

**Fachhochschule  
Dortmund**

University of Applied Sciences and Arts



---

# **Master-Thesis**

Fachbereich 3 – Energiesysteme  
Erstprüfer: Prof. Dr. Michael Karagounis  
Zweitprüfer: M.Eng. Jeremias Kampkötter

Validierung des SLDO-Spannungsreglers für die  
Pixeldetektoren des ATLAS- und des CMS- Experiments  
am HL-LHC und Erweiterung des Shuldo-Testsystems um  
programmierbare Potentiometer

**Maurice Bankowsky**

## Kurzfassung

### **Validierung des SLDO-Spannungsreglers für die Pixeldetektoren des ATLAS- und des CMS- Experiments am HL-LHC und Erweiterung des Shuldo-Testsystems um programmierbare Potentiometer**

Die Master-Thesis umfasst die Einführung in die CERN-Projekte und in den Shunt-LDO-Regler. Der Regler wird von der Revision 0.1 hin zur Revision 0.3 weiterentwickelt. Hierbei wird eine Leiterplattenentwicklung mittels Altium Designer, eine grafische Benutzeroberfläche mittels Qt programmiert, sowie Montage- und Lötarbeiten durchgeführt.

Der Kernpunkt der Thesis entspricht der Validierung des SLDO Spannungsreglers für die Pixeldetektoren des ATLAS- und des CMS- Experiments am HL-LHC. Ein weiterer Kernpunkt ist die Implementierung digitaler Potentiometer über denen ein automatisierter Messvorgang mittels der „Shuldo-Test-Messsteuerungs-Software“ durchgeführt werden kann. Hierdurch wird dem Benutzer eine anwendungsfreundliche Umgebung zur Verfügung gestellt, um die Steigung und den Offset der Eingangsspannung über das Testtool in einem bestimmten Bereich automatisiert zu variieren.

## Abstract

### **Validation of the SLDO voltage regulator for the pixel detectors of the ATLAS and CMS experiments at HL-LHC and extension of the Shuldo test system with programmable potentiometers**

The master thesis includes the introduction to the CERN projects and to the Shunt LDO controller. The regulator will be developed from revision 0.1 to revision 0.3. This will involve PCB development using Altium Designer, graphical user interface programming using Qt, as well as assembly and soldering work.

The main point of the thesis is the validation of the SLDO voltage regulator for the pixel detectors of the ATLAS and CMS experiments at the HL-LHC. Another key point is the implementation of digital potentiometers over which an automated measurement process can be performed by means of the "Shuldo Test Measurement Control Software". This provides the user with an application-friendly environment to automatically adjust the slope and offset of the input voltage within a certain range via the test tool.

# Inhaltsverzeichnis

|  |           |
|--|-----------|
| <b>1. EINFÜHRUNG .....</b>   | <b>1</b>  |
| <b>1.1 - Die Aufgabe des CERN's .....</b>                                | <b>1</b>  |
| <b>1.2 - Das ATLAS-Projekt.....</b>                                      | <b>1</b>  |
| 1.2.1 - Atlas .....  | 1         |
| 1.2.2 - Detektor & Technologie .....                                     | 2         |
| <b>1.3 - Das CMS-Projekt .....</b>                                       | <b>3</b>  |
| <b>1.4 - Detektorsystem .....</b>  | <b>4</b>  |
| 1.4.1 - Hybrider Pixeldetektor .....                                     | 4         |
| <br>   |           |
| <b>2. DER SHUNT-LDO-REGLER .....</b>                                     | <b>10</b> |
| <b>2.1 - Entwicklung vom Shunt-Regler hin zum Shunt-LDO-Regler .....</b> | <b>10</b> |
| 2.1.1 - Nebenschlussregler .....   | 10        |
| 2.1.2 - Shunt mit Eingangswiderstand .....                               | 12        |
| <b>2.2 - Funktionsprinzip ShuntLDO-Regler.....</b>                       | <b>14</b> |
| 2.2.1 - LDO-Spannungsregler .....  | 14        |
| 2.2.2 - Shunt-Regler .....   | 15        |
| 2.2.3 - Stromspiegel zur Detektion des Stromflusses .....                | 16        |
| 2.2.4 - Verbesserung der Spiegelgenauigkeit.....                         | 17        |
| 2.2.5 - Referenzspannung für den invertierenden Eingang von A3 .....     | 18        |
| 2.2.6 - Definition des Referenzstroms $I_{ref}$ .....                    | 19        |
| <b>2.3 - Spannungs-/Strom-Charakteristik des Shunt-LDO-Reglers .....</b> | <b>20</b> |
| <b>2.4 - Bandgap-Referenzspannungsschaltung .....</b>                    | <b>22</b> |
| <b>2.5 - Undershunt Current Protection (USCP) .....</b>                  | <b>24</b> |
| <b>2.6 - Überspannungsschutz (OVP) .....</b>                             | <b>26</b> |
| <b>2.7 - AC-Gleichrichter / Low- und High-Power-Modus.....</b>           | <b>28</b> |
| <br>   |           |
| <b>3. MESSSYSTEM .....</b>   | <b>29</b> |
| <b>3.1 - Motivation .....</b>  | <b>29</b> |
| <b>3.2 - Anforderungen an das Messsystem .....</b>                       | <b>29</b> |
| <b>3.3 - Hardwarekomponenten.....</b>                                    | <b>30</b> |
| 3.3.1 - Leiterplatte für den SLDO-Testchip.....                          | 30        |
| 3.3.2 - Leiterplatte Prepower .....                                      | 35        |
| 3.3.3 - Instrumentenverstärker .....                                     | 38        |
| 3.3.4 - Schaltung für Laststrompulse.....                                | 39        |
| 3.3.5 - Digitale Potentiometer .....                                     | 40        |

|   |            |
|---|------------|
| <b>3.4 - Erweiterung der Platine um digitale Potentiometer .....</b>                | <b>43</b>  |
| <b>3.5 - Versuchsaufbau: Konzept zur Integration digitaler Potentiometer .....</b>  | <b>46</b>  |
| <b>3.6 - Schaltpläne der digitalen Potentiometer.....</b>                           | <b>47</b>  |
| <b>3.7 - Modifikation der Leiterplatte (Rev 0.4) .....</b>                          | <b>49</b>  |
| <b>3.8 - Messaufbau .....</b>   | <b>50</b>  |
| <b>3.9 - Messgeräte .....</b>   | <b>52</b>  |
| 3.9.1 - Sourcemeter Keithley 2460 und Keithley 2401.....                            | 52         |
| 3.9.2 - Labornetzgerät Rohde & Schwarz HMP4030 .....                                | 53         |
| 3.9.3 - Oszilloskop Rohde & Schwarz HMO1024.....                                    | 53         |
| 3.9.4 - Multimeter Keithley DMM6500 und Keithley 2000.....                          | 54         |
| 3.9.5 - Waveform Generator Rohde & Schwarz HMF2525 .....                            | 54         |
| <b>3.10 - Die Shuldo-Test-Messsteuerungs-Software.....</b>                          | <b>55</b>  |
| 3.10.1 - Einleitung .....   | 55         |
| 3.10.2 - Starten einer Messung:.....  | 56         |
| <b>3.11. Erweiterung der Software zur Ansteuerung digitaler Potentiometer .....</b> | <b>58</b>  |
| <b>4. MESSERGEBNISSE.....</b>   | <b>68</b>  |
| <b>4.1 - Spezifikation .....</b>  | <b>68</b>  |
| <b>4.2 - Statische Messungen .....</b>  | <b>69</b>  |
| 4.2.1 Line Regulation.....  | 69         |
| 4.2.2 Load Regulation .....   | 71         |
| <b>4.3 - Transiente Messungen .....</b>   | <b>72</b>  |
| 4.3.1 Einfluss von Variationen der Eingangsspannung .....                           | 72         |
| 4.3.2 Einfluss von Lastwechseln .....   | 74         |
| 4.3.3 Untersuchung des Einschaltverhaltens .....                                    | 76         |
| 4.3.4 Validierung der Overvoltage Protection .....                                  | 78         |
| 4.3.5 Validierung der Undershunt Current Protection.....                            | 79         |
| 4.3.6 Validierung des Low-Power-Modus .....   | 81         |
| <b>4.4 - Messungen im seriellen Betrieb .....</b>                                   | <b>82</b>  |
| <b>4.5 - Messung im redundaten Betrieb.....</b>                                     | <b>86</b>  |
| <b>4.6 - Validierung der OVP im seriellen Betrieb .....</b>                         | <b>88</b>  |
| <b>4.7 - Auswertung der Messergebnisse .....</b>                                    | <b>94</b>  |
| <b>4.8 - Messergebnisse aktuelle Testplatine (Rev. 0.3).....</b>                    | <b>96</b>  |
| <b>5. FAZIT.....</b>  | <b>101</b> |
| <b>6. ABBILDUNGSVERZEICHNIS.....</b>  | <b>102</b> |

|   |            |
|---|------------|
| 7. QUELLENVERZEICHNIS .....                       | 105        |
| 8. EIDESSTATTLICHE VERSICHERUNG.....              | 112        |
| 9. ANHANG .....                                   | 113        |
| <b>9.1 - Montage- und Lötarbeiten .....</b>       | <b>113</b> |
| 9.1.1 - Arbeitsschritte.....                      | 113        |
| <b>9.2 - Schematic der TestPCB-RD53B .....</b>    | <b>117</b> |
| <b>9.3 - Schematic der PrePower-Platine .....</b> | <b>118</b> |
| <b>9.4 - Programmcode Qt.....</b>                 | <b>119</b> |
| 9.4.1 - Header „digitalepotis.h“ .....            | 119        |
| 9.4.2 - Klasse „digitalepotis“ .....              | 120        |
| 9.4.3 - Klasse „gpib“ .....                       | 123        |
| 9.4.4 - Header „sldo65.h“ .....                   | 163        |

## Abkürzungsverzeichnis

|               |   |
|---------------|---|
| <b>ATLAS</b>  | (A Toroidal LHC ApparatuS)-Experiment bzw. Detektor am LHC                  |
| <b>CERN</b>   | Europäische Organisation für Kernforschung in Genf                          |
| <b>CMS</b>    | (Compact Muon Solenoid)-Experiment bzw. Detektor am LHC                     |
| <b>DAC</b>    | Digital-to-Analog-Converter   |
| <b>GPIB</b>   | General Purpose Interface Bus   |
| <b>GUI</b>    | Graphical User Interface  |
| <b>HPM</b>    | High-Power Modus  |
| <b>LDO</b>    | Low-Dropout-Spannungsregler   |
| <b>LHC</b>    | Large Hadron Collider (Teilchenbeschleuniger am Kernforschungszentrum CERN) |
| <b>LPM</b>    | Low-Power Modus   |
| <b>HL-LHC</b> | Projekt zur Erhöhung der Luminosität am LHC                                 |
| <b>NMOS</b>   | N-Kanal-Metall-Oxid-Halbleiter-Feldeffekttransistor                         |
| <b>PMOS</b>   | P-Kanal-Metall-Oxid-Halbleiter-Feldeffekttransistor                         |
| <b>OVP</b>    | Overvoltage Protection (Überspannungsschutz)                                |
| <b>PCB</b>    | Printed Circuit Board (Leiterplatte)  |
| <b>OP</b>     | Operationsverstärker  |
| <b>SLDO</b>   | Shunt-Low-Dropout-Spannungsregler   |
| <b>USB</b>    | Universal Serial Bus  |
| <b>USCP</b>   | Undershunt Current Protection   |
| <b>VDS</b>    | Drain-Source-Spannung   |
| <b>VGS</b>    | Gate-Source-Spannung  |
| <b>VT</b>     | Schwellenspannung   |

## 1. Einführung

Seit Ende Juni 2019 ist die Fachhochschule Dortmund Teil der Atlas Kollaboration, welche sich mit der Entwicklung, der Installation und des Betriebs des gleichnamigen Experiments am CERN beschäftigt. Für dieses Experiment entwickelt die Fachhochschule Dortmund einen Spannungsregler, der für die Stromversorgung der Pixeldetektormodule im Experiment vorgesehen ist. Die Spannungsregler müssen hierbei mindestens 10 Jahre in einem strahlungsintensiven Umfeld funktionsfähig bleiben. Es ist geplant, die Komponenten im Rahmen eines anstehenden Upgrades einzubauen.

### 1.1 - Die Aufgabe des CERN's

„Die Forschung am CERN beschäftigt sich mit der Frage, aus welchen Elementarteilchen die Materie im Universum besteht und nach welchen Gesetzmäßigkeiten diese miteinander wechselwirken. Mit Teilchenbeschleuniger-Anlagen werden unterschiedliche Experimente durchgeführt, um die Grenzen des menschlichen Wissens in diesem Bereich zu erweitern.

Die Forschungseinrichtung wurde 1954 gegründet und ist heute ein Musterbeispiel für internationale Zusammenarbeit.

Zu den Missionen des CERN's zählen unter anderem:

- Die Bereitstellung von Teilchenbeschleunigeranlagen, um die Forschung auf dem neuesten technologischen Stand und mit den modernsten vorhandenen Methoden zu ermöglichen.
- Die Spitzenforschungen im Bereich der Grundlagenphysik voranzutreiben.
- Das Zusammenführen von Menschen aus den verschiedensten Ländern auf der gesamten Welt, um die Grenzen von Wissenschaft und Technologie zum Nutzen aller zu erweitern.“  
[CER1][CER4]

### 1.2 - Das ATLAS-Projekt

#### 1.2.1 - Atlas

„ATLAS ist eines der vier großen Experimente am „Large Hadron Collider“ (LHC) am CERN. Es handelt sich um ein universelles Experiment der Teilchenphysik, dass von einer internationalen Kollaboration durchgeführt wird und zusammen mit dem CMS Projekt ein großes Entdeckungspotential und die enormen Möglichkeiten der Physik ausschöpfen soll, welche der LHC bietet.

Die wissenschaftlichen Forschungen im Rahmen des ATLAS Experiments erfordern sehr präzise Messungen, um die Grenzen des Wissens zu erweitern und Antworten auf grundlegende Fragen zu geben, wie z.B.:

Was sind die Grundbausteine der Materie? Was sind die fundamentalen Kräfte der Natur? Könnte es eine größere zugrunde liegende Symmetrie für unser Universum geben?

Die ATLAS-Physiker überprüfen die Vorhersagen des Standardmodells, dass unser derzeitiges Verständnis der Bausteine von Materie und ihrer Wechselwirkungen zusammenfasst. Diese Studien haben bisher zu bahnbrechenden Entdeckungen geführt, wie beispielsweise dem des Higgs-Bosons. Darüber hinaus wird aber auch die Physik jenseits des Standardmodells untersucht und die Entwicklung neuer Theorien zur besseren Beschreibung unseres Universums vorangetrieben.

Von vielen Wissenschaftlern werden die Entwicklungen in den kommenden Jahren mit Spannung erwartet, da das ATLAS Experiment die experimentelle Physik in unerforschte Gebiete führt und möglicherweise durch die Entdeckung neuer Prozessen und Elementarteilchen zu einem veränderten Verständnis von Energie und Materie führen können.“ [CER2]

### 1.2.2 - Detektor & Technologie

„ATLAS ist der größte Detektor, der jemals für einen Teilchenbeschleuniger gebaut wurde. Er hat die Form eines Zylinders mit einer Länge von 46 m, sowie einem Durchmesser von 25 m und befindet sich in einer Tiefe von 100 m unter der Erde. Der ATLAS-Detektor wiegt 7.000 Tonnen, ähnlich wie der Eiffelturm.

Der Detektor selbst ist ein vielschichtiges Instrument, mit dem sich einige der kleinsten und dennoch energiereichsten Teilchen nachweisen lassen, die jemals auf der Erde erzeugt wurden. Er besteht aus sechs verschiedenen Erkennungssystemen, die konzentrisch in Schichten um den Kollisionspunkt positioniert sind, um die Flugbahn, den Impuls und die Energie von Partikeln aufzuzeichnen und so einzeln zu identifizieren und zu messen. Ein riesiges Magnetsystem krümmt die Bahnen der geladenen Teilchen, damit ihre Impulse so genau wie möglich gemessen werden können.

Teilchen, die sich mit einer Energie von bis zu sieben Billionen Elektronenvolt oder einer Lichtgeschwindigkeit von bis zu 99,999999% im LHC fortbewegen, kollidieren in der Mitte des ATLAS-Detektors und erzeugen Kollisionsprodukte in Form neuer Teilchen, die in alle Richtungen zerstreut werden. Über eine Milliarde Partikelwechselwirkungen finden hierbei pro Sekunde im Detektor statt. Dies entspricht einer Datenrate von 20 gleichzeitigen Telefongesprächen, die von jedem Menschen auf der Erde geführt werden. Nur eine von einer Million Kollisionen wird als potenziell interessant eingestuft und für weitere Studien aufgezeichnet. Der Detektor verfolgt und identifiziert Partikel, um ein breites Spektrum der Physik zu untersuchen. Dabei reichen die Experimente von der Untersuchung des Higgs-Bosons und des Top-Quarks bis hin zur Erforschung zusätzlicher Dimensionen und Partikel, aus denen sich dunkle Materie zusammensetzen könnte.“ [CER3][CIM1]



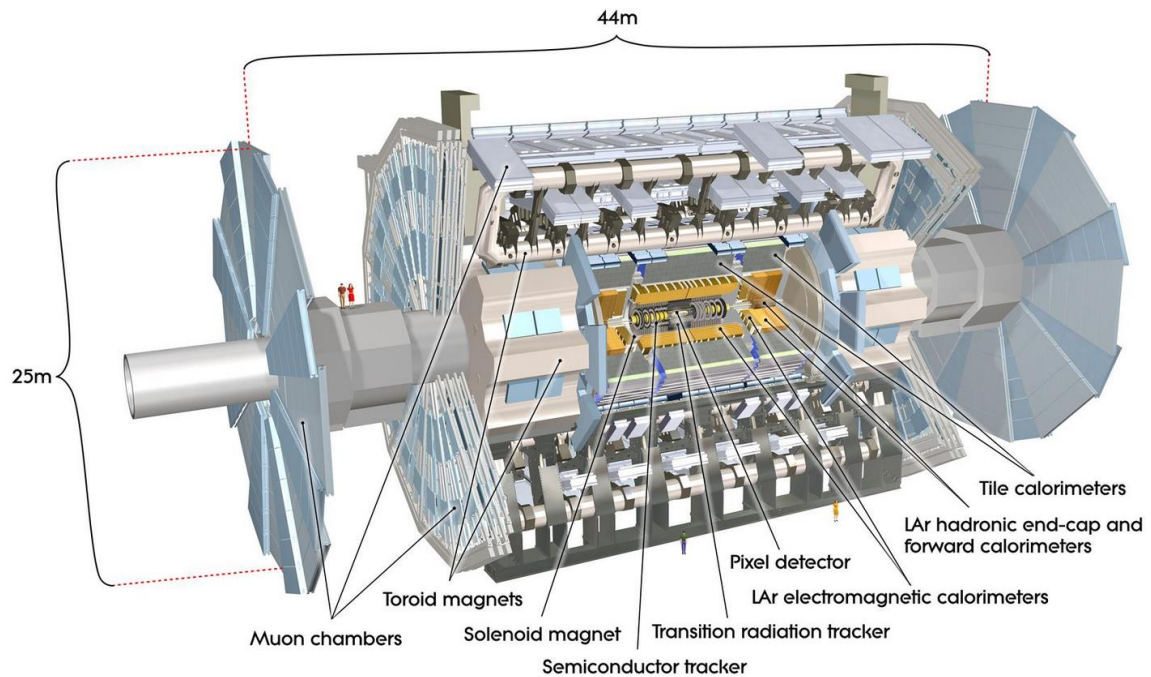


Abbildung 1: Aufbau ATLAS-Detektors [CIM1]

### **1.3 - Das CMS-Projekt**

„Das „Compact Muon Solenoid - Experiment“ (CMS) ist ein Allzweckdetektor am LHC. Der Detektor kommt für verschiedene Forschungsbereiche zum Einsatz, wie z.B. für die Untersuchung des Standardmodells einschließlich des Higgs-Bosons bis hin zur Suche nach zusätzlichen Dimensionen und Elementarteilchen, aus denen dunkle Materie bestehen könnte. Obwohl es die gleichen wissenschaftlichen Ziele wie das ATLAS-Experiment verfolgt, verwendet es andere technische Lösungen sowie ein unterschiedlich angeordnetes Magnetsystem.

Der CMS-Detektor befindet sich um einen riesigen Magneten. Dieser wird in Form einer zylindrischen Spule mit supraleitenden Kabeln aufgebaut und erzeugt ein Feld von 4 Tesla, was in etwa dem 100.000-fachen des Erdmagnetfelds entspricht. Das Feld wird von einem Stahljoch begrenzt, welches den Großteil des 14.000-Tonnen-Gewichts des Detektors ausmacht.

