

# **Development of a New Revision for Floating, High-Voltage Picoamperemeters**

Florian Rössing

Masterarbeit in Physik  
angefertigt im Helmholtz-Institut für Strahlen- und Kernphysik

vorgelegt der  
Mathematisch-Naturwissenschaftlichen Fakultät  
der  
Rheinischen Friedrich-Wilhelms-Universität  
Bonn

06 2020

I hereby declare that this thesis was formulated by myself and that no sources or tools other than those cited were used.

Bonn, .....  
Date .....  
Signature

1. Gutachter: Prof. Dr. Bernhard Ketzer
2. Gutachter: PD Dr. Paul-Dieter Eversheim

Diese Arbeit wurde gefördert von der Stiftung Evolution.



# Acknowledgements

---

At first, I would like to thank Prof. Bernhard Ketzer for the opportunity to work on this topic. I really enjoyed my work in his group.

There are a lot of other people that for whose support I am very grateful. Thanks to Dima Schaab, who always had an ear for my problems. Dima had to deal with the most stupid questions and did this patiently.

Many thanks go to Dr. Christian Honisch for a lot of useful tips. He helped me out when I almost gave up on my prototype.

Thanks to Max Loepke for helping me out with my caffeine addiction over the past year.

I would not have made it through university without Backfisch, thank you people.

Last, but definitely not least, I want to thank my beloved fiancée.

*"Jetzt hamwa's bezahlt, jetzt schalten wir's auch an."*

- Marc-Uwe Kling, Die Känguru Chroniken



# Contents

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Theory</b>	<b>3</b>
2.1	Gaseous Particle Detectors . . . . .	3
2.1.1	Time Projection Chamber . . . . .	3
2.1.2	Gas Electron Multipliers . . . . .	5
2.2	Electronics . . . . .	6
2.2.1	Operational Amplifiers . . . . .	7
2.2.2	Analog-to-Digital Converters . . . . .	8
2.2.3	Electronic Noise . . . . .	11
2.2.4	Low-Level Electrical Effects . . . . .	13
2.2.5	Simulation of Electronic Circuits . . . . .	15
2.3	Precision Measurement Techniques . . . . .	16
2.3.1	Reducing Pick-Up Noise . . . . .	16
2.3.2	Reducing Leakage Currents . . . . .	16
2.4	Microcontrollers . . . . .	17
2.4.1	Timing . . . . .	17
2.4.2	Control & Status Registers . . . . .	18
2.4.3	Interrupts . . . . .	18
2.4.4	Programming . . . . .	18
2.4.5	Digital Communication Protocols . . . . .	18
2.5	XBee Operation . . . . .	19
2.5.1	Mode of Operation . . . . .	20
2.5.2	API Frame Structure . . . . .	20
2.5.3	Power Consumption and Sleep Mode . . . . .	21
<b>3</b>	<b>Initial Status of the Picoamperemeters</b>	<b>23</b>
3.1	The Backend Setup . . . . .	23
3.1.1	Mode Control . . . . .	23
3.2	The Front-end Setup . . . . .	24
3.3	High-Voltage Isolation . . . . .	27
3.4	Design Performance . . . . .	27
3.4.1	Power Consumption . . . . .	27
3.4.2	Measurement Results . . . . .	27
3.5	Infrastructure . . . . .	27
3.5.1	Firmware . . . . .	28

3.5.2	PAMOS . . . . .	30
3.5.3	CalibAna . . . . .	30
3.6	Motivation . . . . .	31
<b>4</b>	<b>Addressing the Issues</b>	<b>33</b>
4.1	Firmware Update . . . . .	33
4.1.1	Code Syntax . . . . .	33
4.1.2	File Structure . . . . .	34
4.1.3	Bidirectional Communication . . . . .	34
4.1.4	Changes to PAMOS . . . . .	35
4.1.5	Changes to CalibAna . . . . .	38
4.2	Non-Linear Measurements . . . . .	40
4.3	Over-Voltage Protection . . . . .	40
4.4	Alternative Over-Voltage Protection Circuits . . . . .	40
4.5	An OVP Prototype . . . . .	44
4.5.1	OVP Diode Choosing . . . . .	45
4.5.2	Voltage Generation . . . . .	45
4.5.3	Test Operation . . . . .	47
<b>5</b>	<b>Development of a new front-end</b>	<b>49</b>
5.1	Design of the Amplification Stage . . . . .	49
5.2	Circuit Layout . . . . .	51
5.3	Implementation of a Prototype . . . . .	56
5.4	Measurement Results . . . . .	62
5.4.1	Power Consumption . . . . .	63
5.4.2	Duty Cycle . . . . .	66
<b>6</b>	<b>Conclusion and Outlook</b>	<b>69</b>
6.1	List Of Acronyms . . . . .	71
<b>Bibliography</b>		<b>73</b>
<b>A</b>	<b>Appendix</b>	<b>77</b>
A.1	Picoamperemeter Revision Three . . . . .	77
A.2	The Calibration Station . . . . .	77
A.3	Prototype Iteration One Schematics . . . . .	81
A.4	Prototype Iteration Two . . . . .	81
A.5	List Of Acronyms . . . . .	82
<b>List of Figures</b>		<b>91</b>
<b>List of Tables</b>		<b>95</b>

# CHAPTER 1

---

## Introduction

---

Modern, large scale physics experiments push the limits for achievable rates as well as precision and accuracy. One prominent example is the ongoing upgrade of the *Large Hadron Collider* (LHC) to achieve a ten times higher luminosity [1]. To keep up with the update, the affiliated experiments at the European Organization for Nuclear Research (CERN) are receiving an upgrade as well. As an example, *A Large Ion Collider Experiment* (ALICE) receives several upgrades. The inner tracking system and the muon forward tracker get replaced by new pixel sensor chips. For the Time Projection Chamber (TPC), a new amplification and readout stage is required. Additionally, a new trigger system is set up. For details on the upgrade see [2].

In order to produce reliable results with such experiments, the fundamentals of their behaviour need to be well understood and precisely quantized. Physicists all around the world conduct small-scale experiments with the hardware and collaborate in order to meet this goal. In the scope of these studies, specialized hardware is developed. One example of this hardware is the picoamperemeter (pAM), originally developed at TU München [3–5]. It allows precision current measurements, in the picoampere regime, on high voltage lines. The pAMs are used for studies on micro-pattern gaseous detectors. To keep up with the requirements of the large scale experiments, equipment as the pAM need to be updated and improved regularly.

In the scope of this thesis, a new revision of pAMs is developed. At first, in chapter 2, background informations necessary for this thesis are provided. In chapter 3, an analysis of the existing design is conducted, revealing weak spots and possible tasks to be addressed. The next step is to find solutions, that could improve the design; which is addressed in chapter 4. On the software side, a new firmware for the pAM is developed, allowing a more flexible use of the devices. On the hardware side, the analysis mainly focusses on the over-voltage protection (OVP). An improved OVP is suggested and evaluated. These considerations lead to the conclusion that the problems could be overcome by a new input stage. Hence, a new front-end, based on a transimpedance amplifier (TIA), is developed and tested in chapter 5. The main goal of the revision is to allow a higher readout rate and more precise measurements at the same time.

A lot of different programs are part of this thesis. For the sake of simplicity, the full software is not printed here but is available on the AG Ketzer GitLab. The AltiumDesigner files for the different designs and first contributions to a documentation for the devices can be found there as well.



# CHAPTER 2

---

## Theory

---

Even though they do not play a significant role in this thesis, a short introduction on Gaseous Particle Detectors is provided here, in order to explain the requirement of a picoamperemeter.

### 2.1 Gaseous Particle Detectors

Detection of ionizing particles using gas-filled detectors is a long-known principle. A first approach was the Geiger-Müller tube, useful for detection of ionizing radiation, e.g. from radioactive decays. Detectors as the Geiger-Müller tube combine a metal cylinder with a thin metal rod in its centre. The cylinder is filled with gas and high voltage is applied between the metal wall as the cathode and the central anode. Particles traversing the tube volume transfer some of their kinetic energy to the detector gas, thus ionizing the gas. The charges created in the ionization process are separated by the electric field and drift towards the corresponding electrode creating a detectable signal. If the applied voltage is high enough, the electrons close to the anode get strongly accelerated due to the high electric field. The energy gained through the acceleration is sufficient to further ionize the gas, creating an avalanche. The underlying processes of secondary ionization, cause the resulting charge to be approximately proportional to the primarily deposited charge. Hence such a detector is called a proportional counter. The principle was further developed into the multi-wire proportional counter (MWPC) by G. CHAPRAK in 1967 [6]. The MWPC is constructed using two parallel cathode planes, and multiple parallel wires running in between them as anodes. Such a setup allows particle detection with a limited spatial resolution. The resolution can be increased significantly by using segmented cathodes, measuring the induced signal over the segments.

#### 2.1.1 Time Projection Chamber

Combining a MWPC with a large drift volume results in a TPC [7], which is a true 3D-tracking device. A basic sketch of a TPC setup is shown in figure 2.1. The largest part of a TPC is the gas-filled drift volume. One end acts as the drift cathode, the opposing end incorporates the readout stage. The amplification and readout stage provides a 2D information in  $x$  and  $y$ . The  $z$  component can be derived from the drift time of the electrons. The amplification stage of a TPC with a MWPC, as depicted in figure 2.2, uses three wire planes. The first one is the gating grid, that keeps the ions, created in the amplification process, from drifting back into the detector volume. When an event is expected, the

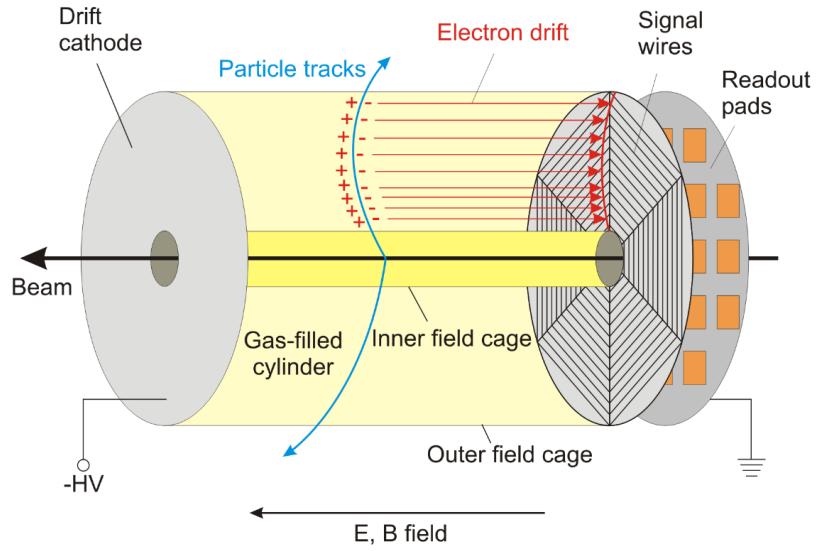


Figure 2.1: Principle setup of a TPC. From [8].

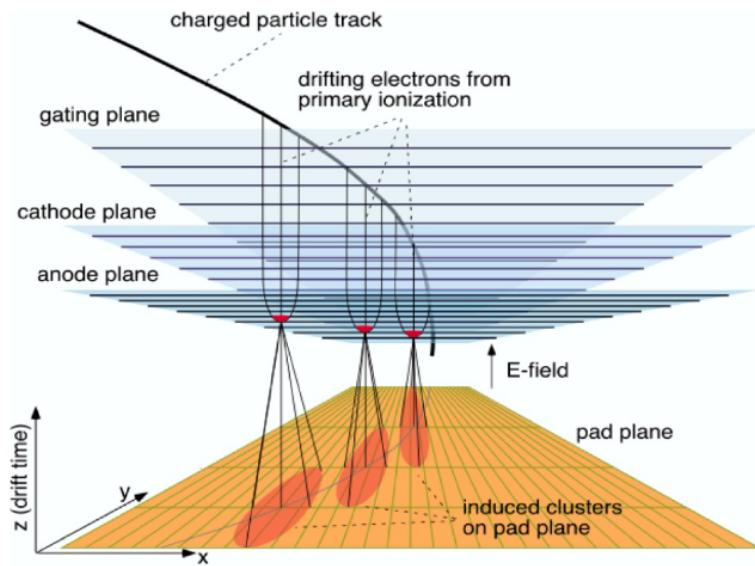


Figure 2.2: Amplification and readout stage of a TPC with a MWPC based amplification stage. From [8].

gating grid is switched off to allow electrons to pass through it. The need to switch the gating grid is a strong limit on the achievable event readout rate. The cathode and anode plane form the amplification region as known from a MWPC. Below the anode plane, a pad plane is implemented to derive the position resolution by measuring the signal induced from the ions created in the amplification region.

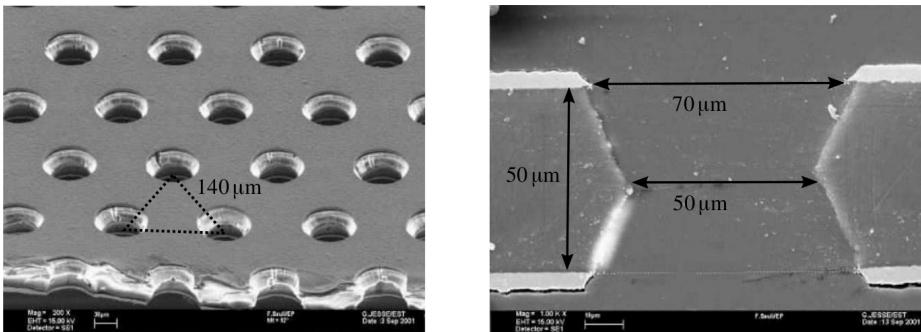


Figure 2.3: Electron-microscope image of a GEM-foil. Dimensions are given on the image. From [9].

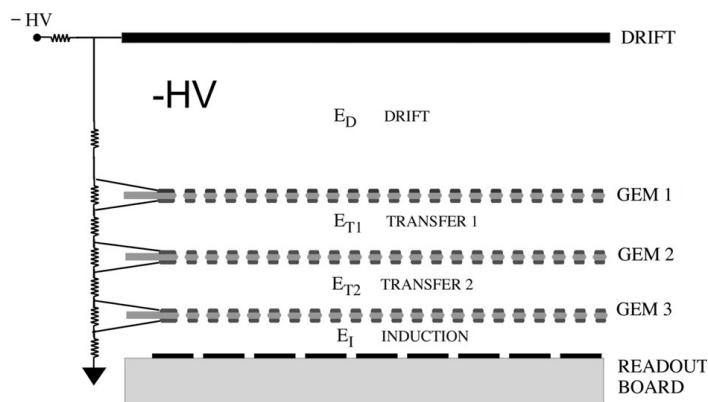


Figure 2.4: A triple GEM stack with the drift, transfer and induction fields. From [6].

### 2.1.2 Gas Electron Multipliers

Traditional approaches for implementing the amplification stage for a gaseous detector are wire-based, as in the MWPC. A "newer" approach is to use a Gas Electron Multiplier (GEM), which is a thin polyimide foil with a copper coating on both sides. For electron multiplication, the foil is perforated with a high density of holes, typically  $100 \text{ mm}^{-2}$ . The GEM was invented by F. SAULI in 1997 [9].

A photograph of a GEM can be found in figure 2.3. The two copper coatings of a GEM-foil are set to a voltage difference of around 400 V, creating a very strong electric field, in the order of  $\text{kV cm}^{-1}$ , in the holes. Electrons created in the drift volume of a detector, see figure 2.4, drift along the drift field  $E_D$  towards the GEM. The electric field guides the electrons into the holes of the GEM. Inside these holes, electrons undergo avalanche multiplication due to the strong field. The created secondary electrons are extracted by the extraction field. In case of the GEM stack in figure 2.4, the extraction field for one GEM is the drift field for the next GEM. The majority of ions created in the avalanche process drift to the topside copper coating and get eliminated this way. Typical particle paths for electrons and ions are depicted in figure 2.5.

For the upgrade of the ALICE TPC, the multi-wire approach, as described above, was replaced by a GEM setup. As discussed in [10], a TPC utilizing a GEM amplification stage allows readout with faster rates, as a gating grid is no longer necessary. In the studies of GEM-foils, the pAMs are widely used. As an example, two topics are briefly described below. Other effects that can be studied using

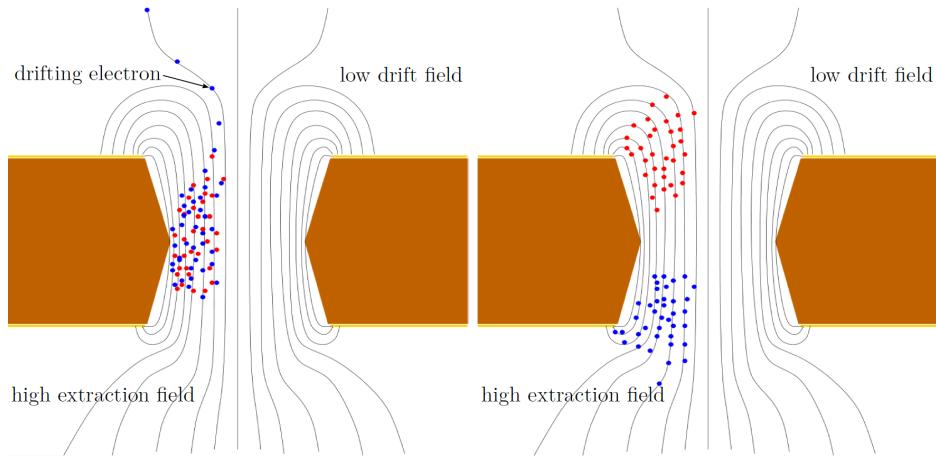


Figure 2.5: Left: An incoming electron enters a GEM hole and ionizes the gas and creates electron (blue) - ion (red) pairs. Right: Electrons are extracted, ions drift back and end up on the copper coating due to their low diffusion and the low drift field. From [10].

the pAM include gain measurements, charge transfer and collection efficiency, see e.g. [11].

### **Ion Backflow**

In contrary to the above stated, some of the ions, that are created in the multiplication process, manage to drift back into the drift volume. This effect is called Ion Backflow. Ions accumulating in the drift volume will disturb the electric field. To quantize ion backflow, precise measurements of all currents in a GEM setup are necessary. Studies of the ion backflow were done in preparation of the ALICE upgrade, see e.g. [12].

### **Charging-Up effect**

During operation of a GEM, some of the charges created in the amplification process may adhere to the polyimide core of the GEM. These charges accumulate over time and influence the electric field inside the holes [13]. This charging-up of a GEM-foil can cause time variations in the effective gain. Figure 2.6 shows a measurement of the anode current of a GEM detector in operation. Together with the knowledge of the input charge, this can be used to determine the effective gain of a GEM. Such measurements were done by [13] to quantize the charging-up effect.

## **2.2 Electronics**

The following section treats some essential parts of electronics that are used in this thesis. At first an insight in Operational Amplifiers (Op-Amps) and Analog-to-Digital Converters (ADCs) is provided. After this, some effects that can disturb precision measurements are introduced. Unless otherwise stated, this section refers to [14].

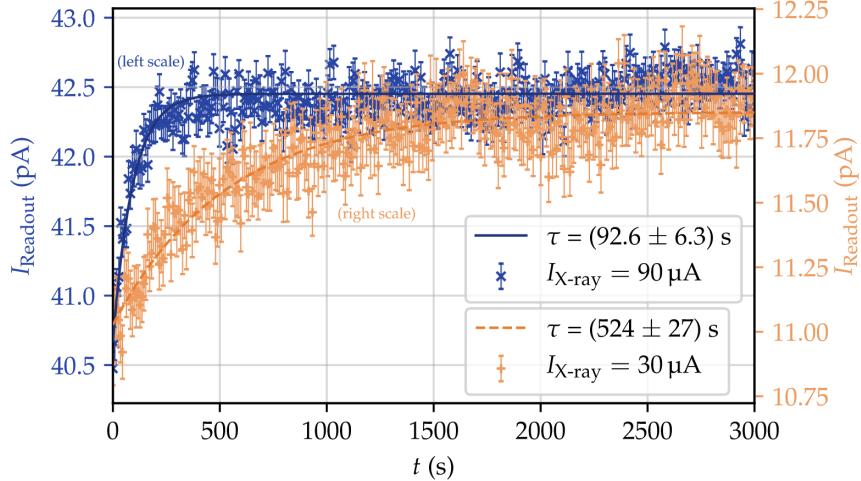


Figure 2.6: Measurements of current from the pad plane of a GEM detector done with a pAM. From [13].

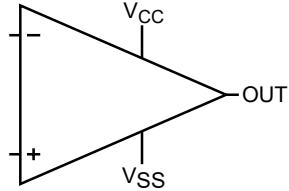


Figure 2.7: Symbol used to depict an operational amplifier in circuit schematics.

### 2.2.1 Operational Amplifiers

Op-Amps, are electrical amplifiers with a very high, theoretically infinite, gain. The conventional schematic symbol for Op-Amps is depicted in figure 2.7. The inputs labelled '+' and '-' are referred to as non-inverting and inverting inputs.  $V_{SS}$  and  $V_{CC}$  are the supply voltages; they can be uni-, or bipolar depending on the amplifier design. Typically Op-Amps are used in combination with a feedback network. The characteristics for an Op-Amp with feedback is determined by the feedback network alone due to the high open-loop gain<sup>1</sup>. The behaviour of an amplifier with feedback can be calculated using Ohm's law, Kirchhoff's laws and the so-called *golden rules* for Op-Amps [14]:

- The output adjusts in order to zero the input voltage difference.
- No current flows into the inputs.

However, this is only true for an ideal Op-Amp. Real Op-Amps deviate from this behaviour. These deviations are expressed by different parameters usually specified in the datasheet. A short introduction on some of these parameters is provided here.

<sup>1</sup>The open-loop gain is the intrinsic gain of an amplifier. In contrast to the closed-loop gain, which is defined by the feedback network.

### Input Bias Current

The golden rules state that no current flows into the inputs of an Op-Amp. In real amplifiers, a small current still is flowing into the inputs. This is called input bias current,  $I_B$ . This current is depending strongly on the amplifier but is in the range of nA down to fA.

### Input Impedance

The input impedance is defined as the impedance of one input with the other input grounded. For an ideal amplifier, the input impedance is infinite; for real implementations, it depends on the specific input stage. The input impedance can range from some MΩ up to 100 TΩ.

### Common-mode Input Range

Op-Amps are usually designed to work with input voltages within the supply voltage range. Input voltages outside of this boundary can lead to drastic gain changes or malfunction of the amplifier.

### Input Offset Voltage

The input offset voltage ( $V_{OS}$ ) is the voltage difference between the inputs, that is required to set the output to zero volt. This voltage can have a strong influence on precision measurements, shifting the zero line by several μV. Some amplifiers offer the possibility to compensate for this voltage. The offset voltage drifts with time and temperature.

### Gain-Bandwidth-Product

For low frequencies and closed-loop gains, that are small compared to the open-loop gain, the frequency behaviour of an Op-Amp is determined by the feedback network. The open-loop gain, however, is only stable up to an amplifier specific-frequency, due to low-pass filter effects in the amplifier. Beyond this point, the open-loop gain decreases continuously, see figure 2.8. This behaviour is quantized with the Gain-Bandwidth-Product (GBW). It is the product of the achievable open-loop gain with the desired bandwidth. The roll-off of the open-loop gain limits the closed-loop gain as well.

## 2.2.2 Analog-to-Digital Converters

Measurements done with electronics usually take an analogue input signal. For further processing and data analysis, the analogue signal needs to be digitised, making use of an Analog-to-Digital Converter. The two main characteristics of an ADC are sampling rate, given in samples per second or sps, and resolution given in bits. Choosing the best ADC can sometimes be a puzzling task, as there are various approaches to implement an ADC, all with their advantages and disadvantages. The three main architectures are Delta-Sigma ADC, Pipelined ADC and SAR ADC [16–18], all three are briefly explained here.

### Sigma-Delta ADC

A Sigma-Delta ADC ( $\Sigma\Delta$  ADC) offers a high resolution, with the downside of lower sample rates. It incorporates an integrator, a comparator and a 1-bit Digital-to-Analog Converter (DAC) a simple

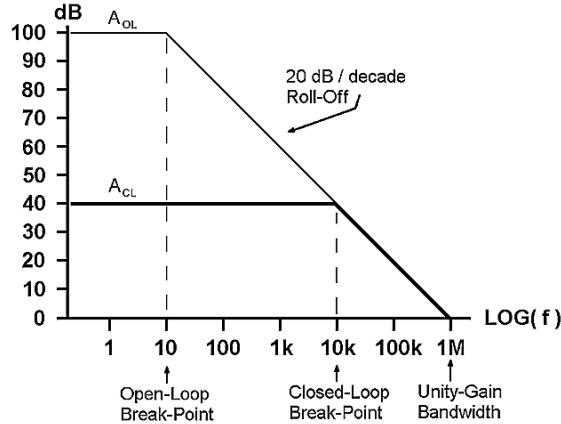


Figure 2.8: Open-loop gain  $A_{OL}$  of an operational amplifier rolling-off at 10 Hz. The closed-loop gain  $A_{CL}$  is affected by the roll-off at higher frequencies, where it is comparable to the closed-loop gain. From [15], modified.

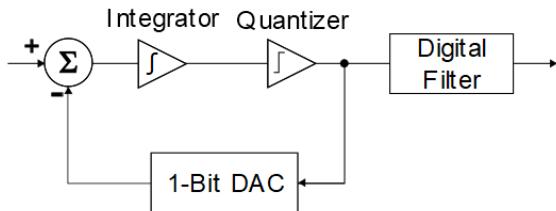


Figure 2.9: Principle of a first-order  $\Sigma\Delta$  ADC. From [19].

block diagram is shown in figure 2.9. A  $\Sigma\Delta$ -ADC is repeating three steps, to achieve a precise result. First, calculating the difference ( $\Delta$ ) between the reference from the DAC and the input signal. Second, integrating the difference ( $\Sigma$ ) and discriminating the integrator output. Third, switch the DAC reference according to the discriminator output.  $\Sigma\Delta$  ADC with resolutions of 8 to 32-bits and sampling rates of up to 1 Msps are commonly available.

### Pipelined ADC

Pipelined ADCs are the fastest of the three kinds of ADCs discussed here; they are available with sampling rates higher than 100 Msps. The downside is the lower resolution of 8 to 16-bit. A pipelined ADC uses several independent stages, hence pipelined. As an example, the principle of a 12-bit ADC is shown in figure 2.10. The input signal of each stage will be digitised using a fast 3-bit flash ADC<sup>2</sup>. The result is then converted back by a 3-bit DAC. The difference of this output and the original input signal is then amplified and fed to the next stage. After four stages the remaining signal is digitised with 4-bit flash ADC. From the 3-bit and the 4-bit outputs of the single stages, the final 12-bit signal is generated. The stages of such an ADC can operate in parallel, allowing four simultaneously running conversions.

<sup>2</sup> A flash ADC is an extremely fast converter that uses a voltage ladder and comparators. High resolution flash ADCs are hard to realize. For a  $n$ -bit flash ADC  $2^n - 1$  precision comparators are needed.

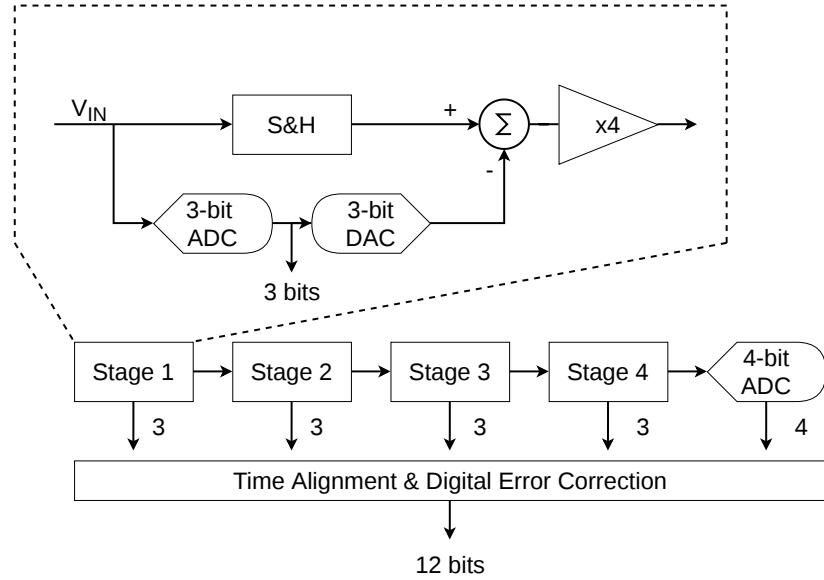


Figure 2.10: A 12-bit pipelined ADC. From [20], modified.

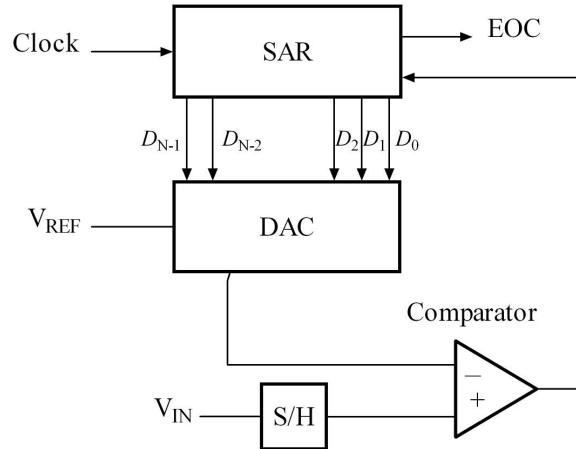


Figure 2.11: Block diagramm of an N-bit SAR ADC. From [21].

## SAR ADC

The midway solution between sampling speed and resolution is the Successive-Approximation-Register (SAR) ADC. Such ADCs are available with sampling rates of up to 10 Msps and resolutions of 8 to 18-bits. On start of a conversion, the SAR is filled with an arbitrary value. The ADC samples and holds the input voltage, which is then compared with a DAC output of the SAR. The comparator output is used to higher or lower the SAR value. This process is repeated until changes occur only on the least significant bit (LSB).

### Input Voltage Range

Another important characteristic of an ADC is its input voltage range. Together with the resolution, the input range defines the voltage level of a single bit, the LSB voltage. Most commonly, an ADC can only digitise positive voltages, therefore, they are called unipolar. A few ADCs are capable of digitising positive and negative voltages, hence bipolar.

#### 2.2.3 Electronic Noise

For electronic precision measurements, noise is a factor that needs to be considered to evaluate the achievable precision. Noise in this context is the random noise generated by electronic components. An insight of the variant sources of noise and their characteristics is provided here.

##### Johnson Noise

Johnson noise, also known as Johnson-Nyquist noise or just thermal noise is the noise generated by every resistance from thermal effects. It has a flat spectrum, which is why it is often called "white noise". The Johnson voltage noise can be calculated using:

$$V_{\text{noise}}(\text{rms}) = \sqrt{4k_B T R \Delta f}, \quad (2.1)$$

with the Boltzmann constant  $k_B$ , the temperature  $T$  in K,  $R$  the resistance and the bandwidth  $\Delta f$ . The amplitude at any moment in time can neither be calculated nor predicted but follows a gaussian probability distribution. It is the absolute lower limit on noise voltage in any circuit. The noise voltage can be converted into a noise current by dividing with the resistance  $R$ . Johnson noise can be represented by adding either a voltage noise source in series with a noiseless resistor or a current noise source in parallel to a noiseless resistor.

##### Shot Noise

Shot noise is caused by the quantized nature of charges, leading to small fluctuations in the current flow. It can be expressed via:

$$I_{\text{noise}}(\text{rms}) = \sqrt{2eI\Delta f}, \quad (2.2)$$

with the elementary charge  $e$ , the current flowing  $I$  and the bandwidth  $\Delta f$ . The equation is valid only under the assumption that the charge carriers behave independently of each other.

