexed to the subsequent parts of the preprocessor. Figure 9.3 illustrates the effect.

The coarse structure is a smoothed step function from about 1.5 mA per channel to approximately 4.1 mA per channel. The lower values correspond to an inactive preprocessor while the higher values correspond to a setting, at which all channels are declared to contain hits. The smoothing of the step is caused by

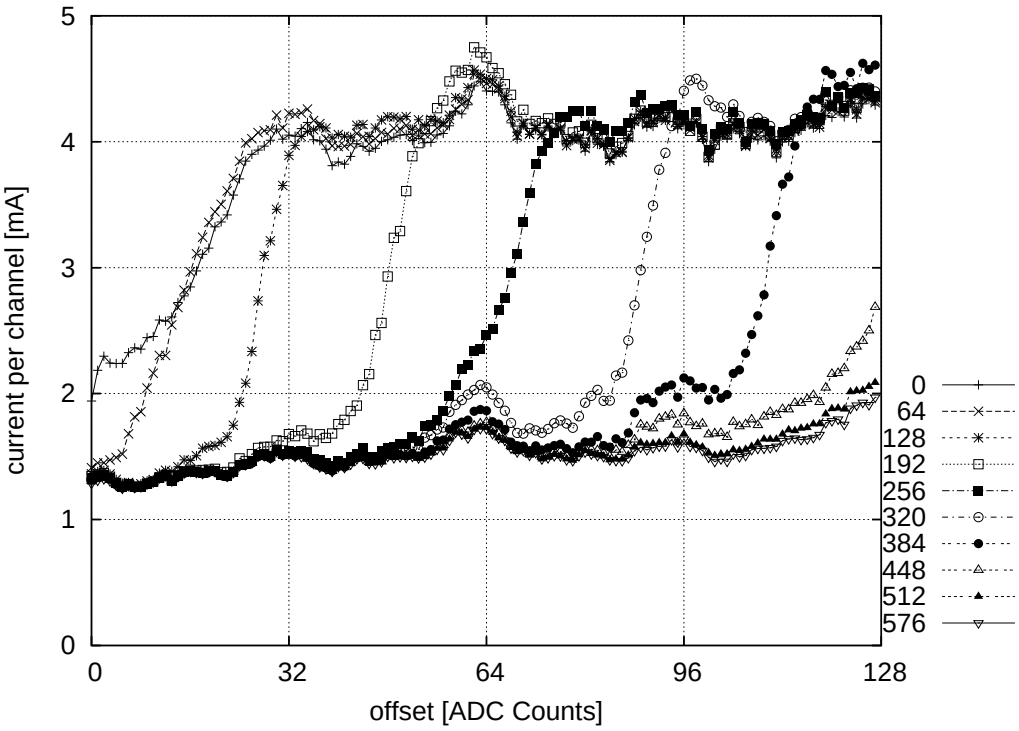

Figure 9.3: dependence of the power consumption of filter and preprocessor from different offset settings at various choices of the hit threshold of the preprocessor

perturbation itself, which leads to some uncertainty of the baseline value.

A fine structure is recognizable in those parts of the graphs, where the step function is almost constant. Here, the binary encoding within the chip is visible. The closer the baseline is to a multiple of a power of two, the more the power consumption increases. This is because all bits up to that, which corresponds to the involved power of two are flipping. While the peaks are quite small at multiples of 16, they are bigger at multiples of 32 or even 64.

Another important parameter is the noise performance of the measurement system. While first prototype measurements contained disturbances of a RMS of 2.7 ADC Counts or the prototype stack of 1.4 ADC Counts, improvements of the power supply and grounding scheme suppressed it down to a level of about 1.0 ADC Counts [ko0]. This has multiple advantages: The signal to noise ratio is improved and the power consumption is decreased. In figure 9.4 the power consumption of filter and preprocessor is shown as a function of the RMS level of the input data. The hit threshold has been set to a critical value, which leads to an overall increment of the power consumption. The figure shows for four different combinations of used filter components, how the consumption is influenced.

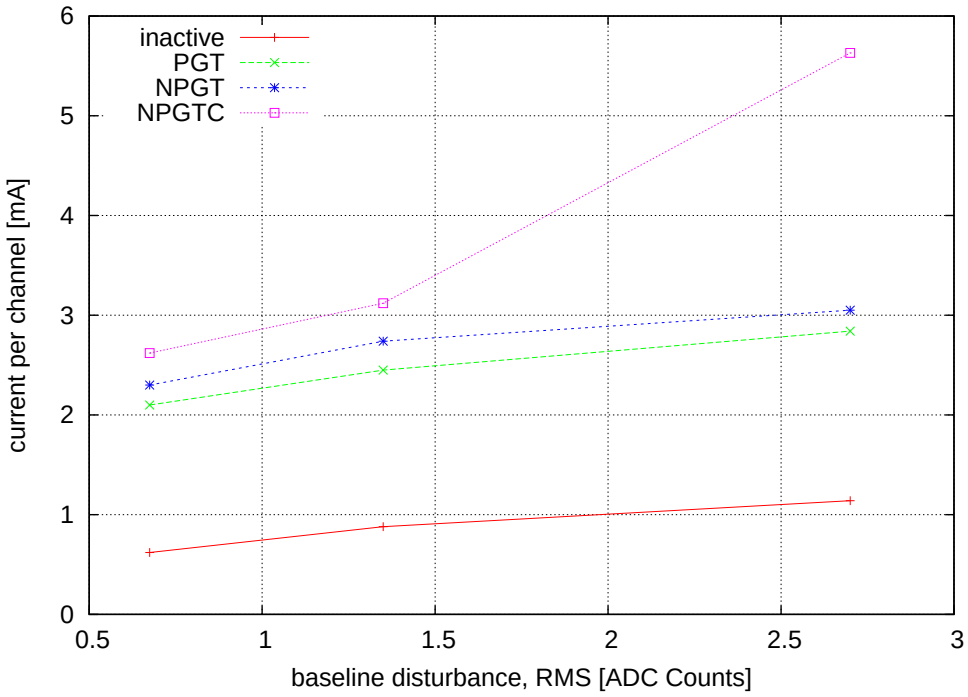

Figure 9.4: variation of the baseline disturbance of the input data at four different sets of used filter components

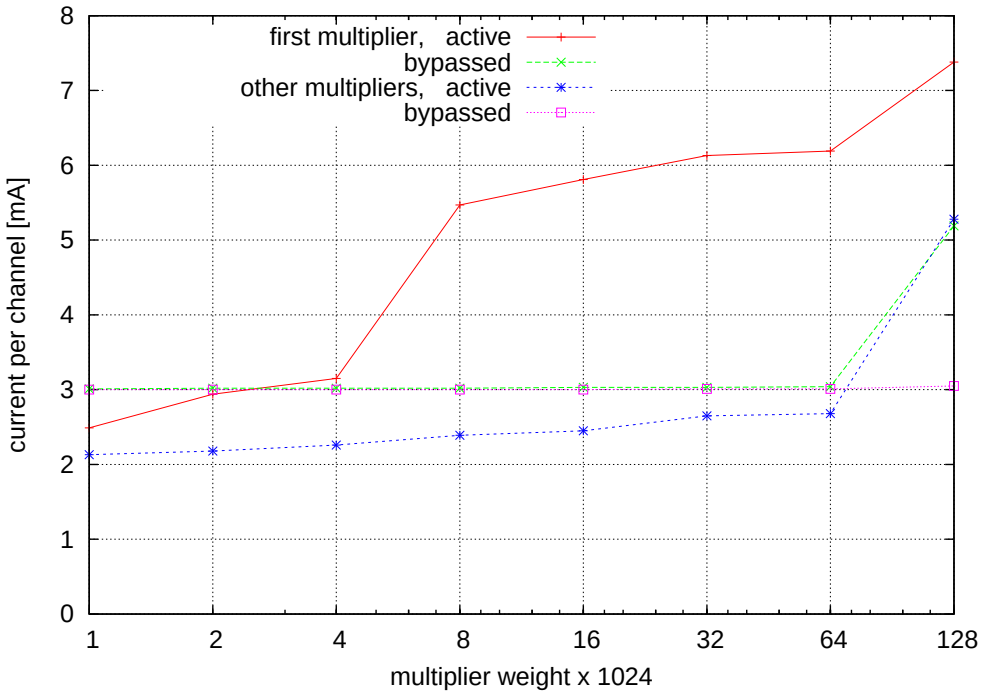

Figure 9.5: variation of single bits of the crosstalk multipliers

The biggest basic components of the filters are multipliers. They have the most important influence on the overall power consumption. Thus the variation of single bits of the weight constants of the crosstalk suppression filter are a convenient method for the analysis of their contribution. The weight constants are restricted to the range of $[-0.125, +0.125)$ and are treated in the two's-complement representation. The noise affects lower bits of the input with a higher frequency than higher bits. Thus, the power consumption is dominated by the bit flips of the lower bits. For the weights constants the upper bits are more important because higher bits do more affect the result than the lower bits (fig. 9.6). The increment due to the last bit $\frac{2^8}{1024}$ is the most obvious one. It is caused by the sign extension of that bit.

Furthermore, the contributions of the first multiplier strongly differ from those of the others. This is because the first multiplier is connected asynchronously to the preceding logic, while the other multipliers get their input values out of the shift register. This effect is still observable even if the filter is bypassed. Finally, the measurement shows that the overall power consumption with an active crosstalk suppression filter, whose filter constants are set to zero, is less than that with a bypassed crosstalk suppression filter. This is another hint to use pipelining in order to save power.