Ein ungewöhnliches Merkmal des CMS-Detektors ist, dass er nicht wie die anderen riesigen Detektoren der LHC-Experimente vor Ort gebaut, sondern in 15 Abschnitten auf Bodenniveau errichtet und anschließend in eine unterirdische Höhle in der Nähe von Cessy in Frankreich abgesenkt und wieder zusammengebaut wurde. Der gesamte Detektor ist 21 m lang, 15 m breit und 15 m hoch.

Das CMS-Experiment ist eines der größten internationalen wissenschaftlichen Kollaborationen in der Geschichte, an der 4300 Teilchenphysiker, Ingenieure, Techniker, Studenten und Mitarbeiter aus 182 Instituten in 42 Ländern (Stand Februar 2014) beteiligt sind.“ [CER5]

## 1.4 - Detektorsystem

Um Teilchen nachweisen und anhand ihrer Eigenschaften bestimmen zu können, werden Detektorsysteme benötigt. Die sogenannten Hybriden Pixeldetektoren werden für die zwei-dimensionale Teilchendetektion mit hoher Auflösung eingesetzt. Die Detektoren im ATLAS- und CMS Experiment des Teilchenbeschleunigers werden zurzeit weiterentwickelt, um eine höhere Trefferrate verarbeiten zu können. [KAR4][KAR5]

### 1.4.1 - Hybrider Pixeldetektor

In diesem Kapitel wird das Funktionsprinzip des hybriden Pixeldetektors erklärt. Der Pixeldetektor besteht aus zwei Komponenten - einem Sensor und einem Chip (siehe Abbildung 2).

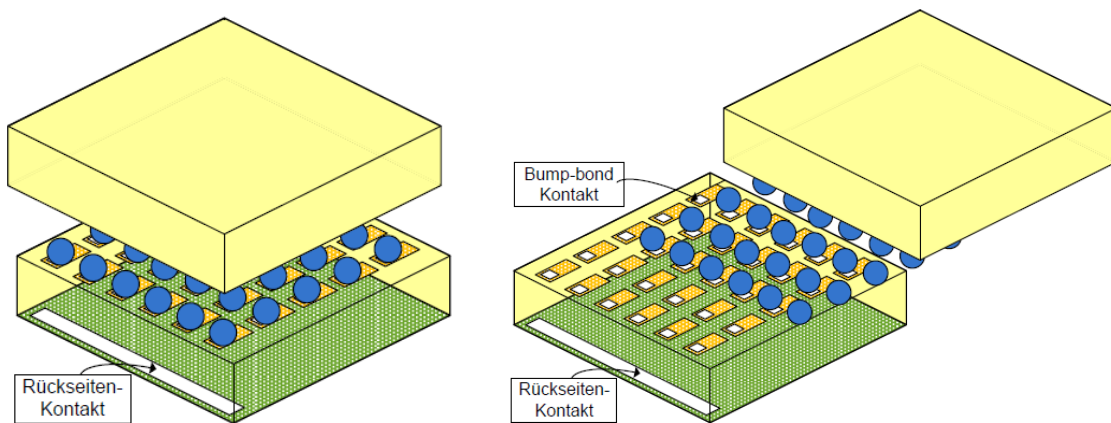


Abbildung 2: Pixeldetektor und Pixeldetektor inkl. (Bump-bond-Kontakt) [KAR5]

Der Sensor besteht vom Prinzip her aus einer zweidimensionalen Anordnung von PN-Dioden, welche in Sperrrichtung betrieben werden. Die Sperrspannung wird dabei so weit erhöht, dass sich die Raumladungszone über die komplette Tiefe des Sensors von 300  $\mu\text{m}$  erstreckt. Wenn nun ein Teilchen durch den Pixeldetektor hindurchfliegt, werden Elektron-Loch-Paare ionisiert. Durch die angelegte Sperrspannung herrscht ein elektrisches Feld, welches die generierten Elektron-Lochpaare entsprechend ihrer Ladung trennt und in Richtung der Ausleseelektroden bewegt. Durch einen ladungsempfindlichen Verstärker wird der Strom im Rückkoppel Kondensator  $C_f$  integriert und in eine Spannung gewandelt (siehe Abbildung 3). Das Stromeingangssignal des ladungsempfindlichen Verstärkers ist pulsformig und kann näherungsweise mathematisch mit einem DIRAC-Puls beschrieben werden, während das Ausgangssignal des Verstärker einer Sprungfunktion ähnelt. Anschließend durchläuft das Signal eine Reihenschaltung aus einem Hoch-Pass-Filter und einem Tief-Pass-Filter, um spektrale Rauschanteile bei sehr tiefen und sehr hohen Frequenzen zu filtern. Im nächsten Schritt wird das Spannungssignal durch einen Komparator mit einer Schwelle verglichen. Steigt das Signal über die Schwelle, so erzeugt der Komparator einen logischen Impuls. Dieser Logikpuls wird im Digitalteil des Pixels verwendet, um eine Zeitmarke zu generieren, die darüber Auskunft gibt, zu welchem Zeitpunkt das entsprechende Teilchen auf den Pixel getroffen ist. Hierbei ist zu erwähnen, dass nicht alle

Treffer von physikalischer Bedeutung sind. Eine Unterscheidung wird mit Hilfe von Trigger-Signalen durchgeführt, welche innerhalb einiger  $\mu\text{s}$  erzeugt werden, um Ereignisse, die von Bedeutung sind zu kennzeichnen. Irrelevante Treffer werden anschließend automatisch gelöscht. Interessante Ereignisse treten relativ selten ein. Um mehr Messergebnisse für eine bessere statistische Auswertung zu erhalten, soll im Experiment, die Anzahl der Kollisionen erhöht werden. Der Nachteil hierbei ist, dass der Chip bei hohen Trefferraten sättigt und mit der Verarbeitung auf Grund von Totzeiten nicht mehr nachkommt. Dieses Problem soll durch eine kleinere Pixelgröße gelöst werden. Bei gleichbleibender Detektionsfläche entspricht dies einer Erhöhung der Pixelanzahl. Durch die größere Anzahl der Pixel verteilen sich die Treffer auf mehrere Pixel und die Trefferraten pro Pixel nehmen auch bei höherer Gesamttrefferzahl nicht zu. Nachteilig hierbei ist, dass die Stromaufnahme an die Anzahl der Pixel gebunden ist und damit eine höhere Pixelanzahl gleichzeitig auch eine höhere Stromaufnahme bedeutet. [KAR4][KAR5]

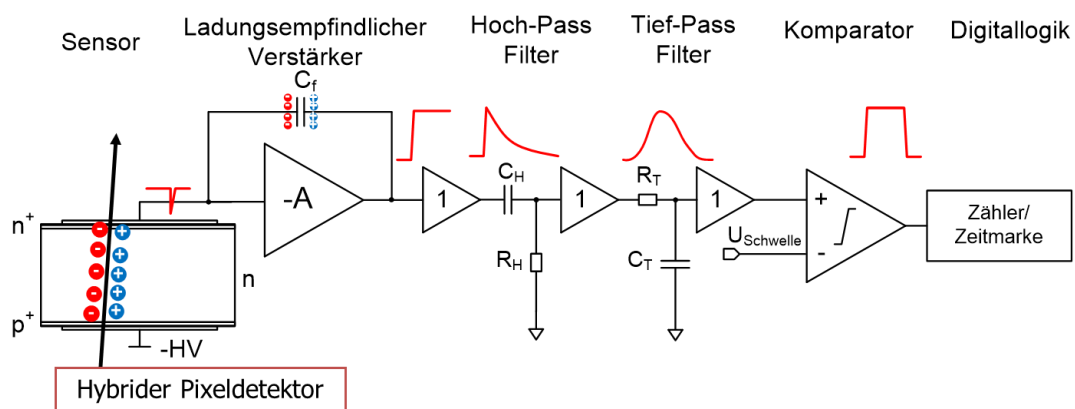


Abbildung 3: Hybrider Pixeldetektor [KAR5]

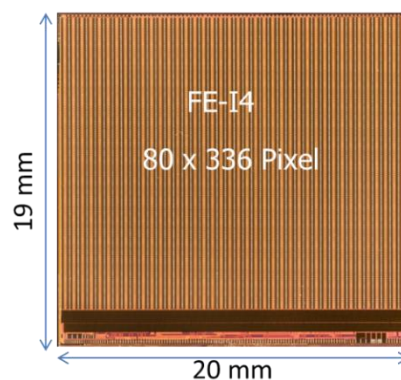


Abbildung 4: Pixel des Detektors [KAR5]

Abbildung 4 zeigt die Pixel des Detektors. Der Detektor bietet  $80 \cdot 336$  Pixel bei einer Fläche von  $20 \cdot 19$  mm.

Aktuell werden die Module mittels Gleichspannung in einer parallelen Anordnung betrieben. Die Spannungsquelle darf sich auf Grund fehlender Strahlungsfestigkeit jedoch nicht im aktiven Bereich des Experiments befinden und ist deshalb in einem Kontrollraum in 300 m Entfernung installiert. Auf Grund des hohen Stromes und des großen Widerstandes der langen Versorgungsleitungen ergibt sich zudem ein relativ großer Spannungsabfall auf den Leitungen. Ohne Gegenmaßnahmen ist zu erwarten, dass die auf den Versorgungs-

leitungen umgesetzte Leistung, der vom Detektor aufgenommenen Leistung entspricht. Die Versorgungseffizienz erreicht dementsprechend nur sehr geringe Werte von etwa 50%.

#### 1.4.1.1 - Varianten der Versorgungsschema

Für die Verbesserung der Versorgungseffizienz wurden die folgenden Optionen diskutiert.

##### Möglichkeit 1: Versorgungsschema mit parallel verschalteten Modulen

Eine Verbesserung in Bezug auf die Effizienz kann erzielt werden, indem die an den Versorgungsleitungen anliegende Spannung vergrößert wird. Ein höherer Spannungspegel reduziert den Versorgungsstrom durch die Kabel und damit einhergehend auch die Leitungsverluste. Mit der nachfolgenden Gleichung können die Verluste abgeschätzt werden:

$$P_{Kabel} = R \cdot I^2 \quad (1.4_1)$$

Aus der Formel 1.4\_1 wird ersichtlich, dass neben dem Strom auch der Leitungswiderstand ausschlaggebend für die Verluste ist. Eine Optimierung des Leitungswiderstandes ist nicht möglich, da aufgrund des begrenzten Platzbedarfs, der Kabeldurchmesser nicht beliebig vergrößert werden kann. Demnach besteht lediglich die Möglichkeit, den Strom und die Leitungsverluste zu verringern, indem die Quellenspannung angehoben wird. Durch den Einsatz von DC/DC-Wandlern, bzw. Tiefsetzstellern, wird die Quellenspannung in unmittelbarer Nähe zu den Pixelmodulen auf ein niedrigeres Spannungslevel gewandelt. Somit wird ein höheres Spannungslevel über den Leitungen ermöglicht und die parallel angeordneten Module können mit der erforderlichen, niedrigeren Spannung versorgt werden.

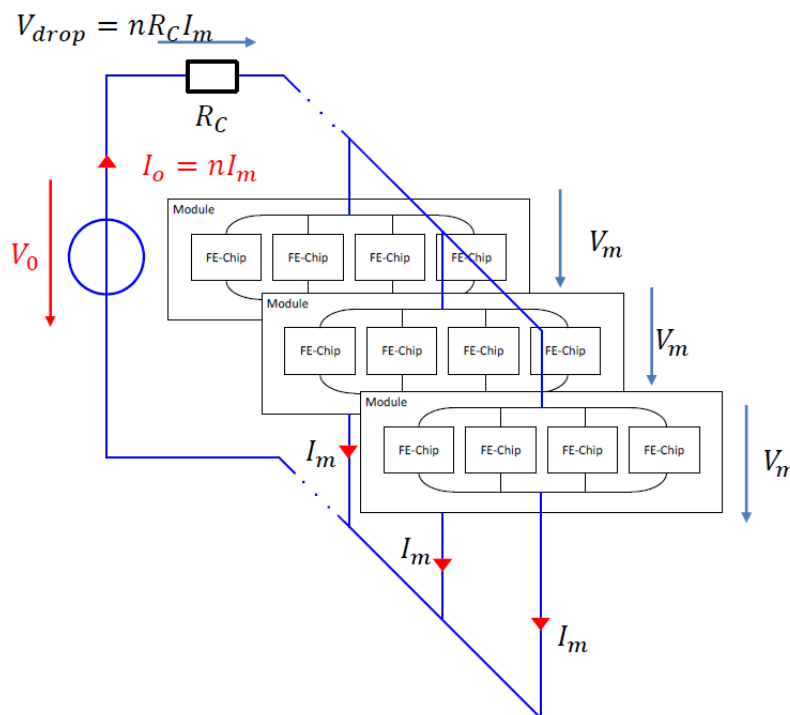


Abbildung 5: Versorgungsschema mit parallel verschalteten Modulen [KAR5]

Abbildung 5 zeigt die parallele Anordnung der einzelnen Pixelmodule. Der Gesamtstrom  $I_0$  teilt sich pro Modul in die Teilströme  $I_m$  auf:

$$I_0 = n \cdot I_m \quad (1.4_2)$$

Vorteilhaft bei diesem Schema ist, dass an jedem Modul eine konstante Spannung  $V_m$  abfällt. Da außerdem jedes Modul individuell mit der Versorgungsquelle verbunden ist, teilen sich die Module untereinander keine Spannungsleitung, wodurch eine gegenseitige Beeinflussung durch die Kabel ausgeschlossen werden kann. Weiter kann im Falle eines Fehlers, die Spannungsversorgung für ein individuelles Modul unabhängig von den anderen abgeschaltet werden.

Der große Nachteil hingegen ist, dass die Größe des Stroms von der Anzahl an Modulen festgelegt wird. So nimmt mit einer großen Anzahl an Pixelmodulen der Strom über den Leitungen zu. Weiter hat eine hohe Anzahl an Versorgungskabeln ein hohes Materialbudget zur Folge. Aus der nachfolgenden Formel wird ersichtlich, wie die Effizienz mit der Anzahl an Modulen zusammenhängt:

$$\eta_{parallel} = \frac{1}{1 + \frac{n \cdot R_c \cdot I_m}{V_m}} \quad (1.4_3)$$

In der Formel 1.4\_3 werden folgende Parameter verwendet: Die Anzahl an Modulen  $n$ , der Modulstrom  $I_m$ , der Modulspannung  $V_m$  und der Leitungswiderstand  $R_c$ . Mit steigender Anzahl an Modulen sinkt die Effizienz.

Für ein paralleles Versorgungsschema werden Spulen im Tiefsetzsteller verwendet, die um einem Ferritkern gewickelt sind, damit hohe Induktivitätswerte erreicht werden können. Der Detektor ist wiederum von einem zwei Tesla starkem Magnetfeld umgeben, um zwischen geladenen und ungeladenen Teilchen unterscheiden zu können. Auf Grund der Lorentzkraft bilden die geladenen Teilchen spiralförmige Flugbahnen, während sich ungeladene Teilchen gradlinig weiterbewegen. Das enorme Magnetfeld treibt die ferromagnetischen Spulen jedoch in Sättigung, wodurch der positive Effekt des Ferritkerns auf die Induktivität eliminiert wird und die Glättungsspule wie eine Luftspule wirkt. Das hat zur Folge, dass die benötigte Induktivität nur bei größerem Spulenvolumen erreicht werden kann. Das stellt einen wesentlichen Nachteil dieser Lösung dar, da der hierfür benötigte Platz im Experiment nicht vorhanden ist. Schlussfolgernd kann festgehalten werden, dass ein Versorgungsschema mit dem Einsatz von Tiefsetzstellern mit einer großen externen Induktivität auf Grund der sehr speziellen Bedingungen im Experiment eine unvorteilhafte Methode darstellt. Deshalb wurde dieser Ansatz zum jetzigen Zeitpunkt nicht weiterverfolgt. [BAN1][BAN2][FUZ][KAR1][KAR4][KAR5][MUE]

Möglichkeit 2: Versorgungsschema mit seriell verschalteten Modulen

Bei diesem Konzept werden die Module in Reihe verschaltet, sodass durch jedes Modul in der Kette der gleiche Versorgungsstrom geführt wird. Die Versorgung der Module erfolgt durch eine Konstantstromquelle statt einer Konstantspannungsquelle. Im Gegensatz zum parallelen Versorgungsschema ergibt sich bei dieser Methode der maximale Versorgungsstrom aus dem höchsten zulässigen Laststrom und vergrößert sich nicht mit zunehmender Anzahl an Modulen.

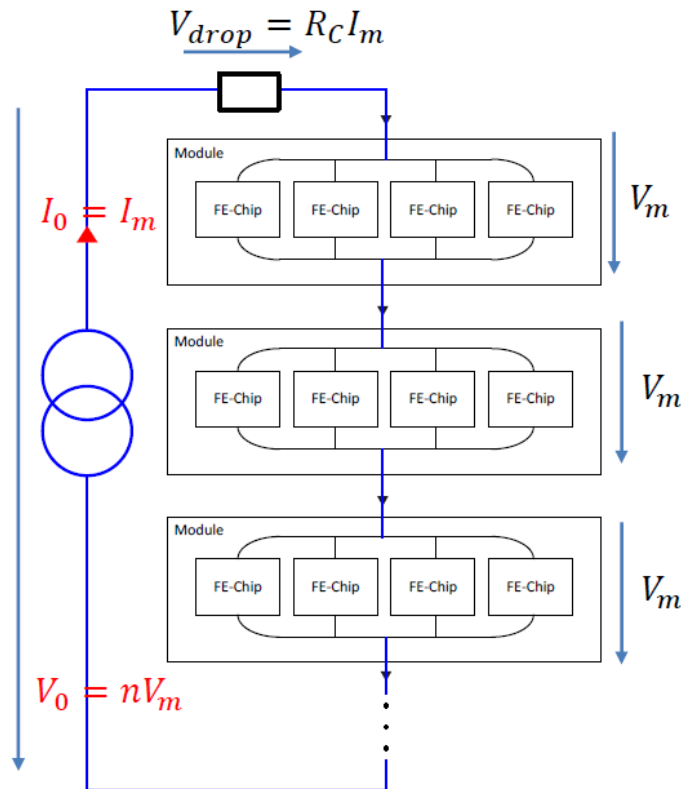


Abbildung 6: Versorgungsschema mit seriell verschalteten Modulen [KAR5]

Die serielle Anordnung der Pixelmodule ist in Abbildung 6 dargestellt. Hier liegt bei jedem Modul der identische Versorgungsstrom an ( $I_0 = I_m$ ) und die Versorgungsspannung  $V_0$  richtet sich nach der Anzahl an Modulen.

$$V_0 = n \cdot V_m \tag{1.4_4}$$

Die Effizienz bei der seriellen Versorgung ergibt sich aus der nachfolgenden Formel:

$$\eta_{serial} = \frac{1}{1 + \frac{R_C \cdot I_m}{n \cdot V_m}} \tag{1.4_5}$$

Ein erheblicher Vorteil ist, dass der Versorgungsstrom unabhängig von der Anzahl an Modulen einer Kette ist und damit die Leitungsverluste im Gegensatz zum parallelen Versorgungsschema reduziert werden können. Versorgt wird die Modulkette mit einer Stromquelle, wobei Spannungsregler benötigt werden, um den Versorgungsstrom in eine konstante Spannung zu wandeln. Ebenfalls reduziert sich hier die Anzahl der Leitungen, da für jede Kette nur ein stromzuführendes sowie -abführendes Kabel benötigt wird. Ebenfalls von Vorteil ist, dass jedes Modul eine konstante Abwärme erzeugt. Damit kann auf eine aufwendige Regelung für das Kühlsystem verzichtet werden.

Nachteilig bei der seriellen Anordnung ist, dass es sich um ein massefreies System handelt und somit ein Masseversatz von 2 V zu jedem Modul besteht. Durch die Anwendung einer AC-Kopplung wird das Problem gelöst. Bei der AC-Kopplung erreicht ausschließlich der Wechselspannungsanteil des Signals das Modul und der Gleichspannungsanteil wird unterdrückt. Realisiert wird dies, indem das Signal durch einen Kondensator eingekoppelt wird. Die Komplexität bei der Nutzung des seriellen Versorgungsansatzes besteht darin, dass aus dem Versorgungsstrom eine Spannung generiert werden muss, wofür Spannungsregler benötigt werden.

