

## Bachelor's Thesis

# Testmessungen mit aktiven Pixelsensoren für den Ausbau des ATLAS Pixeldetektors

## Test measurements with active pixel sensors for the upgrade of the ATLAS pixel detector

prepared by

**Tobias Wolfgang Günter Westphal**

from Nienburg/Weser

at the II. Physikalischen Institut

**Thesis number:** II.Physik-UniGö-BSc-2015/08  
**Thesis period:** 13th April 2015 until 20th July 2015  
**First referee:** Prof. Dr. Arnulf Quadt  
**Second referee:** Priv.-Doz. Dr. Jörn Große-Knetter



# Contents

<b>1. Introduction</b>	<b>1</b>
<b>2. Experimental Setup</b>	<b>3</b>
2.1. Basics of the Large Hadron Collider . . . . .	3
2.2. The ATLAS Experiment . . . . .	4
2.2.1. Inner Detector . . . . .	5
2.2.2. Calorimeter System . . . . .	6
2.2.3. Muon Chambers . . . . .	8
<b>3. Silicon Pixel Sensors</b>	<b>9</b>
3.1. n-in-p Design . . . . .	9
3.2. Active Sensor . . . . .	10
<b>4. Measurement Setup</b>	<b>13</b>
4.1. Front-End I4 . . . . .	13
4.2. USBpix Read-Out System . . . . .	14
4.3. STcontrol . . . . .	17
4.4. Bonn ATLAS Readout System . . . . .	17
4.5. Standard Scans . . . . .	18
<b>5. Results</b>	<b>23</b>
5.1. STcontrol Tuning . . . . .	23
5.2. pyBAR Tuning . . . . .	24
5.3. Comparison of STcontrol and pyBAR . . . . .	25
5.3.1. Injection Calibration . . . . .	25
5.3.2. Threshold Scan . . . . .	26
5.3.3. ToT Verification . . . . .	30
5.3.4. Complete Threshold Tuning . . . . .	32
5.4. Scan/Tuning Duration . . . . .	32
5.5. Active Sensor . . . . .	35

<b>6. Discussion</b>	<b>39</b>
6.1. Comparison of STcontrol and pyBAR . . . . .	39
6.1.1. Injection Calibration . . . . .	39
6.1.2. Threshold Scan . . . . .	39
6.1.3. ToT Verification . . . . .	40
6.1.4. Complete Threshold Tuning . . . . .	41
6.2. Scan/Tuning Duration . . . . .	41
6.3. Active Sensor . . . . .	42
<b>7. Conclusion and Outlook</b>	<b>43</b>
<b>A. Additional Plots</b>	<b>45</b>

# Nomenclature

## Acronyms

Acronyms	Meaning
ATLAS	A Toroidal LHC Apparatus
CCPD	Capacitive Coupled Hybrid Pixel Detector
CERN	Organisation européenne pour la recherche nucléaire
CMS	Compact Muon Solenoid
DAC	Digital Analog Converter
ECal	Electromagnetic Calorimeter
FCal	Forward Calorimeter
FDAC	In-Pixel Preamplifier Feedback Current
FE	Front-End
FE-I3	Front-End Interface 3
FE-I4	Front-End Interface 4
FPGA	Field Programmable Gate Array
GDAC	Global Discriminator Threshold Input
GPAC	General Purpose Adapter Card
GUI	Graphical User Interface
HCal	Hadronic Calorimeter
HVCMOS	High Voltage Complementary Metal-Oxide-Semiconductor
IBL	Insertable B-Layer
IF	Global Preamplifier Feedback Current
LHC	Large Hadron Collider
m.i.p.	minimum ionizing particle
PCB	Printed Circuit Board
PslrDAC	Calibration Injection Voltage Value
pyBAR	Bonn ATLAS Readout in Python and C++

## *Nomenclature*

<b>Acronyms</b>	<b>Meaning</b>
SCC	Single Chip Card
SCT	Silicon Microstrip Tracker
SM	Standard Model
SPS	Super Proton Synchrotron
TDAC	In-Pixel Discriminator Threshold Input
ToT	Time over Threshold
TRT	Transition Radiation Tracker

# 1. Introduction

The newest discovery in particle physics is a particle which shows all properties of the Higgs boson until now. This discovery was made 2012 at CERN with the ATLAS and CMS [1, 2] experiment, which are both part of the Large Hadron Collider (LHC). The search for this particle took about 50 years.

The Higgs boson is the last missing particle of the Standard Model (SM), which explains the mass of the elementary particles. Beside the Higgs boson, the elementary particles in the SM are six quarks and six leptons, structured in three generations. The four gauge bosons belonging also to the SM, mediate the fundamental interactions of the three different forces, the electromagnetic, weak and strong force.

To reconstruct those particles, detectors are necessary, which measure mostly the decay products of the particles. The ATLAS detector is one of the experiments at the LHC. The ATLAS detector and the LHC are described in Chapter 2.

A part of the particle detectors are pixel detectors, which are able to detect charged particles. Current high energy particle physics pixel detectors are hybrid modules with a read-out chip and a sensor made out of silicon. A charged particle crossing the sensor creates electron-hole pairs that can be detected. The mechanism of the silicon pixel detectors is explained in detail in Chapter 3. To use pixel detectors, software and hardware is necessary in order to be able to do the calibration, tuning and read-out of those pixel modules.

In this thesis, two read-out systems are compared. At first with a well-known prototype module and later with a newly developed module with active sensor, that is investigated for usability for the upgrade of the LHC to the High Luminosity - Large Hadron Collider (HL-LHC) around 2023. The different setups which are used for the measurements of this thesis are described in Chapter 4.

Both read-out systems are compared in Chapter 5 with the results of different scans and tunings.





## 2. Experimental Setup

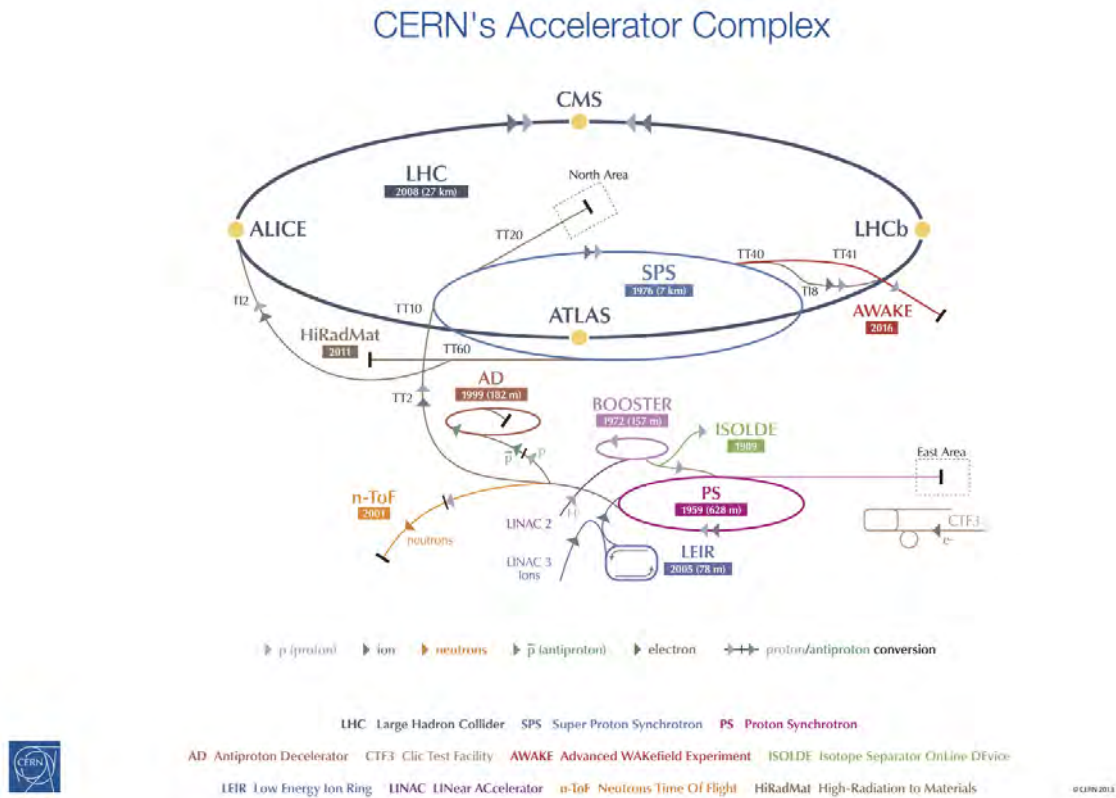
### 2.1. Basics of the Large Hadron Collider

The Large Hadron Collider (LHC) is an accelerator, which was designed for proton-proton collisions. With a circumference of 26.7 km, the LHC is the world largest accelerator [3]. The ring is placed in an underground tunnel system close to Geneva in Switzerland and is a part of CERN (Organisation européenne pour la recherche nucléaire). In Run I, the centre of mass energy was 7 TeV, respectively 8 TeV (2012). This spring (2015), Run II started with a planned centre of mass energy of up to 13 TeV [3, 4]. To reach those high energies, the protons have to be pre-accelerated. The pre-acceleration is done by accelerators at the CERN, which were built before the LHC. The last stage of pre-acceleration is the Super Proton Synchrotron (SPS), which accelerates the particles up to 450 GeV and passes them with this energy to the LHC. A schematic structure of the LHC with its pre-accelerators is shown in Figure 2.1.

Because the LHC should create proton-proton collisions, there is a need of two tubes with an anti-polar magnetic field, so that the protons can be accelerated in opposite directions. Beside proton-proton collisions, the LHC is also able to perform ion-ion collisions of lead ions. There are four interaction points, where the main experiments are placed. These are A Large Ion Collider Experiment (ALICE), A Toroidal LHC ApparatuS (ATLAS), Compact Muon Solenoid (CMS) and Large Hadron Collider beauty experiment (LHCb). ATLAS and CMS are general purpose detectors with the designed physics goals of searching for a Higgs boson, Dark Matter candidates and signatures for supersymmetry. LHCb focuses on matter-anti matter balance and ALICE on heavy ion collisions.

An important value for a collider is its luminosity  $\mathcal{L}$ . It describes how many collision are taking place per surface and time. During Run I, the instantaneous luminosity was around  $\mathcal{L} = 10^{34} \text{ cm}^{-2}\text{s}^{-1}$  [3]. In Run II, there will be a higher luminosity.

## 2. Experimental Setup



*Figure 2.1.:* Sketch of the accelerator complex at CERN [© 2013 CERN ].

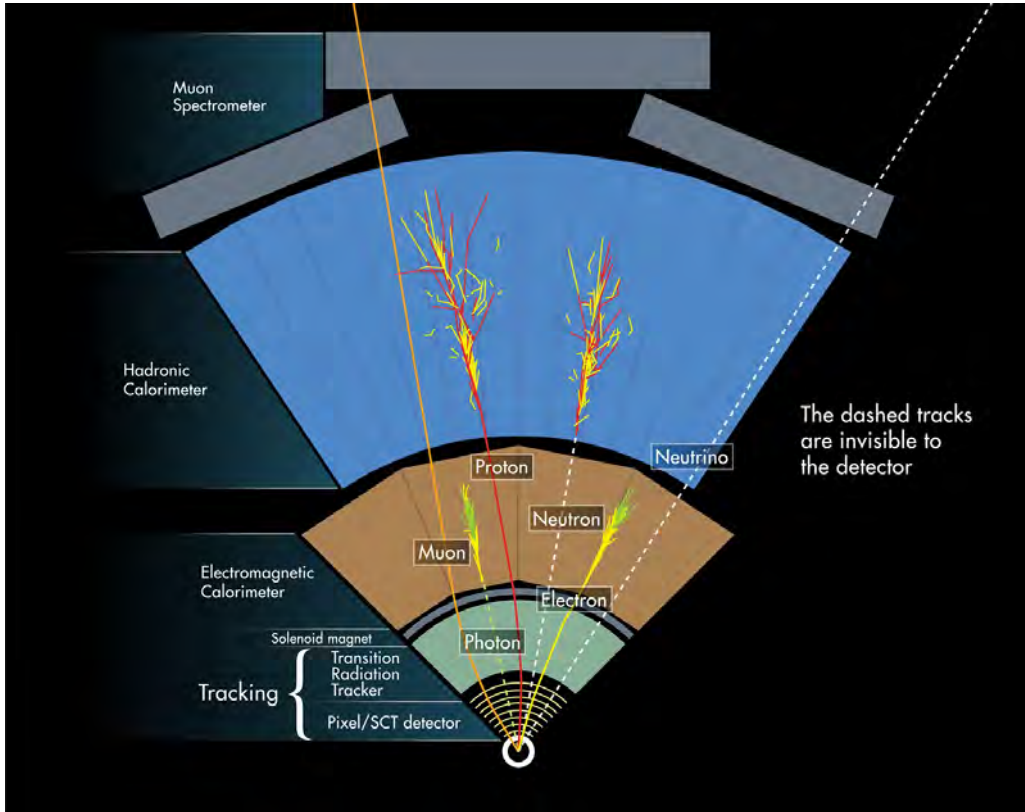
## 2.2. The ATLAS Experiment

In the following, ATLAS and its functionality is described, to give a general understanding of this detector. The detector is a general purpose detector, which covers almost the full solid angle.

The ATLAS detector consists of three main detector systems - the Inner Detector with several layers of tracking, the calorimeter and the muon chamber. A profile of the layout of the three main detector systems is shown in Figure 2.2.

In total, ATLAS has a length of 44 m with a height of 25 m and a weight of 7000 t [5]. The structure of the most parts in the detector is split into a central part with a barrel structure and a forward part with an end-cap structure. A sketch of the whole detector is shown in Figure 2.3.

At the ATLAS detector, a right-handed coordinate system is used, where the beam direction is defined as  $z$ -axis. The  $x$ -axis points from the interaction point in the centre of the ATLAS detector to the centre of the LHC ring, the  $y$ -axis points upwards. The azimuthal  $\phi$  angle lies in the  $x/y$ -plane and the polar angle  $\theta$  is measured from the beam



**Figure 2.2.:** Sector of the arrangement of the ATLAS detector system with illustrations which particles are detected in which area [ATLAS Experiment © 2014 CERN ].

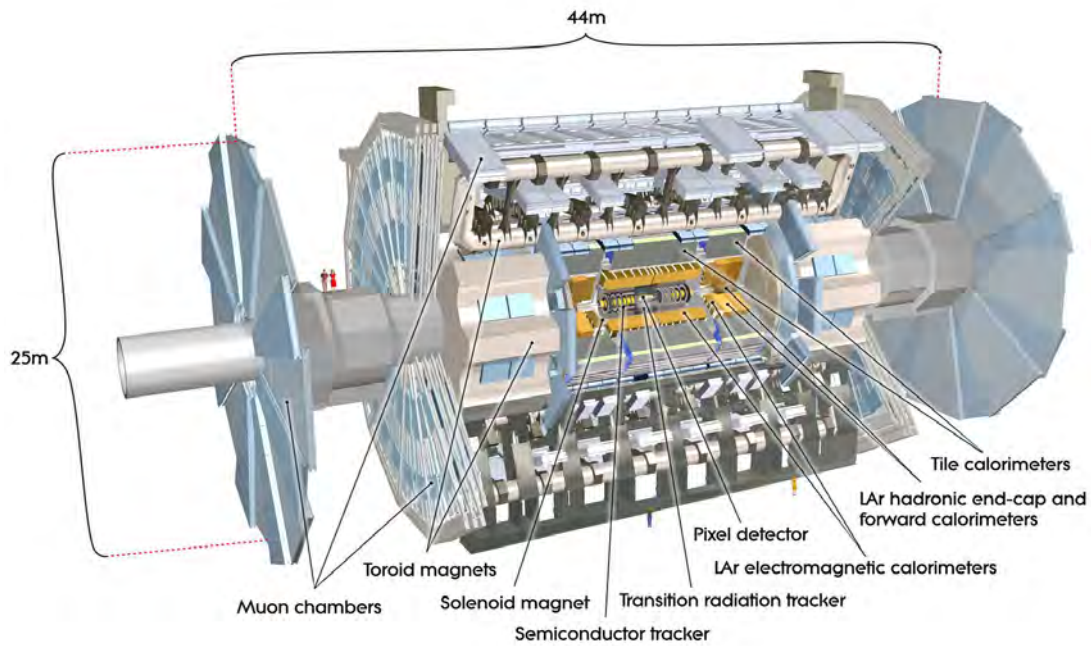
axis [5]. As additional specification the rapidity or the pseudo-rapidity is often used. In particle physics, the rapidity  $y$  is a measure of the particle velocity, which is defined relative to the beam axis as  $y = \frac{1}{2} \cdot \ln \left( \frac{E+p_L}{E-p_L} \right)$  with  $E$  as energy and  $p_L$  as longitudinal momentum component [6]. Instead of the rapidity the pseudo-rapidity  $\eta = -\ln \left( \tan \left( \frac{\theta}{2} \right) \right)$  with the approximation  $p \gg m$  is often used, because the pseudo-rapidity depends only on the polar angle of the particle's trajectory, and not on the energy of the particle. In addition the differences in pseudo-rapidity  $\Delta\eta$  are Lorentz invariant under boosts along the longitudinal axis, which is useful to compare results of different colliders.

In the following sections the detector systems are briefly explained.