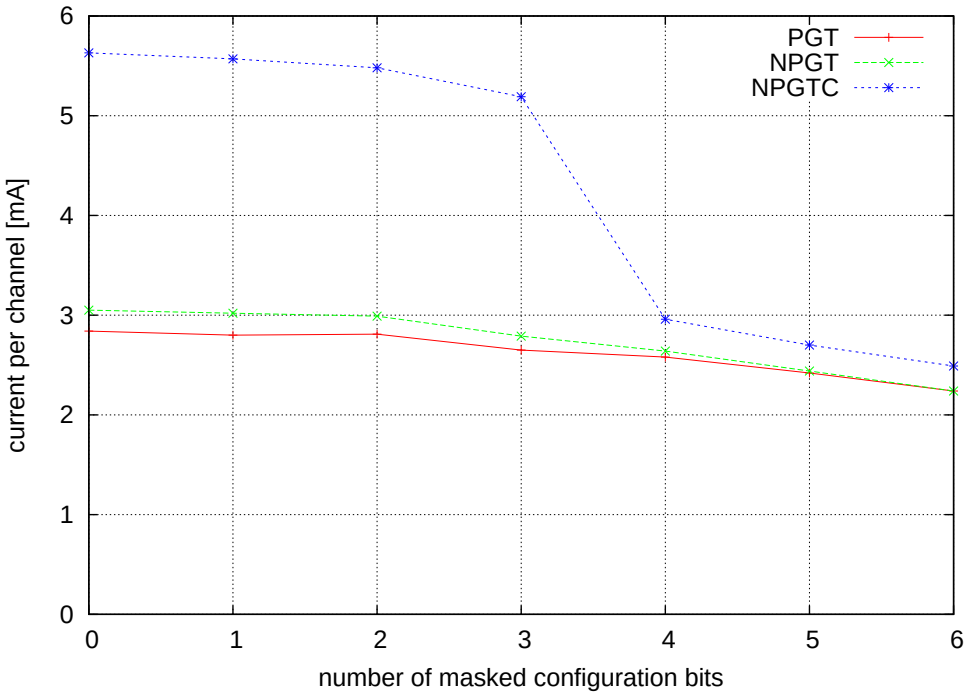

Figure 9.6: series of masked lower configuration bits with three different sets of used filter components

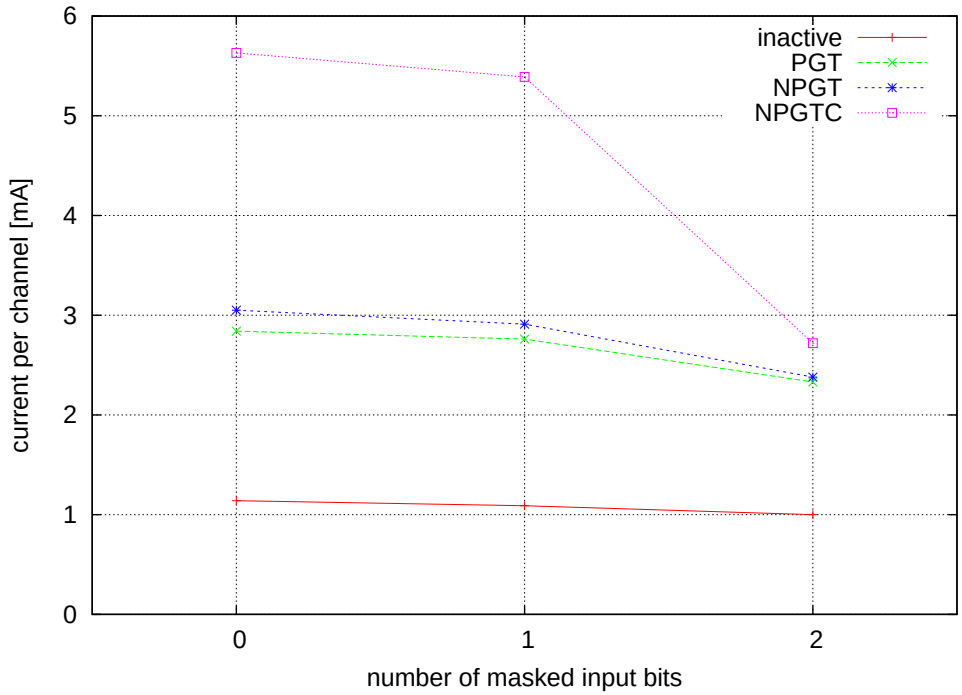

Figure 9.7: power consumption as a function of masked input bits

Another option to save power is to reduce the demand of accuracy of the multiplication factors of the whole filter configuration. Therefore, a certain number of the lower bits of all factors are set to zero (fig. 9.6). Unfortunately, the effect is quite small in the range up to three bits, corresponding to an accepted relative error of 2^{-7}. If more bits were masked, the savings would be larger but the filter accuracy would deteriorate into a non-acceptable range.

Finally, the option to mask input bits has been studied. The idea is to mask the lower input bits within the time between two event. The masking would take place after the pedestal correction filter, such that it can still settle to the actual baseline value. The tail cancellation filter needs to be operated continuously to be able to react on un-triggered peaks or to settle to its equilibrium. This would still be possible with input values of lower accuracy. This option has not been foreseen in the present design of the chip, but its effects can be investigated by the corresponding manipulation of the input data. The results can be taken into account for improvement of future designs. As shown in figure 9.7, a significant amount of power is saved by masking the lower two input bits. Of course, the quantity strongly depends on the noise situation on the input ports, but especially the savings in case of the usage of the crosstalk suppression filter are quite promising.

9.3 Dynamic Power Increment

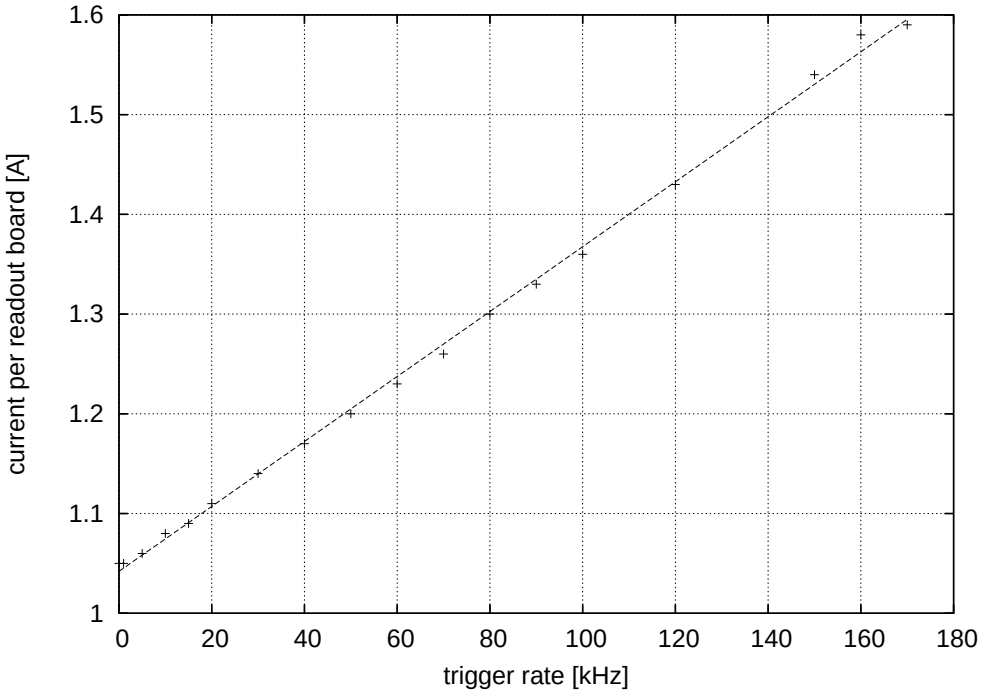

Figure 9.8: power consumption increment caused by the processing and transmission of tracklets

In addition to the continuously active devices in the chip, the CPUs are involved in the data processing. They are activated automatically after the drift time to merge potential tracklet candidates and to transmit their parameterization. Figure 9.8 shows the corresponding increment of the power consumption in case of the processing of four tracklets per MCM. The measurement has been taken on a complete readout board to include the board-level tracklet merging into the analysis. The dependence of the power increment from the trigger rate is rather linear. A straight line fit results in a ratio of $C_{\mathrm{PT}} = (3.26 \pm 0.03)\ \frac{\mathrm{mA}}{\mathrm{kHz}}$ per readout board. Here, the pre-trigger rate of the TRD f_{PT} has to be taken into account:

$$\Delta I_{\mathrm{PT}} = C_{\mathrm{PT}} \cdot f_{\mathrm{PT}} \tag{9.1}$$

After an event has been accepted to be read out, it is the task of the CPUs to take the data out of the event buffer and to transmit it in a suitable format. Figure 9.9 shows the linear dependency of the readout frequency and the power, which is needed for the complete data transmission process. The measurement has been taken with four different settings of the number of samples, which have to be transmitted per channel and event. The slopes of the straight line fits