Ein weiteres Problem ist die Gewährleistung der Ausfallsicherheit. Fällt ein Regler aus, besteht die Gefahr, dass die Kette an dieser Stelle unterbrochen wird. Das hat zur Folge, dass bei einem Defekt eines Reglers, alle weiteren seriellen Module in der Kette ebenfalls einen Defekt erleiden und somit ein Teil des Detektors insensitiv d.h. blind wird. Würde beispielsweise das erste Modul ausfallen, dann wäre die gesamte Kette nicht mehr in Betrieb. Um die Ausfallsicherheit zu vergrößern, wurde ein redundanter Regler Betrieb eingeführt, bei dem mehrere Regler auf Modulebene parallelgeschaltet werden. [BAN1] [MUE][FUZ][KAR1][BAN2][KAR4][KAR5]

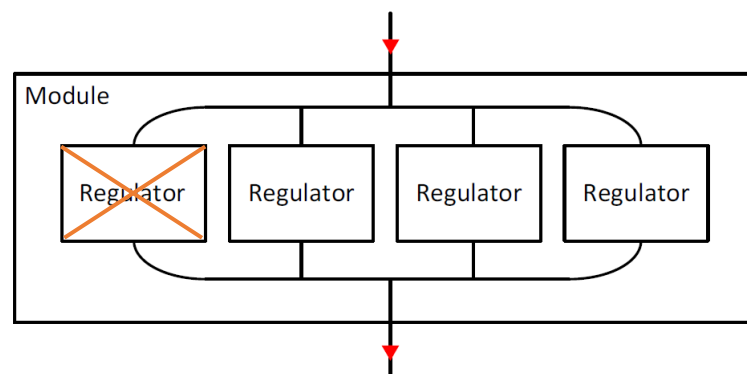


Abbildung 7: Modul mit vier parallelen Reglern [KAR5]

Abbildung 7 zeigt ein Modul mit vier parallel geschalteten Reglern, die einen redundanten Betrieb ermöglichen.

## 2. Der Shunt-LDO-Regler

### 2.1 - Entwicklung vom Shunt-Regler hin zum Shunt-LDO-Regler

#### 2.1.1 - Nebenschlussregler

Für die ersten Konzepte der seriellen Versorgung wurden Shunt-Regler betrachtet, die aus dem Versorgungsstrom eine Spannung generieren und den überschüssigen Strom, der nicht von der Last aufgenommen wird, an das nächste Modul weiterleiten.

Abbildung 8 zeigt den Aufbau eines Shunt-Reglers.

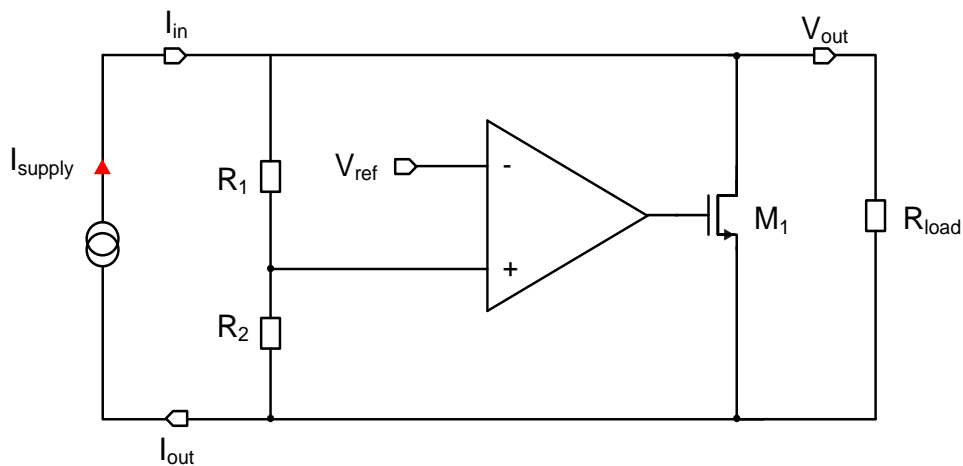


Abbildung 8: Schaltbild eines Shunt-Reglers [KAR5]

Der Shunt-Regler wird verwendet, um aus dem Versorgungsstrom eine konstante Spannung zu generieren und gleichzeitig den überschüssigen Strom zum nächsten Modul weiterzuleiten. Die Ausgangsspannung wird reguliert, indem der Stromfluss durch den Shunt-Transistor  $M_1$  gesteuert wird. Hierbei wird die Ausgangs- bzw. Eingangsspannung über einen resistiven Spannungsteiler auf den nichtinvertierenden Eingang des Operationsverstärker zurück gekoppelt. Der Operationsverstärker vergleicht das rückgekoppelte Signal mit einem Referenzwert und steuert den Shunt-Transistor so an, dass beide Eingänge ein identisches Potential führen. Im Idealfall liegt die Spannungsdifferenz der Eingänge bei null, durch Prozessvariationen oder Temperatureinflüssen tritt jedoch ein Offset auf. Die Ausgangsspannung  $V_{out}$  ergibt sich bei Vernachlässigung des Offsets aus der nachfolgenden Gleichung:

$$V_{out} = \left( 1 + \frac{R_1}{R_2} \right) \cdot V_{ref} \quad (2.1_1)$$

Entscheidend für den Wert der Ausgangsspannung sind demnach der Referenzwert  $V_{ref}$  sowie das Rückkoppelnetzwerk mit dem resistiven Spannungsteiler. [KAR4][KAR5]



Mit Hilfe eines Beispiels wird die Funktionalität des Reglers im Folgenden beschrieben.

Der Regler wird mit einem konstanten Versorgungsstrom gespeist. Im stationären Zustand fällt über der Last am Ausgang eine Spannung von 1 V ab. Der resistive Spannungsteiler wird dementsprechend so gewählt, dass die halbe Ausgangsspannung auf den nicht-invertierenden Eingang des Operationsverstärker zurück gekoppelt wird. Der Referenzwert für den invertierenden Eingang liegt bei 500 mV. Mit einem Lastwiderstand von 1  $\Omega$  ergibt sich ein Laststrom von 1 A. Bei einer plötzlichen Änderung des Lastwiderstandes von 1  $\Omega$  auf 2  $\Omega$  fließt zunächst weiterhin ein Strom von 1 A durch die Last, jedoch steigt die Ausgangsspannung auf 2 V. Da nun der nichtinvertierende Eingang (+) des Operationsverstärkers wesentlich größer als der invertierende Eingang (-) ist, nimmt das Ausgangspotential des OPs zu. Dies führt zu einer Vergrößerung der Gate-Source-Spannung des Shunt-Transistors und zu einem höheren Stromfluss durch diesen. Ein Teil des Laststroms fließt nun durch den Shunt-Transistor, wodurch die Last einen kleinen Strom sieht, bis sich wiederum eine Ausgangsspannung von 1 V einstellt. Da der überschüssige Strom, der nicht von der Last aufgenommen wird, nach Masse abgeleitet wird, ist der Regler sehr ineffizient und für viele praktische Anwendungen z.B. mobile Geräte ungeeignet.

Abbildung 9 zeigt den Verlauf der Eingangs- bzw. Ausgangsspannung über dem Versorgungsstrom von drei parallel geschalteten Shunt-Reglern. Hier wurde überprüft, ob mit dem Konzept des Shunt-Reglers ein redundanter Betrieb möglich ist, um die Ausfallsicherheit zu gewährleisten. Aus dem Verlauf kann jedoch entnommen werden, dass die Regler unterschiedliche Spannungen generieren und damit keine gleichmäßige Stromverteilung gegeben ist. [KAR4][KAR5]

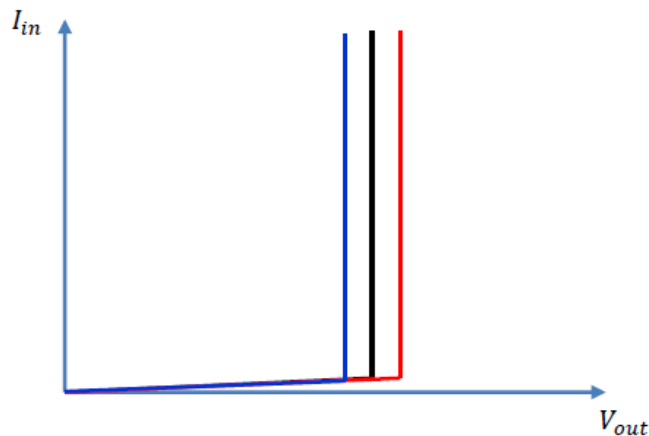


Abbildung 9: Strom/Spannungscharakteristik für drei parallele Shunt-Regler [KAR5]

Unterschiedliche Faktoren haben einen Einfluss auf die abweichenden Spannungen der Regler auf Modulebene. In den Schaltungen, welche die Referenzspannungen erzeugen kommt es zu Abweichungen in den Referenzsignalen, aber auch der Offset am Operationsverstärkers sowie Varianzen der Widerstände des Rückkoppelnetzwerkes haben einen negativen Einfluss auf die Ausgangsspannungsgenauigkeit der Regler. Ein weiterer Nachteil ist, dass die Steigungen der Kurve, welche die Strom-Spannungscharakteristik des Reglers beschreibt, sehr hoch ist, was ebenso dazu beiträgt, dass der Großteil des Stroms durch den Regler mit der kleinsten Spannung fließt. Schlussfolgernd kann festgehalten werden, dass die serielle Verschaltung mit den Shunt-Reglern prinzipiell möglich ist, allerdings ist der redundante Betrieb aufgrund der ungleichmäßigen Stromverteilung nicht anwendbar und könnte zur Zerstörung der Regler führen. [KAR4][KAR5]

## 2.1.2 - Shunt mit Eingangswiderstand

Da mit dem ersten Ansatz ein sicherer redundanter Betrieb nicht erreicht werden konnte, wurde für ein neues Konzept, ein Widerstand  $R_{slope}$  am Eingang des Shunt-Reglers platziert (siehe Abbildung 10).

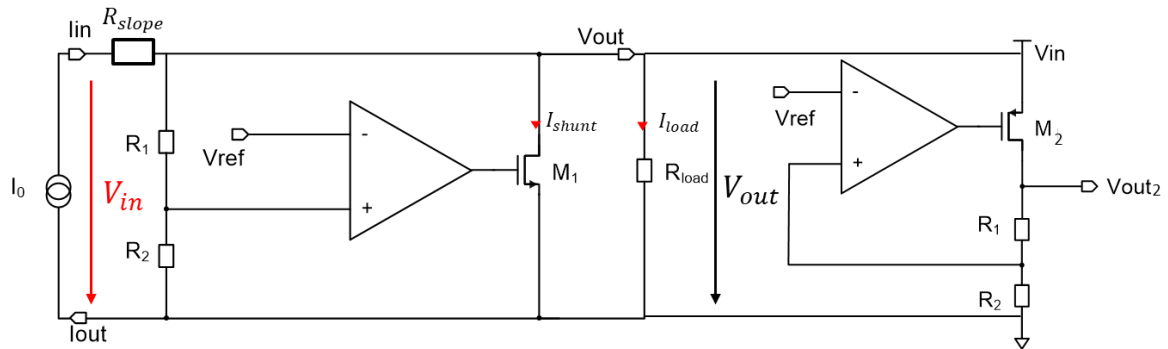


Abbildung 10: Shunt-Regler mit einem Eingangswiderstand parallel zu einem LDO-Regler [KAR5]

Durch die Integration des Eingangswiderstandes beim Regler wird der Eingang vom Ausgang entkoppelt. Im Gegensatz zu einem konventionellen Shunt-Regler bleibt die Spannung im eingeschwungenen Zustand am Eingang nicht konstant, sondern steigt mit zunehmenden Versorgungsstrom an. Dies hängt mit den linearen Eigenschaften des Widerstandes zusammen. Der große Vorteil, der hierdurch erreicht werden kann, ist, dass sich der Strom gleichmäßiger auf die parallel geschalteten Regler eines Moduls verteilt und so ein sicherer, redundanter Betrieb gewährleistet werden kann. Die im vorherigen Kapitel erwähnten Varianzen sind weiterhin präsent, sodass auch die Spannungen weiterhin voneinander abweichen, allerdings wurde die Verteilung des Stromes maßgeblich verbessert. Zur Erzeugung einer weiteren Ausgangsspannung für den Digitalteil wurde ein LDO-Regler an den Ausgang des Shunts platziert. [KAR4][KAR5]

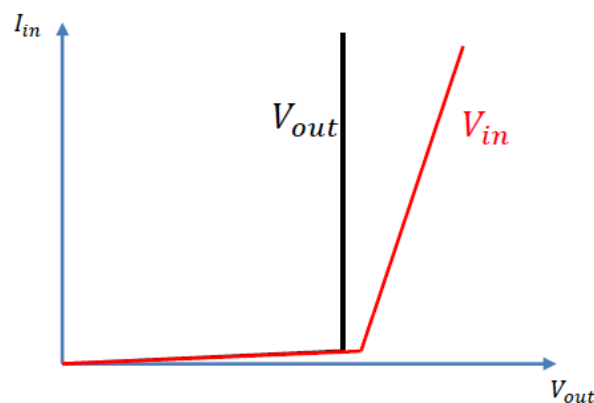


Abbildung 11: Strom/Spannungscharakteristik für einen Shunt-Regler mit einem Widerstand am Eingang [KAR5]

Abbildung 11 zeigt das Verhalten des Reglers aus Abbildung 10. Im eingeschwungenen Zustand bleibt der Ausgang konstant, die Eingangsspannung nimmt jedoch mit

zunehmendem Versorgungsstrom weiter zu. Es gilt, je größer die Steigung der Eingangsspannung, desto gleichmäßiger wird der Versorgungsstrom auf die parallelen Regler verteilt.

Das Konzept des Shunt-LDO-Reglers, welcher ausführlich im nachfolgenden beschrieben wird, basiert auf diesen gewonnenen Erkenntnissen und ist ähnlich zu der Schaltung nach Abbildung 10. Jedoch wurde der LDO-Regler an den Eingang des Shunts geschaltet, da dieser vereinfacht als regelbarer Widerstand angesehen werden kann. Der LDO-Regler generiert die gewünschte Ausgangsspannung und der Shunt wird benutzt, um den überschüssigen Strom an das nächste Modul weiterzuleiten. Die Eigenschaft, dass die Eingangsspannung bei zunehmenden Versorgungsstrom ansteigt, wurde beibehalten und zudem wurde ein konfigurierbarer Offset für die Eingangsspannung integriert. Mit diesem neuen Konzept wurden Verbesserungen in Bezug auf die steile Strom/Spannungscharakteristik des Reglers erzielt und zudem ein sicherer redundanter Betrieb erreicht. [KAR4][KAR5][SFS]

## 2.2 - Funktionsprinzip ShuntLDO-Regler

Der Shunt-LDO-Regler kombiniert die Funktionalität eines LDO-Spannungsreglers mit der Fähigkeit eines Shunt-Reglers, einen konstanten Strom zu gewährleisten. Im nachfolgenden werden die einzelnen Regelkreise des Shunt-LDO-Reglers Schritt für Schritt erklärt.

### 2.2.1 - LDO-Spannungsregler

Abbildung 12 zeigt den Low-Dropout-Regler (LDO), der eine konstante Ausgangsspannung  $V_{out}$  generiert. Die Schaltung setzt sich zusammen aus dem PMOS Pass Device  $M_1$ , dem Verstärker  $A_1$  sowie dem resistiven Spannungsteiler  $R_1$  und  $R_2$ , der das Rückkopplnetzwerk bildet. Die Ausgangsspannung wird geregelt, indem der Stromfluss durch das Pass Device über die Gate-Spannung gesteuert wird.

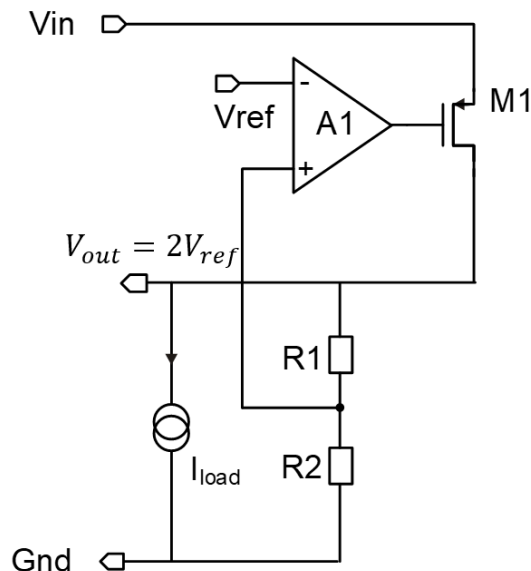


Abbildung 12: LDO-Regler [KAR4]

Die Dropout-Spannung über dem Pass-Transistor ergibt sich aus der Spannungsdifferenz zwischen der Eingangs- und Ausgangsspannung. Vorteilhaft ist, dass der LDO bereits bei einem geringen Spannungsabfall über  $M_1$  funktioniert. Die Ausgangsspannung wird über eine negative Rückkopplung auf den nichtinvertierenden Eingang des Verstärkers zurückgeführt und mit der Referenzspannung verglichen. Ziel der Regelung ist es, eine konstante Ausgangsspannung  $V_{out}$  einzustellen, welche nicht von Variationen der Eingangsspannung oder plötzlichen Lastwechseln abhängig ist. Die Widerstände wurden so gewählt, dass die halbe Ausgangsspannung am Eingang des Verstärkers anliegt. Die zurückgeführte Spannung wird sich auf Grund des Prinzips des virtuellen Kurzschlusses auf die Referenzspannung  $V_{ref}$  einstellen. Damit ergibt sich  $V_{out}$  aus der doppelten Referenzspannung.

$$V_{out} = 2 \cdot V_{ref}$$

(2.2\_1)

## 2.2.2 - Shunt-Regler

Der Shunt-Regler setzt sich zusammen aus dem Shunt-Transistor  $M_4$  und dem Verstärker  $A_3$ . Der Transistor  $M_4$  zieht den überschüssigen Strom, der nicht von der Last aufgenommen wird, nach Masse bzw. zum nächsten Modul.

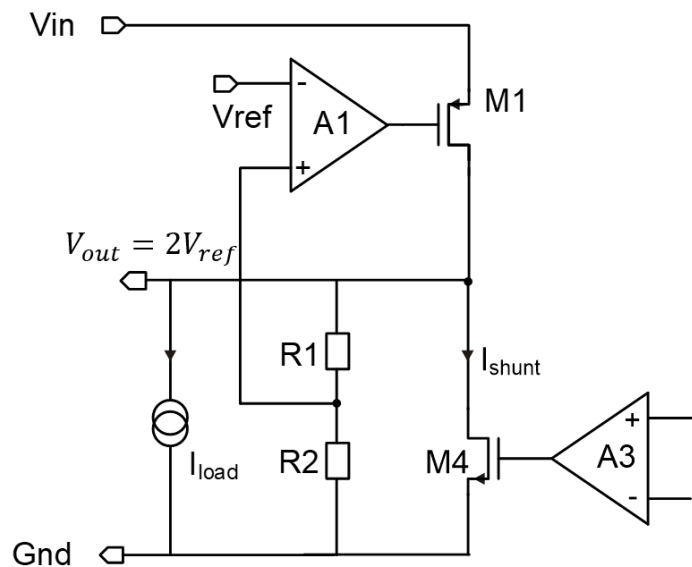


Abbildung 13: Shunt-Regler [KAR4]

Der Strom durch den Pass-Transistor  $M_1$  teilt sich in den Laststrom  $I_{load}$  und den Shunt-Strom  $I_{shunt}$  auf. Für die Steuerung des Shunt-Stroms ist eine Detektion des tatsächlichen Stromflusses und der Vergleich mit einer Referenzgröße nötig, welche im Folgenden beschrieben werden.

## 2.2.3 - Stromspiegel zur Detektion des Stromflusses

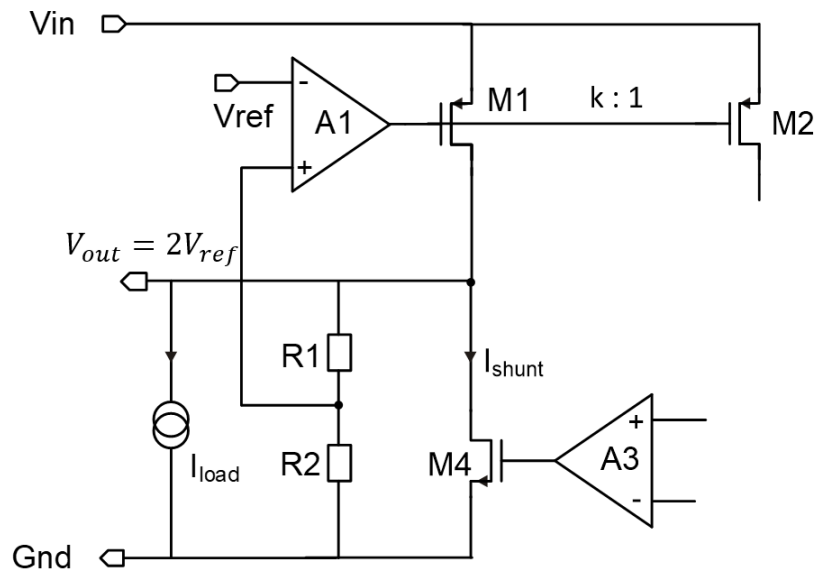


Abbildung 14: Stromspiegel [KAR4]

Für die Regelung des Shunt-Stroms wird der Strom durch den Pass-Transistor  $M_1$  detektiert, welcher dem Stromfluss durch den Regler entspricht. Hierfür wird der Strom über einen Stromspiegel zunächst auf den Transistor  $M_2$  gespiegelt. Die Breite des Transistors  $M_2$  wurde im Vergleich zum Transistor  $M_1$  um einen Faktor 1000 kleiner gewählt, womit auch der Strom um den gleichen Faktor geringer ist. Über die Transistorgleichung für den Strom wird der Zusammenhang ersichtlich. Dabei ist nachfolgend die Gleichung für den Sättigungsbereich mit Vernachlässigung der Kanallängenmodulation aufgeführt:

$$I_D = \frac{1}{2} \cdot \mu \cdot C_{ox} \cdot \frac{W}{L} \cdot (U_{GS} - U_{th})^2 \quad (2.2\_2)$$

Durch die Variation der Breite  $W$  verändert sich der Strom durch den Transistor proportional zur Transistorbreite. Besitzen zwei Transistoren die gleiche Gate-Source-Spannung  $U_{GS}$  und die gleichen Transistorlängen  $L$ , wie das bei den Transistoren  $M_1$  und  $M_2$  der Fall ist, stellt sich ein Drain-Stromverhältnis  $I_{D2}/I_{D1}$  an den Transistoren ein, das hauptsächlich durch das Breitenverhältnis der Transistoren  $W_2/W_1$  bestimmt ist.