### 2.2.1. Inner Detector

The Inner Detector is used to measure the momentum of charged particles. Therefore it is built up with three systems, the Silicon Pixel Detector, the Silicon Microstrip Tracker (SCT) and the Transition Radiation Tracker (TRT). All three systems are enclosed in a 2 T solenoidal field for momentum determination and charge separation. Closest to the

## 2. Experimental Setup



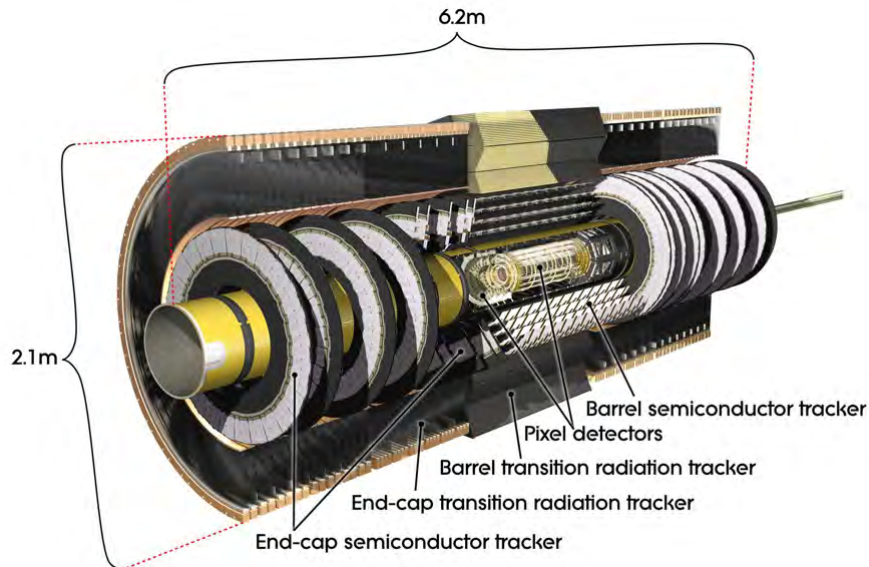
*Figure 2.3.:* Sketch of the ATLAS detector and its major components [5].

beam pipe is the Pixel Detector, which is surrounded by the SCT, the outermost tracking system in the Inner Detector is the TRT. The Pixel Detector and the SCT are using silicon sensors for measuring the particle track and differ mainly in the spatial resolution and their placement. If a charged particle crosses a silicon detector, it produces electron hole pairs, which leads in the end to a charge signal. A more detailed explanation of the silicon detectors in the Pixel Detector can be found in Chapter 3. The Pixel Detector provides the information for tracking, vertex reconstruction and b-tagging. At the TRT a straw tube detector is used, consisting of many parallel arranged straw tubes. These tubes are filled with gas mixture of Xe, CO<sub>2</sub> and O<sub>2</sub> [7]. The whole tracking system is shown in Figure 2.4.

In Run I, the Pixel Detector consists of three layers. For Run II, the Pixel Detector received an upgrade. An additional layer, the Insertable B-Layer (IBL), close to the beam pipe was added. To create the space for this subdetector the diameter of the beam pipe was shrunk.

### 2.2.2. Calorimeter System

The calorimeter system measures the energy of the particles. Therefore it is important, that the electromagnetic and hadronic showers are stopped completely in the calorimeter system. The first part of calorimeter system is the electromagnetic calorimeter (ECal),



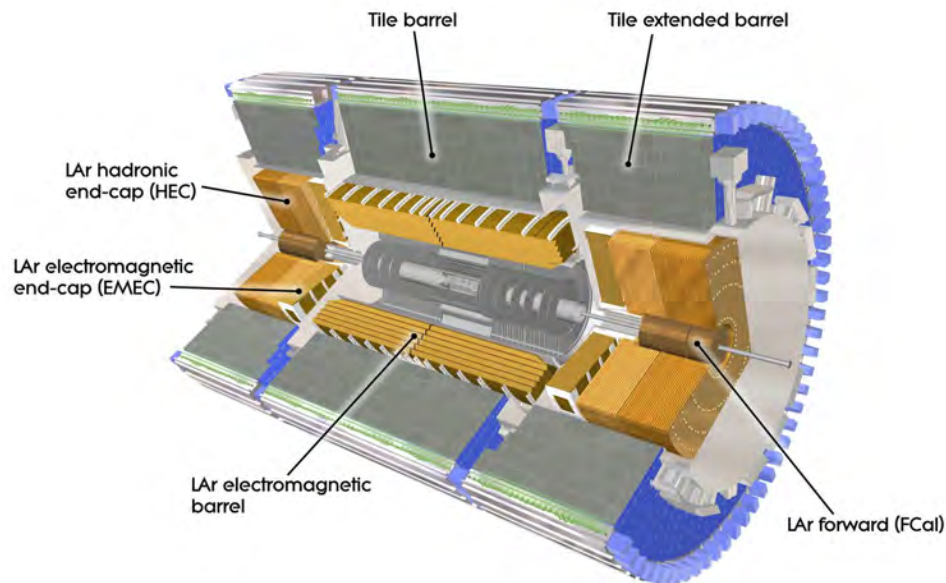
**Figure 2.4.:** Showing the arrangement of the components in the inner detector [5].

where the most electrons and photons are stopped via electromagnetic showering. After the ECal is the hadronic calorimeter (HCal), which is needed to stop the hadronically showering particles. The distance after a particle loses the energy of factor of  $1/e$  is called radiation length  $X_0$  for electromagnetic interactions and nuclear interaction length  $\lambda_0$  for hadronic interactions.

There are two types of calorimeters homogeneous calorimeters and sampling calorimeters. At ATLAS, only sampling calorimeters are used because of easier handling and lower costs. The ATLAS calorimeter system is shown in Figure 2.5.

In the electromagnetic calorimeter liquid argon is used as read-out material and lead and stainless steel are used as absorber material. The hadronic calorimeter uses steel in the barrel area as absorber material and plastic as scintillator, in the end-cap structure copper is the absorber material and liquid argon (LAr) is used as read-out material [5]. In the forward calorimeter (FCal) copper is used in the electromagnetic and wolfram in the hadronic part as absorber materials, in both parts liquid argon is used as active material. The electromagnetic calorimeter is approximately  $20 X_0$  long in the range of  $|\eta| < 1.4$  and approximately  $30 X_0$  long in the range of  $1.4 < |\eta| < 3.2$ . The combined calorimeter system of electromagnetic, forward and hadronic calorimeter has a thickness of approximately  $10 \lambda_0$  [5].

## 2. Experimental Setup



*Figure 2.5.:* Structure of the calorimeter system of the ATLAS detector [5].

### 2.2.3. Muon Chambers

Muons and for the detectors non-visible neutrinos are the only particles which are not stopped by the calorimeters. The track of the muons is detected by the Muon Chamber, placed behind the calorimeter system. With an additional toroidal magnetic field placed outside the calorimeter, causing a curvature of the muon tracks, it is possible to measure the momentum of the muons. There are four different tracking systems: Monitored drift tubes, Strip Chambers, Resistive plate chambers and Thin-gap chambers. The tracking information of the muons are combined with the data from the inner detector to get a combined muon track [5].

## 3. Silicon Pixel Sensors

When a charged particle hits the depletion zone of a semiconductor, it loses energy mainly by ionisation. Via the ionisation process electron-hole pairs are produced. Silicon is the most common semiconductor for electronic usage. The amount of energy  $E$  lost per distance  $x$  is described by the BETHE-BLOCH equation [8]

$$-\left\langle \frac{dE}{dx} \right\rangle = 2\pi N_A r_e^2 m_e c^2 \rho \frac{Z}{A} \frac{z^2}{\beta^2} \left[ \ln \left( \frac{2m_e c^2 \beta^2 \gamma^2 W_{\max}}{I^2} \right) - 2\beta^2 - \delta - 2\frac{C}{Z} \right], \quad (3.1)$$

where  $N_A$  is the AVOGADRO's number,  $r_e$  the classical electron radius,  $m_e$  the electron mass,  $c$  the speed of light in vacuum,  $\rho$  the density of the absorbing material,  $Z$  the atomic number of the absorbing material,  $A$  the atomic weight of absorbing material,  $z$  the charge of an incident particle in units of  $e$ ,  $W_{\max}$  the maximum energy transfer in a single collision,  $I$  the mean excitation potential,  $\delta$  the density correction and  $C$  the shell correction.

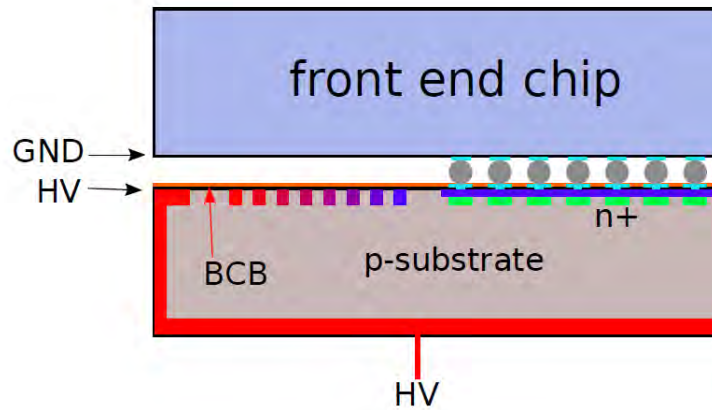
This function has a minimum at  $\beta\gamma=3.5$  with an energy loss of  $\left\langle \frac{dE}{dx} \right\rangle_{\min} \approx 1.5 \frac{\text{MeV}}{\text{g cm}^{-2}}$ . Most of the particles and absorber materials used in particle physics have an energy loss close to the ideal minimum, they are summarized as minimum ionizing particles (m.i.p.).

To detect charge in the depletion zone, an external voltage is applied, to create a drift field. The signal is created by the drifting charges in the read-out electrodes [9]. Because of the fact, that the depletion zone is free of thermal charge carriers, a high voltage is applied on the sensor, that nearly the whole sensor is depleted. For a m.i.p. passing a 200  $\mu\text{m}$  thick silicon sensor vertically, a signal of about 16000  $e$  is expected.

### 3.1. n-in-p Design

The sensor used on the prototype module is a planar n-in-p sensors. They consists of oxygenated silicon p-type bulk material with  $n^+$  pixel implants on the front side and a large p-type implantation on the back side [10]. Those sensors are also used for this thesis.

The n-in-p design is potentially cheaper to produce in comparison to the previous



*Figure 3.1.:* Illustration of the structure of a n-in-p sensor [11].

common design of n-in-n sensors, due to the fact, that the n-in-p sensors just need to be processed from one side.

The drop between high voltage and the grounded front-end chip is on the side facing the electronics (see Figure 3.1). To avoid shorts at this drop, a guard ring structure is needed to be added to the n-in-p design. The guard ring is placed on the front side of the sensor and is covered with an insulating film like Benzo Cyclo Butene (BCB).

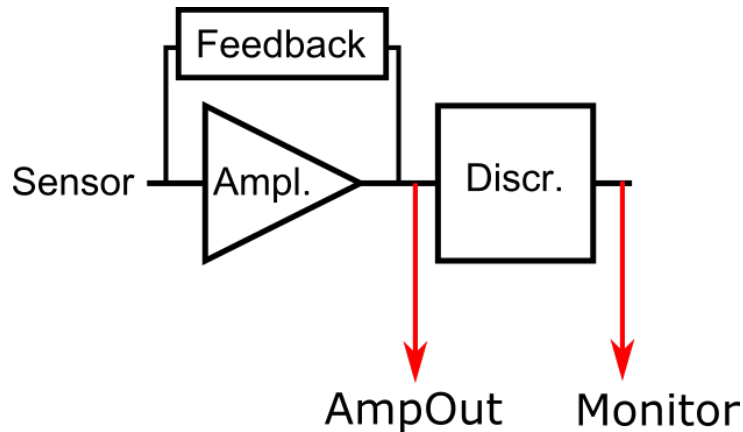
## 3.2. Active Sensor

Looking towards to the next upgrade of the LHC to the High Luminosity - Large Hadron Collider (HL-LHC) four or even five layers of pixel detectors are planned. This means, that the area, that needs to be covered with the pixel detector is increasing. Due to the fact, that the budget for detector building is limited, it is interesting to find more cost efficient detectors.

One option are active CMOS (Complementary Metal-Oxide-Semiconductor) sensors with an external voltage applied. They are called HVCMOS (High Voltage - CMOS) sensors [12]. These sensors are active, which means, that there are transistors on the sensor that can already amplify the signal. Until now the used sensors are passive, meaning the signal is created in the sensor and is then passed without any amplification to the read-out electronics. The CMOS process is an industrial standard and because of this, it is easily available and very cost efficient.

The depletion zone of the HVCMOS sensor used in this thesis is only around 10  $\mu\text{m}$ -15  $\mu\text{m}$  thick. There exists the problem that these sensors are less radiation hard because of the logic on the sensor, but at least the effect of charge trapping is reduced because





**Figure 3.2.:** Schematic circuit of a pixel of the active sensor. AmpOut and Monitor are pins on the board of the sensor.

the electrodes are closer.

The high voltage is needed, to have drifting charges. Without the high voltage the sensor is also working, because of the doping of the sensor, but then the charge carriers move only by diffusion, which is too slow for the high luminosity at LHC.

There are two ways of interconnection between sensor and read-out chip, direct coupling e.g. bump bonds, and capacitive coupling, e.g. glue. It could be, that for capacitive coupling less material is necessary, which would be good for the calorimeter. The connection for the communication with the active sensor is done via wire bonding in the test setup. The resulting modules are called capacitive coupled hybrid pixel detectors (CCPDs). The used chip is a HV2FEI4 in the AMS 180 nm HVCMOS process [13]. The pixels on this sensor have a size of  $33\ \mu\text{m} \times 125\ \mu\text{m}$ , which means, that three of the active sensor pixels are connected to one front-end pixel. For the test setup the active sensor has  $24 \times 60$  pixel, which covers only a small area of the used read-out chip.

A schematic circuit of a pixel of the active sensor is shown in Figure 3.2.



# 4. Measurement Setup

For the measurements in the laboratory, a system is needed which is able to access and read-out the front end electronics. This system needs to configure the test assembly and to run standardised tests. Therefore, the USBpix read-out system is a modular read-out system, which has a high flexibility to work with different types of sensors. This system, the used software and scans are described in the following.

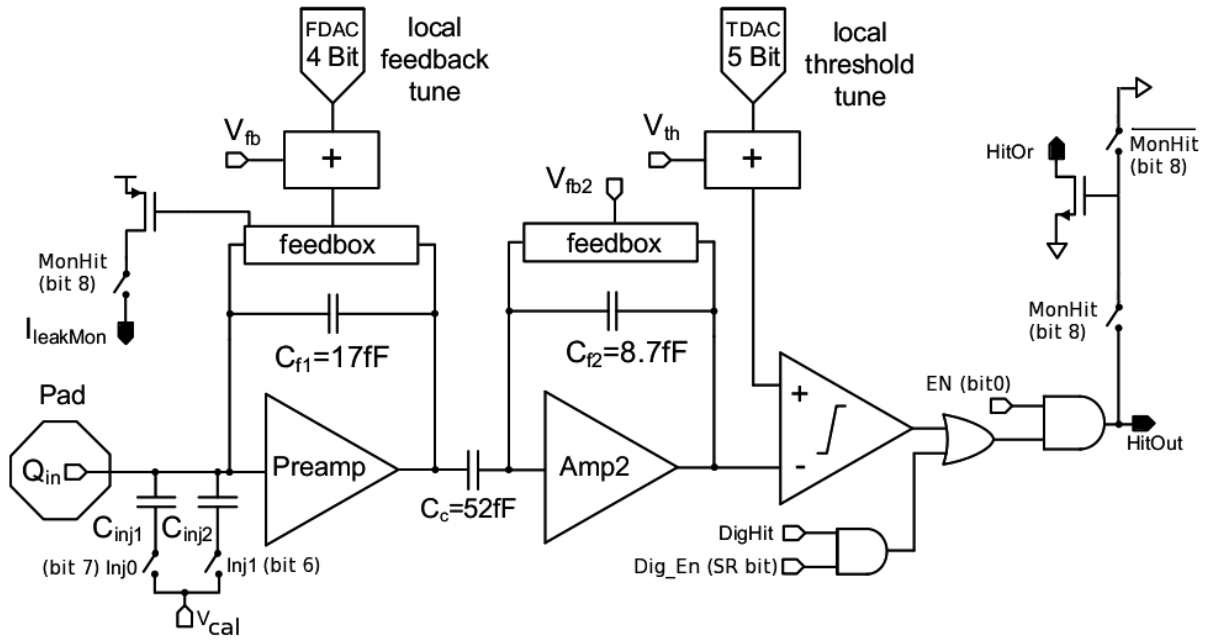
## 4.1. Front-End I4

The Front-End (FE) is the read-out chip, which processes the signal of every single pixel. The FE-I4 is used in the IBL, the other three pixel layers in the Pixel Detector are working with the older version, the FE-I3. To process the signal, there is a circuit with amplifiers and a discriminator for each sensor pixel as shown in Figure 4.1. The connection between the sensor and the front-end is done by bump-bonding, an illustration of this is shown in Figure 3.1. The FE-I4 has 336 rows and 80 columns, in total there are 26880 pixel, the size of one pixel cell is  $50 \times 250 \mu\text{m}^2$  [14].

In the FE electronics there is an additional circuit to inject a well-defined charge in each pixel cell. This is helpful for test and tuning purposes without the need of radioactive source. The charge is generated by applying a voltage pulse to a capacitor. There are two different capacitors  $C_{\text{inj}1} = 1.9 \text{ fF}$  and  $C_{\text{inj}2} = 3.9 \text{ fF}$  that can be selected, as shown in Figure 4.1. Both capacitors can be selected at the same time, which leads to a capacity of  $C_{\text{tot}} = 5.7 \text{ fF}$ . The amplitude of the voltage pulse is controlled by the 10-bit `PlsrDAC` value.

There are two circuits of amplification. The first is the pre-amplification connected to the 4-bit `FDAC` to adjust the in-pixel feedback current while tuning. This pre-amplifier is there to shape the signal. At the second amplifier a voltage gain is provided and the signal keeps its shape. After the amplification is the discriminator, which digitizes the analogue signal. The discriminator is connected to the 5-bit `TDAC`, which is used to control the in-pixel threshold in the discriminator to a certain value. When the signal is above the threshold the time is measured until the value drops below the threshold, as shown

## 4. Measurement Setup



**Figure 4.1.:** Schematic diagram of the analogue part of a pixel cell [14].

in Figure 4.2. The Time-over-Threshold (ToT) is measured in 25 ns units, which is the bunch crossing time; e.g. a ToT of 5 means that the signal is 125 ns above the Threshold. With the knowledge of the threshold value and the slope of the rising and the falling edge, it is possible to calculate which charge the particle produces in the sensor.