##### 1/f Noise

In contrary to Johnson and shot noise, the  $1/f$  noise or flicker noise can not be derived purely by physical principles. Various sources influence the exact value of this noise, but they combine to a spectral density inversely proportional to a power of the frequency.

##### Pick-Up Noise

Pick-up noise, also known as interference, summarizes all various kinds of noise that are picked up from external sources. A prominent example is the pickup of the 50 Hz signal from the power grid, but

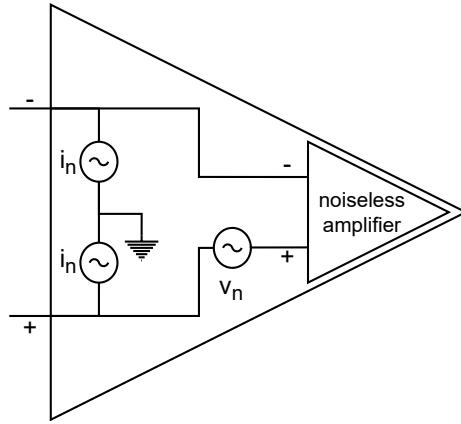


Figure 2.12: Model to ease the noise calculation for Op-Amp circuits. With  $i_n$  denoting the current noise density and  $v_n$  the voltage noise density. From [22], modified.

there are plenty of other sources. In contrary to the other noise sources discussed here, interference noise can not be calculated or even approximated. Interference noise can, however, be reduced by shielding or filtering.

### Noise Spectral Density

From the above examples, it arises that noise is a frequency-dependent property. As a convention, noise is expressed in terms of noise spectral density  $v_n$ , given in  $\text{V}/\sqrt{\text{Hz}}$ . For a resistor, this would yield:

$$V_{\text{noise}}(\text{rms}) = \sqrt{4k_B T R} \sqrt{\Delta f} = v_n \sqrt{\Delta f}. \quad (2.3)$$

Equivalently for current noise, the current noise density  $i_n$  is defined.

### Amplifier Noise

Operational amplifiers are complex devices; an exact calculation of the noise is impossible. Manufacturers, therefore, determine the noise of their amplifiers by measuring the characteristics for larger quantities of the product. To create a usable model, the measured noise of an amplifier gets split up in current-based and voltage-based noise. The noise will then be combined in noise sources, as depicted in figure 2.12. The input current noise density  $i_n$  is summarized in two sources between the inputs and ground. The voltage noise density  $v_n$  is summarized in a single noise source in series with the non-inverting input [22]. The values for  $i_n$  and  $v_n$  are usually given in the amplifiers datasheet for different frequency regimes.

### Quantisation Error

The conversion of an ADC introduces an additional kind of error. The approximation of a continuous value into a digitised produces an error. The corresponding noise density can be calculated using:

$$v_n = \frac{q_s}{\sqrt{6\Delta f}}, \quad (2.4)$$



Figure 2.13: The Triboelectric Effect depicted by a cat. From [26].

where  $q_s$  is the quantisation step size [23].

### Calculating Output Noise

When all the different noise sources in a circuit are known, the overall output noise can be calculated. This can be done utilizing the principle of superposition. The output noise for every component is calculated individually; Quadratically adding up the results yields the total output noise [22].

#### 2.2.4 Low-Level Electrical Effects

Besides the noise discussed above, there are some other electrical effects that need to be kept in mind when working with precision electronics. For normal applications, these effects are often neglected due to their low influence, but on the pA or  $\mu$ V level they can have a significant influence. This section refers to [24] unless otherwise stated.

### Diode Leakage Currents

A diode, in theory, is a device that is conductive only in one direction and has an infinite resistance in the other. The diodes commonly used in electronics are semiconductor diodes. A semiconductor diode is formed by a junction of a p-doped with an n-doped semiconductor. A depletion region is formed at the junction. In case of a forward-biased diode, the voltage difference allows the majority charge carriers<sup>3</sup> to pass the depletion zone, and the diode becomes conductive.

In case of a backwards biased diode, the depletion zone gets widened, effectively blocking the diode for the majority charge carriers. However, the minority charge carriers can still pass the depletion region resulting in a very low current flow through the diode [25]. For normal operations this current is negligible, but as measured in section 4.5.1 can have a considerable influence in the pA range. As the generation of minority charge carriers in semiconductors increases with temperature, the leakage current scales with temperature.

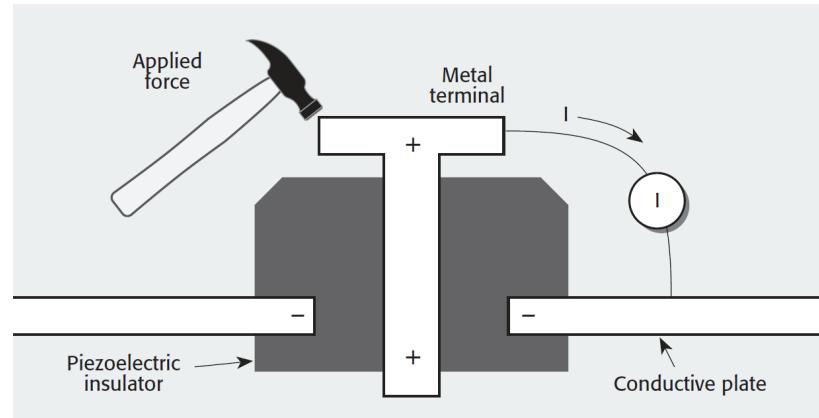


Figure 2.14: Piezoelectric current generated through mechanical stress on a plastic socket. From [24].

### Triboelectric Effect

The Triboelectric effect, see figure 2.13, occurs, when certain materials get in contact and then get separated again. Due to unknown effects in the contact area, the materials exchange electrons, creating a charge imbalance [27]. On separation, the two materials are left charged up. This effect is the cause of Electrostatic Discharges (ESDs), that can damage electronic components. Triboelectricity can also occur in cables due to insulator and conductor rubbing against each other. The charges created can result in a current flow, reducing the accuracy of low-level current measurements. The effect can be reduced using special cables, that coat the insulation with graphite.

### Piezoelectric Effect

In some materials as crystals, ceramics or biological materials, mechanical stress can cause small current flows. In some plastics, stored charges can cause a similar effect [24], as shown in figure 2.14.

### Thermoelectric Effects

Thermoelectricity incorporates three similar effects. The Seebeck effect, the Peltier effect and the Thomson effect. The first two are briefly explained here.

### Seebeck Effect

The Seebeck effect occurs in material junctions with a temperature gradient. In such junctions, a small voltage is generated. The effect is quantized with the so-called Seebeck coefficient, usually given in  $\mu\text{V}/^\circ\text{C}$ . A small collection of Seebeck coefficients is given in table 2.1. The effect is sometimes used to power small devices but can also influence low-level measurements. Sockets are prone to build a small Seebeck generator from an oxidised copper contact.

---

<sup>3</sup>In case of the p-doped regions the majority charge carriers are holes. For n-doped regions, the electrons are the majority charge carriers.

Material Junction	Seebeck Coefficient
Cu - Cu	$\leq 0.2 \mu\text{V}/^\circ\text{C}$
Cu - Pb/Sn	$\leq 1-3 \mu\text{V}/^\circ\text{C}$
Cu - Si	$\leq 400 \mu\text{V}/^\circ\text{C}$
Cu - CuO	$\approx 1\,000 \mu\text{V}/^\circ\text{C}$

Table 2.1: Different junction Seebeck coefficients. From [24], modified.

### Peltier Effect

The Peltier effect is a reversed Seebeck effect. When a current flows through a material junction, a temperature difference is created. This effect is widely used for cooling or heating applications.

### Contamination Effects

Accumulated dirt, solder flux and moisture can degrade the performance of precision measurements. They can drastically decrease insulation resistance on printed circuit boards (PCBs) or even cause electrochemical reactions. Such reactions create low-voltage batteries [24]. *Analog Devices* performed test measurements with flux residues, resulting in a weak battery with an open-circuit voltage of 15 mV [28]. Contaminations can be eliminated by proper cleaning.

### 2.2.5 Simulation of Electronic Circuits

For the verification of an electronic design, building it is the most direct approach. However, this is often not the best option, as it can be rather costly and time-consuming. An alternative is to simulate a circuit with the aid of computers. Today's standard program for circuit simulation is the Simulation Program with Integrated Circuit Emphasis (SPICE). As an input SPICE takes ASCII text files that describe a circuit combining sources, simple devices as resistors, non-linear devices as diodes and more complex parts as integrated circuits. To ease the use of SPICE, various graphic interfaces are provided, amongst these are LTspice by Analog Devices [29] or TINA by Texas Instruments (TI) [29]. The graphic interfaces generate the ASCII simulation files from a drawn circuit. SPICE allows different forms of simulation, the most important are briefly described here. For a more detailed introduction to SPICE refer to [30].

### The DC Analysis

This form of analysis calculates the voltage for all circuit nodes, with a DC source. In this simulation, capacitors are treated as disconnected and inductors as short-circuits. The non-linear behaviour of semiconductors is taken into consideration.

### The AC Analysis

SPICE can also be used to calculate the complex node voltages as a function of frequency, under consideration of impedances and non-linear effects. To produce a reliable solution a small signal assumption is made. A variant of the AC analysis is the noise analysis, where all components are

expanded with a noise source, resulting in the total noise spectral density. The noise analysis includes the types of noise from section 2.2.3.

### The Transient Analysis

Additionally SPICE can perform time analysis of a circuit, with an input signal of adjustable shape.

## 2.3 Precision Measurement Techniques

To allow precise and accurate measurements, special care must be taken. The two main issues that can be controlled by careful layouting are interference noises and leakage currents.

### 2.3.1 Reducing Pick-Up Noise

The combined noise originating from components as resistors or amplifiers, that can be calculated as explained above, is irreducible. The pick-up noise, also known as electromagnetic interference, however, can be reduced. It is created by electromagnetic fields via induction or electrostatic coupling. The influence of interference can be reduced by shielding. Simply speaking, to shield a circuit it is surrounded with metal box or mesh that is connected to ground, also known as a Faraday cage. Shielding relies on the induced currents from the electric fields and the eddy currents created by magnetic fields [31].

### 2.3.2 Reducing Leakage Currents

For current measurements in the pico- or even femtoampere range, leakage currents can be a big limit on achievable accuracy. For regular applications, printed circuit boards are treated as perfect insulators. This assumption is no longer valid for applications with high impedances. A typical PCB material is Flame Retardant 4 (FR-4), which is a composite of epoxy and fibre-glass. FR-4 PCBs have a finite surface resistance of typically  $1\text{ T}\Omega$  ( $10\text{ G}\Omega$  minimum) [32]. This resistance can cause leakage currents between nodes at different voltages. There are multiple ways to reduce such leakage currents.

### PCB Materials

A first option is to go for PCB materials with higher resistance. A variety of different materials for special-purpose applications exist, like RF electronics or low-leakage. In contrast to typical FR-4 boards, these often use Ceramic or Teflon laminates. An example would be the RO400 series from Rogers with a specified surface resistance of  $4\text{ 200 T}\Omega$  [33]. The usage of pure insulators, like a slab of Teflon is no option, as it can lead to accumulation of surface charges that can not dissipate [34].

### Insulating Stand-offs

When special PCB materials are too expensive or not practical, another option is the usage of insulating stand-offs. They are made from a good insulating material as Ceramic or Teflon, that isolates the node from the board. The circuitry is soldered floating over the PCB, with the downside of mechanical instability.

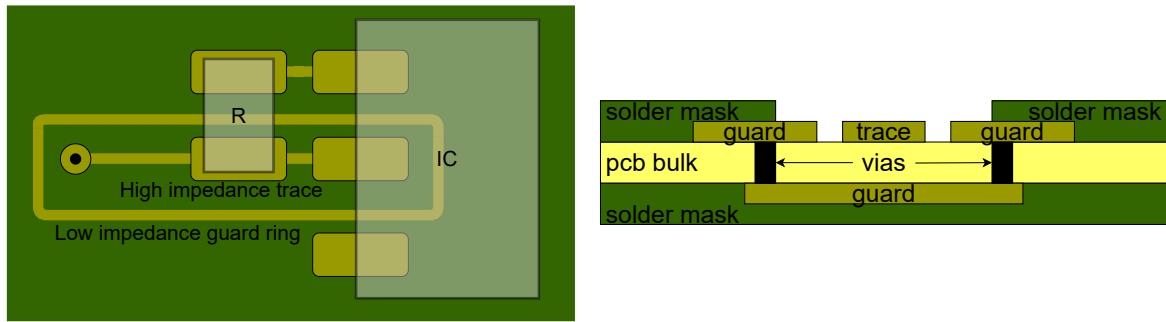


Figure 2.15: Left: Guard ring surrounding a high impedance IC input trace. For best performance, the guard trace should be hooked up to a driving circuit. Right: Implementation of a guarding structure through the PCB bulk. For connecting the guard ring and the guard plane on the bottom side, vias are implemented. Solder mask between trace and guard is removed to reduce leakage paths.

### Guarding

Guarding is a technique where high-impedance nodes or traces are surrounded with a low impedance trace that is at a similar potential. Guarding techniques can be implemented on PCBs, as well as in cables. An example of guard ring implementation on PCB level is shown in figure 2.15. A guard structure prevents current flows from the shielded trace to nodes of another potential by surrounding the trace with an equipotential. The guard ring can be extended by using a via stitching to connect top and bottom layers. Via stitchings are usually used for radio frequency applications but come at no extra cost and can, therefore, be implemented right away.

### Cleaning

Proper cleaning of the PCB can prevent the degradation of insulation resistance and the forming of electrochemical batteries. There are many different guides on how to clean a PCB provided by part suppliers like [35] or PCB manufacturers as [36] or in datasheets of components like [28]. Two approaches are often recommended, a brush with alcohol for cheap and coarse cleaning, and an ultrasonic cleaner for best performance.

## 2.4 Microcontrollers

A Microcontroller ( $\mu$ C) is a variant of Integrated Circuit (IC) combining a microprocessor with memory, timing modules and I/O modules with the advantage of decreased size. Some basics necessary for an understanding of this thesis will be summarized here. A more detailed introduction can be found in [37]. As the pAM use an MSP430f169  $\mu$ C from TI, the specifications given here refer to this particular device, for more details see [38].

### 2.4.1 Timing

To ensure, that operations in  $\mu$ Cs have a comparable timebase, they are all synchronized to a common clock. There are different ways to generate a clock signal, amongst of the most prominent are crystal oscillators making use of the Piezo effect, see paragraph 2.2.4. Such Crystals produce voltage signals

of fixed frequency. The MSP has terminals for a 32 kHz and a 1.8 MHz clock. The clock frequency determines the minimal time per operation; a  $\mu$ C is limited to one action per clock pulse. A counter module, counting clock cycles, is used to measure time in a  $\mu$ C. The MSP comes with two 16-bit time counters.

#### 2.4.2 Control & Status Registers

In general,  $\mu$ C are very flexible and offer lots of adjustable features. All these configurable options are summarized in control registers. A register is an 8 or 16-bit memory. Every-bit in a register represents an option, that can be turned on or off. The status of the clock counter and the input/output (I/O) pins are saved in status registers and can be read from there. The registers for the various modules of the MSP can be found in [38].

#### 2.4.3 Interrupts

A  $\mu$ C often needs to react to external conditions, signalled by a change of state on one of the I/O pins. One way to do this is to periodically check the pin state in the program flow, this method is called polling. Polling has a lot of disadvantages, in terms of time resolution, waste of processing time and power consumption. To circumvent this,  $\mu$ Cs use interrupts. The interrupt logic is implemented independently of the program flow and will halt the running program when an interrupt condition occurs. An interrupt condition is defined by two-bits. The interrupt enable (IE), and the interrupt flag (IFG). The IE is set in one of the control registers and indicates if an interrupt is allowed. The IFG is part of a status register and is set automatically when the corresponding interrupt condition occurs. If IE and IFG are both set, the program is halted, and the interrupt service routine (ISR) is called. An ISR for every enabled interrupt has to be provided by the programmer. The MSP as well as most other  $\mu$ Cs has multiple possible interrupts. As switching all these IEs is time consuming,  $\mu$ Cs have an additional general interrupt enable (GIE).

A special interrupt is the Watchdog, that is present on most  $\mu$ Cs. The Watchdog is a timer module that causes an interrupt on overflow, leading to a restart of the  $\mu$ C. Its main purpose is to ensure that the  $\mu$ C does not get stuck in the program flow.

#### 2.4.4 Programming

For a  $\mu$ C to perform a task, it needs to be programmed accordingly. In case of the MSP, a JTAG interface is implemented. It can be used to hook up to a flashing tool provided by TI. Via the flashing tool, the MSP can be programmed from a Computer. By design a  $\mu$ C only works with binary values. As programming in binary code is not practical, first assembler was developed. Over time compilers were developed allowing to compile higher programming languages as C to assembler. For the MSP TI provides a special compiler enables various languages as C or Arduino to work with the MSP.

#### 2.4.5 Digital Communication Protocols

A  $\mu$ C is usually used as the coordinating device and controls the peripheral modules. In case of the pAM, these are an ADC, a temperature sensor and the XBee modules. To standardize the

communication, several protocols were developed over time. Such protocols usually connect one coordinating device, the master, to multiple other devices, the slaves.

### Inter-Integrated Circuit Bus

The inter-integrated Circuit ( $I^2C$ ) Bus was developed by Philips Semiconductors; the specifications can be found in [39]. The  $I^2C$  only needs two wires, one for the serial clock (**SCL**) and one for data (**SDA**) to allow bidirectional communication. The master generates the **SCL** signal for all slaves.

The logic levels for an  $I^2C$  bus are not fixed but defined by the supply voltage  $V_{DD}$ . The lower level (LOW) is defined as  $< 0.3V_{DD}$  and the high level (HIGH) as  $> 0.7V_{DD}$ . The bus lines have to be normally HIGH, which can be assured by the use of pull-up resistors.

The transfer of data is dependent on the **SCL** signal. Changes on the **SDA** are only valid for data transfer when they occur during a LOW on **SCL**. A data transfer is started with a START condition and finished with a STOP condition. START is defined as a HIGH to LOW transition on **SDA** with **SCL** staying HIGH. Accordingly STOP is a LOW to HIGH **SDA** transition. Between START and STOP an arbitrary number of bytes can be transferred, but each byte has to be followed with an acknowledge. For an acknowledge the transmitter has to release the **SDA** line after the eighth clock pulse, the receiver has to pull it LOW and keep it LOW during the ninth clock pulse. If no acknowledge is received, a valid data transmission can not be ensured.

Selection of a Slave device for reading or writing is done by transmitting the address of the device; followed by a  $R/\overline{W}$ -bit to indicate the direction of the communication.

### Serial Peripheral Interface

The Serial Peripheral Interface (SPI) was developed by Motorola in the year 2000; the specifications can be found in [40]. SPI, in contrary to the  $I^2C$  bus, uses 3 wires. One for the serial clock (**SCL**), one for serial data in (**SDI**) and one serial data out (**SDO**). In some configurations, there is an additional select line per slave. SPI allows parallel read and write of data and in contrast to  $I^2C$  does not use any start/stop conditions or acknowledges. The clock polarity of an SPI is configurable. One clock state allows changing the **SDO** and **SDI** values; on the other state, the values are valid and can be read.

## 2.5 XBee Operation

In the following chapters, the XBee radio module plays an important role, as it is used for communication with the pAM. The basics of operation will be explained here, for more detailed information refer to [41]. Communication with the XBee modules is done via a Universal Synchronous and Asynchronous Serial Receiver and Transmitter (USART) interface with a 3.3 V logic. The pinout of an XBee module is shown in figure 2.16. For the basic operation, the **VCC** is connected to 3.3 V. **DOUT** and **DIN** are used for communication and hooked up to the host device. The **SLEEP\_RQ** is connected to the host as well and used to toggle sleep mode. **GND** is connected to ground. All the other pins are left unconnected.

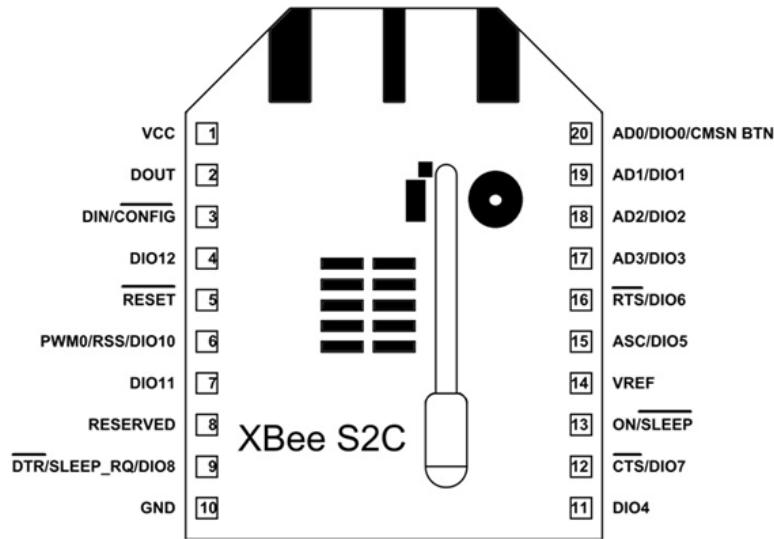


Figure 2.16: Pinout of the XBee radio modules. From [42].

### 2.5.1 Mode of Operation

The XBee modules have two modes of operation. First, the transparent mode, where a module acts as a piece of transmission line. All data is directly streamed from one device to the other. The Second mode is the Application Programming Interface (API) mode, which arranges communication in frames. The API mode comes with some benefits, including acknowledges for send packages, signal strength indicators, source addresses, and the possibility to change the module configuration. For operation with a pAM, the XBee modules are operated in API mode.

### 2.5.2 API Frame Structure

The API frames can be sorted in three general categories:

- Command frames for programming the module,
- RX frames, send over the USART when data is received,
- TX frames, received over the USART with data to be send,

The modules used in the pAM are programmed from the beginning<sup>4</sup>. For operation in a pAM, only the TX and RX frames are of interest. Both frames exist in 2 variations, one with 64-bits addressing and one with 16-bit addressing. The pAMs are currently only using 16-bit addresses. The frames for receiving and transmitting are shown in figure 2.17.

The checksum for all frames is calculated by adding up all bytes of the message, excluding frame delimiters and length. The eight lowest-bits of the result are subtracted from 255 (0xFF).

---

<sup>4</sup>This is done with the XCTU firmware provided by Digi [43].

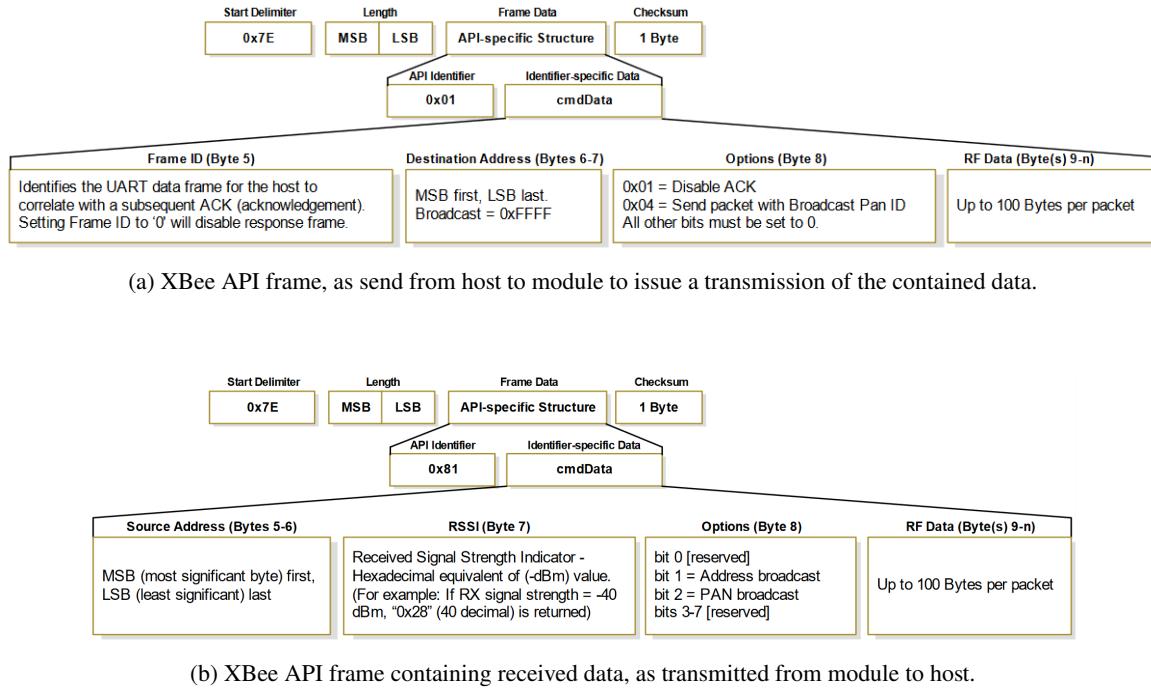


Figure 2.17: Important XBee API frames. From [41].

### 2.5.3 Power Consumption and Sleep Mode

An active XBee module draws around 40 mA of current, making it unsuitable for low power applications. To reduce the average power consumption, the modules come with sleep modes. To put a module in sleep mode, the **SLEEP** pin has to be pulled high. In sleep mode the module draws less than 50  $\mu$ A, but can not receive or transmit data. Waking up from sleep mode takes 13.2 ms.



# CHAPTER 3

---

## Initial Status of the Picoamperemeters

---

The pAM currently in use were designed and built at the TU München. The design was revised multiple times both revision three and four<sup>1</sup> are in use. The main focus in the following chapters is on the revision three devices, as revision four does not deviate significantly. The schematics and PCB layout for the design can be found in appendix A.1. Before conducting an analysis of the devices and the affiliated infrastructure an introduction on the overall setup is necessary, and therefore provided here.

### 3.1 The Backend Setup

The control unit of the pAM is the MSP430f169 (MSP) µC, which is a low power device produced by TI<sup>2</sup>. It is highly flexible and offers pins with internal compatibility for common protocols as I<sup>2</sup>C and SPI. The MSP comes with 6 Ports, 8 I/O pins each, ports 1 and 2 have full interrupt capability. For programming of the MSP430 a JTAG interface is implemented. The firmware running on the MSP coordinating the measurements is described in section 3.5.1. The µC is driven by a single 3.3 V supply, fixing the logic levels accordingly.

For communication with an external computer while in operation the devices are equipped with an XBee wireless module; which is operated via the USART in- and output ports of the µC. The operation of the XBee modules is described in 2.5.

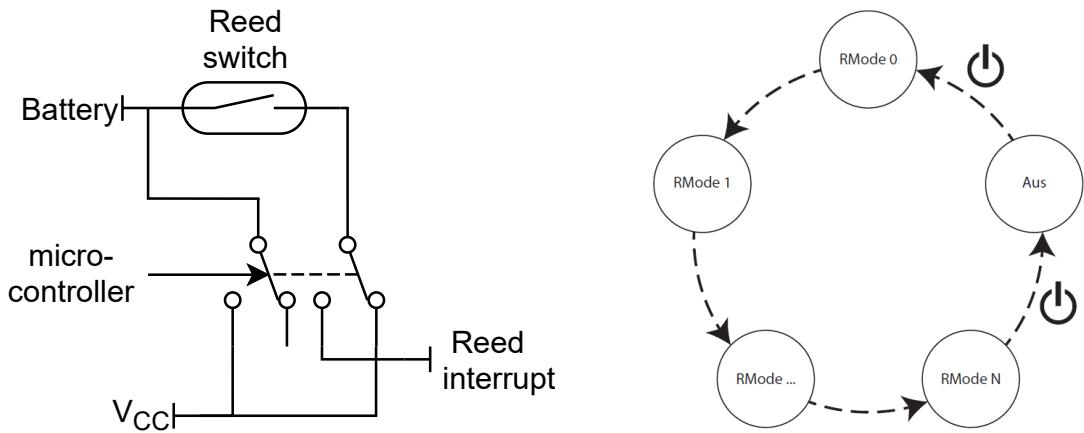
#### 3.1.1 Mode Control

The pAM come with multiple modes of sensitivity, depending on the revision. Mode switching and the turn on/off are combined in one circuit as depicted in figure 3.1(a). Cycling the modes is handled in the Reed interrupt routine, according to figure 3.1(b).

---

<sup>1</sup>Revision 4 was developed by [3].

<sup>2</sup>For details refer to the manual [38].



(a) Mode switching and reed interrupt trigger for the pAM revisions 3 and 4.

(b) Reed interrupt cycling through the different modes and turning on and off of the device. From [4].

Figure 3.1: Implementation of mode switching in the pAM revision 3 and 4.

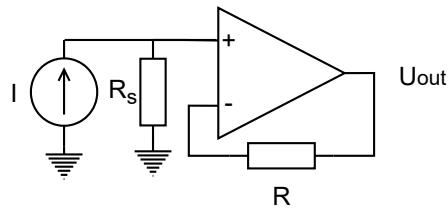


Figure 3.2: Current measurement circuit using an operational amplifier and a shunt resistor.

## 3.2 The Front-end Setup

The pAM measure current indirectly by determining the voltage drop over a shunt resistor, following Ohm's law. This is done by the use of an Op-Amp in a simple amplifier configuration, as depicted in figure 3.2. Using Ohm's law and the golden rules of operational amplifiers, see section 2.2.1, the simple relation in equation 3.1 can be derived.

$$U_{out} = R_s I \quad (3.1)$$

This first amplification stage uses an AD549LHZ Op-Amp which features a high input impedance, low input current and offset voltage adjustment [44]. According to its datasheet, it is designed to be used for electrometer amplifiers or photodiode interfaces. Driving the amplifier with a bipolar supply allows measurement of both positive and negative currents. For matching the bipolar voltage signal from the input stage with the unipolar ADC input, a second amplifier is introduced as depicted in figure 3.3, expanding equation 3.1 to:

$$U_{ADC} = I R_s + U_{ref} \quad (3.2)$$

Where  $U_{ref} = 2.048$  V. As a second amplifier, the  $\mu$ A741 is used. It is a general-purpose amplifier, designed for voltage amplification. For digitisation, the design uses the AD7790 buffered  $\Sigma\Delta$  ADC

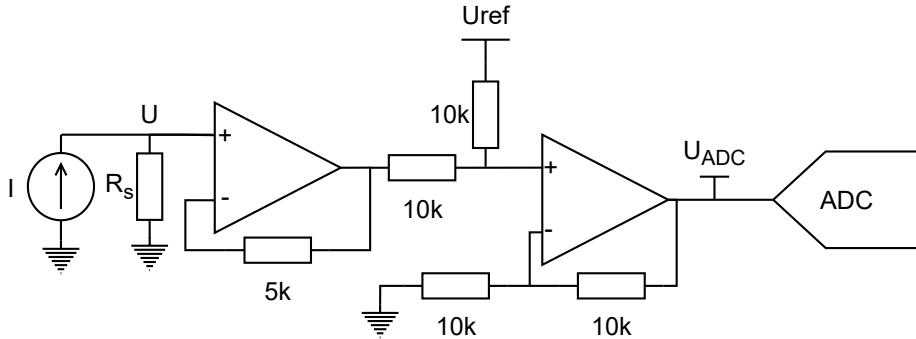


Figure 3.3: Front-end circuitry used in the revisions 3 and 4. The additional OVP diodes are depicted in figure 3.4(b).

which features a suitable low noise for the application and has a sampling frequency of 9.5 to 120 Hz [45]. Taking into account the digitisation range of the ADC and the highest shunt resistance of  $100\text{ M}\Omega$  the setup could reach a theoretical precision of 0.625 pA.