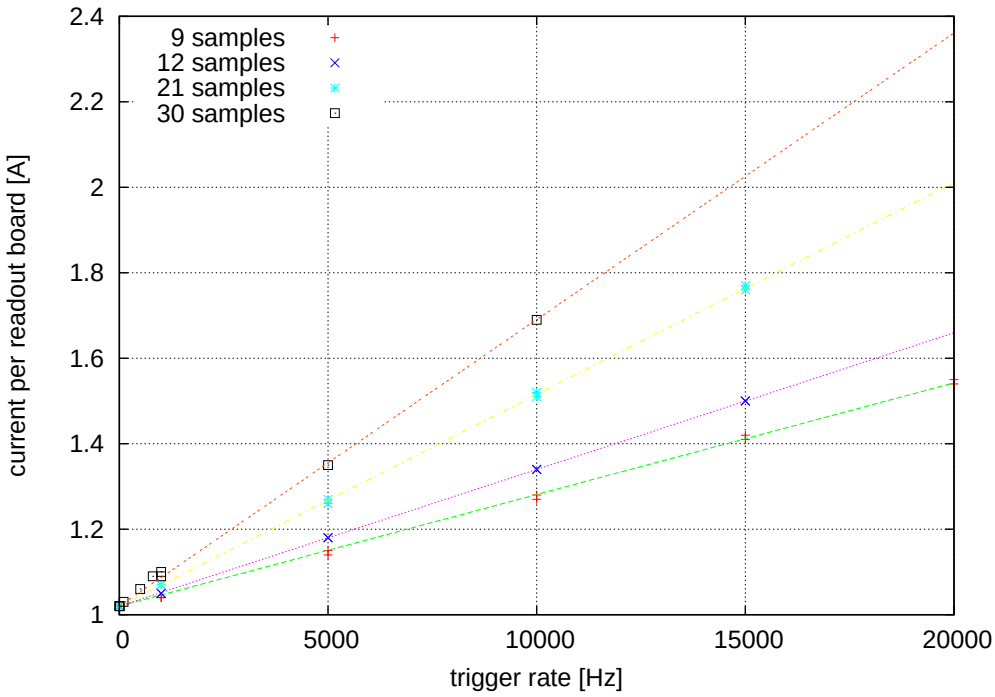

Figure 9.9: power increment due to a full readout

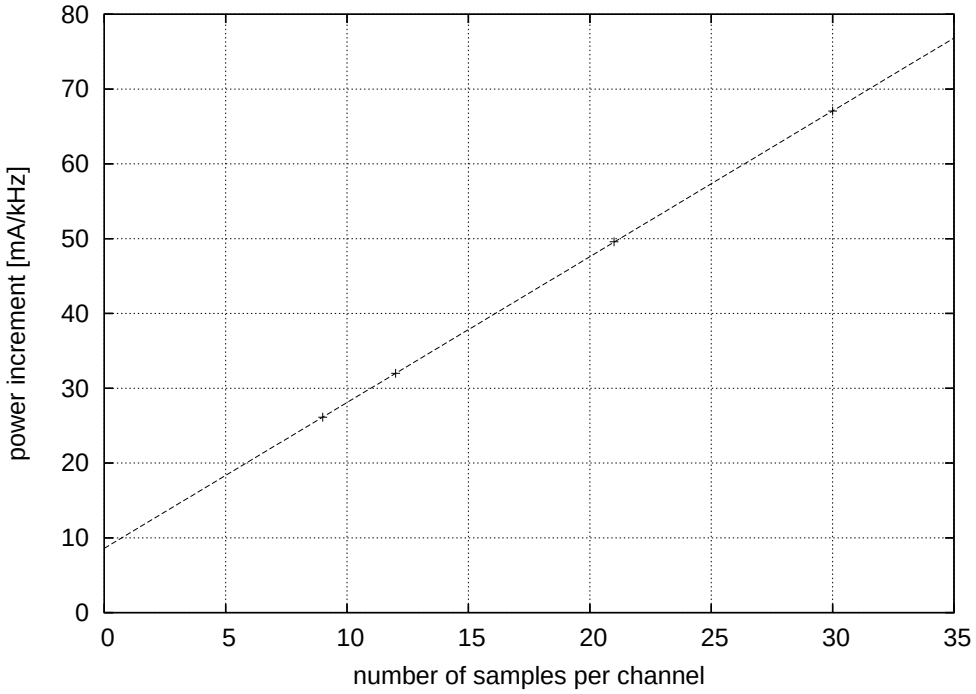

Figure 9.10: power increment as a function of transmitted samples

through the four sets of data points are plotted in figure 9.10. Again, there is a linear relation, because the run time of the CPUs is proportional to the number of samples. In addition, the power implications of some program overhead are visible. With the straight line parameters $C_{\mathrm{RO}} = (1.949 \pm 0.002)\ \frac{\mathrm{mA}}{\mathrm{kHz}\cdot\#\mathrm{samples}}$ and $A_{\mathrm{RO}} = (8.60 \pm 0.04)\ \frac{\mathrm{mA}}{\mathrm{kHz}}$, the dependencies are given by:

$$\Delta I_{\mathrm{RO}} = (C_{\mathrm{RO}} \cdot \#\mathrm{samples} + A_{\mathrm{RO}}) \cdot f_{\mathrm{RO}} \tag{9.2}$$

Here, the readout rate of the TRD f_{RO}, which is equal to the level-1 accept rate, has to be taken into account.

9.4 Conclusions

The power consumption of the TRAP chip is composed of a permanent and a dynamic contribution. The dynamic contribution is caused by the CPUs' activity. It is proportional to the run time of the programs and the trigger rate. Since the trigger rate is a given quantity in the experiment, the optimization focuses on the minimization of the run time of the assembler programs. In addition, this reduces the latency for the trigger decision and the raw data readout, respectively.

The permanent power consumption is caused by fluctuations of the input signal. It can be reduced by minimizing the number of the affected gates in the circuit. The main benefit is achieved by masking of unused signal paths. This can be observed by the variation of the hit threshold, which disables the masking of the preprocessor input ports. In addition, the masking of lower bits at the beginning of the signal path suppresses most of the fluctuations. Since the maximum precision is needed only during the events, one could implement a trigger-dependent mask in future designs. Another option is the manipulation of the target baseline of the pedestal correction filter. Here, one should choose a pedestal value, which is far away from multiples of higher powers of two. Finally, the number of pipeline stages should be increased. This reduces the number of toggles per clock cycle of the gates in the rear parts of the circuit.

10

Application

10.1 Overview

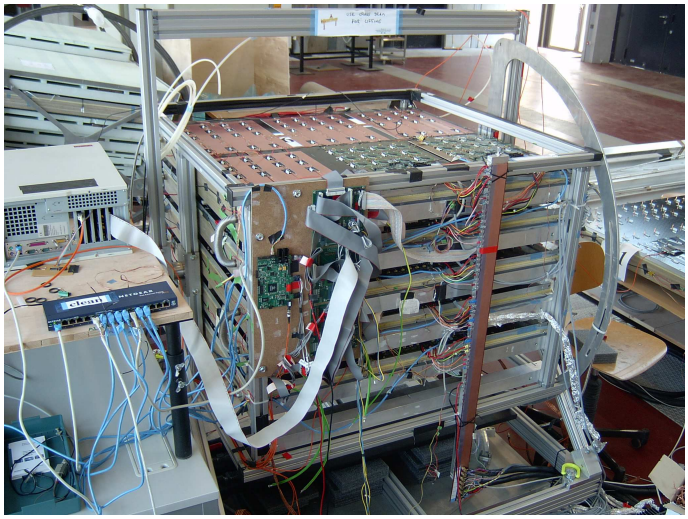

Figure 10.1: TRD prototype stack at KIP

i

The final proof of the functionality of an experimental device like the TRAP chip can only be given by the application in an experimental environment. Therefore, an adequate part of a TRD prototype stack is equipped with prototypes of the front-end electronics (fig. 10.1). This setup has been used for data acquisition and first online processing tests in the course of a beam-time at CERN in fall 2004 (section 10.2). Furthermore, it is continuously in use at KIP. It is the basis of integration and noise studies [ko0], power measurements (section 9.3) and advanced online processing tests (section 10.3).

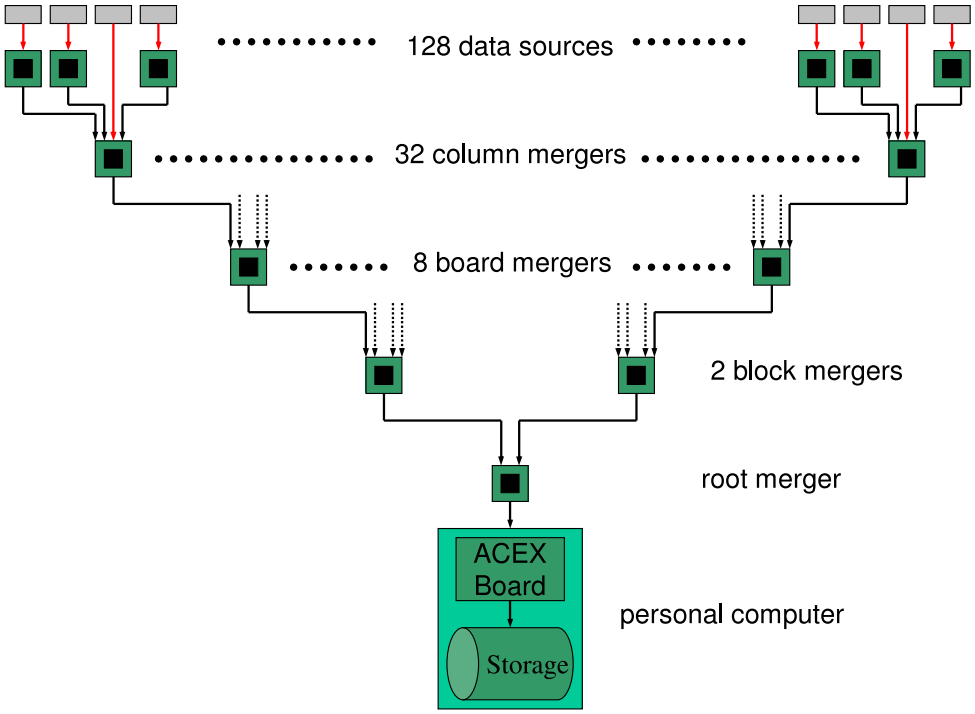

Figure 10.2: topology of the prototype readout tree

The setup has been built with eight readout boards, one on each intermediate layer and two on the lowest and on the highest layer. All readout boards are of the same type, each of them contains one DCS board (see fig. 2.15, p. 16). It is responsible for power control, clock distribution, configuration and triggering of the readout board.

The data and tracklet readout is performed via the readout network interface (fig. 10.2). On each board there are 16 MCMs, that are taking data. They are organized in four columns. One MCM within a column merges its own data and that of the other three of its column to a common data stream. The streams of the four column mergers are merged again on board level. The data of four readout boards is combined in block mergers, which are hosted by MCM tester PCBs as well as the root merger. It combines the streams of the two block mergers to a common data stream. Finally, the data is received by an ACEX card [acex]. It is available via the PCI bus of the PC, which incorporates the card. For data distribution to monitoring and storage systems, a publisher-subscriber framework has been used [ts0].