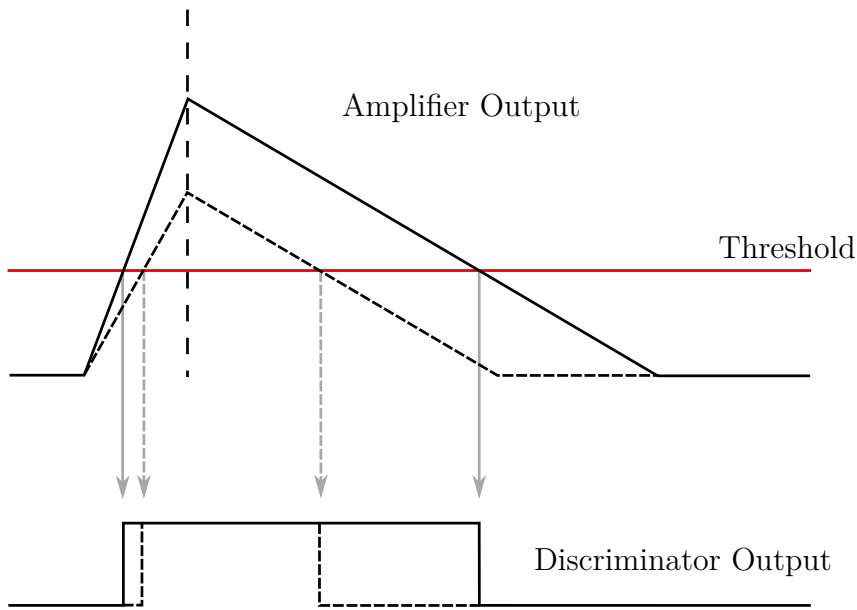
The global threshold of a chip is controlled by two 8-bit values  $V_{thin\_Alt\_Coarse}$  and  $V_{thin\_Alt\_Fine}$ , which are for coarse and fine adjustment. The global feedback current is controlled by the 8-bit value  $PrmpVbpf$ , the step size is set by the 8-bit value  $FdacVbn$ . All the described bit values are set in the configuration of the chip, the in-pixel values are stored in different pixel registers, the global values are set in global register.

There are two versions of the FE-I4, the FE-I4A and the FE-I4B. The FE-I4A is a prototype, with special test columns at the double columns 1, 39 and 40, the FE-I4B is used in the IBL.

## 4.2. USBpix Read-Out System

The USBpix read-out system [15, 16] is a system which is able to operate as a read-out system for FE-I3 and FE-I4 read-out chips. Two different setups were used during the thesis. One setup with the hybrid module with a planar n-in-p sensor and a FE-I4A and the other with the active HVCMOS sensor glued on a FE-I4B.

The setup with the hybrid module with a planar n-in-p sensor consist of a Multi-IO



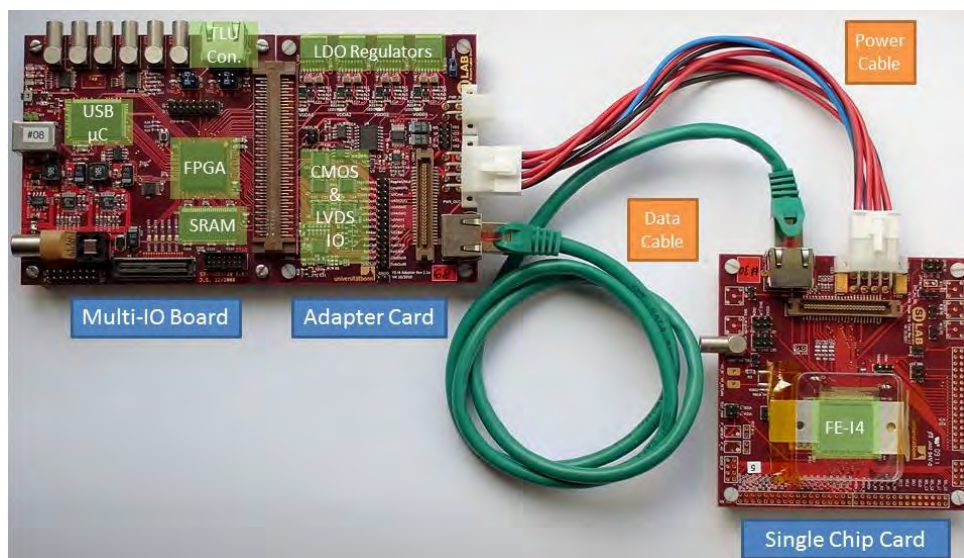
**Figure 4.2.:** Illustration of the conversion process “time over threshold” in the discriminator.

board (MIO), which is connected via USB to the computer, an Adapter Card and the Single Chip Card (SCC), this setup is shown in Figure 4.3. The MIO board is connected to the adapter card which is then connected via an RJ-45 cable to the SCC. Connecting the FE to the SCC is done by wire-bonding. The sensor itself is connected via bump bonding to the read-out chip (the FE-I4) as explained in Chapter 4.1.

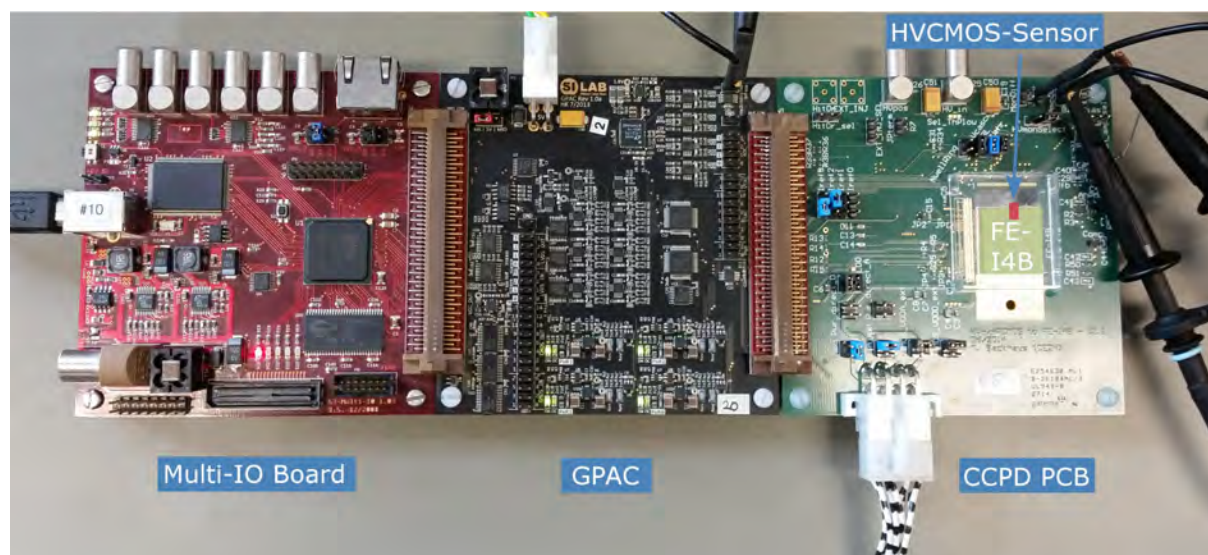
The supply voltage of the adapter card is 2.0 V, the sensor board is supplied with 1.2 V as digital supply voltage and 1.5 V as analogue supply voltage. In addition, a high voltage of -51 V is applied to the sensor to form the depletion zone.

The other setup is for the active sensor (see Figure 4.4). It contains also the MIO board, which is now connected to a General Purpose Adapter Card (GPAC). The CCPD Printed Circuit Board (PCB) is connected to the GPAC. To supply the GPAC, a voltage of 5.0 V is needed, the FE regulators are supplied with 2.2 V. The HVCMOS is powered with different supply voltages via the GPAC.

#### 4. Measurement Setup



*Figure 4.3.:* The USBpix read-out system, which consists of the Multi-IO board, an Adapter Card and a Single Chip Card with a FE-I4A on [15].



*Figure 4.4.:* Measurement setup, with Multi-IO Board, GPAC and CCPD PCB with a FE-I4B with a capacitive coupled HVC MOS.

Name of scan/tuning in STcontrol	Description
GDAC_TUNE	Global threshold tuning
TDAC_TUNE	In pixel threshold tuning
IF_TUNE	Global feedback tuning
FDAC_TUNE	In pixel feedback tuning
ANALOG_TEST	Analog scan
DIGITAL_TEST	Digital scan
THRESHOLD_SCAN	Threshold scan
TOT_VERIF	Time-over-Threshold verification

**Table 4.1.:** Selection of scans and tunes, which were used in STcontrol.

### 4.3. STcontrol

STcontrol is a read-out and tuning software with a graphical user interface (GUI). It is written in C++ and is able to work with either the FE-I3 or the FE-I4. The program can execute calibrations, scans and tunings, which are used to tune the FE and to check how the chip responds to a signal. The measured data is stored in `DBroot` files, but there is a tool, which can be used to create `root` files from the stored data. The used version of STcontrol for this thesis is the release 5.3.

The interface of STcontrol is split into two panels. On the left panel is the module view, where for example the configuration of the front-end can be accessed and the different voltages of the chip are shown. The right panel contains the action panels, which are used to edit the settings of tests and scans and execute them. Also a data viewer, to access the scan results, is one of the possible panels for the right panel.

In Table 4.1 are the scans and tunings listed which were used in this theses, but there are more scans and tunings in STcontrol included.

For using the active sensor with STcontrol, there exists an addition to the release 5.3, which adds an Analog Scan specialised for the active sensor, some index tabs for the configuration and a modified FPGA configuration that supports the CCPD.

### 4.4. Bonn ATLAS Readout System

The Bonn ATLAS Readout in Python and C++ (pyBAR) is a readout and test system for FE-I3 and FE-I4 chips. It uses the basil framework to access the readout hardware. In general it has the same features as STcontrol. pyBAR is script based and the output data is stored in the HDF5 file format.

The software is developed at the SiLab at the University of Bonn. Due to the fact that

## 4. Measurement Setup

it is under development, the description in this thesis is based on the development branch from the May, 6th, 2015 with the bug fixes #17 and #20 included. The used basil is the master branch of the May, 6th, 2015.

For the Pulser DAC calibration Scan (explained in section 4.5), a multimeter, which can be read-out by the software, is necessary. For using the multimeter with pyBAR some additions in the code are necessary.

To use the active sensor with pyBAR and extra software package is needed, which is called pyCPIX. There are different versions of this software, the used one is ccpdv2. This software needs the development branch of basil from the April, 27th, 2015, the pyBAR version needs no change. The paths in the pyBAR configurations files needs to be adjusted to link to the pyCPIX files.

### 4.5. Standard Scans

In the following, the standard scans used of STcontrol and pyBAR are described. Basically there are the same standard scans in STcontrol and pyBAR and to avoid confusion, the names of the scans in STcontrol are taken for both programs.

**Analog and Digital Test** These tests are used to check if the analogue and digital part of the pixel cells are working as expected. Therefore a charge can be injected several times into each pixel in the amplifier for the Analog Test or the discriminator for the Digital Test. The injection is done by the external charge injection circuit, to have a well-defined charge. A pixel works properly, if the number of hits which is read back is the same number as were injected.

It is not injected into all pixels the same time. There is a pre-set mask for both tests, that in every sixth pixel the charge is injected. This mask is shifted six times by one pixel, to address every pixel. This is done to avoid too high occupancy on the chip.

**Threshold Scan** To measure, which threshold each pixel cell has, the Threshold Scan is applied. While this scan charge is injected into the analogue part of the FE, the charge is increased step wise. For each charge step the charge is injected multiple times. The charge steps are controlled by the P1srDAC. The ideal result of this scan would be a step function with no hits below the threshold and above the threshold as many hits measured as injected. Because of noise effects, there is a fluctuation in the charge values and some values close to the threshold will cause a hit and some will not. This leads to the function fitted to the data, which is described by a convolution of a step function and a Gaussian



distribution. The probability to detected a hit  $p_{\text{hit}}$  is described by

$$p_{\text{hit}}(Q) = \frac{1}{2} \text{Erfc} \left( \frac{Q_{\text{thresh}} - Q}{\sqrt{2}\sigma_{\text{noise}}} \right) . \quad (4.1)$$

$Q$  is the charge of the injected pulse in DAC units,  $Q_{\text{thresh}}$  is the charge value, which is the threshold in the discriminator,  $\sigma_{\text{noise}}$  is the equivalent noise charge of the analogue part and the complementary error function Erfc is defined by

$$\text{Erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt . \quad (4.2)$$

The function which is described by Equation 4.1 is called S-curve due to its shape. This S-curve fit is applied on every pixel to get the distribution of the threshold over the chip. Therefore  $Q_{\text{thresh}}$  and  $\sigma_{\text{noise}}$  are stored. The stored data is plotted in three different plots, first in a colour-coded pixel map, second the values of all pixel on one chip in a histogram and third a scatter plot, where each pixel has its own identification number (ID) calculated by  $\text{ID} = \text{row} + 336 \cdot \text{column}$ , which is plotted against its value. The aim of the tuning is, to keep the threshold difference between the pixels as small as possible.

**ToT Verification** To check the ToT tuning from the IF and FDAC tune, the ToT verification is applied. During the tuning, the FE is tuned to respond to certain amount of charge with a certain ToT. For the ToT Verification this amount of charge is injected into the pixels. The injection procedure is the same as the Analog Test. In addition, the ToT value of each injection is readout. The result of the ToT for each pixel is the mean over the ToT values from the multiple injections of the pixel. In pyBAR this is integrated into the Analog Test, in STcontrol there is an extra scan called ToT\_Verif.

**Injection Calibration Scans** Before the tuning, the charge injection mechanism needs to be calibrated, in order to determine the charge which is injected by the injection circuit. The charge depends on the voltage pulse that is injected into the injection circuit (see Chapter 4.1). With the Pulser DAC Calibration Scan (DAC Scan) this voltage is measured. To run the DAC Scan a multimeter, which measures the voltage at the ExtAnInjectIn (a pin on the SCC), is necessary. Therefore the multimeter needs to be connected to the computer and the software has to be able to readout the values of the multimeter. The scan measures for each pulser DAC (0-1023) the voltage, this is controlled by the 10-bit P1srDAC value. This can be done in four different modes selected in the

#### 4. Measurement Setup

Colpr_Mode	Function
0	Scan only the addressed double-column
1	Scan every 4th double-column starting at addressed double-column
2	Scan every 8th double-column starting at addressed double-column
3	Scan all double-columns

**Table 4.2.:** The different modes for the pulser DAC calibration and their function [14].

global register with `Colpr_Mode`. The modes are explained in Table 4.2, the addressed double column can be set with `Colpr_Addr` (0-39). The injected charge  $Q_{inj}$  can be calculated after the scan by

$$Q_{inj} = C_{inj} \cdot (PlsrDAC + V_{Grad} + V_{offset}) . \quad (4.3)$$

$V_{Grad}$  is the slope of the linear fit, which is created from the measured data,  $V_{offset}$  the calibration offset and  $C_{inj}$  the chosen capacity.

The charge has an offset  $Q_{offset} = C_{inj} \cdot V_{offset}$ . In STcontrol there is an extra scan called Injection Calibration to determine the offset  $V_{offset}$ . The Injection Calibration Scan executes three Threshold Scans, with the three capacitor settings  $C_{inj1}$ ,  $C_{inj2}$  and  $C_{inj1} + C_{inj2}$  to determine the offset. In pyBAR the intercept from linear regression is taken as offset. The values of  $V_{Grad}$  and  $V_{offset}$  are written into the configuration. The calibration of the injection mechanism needs to be done before the tuning.

**Threshold Tuning** The tuning of the discriminator threshold is done in two steps. At first the global value is tuned with the GDAC Fast Tune, after this the in-pixel tuning is done by the TDAC Fast Tune.

The GDAC Fast Tune is adjusting the `Vthin_Alt_Coarse` and `Vthin_Alt_Fine` values until the desired threshold is reached. There is charge is injected, that is equal to the desired threshold. The hit occupancy is monitored and if the percentage of the injections that cause an injection is more than 50% the value is increased and if it is less than 50% it is decreased. The tune has a fixed number of iterations with decreasing step width. The resulting values of `Vthin_Alt_Coarse` and `Vthin_Alt_Fine` are written to the global register of the FE.

The TDAC Tune is working with the same mechanism as the GDAC Tune, but for each pixel on its own. This in-pixel adjustment is necessary to decrease the dispersion from around  $600e$  to an acceptable dispersion of less than  $50e$ .

**ToT Tuning** In addition to the threshold the ToT needs to be tuned as well for a complete tuning. There are two more tunings necessary, the IF Tune, that is adjusting the global ToT and the FDAC Tune, which is doing the in-pixel adjustment. Before the tuning, the values of `PrmpVbpf`, for the global feedback, and `FdacVbn`, the FDAC step size, are pre-set as described in [17].

The IF Tune in STcontrol scans the whole range of `PrmpVbpf`. The injected charge is there fixed and the ToT response is changing. After the tuning, the best fitting value is written to FE configuration. The mechanism is the same for the FDAC Tune for in-pixel adjustment.

In pyBAR the global feedback and the FDAC are tuned via fast tunings. This means, that the step size is decreasing while getting closer to the desired value. With this method it is possible to be faster, because less values need to be scanned.



## 5. Results

The main thesis focus is the comparison between STcontrol and pyBAR. To compare a scan, executed by STcontrol and pyBAR, it is important that both programs run the scan with identical settings. The equal global register and the calibration is easy to set manually. The pixel registers including the TDAC and FDAC registers, are converted from the STcontrol configuration to a configuration that is readable by pyBAR using a specially developed tool. Because a FE-I4A is used, which has special test columns with different electronics implemented, the double columns 1, 39 and 40 are turned off.

For the active sensor, an Analog Scan is executed by STcontrol and pyCPIX using the same configuration. In addition, the output signals of the active sensor are measured with an oscilloscope.