### Over-Voltage Protection

The pAMs are equipped with protection diodes as depicted in figure 3.4(a), to prevent damage to the amplifier from over-voltage conditions. In figure 3.4(b) an extension to the input stage is depicted, which is soldered floating over the board. It consists of a  $20\text{ k}\Omega$  resistor in parallel to a SA5.0CA bipolar TVS diode<sup>3</sup>. The resistor forms a voltage divider with the  $5\text{ k}\Omega$  resistor, adding an extra factor to the transfer function. The transfer-function in equation 3.2 is expanded to:

$$U_{\text{ADC}} = 1.25 \cdot IR_S + U_{\text{ref}} \quad (3.3)$$

The influence of the TVS diode will be discussed in section 4.2.

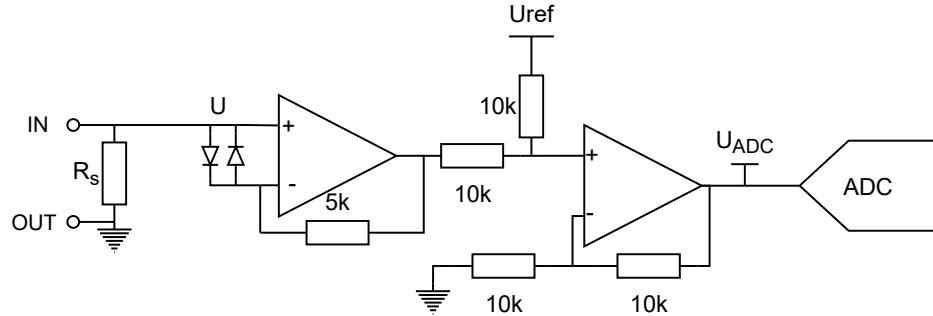
### Selectable Modes

As introduced before the pAMs have a selectable resolution and measurement range. This is achieved by adding a second smaller resistor  $R_{\parallel}$  in parallel to the  $100\text{ M}\Omega$  resistor; this reduces the shunt resistance to:

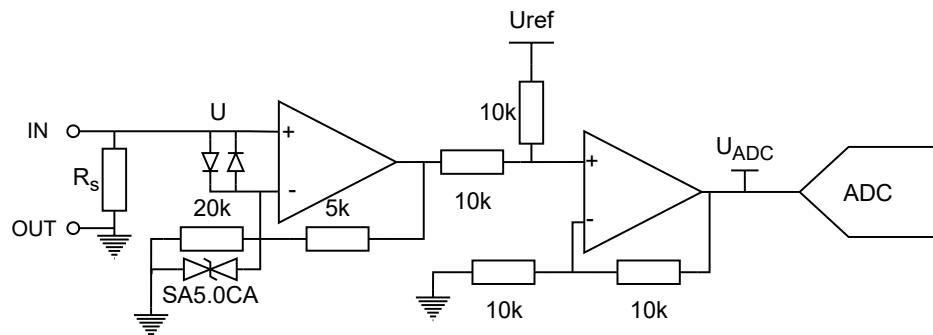
$$R_S = \frac{1}{\frac{1}{R_{\parallel}} + \frac{1}{100\text{ M}\Omega}} \quad (3.4)$$

Modes for the different revisions are shown in table 3.1. The low-resolution modes also use a varistor parallel to the shunt resistor as additional over-voltage protection.

<sup>3</sup>It is unknown why this circuitry was attached or by whom. It was already mentioned in [3] in 2013.



(a) Front-end equipped with over voltage protection diodes between the inputs of the first amplifier.



(b) The same front-end with additional 20 kΩ resistor and parallel SA5.0CA TVS-diode.

Figure 3.4: The two expansions that are made to the front-end depicted in figure 3.3.

mode	shunt resistance	range	resolution	raw mode	
				revision 3	revision 4
1	$100\Omega  100M\Omega$	$\pm 16.4\text{ mA}$	625 nA	0	
2	$10k\Omega  100M\Omega$	$\pm 164\mu\text{A}$	6.25 nA	1	0
3	$100k\Omega  100M\Omega$	$\pm 16.4\mu\text{A}$	625 pA		1
4	$1M\Omega  100M\Omega$	$\pm 1.64\mu\text{A}$	6.25 pA	2	2
5	$10M\Omega  100M\Omega$	$\pm 164\text{ nA}$	0.625 pA		3
6	$100M\Omega$	$\pm 16.4\text{ nA}$	0.5 pA	3	4

Table 3.1: Modes of the different pAM, the range is given by  $\frac{2.048\text{ V}}{1.25R_s}$ . This sets the resolution to range divided by  $2^{15}$ . Translated and modified from [5].

---

Average	current draw 1/mA
without XBee	(1.773±0.012)
with the Xbee sending	(48.1±0.6)

Table 3.2: Current draw of a pAM. From [5], modified.

### 3.3 High-Voltage Isolation

The pAMs are designed for usage on the high-voltage supply lines of a GEM-foil. This requires some precautions to protect a user. To ensure isolation, the pAMs are enclosed in a plastic box. The only openings in this box are the terminals for the signal lines. The terminals are SHV connectors, the outer conductor is connected to earth ground, only the inner conductor is on a high-voltage. As a power supply the devices use 9 V-blocks, as these allow operation without any ground connection. In addition, a photovoltaic powering was developed, to achieve longer run times, see [5] for details.

### 3.4 Design Performance

#### 3.4.1 Power Consumption

The power consumption characteristics were studied by [5] and are only briefly summarized here. Originally the devices are powered by 9 V-block batteries, hence low power consumption is a design goal. Table 3.2 summarizes the current draw for an operating device.

#### 3.4.2 Measurement Results

The measurement performance of the device was analysed by [4]. It was found that the devices show a characteristic non-linear behaviour with deviations in the range of some ten picoampere (pA) in the most sensitive mode. This non-linearity scales linear with temperature, which was also shown by [4]. The origin of the effect is discussed in section 4.2.

Calibration measurements showed that the error on the pAM reading, the standard deviation of 128 measurements, is around 1.5 to 2 ADC channels, resulting in an uncertainty of around 1 pA. Showing that the design is capable of precise measurements on the pA scale.

The pAM take 8 s to finish a complete sample of 128 ADC readings. This is mainly due to the low ADC sampling rate that is used. This low sample rate results in a bad time resolution.

### 3.5 Infrastructure

To operate the pAM the firmware for the µC is needed, as well as the PicoAmpereMeter Operating System (PAMOS) software; which is responsible for receiving the data, applying the calibration and saving the results. To obtain a reliable calibration for the pAMs, a calibration station was built by [4]. A short introduction to the software is given here.

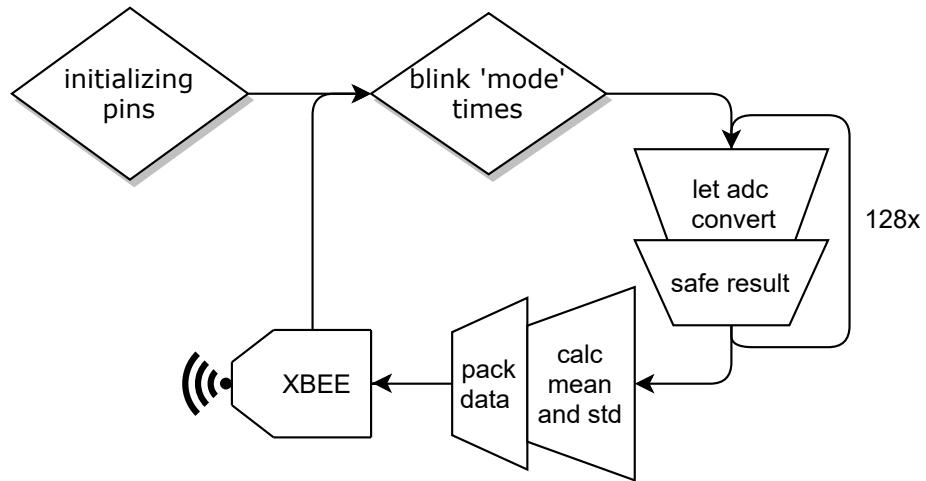


Figure 3.5: pAM firmware workflow for firmware versions 3.x.

### **3.5.1 Firmware**

With the beginning of this thesis, the MSP firmware is present in versions 3 and 4 for the respective revisions of the design. It is written in C and compiled via the opensource msp430-gcc compiler in version 3.2.3; which is only running on Ubuntu 10.04.4 LTS, according to [46]. The rather complicated flashing process is described there as well. The basic workflow of the firmware is depicted in figure 3.5.

The firmware comes in different builds featuring different functionalities. The firmware versions x.0 are the basic versions, the versions x.1 additionally send out all 128 ADC raw values. All firmware versions also have a second build, that sets the *CALIB* flag, which adds auto mode switching to the firmware.

To simplify coding and increase readability of the code, a lot of alternative symbols are defined in the header files *iostructures.h* and *pAM-definitions.h*. The ports and pins were assigned an alias according to their purpose. For example, the ADC communication is done with port 3, pins 0 to 3, hence the aliases are:

```
#define ADC_PORT port3
#define ADC_SCLK pin3 /* P3.3 SCLK */
#define ADC_DOUT pin2 /* P3.2 SOMI */
#define ADC_DIN pin1 /* P3.1 SIMO */
#define ADC_CNV pin0 /* P3.0 */
```

## Initialisation

On start-up of the microcontroller, some basic settings have to be made. At first, the clock and timer modules have to be set up. The device is equipped with a 32 kHz crystal on **XIN** and a 1.8 MHz crystal on **XT2**. As main clock source in the pAMs, the 1.8 MHz clock is used.

The next step is to set up the pins according to their purpose. Each port has several registers defining their usage, see table 3.3. This happens with the function `port_setup()`, the registers for the different

register	register purpose	ports			states	
		1	2	3 - 6	1	0
dir	select if pin is input or output	✓	✓	✓	output	input
in	state applied to pin	✓	✓	✓	set externally	
out	state applied to pin	✓	✓	✓	set output high	set output low
sel	function selection	✓	✓	✓	peripheral module	I/O functionality
ie	interrupt enable	✓	✓		interrupt enabled	interrupt disabled
ies	interrupt edge select	✓	✓		falling edge	rising edge
ifg	interrupt flag	✓	✓		interrupt pending	no interrupt pending

Table 3.3: Registers, their purpose and ports that feature them. Only ports 1 and 2 have interrupt capability [38].

ports are summarized in table 3.3. The USART module is set up to interface the XBee, as depicted in listing 3.1. The modules setup allows two-way communication. However, the communication protocols for this are not implemented.

Listing 3.1: Setup of the USART ports for communication with the XBee . Comments preceded with //< were added for explanation.

```
void usart_setup() {
    /*TX*/
    //<TX pin for transmitting data:
    port3.dir.pin4 = IO_DIRPIN_OUTPUT; //<=1
    port3.sel.pin4 = IO_ALTPIN_SELECT;//<=1
    /*RX*/
    //<RX pin for receiving data:
    port3.dir.pin5 = IO_DIRPIN_INPUT; //<=0
    port3.sel.pin5 = IO_ALTPIN_SELECT;//<=1

    ME1 |= UTXE0 + URXE0; // USART0 TX, RX enable
    //<set protocol to 8 data-bits, 1 stop-bit, no parity (8N1):
    U0CTL |= CHAR;      // 8 data-bits, 1 stop-bit, no parity (8N1)
    //< set clock source to MCLK (1.8MHz):
    U0TCTL |= SSEL_2;   // UCLK <- SMCLK
    /* clock: 1843200Hz
    desired baud rate: 115200bps
    division factor: 16
    effective baud rate: 115200bps
    maximum error: 0us 0.00%
    */
    //< set divider to 16, modulation to 0:
    U0BR0=0x10; U0BR1=0x00; U0MCTL=0x00;
    //<open USART for operation:
    U0CTL &= ~SWRST;      /*USART freigeben*/
}
```

After these setups, the mainloop is started. First the ADC is reset, and the LED blinks to signal operation. After this the ADC is setup for reading data. The firmware reads 128 values from the ADC, entering a low power mode in between conversions. The ADC **RDY** pin is used as an interrupt to

mean	error	mode	voltage	...
...	conversion time	time until sending	0	

Figure 3.6: Datasets as sent out by pAMs running the firmware v3.0 and v4.0. The package contains mean and error of the acquired data, the current operating mode, the SVS voltage output, the time needed for conversion, the time between finishing a conversion and sending data, and the number of remaining packages; which is 0 in this case.

mean	error	mode	voltage	...
...	conversion time	time until sending	0	
ADC value	ADC value	...	ADC value	3
ADC value	ADC value	...	ADC value	2
ADC value	ADC value	...	ADC value	1
ADC value	ADC value	...	ADC value	0

Figure 3.7: Datasets as sent out by pAM running the firmware v3.1 and v4.1. The first package contains mean and error of the acquired data, the current operating mode, the SVS voltage output, the time needed for conversion, the time between finishing a conversion and sending data, and the number remaining packages. The next four packages contain 32 ADC readings each, followed by the number of remaining packages.

enter the corresponding ISR, which handles data reading.

After data is read-out, mean and standard deviation of the 128 values is calculated. This averaging has the advantage of reducing the influence of noise on the data and allows more precise measurements. However, it elongates the readout time. The derived values get sent out along with other parameters. The data frame for the firmware revisions x.0 is depicted in figure 3.6; the versions x.1 packages are shown in figure 3.7. Afterwards, the readout is finished, and another cycle starts.

### 3.5.2 PAMOS

The PicoAmpereMeter OperatingSystem is the software used to receive and handle data, send out by the pAMs. It is written in C++ to be run on Linux machines. With the beginning of this work, it was available in version 2. The software reads data, applies the calibration and stores data in an online database or locally as a root file. Discrimination between the firmware versions described above is done by the message length.

### 3.5.3 CalibAna

The CalibAna software was developed by [4]. A detailed description of its principle can be found there. It controls the calibration setup by communicating with, an Arduino to adjust the temperature, the Keithley 6517b electrometer (Keithley), which used as a reference for calibration and reads the pAM data. The output can be stored in a calibration database and be used to create a plot with the calibration data, using a gnuplot script. An example plot is shown on figure 3.8. The top left plot

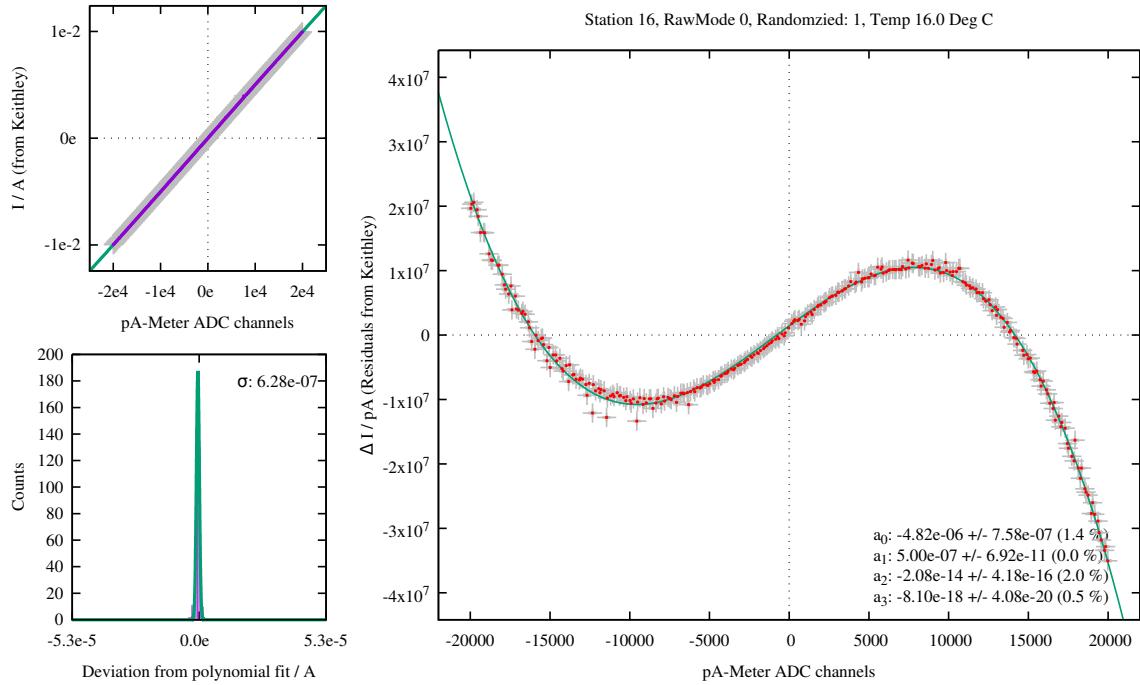


Figure 3.8: Calibration plot produced by CalibAna2.

shows current measured by the Keithley vs ADC readings and a linear fit of the data. The central plot contains the deviation of current values from the linear fit applied to the ADC channel range. These deviations are fitted with a third-order polynomial. The bottom left plot shows the deviations from the polynomial fit as a histogram. A gaussian fitted to the histogram is used to approximate the resolution of the pAM.

## 3.6 Motivation

Now, having provided a look into the device, why is a revision of the design necessary? There are several shortcomings in operation and handling of the design that were discovered over time. As described above, the firmware is not up to date and is not compatible with modern compilers. Hence upgrading the firmware is one of the bullet points. This process can also be used to add new features to the firmware to make operation of the device more flexible.

A problem that occurs during operation is a breakdown of the Op-Amp in the input stage, probably due to over-voltage conditions on the inputs. On the one hand, replacement of an amplifier is complex and rather costly with  $\approx 60$  € per amplifier. On the other hand, the running measurement is interrupted and needs to be repeated.

One of the biggest issues, in the end, is non-linear current to ADC channels relation mentioned above in combination with a temperature dependence, making a complex and time-consuming calibration necessary.



# CHAPTER 4

---

## Addressing the Issues

---

In chapter 3 an introduction on open topics was given. An update of the firmware, investigations on the cause of the non-linear behaviour and development of a new OVP solution are necessary. These three topics will be addressed in this chapter.

### 4.1 Firmware Update

As previously stated, the firmware for the pAM is not compatible with the compiler provided and maintained by TI. One of the goals in the scope of this work was to adapt the firmware to achieve compatibility. As this progress requires some significant changes in the code, it was used to develop new features and restructure the program, in parallel.

#### 4.1.1 Code Syntax

To be compatible with the compilers provided by TI some changes to the syntax of the code were necessary.

##### Pin addressing

One of the biggest changes that had to be made was the way the pins and ports of the µC were addressed. In the previous firmware versions, the pins and ports were accessible similarly to objects, via the "." operator. The new compiler fills registers from 8-bit numbers. As an example, the direction of pin 2 of port 1 should be set to 1, allowing usage as an output pin. On the Left, the way it was done with the previous compiler; On the right, the way the new compiler requires:

```
port1.dir.pin2 = 1;           port1_dir = 0b00000100;
```

Register content can be changed in two ways. Either assigning a value to the register, however, this would overwrite the status of all other pins. This can be circumvented by using a-bitwise operation:

```
port1_dir |= 0b00000100;
```

The 8-bits are individually or'ed with the 8-bits of 0b00000100, changing the value of the second-bit without influencing the other-bits. The inverse operation to this is-bitwise *not and*:

```
port1_dir &= 0b00000100;
```

To further increase readability, the-bitwise operations were assigned aliases. A small example:

```
#define set_high |=  
#define set_low &= ~
```

Correspondingly the binary numbers that represent each pin are assigned with the pin name:

```
#define pin0 0b00000000  
#define pin1 0b00000010
```

The initial example hence is now written as:

```
port1_dir set_high pin2;
```

### 4.1.2 File Structure

To ease handling of the code, it was split up in different source files and their headers. All functions concerning the XBee for example, are gathered in *xbee.c* with the corresponding *xbee.h*. These files are shared between the different versions of the firmware, allowing introduction of new features on all of them.

### 4.1.3 Bidirectional Communication

As mentioned before there, are possible options that change behaviour and output of the pAM. The options are set together with the receiver address in the flashing process. The devices can be made more flexible by allowing wireless two-way communication between a pAM and the receiving computer. The hardware setup of all generations of pAMs already supports this. On the software side, functions handling the reception of data are needed. The parameters need to be set before the measurement is started. Therefore, the firmware was modified to enter an idle state on start-up, waiting for a handshake with the external setup. To indicate the device is waiting for setup, the indicator LED is turned on. The *set\_up* flag is introduced to indicate that the device has been set up successfully. As the USART module of the MSP430 has interrupt capability, the corresponding ISR will be used to handle incoming messages.

An XBee module will immediately send out data it received in an API frame, see section 2.5. The µC will read the first character, store it in the USART receive buffer (U0RXBUF) and set the USART receive interrupt flag (URXIFG0). This generates an interrupt request, causing the program to jump to the USART interrupt routine.

The ISR at first checks if the device is already setup. Next step is to disable the USART interrupt and hold the Watchdog timer. The received data is read using the *rx\_request()* function. This function will read the message characters into a buffer array by polling on URXIFG0. To simplify the message handling on the microcontroller site, an upper limit *length* was set on the message length, and the char *0xFF* was reserved to indicate the end of the message. The polling is stopped when either *length* bytes were read or the second to last byte was the end of message indicator. From the message array, the parameters are derived by scanning for two indicators. The first indicates that the next two bytes contain the address of the receiving station. The second indicates that the following bytes contain the runtime parameters. Currently, there are two parameters for the revisions three and four to be set.

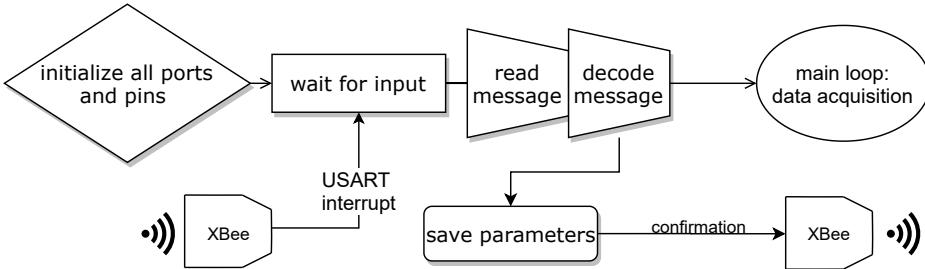


Figure 4.1: The initialisation process for pAM, as implemented in the firmware versions 3.2 and 5.0.

The sensitivity mode to operate in and if the ADC raw values are to be sent in addition. The list of parameters can be extended in the future.

To ensure correct setup, the pAM answers with a frame containing all the set flags. The USART interrupt is disabled after this handshake, to ensure stable operation. It was tried to allow changing the parameters during runtime by allowing further interrupts. This led to spurious interrupts, disturbing the device operation.

A flow-diagram showing this initialisation loop can be found in figure 4.1.

Only after this handshake of the pAM with the external setup, readout is possible, as introduced in section 3.5.1.

The changes described above are combined in the firmware version 3.2. To ensure compatibility with the new firmware, the PAMOS and CalibAna (CalibAna) software were modified. The necessary modifications are discussed in the next section.

#### 4.1.4 Changes to PAMOS

The PAMOS software is handling the data acquisition of the pAM on the computer side. It is the main software used to communicate with the devices. It needs to be adapted to support the two-way communication implemented in the firmware. A first approach was to use one of the various open-source libraries available for the XBee modules. The *libxbee3* library [47] was chosen. The approach was discarded after the library proved to be too slow to keep up with the revision 5 of the pAM, developed in chapter 5.

Therefore, a custom solution was developed based on the *termios* library [48], which was already used for data reading in the previous version of PAMOS. The *termios* library provides a set of functions that can be used to control an asynchronous communication port of a computer, e.g. the USB port. Reading and writing to the port is handled as it is done for regular files. The message handling on the computer side was made more complex, including different error-handling methods. This is possible, as a computer is noticeably faster than the MSP used in the pAMs.

An introduction on the functions created around this library to manage the pAM communication will be given here. To ease handling, all pAM related variables and functions are gathered in the *pAMeter* class:

Listing 4.1: pAMeter structure in the PAMOS software. The structure contains all parameters that are necessary to allow flawless communication.

```

class pAMeter {
public:
    int fd; //!< file descriptor pointing to a file managed by a termios structure.
            Used to read and write from usb port
    unsigned int station; //!< The stations 16-bit address, filled from startup
                          argument -s
    unsigned int receiver; //!< The 16-bit address of the receiver used, filled
                          from startup argument -r
    unsigned char mode; //!< The mode the pAMeter should use, filled from startup
                        argument -m
    unsigned char dataFormat; //!< Determines what data to be transmitted, filled
                            from startup argument -F
    unsigned char REVISION; /*!< Revision of the pAMeter in use, transmitted by the
                           pAM on setup. Can be mapped to a readable value with calc_mode()*/
    unsigned int frame_id = 0; //!< frame id for counting frames
    unsigned char delay; //!< Rev 5 only, determines the number of wait cycles put
                         between two readings
    int debug=1;
    struct pAMmessage msg; //!< used to store a single frame transmitted by a
                           pAMeter
    std::map<unsigned short, int> revision; //!
    std::map<unsigned short, calibration_t> cal; //! contains the calibration for
                                                 the station
    int initiate(); //!< organizes the setup of a pAM with mode, dataFormat etc.
                    and checks if the setup was successfull
    int setup_interface(); //!< sets up all termios attributes and applies them to
                          the file behind fd
    int send_init(); //!< sends the initialisation frame and checks for a confirm
    int TX(unsigned char buffer, unsigned int length); //!< transmits buffer with
                                                     size length with the defined api framework
    int RX(); //!< used to receive a frame and store it in a pAMmessage, verifies
              checksum
    std::tuple<unsigned char, unsigned int> message_handler(); //!< identifies a
                                                               received message and fills the pAMmessage
};

```

## Initialisation

The port is handled alike a file, as stated before. Therefore, the first step is to open the port and link it to the file descriptor *fd*:

```
pAM->fd = open( "/dev/ttyUSB0" , O_RDWR | O_NOCTTY);
```

The *termios* library associates a terminal to the file that handles the communication. This terminal has to be set up with the communication parameters, e.g. baud rate, parity, word length. This is done by the *setup\_interface()* function. This allows flawless communication with the XBee. The initialisation of the pAM is done in the *initiate()* function. The first step is to send out the initialisation frame for

the pAM, see section 4.1.3. This is done by the `TX()` function, shown in listing 4.2. It fills all required frame data in the `buffer` array and calculates the checksum. Afterwards, the array is written to the USB port, but it is only transmitted if the data is followed by the carriage return character '\r'. The sending is repeated until an acknowledge from the XBee is received. If a frame is received, that is no acknowledge, the program prints an error message and holds, as a successful setup can not be guaranteed. When an acknowledge is received, the program awaits the confirmation frame from the pAM. If one of the parameters in the confirmation frame does not match, the program returns an error message and stops.

Listing 4.2: C++ (C++) function for transmitting data with an XBee module and checking if an acknowledge was received.

```
int pAMeter:: TX(unsigned char* buf, unsigned int length)
{
    tuple<unsigned char, unsigned int> handle;
    unsigned char buffer[1 + 2 + 1 + 1 + 2 + 1 + length + 1];
    buffer[0] = 0x7e;
    buffer[1] = ((length + 5) >> 8);
    buffer[2] = ((length + 5) & 0x00ff);
    buffer[3] = 0x01;
    buffer[4] = 1; //frame_id++;
    buffer[5] = (station >> 8);
    buffer[6] = (station & 0x00ff);
    buffer[7] = 0x00;
    for (unsigned int i = 0; i < length; i++) {
        buffer[i + 8] = buf[i];
    }
    buffer[8 + length] = 0;
    buffer[8 + length] = checksum(buffer, sizeof(buffer));
    do { //< send until acknowledge is received
        sleep(0.3);
        write(fd, buffer, sizeof(buffer));
        write(fd, "\r", sizeof("\r"));
        sleep(0.5);
        if (RX() == 1) {
            handle = message_handler(); //< analyzes the received frame
            if ((get<0>(handle) == 0x89) && (get<1>(handle) == 0)) {
                //< is acknowledge
                if(debug) cout << "send init success" << endl;
                return 1;
            } else {
                printer("TX failed", " ");
                return 0;
            }
        }
    } while (1);
    return 1;
}
```

## Receiving Data

The `RX()` function, see listing 4.3, reads one char from the port. If the char is the frame delimiter from an XBee API frame, the function will try to read the message, else it will return a *nothing received*. The next two chars in the buffer should be message length, see section 2.5. The message length is extracted from these, and a message of this length is read. The message is validated by calculating its checksum. If the checksum does not hold, *invalid message* is returned. The full message is stored in the message buffer to be further analysed by the `message_handler()` function.

Listing 4.3: C++ function, that reads data from the USB port.

```
int pAMeter::RX()
{
    unsigned char del = 0;
    read(fd, &del, 1);
    if (del == 0x7e) {
        char len_buf[2] = { 0 };
        read(fd, &len_buf, 2);
        msg.length = ((len_buf[0] << 8) + (len_buf[1]));
        if(debug) cout << "received masse w length : " << msg.length << endl;
        if ((msg.length < 0) | (msg.length > 255)) {
            printer("message length out of boundaries");
            cout << msg.length << endl;
            return 0;
        }
        read(fd, &msg.buffer, msg.length);
        unsigned char checksum_buf[1];
        read(fd, &checksum_buf, 1);
        msg.checksum = checksum_buf[0];
        unsigned char temp = 0;
        for (unsigned int i = 0; i < msg.length; i++) {
            if(debug) cout << " " << (int)msg.buffer[i];
            temp += msg.buffer[i];
        }
        if(debug) cout << endl;
        if (msg.checksum + temp != 255) {
            printer("checksum not valid");
            return 0;
        }
        return 1;
    } else {
        if(debug) cout << "not received anything" << endl;
        return -1;
    }
}
```

The `message_handler()` function identifies the type of API frame and extracts the contained data.

### 4.1.5 Changes to CalibAna

For the CalibAna software, similar functions were implemented to control the pAM. However, the

increased readout frequency of the new revision, developed in chapter 5, caused problems. The Keithley, needs some time to perform a precise measurement<sup>1</sup>. To ensure this, the data acquisition is artificially lengthened by adding a delay of one second. The fast pAM will perform multiple measurements during this time, leading to a pile-up. Reading from the port read the next available message, not the last received one. A single read command per voltage step hence will result in outdated data. To work around this problem, the reading process was moved to a separate thread running the *continuousread()* function, see listing 4.4. The function calls *RX()* until a frame is received. The *message\_handler()* function checks if the received message is a receive frame, which sets the *valid* flag. When the most recent frame is to be saved for calibration, the main program sets the *pause* flag, holding the readout when the next valid data-frame is available.

Listing 4.4: C++ function, that continuously tries to read from the USB port and on success stores the message.

```
void parameterneu::continuousread()
{
    while (!stop) {
        valid = 0;
        tuple<unsigned char, unsigned int> handle;
        sleep(0.2);
        int status;
        do {
            status = RX();
        } while (!(status == 1));
        handle = message_handler();
        if (!((get<0>(handle) == 0x81) && (get<1>(handle) == 1))) {
            valid = 0;
        } else {
            valid = 1;
        }
        while (valid && pause) {
            cout << "waiting for data read" << endl;
        }
    }
}
```

## CalibAna Data Analysis

To simplify data analysis and allow more flexibility, a new plot program was developed using *python*, to replace the gnuplot script developed by [4]. Python was chosen as it is more versatile than gnuplot and more commonly known. The plot program was developed to be capable of producing calibration data for the revision 3 and 4 pAM as well as the revision 5 devices. The calibration data output of CalibAna was modified, adding a header to the file. The header contains calibration parameters like mode, number of measurement runs, station and revision. The python script reads data from this header and uses it to fit calibration functions to the data and produce plots according to the revision and range.