## 2.2.4 - Verbesserung der Spiegelgenauigkeit

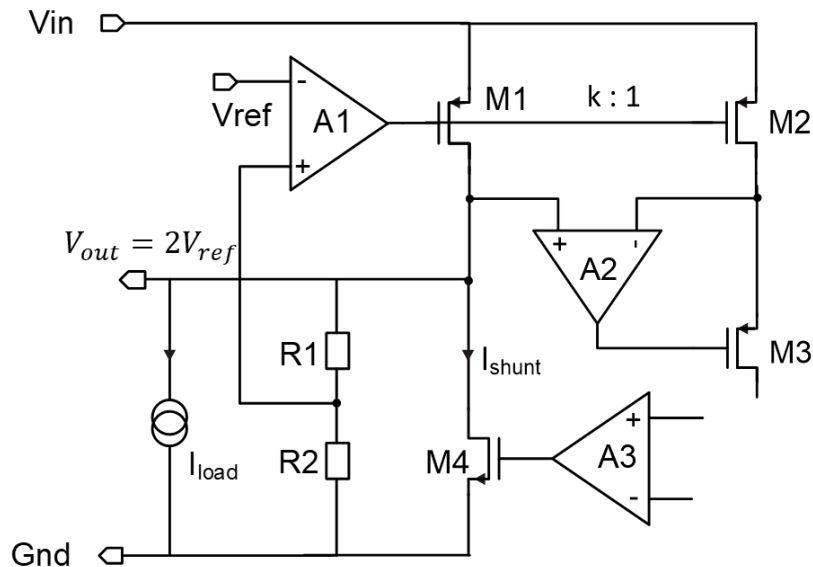


Abbildung 15: Verbesserung der Spiegelgenauigkeit [KAR4]

Simulationen haben gezeigt, dass Probleme mit der Stromspiegelgenauigkeit während der Power-Up-Phase des Reglers auftreten, wenn keine zusätzlichen Maßnahmen ergriffen werden. Fällt nur eine kleine Drain-Source-Spannung  $U_{DS}$  über den Transistoren  $M_1$  und  $M_2$  ab, befinden sich die Transistoren im linearen Bereich, statt in Sättigung. In diesem Arbeitsbereich ist der Strom durch die Transistoren nicht nur abhängig von der Gate-Source-Spannung, sondern auch von der Drain-Source-Spannung. Die ist aus der Transistorgleichung im linearen Bereich erkennbar, für die gilt:

$$I_D = \mu \cdot C_{ox} \cdot \frac{W}{L} \cdot [(U_{GS} - U_{th}) \cdot U_{DS} - U_{DS}^2] \quad (2.2\_3)$$

Das hat zur Folge, dass durch  $M_1$  und  $M_2$  unterschiedliche Ströme fließen. Zur Verbesserung der Stromspiegelgenauigkeit wurden der Verstärker  $A_2$  sowie der Transistor  $M_3$  in die Schaltung integriert. Transistor  $M_3$  arbeitet als Source-Folger und der Verstärker wird verwendet, um das Prinzip des virtuellen Kurzschlusses auszunutzen. Das Drain-Potential des Pass-Transistors  $M_1$  wird auf den nichtinvertierenden Eingang von  $A_2$  gegeben und das Drain-Potential von  $M_2$  auf den invertierenden. Der Ausgang des Verstärkers liegt an der Gate-Elektrode des Transistors  $M_3$ . Da der Strom durch  $M_3$  von  $M_2$  vorgegeben wird, ändert sich bei einem Anstieg oder Abfall des Gate-Potentials auch das Source-Potential. Die Rückwirkung des Verstärkers sorgt demnach dafür, dass sich die Potentiale am invertierenden und nichtinvertierenden Verstärkereingang angleichen und dadurch das Potential am Drain-Anschluss von  $M_2$  identisch zu dem Drain-Potential des Transistors  $M_1$  ist.

### 2.2.5 - Referenzspannung für den invertierenden Eingang von A3

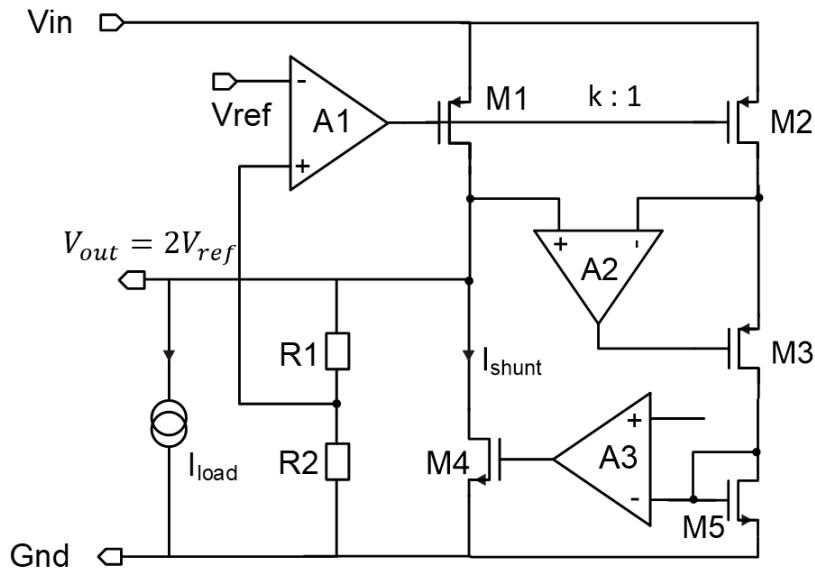


Abbildung 16: Referenzwert für den invertierenden Eingang des Verstärkers  $A_3$  [KAR4]

Wie zuvor beschrieben, fließt durch den Transistor  $M_2$  und  $M_3$  eine um den Faktor 1000 geteilte Kopie des Reglereingangsstroms. Über den Gate-Drain verbundenen Transistor  $M_5$  wird dieser Strom in eine entsprechende  $U_{GS}$  Spannung gewandelt. Anhand der Gleichung 2.2\_2 ergibt sich der folgende Zusammenhang:

$$U_{GS} = \sqrt{\frac{2 \cdot I_D}{\mu \cdot C_{ox} \cdot \frac{W}{L}}} + U_{TH} \quad (2.2_4)$$

Wie der Gleichung 2.2\_4 zu entnehmen ist, hängt die Spannung  $U_{GS}$  des Transistors  $M_5$  maßgeblich vom Stromfluss durch den Transistor  $M_5$  ab. Die Gate-Source-Spannung wird schließlich an den Verstärker  $A_3$  geführt, um den Regelkreis für den Shunt-Strom zu schließen.



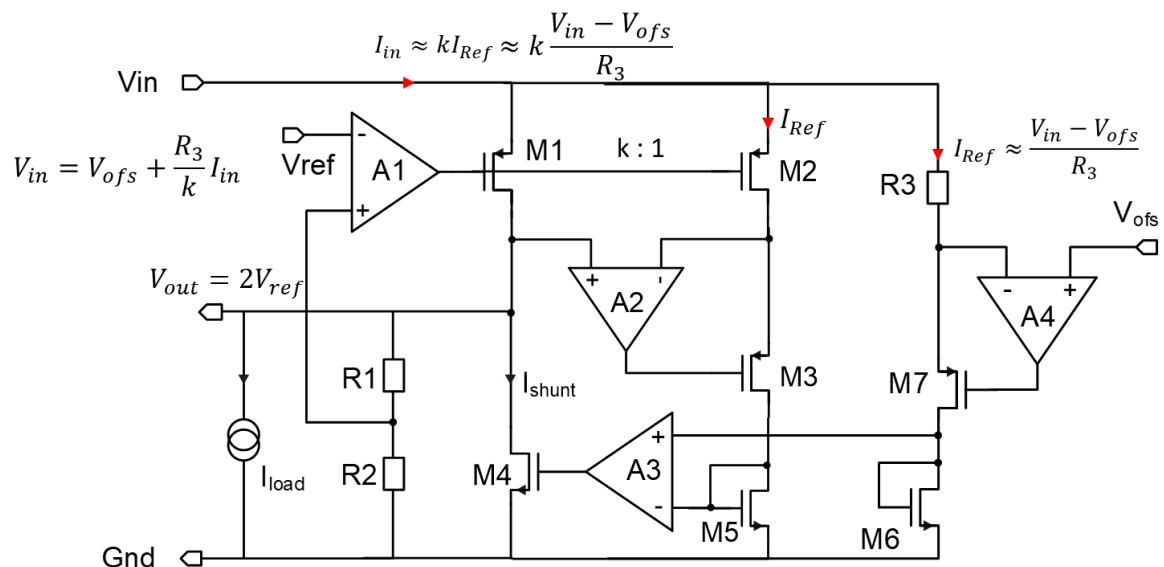
2.2.6 - Definition des Referenzstroms  $I_{ref}$ 

Abbildung 17: Gesamtes Schaltbild des Shunt-LDO-Reglers [KAR4]

Für eine funktionierende Regelung wird jedoch noch ein Sollwert benötigt, d.h. eine Spannung die als Referenzwert für den zu erzielenden Stromfluss durch den Regler herangezogen wird. Diese wird im dritten Pfad der Schaltung über den Widerstand  $R_3$  definiert. Der obere Anschluss von  $R_3$  ist mit der Versorgungsspannung  $V_{in}$  verbunden, während am unteren Anschluss die Offsetspannung  $V_{ofs}$  anliegt. Die Offsetspannung  $V_{ofs}$ , welche von der Bandgapspannung (Kapitel 2.4) generiert wird, wird über den Verstärker  $A_4$  sowie den Transistor  $M_7$  in die Schaltung an der unteren Elektrode des Widerstandes  $R_3$  eingepreßt. Die Schaltung aus Verstärker  $A_4$  und Transistor  $M_7$  arbeitet ähnlich wie  $A_2$  und  $M_3$  ebenfalls auf der Basis eines Source-Folgers und unter Ausnutzung des Prinzips des virtuellen Kurzschlusses, wodurch die Spannung am invertierenden Eingang eine Spannung gleich der Spannung am nichtinvertierenden Eingang einstellt. Der Referenzstrom ergibt sich aus dem Spannungsabfall über  $R_3$  zu:

$$I_{ref} = \frac{V_{in} - V_{ofs}}{R_3} \quad (2.2_5)$$

Der Gate-Drain verbundene Transistor  $M_6$  wandelt den Referenzstrom in eine äquivalente Spannung, welche schließlich als Referenzwert der Stromregelung verwendet und an den nichtinvertierenden Eingang des Verstärker  $A_3$  geführt wird.

Über den Verstärker  $A_3$  wird der Referenzstrom  $I_{ref}$  mit dem um den Faktor 1000 skalierten Reglereingangsstrom verglichen. Liegt eine positive Differenzspannung an den Eingängen an, dann vergrößert sich die Spannung am Ausgang des Verstärkers, wodurch die Gate-Source-Spannung des Shunt-Transistors  $M_4$  zunimmt und der Shunt-Transistor einen größeren Strom zieht. Dadurch steigt der Reglereingangsstrom an, bis schließlich der skalierte Reglereingangsstrom den gleichen Wert wie der Referenzstrom erreicht hat. Falls eine negative Differenzspannung am Verstärker anliegt, bedeutet dies, dass der Reglereingangsstrom höher als gewünscht ist. In diesem Fall sinkt die Ausgangsspannung des Verstärkers und damit die Gate-Source-Spannung des Transistors  $M_4$ , wodurch der

Transistor einen geringeren Shunt-Strom zieht, bis der Reglereingangstrom den gewünschten Wert erreicht hat.

Der Regler wurde in einer 65 nm Technologie von TSMC entwickelt. Dabei kommen ausschließlich Transistoren mit dünnem Gate-Oxid zum Einsatz, die eine hohe Strahlenfestigkeit aufweisen. Da die Transistoren aufgrund ihres dünnen Gate-Oxids jedoch nur Spannungen von maximal 1.32 V standhalten, wurde der Spannungsregler kaskodiert aufgebaut. Das Pass Device bildet hier jedoch eine Ausnahme, da eine Kaskodierung zu einer Erhöhung der minimalen Drop-Out-Spannung des Reglers führen würde, was eine Absenkung der Reglereffizienz bedeuten würde.

### 2.3 - Spannungs-/Strom-Charakteristik des Shunt-LDO-Reglers

Abbildung 19 zeigt den Verlauf der Eingangs- und Ausgangsspannung über dem Versorgungsstrom. Die Eingangsspannung weist eine resistive Charakteristik auf, die eine gleichmäßige Stromverteilung auf die parallelen Regler ermöglicht. Über den Widerstand  $R_3$  wird die Steigung definiert und die konfigurierbare Offsetspannung ist äquivalent zur Eingangsspannung bei niedrigen Versorgungsströmen. Vereinfacht kann der Shunt-LDO mit einem Widerstand und einer Spannungsquelle dargestellt werden, siehe Abbildung 18 links.

In Abbildung 18 rechts ist eine alternative Ersatzschaltung des Reglers zu sehen, und zwar eine Zener-Diode in Reihe zu einem Widerstand. Die Spannungsquelle bzw. die Zenerdioden-Spannung entspricht dabei der Offsetspannung  $V_{ofs}$ , während der Reihenwiderstand dem effektiven Eingangswiderstand des Reglers entspricht, der über den Widerstand  $R_3$  eingestellt werden kann. [KAR4]

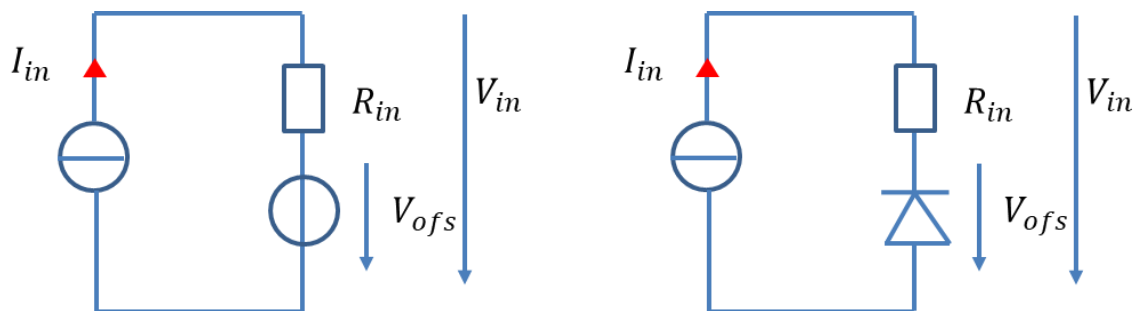


Abbildung 18: Shunt-LDO-Ersatzschaltung mit einem Widerstand in Reihe zu einer Spannungsquelle (links) und mit einer Zener-Diode in Reihe zu einem Widerstand (rechts) [KAR4]

Die nachfolgende Gleichung kann durch Umformung nach der Eingangsspannung des Reglers aufgelöst werden.

$$V_{in} = V_{ofs} + \frac{R_3}{k} \cdot I_{in} \quad (2.3\_1)$$

Der Graph der Gleichung entspricht einer Geraden, wobei der Faktor  $R_3/k$  der Steigung der Geraden und  $V_{ofs}$  von dem Y-Achsenabschnitt entspricht.

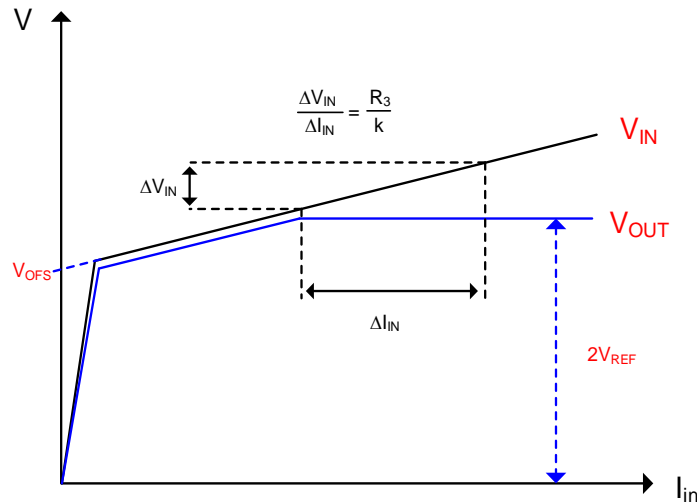


Abbildung 19: Kurvenverlauf Shunt-LDO V/I Charakteristik [KAR4]

Abbildung 19 zeigt den Spannungs-Stromcharakteristik Verlauf des Shunt-LDO-Reglers. Die Ausgangsspannung folgt dem Eingang, bis der Ausgang konstant ist. Die Differenz im Anstieg zwischen den beiden Verläufen entspricht der Dropout-Spannung des Pass Devices. Hierbei ist zu erkennen, dass die Differenz sehr gering ausfällt. Wenn der eingeschwungene Zustand erreicht wird, steigt die Eingangsspannung weiterhin an, um die gleichmäßige Stromverteilung auf die parallel geschalteten Regler zu ermöglichen. Der Widerstand  $R_3$  definiert dabei wie bereits beschrieben die Steigung (Slope) der Eingangsspannung  $V_{in}$ . [KAR4][ORF2]

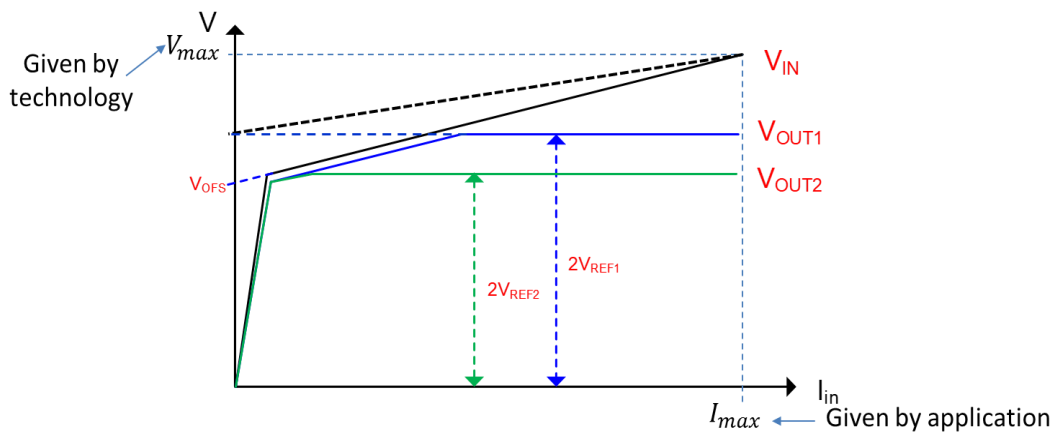


Abbildung 20 Spannungsverlauf von zwei parallel geschalteten Shunt-LDO-Reglern [KAR4]

Abbildung 20 zeigt ein Beispiel mit zwei Reglern, die unterschiedliche Ausgangsspannungen generieren. Ein Regler generiert die Spannung für den analogen Schaltungsteil und der andere die Spannung für den digitalen. Der Verlauf zeigt, dass die Offset- und Ausgangsspannungen für jeden Regler unabhängig voneinander gewählt werden können.

## 2.4 - Bandgap-Referenzspannungsschaltung

Wie in Kapitel 2.2 beschrieben, benötigt der Regler die beiden Referenzsignale  $V_{ref}$  und  $V_{ofs}$ . Diese Referenzsignale werden mit Hilfe einer Bandgapschaltung erzeugt.

Bandgap-Referenzspannungsquellen werden verwendet, um Spannungen zu generieren, die von der Temperatur und der Versorgungsspannung unabhängig und konstant sind. Die Namensgebung der Schaltung soll darauf hindeuten, dass sich das Funktionsprinzip dieser Schaltung die sogenannte Bandlücke (engl. bandgap) von Halbleitermaterialien zur Nutzen macht, welche sowohl bei Isolatoren als auch Halbleitern bei einer Darstellung im Bändermodell, zwischen dem Leitungs- und dem Valenzband liegt. Die Bandlücke bestimmt, wie viel Energie aufgewendet werden muss, um ein Elektron aus der Festkörperbindung herauszulösen. Diese Bandlückenenergie ist nahezu temperaturunabhängig. Diese Eigenschaft wird für die Generierung einer temperaturstabilen Referenzspannung ausgenutzt.