### 5.1. STcontrol Tuning

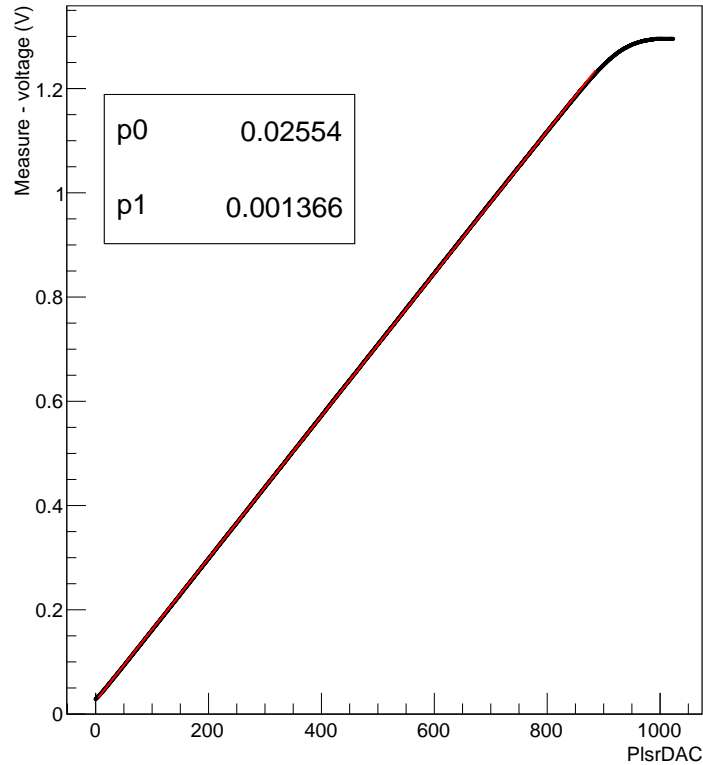
Due to the fact that the conversion of the configuration is only possible in one direction, the tunings for the comparison of single scans are done in STcontrol.

Before the tuning, the injection mechanism is calibrated. This is done via the DAC Scan and the Injection Calibration Scan as described in Chapter 4.5. The linear regression of the DAC Scan results is done manually after the scan and has to be entered by hand to the configuration. The linear increasing range is used as fitting range of the measured values. After this the Injection Calibration Scan (see Chapter 4.5) is executed, to get the offset of the injected voltage. At the end of the scan, the offset is stored in the configuration automatically.

The result of a DAC Scan with `Colpr_Mode=0` and `Colpr_Addr=2` is shown in Figure 5.1, the fitting range is from 0 to 890 DAC. The resulting calibration constants, which are used for all scans, are

$$V_{\text{Grad}} = 1.366 \text{ mV/DAC and}$$
$$V_{\text{Offset}} = 8.7658 \text{ mV .}$$

## 5. Results



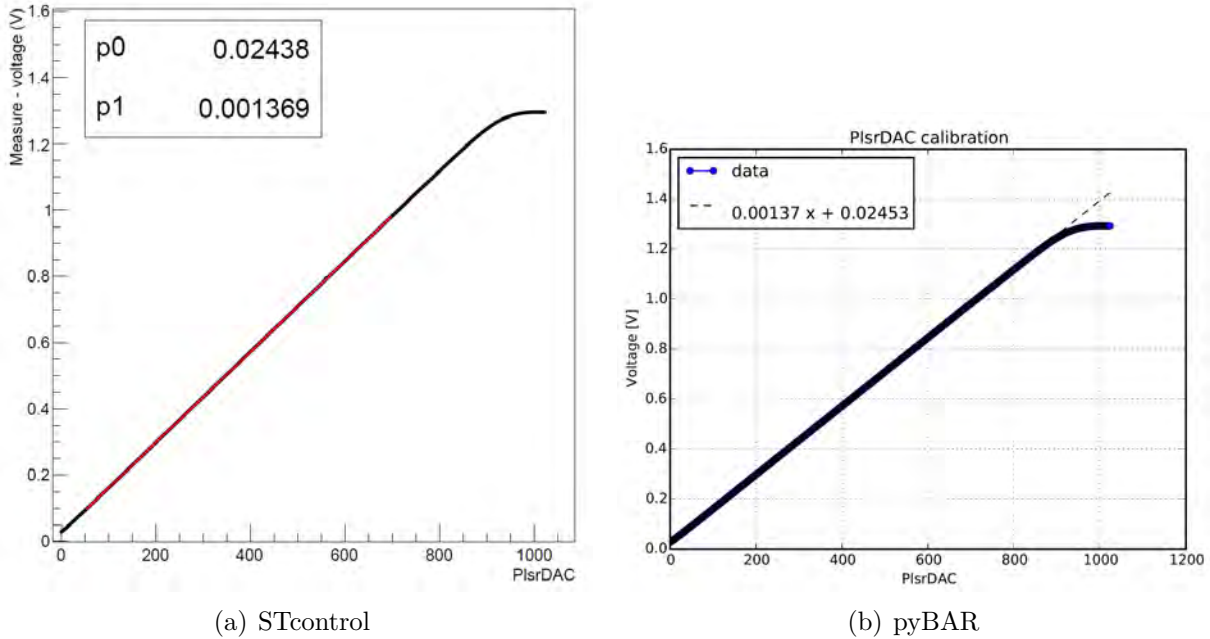
**Figure 5.1.:** Plot of a DAC Scan with Colpr\_Mode=0 and Colpr\_Addr=2.

The tuning of the discriminator thresholds and feedback currents is done by running through a sequence of scans: GDAC Fast Tune, TDAC Fast Tune, IF Tune, FDAC Tune, GDAC Fast Tune and TDAC Fast Tune. The target threshold and ToT values are specified in the tuning settings beforehand. After the tuning, the results are checked with a Threshold Scan and a ToT Verification Scan.

At first the FE is tuned to a threshold of 3000  $e$  and a ToT of 6 for an injected charge of 20000  $e$ . To tune the FE to a lower threshold, the configuration of the tuning for 3000  $e$  is used as the starting point and the TDAC Fast Tune scan is executed with the present values as start values. For this thesis, tunings were prepared for a threshold of 3000  $e$  with a ToT of 6 for an injected charge of 20000  $e$  and for a threshold of 1600  $e$  with a ToT of 10 for an injected charge of 15000  $e$ .

### 5.2. pyBAR Tuning

To tune the FE-I4 with pyBAR the same scans as in STcontrol are applied. The differences are, that the IF Tune and the FDAC Tune are fast tunings (see Chapter 4.5), the ToT



**Figure 5.2.:** Comparison of the DAC Scan in STcontrol and pyBAR with Colpr\_Mode=0 and Colpr\_Addr=2.

Verification is included in the Analog Scan and the injection calibration takes the axis intercept from the linear regression of the DAC Scan result as offset of the voltage.

## 5.3. Comparison of STcontrol and pyBAR

At first, single scans with the same configuration and tuning are compared. In the end, a comparison of both programs of a complete tuning is done. A complete tuning in this case means the tuning of the discriminator threshold and the feedback current.

### 5.3.1. Injection Calibration

The results of the DAC Scan are nearly the same in STcontrol and pyBAR, as can be seen in Figure 5.2. Both programs use for the linear regression the  $\chi^2$  method in the range of 50 to 700 PlsrDAC. It is worth noticing, that the measured voltage is assumed to be exact, although in reality the uncertainties are on the order of 30 mV, which was measured with an oscilloscope (see Figure A.1). This error is small for the measured voltage range from 0 to about 1.4 V, which means that the effect on the linear regression would be very small. In the end it is acceptable to neglect the uncertainties of the axis intercept and the slope.

## 5. Results

STcontrol			pyBAR		
Threshold [e]	Dispersion [e]	Noise [e]	Threshold [e]	Dispersion [e]	Noise [e]
2969	32	146	2977	36	145
1625	24	128	1614	29	126
4874	594	152	4876	597	151

**Table 5.1.:** Results of the Threshold Scans in STcontrol and pyBAR for tuned threshold of  $3000e$ ,  $1600e$  and the untuned Threshold Scan.

### 5.3.2. Threshold Scan

To compare the results of the Threshold Scan of STcontrol and pyBAR, configurations from STcontrol are used, where the thresholds were tuned to  $3000e$  and  $1600e$ , respectively. An comparison of the results for an untuned Threshold Scan was made as well. The results are shown in the Figures 5.3, A.2 and A.3 and are summarized in Table 5.1. A difference of a few electrons can be seen between both scans.

Figures 5.4 and A.4 show the correlations of the threshold values between measurements in STcontrol and pyBAR for thresholds of  $3000e$  and  $1600e$  respectively. Unphysical values of the measured thresholds due to masked and non-working pixels are excluded from the plots. In both plots the correlation between the threshold values of STcontrol and pyBAR is over 90 %.

In addition there is a correlation plot shown in Figure 5.5, with the results of a Threshold Scan from an untuned FE of STcontrol and pyBAR. This is done to figure out, if there really is a correlation between the threshold values, because for a tuned chip the fluctuations of the values are very small, which makes it difficult, to find a correlation. The correlation of the values for the untuned Threshold Scan is close to 100 %.

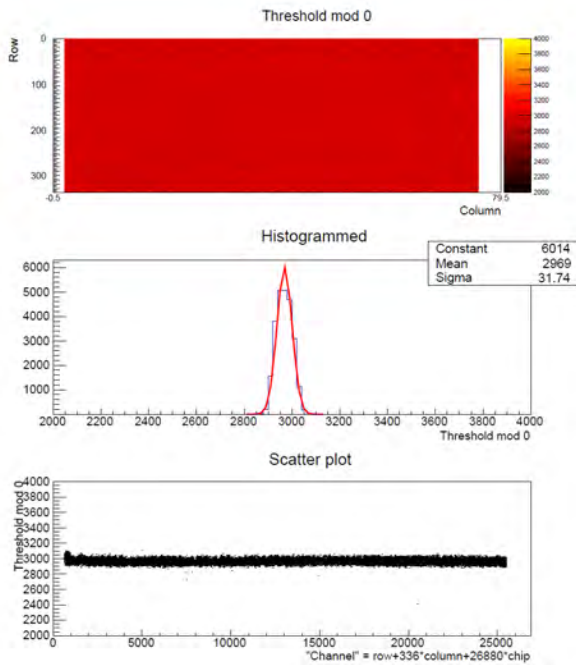
To check if there is a systematic difference in the programs the difference of the measured thresholds  $\Delta thr$  for each pixel are plotted against the channel, to see if the scattering of the values is symmetric. The difference of the thresholds was calculated for each pixel by

$$\Delta thr = thr(\text{STcontrol}) - thr(\text{pyBAR})$$

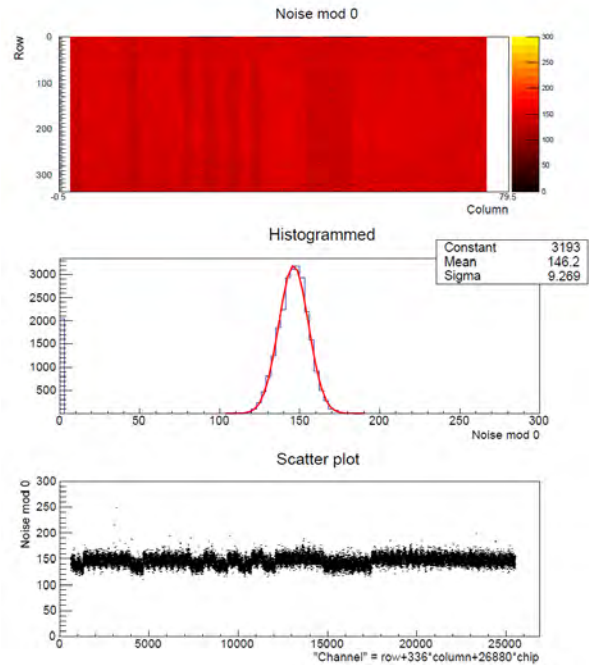
where  $thr(\text{STcontrol})$  is the measured threshold value of STcontrol and  $thr(\text{pyBAR})$  the value of pyBAR. This was done for the tuned Threshold Scans of  $3000e$  and  $1600e$  and the untuned Threshold Scan respectively (see Figures 5.6, A.5 and A.6). The Figures also show a histogram to see if the values are Gaussian distributed around 0 as expected. All unphysical values of the threshold differences due to masked and non-working pixels are excluded from the plots. To calculate the uncertainty of  $\Delta thr$  the propagation of uncertainties was used. The uncertainties are not shown in the plot, because otherwise



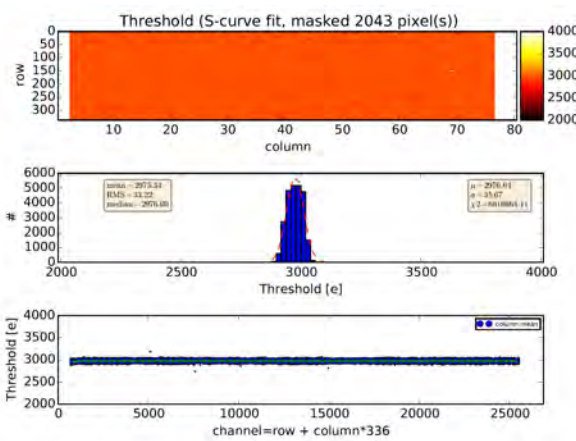
### 5.3. Comparison of STcontrol and pyBAR



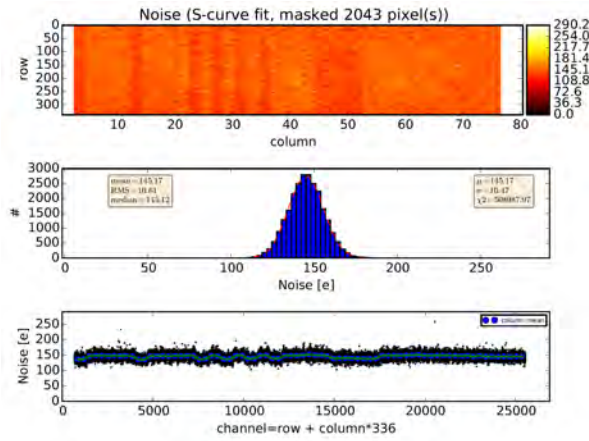
(a) STcontrol threshold



(b) STcontrol noise



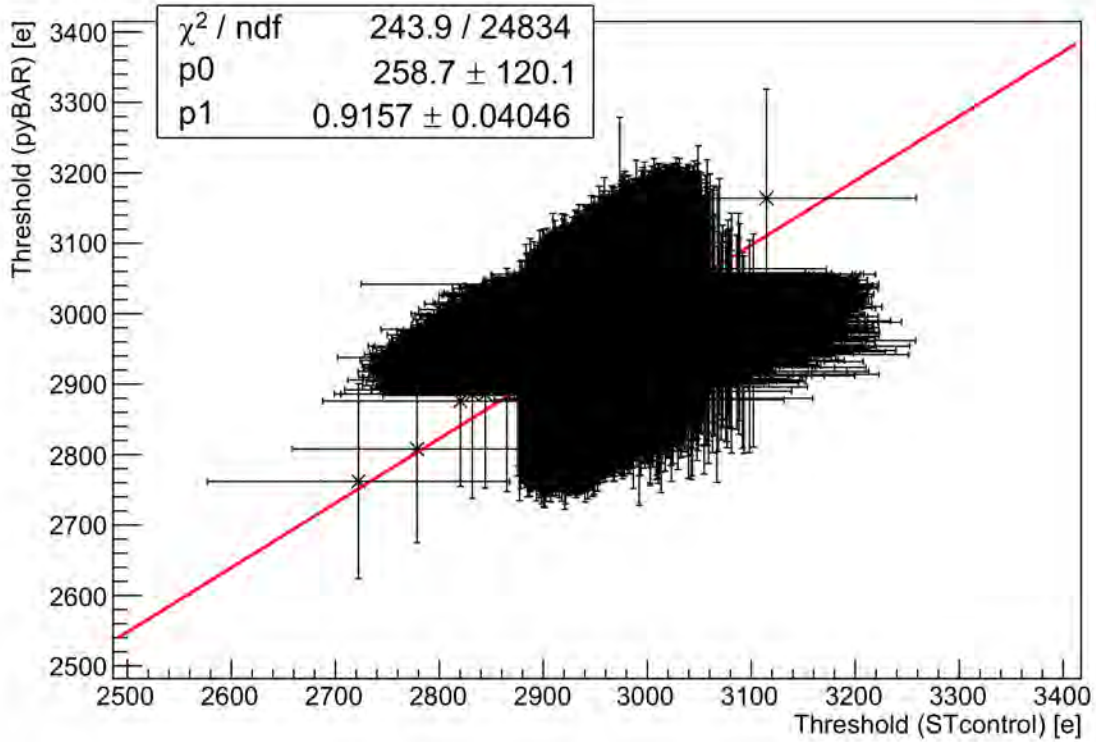
(c) pyBAR threshold



(d) pyBAR noise

**Figure 5.3.:** Comparison of the Threshold Scan at a threshold tuned to 3000 e. The scan results are summarized in Table 5.1.