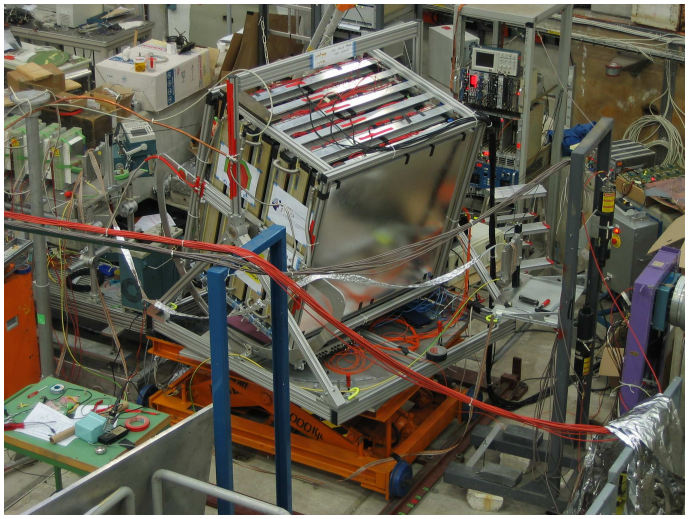

Figure 10.3: the TRD prototype stack at CERN PS (T9)

10.2 CERN 2004 Beam-time

The TRD prototype stack has been operated for three weeks at area T9 of the Proton Synchrotron (PS) at CERN (fig. 10.3). The detection performance of the TRD chambers has been analyzed for a variety of beam directions relative to the stack and for different beam momenta between 1.0 and 10.0 GeV/c [bv1]. The events have been initiated by an electron trigger [aa2]. One readout board was disabled due to the observation of contact problems on the input clock connection, which were caused by mechanical stress during transportation of the stack. The remaining front-end electronics ensemble has been shown to work reliable in continuous operation for the whole beam-time of almost three weeks, taking 60.3 GB of event data. This includes data acquisition in various regions of interest of the stack in combination to the usage of the pedestal correction filter, which is keeping the data almost raw for offline analysis.

Furthermore, the major online processing capabilities of the front-end electronics have been tested in separate runs. The functionality of the tail cancellation filter in online operation was proven with estimated parameters. The main features of the filter become apparent on the average signal shape along the tracks (fig. 10.4). Peaks like the first one, which is caused by the transition of the particle through the amplification region, are only slightly affected by the filter. The plateau and the pile-up during the drift time are scaled down by more than a factor of two. The exact quantities strongly depend on the filter settings. Within this run, the filter is over-steering as it is indicated by the negative signal (relative to the baseline) at the end of the drift time. The observation of the average signal shape is the basis of the tail cancellation filter optimization (sec-

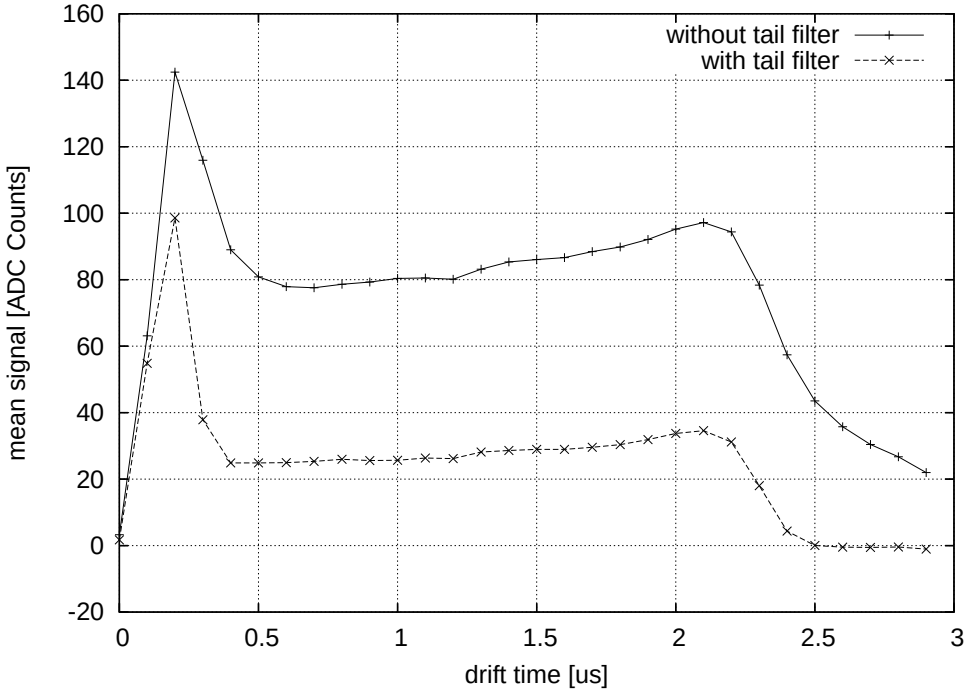

Figure 10.4: mean signal along the tracks with different filter settings

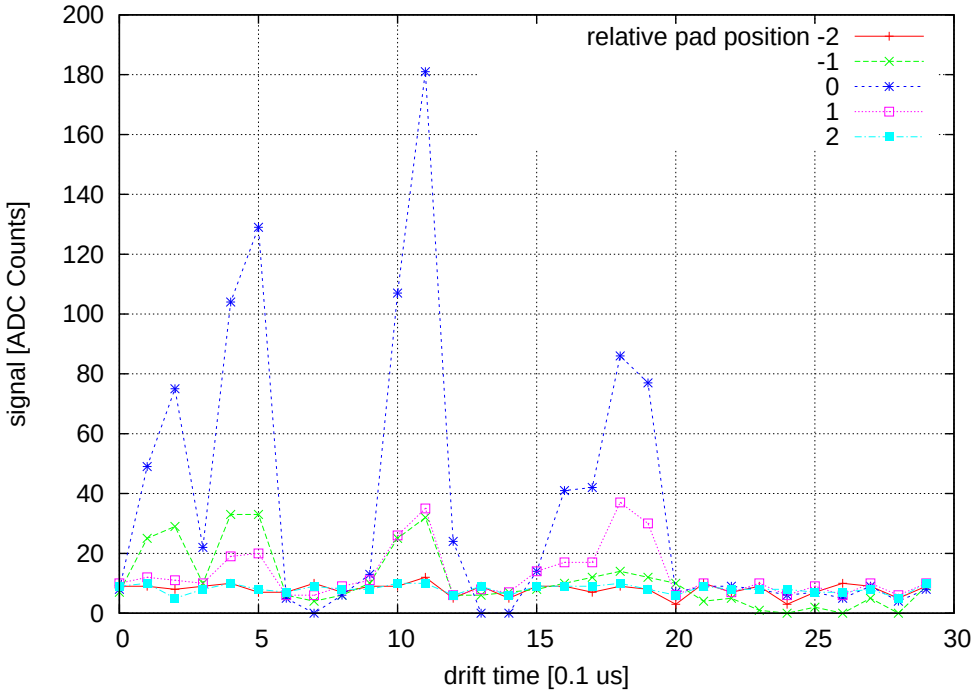

Figure 10.5: example event with tail cancellation filter

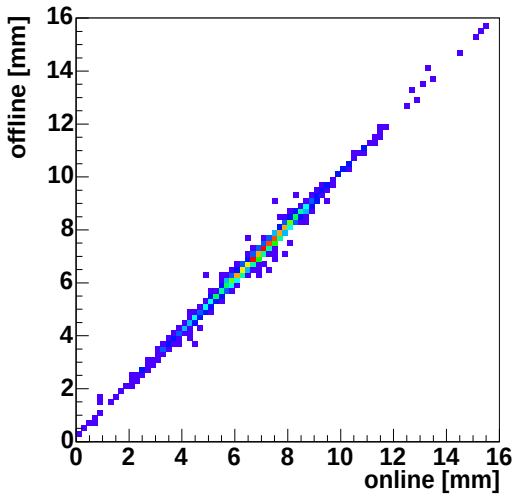

Figure 10.6: correlation of deflection lengths, which are based on online and offline analysis

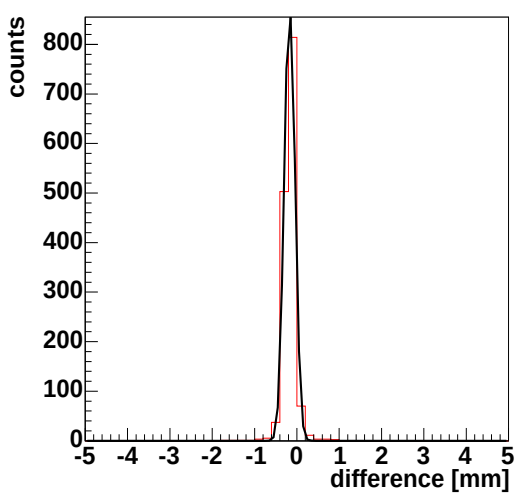

Figure 10.7: distribution of the difference of online and offline deflection lengths (red) and a Gaussian fit (black)

tion 7.4). The basic idea of the tail cancellation is plainly visible in single events (fig. 10.5). Here, the drift region contains a set of randomly distributed peaks, which are clearly separated without the smoothing of the tail. In single events, the over-steering of the non-optimized filter is more obvious since there is more dynamics in the data than in average events.

The tracking functionality of the preprocessor has been tested in an independent run. All status registers of the preprocessor have been read and added to the raw data stream. This comprises the fit registers of all channels (see fig. 6.3, p. 59) and the numbers of those channels, which have been selected as tracklet candidates.

Because the amount of data, which has to be sent, is larger than the capacity of the FIFOs of the readout network interface, the CPUs had to be involved in the data management. After the transmission of a first bunch of data, the FIFOs are emptied and the CPUs are restarted with an appropriate interrupt. Subsequently, a second bunch of data is flushed into the FIFOs. Since this procedure works properly up to the board merger level, only the behavior of the preprocessors on single readout boards was traced. Thus, the online processing data of the beam-time does not allow to analyze complete particle trajectories through the stack. Only the tracking information of one layer is available.

The analyzed run is focused on the uppermost central readout board on layer six. The particle beam was collimated such that the tracklets were spread over a few channels. Thus, all tracklets were concentrated on one MCM. 1622 events

have been recorded at a beam direction of $\phi = 15°$, $\Theta = 15°$ and a beam momentum of 6 GeV/c. The acquisition and preprocessor operation was combined with the pedestal correction filter.

A comparison of online and offline analysis with a similar algorithm shows a quite good agreement. Only within a few events, a discrepancy has been observed. In those cases, the number of detected hits differs slightly. There are 21 examples, in which the tracklets have not been identified by the offline algorithm because the number of hits of these tracklets is around the threshold.

The online calculations are based on the 12-bit values out of the filter, including two digits after the decimal point. For storage in the event buffer they are truncated to 10-bit values. The offline analysis is based on event buffer data. All differences in terms of hit points are related to clusters, whose hit charge is in the close neighborhood of the hit charge threshold.