Da die Schaltung in einer Umgebung mit hoher Strahlung eingesetzt wird, müssen strahlenharte Transistoren für den Kern der Bandgap-Spannung, als auch für die Steuerlogik der Schaltung verwendet werden. In der 65 nm CMOS-Technologie von TSMC besitzen lediglich Dünn-Gate-Oxid-Transistoren die benötigte Strahlungshärte. Daraus folgt, dass die Spannung der Referenzschaltung stärker von Prozessschwankungen abhängig ist, als das z.B. der Fall wäre, wenn Dioden oder Bipolartransistoren im Bandgap-Kern verwendet würden. Das macht es notwendig die Bandgap-Schaltung nach der Produktion zu kalibrieren, um die Prozessschwankungen auszugleichen. Hierfür werden digitale Signale, sogenannte Trimbits verwendet, welche durch digitale Gatter und Schalter in die Bandgap-Schaltung einwirken können. Diese Transistoren sind jedoch nur für eine maximale Spannungen von bis zu 1.32 V ausgelegt. Die Eingangsspannung  $V_{in}$  kann relativ hohe Werte bis zu 2 V annehmen. Die für das Trimming genutzten digitalen Schaltungen können jedoch nicht mit 2 V betrieben werden.

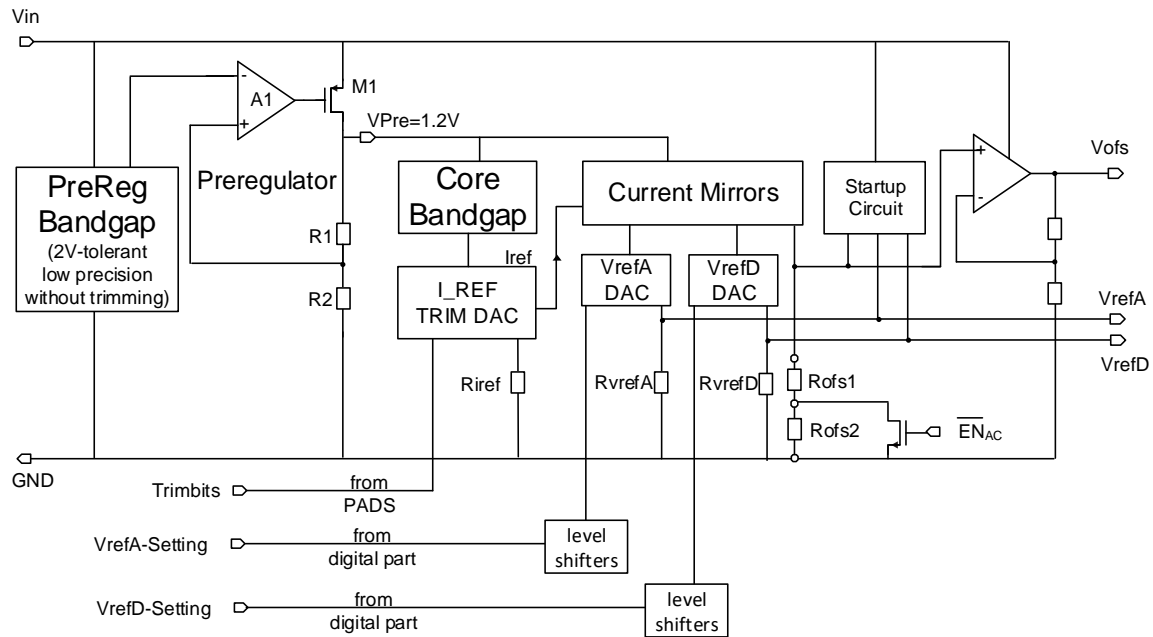


Abbildung 21: Bandgap Schaltung [KAR4]

Um zeitgleich Strahlungsfestigkeit, die Toleranz von bis zu 2 V Versorgungsspannungen und hohe Präzision durch Justierbarkeit zu erzielen, wurde eine Architektur mit zwei Bandgaps und einem Vorregler gewählt (siehe Abbildung 21). Die beiden Bandgaps führen die Bezeichnung „PreReg Bandgap“ und „Core Bandgap“. Die PreReg Bandgap wird mit der unregulierten Eingangsspannung versorgt, die Werte bis zu 2 V annehmen kann. Die Bezeichnung PreReg Bandgap ist eine Abkürzung für „Pre Regulator Bandgap“ und wurde so gewählt, weil sie eine Referenzspannung für einen Vorregler liefert. Die PreReg Bandgap erzeugt eine Spannung von 600 mV, welche jedoch auf Grund von Prozessschwankungen um bis zu  $\pm 25$  mV variieren kann.

Der Vorregler (Pre Regulator) ist ein LDO-Regler, welcher eine Spannung von 1.2 V mit einer Variation von  $\pm 50$  mV für die zweite Bandgap generiert. Da die Core Bandgap einen niedrigen Versorgungsspannungsdurchgriff im Bereich zwischen 1 V und 1.3 V aufweist, ist die Variation der PreReg Bandgap bzw. die Variation der Vorregler-Spannung, die dadurch verursacht wird, unproblematisch.

Durch die niedrige Versorgungsspannung der Core-Bandgap, ist es möglich, die Referenz durch digitale Gatter und Schalter zu kalibrieren, um Prozessschwankungen zu kompensieren und eine hohe Präzision zu erreichen. In der derzeitigen Ausführung müssen die Trimbits über Jumper auf der Platine konfiguriert werden, so dass eine Justierung der Bandgapspannung vor einer Messung manuell vorgenommen werden muss. Diese temperaturstabile und präzise Spannungsreferenz der Core-Bandgap wird verwendet, um durch einen externen Widerstand mit verschwindend geringem Temperaturkoeffizienten einen temperaturstabilen Strom zu generieren.

Der externe Widerstand  $R_{I_{ref}}$  beträgt 150 k $\Omega$  und erzeugt bei einer Referenzspannung von 0.6 V einen Strom  $I_{ref}$  von 4.0  $\mu$ A. Der Strom  $I_{ref}$  wird im Chip auf vielfältige Weise verwendet, um beispielsweise die Arbeitspunkte von Transistorschaltungen einzustellen. Außerdem wird dieser Strom verwendet, um die Referenzspannung des Shunt-LDO-Reglers  $V_{ref}$  zu erzeugen. Der Referenzstrom wird hierfür im Chip durch Stromspiegel repliziert und bei den Testchips auf 4  $\mu$ A bzw. bei den Produktionschips RD53B auf 20  $\mu$ A skaliert. Dieser Strom



Steckbrücken (Jumper) auf der Testplatine vorgenommen. Der eingestellte, variable Strom wird von dem konstanten Strom überlagert und kann hierdurch einen größeren oder einen niedrigeren Strom bzw. Spannungsabfall am Widerstand erzeugen. Wenn nun in einem Überlastfall der Shunt-Strom eine Schwelle unterschreitet, wird der variable Stromanteil statt in den Widerstand, in einen alternativen Strompfad über den Transistor M16 umgeleitet. Die Unterschreitung des Shunt-Stroms wird durch das folgende Schaltungsprinzip detektiert:

Die Transistoren M4 und M15 sind über einen Stromspiegel mit dem Faktor 1:2160 miteinander verbunden. Der Transistor M15 ist dabei um den Faktor 2160 kleiner als der Transistor M4 und wird mit der gleichen Gate-Source-Spannung wie M4 angesteuert. Hierdurch wird der Strom durch M15 um den gleichen Faktor von 2160 verringert, wie der Strom durch den Shunt-Transistor M4. Dieser skalierte Strom wird über die Stromquelle mit einem Wert von  $3\ \mu\text{A}$  verglichen. Dieser Wert wird erreicht, wenn im Shunt-Transistor ein Strom von nahezu  $10\ \text{mA}$  fließt. Wenn der skalierte Strom die Schwelle von  $3\ \mu\text{A}$  unterschreitet, fließt ein Teil des Stromes, den die Stromquelle einprägt, durch den Transistor M13, welcher entsprechend der Verschaltung in den Pfad nach M16 gespiegelt wird. Dies hat zur Folge, dass ein Teil des variablen Stroms, der durch den DAC eingestellt wird, nicht mehr in den Widerstand zur Generierung der Referenzspannung fließt, sondern vom Transistor M16 aufgenommen wird. Dadurch wird je nach tatsächlicher Größe des Shunt-Stroms der variable Stromanteil komplett umgeleitet und nur noch der konstante Stromanteil von  $2,3\ \mu\text{A}$  bei den Testchips bzw.  $11,5\ \mu\text{A}$  bei den RD53B Produktionschips in den externen Widerstand ( $150\ \text{k}\Omega$  Testchip bzw.  $30\ \text{k}\Omega$  RD53B) geführt. Somit sinkt die Referenzspannung  $V_{\text{ref}}$  auf den benötigten Wert von  $350\ \text{mV}$ .

Dadurch stellt sich eine Ausgangsspannung von etwa  $700\ \text{mV}$  ein. Bei dieser Ausgangsspannung von etwa  $700\ \text{mV}$  fällt auch bei der maximal erlaubten Reglereingangsspannung von  $2\ \text{V}$  nicht mehr als  $1.32\ \text{V}$  über das Pass Device des Reglers ab, was andernfalls bei höheren Spannungsabfällen beschädigt werden könnte. Die Referenzspannung wird, wie im Kapitel 3.2 beschrieben, über einen externen Widerstand erzeugt, durch den ein temperaturstabiler Strom fließt, der mit Hilfe der Bandgap-Schaltung und eines weiteren externen Widerstands generiert wird. Dieser Strom besteht aus einem konstanten- und einem variablen Anteil. Der konstante Strom beträgt  $2,3\ \mu\text{A}$  (Testchip) bzw.  $11,5\ \mu\text{A}$  (Produktionschip RD53B), während der variable Widerstand über einen Digital/Analog-Wandler mit Stromausgang eingestellt werden kann, der aus dem Digitalteil des Chips gesteuert wird.

## 2.6 - Überspannungsschutz (OVP)

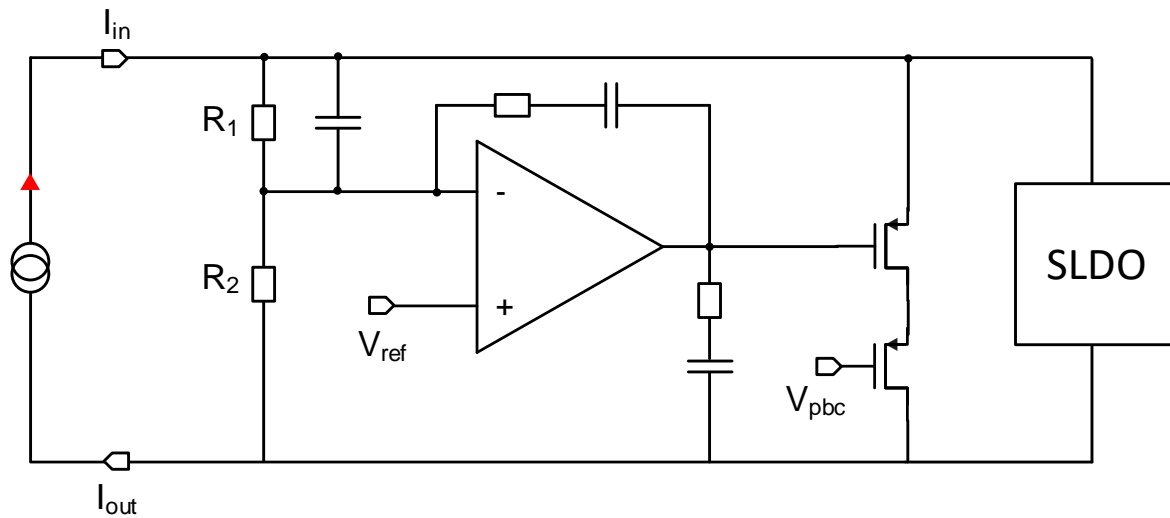


Abbildung 23: Schaltung der Overvoltage Protection [KAR4]

Neben der Überlastschutz-Schaltung existiert ebenfalls eine Schaltung zum Schutz vor Überspannungen (engl. Overvoltage Protection), welche die Funktion einer Spannungsklemme (engl. Voltage Clamp) auf einen Wert von 2 V übernimmt. Die Eingangsspannung des Reglers wird begrenzt, indem die Impedanz des Reglers bei Bedarf reduziert wird. Die verringerte Impedanz soll eine erhöhte Belastung der Spannungsquelle zur Folge haben und durch den endlichen Innenwiderstand der Quelle, die Spannung der Quelle reduzieren und damit die Überspannung verhindern.

Wie in Abbildung 23 abgebildet, wird dies realisiert, indem der Shunt-LDO-Regler parallel zu einem konventionellen Shunt-Regler geschaltet wird. Der Shunt-Regler regelt die Eingangsspannung des SLDO auf 2 V, indem der Stromfluss durch den Shunt-Transistor  $M_1$  gesteuert wird. Die Eingangsspannung des SLDO wird durch den Spannungsteiler aus den Widerständen  $R_1$  und  $R_2$  geformt und um den Faktor 0.3 verringert. Die erzeugte Spannung wird mit der Referenzspannung des Vorreglers  $V_{refPre}$  von 600 mV verglichen.

Nachfolgend werden zwei Betriebsfälle erläutert, die unterschieden werden müssen.

Fall 1: Die angelegte Spannung ist größer oder ungefähr gleich 2 V. Dementsprechend wird die runtergeteilte Spannung größer als die Referenzspannung. Das hat zur Folge, dass die Spannung am invertierenden Eingang größer als die Spannung am nichtinvertierenden Eingang ist. Hierdurch liegt am Operationsverstärker eine negative Differenzspannung an, was zur Folge hat, dass die Spannung am Ausgang des Verstärkers sinkt. Die Source-Gate-Spannung des PMOS wird hierdurch größer, was zu einem größeren Stromfluss durch den Shunt-Transistor  $M_1$  führt. Das hat zur Folge, dass der Eingangsstrom am SLDO-Regler vorbei nach Masse geleitet wird und somit die SLDO-Spannung absinkt. Das System regelt sich so ein, dass die skalierte zurückgeführte Spannung genauso groß wird wie die angelegte Referenzspannung, was zeitgleich bedeutet, dass die Eingangsspannung exakt 2 V erreicht.



Fall 2: Die Spannung am Shunt-Regler, welche gleichzeitig auf Grund der parallelen Verschaltung der Eingangsspannung des SLDO-Reglers entspricht, ist kleiner als 2 V. Somit ist die skalierte Spannung am invertierenden Eingang des Operationsverstärkers kleiner als die angelegte Referenzspannung. Die Spannung am nichtinvertierenden Eingang ist demnach größer als die Spannung am invertierenden Eingang, was zur Folge hat, dass eine positive Differenzspannung am Eingang des Operationsverstärkers vorherrscht. Eine positive Differenzspannung wiederum bewirkt, dass der Ausgang des Operationsverstärkers steigt und die Source-Gate-Spannung des Shunt-Transistors  $M_1$  und damit auch der Strom durch den Shunt-Transistor kleiner wird. Liegt die Eingangsspannung des SLDO weit unter 2 V, wird der Shunt-Transistor vollkommen geöffnet und der Strom fließt nur noch in den SLDO-Regler hinein. Das ist im nominellen Betrieb der Fall, bei dem die Eingangsspannung des SLDO-Reglers 1.4 V beträgt. Die Differenzspannung des Operationsverstärkers nimmt in diesem Fall sehr große Werte an, wodurch das Operationsverstärker-Ausgangssignal auf die Versorgungsspannung des Verstärkers ansteigt und die Source-Gate-Spannung des Shunt-Transistors 0 V beträgt. Das bedeutet, dass der konventionelle Shunt-Regler im nominellen Betrieb nicht eingreift und der SLDO die alleinige Regelung übernimmt - Im Normalbetrieb ist demnach die „Overvoltage Protection“ abgeschaltet.

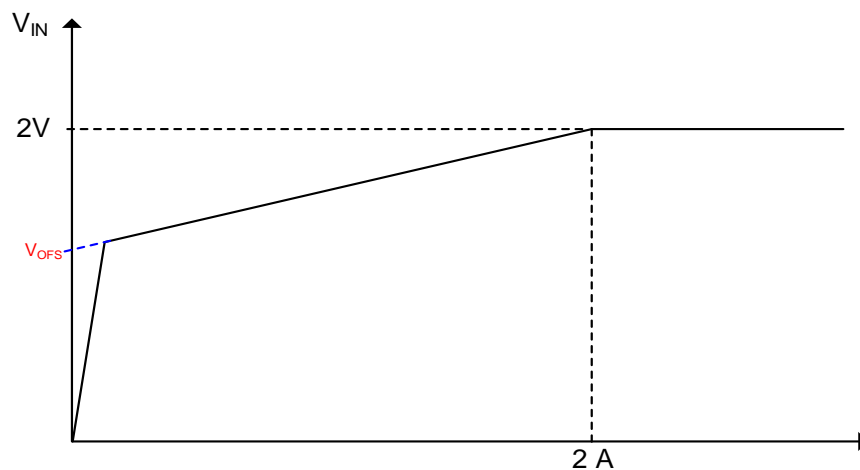


Abbildung 24: Kurvenverlauf Überspannungsschutz [KAR4]

Bei aktiviertem Überspannungsschutz erhält man die in Abbildung 24 dargestellte Eingangsspannung-Charakteristik. Bis zu der Grenze von 2 V ist ausschließlich der SLDO aktiv. Die Eingangsspannung nimmt zu Beginn den Wert der Offset-Spannung an und mit zunehmenden Strom steigt diese linear an. Erreicht die Eingangsspannung den Wert von 2 V greift die Überspannungsschutzschaltung ein und die Eingangsspannung wird auf 2 V gehalten. Die Schaltung besitzt hierbei eine Festigkeit bis maximal 2 A Eingangsstrom. Bei größeren Strömen ist die Limitierung der Eingangsspannung auf 2 V nicht garantiert und Schaltungsdefekte dementsprechend nicht ausgeschlossen. [BAN1]

## 2.7 - AC-Gleichrichter / Low- und High-Power-Modus

Während der Installation, bzw. während eines Wartungsprozesses wird der Chip ohne die volle Kühlleistung betrieben. Für diesen Fall wurde ein Low-Power-Modus (LPM) implementiert. Der LPM besitzt eine Offsetspannung von 1.3 V, wodurch der Stromverbrauch und die generierte Wärme reduziert werden.

Im regulären Betrieb wird der High-Power-Modus (HPM) verwendet. Hierbei wird der Chip mit einer Offsetspannung von 800 mV oder höher betrieben.