## 5. Results



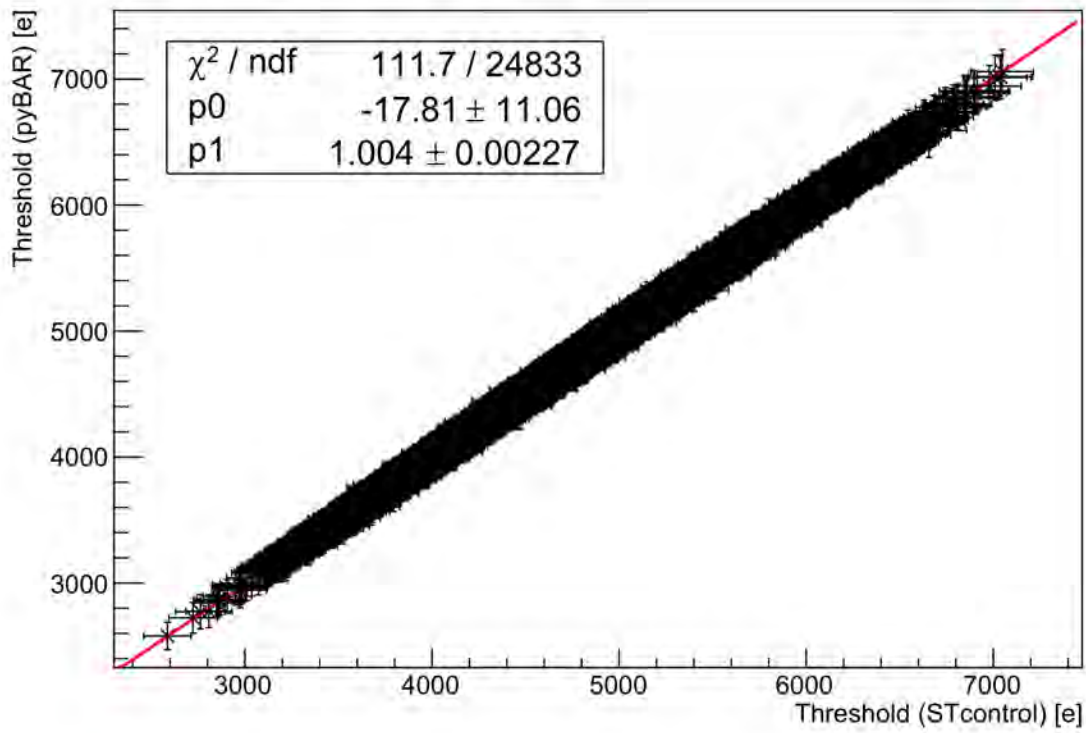
**Figure 5.4.:** Correlation of the Threshold Scan from STcontrol and pyBAR at a Threshold of 3000  $e$ . The shown errorbars are the noise of each pixel.

the shape of the values would not be visible. In Table 5.2 the results of the histogram are summarized. For all three Threshold Scans  $\Delta thr$  is close to 0.

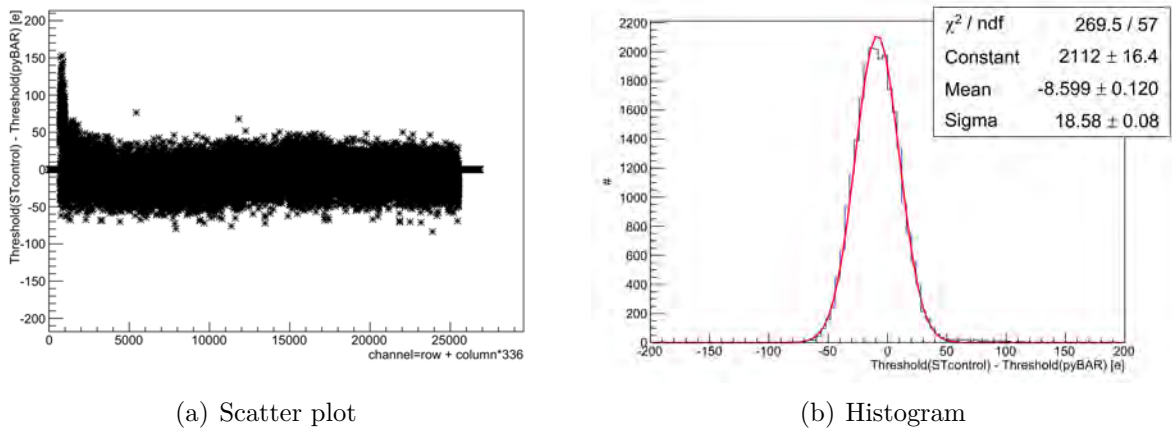
Uncertainties that are not reflected in the scan results displayed in the Figures above arise from the uncertainty of the injected charge itself. This stems from the production fluctuation of the capacitance values, which have a spread of about 10%. For repeated measurements of the threshold of one single pixel this yields a systematic uncertainty, while for the distribution of thresholds across all pixels this is a statistical uncertainty. Another, purely statistical, contribution to the uncertainty of the injected charge is the

Tuned Threshold [e]	Mean $\Delta thr$ [e]	Dispersion [e]
1600	8.5	10.7
3000	-8.6	18.6
untuned	0.1	14.1

**Table 5.2.:** Results of the histograms of the differences in the threshold of STcontrol and pyBAR for the Threshold Scans of a threshold tuned to 3000  $e$  and 1600  $e$  and the untuned Threshold Scan.



**Figure 5.5.:** Correlation of the Threshold Scans from STcontrol and pyBAR for an untuned configuration. The shown errorbars are the noise of each pixel.



(a) Scatter plot

(b) Histogram

**Figure 5.6.:** Difference of threshold measured by STcontrol and pyBAR of a Threshold Scan tuned to 3000 e. Illustrated in a scatter plot (a) and histogram (b). Error bars are not shown in the scatter plot.

## 5. Results

fluctuation of the voltage applied to the injection capacitance. It was measured with an oscilloscope, the result can be seen in Figure A.1. From this measurement an uncertainty of  $\pm 30$  mV was obtained.

With the propagation of uncertainties the uncertainty of the charge  $\sigma_{Q_{\text{inj}}}$  is calculated to

$$\sigma_{Q_{\text{inj}}} = \sqrt{(\sigma_{C_{\text{inj}}} \cdot (V_{\text{inj}} + V_{\text{offset}}))^2 + (\sigma_{V_{\text{inj}}} \cdot C_{\text{inj}})^2}$$

This leads to a large uncertainty of a single charge injection in one pixel, but while a Threshold Scan each pixel is injected 100 times ( $N_{\text{inj}}$ ) for every DAC value, which reduces the uncertainty by the factor of  $1/\sqrt{N_{\text{inj}}}$ . A perfect discriminator characteristic as a function of injected charge would be a sharp step function, but in reality this is folded with a Gaussian distribution, which arises from random noise. Relating to this, the uncertainty of the threshold for a single pixel is the width of the Gaussian distribution and this are the values, which are shown in the noise plot.

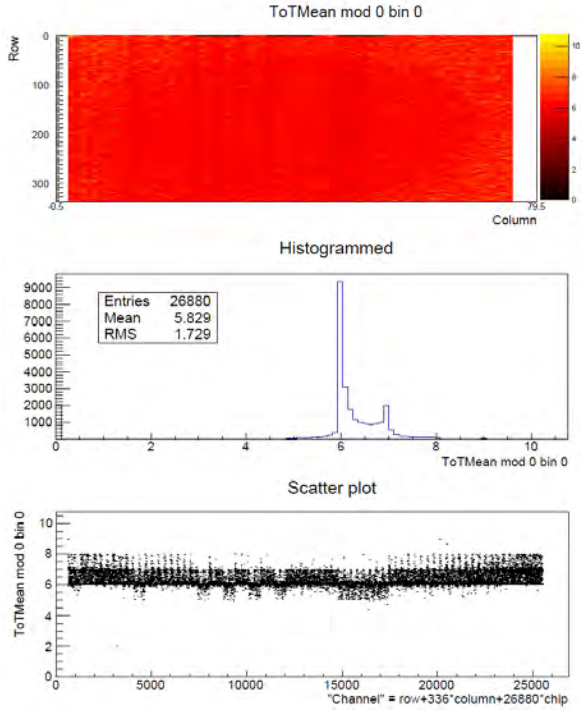
### 5.3.3. ToT Verification

The ToT Verification Scans are compared for a tuned threshold of  $3000 e$  and a ToT of 6 for an injected charge of  $20000 e$  and a tuned threshold of  $1600 e$  and a ToT of 10 for an injected charge of  $15000 e$ . In STcontrol the ToT Verification scan is used for this, in pyBAR this is implemented in the Analog Scan. The results are compared in Figure 5.7. Due to the fact that pyBAR fails to calculate the mean value of the histogram only the shape can be compared, which looks similar in both plots.

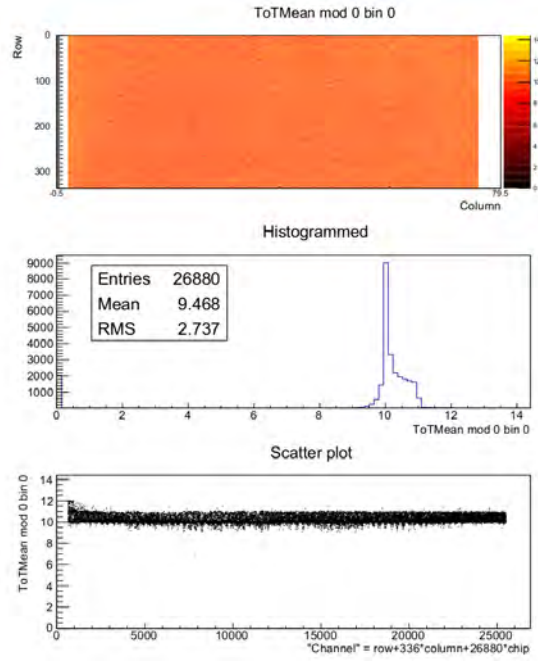
Also for the ToT verification a correlation plot is helpful to find out if the programs do the same. For a ToT of 6 with a charge of  $20000 e$  the correlation plot is shown in Figure 5.8. The values of the measured ToT that are unphysical due to masked and non-working pixels are excluded from the plot. A correlation of almost 100% results from this plot. For a ToT of 10 with a charge of  $15000 e$  a correlation plot would make no sense, because ToT values that are larger than 12 are also measured as a ToT of 12, results in a nearly circular dispersion.

The uncertainties for the ToT values in STcontrol are written to the sigma mean plot. For this plot the uncertainties of each pixel are calculated via the standard deviation of the 200 injections. The uncertainties for the ToT values in pyBAR need to be calculated manually. This is done via the standard deviation for each pixel. Both uncertainties are considered in the correlation plot, they are scatter around 0.03 ToT.

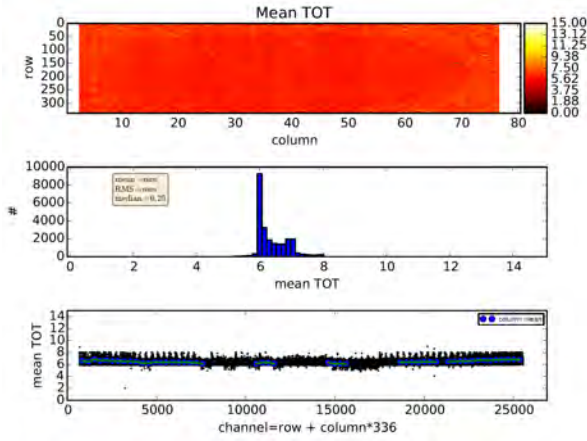
### 5.3. Comparison of STcontrol and pyBAR



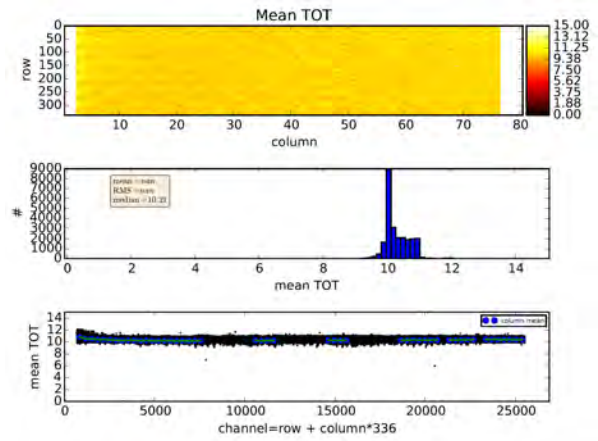
(a) STcontrol, ToT=6



(b) STcontrol, ToT=10



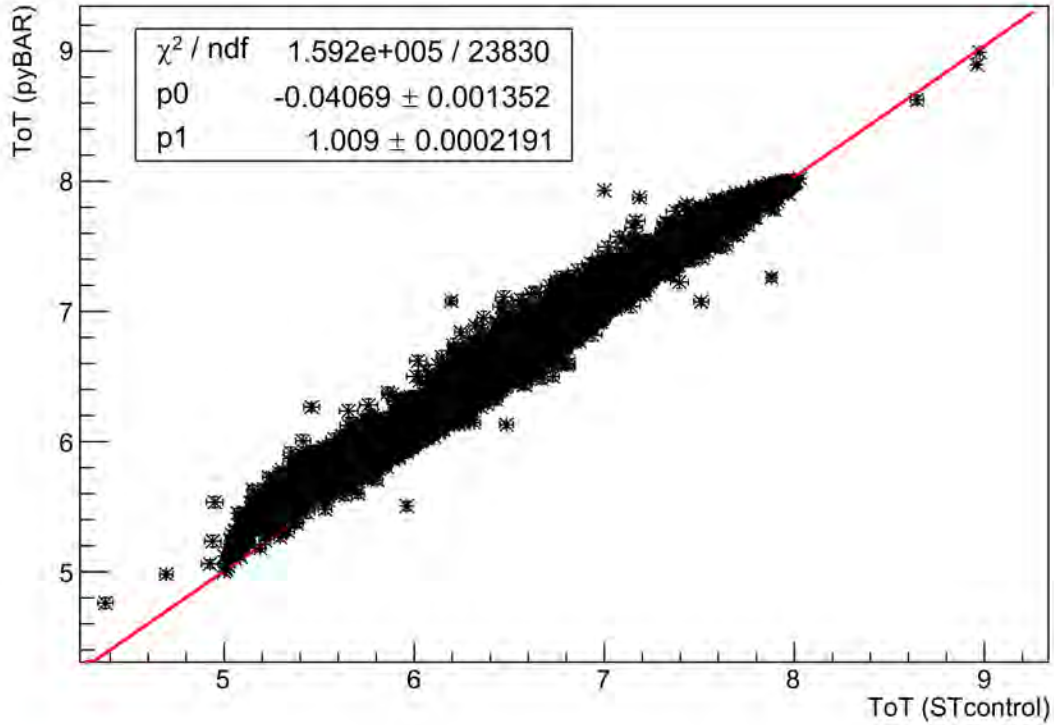
(c) pyBAR, ToT=6



(d) pyBAR, ToT=10

**Figure 5.7.:** Comparison of the ToT Verification Scan at a ToT tuned to 6 for an injected charge of 20000  $e$  and a ToT tuned to 10 for an injected charge of 15000  $e$ .

## 5. Results



*Figure 5.8.:* Correlation of the ToT verification from STcontrol and pyBAR at a ToT of 6 for an injection charge of 20000  $e$ .

### 5.3.4. Complete Threshold Tuning

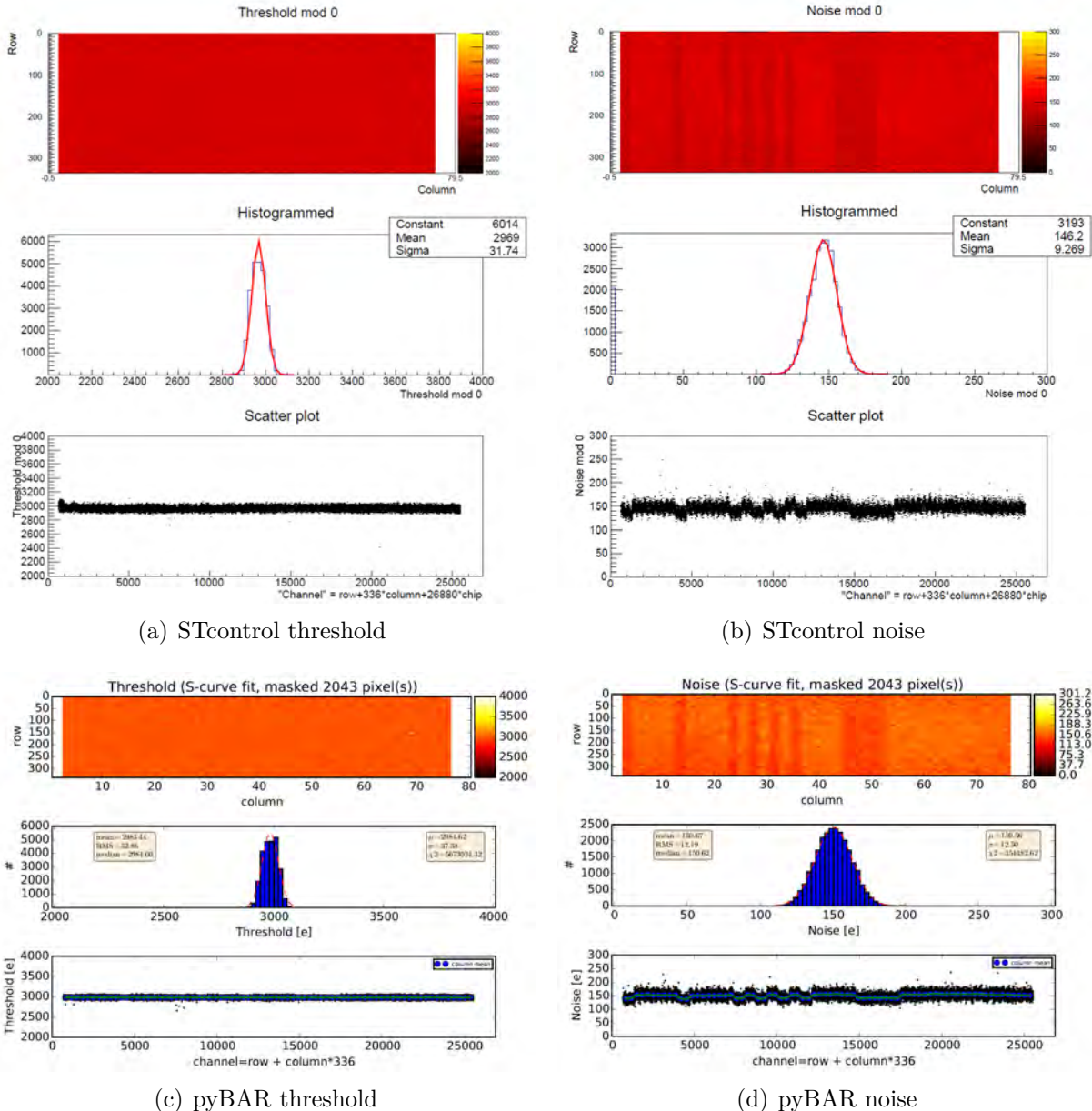
On top of the single scans, two complete tunings of the discriminator threshold and the feedback current are compared, in order to see if the tuning algorithms yield the same results. For STcontrol the already existing tunings to a threshold of 3000  $e$  and a ToT of 6 for an injected charge of 20000  $e$  and a threshold of 1600  $e$  and a ToT of 10 for an injected charge of 15000  $e$  are taken, for pyBAR the tunings are executed with the same injection calibration and the same settings. The results are shown in Figures 5.9, 5.10, A.8 and A.7 and are summarized in Table 5.3. For the mean threshold and the dispersion of the threshold the results differ by a few electrons. The shape of the ToT values in the histograms look different, but the highest bin is at the same ToT value in both plots.