The most important and most sensitive parameter of the tracking procedure is the slope of the tracklet, which is expressed as deflection per chamber. In figure 10.6 the tracking functionality, which is based on the online hit sums, is compared to tracking results, which are calculated offline from the event data. It shows a correlation diagram with a bin size of 0.2 mm $\times$ 0.2 mm. There is a good agreement between both calculations. The variations of the beam direction are visible as well as a systematic shift to a mean deflection of (6.81 ± 0.04) mm while the orientation of the stack would lead to a deflection of 8.04 mm. This is due to the ion tail, which has not been canceled out of the data.

The slight deviation of online to offline results (fig. 10.7) is caused by rounding errors within the fixed point arithmetics of the chip and the discrepancy of the data basis explained above. The typical variation is (0.122 ± 0.003) mm. The systematic difference from online to offline calculation of (0.171 ± 0.003) mm is also caused by internal truncations within the chip arithmetics.

10.3 Cosmic Radiation

The full signal processing functionality is tested in continuous measurements with the prototype stack at KIP (fig. 10.1). This implies the development of calibration procedures for filter and tracking parameters (chapter 7). Cosmic radiation is used as particle source.

Typically, these are minimum ionizing particles with momenta of less than 1 GeV/c. While a mixture of Xenon and Carbon-dioxide (Xe-CO_2) has been used for the CERN beam-time and will be used in the final experiment, there is only a mixture of Argon and Carbon-dioxide (Ar-CO_2) available at KIP. The combination of the different gas filling of the chambers with heavier low momentum particles (e.g. muons) leads to rather weaker signals. This leads to the signal amplitude distribution, which is shown in figure 10.8. The fit of a Landau

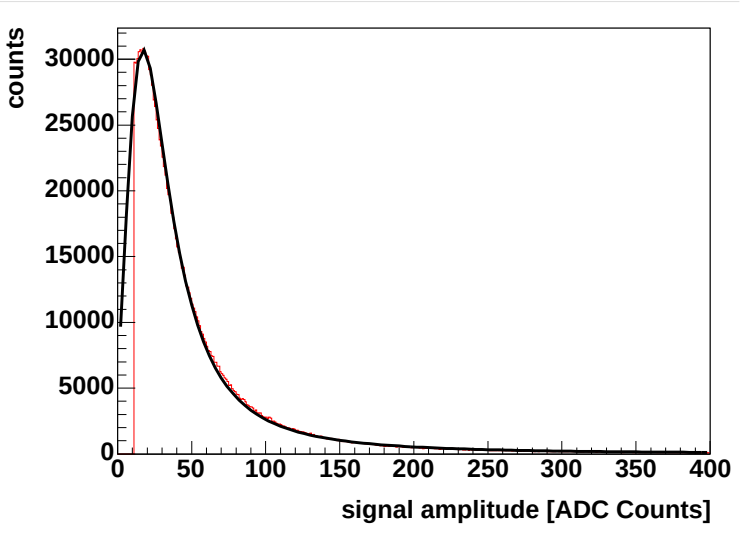

Figure 10.8: signal amplitude distribution (red) and a Landau fit (black)

distribution gives a most probable value of $\mathrm{MPV}_{\mathrm{Amp}} = 19.3$ ADC Counts with a spread of $\sigma_{\mathrm{Amp}} = 9.2$ ADC Counts. In combination with a typical noise performance of the stack of 1.3 ADC Counts, this results in a signal to noise ratio of $\mathrm{SNR}_{\mathrm{stack}} = (14.8 \pm 7.1)$. In the ALICE experiment a value of $\mathrm{SNR}_{\mathrm{ALICE}} = 40$ is expected. Due to activities, which are related to noise improvements, power distribution analysis and DCS board debugging, only the inner four modules of the stack are included into the present measurements [atw].

The events are triggered by coincidence of scintillator pads above and below the stack. An ACEX card [acex] has been integrated into the trigger system which makes it possible to generate an ALICE-like trigger stream out of single trigger pulses. Thus, the chips are configured such that data acquisition, tracking and raw data readout are controlled by a sequence of pre-trigger, level-0 and level-1 accept signals instead of only one trigger signal. The programming of the CPUs has been expanded by the tracklet assembly functionality (section 6.3) which leads to a tracklet content in the data stream. A monitoring software has been developed [aw0] which analyzes the data stream and composes all important runtime parameters.

In figure 10.9 an example event is shown. It displays the four pad rows, which are covered by a readout board in each of the layers. The samples of the different layers are put on top of each other to illustrate the geometrical context. The ratio of the sampled area to the space in between, which is mainly due to the radiators, is larger than the relative size of the drift region of the chambers because a couple of pre- and post-samples are shown, too. The example track is traversing all connected modules, chamber five in the first pad row and chamber two to four in the second pad row. The readout board on layer three contains

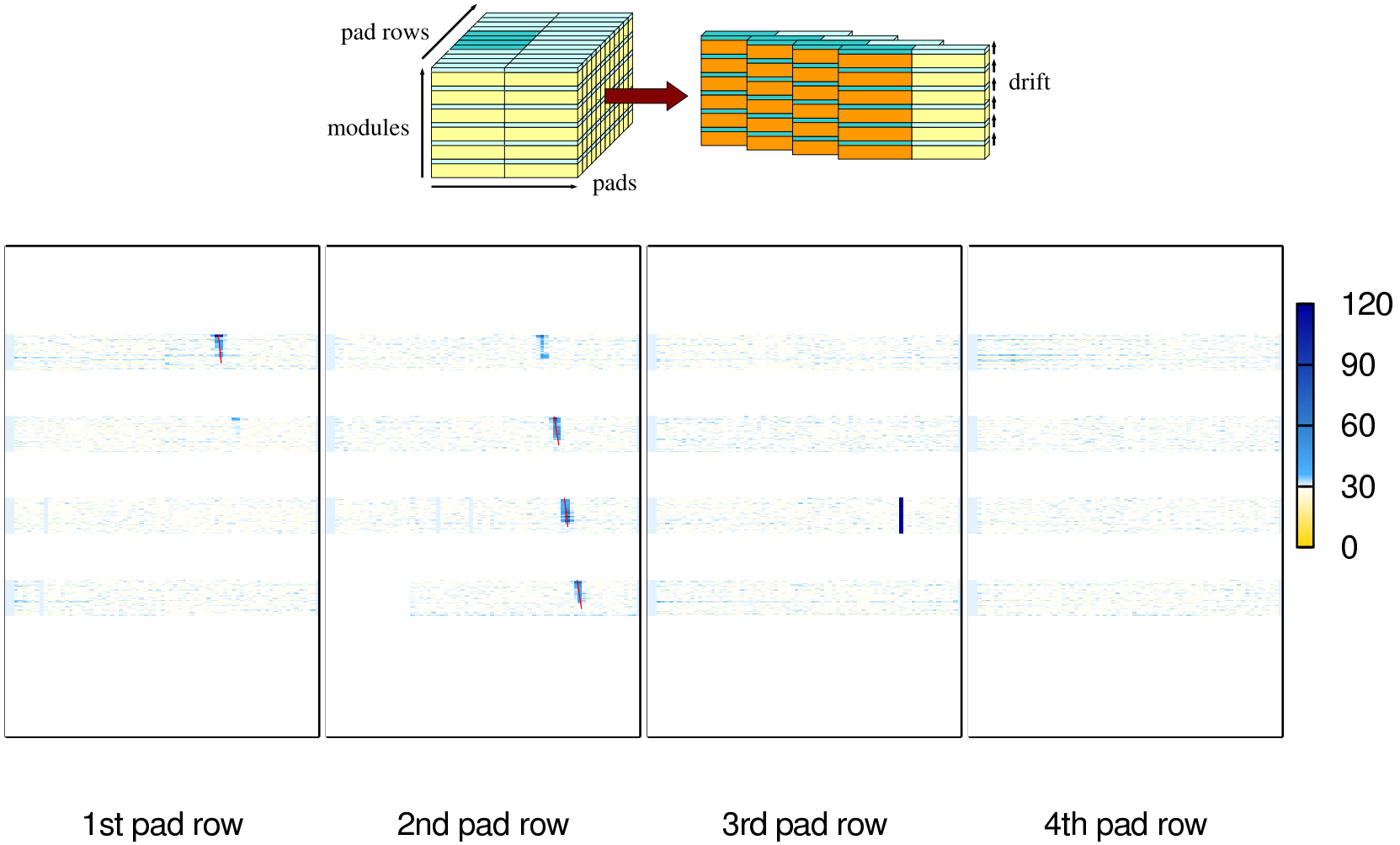

Figure 10.9: example event with a common track in 1st and 2nd pad row, the active part of the stack is marked with orange and dark blue color in the upper drawing

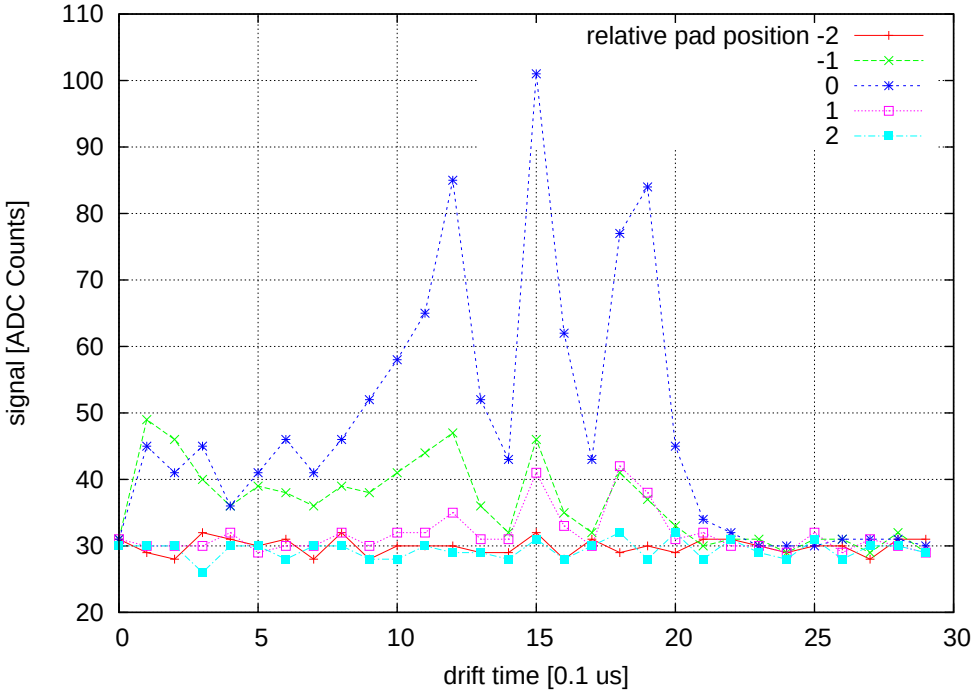

Figure 10.10: example tracklet with tail cancellation, taken from third layer, second pad row of the same event as in fig. 10.9