---

<sup>1</sup>The influence of measurement time on the precision is briefly discussed in section 5.4

## 4.2 Non-Linear Measurements

As introduced above the devices show a characteristic non-linearity; the origin of this is explored here. The non-linearity can be found on all modes of sensitivity, see figure 4.2. Hence the effect seems to be independent of the measured current and probably occurs in the voltage signal shaping of the front-end. For investigation, the front-end was simulated using LTspice. The effect could not be reproduced quantitatively, as a model for the SA5.0CA Diode was missing. However using a model for a diode of a similar type, it could be shown, that the diode adds a non-linear characteristic to the design. The non-linearity is caused by the diode clipping the feedback voltage, which propagates to the output voltage. Measurements, done with all diodes removed, indeed showed no such effects, see figure 4.3. Removing diodes from the front-end leaves the pAM without over-voltage protection. A new solution has to be found; the next sections will treat this topic.

## 4.3 Over-Voltage Protection

The pAMs are operated at high voltages. Fast ramping or higher than expected currents, e.g. discharges in the GEMs attached, can lead to large voltage differences between the input terminals. Such over-voltage conditions can result in malfunction of the input stage amplifier. To prevent such events, diode clamping is a commonly used technique. As introduced in section 3.2, the devices come with two OVP structures. The first one consists of two antiparallel diodes between the Op-Amp inputs, see figure 3.4(a). These diodes are no appropriate OVP, as they connect the non-inverting amplifier input, with the inverting input. As introduced in section 2.2.1, an ideal Op-Amp adjusts the output to have matching inputs. Applying large differential voltages between the inputs can lead to the destruction of an amplifier. In this setup, however, the inverting input is only defined by the output voltage and the feedback loop. An over-voltage condition between the Op-Amp inputs is highly improbable.

A similar argument is valid for the SA5.0CA diode that was added to the feedback loop, see figure 3.4(b). It limits the feedback loop voltage but has no effect on the non-inverting input.

Expected over-voltage conditions occur between the input and output of the pAM. The input terminal is connected to the non-inverting Op-Amp input; the output is connected to the device ground-plane. An over-voltage between the terminals can force the input common-mode voltage to be outside of the amplifier supply voltage range, which can damage the amplifier. The input trace should, therefore, be clamped. Three methods of over-voltage protection, using diode clamping, are discussed in the next section.

## 4.4 Alternative Over-Voltage Protection Circuits

There are different approaches to limit the voltage range of a trace that employ diodes. Three possibilities are discussed here, to provide an insight into the possible errors that can occur. Version 1 uses two antiparallel diodes from the protected trace to ground, see figure 4.4. Either diode becomes conductive when the input voltage exceeds its forward voltage. A second possible setup, depicted in figure 4.4(b), is to put the diodes back-to-back or front-to-front, extending the input range to the reverse breakdown voltage of the diodes. A third solution would be to clamp to an external potential, as in figure 4.4(c). All three solutions were simulated in LTspice to test their behaviour. For this a full

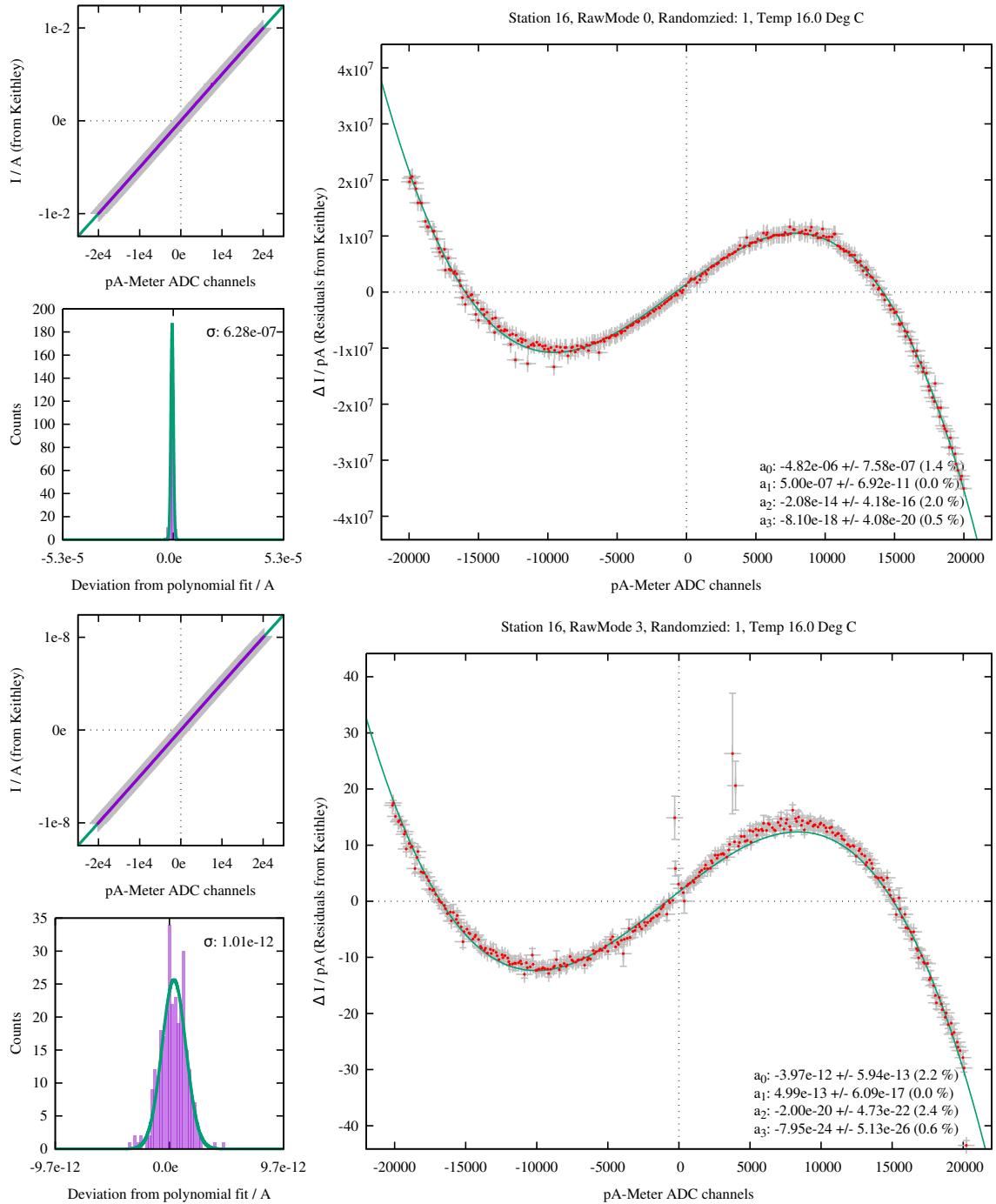


Figure 4.2: Calibration measurement for station 16 in lowest and highest sensitivity mode.

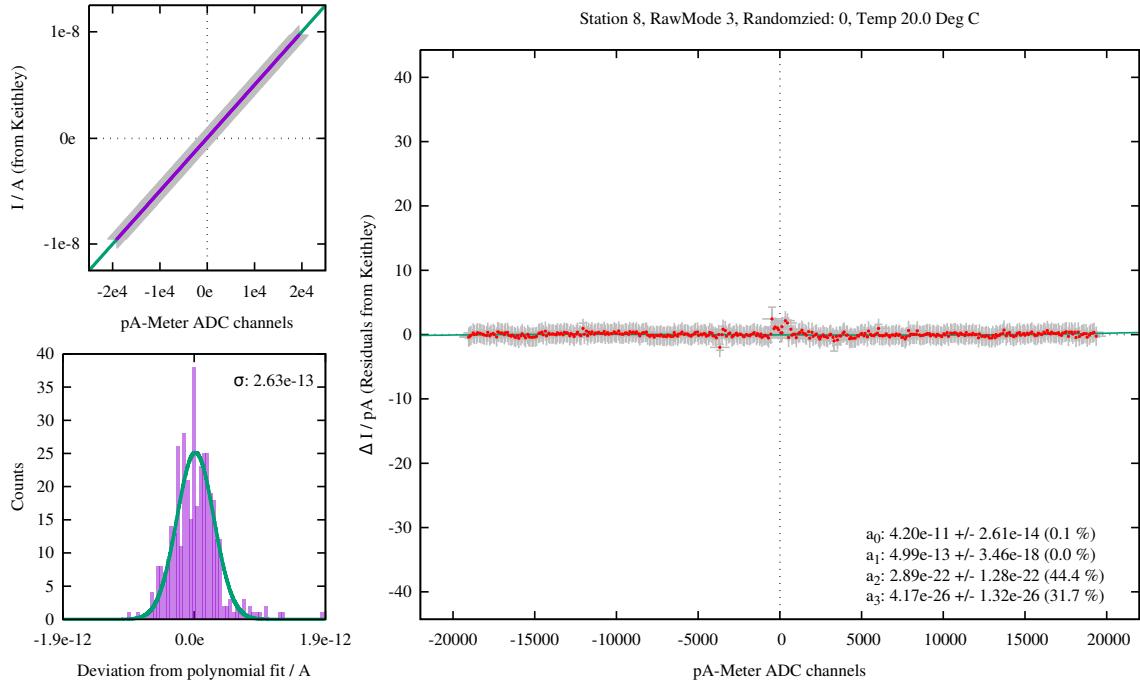


Figure 4.3: Calibration measurement for a station with removed diodes.

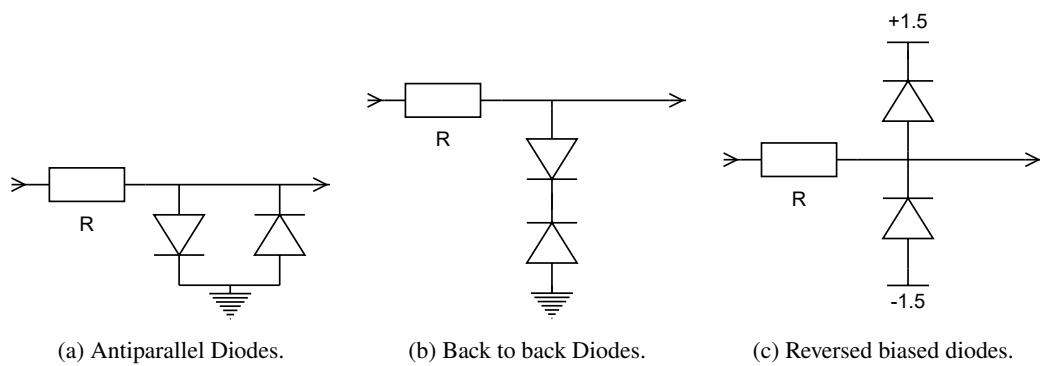


Figure 4.4: 3 different diode clamping circuits, that could be used as OVP.

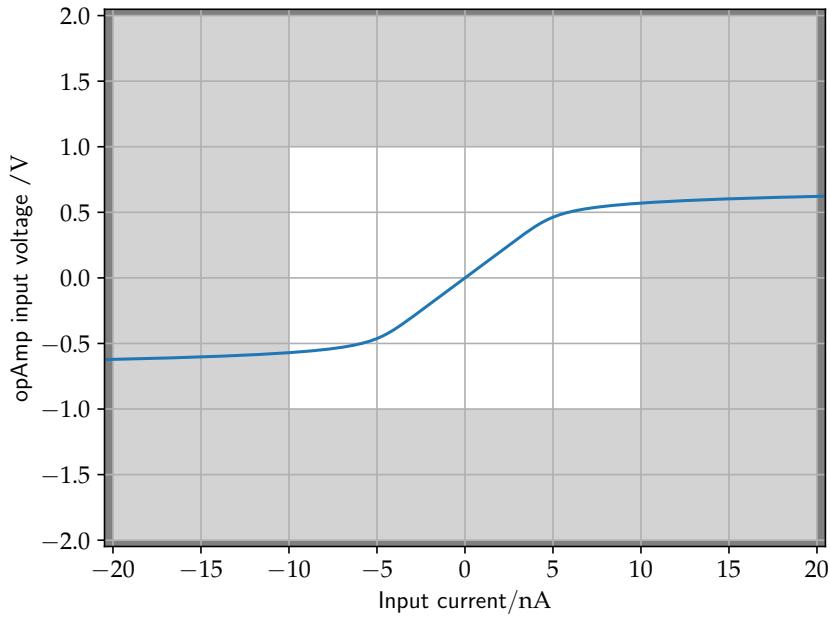


Figure 4.5: Over-voltage protection using antiparallel diodes. The dark grey area would not be measured by the pAM. The light grey area is the range, that can be digitised by the ADC, and the white area is the range that is calibrated.

model of the circuit shown in figure 3.3 was implemented, and the respective diode configuration was placed on the non-inverting input of the first amplifier.

### Antiparallel Diodes

Using this configuration will strongly limit the input voltage range to the diodes forward voltage, which typically is around 0.7 V. The simulation results are depicted in figure 4.5, showing that such a setup is not a viable solution.

### Back-to-Back Diodes

The second setup utilizes the reverse breakdown of a diode, making it conductive. For such a setup a diode with a low reverse breakdown voltage is needed, a Zener diode for example. Using diodes in this configuration allows a larger measurement range, depending on the reverse breakdown voltage of the diodes. The non-linearity is relatively small, but diode leakage currents can occur and influence the measurement. The simulation results are shown in figure 4.6.

### Clamping to an External Potential

The third option relies on the forward conductivity of the diodes. In contrast to the first design, it does not put such a sharp limit to the range. It allows an adjustable input range, depending on the external

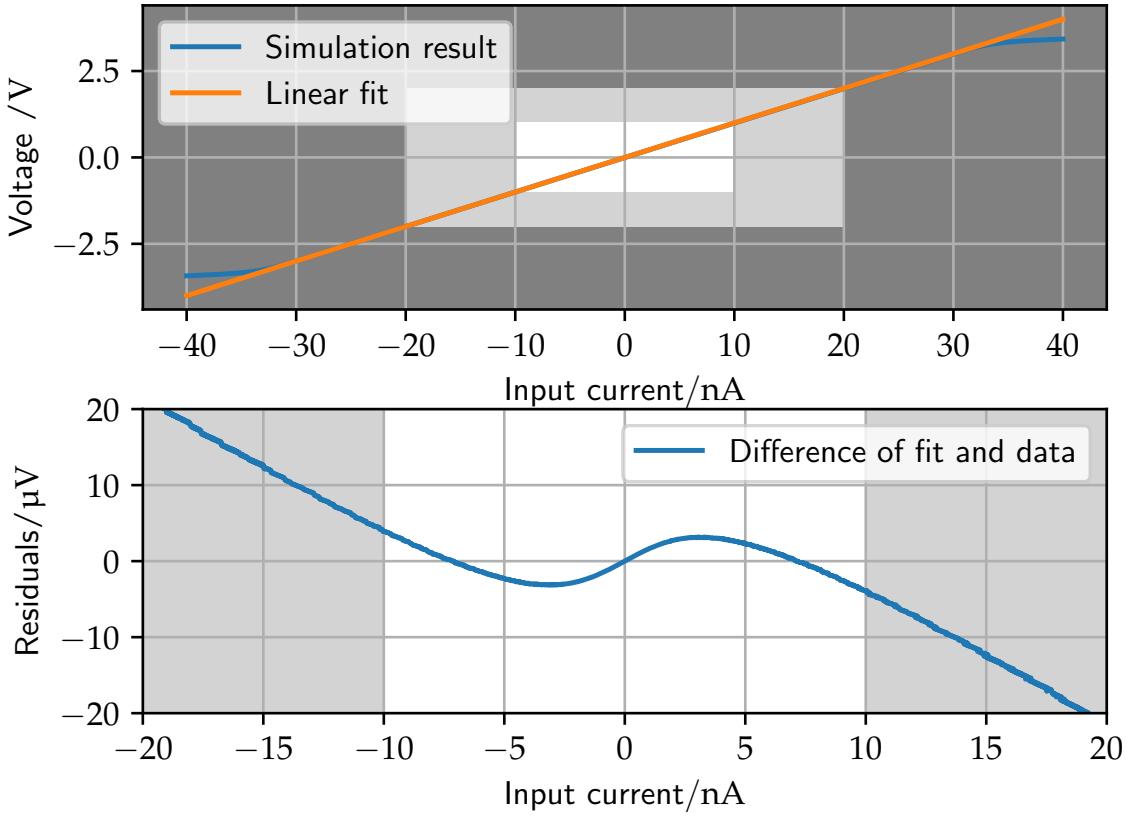


Figure 4.6: Over-voltage protection using back-to-back diodes. The dark grey area would not be measured by the pAM. The light grey area is the range, that can be digitised by the ADC. The white area represents the calibration range. The upper plot shows voltage on the non-inverting input vs input current and a fit on this data. The fit range is limited to the calibration range. The lower plot shows the deviation of simulation results from the linear fit.

potential. The resulting non-linearity is small compared to the other two options, as shown in figure 4.7. In normal operation both diodes are driven in a backwards biased configuration, leading to small leakage currents that can degrade the accuracy.

## 4.5 An OVP Prototype

For testing purpose, the third option was set up on a PCB, to be plugged in the signal path of a pAM station. Therefore, two considerations had to be made. First, a diode with low leakage and a high backwards breakdown voltage is necessary. Second, reliable voltage generators for the external potentials are needed.

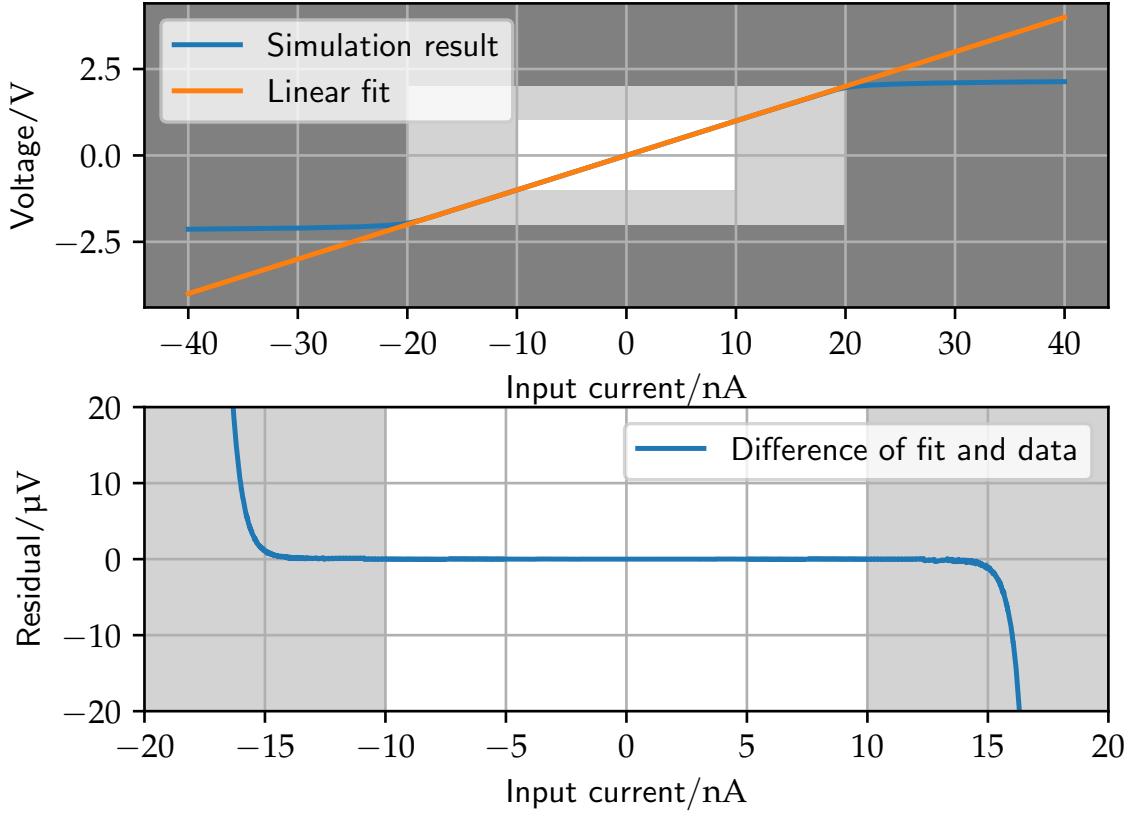


Figure 4.7: Over-voltage protection using diode clamping to an external potential. The dark grey area would not be measured by the pAM. The light grey area is the range, that can be digitised by the ADC. The white area represents the calibration range. The upper plot shows voltage on the non-inverting input vs input current and a fit on this data. The fit range is limited to the calibration range. The lower plot shows the deviation of simulation results from the linear fit.

#### 4.5.1 OVP Diode Choosing

For the choice of suitable OVP diodes, [49] suggests using the base-collector-diode of a small signal npn-transistor, as depicted in figure 4.8. Supposedly they have a lower leakage current than commonly available low-leakage diodes. To verify this claim, the leakage currents of several different transistors and diodes were measured using a Keithley 6517B Electrometer. Results of these measurements are shown in figure 4.9, leading to the conclusion that the base-collector np-transition indeed has a suitably low leakage current. For setup of a prototype, a BC546B transistor was chosen.

#### 4.5.2 Voltage Generation

For this setup, adjustable voltage regulators are needed, one to provide a negative and one for a positive voltage. Common regulators are designed to source currents. In case of an over-voltage condition in this setup, the positive supply would need to sink mA currents<sup>2</sup>. This problem needs to

<sup>2</sup>Correspondingly, regulators for negative voltages are meant to sink currents but fail at sourcing them.

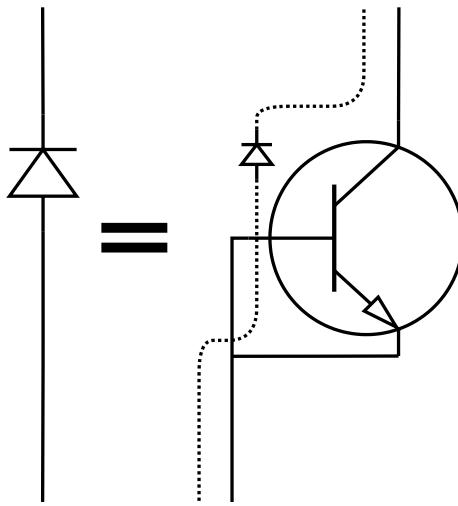


Figure 4.8: Equivalent circuit for a diode formed by the base-collector pn-junction of a transistor.

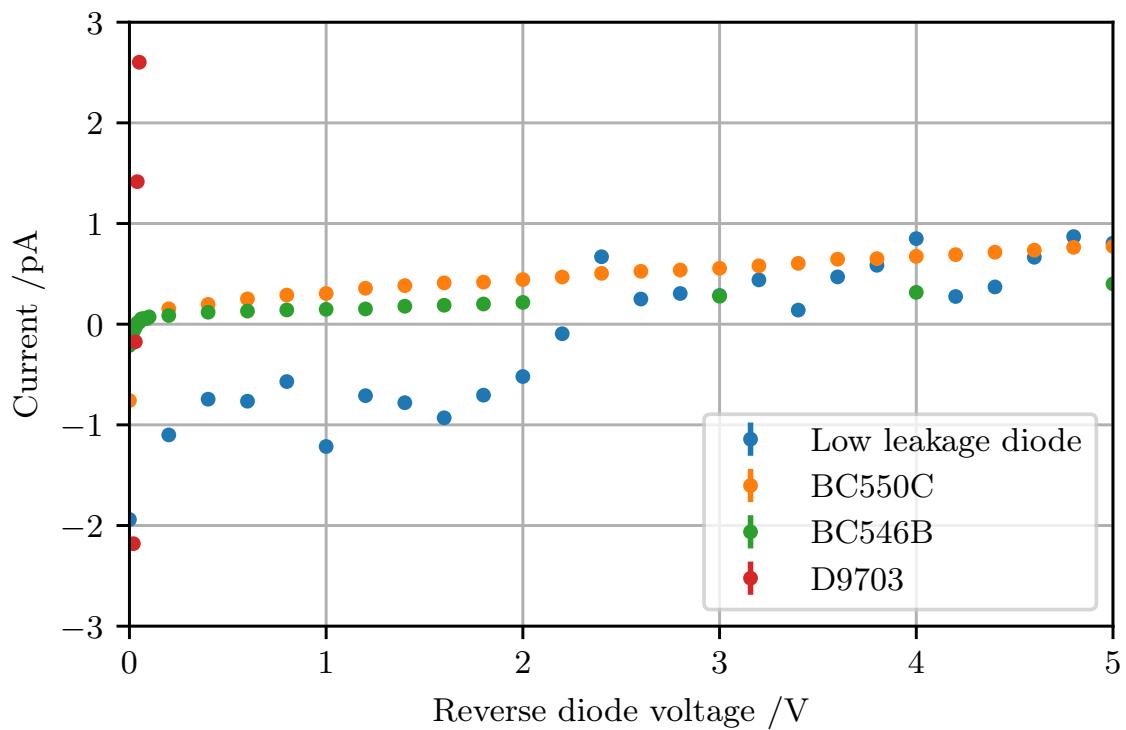
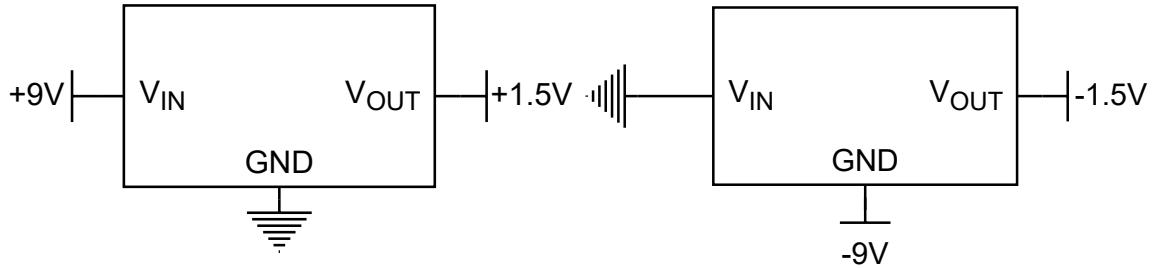


Figure 4.9: Measured leakage currents for different diodes. Error bars are too small to be displayed.



(a) Voltage regulator in its normal setup.

(b) How the same regulator can be used to provide a negative voltage with a higher source capability.

be circumvented.

A regulator for stepping down a positive voltage as in figure 4.10(a), can be used to provide a negative voltage, with a good sourcing capability. Connecting the **GND** pin to  $-9\text{ V}$  and the input to ground and adjust the output to  $7.5\text{ V}$  referred to **GND** results in a referred-to-ground output of  $-1.5\text{ V}$ . Similarly a regulator for negative voltages can be used to provide a positive voltage.

#### 4.5.3 Test Operation

To test the protection capability of the prototype, a  $5\text{ kV}$  DC source was connected between the input of the prototype and ground. The output was monitored with a voltmeter. In repeated test operations the output stayed at ground level, showing no influence from the input voltage.

As the setup proved to be a suitable OVP, it was inserted in the input path of a pAM. A calibration measurement, see figure 4.11, showed no non-linear effects. Something that could not be evaluated here is the temperature influence of this setup, but as diode leakage currents increase with temperature, it is expected to introduce temperature-dependent errors.

Unfortunately, updating all devices in operation with this OVP is not an option. The OVP needs to be connected in series between the input terminal and the non-inverting Op-Amp input. On the existing pAM boards, this is not possible. The setup of a new revision is necessary.

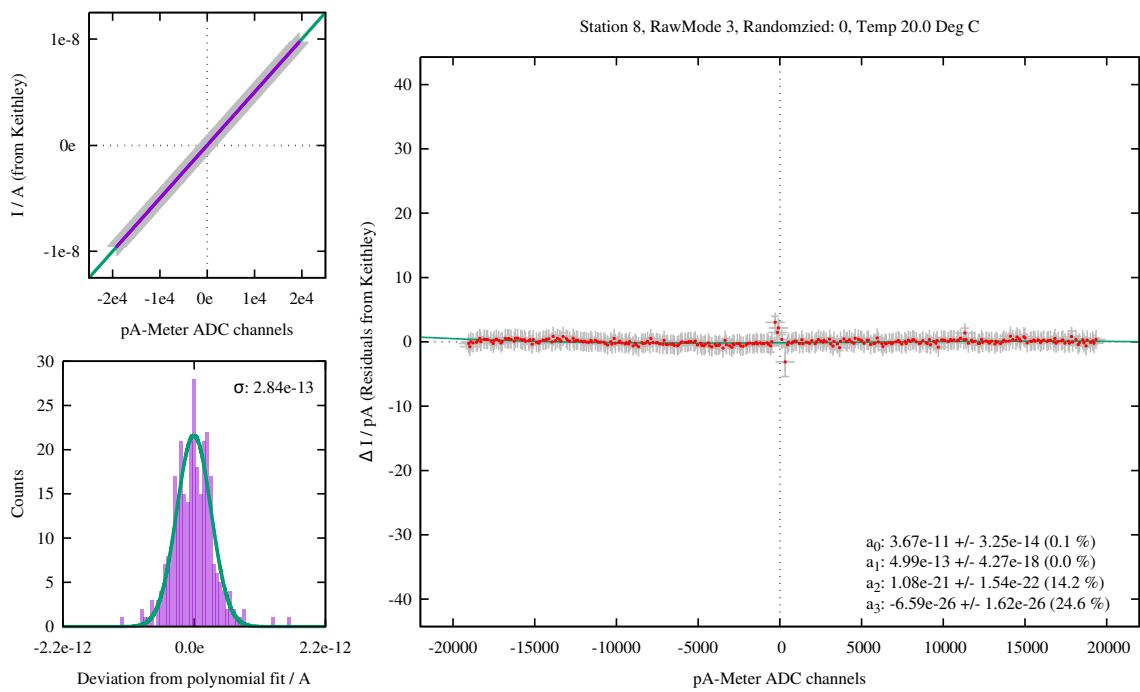


Figure 4.11: Calibration measurement run with the OVP prototype inserted in the input path.

# CHAPTER 5

## Development of a new front-end

Analysis of the design in chapter 4 led to the conclusion that a new revision is necessary. As modifications of the input stage were necessary, another option for a front-end setup should be evaluated here. A different type of measurement circuit should be capable to perform more precise measurements, allow faster readout and show a significantly lower temperature influence.

### 5.1 Design of the Amplification Stage

Modern current meters no longer use shunt resistor based configurations as an input stage, but rely on a so-called transimpedance amplifier [24]. Refer to figure 5.1 for a sketch of the basic principle. It allows measuring without a burden voltage and a direct current to voltage conversion:

$$U_{\text{OUT}} = -R_F I_{\text{IN}} \quad (5.1)$$

A TIA based front-end has some advantages over a common shunt resistor [50]. It allows precision measurements in the pA range and is already widely used in the field [51]. Making use of a TIA as the input stage will allow to build an OVP, without having to drive diodes at reverse voltages, see

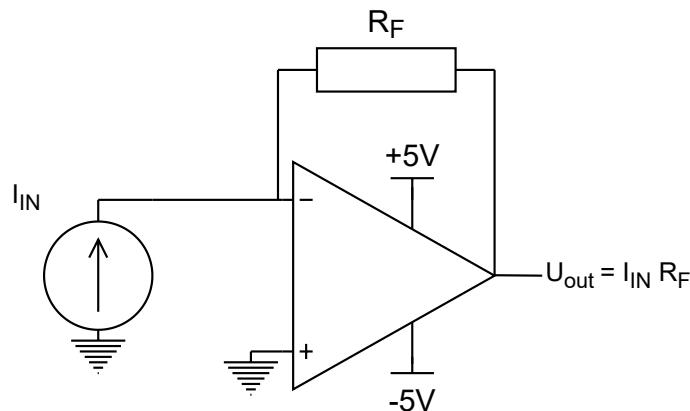


Figure 5.1: Basic principle of a TIA.

Op-Amp	Input bias current 1/fA typical	max	Offset voltage drift 1/( $\mu$ V °C)	Comment
ADA4530 [28]	<1	$\pm 20$	+0.13/-0.7	offers guard buffer
AD549L [44]	75	100	5	offers offset voltage correction
OPA128 [52]	40	75	5	offers offset voltage correction
LMC6442 [53]	5	4000	0.4	

Table 5.1: Characteristics of different Op-Amps. The AD549L is used in the older pAM designs; the OPA128 is its successor. The LMC6442 is used by [51].

section 5.2. Thus, reducing leakage currents and improving the accuracy.