### 5.4. Scan/Tuning Duration

The duration of the scans and tunings are interesting, because a faster tuning would be helpful during production tests. Therefore, the scan duration of STcontrol and pyBAR are

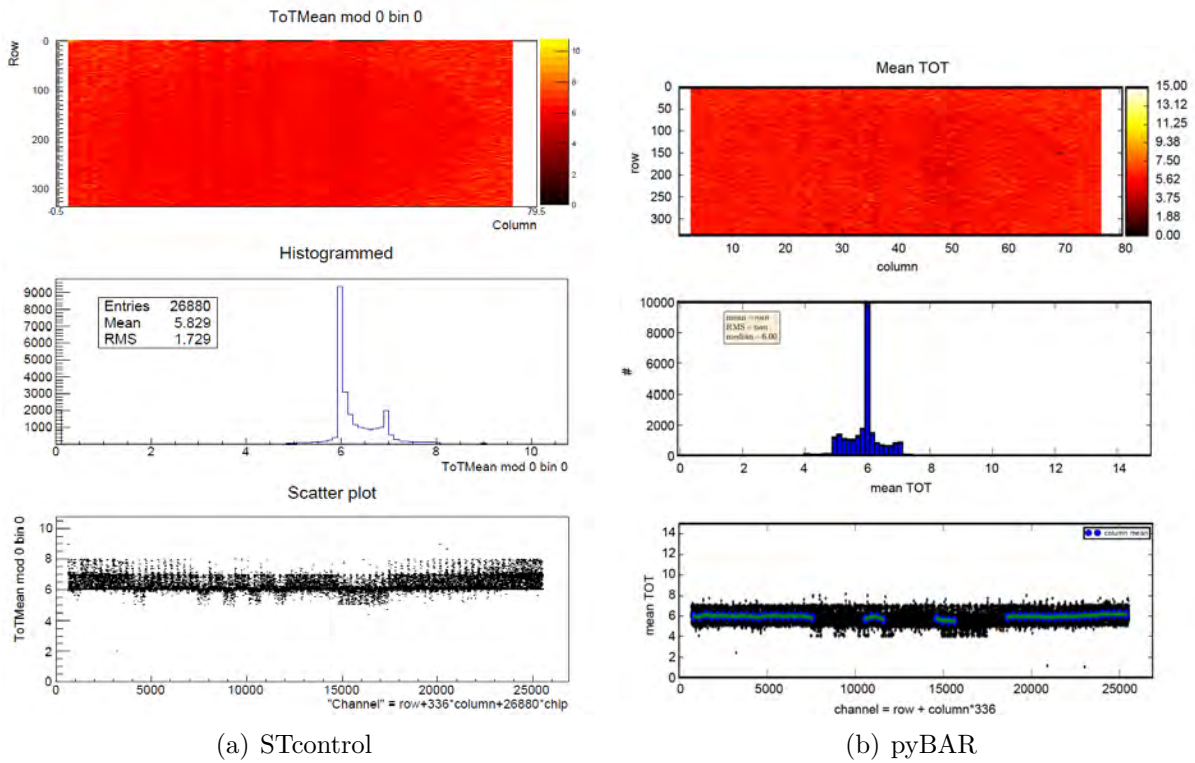
STcontrol			pyBAR		
Threshold [e]	Dispersion [e]	Noise [e]	Threshold [e]	Dispersion [e]	Noise [e]
2969	32	146	2985	38	151
1625	24	128	1626	31	130

**Table 5.3.:** Results of the Threshold Scans for separately tuned thresholds of 3000 e and 1600 e by STcontrol and pyBAR.



**Figure 5.9.:** Comparison of the Threshold Scan at a threshold tuned separately to 3000 e by both programs. The scan results are summarized in Table 5.3.

## 5. Results



**Figure 5.10.:** Comparison of the ToT Verification Scan for a separately tuning of STcontrol and pyBAR to a ToT of 6 for an injected charge of 20000  $e$ .

compared. The duration of each scan or tuning is written to the log file in both programs. This time is used to compare the programs. The result of the comparison is shown in Table 5.4. For a complete tuning of the discriminator threshold and the feedback current pyBAR is nearly three times faster than STcontrol.

Even with the same components and same settings there are fluctuations in the duration of the scans. The fluctuations are in the range of few seconds for scans taking longer than 2 min, for shorter scans the fluctuations are around 1 s. Therefore, the results in Table 5.4 are rounded in half minutes steps, a more precise specification would not make sense.

The time difference at the DAC Scan is not comparable, because the multimeter was connected differently during the measurements for STcontrol and pyBAR. For STcontrol the multimeter was connected via GPIB and for pyBAR it was connected via USB. The other scans are not affected by this problem, because only for the DAC Scan the multimeter is used.

The IF and FDAC Scans are also not fully comparable, because there is no implementation of a fast tune, which is working bit wise and is not going step by step, for the IF and FDAC Tune in STcontrol.



Scan/Tuning	STcontrol [min]	pyBAR [min]
DAC Scan*	24:00	1:30
INJ Calibration	20:00	-
GDAC Fast Tune	3:30	0:30
TDAC Fast Tune	2:00	1:00
IF Tune**	3:00	0:30
FDAC Tune**	2:30	0:30
Threshold Scan	13:00	7:00
ToT Verification	0:30	See Analog Scan
Analog Scan	0:30	0:30
Digital Scan	0:30	0:30
2x GDAC, 2x TDAC, IF, FDAC, Threshold, ToT verif	30:00	11:30

**Table 5.4.:** Comparison of some scans and tunings of STcontrol and pyBAR. The time is rounded to half minutes. \*A different Interface was used (GPIB for STcontrol and USB for pyBAR). \*\*These tunings are structured differently in STcontrol and pyBAR.

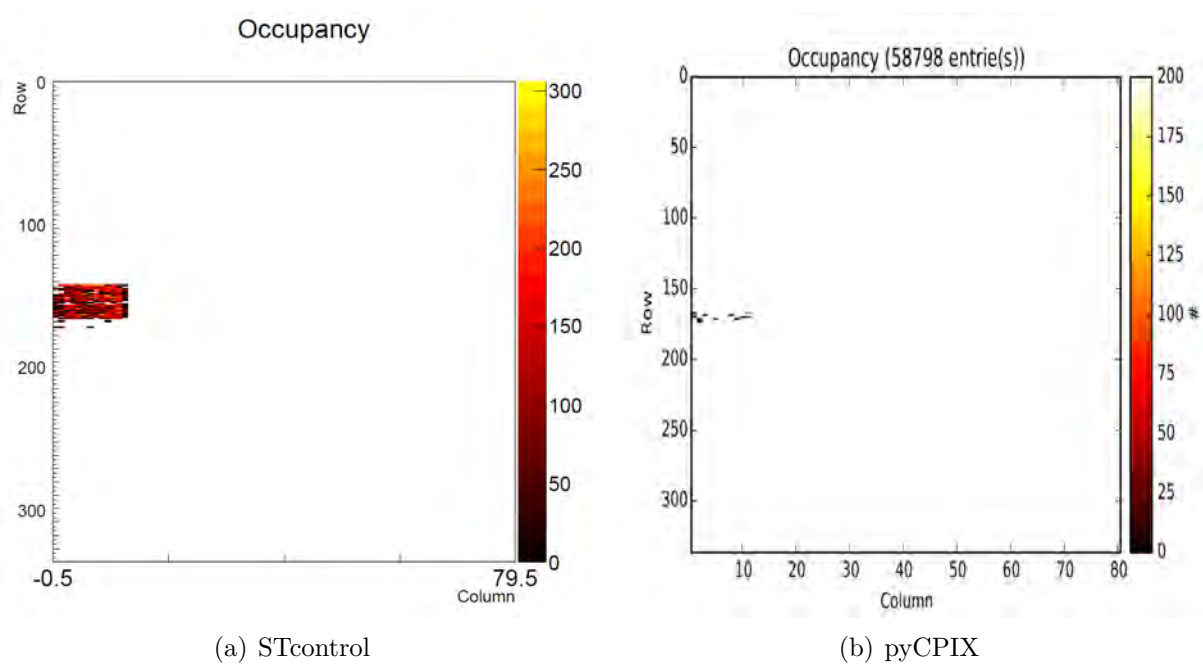
## 5.5. Active Sensor

For the active sensor only the Analog Scan is available, which is compared while using the same settings in STcontrol and pyCPIX. Because of a different design rows 51-60 of the active sensor are not used. The results for the Analog Scan are shown in Figure 5.11. It can be seen, that the occupancy of the Analog Scan does not look homogeneously as expected. The counted hits are not equal over the sensor, as they should be, but there are large fluctuations. Therefore the charge injection into the sensor is studied using an oscilloscope.

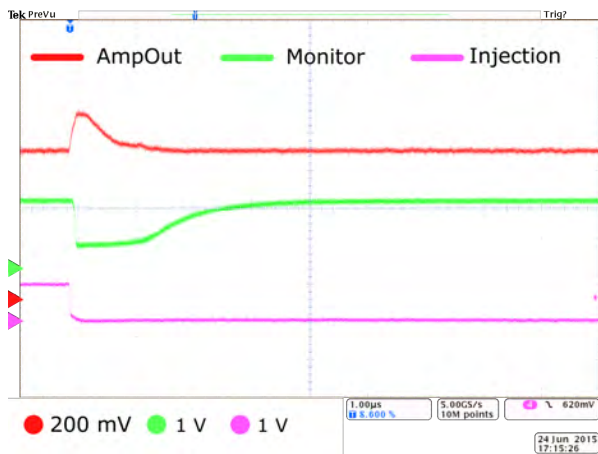
The AmpOut is measured to have a look at the signal after the amplification and the Monitor measures the signal after the discriminator (see Figure 3.2). This is helpful to see where differences of the programs might be.

To get a signal on the AmpOut pin and the Monitor pin, the configuration needs to be adjusted. When the Monitor is turned on, no signal can be read-out by the programs, because the signal is routed by a switch either to the Monitor pin or to the FE. The settings that are used in STcontrol and pyCPIX are the same. A comparison of the snapshots from the oscilloscope is shown Figure 5.12. Three different channels were measured with the oscilloscope: Injection, AmpOut and Monitor. The snapshots are selected in a way to have a look at one injection in detail and to see if there are fluctuations for different injections. The injections were chosen randomly, but all of the 200 injections look nearly the same.

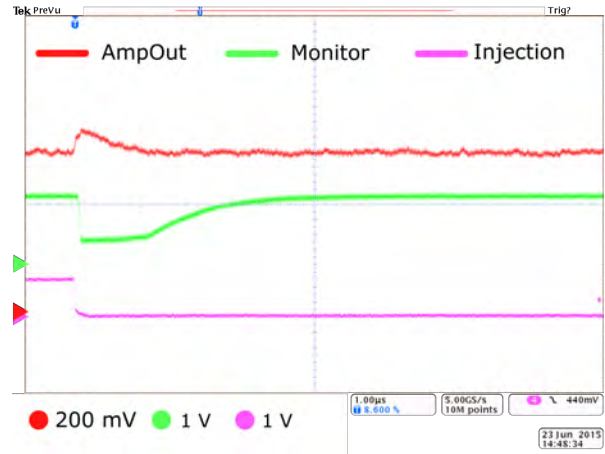
## 5. Results



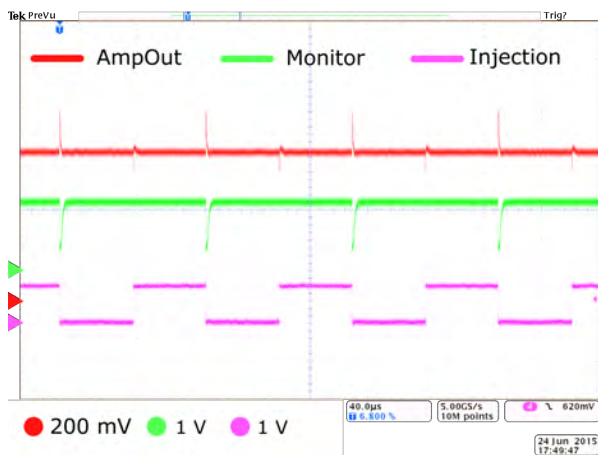
**Figure 5.11.:** Comparison of the Analog Scan for the active sensor.



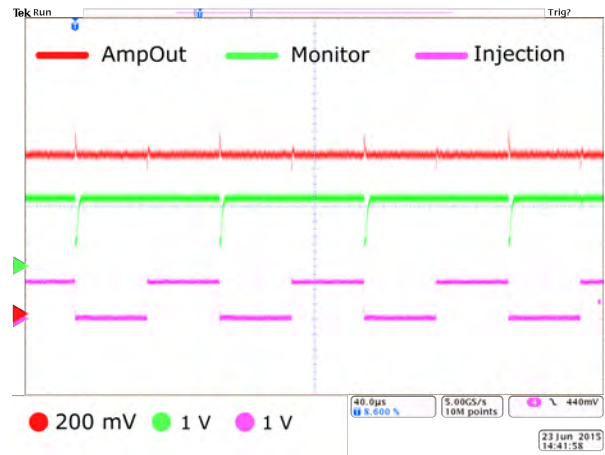
(a) pyCPIX 1  $\mu$ s scale



(b) STcontrol 1  $\mu$ s scale



(c) pyCPIX 40  $\mu$ s scale



(d) STcontrol 40  $\mu$ s scale

**Figure 5.12.:** Comparison of the oscilloscope measurements with the active sensor.



## 6. Discussion

### 6.1. Comparison of STcontrol and pyBAR

It was possible to execute all desired scans with STcontrol and pyBAR and to compare them. The comparison of each scan in detail is discussed in the following.

#### 6.1.1. Injection Calibration

The results of the calibration by the DAC Scan show, that both programs do the same. There are small differences in the resulting values, but the differences are in the same magnitude as the differences of two DAC Scans with the same program. It is expected, that the temperature of the chip has a bigger influence on the results, than the chosen program.

A disadvantage of pyBAR is that the fitting parameters need to be chosen before executing the scan. It would be easier to set the fitting parameters after the scan.

One of the biggest differences of both programs is the measurement of the offset. Taking the axis intercept instead of running the Injection Calibration leads to a higher offset for all measurements that are done. The difference for the used chip is about 15 mV, which means approximately 500  $e$ . An increased offset by 500  $e$  is a lot in comparison to a threshold of 1600  $e$  or 3000  $e$ . Therefore it was decided, to use the calibration of STcontrol - a scan specially designed to find the offset seems more trustworthy.

#### 6.1.2. Threshold Scan

The results of the Threshold Scan are compared in different ways. Big differences would be visible in the pixel map, the histogram or the scatter plot, the plots of both programs look similar. Small differences can be found in the mean threshold and the dispersion of the Gaussian distribution of the histogram, but differences of a few electrons can be neglected, because they are within the accuracy of the Threshold Scan.

All three correlation plots of the Threshold Scans for 3000  $e$ , 1600  $e$  and especially the untuned show a strong correlation between pyBAR and STcontrol. The correlation of

## 6. Discussion

the Threshold Scans for 3000  $e$  and 1600  $e$  are over 90 % (see Figure 5.4 and A.4), for the untuned scan a strong correlation close to 100 % could be seen (see Figure 5.5).

That the Threshold Scans for 3000  $e$  and 1600  $e$  are less correlated than the scan of the untuned threshold makes sense, because the threshold values after the tuning are nearly all in a close range to the tuned threshold (about  $\pm 50 e$ ) and if the values were all too close to one point a correlation would not make sense.

A good proof that the results of both programs are nearly the same are the plots that show the difference of the threshold values of STcontrol and pyBAR (see Figures 5.6, A.5 and A.6). The difference of the threshold values is Gaussian distributed and scatters nearly around 0, which means that the differences between STcontrol and pyBAR are mostly statistical.

The reduced  $\chi^2$  for the linear fit in the correlation plots is much smaller than 1, which means in general that the uncertainties are overestimated. Choosing the noise as the uncertainty of a single pixel seems to be wrong or there is an error in the calculation of the noise. To figure out where the error occurred further investigation is needed.

In double column 2 there is the artefact found (see e.g. 5.6(a)), that the threshold values are higher than the values in the other columns. This effect can be seen only in STcontrol for all Threshold Scans. That this behaviour is not seen in pyBAR cannot be an effect of a mistaken conversion, because it occurs also at the untuned Threshold Scan. One reason might be that some kind of an edge effect occurs. To check this a closer look at the program code and more investigation would be necessary.

There is always one pixel that shows very different values for STcontrol and pyBAR. This might be a hint, that there is a broken pixel, causing this results.

### 6.1.3. ToT Verification

The ToT Verification scans have similar results in STcontrol and pyBAR. At the correlation plot (see Figure 5.8) for ToT tuned to 6 for an injection charge of 20000  $e$  a strong correlation close to 100% can be seen and the shape of the ToT histograms are similar. Even the shape of the histograms looks similar.