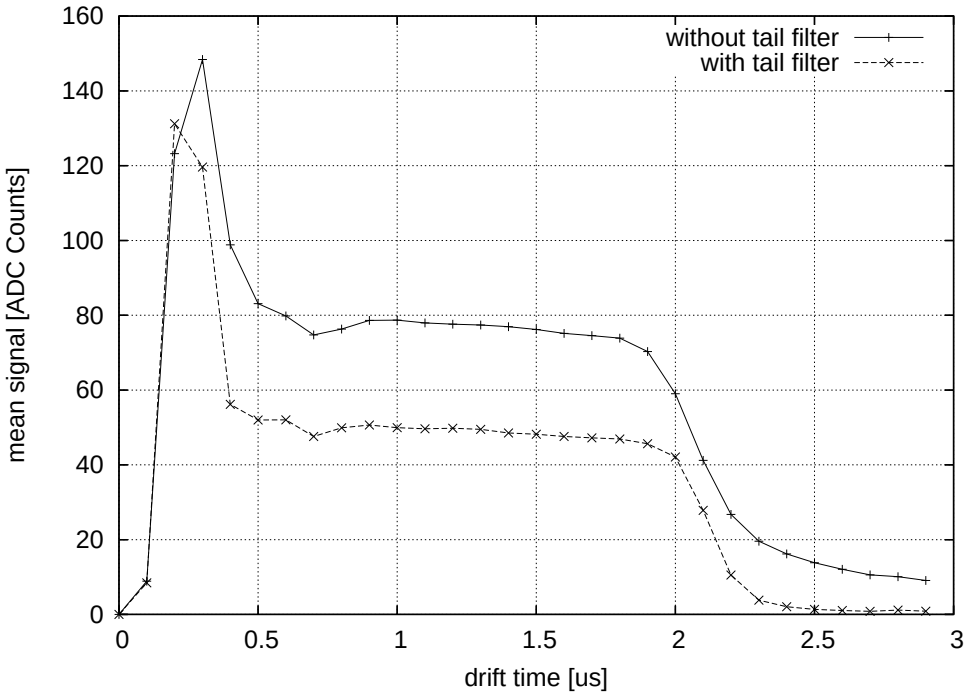

Figure 10.11: mean signals along the tracklets

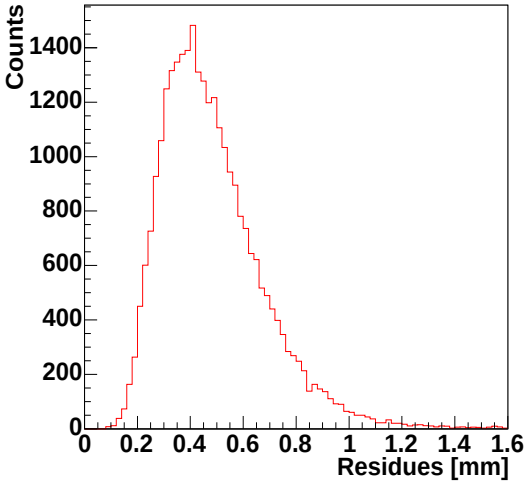

Figure 10.12: mean deviation of the hits of a tracklet from the corresponding fit

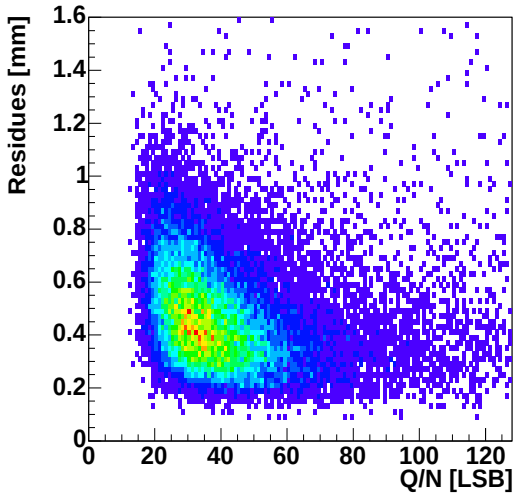

Figure 10.13: histogram to show the relation between mean hit charge of a tracklet and mean hit deviation

one bad channel in a chip, which are connected to the third pad row. In this extreme example the particle was located very close to the border between the two pad planes on the uppermost layer as it is indicated by the charge sharing between those two rows. One layer below, some little portion of the signal is still left in the first pad row. The online tracking ignores that, it only works on the basis of the two dimensional data of one pad row. The tracklets, which were found by the affected TRAP chips, are marked by red lines. In figure 10.10 the signal along the tracklet of the third layer on the second pad plane is shown. It is clearly visible, that the tail cancellation filter cuts off the whole tail, the signal drops almost immediately back to the baseline without any over-steering.

The monitoring software is used especially to extract information, which is necessary to generate or to check the filter calibration. In figure 10.11 the mean signal along all tracklets of two different runs is shown. Initially, the setup is operated without tail cancellation filter. The corresponding mean signal is the basis of the calibration of the tail cancellation filter (section 7.4). For comparison, the mean signal of the subsequent run with the optimized tail cancellation filter has been added. The peak amplitude is almost unaffected and its tail is negligible. Furthermore, the signal level of the plateau, which is due to the drift time, is decreased by a factor of 0.67 and the end of the drift region is much sharper.

To evaluate the tracking procedure, various parameters have to be taken into account. As a good estimate of the position resolution in pad direction, the mean

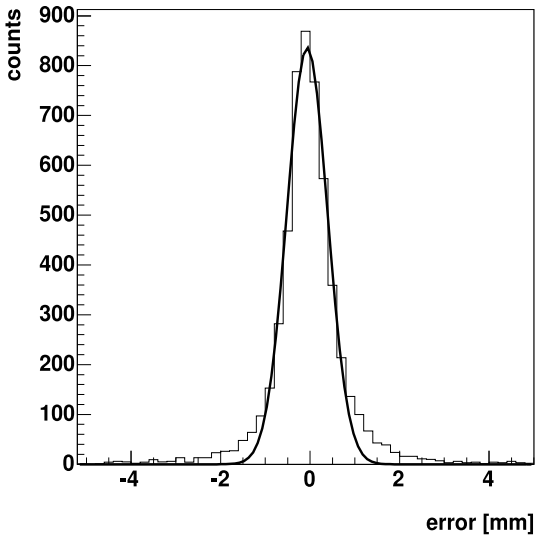

Figure 10.14: error distribution of the online calculated deflection length in a run with active tail cancellation filter and a Gaussian fit (σ=0.45 mm, RMS=1.05 mm)

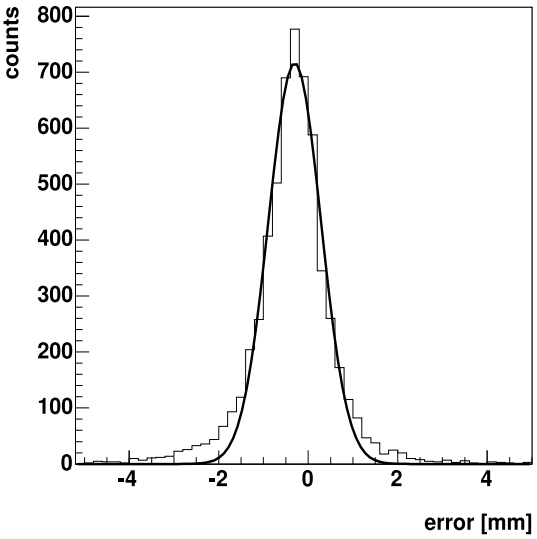

Figure 10.15: error distribution of the online calculated deflection length in a run with inactive tail cancellation filter and a Gaussian fit (σ=0.59 mm, RMS=1.09 mm)

deviation of the hits of a tracklet from the fitted line is estimated (see eqn. 6.5, p. 55). The corresponding histogram (fig. 10.12) shows a typical performance of

$$\overline{\sigma_y^2 \chi^2} = (480 \pm 190)\ \mu\text{m}$$

In figure 10.13 a two-dimensional histogram demonstrates the relation between mean hit charge $\frac{Q}{N}$ and the tracking performance. Obviously, position resolution and fitting quality deteriorate with lower values of the mean hit charge. In general, there are a couple of outliers, which are caused by electric discharges which do not correspond to a particle track and by singular disturbances from the environment. Thus, the median of the distribution in figure 10.12 gives a better information of the typical fitting performance:

$$\text{Median}(\sigma_y^2 \chi^2) = 450\ \mu\text{m}$$

The final evaluation of the track fitting performance is given by the analysis of the reconstructed tracklets, especially the deflection length. Since the particle trajectory is unknown and there are no further tracking detectors, no track reference is available a priori. It has to be calculated from the TRD data as well. For the present analysis, a GTU-like attempt by Dr. B. Vulpescu has been used. It is based on a fit through all layers, fixing position and direction of a track. Therefore, only one space point is used per chamber. It corresponds to the start points

of the tracklets. The error of the deflection lengths is rather small because the points are distributed over a long distance, the width of three chambers. After the global fit, the Θ-angle is known and the locally calculated deflections have to be corrected for the tilted pad effect (see section 7.8). A comparison of the corrected deflection lengths with those, that are expected due to the global fit, is shown in figure 10.16. It is corrected for the error of the global fit, which has been estimated on simulated data. The resolution is given by:

$$\Delta\tilde{\vartheta} = (0.454 \pm 0.001) \text{ mm}$$

In figure 10.15, the error distribution with inactive tail cancellation filter is shown. The resolution is worsened to (0.587 ± 0.001) mm. In the final experiment, the effect will be even stronger: As shown below, the tail mostly influences the tracking performance at higher deflection values. The resolution calculated above is the integral resolution of the tracks, which are detected by the stack. In the ALICE experiment, the distribution will have more weight on tracks with higher deflection lengths.