## Amplifier Selection

When designing an amplification stage, the first step is to choose a suitable Op-Amp. For details on Op-Amps, see section 2.2.1. Therefore an insight into the requirements is necessary.

### Input Bias Current

One of the critical parameters for a TIA front-end is the input bias current. The input bias current places an absolute lower limit on the achievable accuracy. The amplifier chosen should have an input current possible as low as possible. The input bias current should not exceed 10 fA to allow accurate measurements of 100 fA.

### Noise

A total noise analysis of this stage can only be performed when the design is complete. The amplifier should have a voltage noise density in the regime of  $nV/\sqrt{Hz}$ , in order to have a maximum of total noise of a few  $\mu$ V. Current noise needs to be below  $1 fA/\sqrt{Hz}$ , as it gets amplified along with the current that is to be measured.

### Offset Voltage

The offset voltage of the amplifier needs to be significantly lower than the LSB of the ADC chosen. A maximum of a few  $\mu$ V with time and temperature is desired.

A comparison of different amplifiers suitable for this purpose is shown in table 5.1. For the design of a prototype, the ADA4530 was chosen as it offers the best compromise of all the considered amplifiers. Its internal guard buffer allows the implementation of an appropriate guarding structure, see 2.3.2. The ADA4530 is designed to be used as an electrometer amplifier. It offers a very low input offset drift over time and temperature as desired, see figure 5.2 for details. The datasheet provides a design guide for usage as a TIA to interface a photodiode, which can be used to deduce rules for the usage in the pAM.

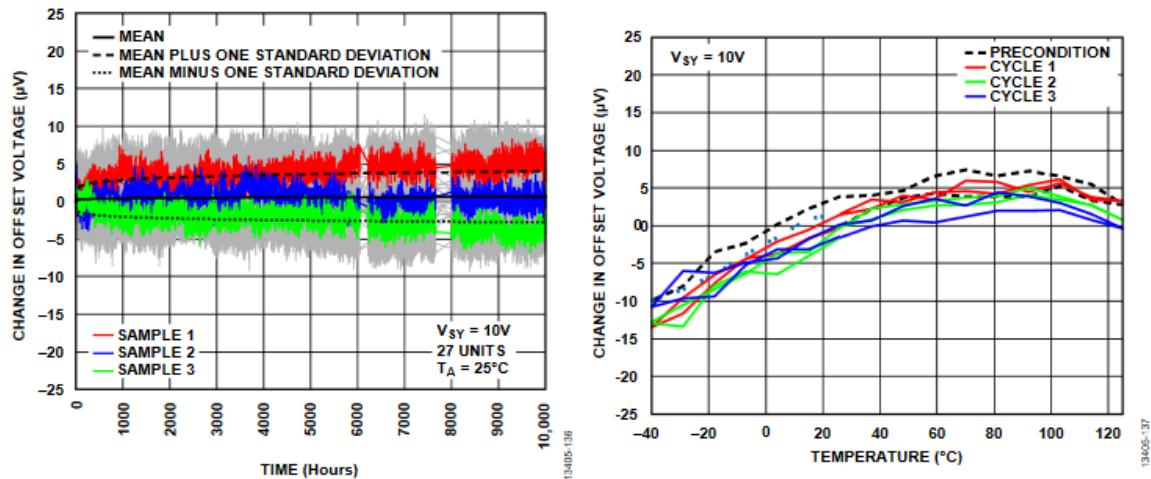


Figure 5.2: Drift of the ADA4530 offset-voltage drift vs time and temperature [28].

## ADC Selection

The next essential step is to choose a suitable ADC. To allow faster measurements as the revision 3 and 4 pAM, a higher sampling rate is necessary, in the order of kHz is the goal. As bipolar measurements are desired, an ADC with a bipolar input range is necessary. The LTC2327 was chosen here<sup>1</sup>, as it offers the best compromise of these parameters that could be found. It is a 16-bit SAR ADC, see section 2.2.2. The SAR architecture was chosen as offers a good compromise of sampling speed and resolution. The LTC2327 has a true bipolar input, offers an adjustable voltage range, 500 ksps maximum sampling frequency and non-linearity of less than one LSB.

## 5.2 Circuit Layout

Combining the ADC and amplifier, the front-end can be set up. The first step is to choose suitable voltage levels for driving the components.

### Supply Voltage Levels

The supply voltage for the LTC2327 is fixed to be 5 V. The ADA4530 can handle supply voltages ranging from  $\pm 2.5$  V to  $\pm 8$  V. For the design of a prototype a  $\pm 5$  V supply was chosen, as a 5 V source was necessary for the ADC. With this supply voltage for the Op-Amp, the maximum output of the amplifier stage is close to 5 V. The ADC range should accordingly be set to  $\pm 6.25$  V to ensure best overlap of signal and digitisation range. This input range digitised with 15-bits sets an LSB voltage of  $190.73 \mu V$ . Knowing this, the next step is to set up the feedback loop for the amplifier stage considering noise, resolution and compensation.

<sup>1</sup> Any information regarding performance and operation is taken from the datasheet [54].

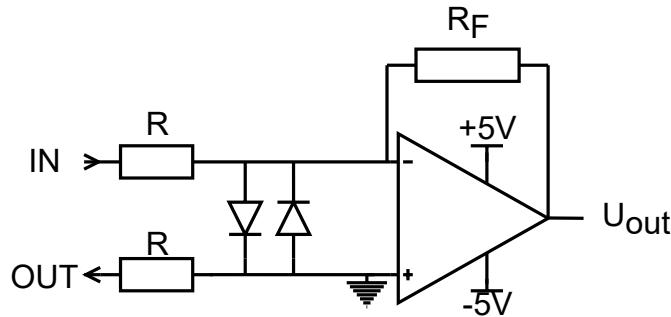


Figure 5.3: OVP diodes implemented in the TIA setup.

$R_F$	Input current range	LSB current
10 GΩ	$\pm 500 \text{ pA}$	19 fA
1 GΩ	$\pm 5 \text{ nA}$	190 fA
100 MΩ	$\pm 50 \text{ nA}$	1.9 pA
10 MΩ	$\pm 500 \text{ nA}$	19 pA

Table 5.2: Input current ranges and achievable resolution for different feedback resistors  $R_F$ , with a  $\pm 5 \text{ V}$  output range and an ADC input range of  $\pm 6.25 \text{ V}$  digitised in 15-bits.

### Over-Voltage Protection

The operational amplifier in this setup is sensible to over-voltages between the two inputs. In order to prevent this, OVP diodes are introduced in this setup as displayed in figure 5.3. The resistors  $R$  are supposed to limit the current through the diodes in case of an over-voltage condition. A resistor value of  $100 \text{ k}\Omega$  is chosen here, limiting the current to  $10 \text{ mA}$  in case of a  $1 \text{ kV}$  over-voltage. In theory, the voltage between the amplifier inputs is zero, however, the diodes still should have as small as possible leakage currents because of the small offset voltages from the amplifier. In accordance with findings from chapter 4.3, the BC546B transistor is used here.

### Considerations on the Feedback Loop

As introduced before, the output voltage is directly proportional to the feedback resistor and digitised in 15-bits. The resulting ranges and achievable resolutions for different feedback resistors are summarized in table 5.2. These considerations, however, are only valid for a pure DC case, which has its limitations as noise generated by parts as resistors<sup>2</sup> introduces AC components. Therefore, an insight into the frequency behaviour is necessary, requiring a model of the complete input stage. An approach to this is depicted in figure 5.4. It models capacitances between the two amplifier inputs combined into the shunt capacitance  $C_S$ , summarizing the parasitic capacitances from traces, OVP diodes and the external components. The parasitic capacitance is relatively small. The contribution from external devices is hard to model, as it contains various factors. In [51] the shunt capacitance was estimated to be  $5 \text{ nF}$ . For simulations, here an upper limit of  $10 \text{ nF}$  is set. The resistance between the inputs is modelled with  $R_S$ ; the resistance between the two input paths through the PCB is the main contributor.

<sup>2</sup>For details of electronic noise refer to chapter 2.2.3.

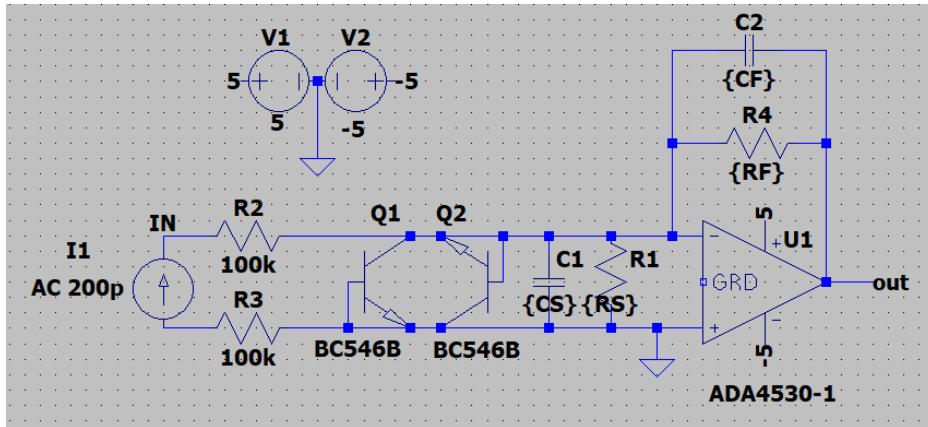


Figure 5.4: Model as implemented in LTspice for simulating the behaviour of the input stage. The Resistors R2 and R3 are the current limiters for the OVP setup. Models for the BC546B are implemented in the transistors Q1 and Q2. C1 and R1 allow adjusting of shunt capacitance and resistance. C2 and R4 allow adjustment of the feedback impedance. For the ADA4530 a model is provided by Analog Devices.

It is estimated to be in the order of  $T\Omega$  when the board is perfectly clean but gets significantly smaller when there are contaminations in between the traces. Any capacitance between input and output of the amplifier is modelled in  $C_F$ ; it can also model an additional feedback capacitance introduced. This model was used to run simulations on the feedback network.

### Frequency Analysis

In figure 5.5(a) different feedback resistors were simulated with both shunt and feedback capacitance set to 0 F. The frequency response shows a peak at a frequency depending on the feedback resistance. This peak due to resonance of an RC-network formed by  $R_F$  and the shunt capacitance  $C_S$ . Figure 5.5(b) shows how the position of the resonance varies with  $C_S$ . These resonances are potentially dangerous and could destroy the amplifier or the ADC. The resonances can be compensated by adding a feedback capacitance. There are mathematical approaches to calculate  $C_F$  depending on the Op-Amp,  $R_F$  and  $C_S$  [55]. Figure 5.5(c) shows the combination of different feedback capacitances with different shunt capacitances. Leading to the conclusion, that for the choice of feedback capacitance both compensation and bandwidth need to be considered.

### Noise Performance

All these parameters influence the noise of the setup as well, see section 2.2.3. The LTspice noise simulation determines the noise spectral density of a circuit. The above model was used to run noise simulations, again depending on the parameters introduced. One of the major contributors to the noise is the feedback resistance thermal noise. For a  $10\text{ G}\Omega$  resistor at  $25^\circ\text{C}$  it is around  $12.8\text{ }\mu\text{V}/\sqrt{\text{Hz}}$ , see also figure 5.6(a). The output signal (S) scales with  $R_F$  (equation 5.1), the noise (N) of the resistor scales with  $\sqrt{R}$  (equation 2.1). Therefore the signal to noise ratio ( $S/N$ ) scales with  $\sqrt{R}$ , making a larger feedback resistance the preferred choice.

Another important factor in noise considerations is the feedback capacitance. As shown in figure 5.6(c) a lower value of  $C_F$  leads to lower overall noise.

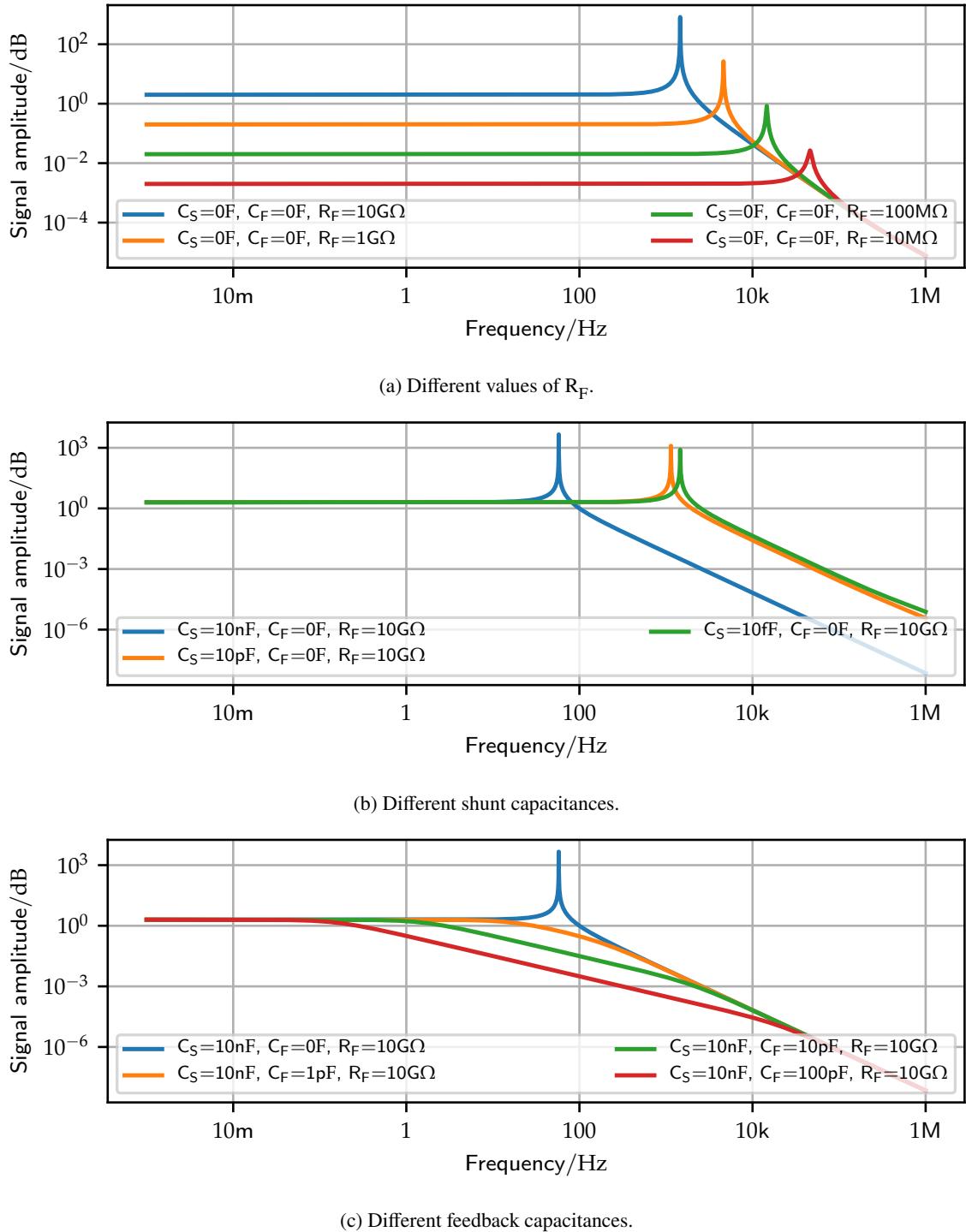


Figure 5.5: The effect of different parameters on the frequency behaviour of the TIA model. The y-axis shows the signal amplitude in dB. The current source is adjusted to achieve 1 V DC output at  $R_F = 5 G\Omega$ .

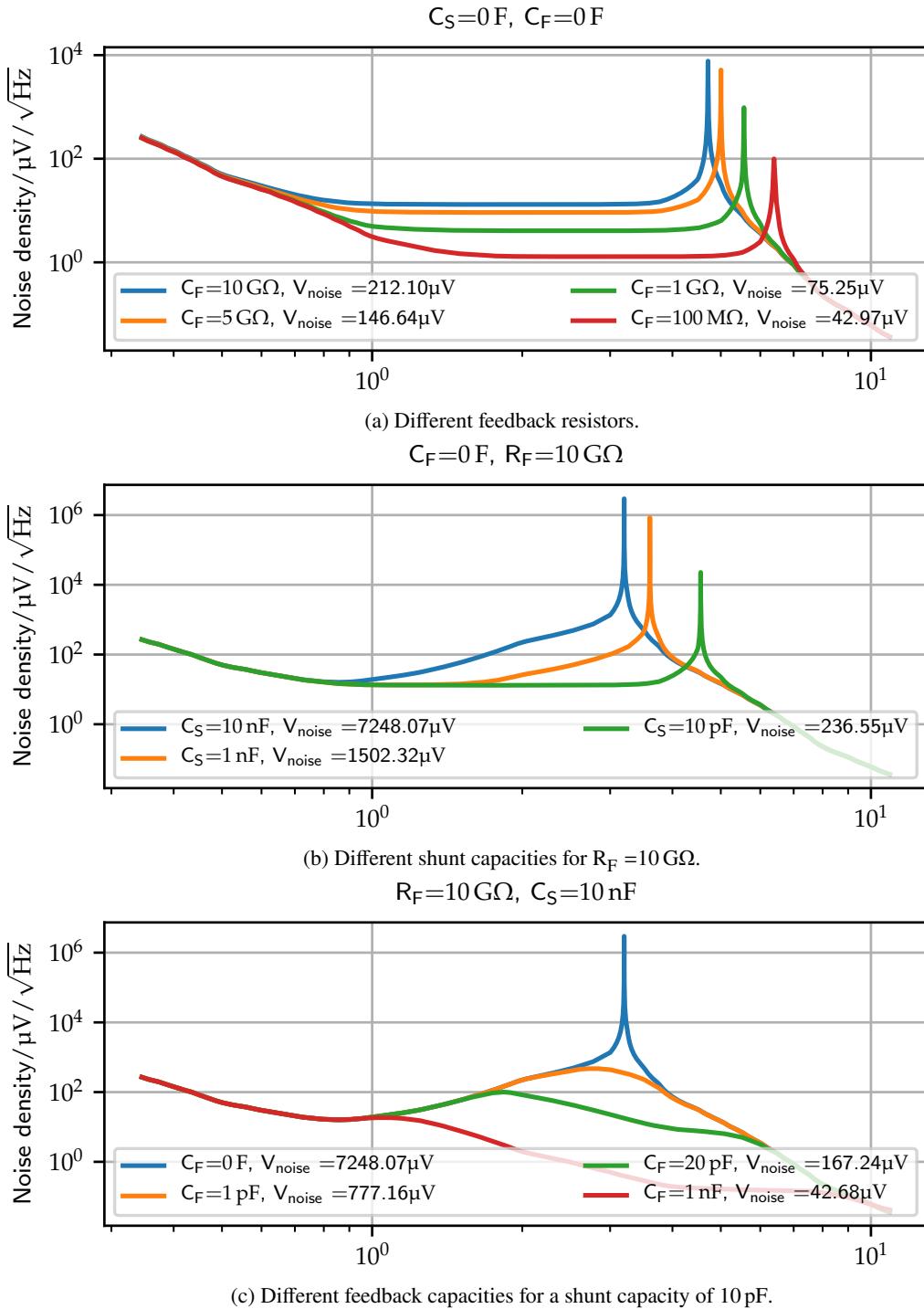


Figure 5.6: The effect of different parameters on the spectral noise density.  $V_{\text{noise}}$  denotes the integrated rms noise.

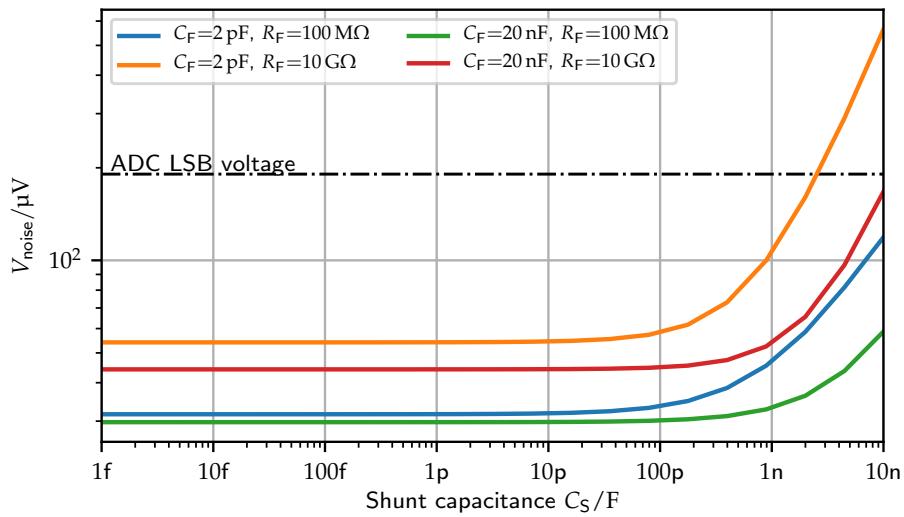


Figure 5.7: Noise scaling with the shunt capacitance. To allow comparison, the ADC LSB voltage is shown. It denotes the voltage corresponding to one ADC channel.

In Figure 5.7, the influence of the shunt capacitance on the integrated rms noise is depicted. The ADC LSB is only exceeded with  $C_F \geq 2\text{ nF}$ . For the prototype, the feedback capacitor was chosen to be  $2\text{ pF}$ , with a maximum feedback resistance of  $10\text{ G}\Omega$ . This setup should allow a bandwidth of  $\approx 20\text{ Hz}$ <sup>3</sup> and an integrated rms noise of around  $60\text{ }\mu\text{V}$  for low values of  $C_F$  and up to  $242\text{ }\mu\text{V}$  for higher values. This choice of feedback components offers a good compromise on stability, bandwidth and noise reduction.

### 5.3 Implementation of a Prototype

The effects discussed above were used to design a prototype. The setup of the back-end circuitry was kept from older designs to ensure maximal compatibility with the available infrastructure. The firmware described in section 4.1 was modified to be compatible with the new components. The front-end was revised completely. The implementation of the prototype is discussed in the following paragraphs.

#### Supply Voltage Generation

As stated previously, the LTC2327 requires a  $5\text{ V}$  supply, and the amplifier is planned to be driven from a  $\pm 5\text{ V}$  supply. Additionally a  $1.25\text{ V}$  precision reference is necessary to set the ADC input range to  $6.25\text{ V}$ . As the pAM were so far driven by  $9\text{ V}$  block batteries, or a photovoltaic based supply delivering  $(5.9 \pm 0.1)\text{ V}$  [5], the regulators generating the supply voltages should work with an input range of  $5.5$  to  $10\text{ V}$ . The datasheet for the ADA4530 suggests using an ADP7118 LDO for the positive supply rail and an ADP7182 LDO for the negative supply rail as they provide a stable low

<sup>3</sup>The readout frequency achievable with this design is around  $7\text{ Hz}$ , limited by the  $\mu\text{C}$ , as discussed in section 5.4. Hence a bandwidth  $20\text{ Hz}$  is sufficient.

noise voltage [28]. The reference voltage to the ADC is generated with an LTC6655-1.25 as suggested by the datasheet.

## Temperature Sensor

As temperature was one of the weak spots in older designs, a temperature sensor was added to the design. For precision temperature measurements, an ADT7410 was chosen<sup>4</sup>. It offers a  $0.5\text{ }^{\circ}\text{C}$  accuracy and low power consumption of  $7\text{ }\mu\text{W}$  in sleep mode. It has indicators for over- as well as under- temperatures (INT pin) and for critical temperature (CT pin). These threshold values can be programmed remotely. As the pAMs are not supposed to be exposed to extreme temperatures, these pins are not wired to the  $\mu\text{C}$ . With the A0 and A1 pins, the address of the ADT can be set, as only one is used in the design, they are grounded. The setup was done as specified in the datasheet. For communication, the sensor uses the I<sup>2</sup>C protocol, see section 2.4.5. The ADT7410 can convert temperature either with 13 or 16-bits, the latter is chosen here. Measuring temperature is implemented in the mode signal blinking, in order to save time, see the code snippet in listing 5.1. To save power and processing time, the signal blinking and temperature read is only run all 100 measurement cycles.

Listing 5.1: The temperature reading and mode blinking loop.

```
if(CYCLE_COUNTER>=100){
    //blink \& read temperature every few seconds only, to save power.
    temperature_sensor_write(0x03, 0xA0, 1); // start measurement, data ready in 240ms
    for (int i = 0; i < mode; i++) {
        \\blink to indicate the mode of operation
        WDTCTL = WDT_ARST_1000;
        debug_led_on();
        wait(300);
        debug_led_off();
        wait(300);
    }
    //if reading not successful set temperature to 255
    if (!(temp = temperature_sensor_read(0x00, 2)))
        temperature=255;
    //convert \ac{adc} channels to degree celcius
    if (temp > 0) { //positiv temperature
        temperature = temp / 128.;
    } else { //negative temperature
        temperature = (temp - 65536.) / 128.;
    }
    CYCLE_COUNTER=0;
}
```

## Interfacing the ADC

The pinout of the LTC2327 is summarized in table 5.3. The table also shows the implemented connections. Over-driving the **REFIN** pin with 1.25 V sets the input range to be  $\pm 6.25\text{ V}$ . An ADC

<sup>4</sup>Information on the ADT7410 are taken from the Datasheet [56].

PIN	Name	Purpose	Implementation
1	<b>V<sub>DDLBYP</sub></b>	Supply bypass pin	Bypassed with a 2.2 $\mu$ F ceramic capacitor
2	<b>V<sub>DD</sub></b>	Power supply pin	5 V supply input
3,6,16	<b>GND</b>	Ground	
4	<b>IN+</b>	Signal input	Connected to Op-Amp output
5	<b>IN-</b>	Analogue ground sense	Signal ground sensing. Connected to the groundplane.
7	<b>REFBUF</b>	Internal reference buffer output	Bypassed with a 47 $\mu$ F ceramic capacitor
8	<b>REFIN</b>	Internal Reference output / Input to reference buffer	Overdriven with a 1.25 V reference. Bypassed with a 100 nF ceramic capacitor
9	<b>CNV</b>	Triggers convert	Connected to the $\mu$ C
10	<b>CHAIN</b>	Chain mode selector pin	Grounded as chain mode is not needed
11	<b>BUSY</b>	Indicator for a running conversion	Connected to the $\mu$ C to serve as an interrupt signal
12	<b>RDL/SDI</b>	Serial data input	Grounded, as it is not needed in normal operation
13	<b>SCK</b>	Clock input	Connected to the $\mu$ C
14	<b>SDO</b>	Serial data out	Connected to the $\mu$ C
15	<b>OV<sub>DD</sub></b>	Logic voltage level input	Connected to the 3.3 V regulator from the back-end

Table 5.3: Pin configuration of the LTC2327 [54].

conversion is started, when the **CNV** pin is set high from the microcontroller, the LTC pulls the **BUSY** pin high and converts data. On the falling edge of **BUSY**, data is ready and can be read from **SDO**. The timing diagram for this is depicted in figure 5.8. The original idea was to use the falling edge of **BUSY** as an interrupt, as it was done in previous revisions. However,  $t_{CONV}$  is only 1.5  $\mu$ s, which is less than 3 clock ticks of the  $\mu$ C and in almost all cases does not trigger an interrupt. To overcome this, a delay is implemented after starting a conversion, see listing 5.2.

Listing 5.2: ADC conversion loop for taking 128 samples.

```
for (pos = 0; pos < NUM ; /* pos++ in loop */) {
    WDTCTL = WDT_ARST_250;
    ADC_PORT_out.set_high(ADC_CNV);
    ADC_PORT_out.set_low(ADC_CNV); // trigger conversion
    __delay_cycles(10); // wait till conversion is ready
    mem[pos++] = clock_bits(0x0000,16) // reading 16 ADC bits
}
```

Here the internal function `_delay_cycles()` is used instead of `wait()`. The wait function uses a timer overflow interrupt to measure the time passed and does not work reliable with times less than 1 ms. `_delay_cycles()` has the disadvantage of higher power consumption but allows a much shorter time

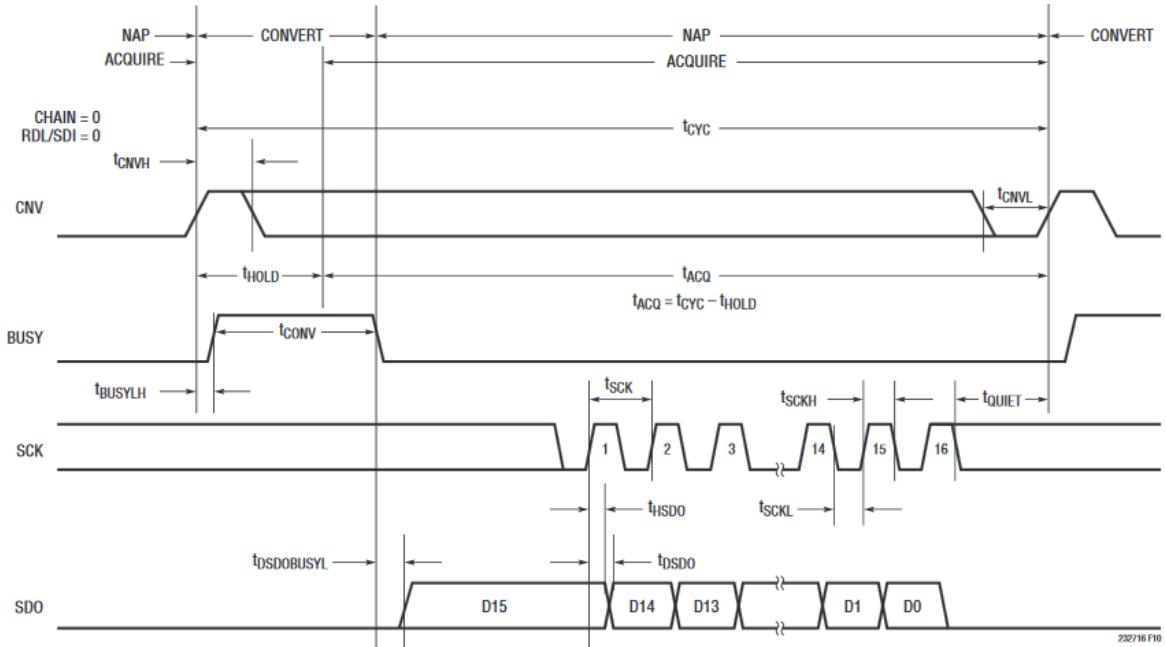


Figure 5.8: Conversion timing characteristics of the LTC2327 [54]. A conversion is started by pulling **CNV** HIGH; it can be pulled LOW after  $t_{CNVH} = 20$  ns. A conversion takes  $t_{CONV} = 1.5$   $\mu$ s and is indicated by **BUSY** being HIGH. After a maximum of  $t_{DSDOBUSYL} = 5$  ns data is valid for reading on **SDO** and can be read out via the SPI interface.