For the ToT tuned to 10 for an injection charge of 15000  $e$  the shape of the histograms is similar. To analyse the correlation is because of the limit to a ToT of 12 not possible.

pyBAR is for some reasons not able to calculate the mean and the RMS. This might be caused by an error in the data analysis script of pyBAR.

### 6.1.4. Complete Threshold Tuning

For the complete threshold tuning there are differences in the results, but it was expected that the tunings would not be identical. The main influences are expected from the IF and FDAC Tune, because they are differently designed in STcontrol and pyBAR. That the shape of the resulting ToT histograms looks different points in this direction.

The difference might be caused by the use of different units in the setting. The values in STcontrol are entered in  $e$ , in pyBAR DAC is used as the unit for charge. One DAC corresponds to approximately  $55 e$ , the exact value depends on the injection calibration. This may cause differences during the internal calculations of the programs.

## 6.2. Scan/Tuning Duration

An important advantage of pyBAR is the reduced time which is needed for the scans and tunings in comparison to STcontrol. A complete tuning of the discriminator threshold and the feedback current is nearly three times faster in pyBAR than in STcontrol (see Table 5.4).

For the DAC Scan the result is not comparable, because the multimeter was used with the GPIB port in STcontrol and with USB in pyBAR, which was caused by software problems in pyBAR. GPIB is known to be the slower connection, which explains the big difference in the scan duration. As part of the thesis, it was not possible to solve the software problems and to use USB with STcontrol or GPIB with pyBAR. Due to this fact it is not possible to say for the DAC Scan if and how much pyBAR is faster than STcontrol. Other scans are not affected by this problem.

The duration for the IF and FDAC Tune are not comparable, because of the different design of the scans. In pyBAR these are fast tunes, which means that there are less steps which need to be scanned, causing a faster scan. On the other hand there are no other scans than the fast tunes in pyBAR. This means that the tunes are not comparable individually, but for a scan duration comparison of a complete discriminator threshold and feedback current tuning of STcontrol and pyBAR these are the existing tunes that are used.

Another important impact on the duration are the settings for each scan or tuning. To inject 200 times takes longer than 100 times and more steps for example at the Threshold Scan would need also more time. Therefore it is helpful to figure out how many injections and which range for scanning make sense, to find the best combination of time efficiency and useful results.

## 6. Discussion

The uncertainties increase with increasing duration of the scan, which could be an effect of the FPGA firmware or the specific C++/python code.

The results of the duration comparison are not independent from the whole setup, a different computer with a slower or faster processor for example, could lead to faster or slower scans.

Most of the scans in pyBAR are faster even if they are doing in principle the same as in STcontrol. Why this happens is not clear at the moment. The guesses are that pyBAR is doing more calculations on the computer, instead of doing them on the FPGA of the MIO board and that the whole structure of pyBAR is less complex than STcontrol.

### 6.3. Active Sensor

The reason for the wrong histogramming in STcontrol and pyBAR still needs to be investigated. The raw data in pyBAR seem to be as expected, but the histogramming that is done after the scan is not working. There was a recent fix for pyBAR, that could solve this problem, but it could not be considered for this thesis. Possibly there is a solution to the problem of wrong hit counting in the near future for STcontrol as well.

The measurements with the oscilloscope show that STcontrol and pyCPIX do in principle the same, but there are differences in the amplitude of the AmpOut (see Figure 5.12). The AmpOut amplitude in pyCPIX is higher than in STcontrol. A reason could be that the programs consider different numbers of pixels, but changing the number of active pixels has a bigger influence on the shape than on the peak height especially for the Monitor. More likely the calibration of the GPAC which supplies the sensor voltage is different.

STcontrol has one specific GPAC calibration, whereas pyBAR reads the GPAC calibration provided by the GPAC and uses this.

The shape of the signals is the same and for 200 injections there are 200 responses at the Monitor and the AmpOut. Interesting for the Monitor is the length of the peak, because after the discriminator this is a digital signal and therefore only a logical 0 and a logical 1 are necessary. A lower input signal (AmpOut) into the discriminator should lead to a shorter Monitor signal, but there are only small differences measured between STcontrol and pyCPIX.



## 7. Conclusion and Outlook

The aim of this thesis was to compare the read-out systems STcontrol and pyBAR. Both programs are capable of executing scans, tests and tunings on a FE-I4 in the desired parameter range.

The main difference is the duration of scans and tunings. pyBAR executes a complete tuning of the discriminator threshold and the feedback current nearly three times faster than STcontrol. Answering the complex question why exactly STcontrol is slower than pyBAR in a couple of scans and tunings is beyond the scope of this thesis. For this investigation an implementation of a fast IF and FDAC tuning in STcontrol is advisable.

The results of the scans of both programs are in a good agreement. The differences are in the range of the uncertainties and thus both systems are suitable to test and tune the FE-I4. In a future comparison it would be interesting to include for example the source scan to see the behaviour of the programs with external hits.

It is important to remember that both systems, especially pyBAR, are still developing. Therefore, this thesis is a snapshot of the current status of the read-out systems.

The results of the active sensor show that the configuration of pyCPIX and STcontrol work in general, but the programs need to be further developed to fulfil all desired requirements of a read-out system. Counting and clustering of the hits for example need to be fixed, before a test setup with an active sensor like the CCPDv2 as a sensor is fully usable, but the investigated problems at the time of writing this theses are on an acceptable level to continue the development.

The advantages in cost efficiency and availability of CMOS sensors in general are obvious, but before they can be used in an experiment all problems need to be solved. Beside the software there are other problems, like the gluing of the sensor to the FE without any air bubbles, but in the end the advantages are worth to continue the investigations and solve those problems.



# A. Additional Plots

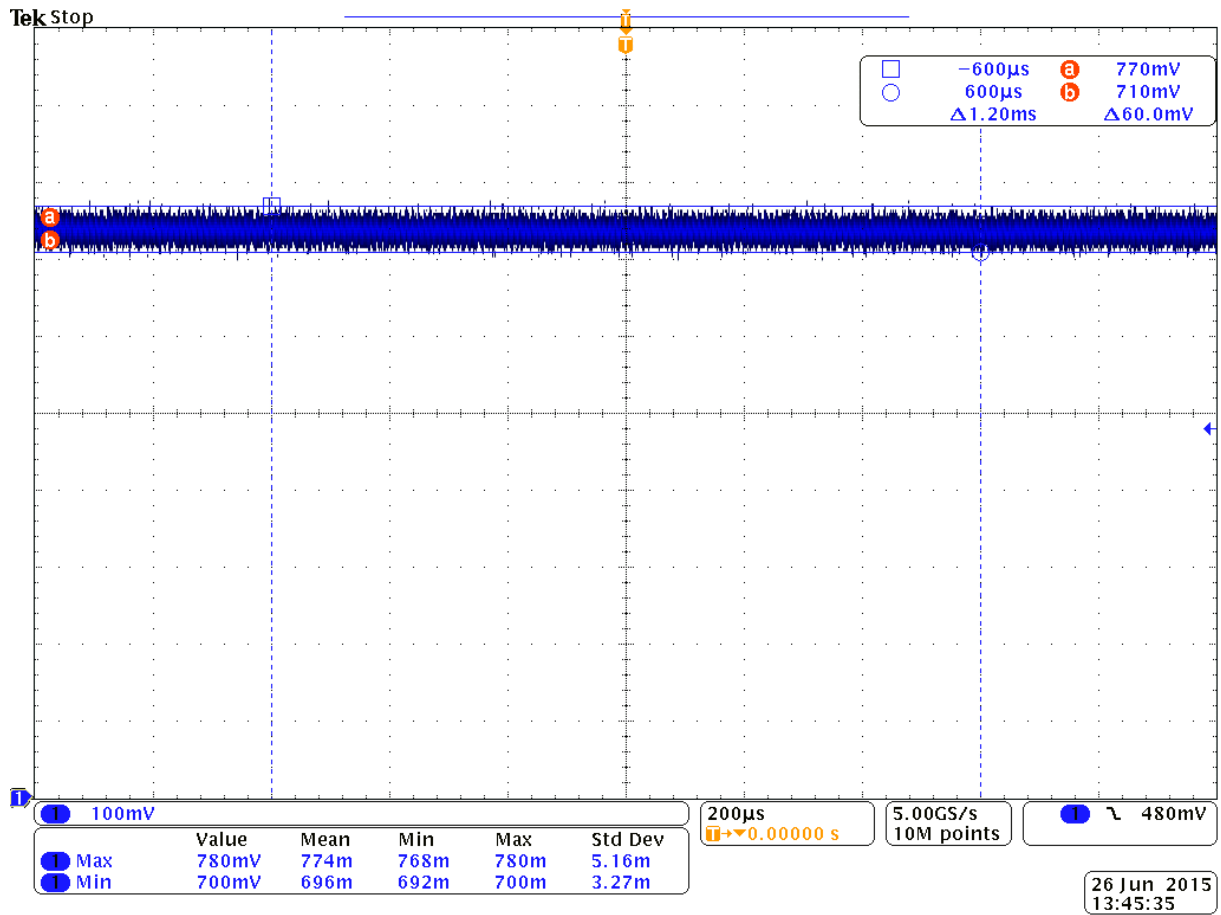
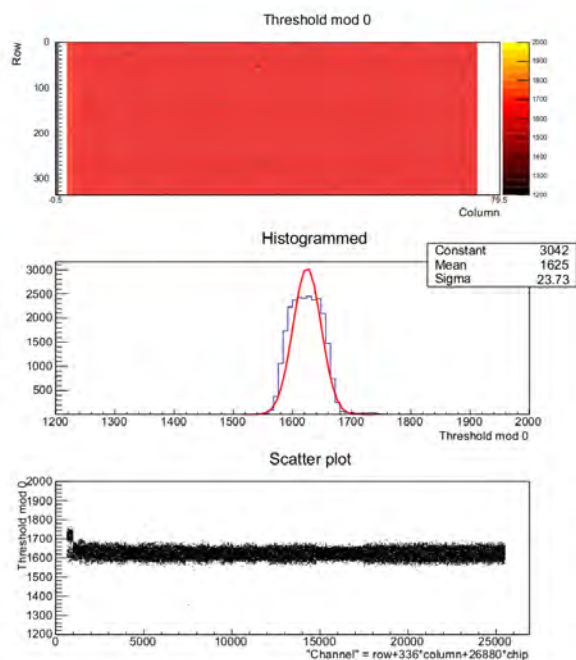
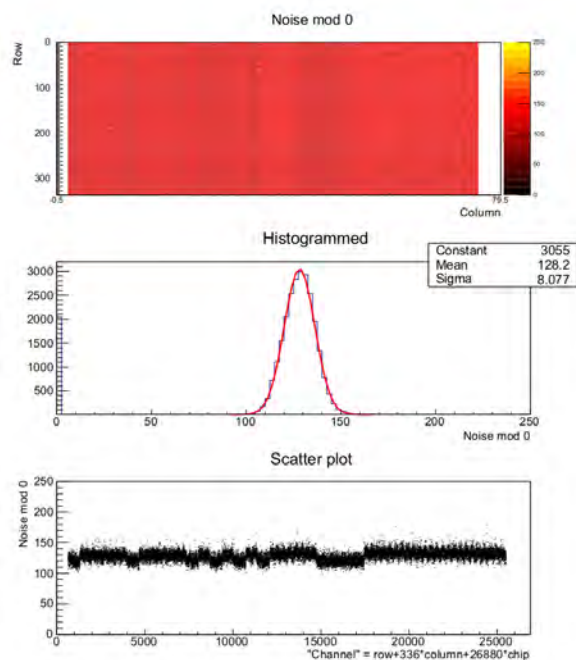


Figure A.1.: Oscilloscope measurement of the fluctuation of the voltage at the injection circuit.

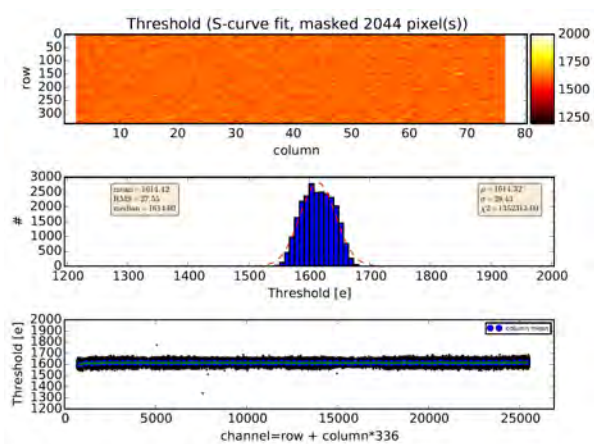
## A. Additional Plots



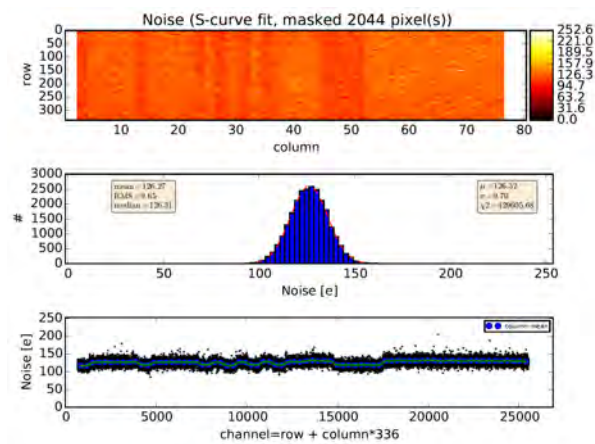
(a) STcontrol threshold



(b) STcontrol noise

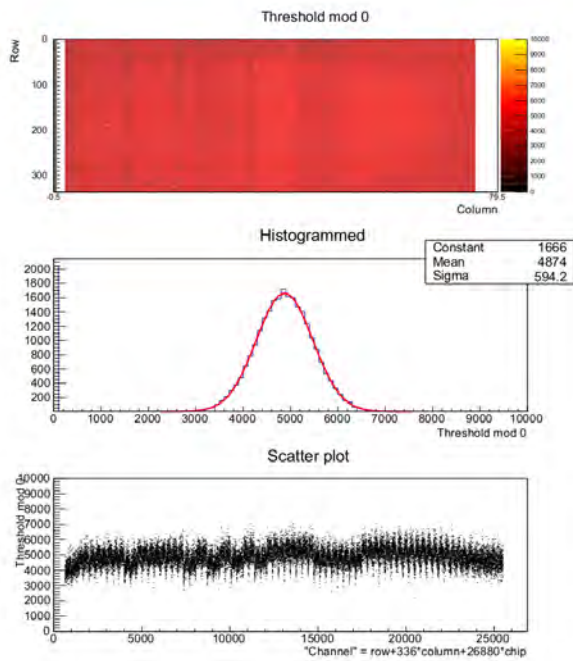


(c) pyBAR threshold

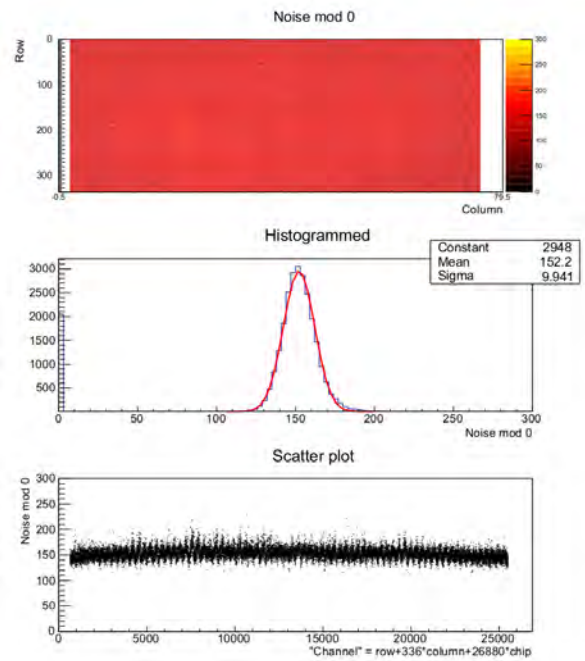


(d) pyBAR noise

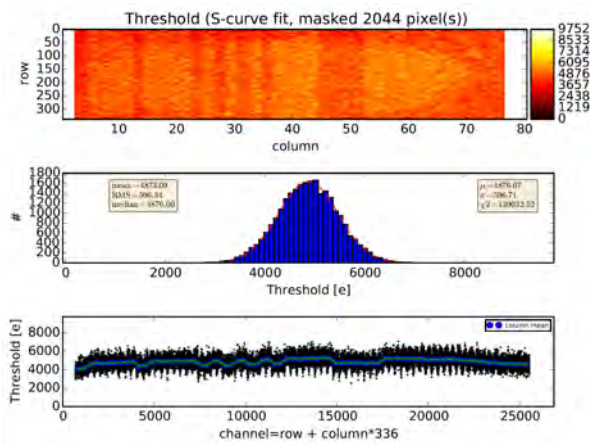
**Figure A.2.:** Comparison of the Threshold Scan at a threshold tuned to 1600 e. The scan results are summarized in Table 5.1.