The deviation of the online calculated deflections slightly depends on the deflection itself (see fig. 10.16). So, there is a systematic error of (9.9 ± 3.5) % and a shift in positive direction of (0.13 ± 0.08) mm. In case of inactive tail cancellation filter, the systematic error is even more dominant (see fig. 10.17). It is about (-19.0 ± 3.8) % with a shift in positive direction of (0.16 ± 0.08) mm.

These errors are a hint, how much the data is disturbed. They are a minor problem because only the correlation of the local deflections is used in the GTU and it is possible to correct for it. A stronger argument for the usage of the tail cancellation filter is given by the observation of the resolution at various deflection values. It is visible by the error bars in figures 10.16 and 10.17 and in figures 10.18 and 10.19 directly.

For small deflection lengths, the resolution with tail cancallation is worse than without. At zero deflection, fits with parabolas lead to values of 0.33 mm with tail cancellation and of 0.28 mm without. This is reasonable, since the filter reduces the signal height and the tail does not disturb those tracklets. The more the deflection differs from zero, the more the statistical error increases. This effect is much stronger for the non-filtered tracklets. It is parameterized by the second derivative of the parables in their minimum. With 0.033 $\frac{1}{mm}$ it is more than twice as large as without tail cancellation filter than with it ($0.015 \frac{1}{mm}$). Because the deflections reaches more than 12 mm in the ALICE experiment without correction of the Lorentz deflection (section 7.8), this is the dominant effect.

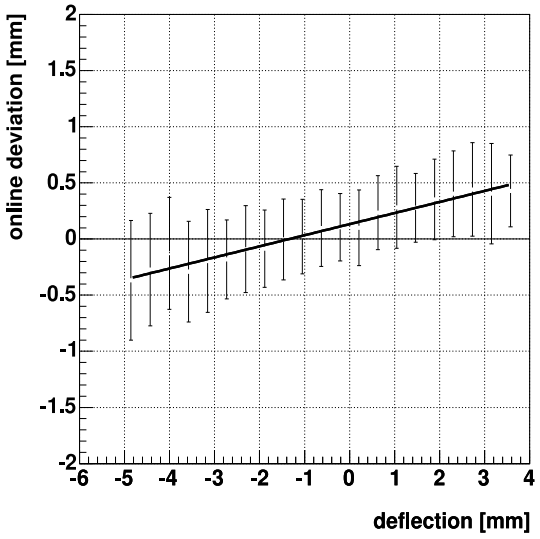

Figure 10.16: deviation of the online deflection as a function of the deflection with active tail cancellation filter

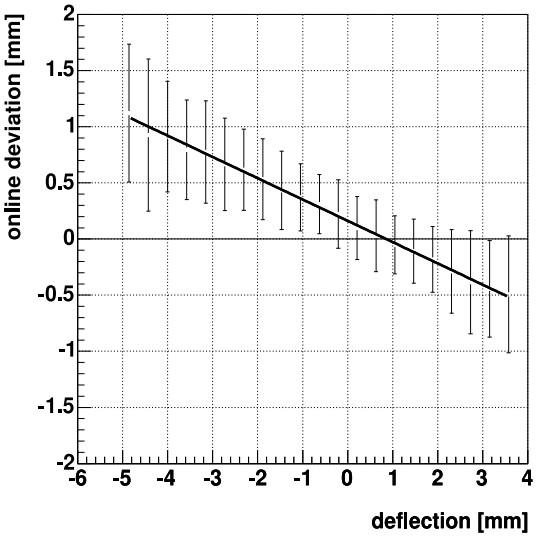

Figure 10.17: deviation of the online deflection as a function of the deflection with inactive tail cancellation filter

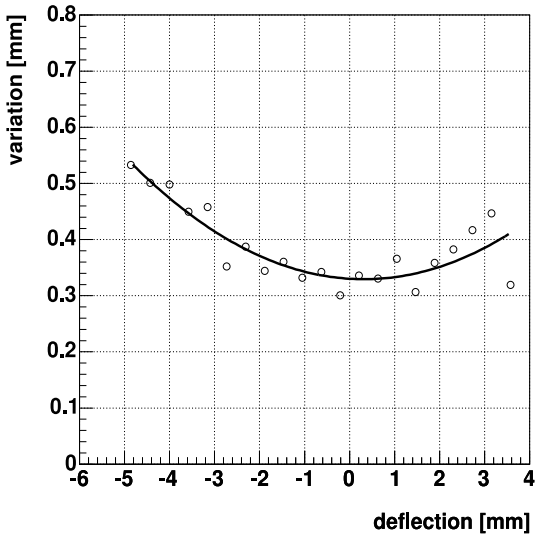

Figure 10.18: resolution of the online deflection as a function of the deflection with active tail cancellation filter

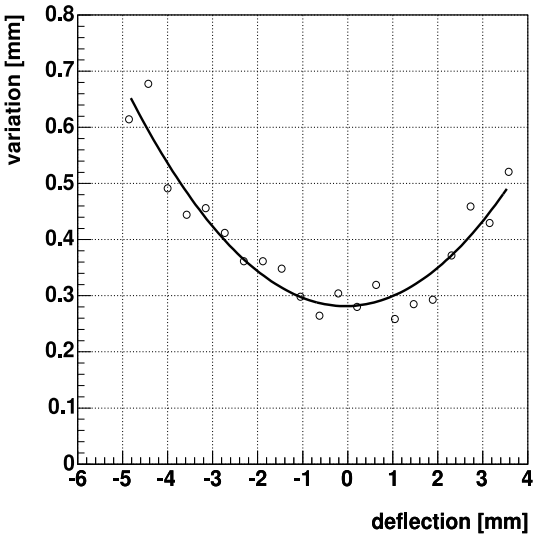

Figure 10.19: resolution of the online deflection as a function of the deflection with inactive tail cancellation filter

11

Summary and Outlook

The ALICE Transition Radiation Detector is used for a trigger decision, which is based on the detection of electrons with high transverse momentum. Therefore, particle tracks have to be recognized and the particle identity has to be determined in the whole detector within a time budget of 6 µs. This is achieved by a hierarchic approach. The tracks are detected in segments by the TRAP chips in a decentralized approach. Only the tracking information is sent to a central device, the Global Tracking Unit, which performs the final track assembly and momentum reconstruction. In addition to the processing, the detector data is stored for readout in case of a positive trigger decision. The raw data readout is based on the same network infrastructure as the tracking data transmission.

The TRAP chip is structured according to its objectives: The signal path starts at the 21 differential analog ADC inputs. They provide up to 30 samples per event, which is represented by a two-dimensional data matrix. Next, the signals are improved by a digital filter. Afterwards, the data is distributed to an event buffer and to a preprocessor. The event buffer stores the data of one event. The preprocessor prepares the fit calculation of the track segments. The entries of the event buffer as well as the fitting information of the preprocessor are accessible by four on-chip CPUs. They conduct final local track parameterization and data compression in case of event data readout. Therefore they access a network interface module, which handles the assembly of the data stream.

The digital filter corrects artefacts of the signal generation process. These are common nonlinearity, baseline and gain variations. Furthermore, the ion tail and the capacitative crosstalk of neighbor signals are suppressed. The ion tail is caused by the gas amplification process in the chamber. It is visible as a long signal decay of about 2 µs in contrast to a signal rise time of 70 ns. The approximation of the tail by the sum of two exponential functions corresponds to the second order design of the tail cancellation filter. The crosstalk generation is described by a two-dimensional filter matrix. The crosstalk suppression corresponds to the application of the inverse of that matrix.

The tracking is based on a straight line model. Each track segment is represented by a series of hits. Because the tracks are assumed to be sufficiently stiff, the hits of one segment are distributed over a maximum of two channels. The fitting is separated in two steps. In parallel to the data acquisition, the preprocessor detects the hits and calculates the fits channel-wise. Since the tracks are assumed to be isolated and the hits induce a signal in several adjacent channels, the hit positions are determined with a sub-pixel precision. Afterwards, the CPUs are assigned to those channel pairs, in which track segments have been observed. The fits of these channels are merged and the final tracking parameters are calculated.

The readout of the event data is realized by the CPUs. They can access the full data of the event buffer. In addition, a couple of hardware-based data indicators are available as well. They accelerate the implementation of typical data compression strategies, which lead to compression ratios of 40 down to 15 %, depending on the occupancy of the detector.

The digital filter and the tracking system incorporate a huge amount of free parameters. Calibration strategies have been developed and their functionality has been shown. Common or almost constant parameters are optimized by offline user programs. This affects especially the nonlinearity correction, the tail cancellation and the crosstalk suppression filter as well as the tracking parameters. The adjustment of individual and fluctuating parameters, as expected for the pedestal and the gain correction filter, is simplified such that it can be computed by the TRAP chip's CPUs.

The TRD will contain more than 65,000 MCMs. A test device has been set up for their production. It performs a couple of external and internal tests. The internal tests are based on CPU programs. The devices on the TRAP chip are accessible by the CPUs in a much faster way than externally via the configuration and control interface.

The advantages of placing the signal processing capabilities as close as possible to the signal source have to be compared to the disadvantages. Theses are local digital noise generation and power consumption. The noise contribution of the digital TRAP circuits is negligible during the acquisition phase. The power consumption leads to an increased effort for power supply and cooling infrastructure. Thus, the power consumption of the MCMs has been analyzed rather detailed, especially their dependence on parameter settings and environmental conditions. The main results are that the power consumption can be reduced by masking of unused signal paths, by pipelining of the arithmetics and by masking of lower input data bits as long as they are not needed like between two events. All but the last strategy is realized by proper settings of the TRAP chips. The masking of lower input data bits has been analyzed by manipulation of the input test patterns and can be considered as an option for future designs. In addition to the permanent power consumption, the CPUs' activity causes a huge increment. Its contribution to the average consumption depends on the run time of the CPU programs and the trigger rate. In addition to minimum latency, this

is another motivation for the minimization of the run time of the tracking and readout programs.

The functionality of the measurement and online signal processing system was shown by the operation of a detector prototype. It was tested with a test beam at the CERN Proton Synchrotron and with cosmic radiation at KIP. This includes filter calibration, raw and tracking data acquisition and the subsequent comparison with offline analysis. The agreement is quite good. Some minor differences can be explained by implementation details of the online processing and the offline algorithm. It is even possible to combine several track segments to tracks and to determine the tracking resolution. One can clearly see the improvements, which are caused by the application of the tail cancellation filter.