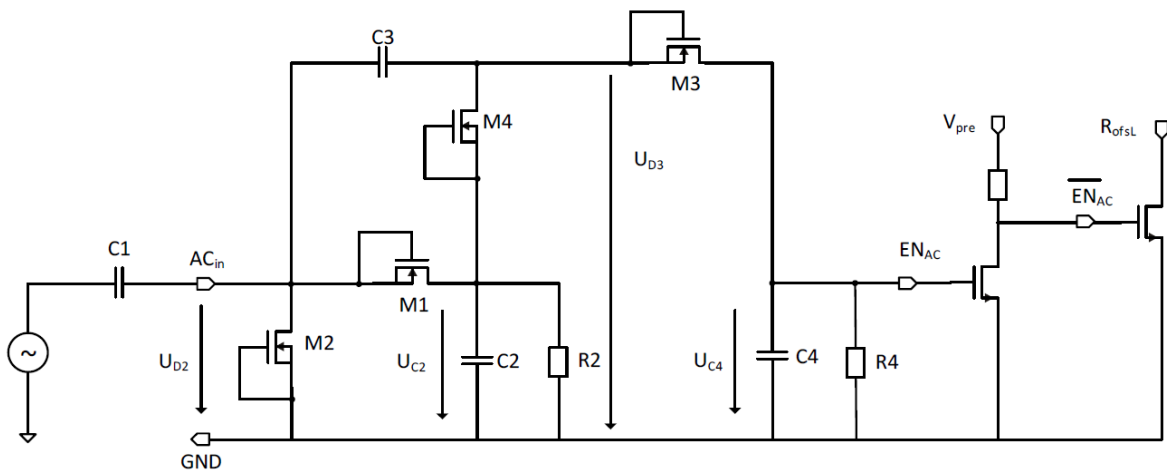


Abbildung 25: Aufbau der AC-Gleichrichter-Schaltung

Bei der Durchführung von Systemtests steht während der Installationsphase des Detektors keine aktive Kühlung zur Verfügung. Damit ist die Leistung begrenzt, welche ohne thermische Schädigung von den Chips aufgenommen werden kann und liegt weit unter der Leistung, welche im nominellen Fall während der Laufzeit des Experimentes mit aktiver Kühlung erreicht wird. Während der nominelle Reglereingangsstrom bei 1-2 A liegt, ist während der Installationsphase nur ein Stromfluss von 150 mA erlaubt. Aus diesem Grund benötigt der Regler einen Betriebsmodus, bei dem der eingeschwungene Zustand schon bei diesen kleineren Strömen erreicht wird. Dies wird realisiert indem für die Generierung des Offsetspannung  $V_{ofs}$  der Widerstand  $R_{ofs2}$  bzw.  $R_{ofsL}$  mit Hilfe des AC-Gleichrichters hinzugeschaltet wird. Liegt kein AC-Signal an, wird der Widerstand  $R_{ofsL}$  durch einen Transistor mit Masse kurzgeschlossen (siehe Abbildung 25). Ist der als MOS Transistor realisierte Schalter geöffnet, vergrößert sich der Gesamtreihenwiderstand und damit auch der Spannungsabfall über dem Gesamtwiderstand, welcher der Offsetspannung  $V_{ofs}$  entspricht. Im nominellen Betrieb (High-Power-Modus) ist der Transistor  $EN_{AC}$  jedoch eingeschaltet und der Transistor schließt den unteren der beiden ersten Widerstände kurz. Die Offsetspannung  $V_{ofs}$  wird dementsprechend über den Widerstand  $R_{ofs1}$  bzw.  $R_{ofsH}$  eingestellt, wodurch die Spannung auf einen kleineren Wert absinkt. [ZOR]

## **3. Messsystem**

### **3.1 - Motivation**

In diesem Kapitel wird ein Überblick über das gesamte Messsystem gegeben, wie es für die Validierung des ShuntLDO-Reglers zum Einsatz kommt. Mit Hilfe der Messergebnisse wird die Funktionalität des Reglers unter verschiedenen Betriebsbedingungen überprüft. So besteht unter anderem die Möglichkeit den Regler mit Hilfe einer konventionellen Line- oder Load-Regulation im statischen Fall zu charakterisieren sowie transiente Untersuchungen, wie z.B. die Reaktion des Reglers auf schnelle Lastwechsel oder Variationen der Eingangsspannung bzw. des Eingangsstroms zu betrachten. Das Ziel hierbei ist es, dem Leser die gesamte Messinfrastruktur vorzustellen und die Einarbeitung zu vereinfachen. Hierfür werden zunächst die Hardwarekomponenten der Platine sowie zusätzliche für die Messung relevante Elektronik und neu hinzugekommene Features vorgestellt. Im weiteren Verlauf folgt eine Einführung in die Messgeräte sowie die Messsoftware, die für die Kommunikation mit den Messinstrumenten und für die Aufzeichnung statischer Messergebnisse verwendet wird. Auf den Grundlagen des Messsystems aufbauend, gibt das darauffolgende Kapitel eine Übersicht auf Beispiele durchgeführter Messungen.

### **3.2 - Anforderungen an das Messsystem**

Das Messsystem sollte so gestaltet sein, dass es dem Benutzer erlaubt, in einer anwendungsfreundlichen Umgebung eine möglichst große Anzahl an Messungen automatisiert durchzuführen. Hierdurch kann gewährleistet werden, dass neue Versionen des ShuntLDO-Reglers möglichst einfach und zeitnah validiert werden können.

Für die Ausführung einiger statischer Messungen, ist die Kommunikation mit den Geräten zur Konfiguration und Auslesung, die Einstellungsmöglichkeiten innerhalb der Messsoftware und die Darstellung der aufgenommenen Messergebnisse erforderlich. Durch eine Erweiterung der Platine mit digitalen Potentiometern sowie eine Weiterentwicklung der Messsoftware wird die Automatisierung weiter vorangetrieben. Die neue Funktion bietet die Möglichkeit, die Steigung und den Offset der Eingangsspannung über das Testtool in einem bestimmten Bereich zu variieren.

### 3.3 - Hardwarekomponenten

#### 3.3.1 - Leiterplatte für den SLDO-Testchip

Der Aufbau der aktuellen Leiterplatte (TSC 3, Rev. 0.3) für die Validierung des Testchips ist in den Abbildung 26 und 27 jeweils in einer zweidimensionalen Darstellung sowie in den Abbildungen 28 und 29 in einer dreidimensionalen Ansicht dargestellt. Die Spannungen der Platine können entweder bezogen auf das Chip Potenzial oder das Massepotenzial der Platine gemessen werden. Im letzten Fall fließt der Spannungsabfall über den Bonddrähten des Mikrochips in die Messungen mit ein. Da die Ausgangsspannung des Reglers für die Versorgung integrierter Schaltungen des RD53-Chips vorgesehen ist, sind die Messergebnisse bezogen auf die Chip Masse von höherer Relevanz.

#### 3.4.1 - Darstellung der Oberseite

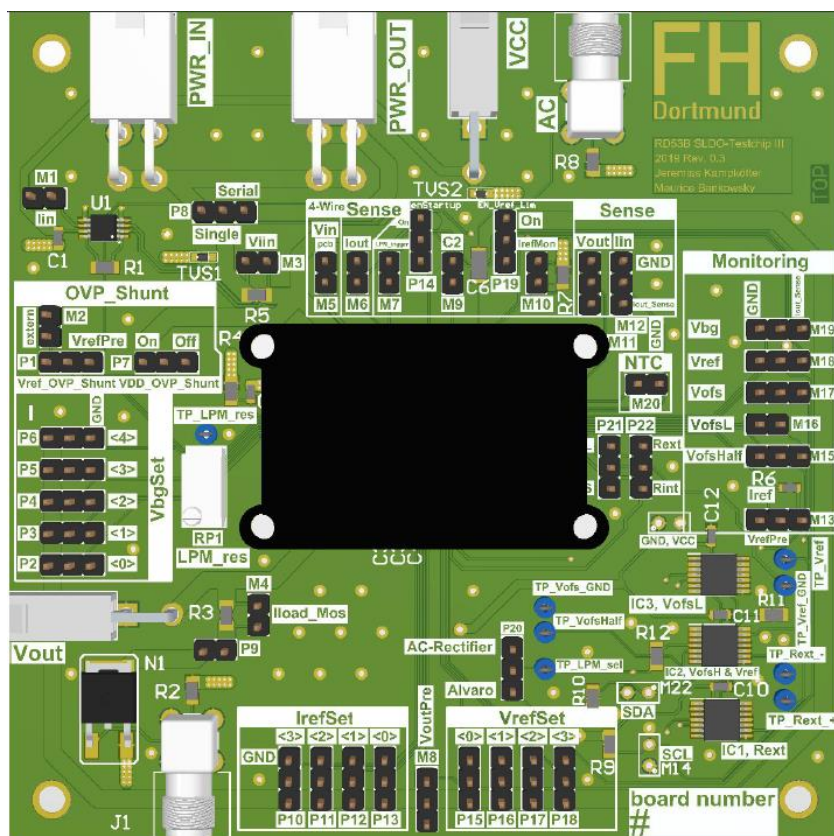


Abbildung 26: Aufbau Leiterplatte SLDO (RD53B) Testchips III, 2019 Revision 0.3 . Ansicht von oben, mit Kappe

Die Platine (Abbildung 26) besitzt unter anderem mehrere Molex-Anschlüsse. An PWR\_IN wird der Versorgungsstrom bzw. die Versorgungsspannung angeschlossen, an VCC eine Spannung für die Versorgung zusätzlicher Komponenten auf der Platine, über V<sub>out</sub> kann ein Laststrom gezogen werden und über PWR\_OUT kann eine Verbindung zu weiteren Reglern für den seriellen Betrieb etabliert werden. Zusätzlich sind einige Steckverbinder mit zwei oder drei Stiftkontakten vorhanden, um entweder Spannungen zu messen oder

zusätzliche Schaltungen auf dem Chip oder der Platine zu aktivieren bzw. zu deaktivieren. Die Stiftkontakte mit der Bezeichnung M werden zum Abgriff von Spannungssignalen verwendet und an den Kontakten mit der Bezeichnung P werden die zusätzlichen Schaltungen mit Hilfe von Jumpers aktiviert. Zum Schutz der sehr dünnen Bonddrähte ist eine schwarze Abdeckung auf der Platine verschraubt. An die beiden SMA-Kontakte (J1, AC) können Wechselsignale mit einem Waveformgenerator eingespeist werden, welche die Ansteuerung des Transistors N1 bzw. die Aktivierung des Low-Power-Modus ermöglichen. Eine Übersicht aller Anschlussmöglichkeiten der Platine ist in Tabelle 1 gegeben.

### 3.4.2 - Darstellung der Unterseite

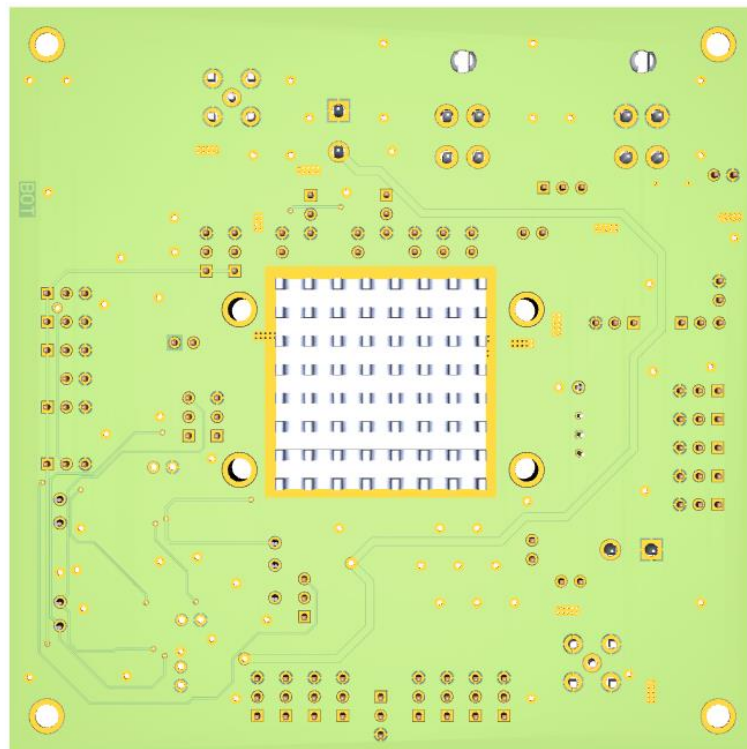


Abbildung 27: Ansicht der Leiterplatte von unten (CAD optimierte Beleuchtung ausgewählt)

Abbildung 27 zeigt die Rückseite der Platine, die großflächig als Masse Potential dient. Zusätzlich sind einige Leiterbahnen zu erkennen, die aufgrund von Platzgründen auf der Unterseite geführt werden. Die quadratische Fläche in der Mitte befindet sich direkt unterhalb des Chips und ist mit einer goldenen Oberfläche beschichtet. Auf die goldene Oberfläche wird ein Kühlkörper angebracht, um Wärme bei hohen Versorgungsströmen abzuführen und damit Temperatureffekte zu reduzieren.

3.4.3 - 3D-Ansicht

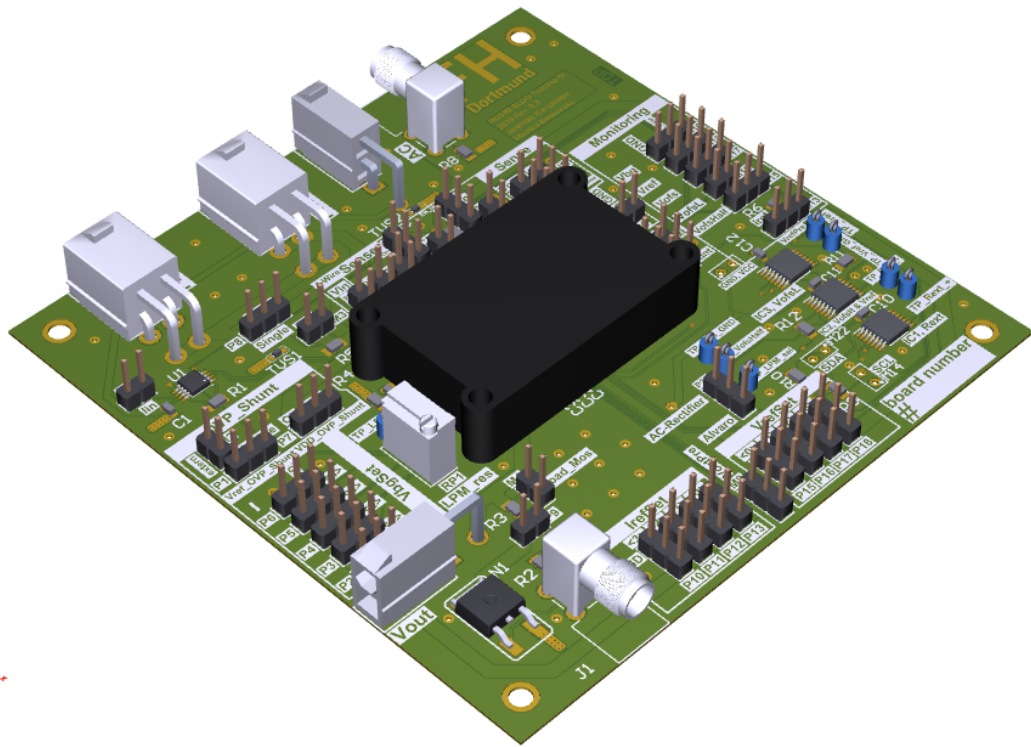


Abbildung 28: RD53B 3D-Ansicht seitlich (CAD optimierte Beleuchtung ausgewählt)

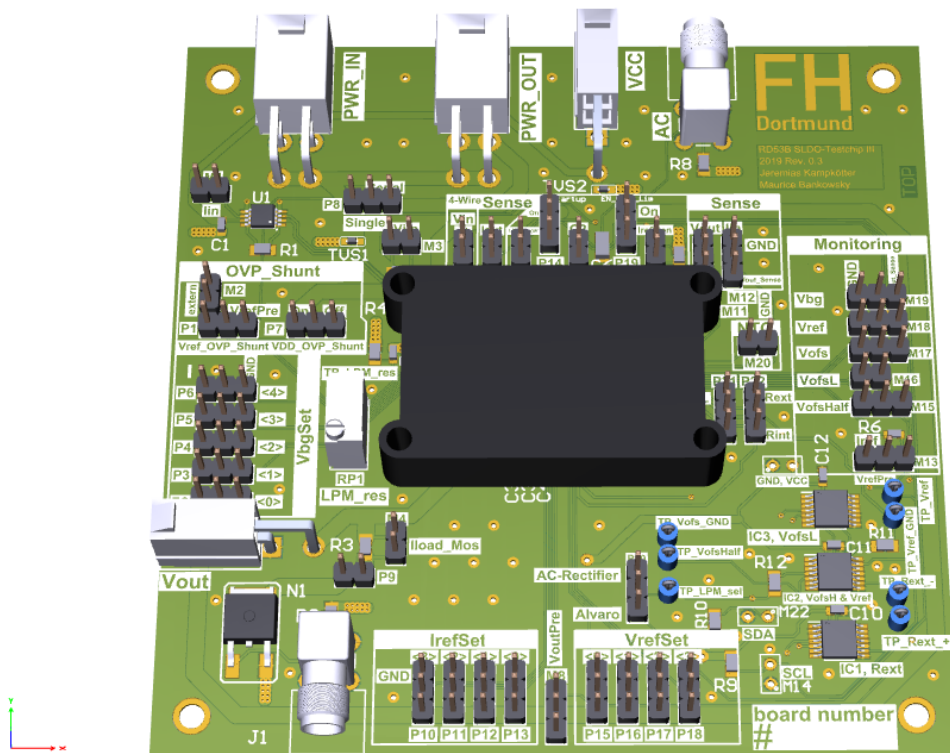


Abbildung 29: RD53B 3D-Ansicht frontal (CAD optimierte Beleuchtung ausgewählt)

Tabelle 1: Auflistung der Anschlüsse: Leiterplatte SLDO (RD53B) [ZOR]

| Anschluss  | Bedeutung der Anschlüsse   |  |
|------------|--|--|
| PWR_IN     | Anschluss der Versorgungsquelle  |  |
| PWR_OUT    | Anschluss weiterer seriell geschalteter Regler   |  |
| VOUT       | Anschluss einer Last   |  |
| VCC        | Anschluss einer Spannungsversorgung für den Instrumentenverstärker U1 und für die Versorgung der digitalen Potentiometer |  |
| AC         | Anschluss eines AC-Signals für die Aktivierung des Low-Power-Modus   |  |
| J1         | SMA Anschluss zur Ansteuerung des Gates des Transistors N1 zur Generierung schneller Lastpulse                           |  |
| P1         | Auswahl zwischen einer internen und einer externen Referenz für die OVP  |  |
| P2, P3, P4 | VbgSet - Bit 0, Bit 1, Bit 2   | Trimm-Bits der Core-Bandgap der Bandgapschaltung |
| P5, P6     | VbgSet - Bit 3, Bit 4  |  |
| P7         | Aktivierung Overvoltage Protection   |  |
| P8         | Auswahl Verschaltung Single / Serial   |  |
| P9         | Verbindet den Transistor N1 mit dem Ausgang $V_{out}$ zur Generierung schneller Lastpulse                                |  |
| P10, P11   | IrefSet - Bit 3, Bit 2   | Trimm-Bits des Stromgenerators                   |
| P12, P13   | IrefSet - Bit 1, Bit 0   |  |
| P14        | Aktivierung Startup-Schaltung  |  |
| P15, P16   | VrefSet - Bit 0, Bit 1   | Trimm-Bits der Referenzspannung                  |
| P17        | Aktivierung Undershunt Current Protection  |  |
| P18        | Auswahl zwischen dem AC-Rectifier und der Alvaro-Schaltung für die Aktivierung des Low-Power-Modus                       |  |
| P19, P20   | VrefSet - Bit 2, Bit 3   | Trimm-Bits der Referenzspannung                  |
| P21        | Wahl zwischen dem LDO und SLDO Betrieb   |  |
| P22        | Wahl zwischen dem internen und externen Widerstand der die Steigung der Eingangsspannung bestimmt                        |  |
| M1         | Ausgangsspannung Verstärker U1 zur Messung des Eingangsstroms  |  |
| M2         | Anschluss externe Referenzspannung OVP   |  |
| M3         | Messung Eingangsspannung Verstärker U1 entsprechend Eingangsstrom bzw. Spannungsabfall R5                                |  |

|     |  |
|-----|--|
| M4  | Messung Spannungsabfall R3 des NMOS-Transistors                                    |
| M5  | Messung Eingangsspannung (Anschluss 4-Wire)  |
| M6  | Messung Potential Chip-Ground  |
| M7  | Messung LPM-Trigger (Alvaro)   |
| M8  | Messung Vorregler-Ausgangsspannung $V_{prereg}$                                    |
| M9  | Messung gleichgerichtetes Signal des AC-Rectiers                                   |
| M10 | Messung Spannung entsprechend $I_{refMon} * R7$                                    |
| M11 | Messung Ausgangsspannung $V_{out}$   |
| M12 | Messung Eingangsspannung $V_{in}$  |
| M13 | Messung Vorregler-Referenzspannung $V_{refPre}$                                    |
| M14 | Anschluss SCL vom FTDI Mini Modul  |
| M15 | Messung Offsetspannung $V_{ofshalf}$   |
| M16 | Messung Offsetspannung $V_{ofsL}$  |
| M17 | Messung Offsetspannung $V_{ofs}$   |
| M18 | Messung Referenzspannung $V_{ref}$   |
| M19 | Messung Bandgapspannung $V_{bg}$   |
| M20 | Messung der Temperatur über NTC1   |
| M21 | Anschluss / Messung VCC und GND  |
| M22 | Anschluss SDA vom FTDI Mini Modul  |
| RP1 | Einstellung Widerstand LPM (Alvaro)  |
| IC1 | Digitaler Potentiometer zum Einstellen des Widerstands $R_{ext}$                   |
| IC2 | Digitaler Potentiometer zum Einstellen des Widerstands $V_{ofshalf}$ und $V_{ref}$ |
| IC3 | Digitaler Potentiometer zum Einstellen des Widerstands $V_{ofsL}$                  |

Die Bauteilbezeichnungen der Leiterplatte des ShuntLDO-Testchips sind farblich kodiert, um die Übersicht zu verbessern. Die Anschlüsse zum Setzen der Trimm-Bits sind grün, die Messkontakte sind orange, die Potentiometer zur Einstellung der Widerstandswerte der Referenzen sind blau und die Anschlüsse zum Wechsel der Betriebsmodi bzw. zur Aktivierung von Schutzschaltungen sind rot hervorgehoben.