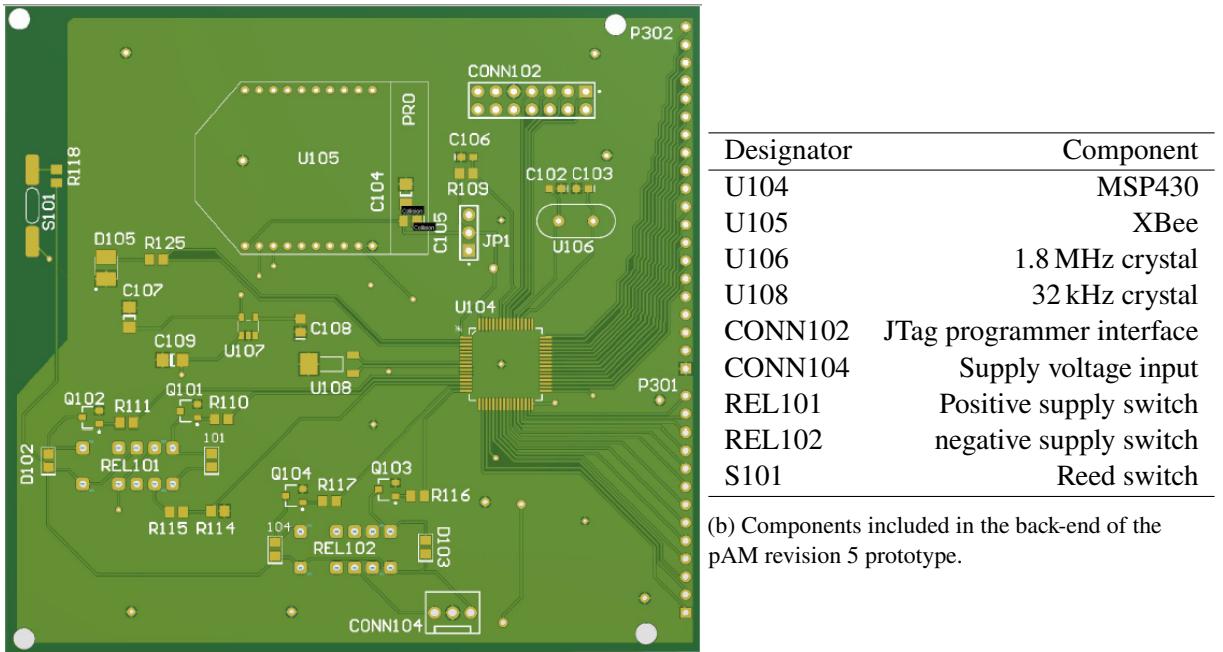
measurements. Still, the delay introduces an extra 7 ms runtime per full sample of 128 ADC readings. Lowering the cycle number below 10 sometimes results in problems with the ADC conversion. These timing issues could be overcome by choosing a faster clock speed on the  $\mu$ C but would result in significantly higher power consumption.

### Amplifier & Guard Structure

For leakage reduction, the prototype relies on a guard ring. The guard structure surrounds both input traces as well as one side of the feedback resistors. Guarding structures do not offer high protection against leakage currents as ceramic stand-offs but are easier to implement and mechanically stable. Refer to section 2.3 for details. The prototype is set up on a common FR-4 board; this can be changed for mass production.

The PCB layout is shown in figures 5.9 and 5.10; schematics can be found in appendix A.3. The prototype was equipped with 3 slots for feedback resistors, selectable with relays. As finding suitable relays that are designed for low leakage currents proved to be an impossible task, they were placed after the feedback resistors in the voltage signal path. The remaining tracks form small antennas, but that is a disadvantage to be accepted.

The prototype was split into two boards to enable testing multiple front-ends. The back-end featuring the MSP430, the XBee module and attached electronics was designed to allow maximum compatibility with possible front-end designs and therefore has most of the available  $\mu$ C pins mapped to one boarder



(a) Back-end PCB designed to control different front-end concepts.

Figure 5.9: Back-end prototype. Schematics can be found in figure A.7.

of the board.

### Shielding

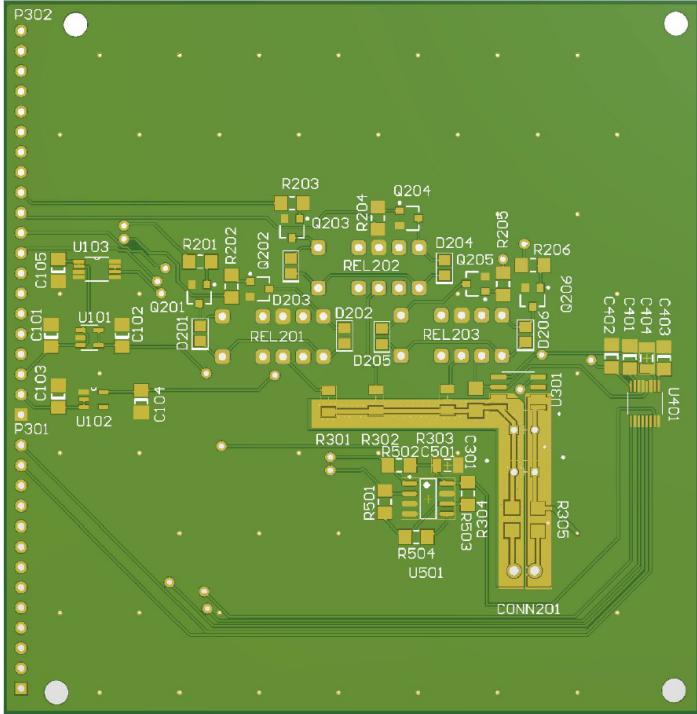
As explained in section 2.3, shielding is an important factor to reduce interference noise. For the construction of this prototype, no further shielding was implemented. The device is operated in the calibration setup, which already features a Faraday box for shielding. The same goes for regular test setups, where the pAM are operated.

For high-voltage supply of GEMs, only the inner conductor is used. The outer conductor is connected to earth ground. Therefore, cables connecting the pAM do not need extra shielding, as the outer conductor acts as a shield.

### Cleaning Procedure

In section 2.2.4, the importance of cleaning was discussed. Cleaning needs to be performed after the board is assembled. The relays slots must be left unequipped. The cleaning is performed according to the recommendations of the ADA4530 datasheet [28]:

1. Ultrasonic cleaning in isopropyl alcohol for 30 min
2. Flushing the board with isopropyl
3. Use a brush scrub the solder joints
4. Blow dry using compressed air



(a) Front-end prototype PCB, set up as described in section 5.3.

Designator	Component
U101	5 V regulator
U102	-5 V regulator
U103	1.25 volt reference
U301	ADA4530 Op-Amp
U401	LTC2327 ADC
U501	ADT7410 temperature sensor
REL201	10 GΩ switch
REL202	1 GΩ switch
REL203	100 MΩ switch
CONN201	Input terminals

(b) Components included in the front-end of the pAM revision 5 prototype.

Figure 5.10: Front-end prototype. Schematics can be found in figure A.8.

5. Solder relays in place

6. Repeat step 2, 3, and 4

7. Bake the board at 80 °C for 3 h

For further cleaning processes, the ultrasonic bath needs to be skipped. Baking should always be performed at the end of a cleaning process, as it reduces the moisture of board and components. The relays need to be soldered in place after the ultrasonic bath, as they often get damaged in the process. An approach using sockets for the relays resulted in degraded precision.

### Expected Temperature Behaviour

Like all kinds of electronics, the above setup will experience a temperature influence. The expected influences of the components are shown in 5.4. Influences from the Op-Amp and the ADC are negligibly small. However, the influence of the feedback resistor over a full scale of ±500 pA results in an error of around 3 pA, over the full temperature range. The temperature of the device should, therefore, be observed while measuring.

Component	Parameter	Fluctuation vs Temperature 1/°C	in the range of 15 to 30 °C
ADA4530 [28]	Offset voltage	$\pm 5 \mu\text{V}$	
	Input bias current	$\pm 0.1 \text{ fA}$	
LTC2327 [54]	Non-Linearity	$\leq 1 \text{ LSB}$	
	Full-Scale Error	$\pm 2 \text{ LSB}$	
	Offset Error	$\ll 1 \text{ LSB}$	
Feedback resistor (10 GΩ)	Temperature Drift	$\pm 200 \text{ ppm}/\text{°C}$	0.3 %

Table 5.4: Temperature dependencies of components in the front-end.

## 5.4 Measurement Results

After construction of the prototype, as described above, first test measurements could be done. To allow measurements with the new front-end using the calibration station, a resistor had to be added in series between the voltage source and the pAM. For the most sensitive mode, a 2 GΩ resistor was chosen. Combined with a  $\pm 1 \text{ V}$  voltage range, this yields a current range of 50 nA. First calibration measurements produced the result displayed in figure 5.11. The resolution of 0.67 pA is much higher, than the noise simulations hinted, see section 5.2. In figure 5.11 on the right, the residuals plot shows large errorbars for the reference current measurement of the Keithley. The pAM results, however, do not show significant errors. For both readings, the errors are the standard deviation of a sample of readings. To investigate the source of the large resolution determined in the calibration, the errors of Keithley and pAM readings are plotted against the driving voltage, as shown in figure 5.12. The pAM measurements have an average error of  $(1.10 \pm 0.21)$  ADC channels, which would correspond to a theoretical current error of  $(20.1 \pm 4.0) \text{ fA}$ . In contrast to this, the current readings have an average measurement error of  $(0.70 \pm 0.31) \text{ pA}$ . The Keithley, therefore, seems to be the limiting factor.

Better resolutions were achieved in calibrations of previous revisions, see figure 4.11. The difference between these measurements is (apart from the pAM in use) the measurement time, that is available for the Keithley. The changes to the CalibAna software, discussed in section 4.1.5, included a delay of 1 second in between conversions of the pAM. With the older revisions, this delay was not necessary, as each conversion of a pAM took eight seconds already. A longer delay of eight seconds was therefore set in the software, to verify if the measuring time is a constraint here. The result of this measurement is shown in figure 5.13. The resolution improved in comparison with the previous measurement but is still not compatible with the pAM performance. The average error on the pAM reading is comparable with the previous measurement. The average current error improved to  $(0.25 \pm 0.03) \text{ pA}$ . Further increasing the measurement time, only had a limited effect on the achievable resolution.

From the manual of the Keithley 6517b electrometer [57], it should be able to perform measurements with a precision of 10 fA in a range of  $\pm 2 \text{ nA}$ . To verify, that this error is not caused by the pAM inserted, a calibration measurement without pAM was performed. The resulting current reading error is displayed in figure 5.14. This measurement is in accordance with the previous results. Hinting, that using the Keithley to investigate on the characteristics is not sufficient for the new pAM revision.

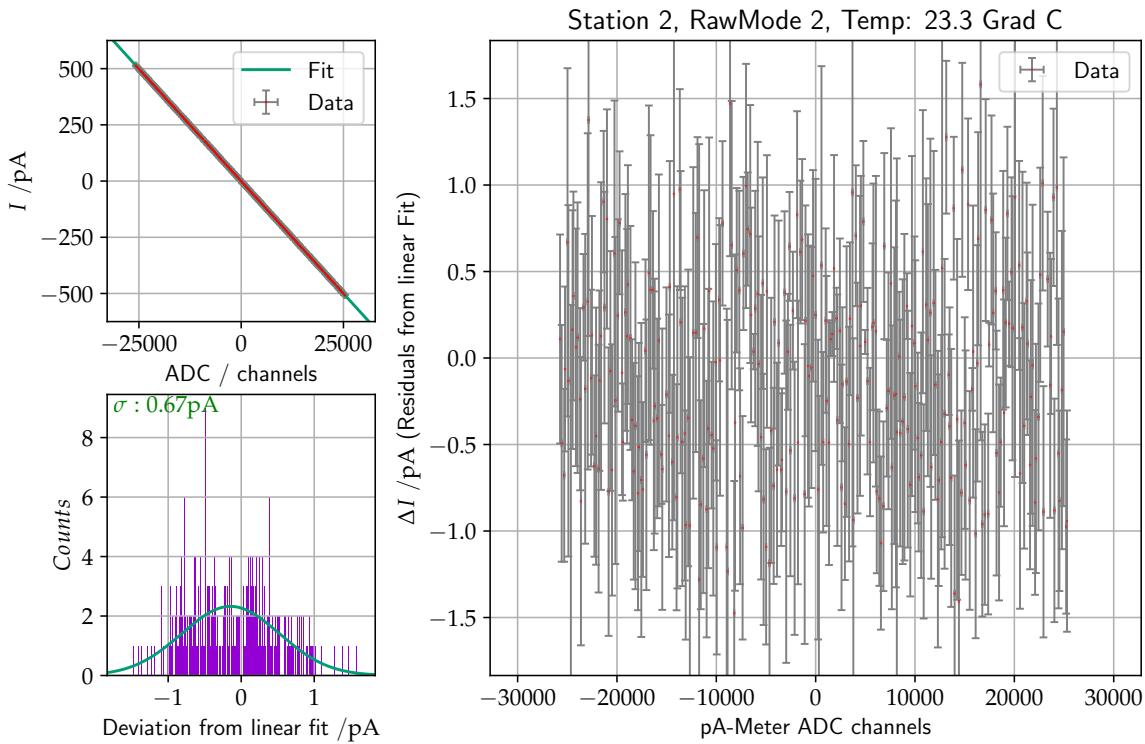


Figure 5.11: Calibration measurement done with the developed prototype. The obtained resolution is estimated to be 0.67 pA.

#### 5.4.1 Power Consumption

The power consumption of the device plays an important role, as the devices need to operate on battery supply. A measurement of the current draw led to the results shown in figure 5.5. The power

	Current 1/mA
Without XBee	(7.0±0.5)
XBee sending	(50±5)
Total average	(11.0±0.7)

Table 5.5: Current draw of the pAM prototype. For the measurement, a  $10\Omega$  resistor was inserted in the 9 V supply line, and the voltage drop was measured using an Oscilloscope.

consumption increased significantly. The main contributor to the baseline power consumption is the LTC2327. According to the datasheet, it draws 6 mA in normal mode. The sleep mode of the ADC proved not to be suitable for the application, as wake up of the ADC takes 200 ms. Another factor that contributes to an increased power consumption is the increased sampling rate, increasing the XBee run time. The increased power demands, lower the run time with a 9 V block battery to around 2 to 3 days. Switching the power supply away from the 9 V blocks could overcome the problem. A more durable supply could be build utilizing lithium-ion accumulators like the Samsung NR18650-25R. Two such accumulators in series can provide around 7 V, which is sufficient to drive a pAM. The large capacity

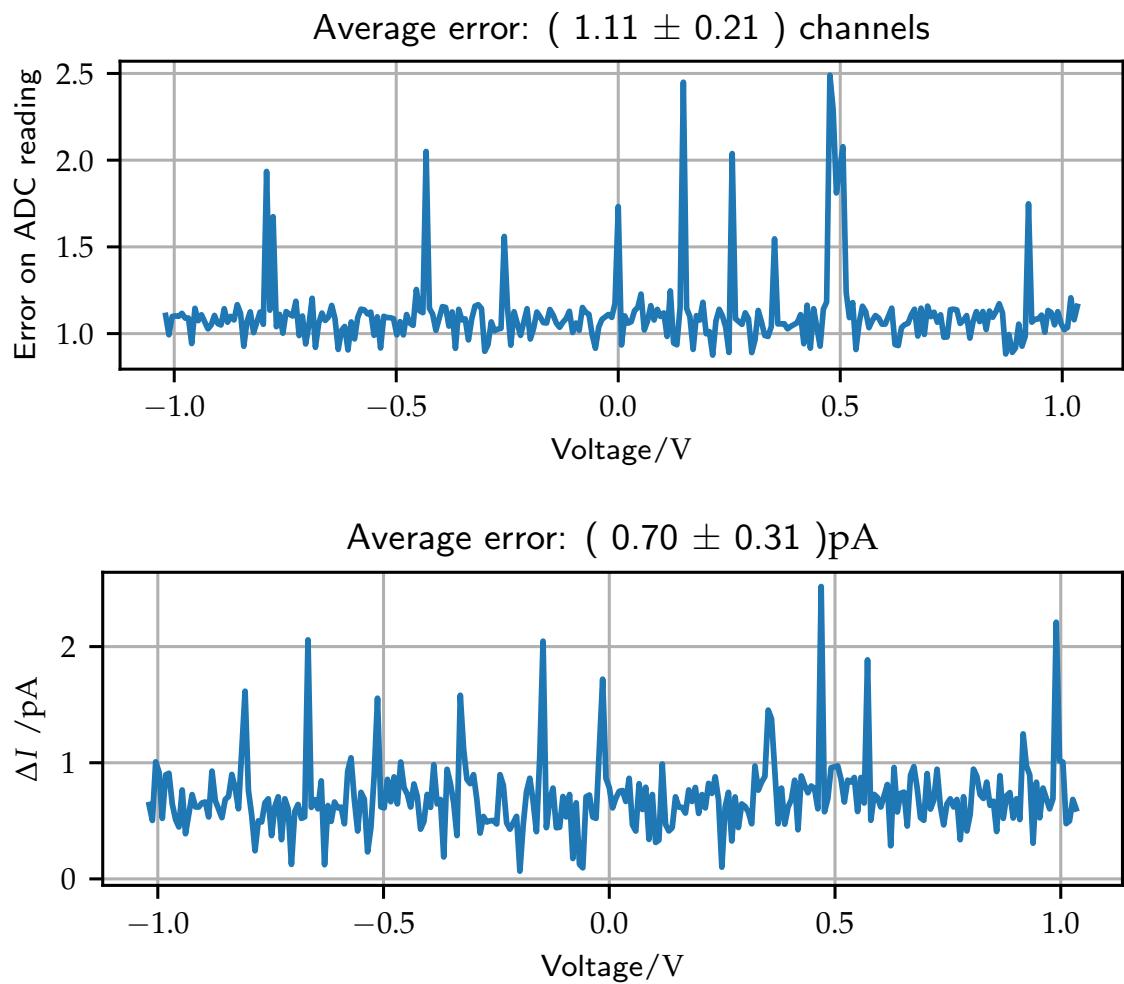


Figure 5.12: Comparison of the errors on the pAM and the reference readings. Both are plotted against the voltage, that is used to generate the current to be measured.

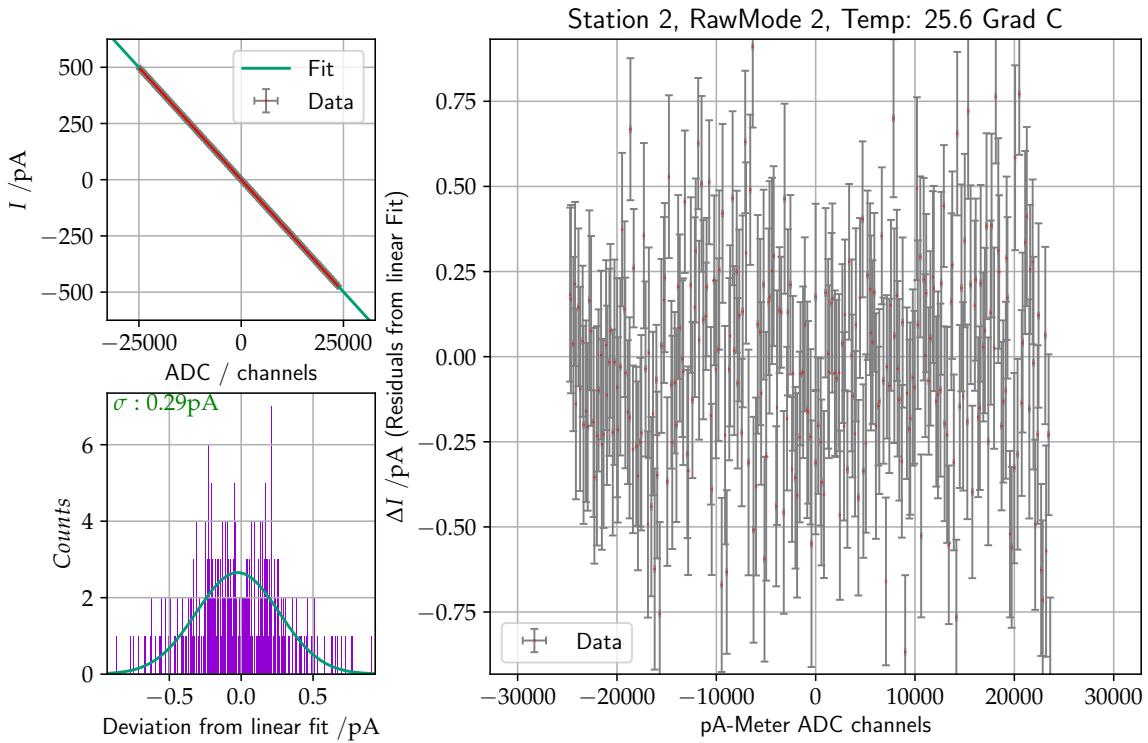


Figure 5.13: Calibration measurement done with the developed prototype and an elongated measurement time for the Keithley.

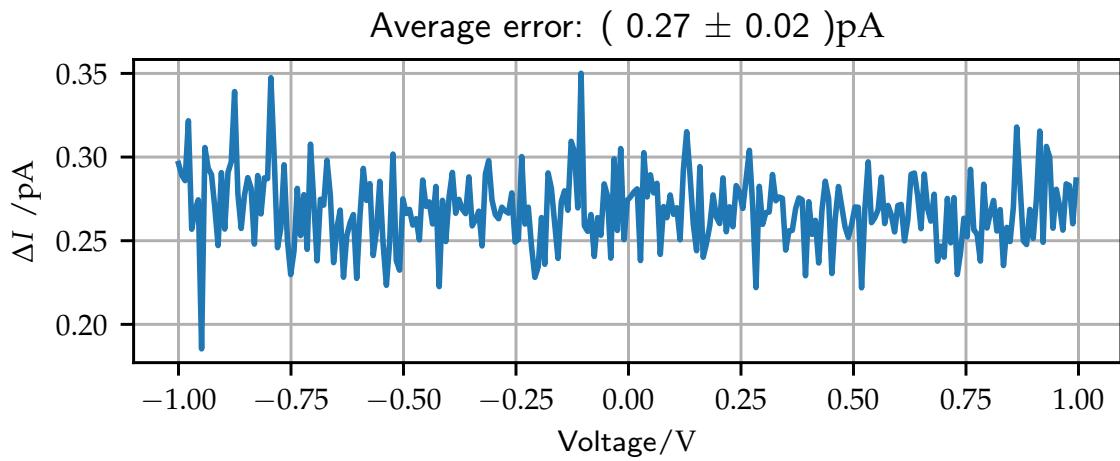


Figure 5.14: Calibration measurement done with the developed prototype and an elongated measurement time for the Keithley.

Parameter	9 V block [58]	Samsung NR18650-25R [59]
Nominal Voltage	9 V	3.6 V
Capacity	950 mA h	2 500 mA h
Benefit	Cheap and easily replaceable.	Rechargeable!
Downside	Has to be replaced regularly.	Requires protection circuit.

Table 5.6: Comparison of a 9 volt-block battery with a lithium-based accumulator.

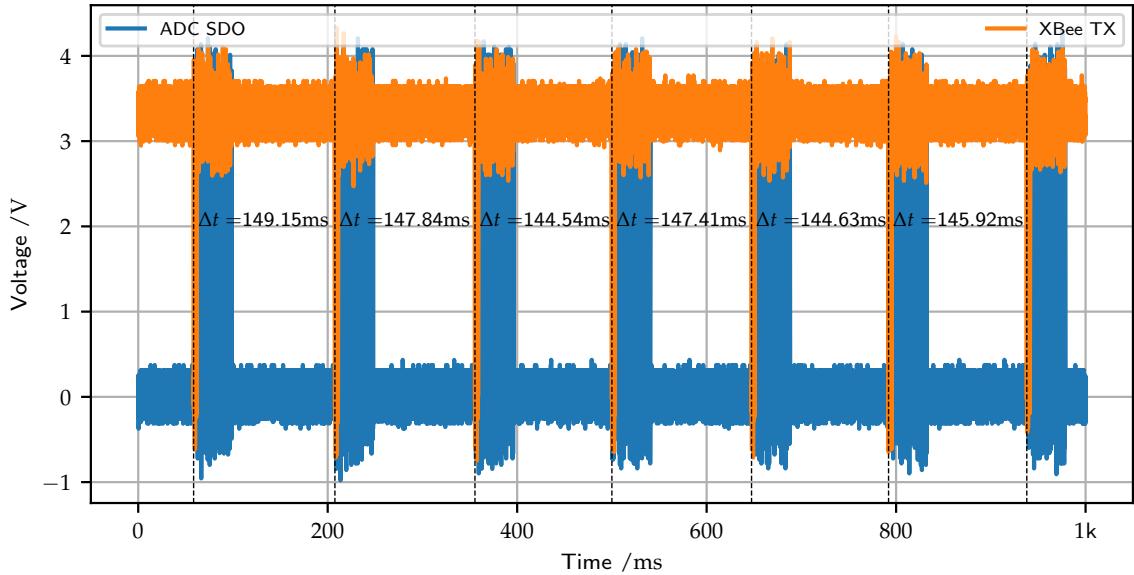


Figure 5.15: Recording of the XBee TX pin and the ADC SDO pin. This measurement allows an estimation of the length of a duty cycle of the pAM.

allows a significantly larger runtime, as opposed to common 9 V blocks. A comparison of both devices is shown in table 5.6. Another option is, to update the photovoltaic based power supply, developed by [5], to allow higher loads and an improved temperature performance.

#### 5.4.2 Duty Cycle

To determine the readout time of the pAM, the ADC and XBee operation was monitored, as shown in figure 5.15. On average between two transmission processes ( $146.6 \pm 1.7$ ) ms pass, resulting in a readout frequency of  $\approx 7$  Hz. From figure 5.15, the run time for different processes can be estimated. A conversion of 128 ADC values takes approximately 36 ms. The time between a transmission-block and a readout-block is too small to be resolved. The time between ADC conversion and conversion is around 100 ms, taking up most of the duty cycle. Between an ADC conversion and sending, the mean and standard deviation of the measured values is calculated, taking up most of the time. Hence, the device readout frequency can be increased by a factor of 2 to 3, if the calculation of the mean and standard deviation is moved to the readout computer. This comes at the cost of more XBee run time, resulting in a higher power consumption.

## 5.4 Measurement Results

---

Overall the prototype performed very well. Showing a high precision of 20 fA on its readings, no visible non-linear effects and a faster 7 Hz conversion with a reasonable power consumption. The limited precision of the reference did not allow further investigation on precision and accuracy.

For further evaluation of the concept, alike investigation of temperature influence, a more robust set of prototypes was designed and ordered. Changes to the design, PCB layout, and schematics can be found in the appendix [A.4](#). However, due to shortages on the manufacturer side, the necessary Op-Amp is temporary not available.



# CHAPTER 6

---

## Conclusion and Outlook

---

The goal of this thesis was to further improve the pAM developed at the TU München. Therefore, an analysis of the design was conducted, resulting in 4 main topics to handle:

1. An upgrade of the firmware, to make it compatible with modern compilers.
2. An improvement of the front-end, to eliminate the internal non-linearity.
3. Setup of a reliable OVP.
4. Enabling a faster readout to achieve a better time resolution.

In chapter 4, the necessary changes to the firmware are introduced. The process was used to implement a bidirectional communication, allowing more flexible use of the device.

In the next step, the source of the observed non-linearity was investigated in chapter 4.2, leading to the need for a new OVP circuit. An approach using diode clamping was suggested in chapter 4.3. Tests proved the approach to be suitable for implementation in the pAM. Unfortunately, an easy insertion of the suggested OVP was not possible without changes to the front-end part of the PCB. As the layout of a new revision was necessary, the front-end concept was changed over completely. The shunt resistor based input stage was replaced by a TIA setup. To allow faster measurements, an ADC with a higher sampling rate was implemented. Layout and performance of the new front-end are discussed in chapter 5. The new design achieved an estimated precision of 20 fA at a readout rate of 7 Hz. The bandwidth is limited to 20 Hz. The power consumption increased to a moderate 7 mA, with frequent spikes of up to 50 mA.

In the future, the readout rate could be further increased by reducing the number of values that are combined in one measurement or by moving the time-consuming calculations to the computer side. Further increase of the readout rate, would require a faster  $\mu$ C. If a higher bandwidth is desired, the front-end could be expanded, using a broadband TIA with lower amplification followed by a voltage amplification stage. To accommodate for the increased power consumption, a new power supply should be set up, either utilizing an improvement of the photovoltaic powering developed by [5], or based on batteries with a higher capacity.

For further investigation of the behaviour of the new design, a second iteration of the prototype was developed. The setup could not be finished, as the ADA4530 Op-Amp used in the new front-end,

## Chapter 6 Conclusion and Outlook

---

could not be delivered in time by the manufacturer. To ensure the reliability of the design, long term stability tests are necessary, as well as the study of possible temperature influences. A comparison with similar devices developed by [51] is supposed to be run as well. To get a good measurement of the achievable precision and accuracy, a new calibration reference is needed. The Keithley, that was used up to now, could not perform measurements with the necessary precision.

## 6.1 List Of Acronyms

<b>pAM</b>	picoamperemeter	<b>MSB</b>	most significant bit
<b>MWPC</b>	multi-wire proportional counter	<b>IC</b>	Integrated Circuit
<b>TPC</b>	Time Projection Chamber	<b>I/O</b>	input/output
<b>ADC</b>	Analog-to-Digital Converter	<b>IE</b>	interrupt enable
<b>ALICE</b>	<i>A Large Ion Collider Experiment</i>	<b>GIE</b>	general interrupt enable
<b>Op-Amp</b>	Operational Amplifier	<b>IFG</b>	interrupt flag
<b>GEM</b>	Gas Electron Multiplier	<b>ISR</b>	interrupt service routine
<b>SPICE</b>	Simulation Program with Integrated Circuit Emphasis	<b>SAR</b>	Successive-Approximation-Register
<b>pA</b>	picoampere	<b>DAC</b>	Digital-to-Analog Converter
<b>μC</b>	Microcontroller	<b>TI</b>	Texas Instruments
<b>MSP</b>	MSP430f169	<b>FR-4</b>	Flame Retardant 4
<b>PAMOS</b>	PicoAmpereMeter OperatingSystem	<b>Keithley</b>	Keithley 6517b electrometer
<b>CalibAna</b>	CalibAna	<b>PCB</b>	printed circuit board
<b>XBee</b>	XBee	<b>API</b>	Application Programming Interface
<b>C++</b>	C++	<b>USART</b>	Universal Synchronous and Asynchronous Serial Receiver and Transmitter
<b>TIA</b>	transimpedance amplifier	<b>LHC</b>	<i>Large Hadron Collider</i>
<b>I<sup>2</sup>C</b>	inter-integrated Circuit	<b>CERN</b>	European Organization for Nuclear Research
<b>OVP</b>	over-voltage protection	<b>ATLAS</b>	<i>A Toroidal LHC Apparatus</i>
<b>SPI</b>	Serial Peripheral Interface	<b>GBW</b>	Gain-Bandwidth-Product
<b>ESD</b>	Electrostatic Discharge		
<b>LSB</b>	least significant bit		



## Bibliography

---

- [1] G. Apollinari et al., *High Luminosity Large Hadron Collider HL-LHC*, (2017), eprint: [arXiv:1705.08830](https://arxiv.org/abs/1705.08830).
- [2] ALICE Collaboration, *Technical Design Report for the Upgrade of the ALICE Inner Tracking System*, tech. rep. CERN-LHCC-2013-024. ALICE-TDR-017, 2013, URL: <https://cds.cern.ch/record/1625842>.
- [3] A. Bugl, *Weiterentwicklung und Charakterisierung von Hochspannungs-Picoamperemetern*, TU München, Bachelorthesis, 2013.
- [4] J. Rödel, *Entwicklung eines voll automatisierten Kalibrationsprozesses für Picoampèremeter*, University Bonn, Bachelorthesis, 2016.
- [5] T. Rudolph, *Entwicklung einer erdfreien Photovoltaik-Stromversorgung für Hochspannungs-Picoamperemeter*, Bachelorthesis, University Bonn, 2019.
- [6] F. Sauli, *Gaseous Radiation Detectors: Fundamentals and Applications*, Cambridge Monographs on Particle Physics, Nuclear Physics and Cosmology, Cambridge Univ. Press, 2014, ISBN: 1107043018.
- [7] J. Marx and D. Nygren, *The Time Projection Chamber*, Physics Today **31** (1978) 46.
- [8] B. Ketzer and N. Wermes, *Physics of Particle Detectors*, Lecture, University Bonn, 2018.
- [9] F. Sauli, *The gas electron multiplier (GEM): Operating principles and applications*, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **805** (2016) 2 , Special Issue in memory of Glenn F. Knoll, ISSN: 0168-9002, URL: <http://www.sciencedirect.com/science/article/pii/S0168900215008980>.
- [10] M. Berger et al., *A large ungated TPC with GEM amplification*, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment (2017).
- [11] J. Ottnad, *Optimierung der GEM-basierten Verstärkungsstufe einer TPC für das CBELSA/TAPS-Experiment*, PhD. Thesis: University Bonn, unpublished.
- [12] M. Ball, K. Eckstein and T. Gunji, *Ion backflow studies for the ALICE TPC upgrade with GEMs*, Journal of Instrumentation **9** (2014) C04025, URL: <https://doi.org/10.1088%2F1748-0221%2F9%2F04%2Fc04025>.
- [13] P. Hauer et al., *Measurements of the Charging-Up Effect in Gas Electron Multipliers*, (2019), eprint: [arXiv:1911.01848](https://arxiv.org/abs/1911.01848).