The MCMs, especially the TRAP chips, are now final, well-working devices and in a state, in which they are ready to be integrated into a larger system. The next step will be the combination with the first version of the Global Tracking Unit. This will be a first complete building block of the ALICE TRD online tracking system. The further steps are the integration into the first super-modules in 2006 and the assembly of the full TRD in 2008.

Acknowledgement

I thank everybody who contributed to the success of this thesis. These are especially Prof. Dr. V. Lindenstruth and Dr. V. Angelov.

For advice, help and suggestions I thank furthermore Dr. C. Adler, J. de Cuveland, D. Emschermann, L. Hess, Dr. I. Kisel, Dr. C. Lippmann, J. Mercado, Dr. K. Oyama, Dr. P. Popov, C. Reichling, Dr. I. Rusanov, U. Sauerzapf, R. Schneider, M. Schwarz, Dr. T. Steinbeck, Dr. H. Tilsner, Dr. B. Vulpescu, Dr. A. Wiebalck.

Appendix A

Abbreviations

ADC	Analog Digital Converter
ALICE	A Large Ion Collider Experiment
ASIC	Application Specific Integrated Circuit
ATLAS	A Toroidal LHC Apparatus
BNL	Brookhaven National Laboratory
CASTOR	Centauro and Strange Object Research
CERN	European Organization for Nuclear Research
CMS	Compact Muon Solenoid
CPU	Central Processing Unit
DDR	Double Data Rate
DMEM	Data Memory
DRC	Design Rule Check
FIFO	First-In First-Out Memory
FMD	Forward Multiplicity Detector
FPGA	Field Programmable Gate Array

FRF	Fit Register File of the TRAP Chip
FWHM	Full Width at Half Maximum
GDS2	Gridded Data Set Format 2
GPIB	General Purpose Interface Bus (IEEE 488)
GSM	Global State Machine of the TRAP Chip
GTU	Global Tracking Unit
HLT	High Level Trigger of the ALICE Experiment
HMPID	High Momentum Particle Identification Detector
IMEM	Instruction Memory
ITS	Inner Tracking System
KIP	Kirchhoff-Institute of Physics, University of Heidelberg
LEF	Library Exchange Format
LEP	Large Electron Positron Collider
LHC	Large Hadron Collider
LHCb	LHC Beauty Experiment
LSB	Least Significant Bit
LUT	Look-Up Table
LVS	Layout versus Schematic Check
MCM	Multi Chip Module
MIMD	Multiple Instruction Multiple Data
PASA	Preamplifier and Shaper Chip
PC	Personal Computer

PCB	Printed Circuit Board
PHOS	Photon Spectrometer
PMD	Photon Multiplicity Detector
PRF	Pad Response Function
PS	Proton Synchrotron
RHIC	Relativistic Heavy Ion Collider
RMS	Root Mean Square
SCSN	Slow Control Serial Network
SNR	Signal to Noise Ratio
SPS	Super Proton Synchrotron
TOF	Time of Flight Detector
TPC	Time Projection Chamber
TRAP	Tracklet Processor Chip
TRD	Transition Radiation Detector
UMC	United Microelectronics Corporation
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
ZDC	Zero Degree Calorimeter

Bibliography

[aa0] A. Andronic: ALICE TRD prototype tests (http://www-linux.gsi.de/ andronic/trd/tests)

[aa1] A. Andronic: Electron identification performance with ALICE TRD prototypes (NIM A, 522 (2004), 40-44)

[aa2] A. Andronic: TRD Experiment Logbook - Oct.11 - Nov.5, 2004 (http://www-linux.gsi.de/∼andronic/trd/tests/oct04/index.html)

[acex] V. Angelov: The ACEX-Board - A Multipurpose Testboard for Experiments and Exercises (University of Heidelberg 2002; http://www.kip.uni-heidelberg.de/ti/ACEXBoard)

[ac0] ALICE Colaboration: Position Reconstruction in Drift Chambers operated with Xe, CO_2(15%) (NIM A, 540 (2005), 140-157)

[alice] A Large Ion Collider Experiment (ALICE) at CERN LHC (http://alice.web.cern.ch/Alice)

[aliroot] The TRD in the ALIROOT Package (http://www-aix.gsi.de/∼alice/aliroot/index.html)

[altro] R. Esteve Bosch, A. Jimnez de Parga, B. Mota, and L. Musa: The ALTRO Chip: A 16-channel A/D Converter and Digital Processor for Gas Detectors (IEEE Transaction on Nuclear Science, Vol. 50 N. 6, December 2003)

[atw] WIKI of the Alice TRD Project (www.kip.uni-heidelberg.de/wiki/TRD)

[aw0] A. Walker: Development of a Monitoring Software for the ALICE TRD Prototype Stack (Projektpraktikum, University of Heidelberg 2005)

[bv0] B. Vulpescu: Position determination in the ALICE TRD (Physikalisches Institut; University of Heidelberg 2001)

[bv1] B. Vulpescu: CERN 2004 Beam Test Analysis
(Physikalisches Institut; University of Heidelberg 2005;
http://www-linux.gsi.de/ vulpescu/CERN04/)

[cr0] C. Reichling: Entwicklung eines Quadport Memory für den Einsatz in einem Triggerprozessor am CERN
(Diploma Thesis, University of Heidelberg 2001)

[dbv0] B. Jähne: Digitale Bildverarbeitung
(4. Auflage, Springer; Berlin, Heidelberg 1997)

[de0] D. Emschermann: Pad dimensions for the TRD modules
(University of Heidelberg 2005;
http://www.physi.uni-heidelberg.de/∼demscher/alice/padplane/)

[dm0] D. Muthers, R. Tielert: A 0.11mm^2 Low-Power A/D-Converter Cell for 10b 10MS/s Operation
(Proceedings of the ESSCIRC 2004, p. 251-254)

[ff0] F. Ferner: Development of a Test Environment for the ALICE TRD Readout Chip (Diploma Thesis, University of Heidelberg 2005)

[fl0] F. Lesser: Entwurf und Realisierung eines vierfach MIMD Prozessor Mikrochips für eine Anwendung in der Hochenergiephysik
(Dissertation, University of Heidelberg 2002)

[fl1] F. Lesser: A MIMD Multi Threaded Processor
(Proc. 13th IEEE HotChips Conference; Palo Alto, USA, August 2001)

[ir0] I. Rusanov: Production Readiness Review of ALICE TRD Multi Chip Module and Readout Board
(University of Heidelberg, December 2004)

[jc0] J. de Cuveland: Entwicklung der globalen Spurrekonstruktionseinheit für den ALICE-Übergangsstrahlungsdetekor am LHC (CERN)
(Diploma Thesis, University of Heidelberg 2003)

[jedi] J. D. Jackson: Klassische Elektrodynamik
(2. Auflage, de Gruyter; Berlin, New York 1982)

[kk] K. Kleinknecht: Detektoren für Teichenstrahlung
(3. Auflage, Teubner; Stuttgart 1992)

[kl0] Lehrstuhl Grundlagen der Mikroelektronik, Prof. Dr.-Ing. R. Tielert
(http://www.eit.uni-kl.de/tielert/)

[ko0] K. Oyama, C. Lippmann: Status of TRD Chamber and Readout Electronics Integration (Transition Radiation Detectors - Present and Future; International Workshop, Cheile Gradistei, Romania, 2005; http://dracula.nipne.ro/cmsimple2_4/?International_Workshop)

[mg0] Marcus Gutfleisch: Digitales Frontend und Preprozessor im TRAP1-Chip des TRD-Triggers für das ALICE-Experiment am LHC (CERN) (Diploma Thesis, University of Heidelberg 2002)

[mhw] J. Krautter: Meyers Handbuch Weltall
(7. Auflage, Meyers Lexikonverlag; Mannheim, Leipzig, Wien 1994)

[pasa0] H.K. Soltveit, J. Stachel: New Front-end Electronics for the ALICE TPC–Detector (GSI; Darmstadt 2002)

[pdg0] S. Eidelmann et al: Review on Particle Physics
(Phys. Lett. B **592**, 1 (2004); http://pdg.lbl.gov)

[phep] D. H. Perkins: Introduction to high energy physics
(4th edition, Cambridge University Press; Cambridge, New York 2000)

[ptk] Povh, Rith, Scholz, Zetsche: Teilchen und Kerne
(5. Auflage, Springer; Berlin, Heidelberg 1999)

[rg0] R. Gareus: Slow Control - Serial Bus and its implementation for the Transition Radiation Detectors's slow control network
(Diploma Thesis, University of Heidelberg 2002)

[rs0] R. Schneider: 64k Networked Multi-Threaded Processors and their Real–Time Application in High Energy Physics
(The 6th World Multiconference on Systemics, Cybernetics and Informatics; Orlando, USA, Juli 2002)

[rs1] R. Schneider: Entwicklung des Triggerkonzeptes und die entsprechende Implementierung eines 200 GB/s Auslesenetzwerks fr den ALICE Üergangsstrahlungsdetektor
(Dissertation, University of Heidelberg, in preparation)

[tdr] The ALICE Collaboration: ALICE Technical Design Report of the Transition Radiation Detector (LHCC 2001-021, CERN; Genf 2001)

[tm] ASICCC: ALICE TRAP User Manual
(Darmstadt, Heidelberg, Kaiserslautern, Mannheim 2005;
http://www.kip.uni-heidelberg.de/ti/TRD/doc/trap)

[ts0] T. Steinbeck: A Modular and Fault-Tolerant Data Transport Framework (Dissertation, University of Heidelberg 2004)

[va0] V. Angelov: ALICE TRD Trigger Architecture
(TRDs for the 3rd Millenium; Bari, Italien, September 2001)

[vl0] V. Lindenstruth, I. Kisel: Overview of trigger systems (NIM A, 535 (2004), 48-56)

[xt0] J. Colas, R. Lafaye, N. Massol, C. de La Taille, L. Serin, P. Pralavorio, D. Sauvage: Crosstalk in the ATLAS Electromagnetic Calorimeter (ATL-LARG-2000-004, CERN; Genf 2000)