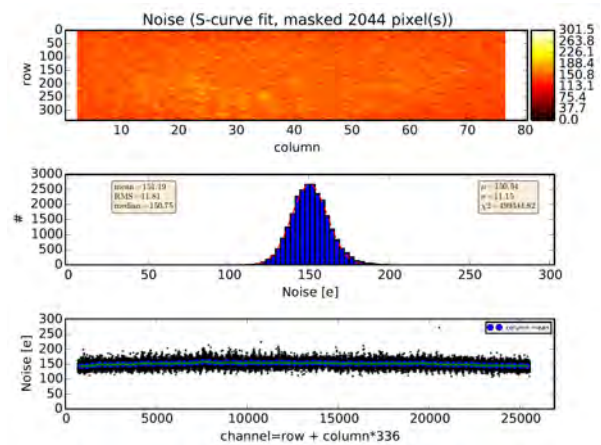
(a) STcontrol threshold



(b) STcontrol noise



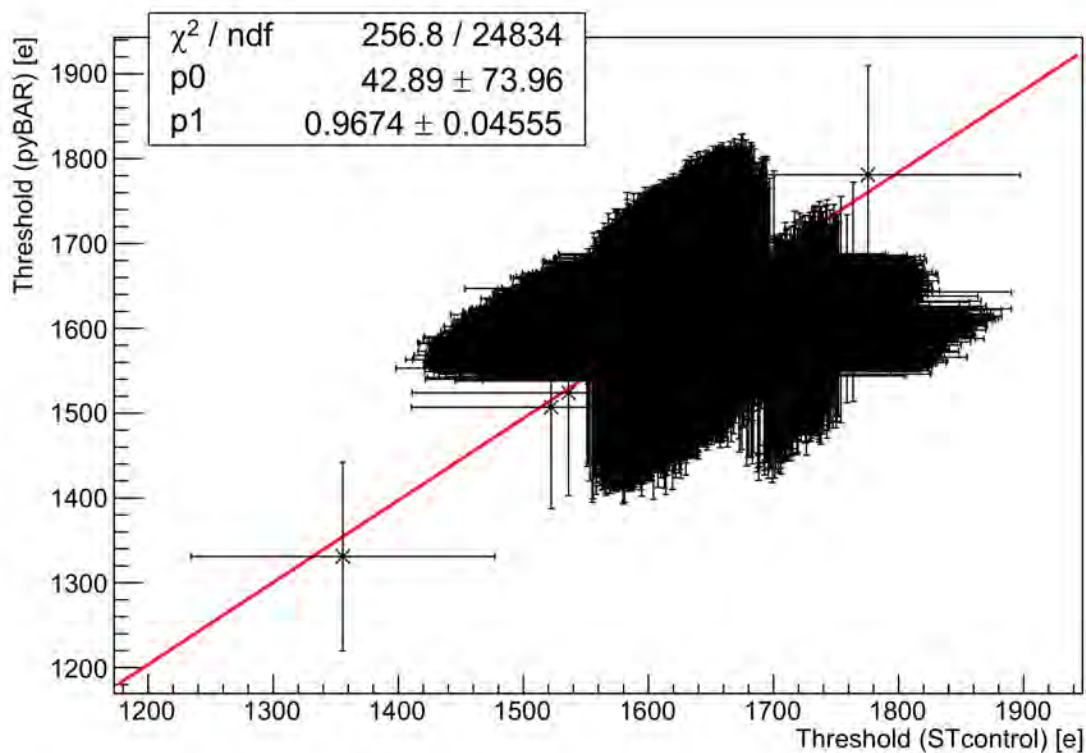
(c) pyBAR threshold



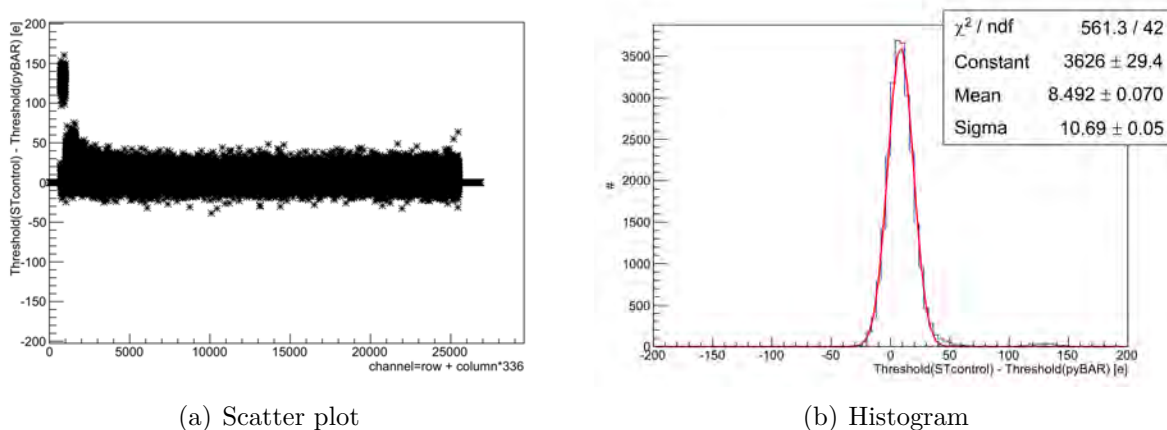
(d) pyBAR noise

**Figure A.3.:** Comparison of the Threshold Scan at an untuned threshold. The scan results are summarized in Table 5.1.

A. Additional Plots



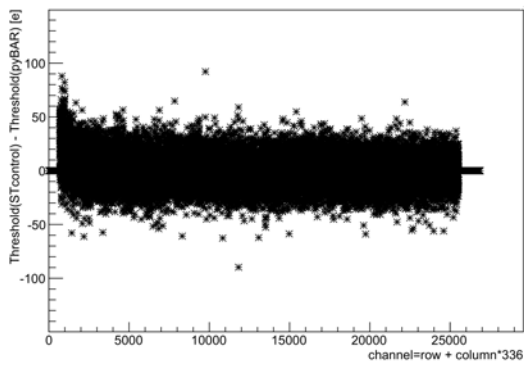
**Figure A.4.:** Correlation of the Threshold Scan from STcontrol and pyBAR at a Threshold of 1600 e.



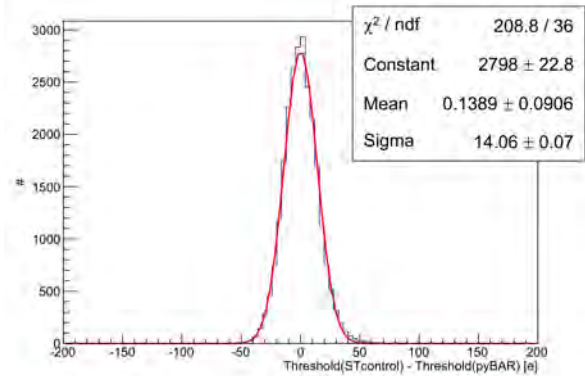
(a) Scatter plot

(b) Histogram

**Figure A.5.:** Difference of threshold measured by STcontrol and pyBAR of a Threshold Scan tuned to 1600 e. Illustrated in a scatter plot (a) and histogram (b). Error bars are not shown in the scatter plot.

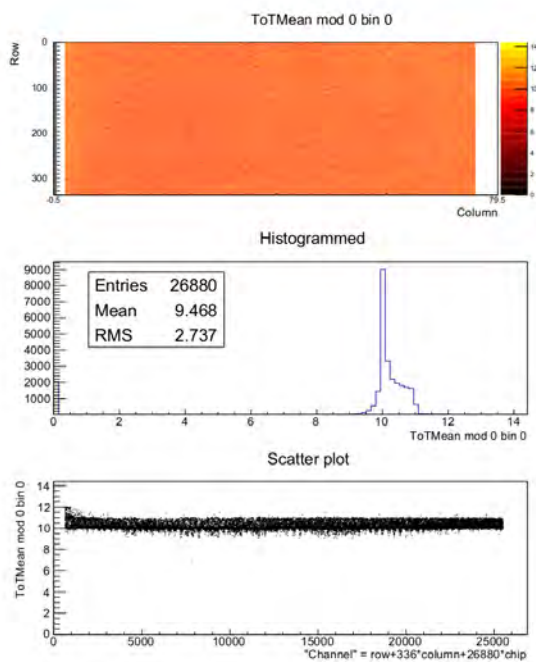


(a) Scatter plot

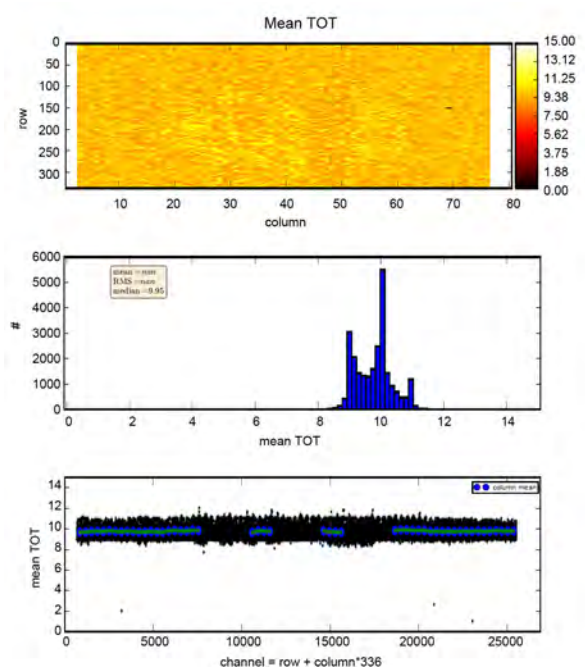


(b) Histogram

**Figure A.6.:** Difference of threshold measured by STcontrol and pyBAR of an untuned Threshold Scan. Illustrated in a scatter plot (a) and histogram (b). Error bars are not shown in the scatter plot.



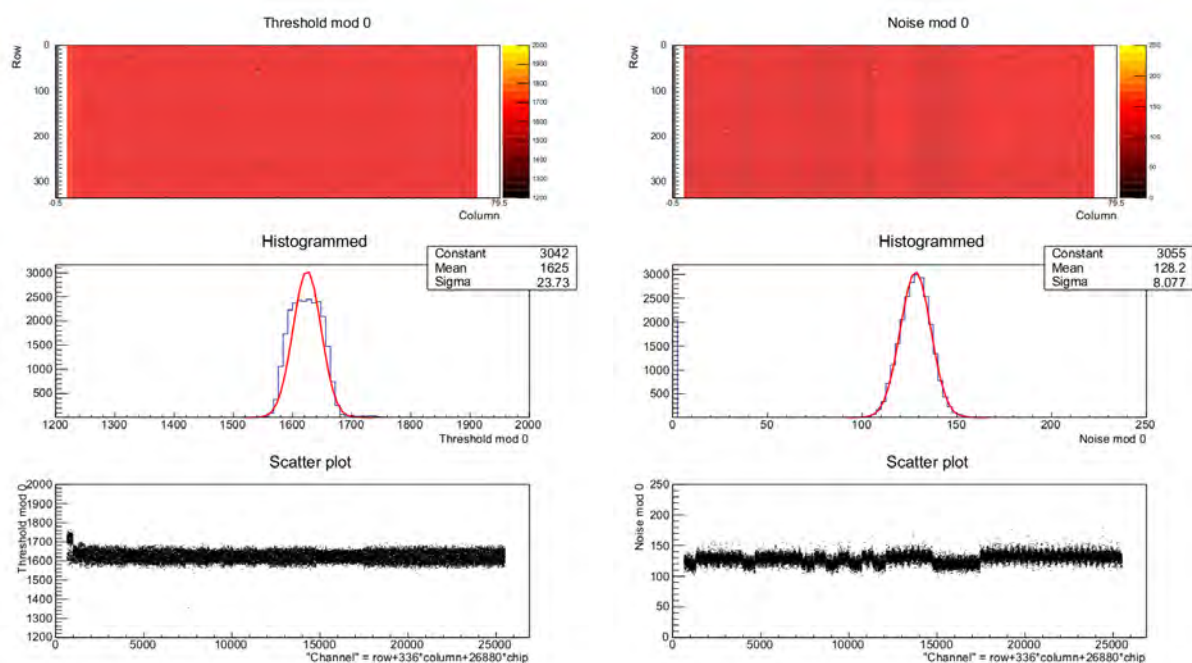
(a) STcontrol



(b) pyBAR

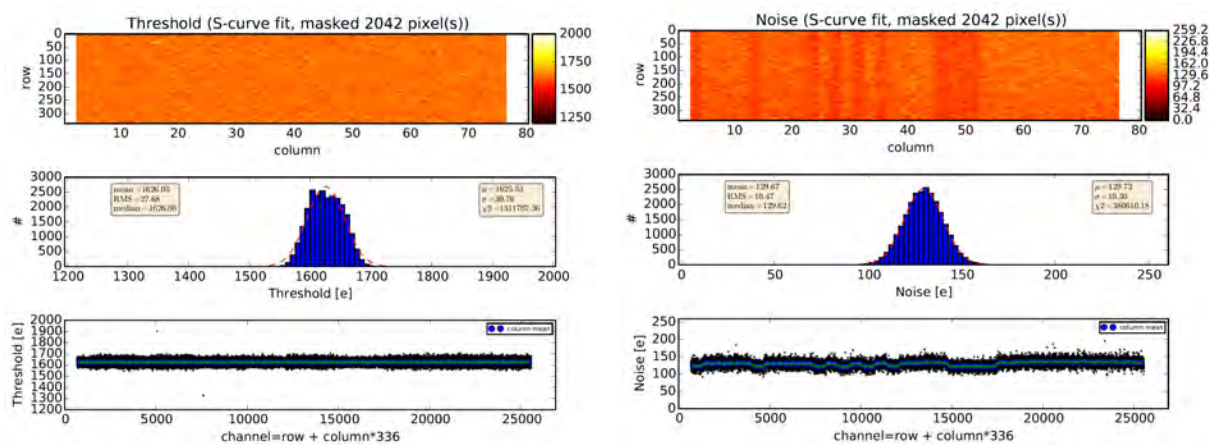
**Figure A.7.:** Comparison of the ToT Verification Scan for a separately tuning of STcontrol and pyBAR to a ToT of 10 for an injected charge of 15000 e.

## A. Additional Plots



(a) STcontrol threshold

(b) STcontrol noise



(c) pyBAR threshold

(d) pyBAR noise

**Figure A.8.:** Comparison of the Threshold Scan at a threshold tuned separately to  $1600 e$  by both programs. The scan results are summarized in Table 5.3.



# Bibliography

- [1] ATLAS Collaboration, *Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC*, Phys.Lett. B **716**, 1 (2012)
- [2] CMS Collaboration, *Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC*, Phys. Lett. B **716**, 30 (2012)
- [3] L. Evans, P. Bryant, *LHC Machine*, JINST **3**, S08001 (2008)
- [4] M. Lamont, *Status of the LHC*, J.Phys.Conf.Ser. **455**, 012001 (2013)
- [5] ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST **3**, S08003 (2008)
- [6] K. A. Olive, et al. (Particle Data Group), *Review of Particle Physics*, Chin. Phys. C **38**, 090001 (2014)
- [7] ATLAS Collaboration, *ATLAS Transition Radiation Tracker (TRT): Straw tube gaseous detectors at high rates*, Nucl.Instrum.Meth. A **732**, 277 (2013)
- [8] W.R. Leo, *Techniques for Nuclear and Particle Physics Experiments - A How-to Approach*, Springer-Verlag (1994)
- [9] S.M. Sze, *Semiconductor Devices*, John Wiley & Sons, Inc., New York (1985)
- [10] A. Macchiolo et al., *Performance of n-in-p pixel detectors irradiated at fluences up to  $5 \times 10^{15} n_{eq}/cm^2$  for the future ATLAS upgrades*, Phys.Procedia **37**, 1024 (2012)
- [11] P. Weigell et al., *Characterization and Testbeam Analysis of irradiated Silicon n-in-p Pixel Detectors for the ATLAS Upgrades*, 18th RD50 Workshop Liverpool (May, 2011)
- [12] I. Peric, *A novel monolithic pixelated particle detector implemented in high-voltage CMOS technology*, Nucl. Instrum. Meth. A **582**, 876 (2007)

## Bibliography

- [13] I. Peric (HVCMOS), *Active pixel sensors in high-voltage CMOS technologies for ATLAS*, JINST **7**, C08002 (2012)
- [14] FE-I4 Collaboration, *The FE-I4A Integrated Circuit Guide*, Unpublished Internal Note **Version 11.6** (Jun, 2011)
- [15] *USBpix - USB based readout system for ATLAS FE-I3 and FE-I4*, <http://icwiki.physik.uni-bonn.de/twiki/bin/view/Systems/UsbPix>, website, visited on July 4th, 2015
- [16] M. Backhaus et al., *Development of a versatile and modular test system for ATLAS hybrid pixel detectors*, Nucl. Inst. and Meth. A **650(1)**, 37 (2011)
- [17] M. Donzelli, *Studies of in-time charge measurement with the read-out chip of the first upgrade stage of the ATLAS Pixel Detector*, Georg-August-Universität Göttingen (Feb, 2012), Bachelor's Thesis

# Danksagung

Ich bedanke mich ganz herzlich bei allen, die mich im Verlauf meiner Bachelorarbeit motiviert und unterstützt haben.

Mein besonderer Dank gilt meiner Familie und Martina, die stets an mich geglaubt haben, sich um mich gekümmert haben und immer ein offenes Ohr für meine Probleme haben.

Ich bedanke mich bei Herrn Prof. Dr. Arnulf Quadt, der es mir ermöglicht hat meine Bachelorarbeit im II. Physikalischen Institut der Georg-August-Universität Göttingen zu schreiben.

Desweiteren möchte ich mich auch bei Priv.-Doz. Dr. Jörn Große-Knetter dafür bedanken, dass er für Fragen stets zur Verfügung stand und sich bereit erklärt hat mein Zweitgutachter zu sein.

Ein besonderer Dank gilt auch Julia Rieger und Dr. Jens Weingarten, die mir stets für Fragen zur Verfügung standen, mir in aller Ruhe bei Problemen weitergeholfen haben und meine Arbeit Korrektur gelesen haben.

Auch möchte ich mich bei meinem Mitbewohner Sven bedanken, der mir den Rücken freigehalten und mich mit gutem Rat immer wieder motiviert hat.

Nicht vergessen möchte ich an dieser Stelle meine Freunde und möchte mich bedanken für die Unterstützung und die nötige Ablenkung, um wieder mit neuen Gedanken zurück an die Arbeit zugehen.

**Erklärung**

nach §13(9) der Prüfungsordnung für den Bachelor-Studiengang Physik und den Master-Studiengang Physik an der Universität Göttingen:

Hiermit erkläre ich, dass ich diese Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe und alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen wurden, als solche kenntlich gemacht habe.

Darüberhinaus erkläre ich, dass diese Abschlussarbeit nicht, auch nicht auszugsweise, im Rahmen einer nichtbestandenenen Prüfung an dieser oder einer anderen Hochschule eingereicht wurde.

Göttingen, den 30. Oktober 2016

(Tobias Wolfgang Günter Westphal)