## Bibliography

---

- [14] P. Horowitz and W. Hill, *The art of electronics*, 3. ed., reprint. with corr., Cambridge Univ. Press, 2015, isbn: 9780521809269, 0521809266.
- [15] J. M. Fiore, *Operational Amplifiers and Linear Integrated Circuits: Theory and Application*, 2018.
- [16] W. Kester, *Which ADC Architecture Is Right for Your Application?*, (Last Visited: 09.06.2020), Analog Devices, URL: <https://www.analog.com/en/analog-dialogue/articles/the-right-adc-architecture.html>.
- [17] *Choose the right A/D converter for your application*, (Last Visited: 09.06.2020), Texas Instruments, URL: <https://www.ti.com/europe/downloads/Choose%20the%20right%20data%20converter%20for%20your%20application.pdf>.
- [18] *Analog to Digital Converter (ADC) Selection Guide*, (Last Visited: 09.06.2020), arrow, URL: <https://www.arrow.com/en/research-and-events/articles/analog-to-digital-converter-selection-guide>.
- [19] Katanzag, *Grundsätzliches Blockbild eines DeltaSigma-Modulators erster Ordnung mit digitalem Ausgang*, (Last Visited: 09.06.2020), URL: <https://commons.wikimedia.org/w/index.php?curid=1233699>.
- [20] *Understanding Pipelined ADCs*, (Last Visited: 09.06.2020), maxim integrated, URL: <https://www.maximintegrated.com/en/design/technical-documents/tutorials/1/1023.html>.
- [21] W. Flye, *Successive Approximation ADC Block Diagram*, (Last Visited: 09.06.2020), maxim integrated, URL: [https://upload.wikimedia.org/wikipedia/commons/6/61/SA\\_ADC\\_block\\_diagram.png](https://upload.wikimedia.org/wikipedia/commons/6/61/SA_ADC_block_diagram.png).
- [22] *Noise Analysis in Operational Amplifier Circuits*, Application Report, 2007, URL: <http://www.ti.com/lit/an/slva043b/slva043b.pdf>.
- [23] R. J. v. d. Plassche, *Integrated Analog-To-Digital and Digital-To-Analog Converters* -, Springer Science & Business Media, 2012, isbn: 978-1-461-52748-0.
- [24] *Low Level Measurements Handbook, Precision DC Current, Voltage, and Resistance Measurements*, 7th ed., Keithley, 2020.
- [25] H. Ibach and H. Lüth, *Festkörperphysik: Einführung in die Grundlagen*, Fünfte, erweiterte und aktualisierte Auflage., Springer-Lehrbuch, Springer Berlin Heidelberg, 1999, isbn: 3540660747, 9783540660743.
- [26] S. McGrath, *Cooper and Styrofoam on Flickr*, (Last Visited: 09.06.2020), URL: <https://commons.wikimedia.org/w/index.php?curid=60287175>.
- [27] D. J. Lacks and R. M. Sankaran, *Contact electrification of insulating materials*, *Journal of Physics D: Applied Physics* **44** (2011) 453001, URL: <https://doi.org/10.1088%2F0022-3727%2F44%2F45%2F453001>.
- [28] *ADA4530 - Femtoampere Input Bias Current Electrometer Amplifier*, B, Analog Devices, 2017.
- [29] *LTspice*, (Last Visited: 09.06.2020), Analog Digital, URL: <https://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html>.

- 
- [30] A. Vladimirescu, *The Spice book*, J. Wiley, 1994, ISBN: 0471609269, 9780471609261.
  - [31] H. Kaden, *Wirbelströme und Schirmung in der Nachrichtentechnik*, Springer Berlin Heidelberg, 2006, ISBN: 978-3-540-32570-3.
  - [32] *Datenblatt Standard FR4 TG135*,  
URL: [https://www.wedirekt.de/web/mandators/media/16\\_wedirekt/spezifikationen/leiterplatten\\_daten\\_downloads/FR4TG135\\_DE.pdf](https://www.wedirekt.de/web/mandators/media/16_wedirekt/spezifikationen/leiterplatten_daten_downloads/FR4TG135_DE.pdf).
  - [33] *RO400 Series, High frequency circuit materials*, URL:  
<https://rogerscorp.com/-/media/project/rogerscorp/documents/advanced-connectivity-solutions/english/data-sheets/ro4000-laminates-ro4003c-and-ro4350b---data-sheet.pdf>.
  - [34] P. Grohe, *Design femtoampere circuits with low leakage, part 1*, (Last Visited: 09.06.2020),  
URL: <https://www.edn.com/design-femtoampere-circuits-with-low-leakage-part-one/>.
  - [35] J. Bishop, *Cleaning PCBs*, (Last Visited: 09.06.2020), 2016,  
URL: <https://www.digikey.de/en/maker/blogs/cleaning-pcbs>.
  - [36] *Dirty PCB - The Ultimate Guide To Make Your PCB Clean And Durable*,  
(Last Visited: 09.06.2020), WellPCB,  
URL: <https://www.wellpcb.com/special/dirty-pcb.html>.
  - [37] G. Gridling and B. Weiss, *Introduction to Microcontrollers*, 1.4, Vienna University of Technology, Institute of Computer Engineering, Embedded Computing Systems Group, 2007,  
URL: <https://ti.tuwien.ac.at/ecs/teaching/courses/mclu/theory-material/Microcontroller.pdf>.
  - [38] *MSP430x1xx Family User's Guide*, Texas Instruments, 2006,  
URL: <http://www.ti.com/lit/ug/slau049f/slau049f.pdf?ts=1591195741417>.
  - [39] *I2C-bus specification and user manual*, 6th, NXP Semiconductors,  
URL: <https://www.nxp.com/docs/en/user-guide/UM10204.pdf>.
  - [40] *SPI Block Guide*, 3rd, Motorola, NXP Semiconductors,  
URL: [https://www.nxp.com/files/static/microcontrollers/doc/ref\\_manual/S12SPIV4.pdf](https://www.nxp.com/files/static/microcontrollers/doc/ref_manual/S12SPIV4.pdf).
  - [41] *XBee/XBee-PRO OEM RF Modules, Product Manual v1.xAx - 802.15.4 Protocol*, MaxStream,  
URL:  
<https://www.sparkfun.com/datasheets/Wireless/Zigbee/XBee-Manual.pdf>.
  - [42] *XBee S2C – RF Module*, (Last Visited: 16.05.2020), URL:  
<https://components101.com/wireless/xbee-s2c-module-pinout-datasheet>.
  - [43] *Next Generation Configuration Platform for XBee/RF Solutions*, (Last Visited: 09.06.2020),  
URL: <https://www.digi.com/products/embedded-systems/digi-xbee/digi-xbee-tools/xctu>.
  - [44] *ADA549 - Ultralow Input Bias Current Amplifier*, K, Analog Devices, 2013.
  - [45] *AD7790, Low Power, 16-Bit Buffered Sigma-Delta ADC*, Datasheet, A, Datasheet, version A, Analog Digital, URL: <https://www.analog.com/media/en/technical-documentation/data-sheets/AD7790.pdf>.

## Bibliography

---

- [46] *AG Ketzer Wiki*, URL: <https://agketzer.hiskp.uni-bonn.de/intern/wiki/index.php?title=PA-Meters>.
- [47] attie, *A C/C++ library to aid the use of Digi XBee radios in API mode*, (Last Visited: 09.06.2020), URL: <https://github.com/attie/libxbee3>.
- [48] *Linux Programmer's Manual*, (Last Visited: 09.06.2020), 2019, URL: <http://man7.org/linux/man-pages/man3/termios.3.html>.
- [49] T. Schärer, *Überspannungsschutz von empfindlichen Verstärkereingängen*, (Last Visited: 09.06.2020), URL: <https://www.elektronik-kompendium.de/public/schaerer/ovprot.htm>.
- [50] A. Mortuza et al., *Pico-current Measurement Challenges and Remedies: A Review*, *Universal Journal of Engineering Science* **5** (2017) 57.
- [51] A. Utrobicic et al., *A floating multi-channel picoammeter for micropattern gaseous detector current monitoring*, *Nuclear Instruments and Methods in Physics Research Section A Accelerators Spectrometers Detectors and Associated Equipment* **801** (2015) 21.
- [52] *Difet Electrometer-Grade OPERATIONAL AMPLIFIER*, Texas Instruments, 1995, URL: <https://www.ti.com/lit/ds/symlink/opa128.pdf?ts=1590309045570>.
- [53] *LMC6442 Dual Micropower Rail-to-Rail Output Single Supply Operational Amplifier*, Texas Instruments, 2013, URL: <https://www.ti.com/lit/ds/symlink/lmc6442.pdf?ts=1590309280339>.
- [54] *LTC2327-16 datasheet*, B, Linear Technology, 2017, URL: <https://www.analog.com/media/en/technical-documentation/data-sheets/232716fa.pdf>.
- [55] T. Wang and B. Erham, *Compensate Transimpedance Amplifiers Intuitively*, Application Report, 1993, URL: <http://www.ti.com/lit/an/sboa055a/sboa055a.pdf?ts=1591195305628>.
- [56] *ADT7410,  $\pm 0.5^\circ\text{C}$  Accurate, 16-Bit DigitalI2C Temperature Sensor*, C, Datasheet, Analog Devices, URL: <https://www.analog.com/media/en/technical-documentation/data-sheets/ADT7410.pdf>.
- [57] *6517B Electrometer/High Resistance Meter*, Keithley, Keithley, URL: <https://docs.rs-online.com/ba67/0900766b810c7ddb.pdf>.
- [58] *Industrial by Duracell 6LF22, Alkaline-Manganese Dioxide Battery*, Datasheet, Duracell, URL: <https://docs.rs-online.com/2527/0900766b8127fc88.pdf>.
- [59] *Introduction of INR18650-25R*, Samsung, 2013, URL: <https://www.powerstream.com/p/INR18650-25R-datasheet.pdf>.
- [60] J. Rödel, *Entwicklung eines voll automatisierten Kalibrationsprozesses für Picoampèremeter*, University Bonn, Bachelor talk, 2016.

# APPENDIX A

---

## Appendix

---

### A.1 Picoamperemeter Revision Three

The revision 3 schematics can be found in figures [A.1](#) and [A.2](#). The PCB layout is shown in figure [A.3](#).

### A.2 The Calibration Station

The calibration process for the pAM was developed by [4]. However, during work on this thesis, the station underwent a complete hardware failure. Restoring the whole setup was therefore necessary. The principle of the station is shown in figure [A.4](#).

Figure [A.5](#) shows the chimney of the calibration station. The funnel section was previously mounted to the metal chimney using duct tape. As duct tape did not provide a stable connection, it was replaced by metal rods.

The joint between funnel and the cooler section was held together by hot glue. During operation this joint broke, leading to a collapse of the cooler chimney. The hot glue joint was, therefore, replaced with Araldite, which should result in more long-term stability. The collapse also damaged the plastic fan mounting of the top fan. Due to time constraints, the replacement could only be built from metal, resulting in a small heat-bridge between the top and bottom side of the Peltier elements. If the heat-bridge proves to be a problem, it should be replaced in the future.

The collapse must have caused a power spike in the circuitry, for unknown reasons. After the collapse, the relays for switching the Peltier elements, and the Peltier elements themselves were destroyed and needed to be replaced. The power board that connects the Arduino with the external setup was destroyed as well. A new board was designed. In the process, an electromagnet was added to the setup. The magnet can be controlled with a switch or from the Arduino, allowing to switch the pAM under test using the CalibAna software<sup>1</sup>. Figure [A.6](#) shows the circuitry that is controlled by the Arduino.

---

<sup>1</sup> An interesting remark: To support the electromagnet, the Companduino software, controlling the Arduino, was modified. During the modification process it was found, that elongating the software any further would cause it to stop working. Probably the memory that is left on the Arduino is necessary to perform calculations during runtime.

## Appendix A Appendix

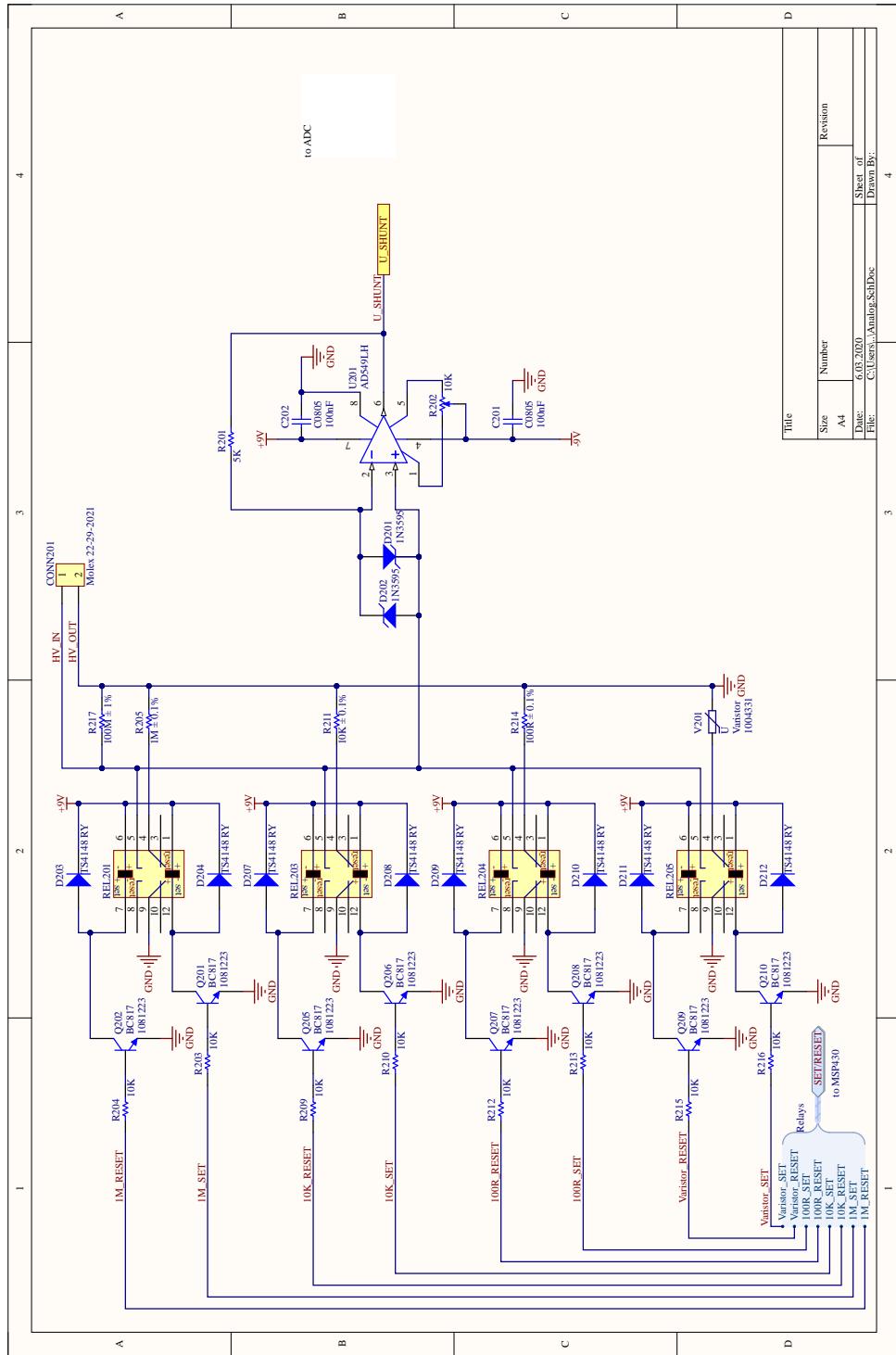


Figure A.1: Schematics for the back-end of the revision 3 pAM.

## A.2 The Calibration Station

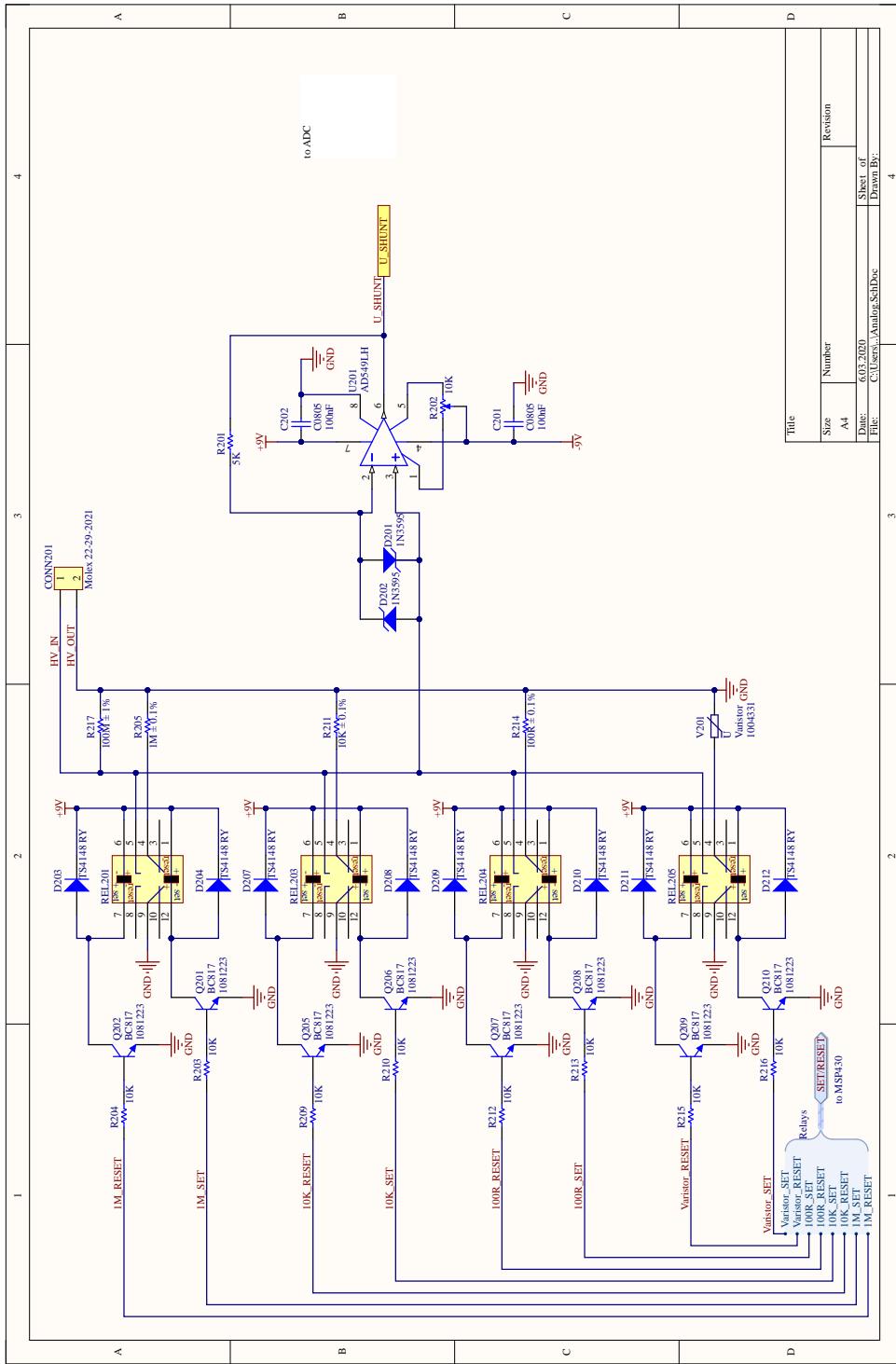


Figure A.2: Schematics for the front-end of the revision 3 pAM. The attachment, with the SA5.0CA diode and the 20 k $\Omega$  resistor, is not depicted here as it is not part of the original design.

## Appendix A Appendix

---

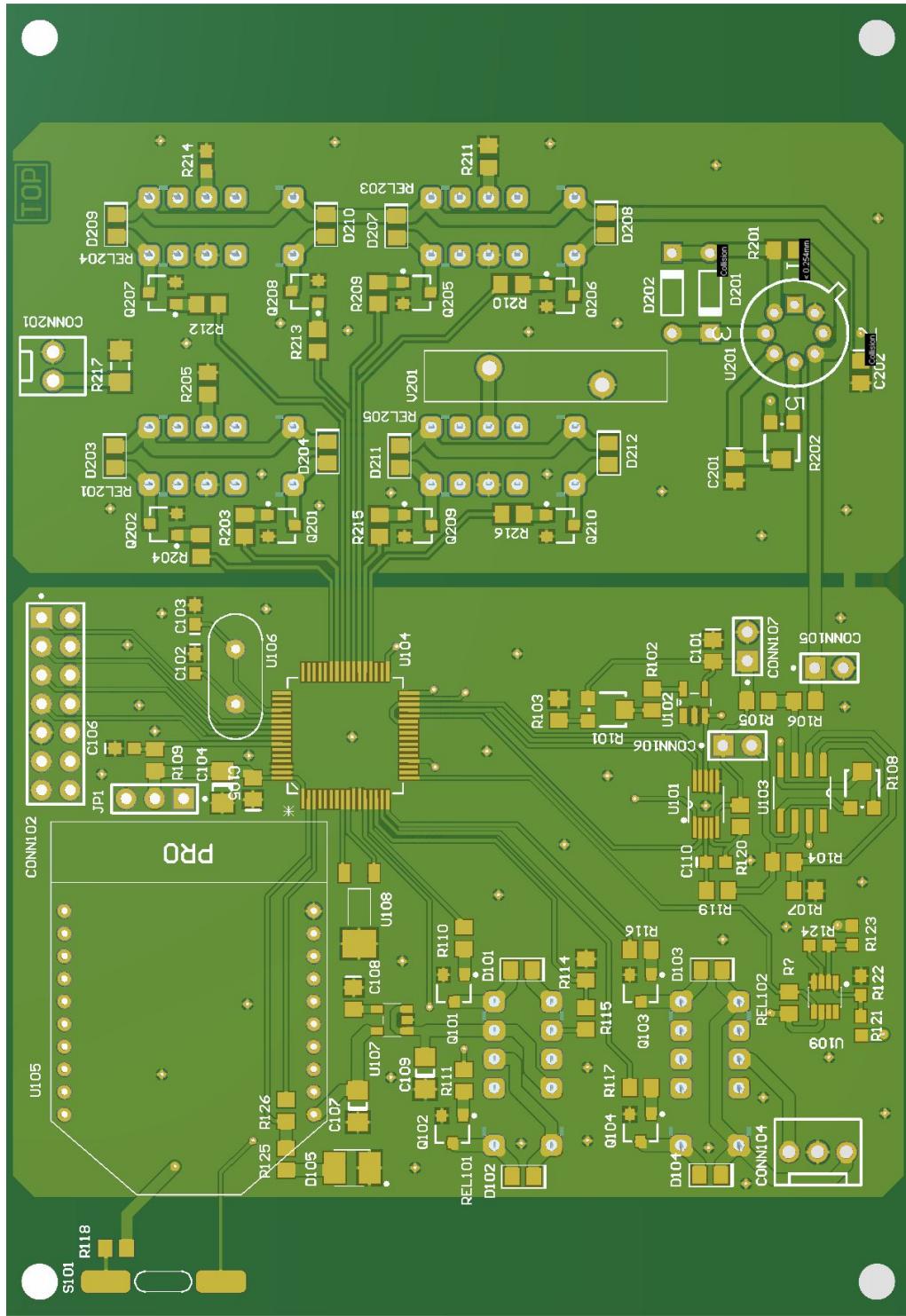


Figure A.3: PCB layout of the revision 3 pAM.

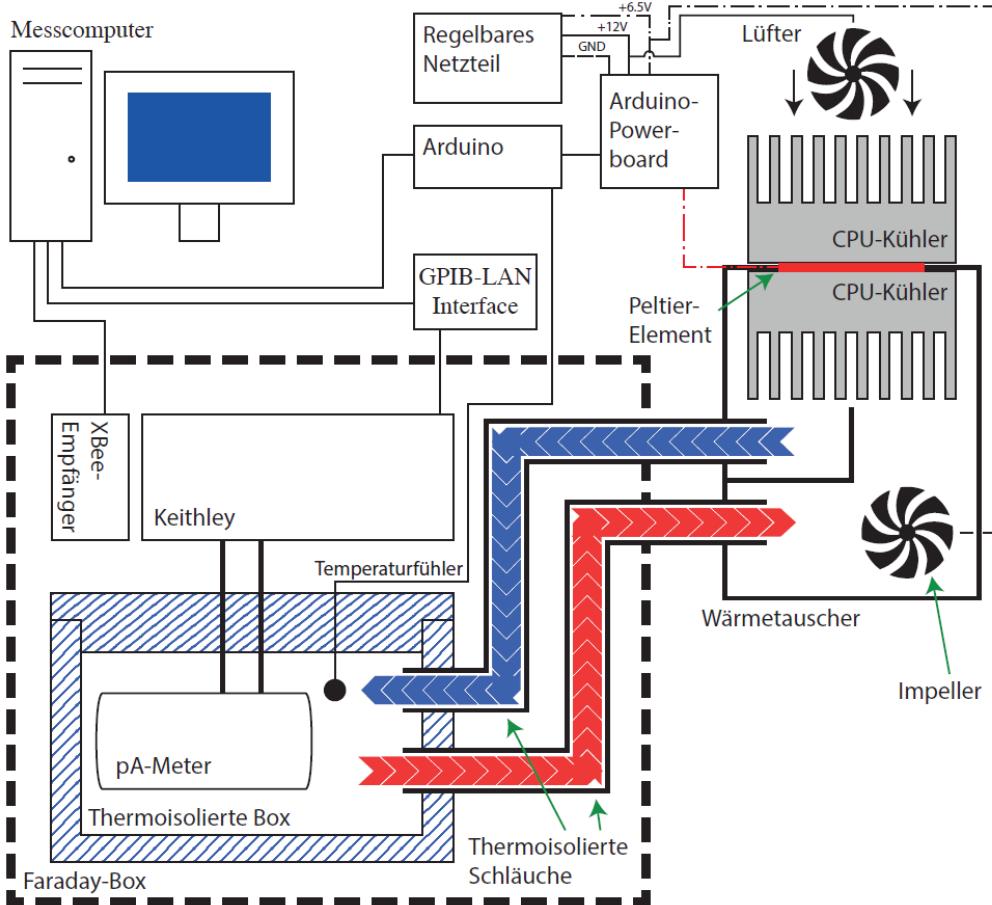


Figure A.4: Principle of the calibration station as designed by [4]. From [60].

### A.3 Prototype Iteration One Schematics

For the back-end schematics of the prototype, see figure A.7. The front-end schematics are displayed in figure A.8.

### A.4 Prototype Iteration Two

The second iteration of a prototype brought some slight changes. The board was recombined, housing back- and front-end on a single PCB. The feedback resistors were replaced by lower temperature drift resistors, decreasing the temperature drift down to 50 ppm/°C. The mode signal LED was replaced by three different coloured LEDs with a decreased power consumption. The original LED drew 20 mA, each of the coloured LEDs only draws 2 mA. The colour also eliminates the need for repeated blinking, further lowering the power requirement. The port 5 of the µC is mapped to a header array, to allow attaching a screen to the device. In the front-end mounting holes are implemented, to allow an additional shielding of the input stage. Schematics for the design can be found in figures A.9 and A.10; the PCB layout is shown in figure A.11.

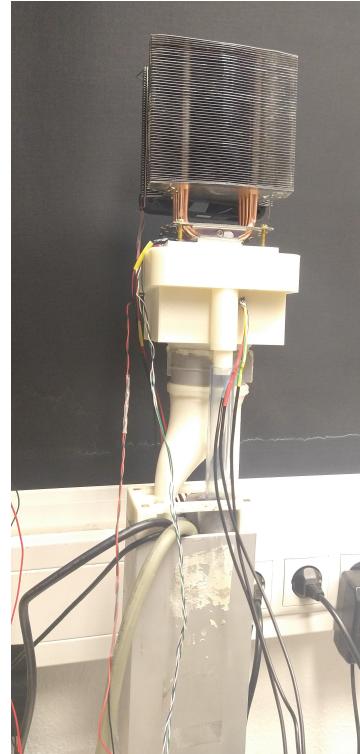


Figure A.5: The chimney section of the Calibrations station. On the bottom the metal chimney to guide cables and the ventilation hose into the faraday box; above this, the funnel connecting the hose to the cooler section; on top the fan for cooling the Peltier elements.