### 3.3.2 - Leiterplatte Prepower

Die Prepower-Platine ermöglicht die Generierung schneller Eingangsstromwechsel mit sehr kurzen Anstiegszeiten im Nanosekundenbereich. Die Platine wird zwischen dem Testchip und dem Sourcemeter verschaltet. Über die SMA-Anschlüsse J1 bzw. J2 können die NMOS-Transistoren auf der Platine angesteuert werden. Sobald die Transistoren geschlossen sind, fließt ein Teil des Versorgungsstromes nach Masse und nicht in den Regler. Durch die Ansteuerung der Transistor Gates mit einem Waveformgenerator, werden Eingangsstromwechsel generiert. Mit der Prepower-Platine können bis zu zwei Regler gleichzeitig versorgt werden. Die Abbildungen 30 bis 33 zeigen unterschiedliche Ansichten der Leiterplatte. In der Tabelle 2 sind die Funktionen der Anschlüsse aufgelistet. [ZOR]

### 3.5.1 - Darstellung der Vorderseite

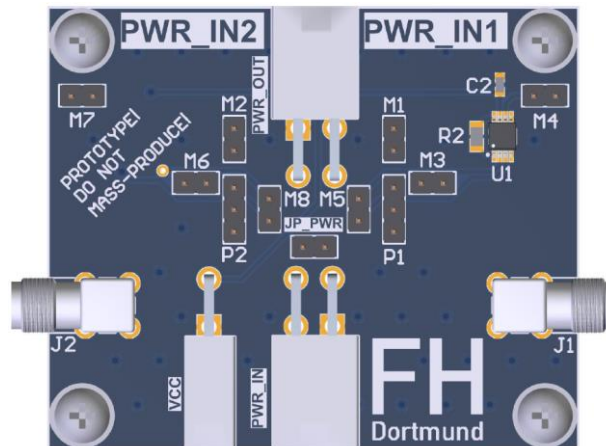


Abbildung 30: Aufbau Leiterplatte SLDO

### 3.5.2 - Darstellung der Rückseite

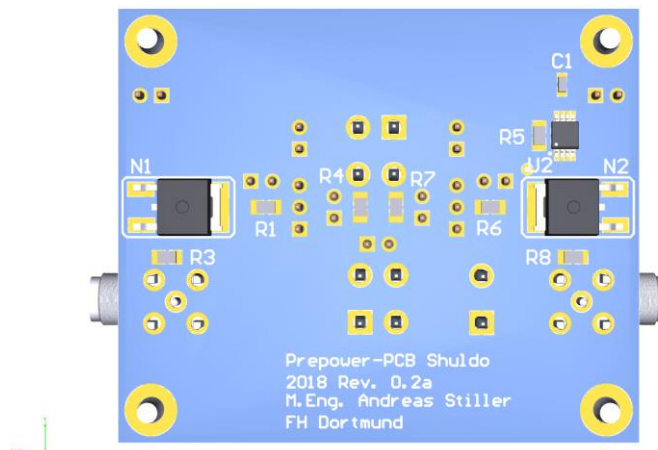


Abbildung 31: Aufbau Leiterplatte SLDO Bottom-Ansicht (CAD optimierte Beleuchtung ausgewählt)

3.5.3 - 3D-Ansicht

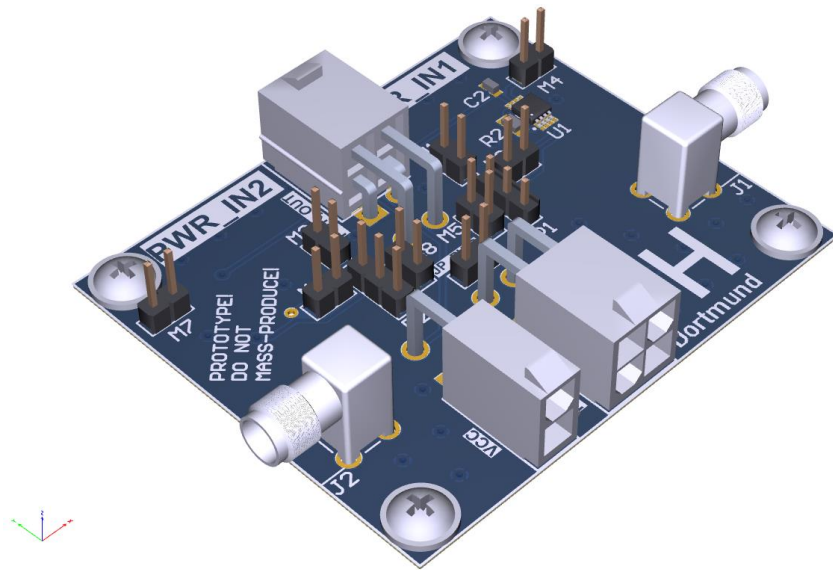


Abbildung 32: 3D-Ansicht seitlich (CAD optimierte Beleuchtung ausgewählt)

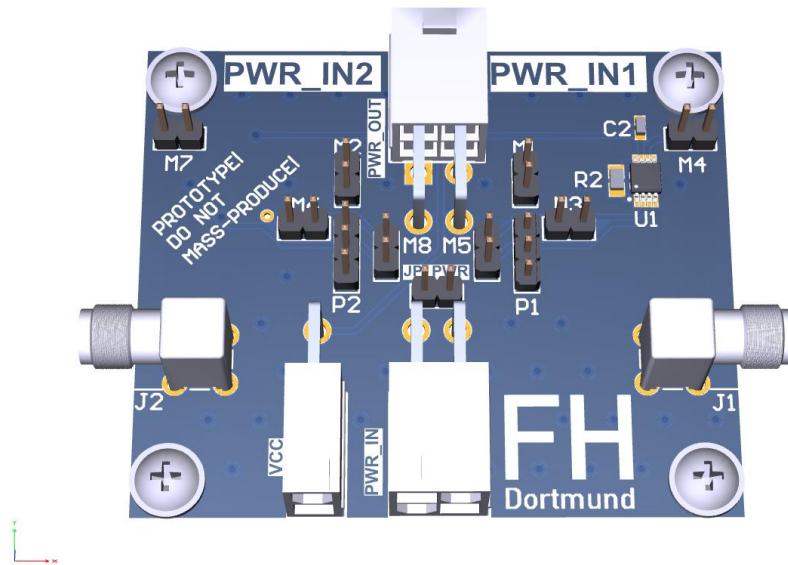


Abbildung 33: 3D-Ansicht frontal (CAD optimierte Beleuchtung ausgewählt)

Tabelle 2: Auflistung der Anschlüsse der Prepower-Platine [ZOR]

| <b>Anschluss</b> | <b>Bedeutung des Anschlusses</b>  |
|------------------|---|
| PWR_IN           | Anschluss Spannungsversorgung   |
| PWR_OUT          | Anschluss Testchip  |
| VCC              | Anschluss Spannungsversorgung Verstärker U1 & U2  |
| J1               | Anschluss Gate Transistor N1  |
| J2               | Anschluss Gate Transistor N2  |
| JP_PWR           | Aktivierung beider Schaltungspfade simultan   |
| P1,P2            | Auswahl Messung vor/hinter Shuntwiderstand  |
| M1               | Messung Reglerspannung  |
| M2               | Messung Reglerspannung  |
| M3               | Messung Spannungsabfall R1 des NMOS-Transistors   |
| M4               | Ausgangsspannung Verstärker U1 zur Messung des Eingangsstroms                             |
| M5               | Messung Eingangsspannung Verstärker U1 entsprechend Eingangsstrom bzw. Spannungsabfall R4 |
| M6               | Messung Spannungsabfall R6 des NMOS-Transistors   |
| M7               | Ausgangsspannung Verstärker U2 zur Messung des Eingangsstroms                             |
| M8               | Messung Eingangsspannung Verstärker U2 entsprechend Eingangsstrom bzw. Spannungsabfall R7 |

## 3.3.3 - Instrumentenverstärker

Der Instrumentenverstärker ist jeweils auf der Platine des Testchips sowie auf der PrePower-Platine implementiert. Der Versorgungsstrom fließt durch den Shunt-Widerstand R5 und die Spannung über diesem wird mit dem Instrumentenverstärker abgegriffen. Der Verstärkungsfaktor wurde nach Gleichung 3.3\_1 so gewählt, dass die resultierende Spannung an M1 äquivalent zum Versorgungsstrom ist.

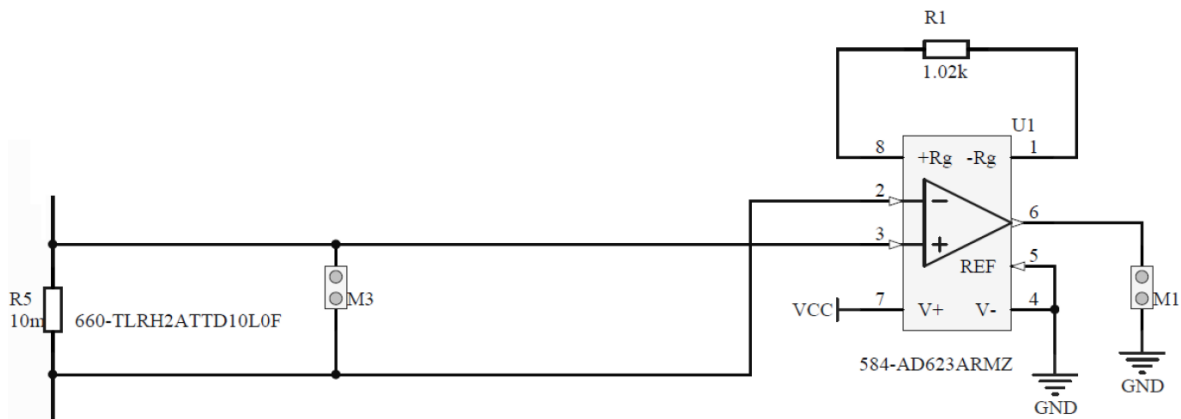


Abbildung 34: Schaltungs-Ausschnitt des Instrumentenverstärkers

$$G = 1 + \frac{100k}{R_1} = 1 + \frac{100k}{1.02k} \approx 99.04$$

(3.3\_1)

Mit Hilfe der Formel für Berechnung der Verstärkung (3.3\_1) wird ersichtlich, dass sich ein etwa um den Faktor 100 erhöhter Wert am Ausgang des Verstärkers ergibt. Der Instrumentenverstärker wird über den Anschluss VCC mit Spannungen von 3 V bis 12 V versorgt. Weitere Kenndaten finden sich in dem Datenblatt [IMV].

## 3.3.4 - Schaltung für Laststrompulse

Die Abbildung 35 zeigt die Schaltung eines NMOS-Transistors auf der Platine des Testchips und die Abbildung 36 eines NMOS-Transistors auf der Prepower-Platine. Der NMOS-Transistor in Abbildung 35 hat die Aufgabe, bei Bedarf Lastwechsel zu generieren. Die Funktion des Transistors wird mit einem Jumper auf dem Anschluss P9 aktiviert und zur Messung des dynamischen Reglerverhaltens verwendet.

Über den Anschluss J1 der Testchip-Platine wird das Gate des Transistors angesteuert. Mit Hilfe eines Waveformgenerators können periodische Rechtecksignale eingespeist und somit transiente Lastströme generiert werden.

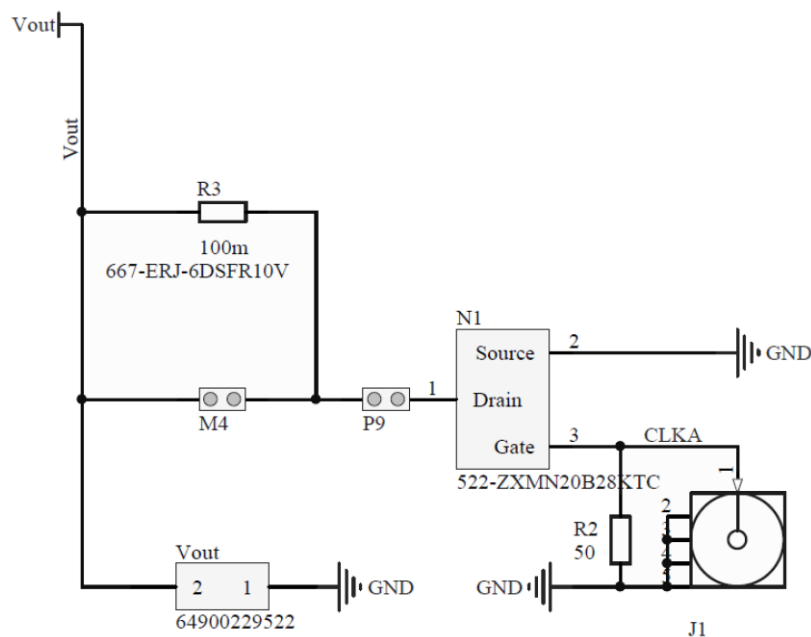


Abbildung 35: Schaltung des NMOS-Transistors auf dem Testchip

$$I_D = \frac{1}{2} \mu C_{ox} \frac{W}{L} (U_{GS} - U_{th})^2 \quad (3.3\_2)$$

Anhand der NMOS-Transistorgleichung (3.3\_2) wird ersichtlich, dass bei jeder positiven Amplitude des Signals ein Strom durch den Transistor fließt. Dieser Strom fließt über den Widerstand R3 und erzeugt dort einen Spannungsabfall, welcher über den Anschluss M4 gemessen werden kann. Des Weiteren kann zusätzlich zu den transienten Lastströmen am Ausgang  $V_{out}$  ein konstanter Strom gezogen werden. [ZOR]

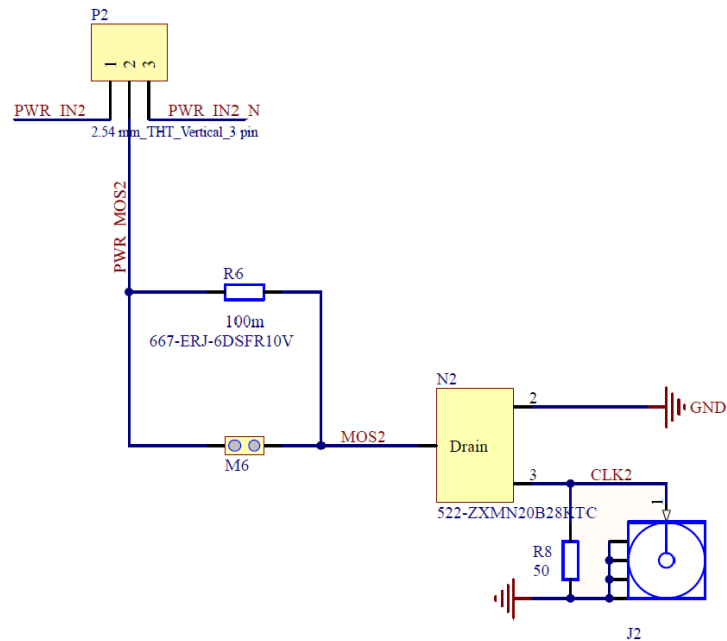


Abbildung 36: Schaltung des NMOS-Transistors auf der Prepower-Platine

Die in Kapitel 3.3.2 gezeigte Prepower-Platine beinhaltet ebenfalls einen NMOS-Transistor. Der NMOS-Transistor wird hier nach dem gleichen Prinzip wie auf der Platine des Testchips verwendet. Mit Hilfe eines Waveformgenerators kann das Gate des Transistors über den Anschluss J2 angesteuert und somit sehr schnelle transiente Eingangsströme erzeugt werden. Sobald der Transistor einen Strom zieht, fällt über dem Widerstand R6 eine Spannung ab, die über den Anschluss M6 gemessen werden kann. [ZOR]

### 3.3.5 - Digitale Potentiometer

Die Hauptplatine des Messsystems wurde um digitale Potentiometer erweitert, um die automatisierte Einstellung der Offsetspannung und der Eingangsspannung mittels der „Shuldo-Test-Messsteuerungs-Software“ zu realisieren.

Digitale Potentiometer sind mikrocontrollerkompatible Bauteile, welche die Einstellung Ihres Widerstandswertes über eine digitale Schnittstelle erlauben. Diese Schnittstelle kann beispielsweise durch einen Prozessor flexibel über Software angesteuert werden, um die benötigten Werte einzustellen. Auf Grund ihres Aufbaus werden digitale Potentiometer oft auch als resistiver Digital-Analog-Wandler (RDAC) bezeichnet. Digitalpotentiometer besitzen gegenüber elektromechanischen Potentiometern keine beweglichen Teile und emulieren die Funktionalität des elektromechanischen Potentiometers mittels integrierter Schaltkreise. Der Widerstandswert wird ermittelt, indem der Poti einen digitalen Code entgegennimmt. Die interne Topologie des Digitalpotentiometers besteht prinzipiell aus einer einfachen seriellen Anordnung von Widerständen mit digital adressierbaren elektronischen Schaltern. Zwischen dem Schleifer (Wiper, W) und den Widerständen (R) wird dementsprechend die gewünschte Wischerposition festgelegt. Dieser Vorgang wird in Abbildung 37 dargestellt. [DGK1][POT1]

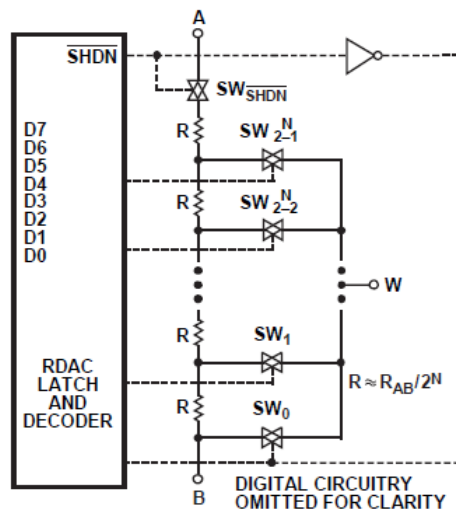


Abbildung 37: Wiper/Schrittweite des digitalen Potentiometers [POT1]

Die Abbildung 37 zeigt das Funktionsprinzip des Digitalpotentiometers, welches aus einer Widerstandskette und elektronischen Schaltern besteht, mit deren Hilfe man eine Verbindung zu einem Punkt in der Widerstandskette herstellen kann. Das Digitalpotentiometer bietet in der Regel zwischen 32 und 1024 Schritt- bzw. Stufeneinstellungen. Die für das Testsystem des SLDO ausgewählten Potis besitzen 256 Schritte (8 Bit).

Table 6.  $R_{WB}$  (D) at Selected Codes for  $R_{AB} = 10\text{ k}\Omega$

| D (DEC) | $R_{WB}$ ( $\Omega$ ) | Output State                                 |
|---------|-----------------------|--|
| 255     | 10021                 | Full-scale ( $R_{WB} = 1\text{ LSB} + R_W$ ) |
| 128     | 5060                  | Midscale                                     |
| 1       | 99                    | 1 LSB  |
| 0       | 60                    | Zero-scale (wiper contact resistance)        |

Abbildung 38: Beispielrechnung eines Digitalpotentiometers mit einem Wert von 10 k $\Omega$  [POT1]

Die Abbildung 38 zeigt beispielhaft die Kennwerte eines Digitalpotentiometers mit einem Widerstandswert von 10 k $\Omega$ . Der Poti besitzt hierbei einen festen Ohmschen-Grundwert R von 60  $\Omega$ , dementsprechend liegt dieser konstante Wert bei einer Schrittweite von 0 an und muss auch bei jedem weiteren Schritt berücksichtigt werden. Die einstellbaren Werte des Digitalpotentiometers lassen sich durch Formel 3.3\_3 anhand der Schrittweite D, des maximalen Wertes  $R_{AB}$  und des Grundwertes ( $R_W$ ) berechnen.

$$R_{WB} = \frac{D}{256} \cdot R_{AB} + R_W \quad (3.3_3)$$

In der nachfolgenden Abbildung ist die Pin-Konfiguration der digitalen Potentiometer dargestellt und Tabelle 3 enthält eine Kurzbeschreibung der Anschlüsse.

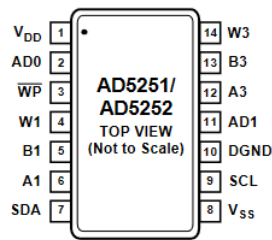


Abbildung 39: Pin-Konfiguration des AD5252 [POT1]

Tabelle 3: Funktionsbeschreibung der Anschlüsse des AD5252 [POT1]

|             |  |
|-------------|--|
| <b>VDD</b>  | Anschluss 1: Positiver Stromversorgungspin. +2.7 V bis +5 V für Einzelversorgung.                            |
| <b>AD0</b>  | Anschluss 2: I <sup>2</sup> C-Geräteadresse 0. Mit AD0 und AD1 können vier AD5252-Geräte adressiert werden.  |
| <b>WP</b>   | Anschluss 3: "Write Protection"; Schreibschutz, aktiv bei 0 = „runtergezogen“.                               |
| <b>W1</b>   | Anschluss 4: "Wiper Terminal"; Schleifer vom RDAC 1  |
| <b>B1</b>   | Anschluss 5: B Terminal vom RDAC 1   |
| <b>A1</b>   | Anschluss 6: A Terminal vom RDAC 1   |
| <b>SDA</b>  | Anschluss 7: Serieller Dateneingangs-/Ausgangs-Pin   |
| <b>VSS</b>  | Anschluss 8: Negative Versorgungspin. Anschluss an 0 V für Einzelversorgung.                                 |
| <b>SCL</b>  | Anschluss 9: Serieller Takt-Pin  |
| <b>DGND</b> | Anschluss 10: Digitale Masse. Anschluss an die analoge Systemmasse.  |
| <b>AD1</b>  | Anschluss 11: I <sup>2</sup> C-Geräteadresse 1. Mit AD0 und AD1 können vier AD5252-Geräte adressiert werden. |
| <b>A3</b>   | Anschluss 12: A Terminal vom RDAC 3  |
| <b>B3</b>   | Anschluss 13: B Terminal vom RDAC 3  |
| <b>W3</b>   | Anschluss 14: "Wiper Terminal"; Schleifer vom RDAC 3   |



### 3.4 - Erweiterung der Platine um digitale Potentiometer

#### 3.4.1 - Einführung

In den vorherigen Versionen waren auf der Platine analoge Potentiometer verbaut, die manuell eingestellt werden mussten. Durch die Integration digitaler Potentiometer kann die Einstellung der Widerstandswerte vereinfacht über die Software erfolgen.