## A.5 List Of Acronyms

<b>pAM</b>	picoamperemeter	<b>MSP</b>	MSP430f169
<b>MWPC</b>	multi-wire proportional counter	<b>PAMOS</b>	PicoAmpereMeter OperatingSystem
<b>TPC</b>	Time Projection Chamber	<b>CalibAna</b>	CalibAna
<b>ADC</b>	Analog-to-Digital Converter	<b>XBee</b>	XBee
<b>ALICE</b>	<i>A Large Ion Collider Experiment</i>	<b>C++</b>	C++
<b>Op-Amp</b>	Operational Amplifier	<b>TIA</b>	transimpedance amplifier
<b>GEM</b>	Gas Electron Multiplier	<b>I<sup>2</sup>C</b>	inter-integrated Circuit
<b>SPICE</b>	Simulation Program with Integrated Circuit Emphasis	<b>OVP</b>	over-voltage protection
<b>pA</b>	picoampere	<b>SPI</b>	Serial Peripheral Interface
<b>μC</b>	Microcontroller	<b>ESD</b>	Electrostatic Discharge
		<b>LSB</b>	least significant bit

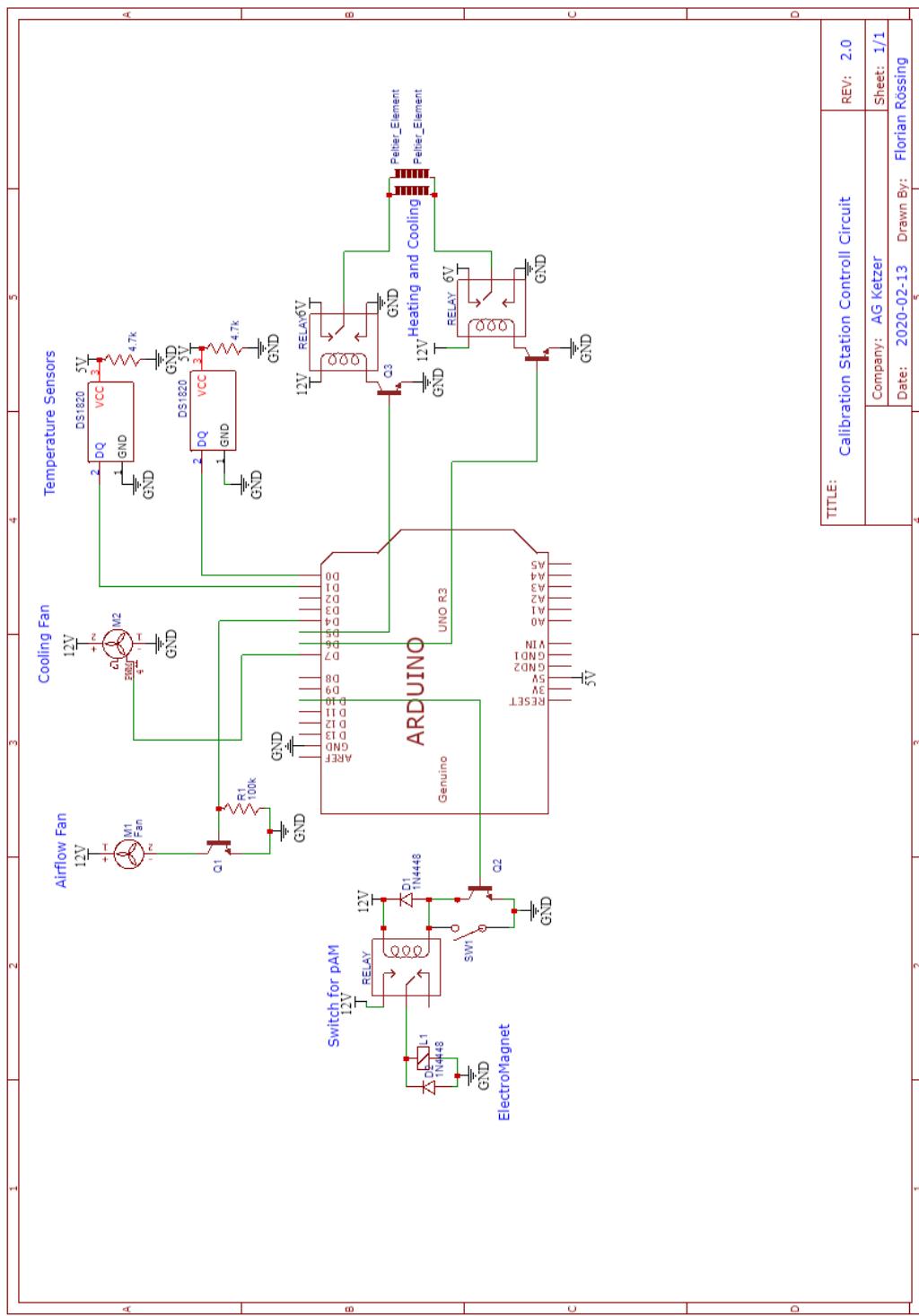


Figure A.6: Schematic of the Calibration station after restoration.

## Appendix A Appendix

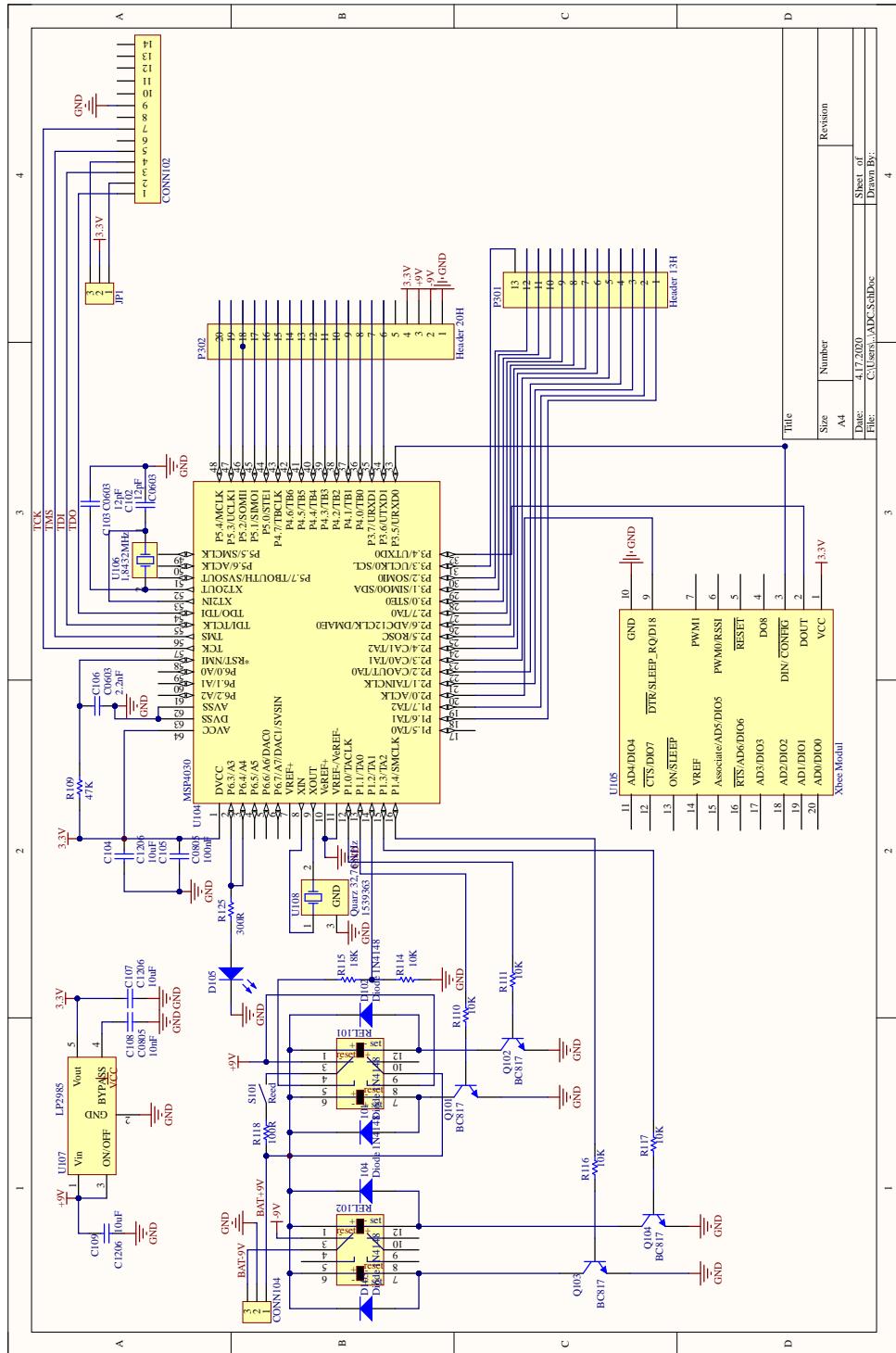


Figure A.7: Schematics for the back-end prototype described in section 5.3. The corresponding PCB is shown in figure 5.9.

## A.5 List Of Acronyms

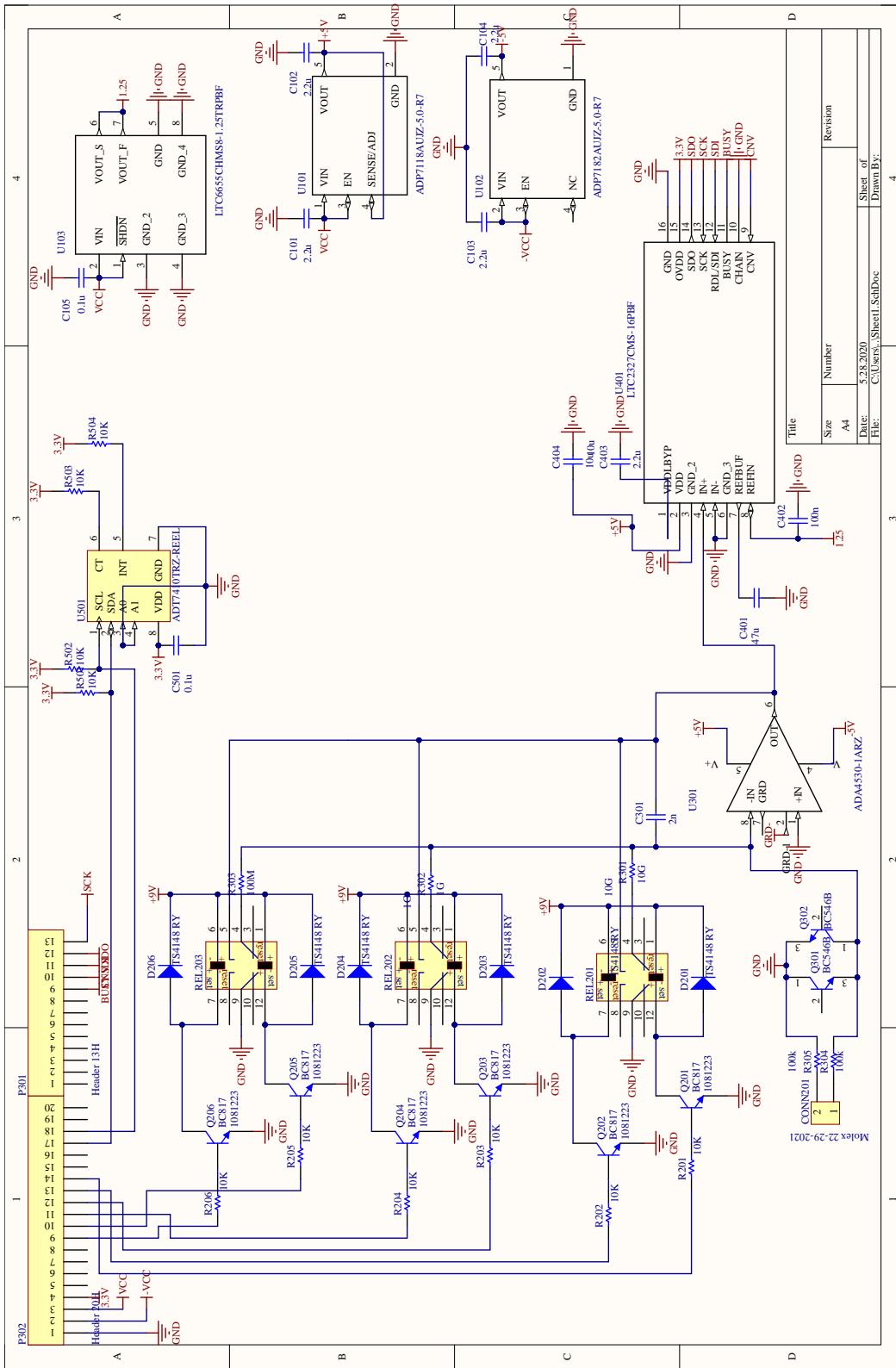


Figure A.8: Schematics for the front-end prototype described in section 5.3. The corresponding PCB is shown in figure 5.10.

## Appendix A Appendix

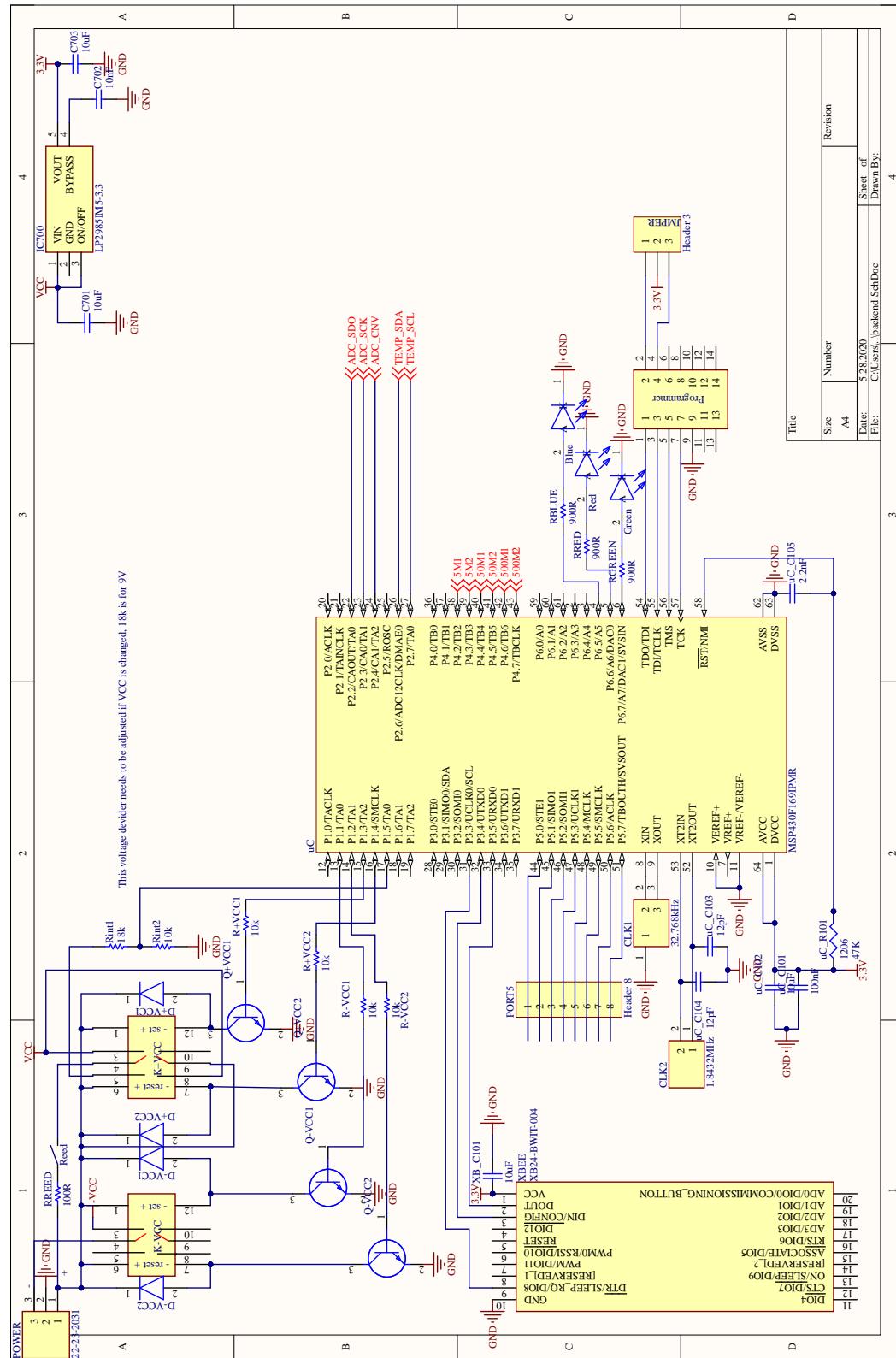


Figure A.9: Schematics for the front-end of the second prototype iteration.

## A.5 List Of Acronyms

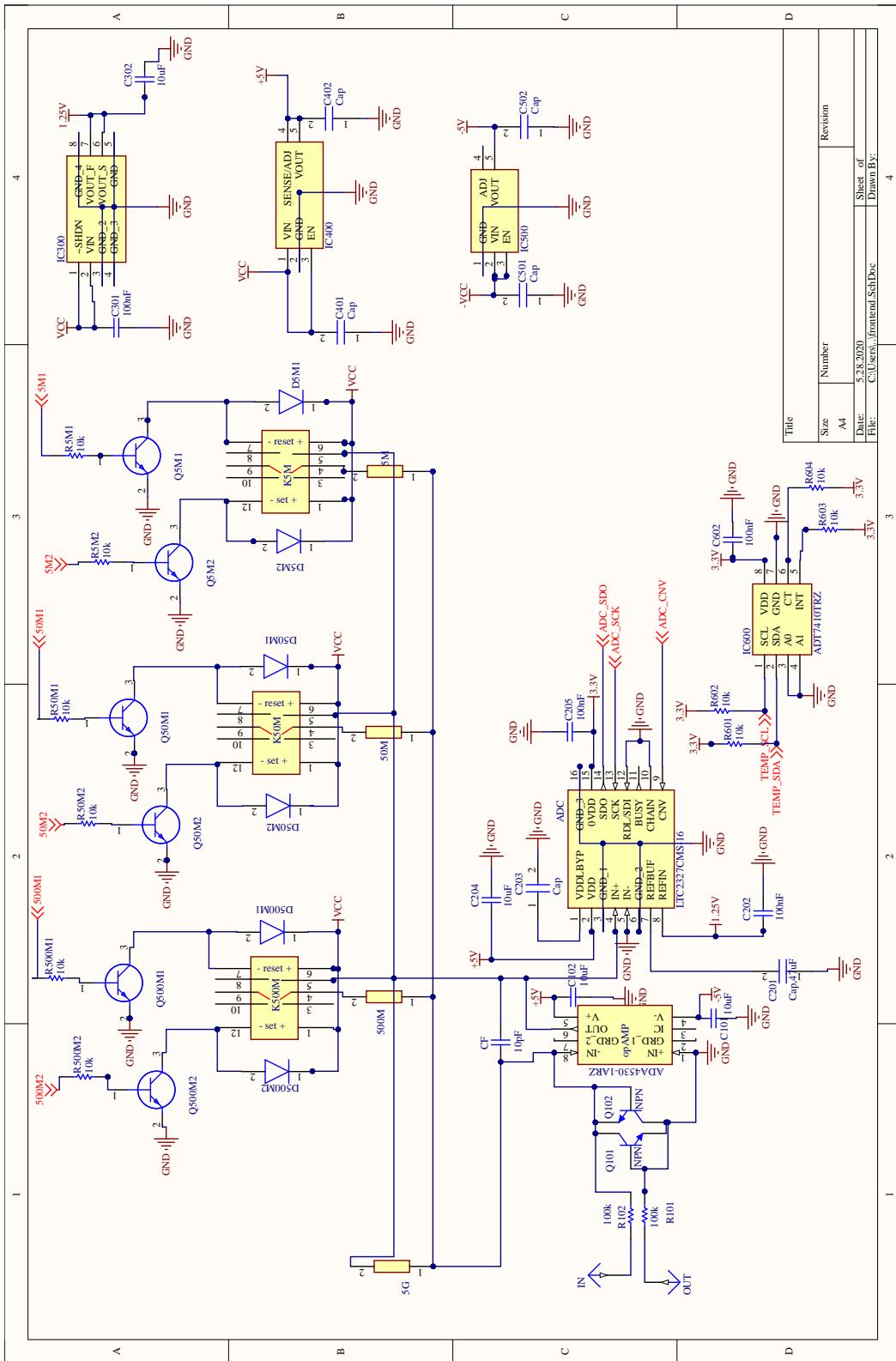


Figure A.10: Schematics for the back-end of the second prototype iteration.

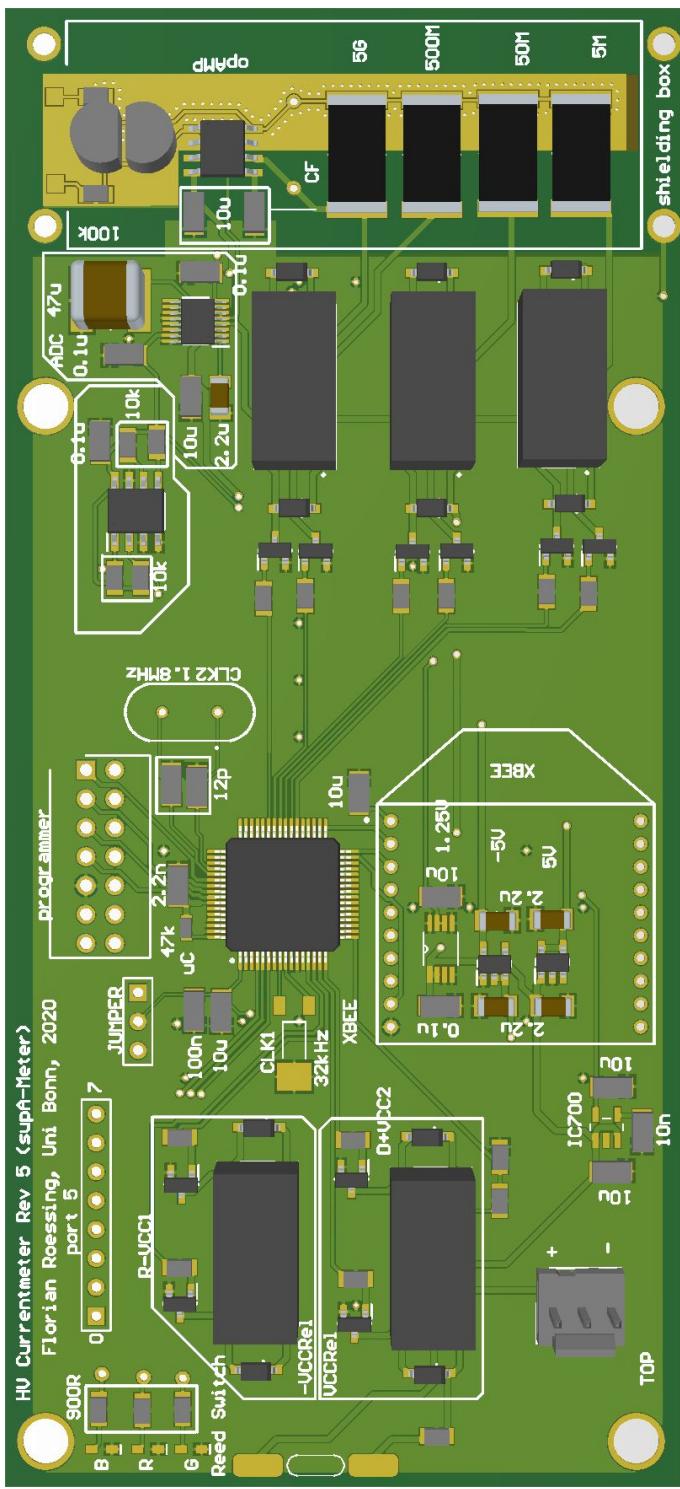


Figure A.11: PCB layout for the second prototype iteration.

---

## A.5 List Of Acronyms

<b>MSB</b>	most significant bit	<b>Keithley</b>	Keithley 6517b electrometer
<b>IC</b>	Integrated Circuit	<b>PCB</b>	printed circuit board
<b>I/O</b>	input/output	<b>API</b>	Application Programming Interface
<b>IE</b>	interrupt enable	<b>USART</b>	Universal Synchronous and Asynchronous Serial Receiver and Transmitter
<b>GIE</b>	general interrupt enable	<b>LHC</b>	<i>Large Hadron Collider</i>
<b>IFG</b>	interrupt flag	<b>CERN</b>	European Organization for Nuclear Research
<b>ISR</b>	interrupt service routine	<b>ATLAS</b>	<i>A Toroidal LHC Apparatus</i>
<b>SAR</b>	Successive-Approximation-Register	<b>GBW</b>	Gain-Bandwidth-Product
<b>DAC</b>	Digital-to-Analog Converter		
<b>TI</b>	Texas Instruments		
<b>FR-4</b>	Flame Retardant 4		



# List of Figures

---

2.1	Principle setup of a TPC. From [8].	4
2.2	Amplification and readout stage of a TPC with a MWPC based amplification stage. From [8].	4
2.3	Electron-microscope image of a GEM-foil. Dimensions are given on the image. From [9].	5
2.4	A triple GEM stack with the drift, transfer and induction fields. From [6].	5
2.5	Left: An incoming electron enters a GEM hole and ionizes the gas and creates electron (blue) - ion (red) pairs. Right: Electrons are extracted, ions drift back and end up on the copper coating due to their low diffusion and the low drift field. From [10].	6
2.6	Measurements of current from the pad plane of a GEM detector done with a pAM. From [13].	7
2.7	Symbol used to depict an operational amplifier in circuit schematics.	7
2.8	Open-loop gain $A_{OL}$ of an operational amplifier rolling-off at 10 Hz. The closed-loop gain $A_{CL}$ is affected by the roll-off at higher frequencies, where it is comparable to the closed-loop gain. From [15], modified.	9
2.9	Principle of a first-order $\Sigma\Delta$ ADC. From [19].	9
2.10	A 12-bit pipelined ADC. From [20], modified.	10
2.11	Block diagramm of an N-bit SAR ADC. From [21].	10
2.12	Model to ease the noise calculation for Op-Amp circuits. With $i_n$ denoting the current noise density and $v_n$ the voltage noise density. From [22], modified.	12
2.13	The Triboelectric Effect depicted by a cat. From [26].	13
2.14	Piezoelectric current generated through mechanical stress on a plastic socket. From [24].	14
2.15	Left: Guard ring surrounding a high impedance IC input trace. For best performance, the guard trace should be hooked up to a driving circuit. Right: Implementation of a guarding structure through the PCB bulk. For connecting the guard ring and the guard plane on the bottom side, vias are implemented. Solder mask between trace and guard is removed to reduce leakage paths.	17
2.16	Pinout of the XBee radio modules. From [42].	20
2.17	Important XBee API frames. From [41].	21
3.1	Implementation of mode switching in the pAM revision 3 and 4.	24
3.2	Current measurement circuit using an operational amplifier and a shunt resistor.	24
3.3	Front-end circuitry used in the revisions 3 and 4. The additional OVP diodes are depicted in figure 3.4(b).	25
3.4	The two expansions that are made to the front-end depicted in figure 3.3.	26
3.5	pAM firmware workflow for firmware versions 3.x.	28

3.6	Datasets as sent out by pAMs running the firmware v3.0 and v4.0. The package contains mean and error of the acquired data, the current operating mode, the SVS voltage output, the time needed for conversion, the time between finishing a conversion and sending data, and the number of remaining packages; which is 0 in this case. . . . .	30
3.7	Datasets as sent out by pAM running the firmware v3.1 and v4.1. The first package contains mean and error of the acquired data, the current operating mode, the SVS voltage output, the time needed for conversion, the time between finishing a conversion and sending data, and the number remaining packages. The next four packages contain 32 ADC readings each, followed by the number of remaining packages. . . . .	30
3.8	Calibration plot produced by CalibAna2. . . . .	31
4.1	The initialisation process for pAM, as implemented in the firmware versions 3.2 and 5.0. . . . .	35
4.2	Calibration measurement for station 16 in lowest and highest sensitivity mode. . . . .	41
4.3	Calibration measurement for a station with removed diodes. . . . .	42
4.4	3 different diode clamping circuits, that could be used as OVP. . . . .	42
4.5	Over-voltage protection using antiparallel diodes. The dark grey area would not be measured by the pAM. The light grey area is the range, that can be digitised by the ADC, and the white area is the range that is calibrated. . . . .	43
4.6	Over-voltage protection using back-to-back diodes. The dark grey area would not be measured by the pAM. The light grey area is the range, that can be digitised by the ADC. The white area represents the calibration range. The upper plot shows voltage on the non-inverting input vs input current and a fit on this data. The fit range is limited to the calibration range. The lower plot shows the deviation of simulation results from the linear fit. . . . .	44
4.7	Over-voltage protection using diode clamping to an external potential. The dark grey area would not be measured by the pAM. The light grey area is the range, that can be digitised by the ADC. The white area represents the calibration range. The upper plot shows voltage on the non-inverting input vs input current and a fit on this data. The fit range is limited to the calibration range. The lower plot shows the deviation of simulation results from the linear fit. . . . .	45
4.8	Equivalent circuit for a diode formed by the base-collector pn-junction of a transistor. . . . .	46
4.9	Measured leakage currents for different diodes. Error bars are too small to be displayed. . . . .	46
4.11	Calibration measurement run with the OVP prototype inserted in the input path. . . . .	48
5.1	Basic principle of a TIA. . . . .	49
5.2	Drift of the ADA4530 offset-voltage drift vs time and temperature [28]. . . . .	51
5.3	OVP diodes implemented in the TIA setup. . . . .	52
5.4	Model as implemented in LTspice for simulating the behaviour of the input stage. The Resistors R2 and R3 are the current limiters for the OVP setup. Models for the BC546B are implemented in the transistors Q1 and Q2. C1 and R1 allow adjusting of shunt capacitance and resistance. C2 and R4 allow adjustment of the feedback impedance. For the ADA4530 a model is provided by Analog Devices. . . . .	53
5.5	The effect of different parameters on the frequency behaviour of the TIA model. The y-axis shows the signal amplitude in dB. The current source is adjusted to achieve 1 V DC output at $R_F = 5 \text{ G}\Omega$ . . . . .	54

---

5.6	The effect of different parameters on the spectral noise density. $V_{\text{noise}}$ denotes the integrated rms noise. . . . .	55
5.7	Noise scaling with the shunt capacitance. To allow comparison, the ADC LSB voltage is shown. It denotes the voltage corresponding to one ADC channel. . . . .	56
5.8	Conversion timing characteristics of the LTC2327 [54]. A conversion is started by pulling <b>CNV</b> HIGH; it can be pulled LOW after $t_{\text{CNVH}} = 20 \text{ ns}$ . A conversion takes $t_{\text{CONV}} = 1.5 \mu\text{s}$ and is indicated by <b>BUSY</b> being HIGH. After a maximum of $t_{\text{BSDOBUSYL}} = 5 \text{ ns}$ data is valid for reading on <b>SDO</b> and can be read out via the SPI interface. . . . .	59
5.9	Back-end prototype. Schematics can be found in figure A.7. . . . .	60
5.10	Front-end prototype. Schematics can be found in figure A.8. . . . .	61
5.11	Calibration measurement done with the developed prototype. The obtained resolution is estimated to be $0.67 \text{ pA}$ . . . . .	63
5.12	Comparison of the errors on the pAM and the reference readings. Both are plotted against the voltage, that is used to generate the current to be measured. . . . .	64
5.13	Calibration measurement done with the developed prototype and an elongated measurement time for the Keithley. . . . .	65
5.14	Calibration measurement done with the developed prototype and an elongated measurement time for the Keithley. . . . .	65
5.15	Recording of the XBee TX pin and the ADC <b>SDO</b> pin. This measurement allows an estimation of the length of a duty cycle of the pAM. . . . .	66
A.1	Schematics for the back-end of the revision 3 pAM. . . . .	78
A.2	Schematics for the front-end of the revision 3 pAM. The attachment, with the SA5.0CA diode and the $20 \text{ k}\Omega$ resistor, is not depicted here as it is not part of the original design. . . . .	79
A.3	PCB layout of the revision 3 pAM. . . . .	80
A.4	Principle of the calibration station as designed by [4]. From [60]. . . . .	81
A.5	The chimney section of the Calibrations station. On the bottom the metal chimney to guide cables and the ventilation hose into the faraday box; above this, the funnel connecting the hose to the cooler section; on top the fan for cooling the Peltier elements. . . . .	82
A.6	Schematic of the Calibration station after restoration. . . . .	83
A.7	Schematics for the back-end prototype described in section 5.3. The corresponding PCB is shown in figure 5.9. . . . .	84
A.8	Schematics for the front-end prototype described in section 5.3. The corresponding PCB is shown in figure 5.10. . . . .	85
A.9	Schematics for the front-end of the second prototype iteration. . . . .	86
A.10	Schematics for the back-end of the second prototype iteration. . . . .	87
A.11	PCB layout for the second prototype iteration. . . . .	88



# List of Tables

---

2.1	Different junction Seebeck coefficients. From [24], modified.	15
3.1	Modes of the different pAM, the range is given by $\frac{2.048\text{ V}}{1.25R_S}$ . This sets the resolution to range divided by $2^{15}$ . Translated and modified from [5].	26
3.2	Current draw of a pAM. From [5], modified.	27
3.3	Registers, their purpose and ports that feature them. Only ports 1 and 2 have interrupt capability [38].	29
5.1	Characteristics of different Op-Amps. The AD549L is used in the older pAM designs; the OPA128 is its successor. The LMC6442 is used by [51].	50
5.2	Input current ranges and achievable resolution for different feedback resistors $R_F$ , with a $\pm 5\text{ V}$ output range and an ADC input range of $\pm 6.25\text{ V}$ digitised in 15-bits.	52
5.3	Pin configuration of the LTC2327 [54].	58
5.4	Temperature dependencies of components in the front-end.	62
5.5	Current draw of the pAM prototype. For the measurement, a $10\Omega$ resistor was inserted in the 9 V supply line, and the voltage drop was measured using an Oscilloscope.	63
5.6	Comparison of a 9 volt-block battery with a lithium-based accumulator.	66