#### 3.4.2 - Weiterentwicklung der Testplatine (Rev. 0.3)

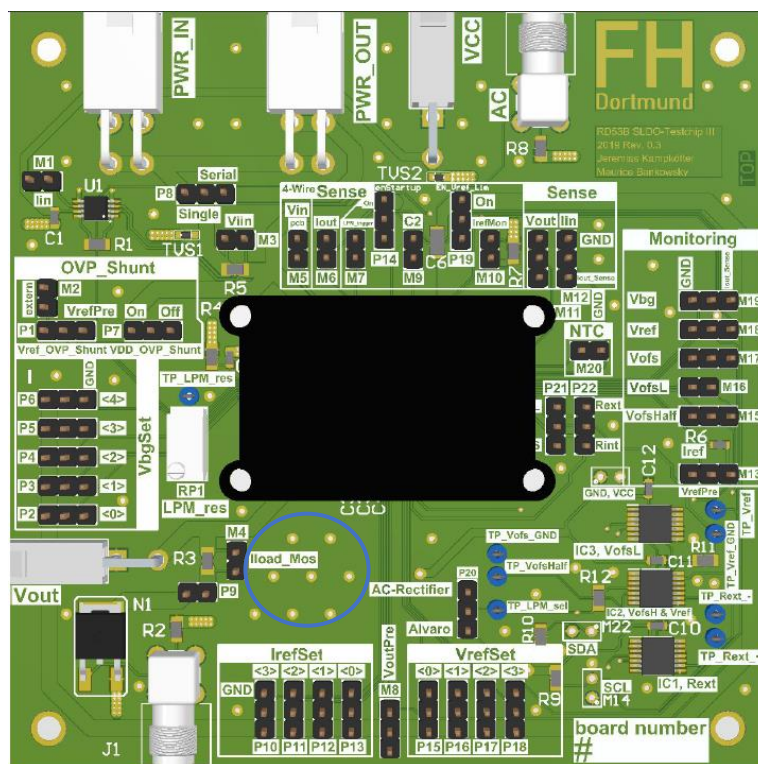


Abbildung 40: Aufbau Leiterplatte SLDO (RD53B) Testchips III, 2019 Revision 0.3 Oben-Ansicht mit Kappe

Wie in Abbildung 40 zu sehen ist, wurden in der Revision 0.3 die analogen Potentiometer an der rechten unteren Ecke auf der Vorderseite der Platine durch digitale Potentiometer ausgetauscht. An den Versorgungsspannungsanschlüssen der digitalen Potentiometer wurden jeweils Abblockkondensatoren gesetzt, um transiente Störungen zu minimieren und niederfrequente Restwelligkeiten zu filtern. Die Signalleitung SCL und die Datenleitung SDA der I<sup>2</sup>C-Schnittstelle wurden jeweils mit einem 10 k $\Omega$  Widerstand an die Versorgungsspannung VCC angeschlossen. Des Weiteren wurden die Testpunkte, zum Abgriff der Widerstandswerte, versetzt.

Eine weitere Änderung gegenüber den vorherigen Revisionen ist die eingefügte „Stitching Group“. Die Durchgangslöcher (Stitching Group, siehe blauer Kreis) verbinden die Massenflächen miteinander, welche auf der Vorder- und Rückseite aufgebracht sind.

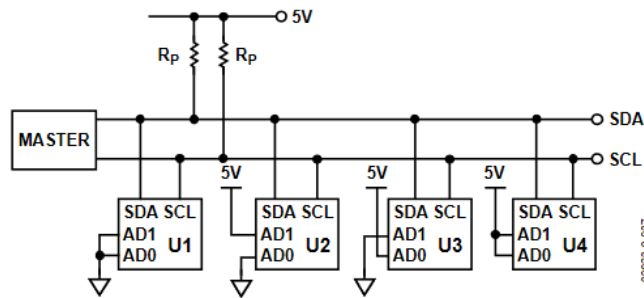


Abbildung 41: Verschaltungskonzept der digitalen Potentiometer laut Hersteller [POT1]

Abbildung 41 zeigt das Verschaltungskonzept der digitalen Potentiometer an einem gemeinsamen Bus und die Verbindung zu einem gemeinsamen Master, der über die Datenleitung (SDA) und die Signalleitung (SCL) mit den Slaves kommuniziert. Für die Kommunikation wird das I<sup>2</sup>C-Protokoll verwendet. Der I<sup>2</sup>C-Bus gehört zu der seriellen-Bus Familie und ist ein Master-Slave-Bus-System. Als Master dient ein FTDI 2232H Mini Modul, das als externes Bauteil an die Leiterplatte mittels separater Leitungen an den I<sup>2</sup>C-Bus angeschlossen wird. Alle Teilnehmer dieses Bus-Systems haben eigene Adressen. Die digitalen Potentiometer arbeiten in diesem Projekt lediglich als Empfänger.

Das FTDI 2232H Mini Modul ist ein Entwicklungsmodul, das entweder als synchrone oder asynchrone, serielle Schnittstelle oder auch als parallele FIFO Schnittstelle konfiguriert werden kann. Für den Aufbau der Verbindung zwischen Mini-Modul und PC wird ein USB-Treiber benötigt. Der Treiber und die Datenblätter werden auf der Internetseite des Herstellers [FTD] kostenlos zur Verfügung gestellt.

Eine detaillierte Anleitung zur Installation und Inbetriebnahme des Treibers sowie allgemeine Details zum FTDI 2232H Mini Modul und zum I<sup>2</sup>C-Protokoll findet sich in der Bachelor-Thesis [ÖZK1] und in der Master-Projektarbeit [ÖZK2] von Herrn Nurullah Özkan. [FTD]

| Standard-Widerstandswerte der Potis |                    |                    |                   |                   |
|-------------------------------------|--------------------|--------------------|-------------------|-------------------|
| Widerstände (Potis)                 | VofsL [ $\Omega$ ] | VofsH [ $\Omega$ ] | Vref [ $\Omega$ ] | Rext [ $\Omega$ ] |
| Wert                                | 65 k $\Omega$      | 100 k $\Omega$     | 150 k $\Omega$    | 0.6 k $\Omega$    |

Abbildung 42: Widerstandswerte für die Potentiometer im nominellen Fall

Wie in Kapitel 2.2 beschrieben, benötigt der Regler die beiden Referenzsignale  $V_{ref}$  und  $V_{ofs}$ . Diese Referenzsignale werden mit Hilfe einer Bandgapschaltung erzeugt. Für eine Ausgangsspannung  $V_{out}$  von 1.2 V muss eine Referenzspannung  $V_{ref}$  von 0.6 V generiert werden. Die Bandgap im Testchip erzeugt einen Referenzstrom von 4  $\mu$ A, sodass für die geforderte Referenzspannung im nominellen Fall ein Widerstandswert von 150 k $\Omega$  erforderlich ist.

Die Offsetspannung  $V_{ofs}$  liegt standartmäßig bei 800 mV. Damit die Bandgapschaltung eine halbe Offsetspannung  $V_{ofsHalf}$  von 400 mV generiert, ist für den nominellen Fall ein Widerstandswert von 100 k $\Omega$  gefordert.

Der Strom im Referenzpfad des ShuntLDO-Reglers wird über den Widerstand R3 ( $R_{ext}$ ) eingestellt und besitzt einen nominellen Wert von 600  $\Omega$ . Durch Variation von  $R_{ext}$  ändert sich bei gegebener Eingangsspannung der Referenzstrom in diesem Pfad, was sich auch auf die Steigung der Eingangsspannung auswirkt.

Der AC-Gleichrichter besitzt die Aufgabe, bei einer Zuschaltung eines Wechselsignals am Anschluss AC vom High-Power-Modus ( $V_{ofshalf} = 400$  mV) in den Low-Power-Modus ( $V_{ofshalf} = 650$  mV) zu wechseln. Hierbei bleibt der Widerstand  $R_{ofsh}$  auf dem Wert für den HPM von 100 k $\Omega$ , während der Wert eines weiteren Widerstands  $R_{ofsl}$  auf 62.5 k $\Omega$  (entsprechend 250 mV) eingestellt wird. Sobald der Low-Power-Modus aktiv wird, sind beide Widerstände in Reihe geschaltet und es ergibt sich ein Gesamt-Reihenwiderstand von 162.5 k $\Omega$ , der eine Offsetspannung  $V_{ofs}$  von 1.3 V einprägt.

Für die ermittelten Werte wurden die nachfolgenden digitalen Potentiometer ausgewählt:

$R_{ext}$ : AD5252BRUZ1;  $V_{ofsh}$  &  $V_{ref}$ : AD5252BRUZ100;  $V_{ofsl}$ : AD5241BRUZ100

$V_{ofsh}$ ,  $V_{ref}$  und  $V_{ofsl}$  sind digitale Potentiometer mit einem maximalen Widerstandswert von 100 k $\Omega$ . Da für die Einstellung von  $V_{ofsh}$  und  $V_{ref}$  Widerstände benötigt werden, die größer als 100 k $\Omega$  sind, werden den digitalen Potentiometern auf der Platine Widerstände mit einem festen Widerstandswert von 75 k $\Omega$  in Reihe geschaltet, wodurch ein maximaler Wert von 175 k $\Omega$  erreicht werden kann. Weitere Informationen sind in den Datenblättern [POT1] und [POT2] nachzulesen. Das Digitalpotentiometer für die Einstellung von  $V_{ofsl}$  besitzt ebenfalls einen maximalen Widerstandswert von 100 k $\Omega$ . Das Digitalpotentiometer für  $R_{ext}$  besitzt wie in den Revision zuvor einen maximalen Widerstandswert von 1 k $\Omega$ .

### 3.5 - Versuchsaufbau: Konzept zur Integration digitaler Potentiometer

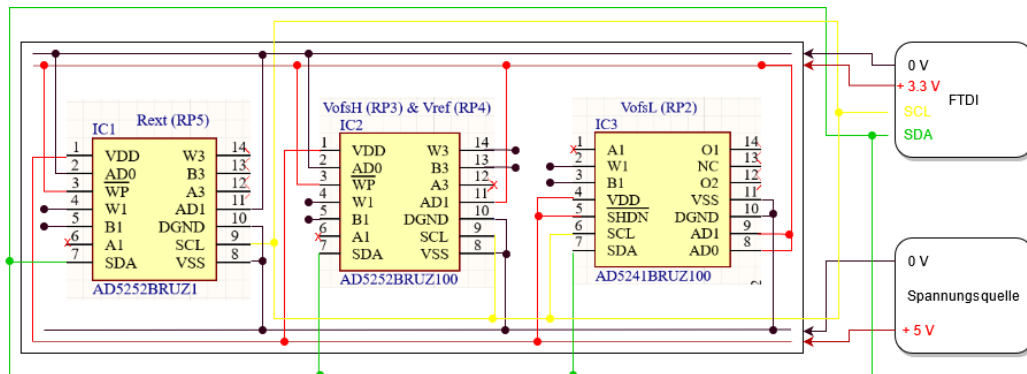


Abbildung 43: Verschaltungskonzept der digitalen Potentiometer

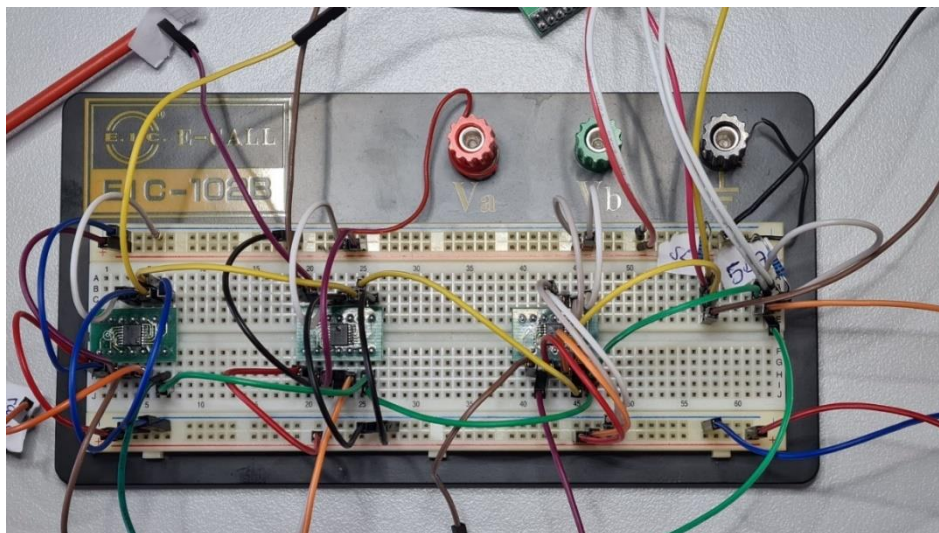


Abbildung 44: Versuchsaufbau zur Überprüfung der digitalen Potentiometer und des FTDI 2232H Mini Moduls

Anhand des Verschaltungskonzeptes aus Abbildung 43 wurde, wie in Abbildung 44 dargestellt, ein Versuchsaufbau durchgeführt. Hierzu wurden die digitalen Potis auf einen sogenannten „SOCKET ADAPTER TSSOP TO 14DIP“ aufgelötet, welche dann als Modul in das Steckbrett eingesteckt werden kann. Die Verschaltung wurde anhand der Abbildungen 41 und 42 sowie der Datenblätter der einzelnen digitalen Potentiometer konzipiert. [POT1][POT2][POT3]

Mit Hilfe des Versuchsaufbaus wurden die nachfolgenden Eigenschaften überprüft. Die Funktionalität der Potentiometer wurde mit einer Versorgungsspannung von 3.3 V und 5 V getestet. Das FTDI 2232 Mini Modul bezieht eine Versorgungsspannung von 3.3 V über die USB-Schnittstelle und kann zusätzlich die digitalen Potentiometer versorgen. Zusätzlich wurde die Funktionalität der "Write Protection" getestet. Der Schreibschutz ist im Datenblatt sehr umständlich beschrieben und musste deshalb zunächst überprüft werden. Nach erfolgreicher Überprüfung konnte festgehalten werden, dass der Schreibschutz (Write Protection) zuverlässig funktioniert und aktiv ist, sobald der Anschluss WP auf Masse gezogen wird.

Außerdem wurde beobachtet, dass die Spannungsversorgung der digitalen Potentiometer der Spannungsversorgung des FTDI 2232H Mini Moduls von 3.3 V entsprechen muss, da es ansonsten zu Taktungs-Problemen kommt [POT1]. Bei einer Spannungs-versorgung von 3.3 V kam es jedoch zu Kommunikationsproblemen zwischen den Potentiometern AD5252-BRUZ100 und AD5241-BRUZ100, so dass die Potis kein Acknowledge in einer Messung zurück gegeben haben, sprich diese konnten nicht gleichzeitig angesprochen werden. Dieses Problem wurde behoben, indem das Potentiometer AD5241-BRUZ100 durch ein AD5252-BRUZ100 ersetzt wurde, so dass alle digitalen Potentiometer aus derselben Baureihe stammen.

### 3.6 - Schaltpläne der digitalen Potentiometer

Die Abbildungen 45 bis 47 zeigen Ausschnitte der Schaltpläne für die digitalen Potentiometer auf der Testplatine. In braunrot sind die Namen der jeweiligen Anschlüsse dargestellt z.B. die Versorgungsspannung VCC oder das Massepotential GND. Die roten Kreuze bedeuten, dass die Anschlüsse nicht belegt sind. Tabelle 1 im Kapitel 3.3.5 zeigt eine Übersicht der Anschlüsse des Digitalpotentiometers AD5252.

Das analoge Potentiometer RP1 stellt den Widerstandswert für  $R_{ext}$  ein und wird durch das Potentiometer IC1 ersetzt, siehe Abbildung 45.

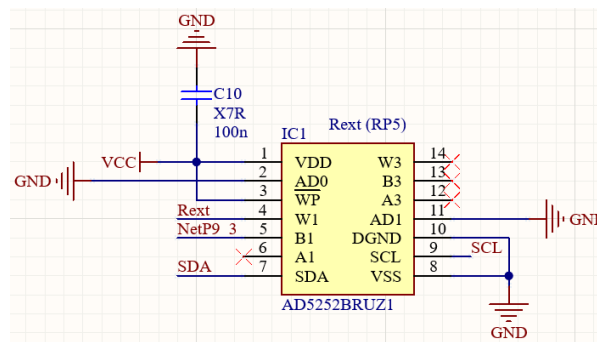


Abbildung 45: Digitalpoti IC1 (RP5, ersetztes, analoges Poti) für  $R_{ext}$

Das digitale Potentiometer IC2 in Abbildung 46 ersetzt RP3 und RP4 und stellt die Referenzspannung  $V_{ref}$  sowie die halbe Offsetspannung  $V_{ofsHalf}$  ein.

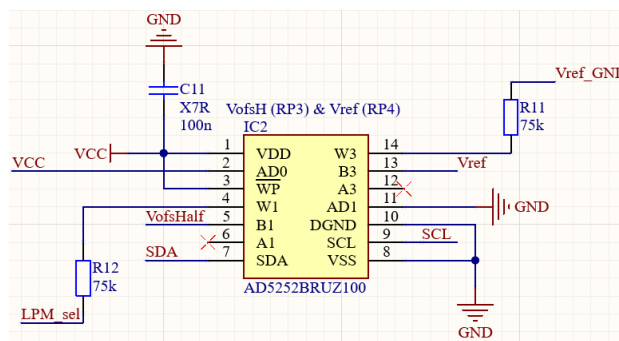


Abbildung 46: Digitalpoti IC2 (ersetzt die analogen Potentiometer RP3 und RP4) für  $V_{ofsHalf}$  und  $V_{ref}$

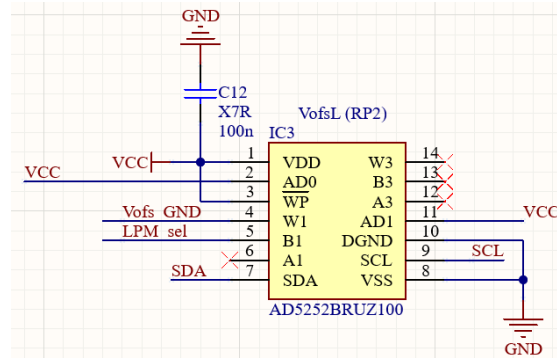


Abbildung 47: Digitalpoti IC3 (RP2, ersetztes, analoges Poti) für  $V_{ofsL}$

Das digitale Potentiometer IC3 ersetzt das analoge Potentiometer RP2 und setzt den Wert von  $V_{ofsL}$ .

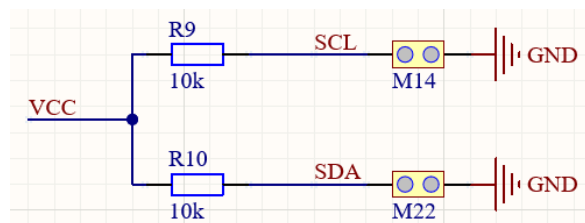


Abbildung 48: Schaltplan der Leitungen SDA, SCL

In Abbildung 48 sind die Leitungen SCL und SDA dargestellt, die jeweils mit einem Widerstand von 10 k $\Omega$  an die Versorgungsspannung VCC angeschlossen sind, um das Potenzial ggf. anzuheben. Über die Steckverbinder M14 und M22 werden die Leitungen mit dem Master verbunden.

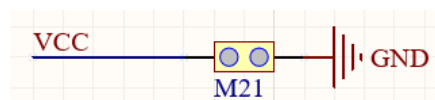


Abbildung 49: Schaltplan des Anschlusses M21 zum Messen der Versorgungsspannung der digitalen Potis

Am Anschluss M21 besteht die Möglichkeit, die Versorgungsspannung der digitalen Potentiometer zu messen oder ggf. eine externe Versorgungsspannung für die Potentiometer und den Instrumentenverstärker anzulegen.

### **3.7 - Modifikation der Leiterplatte (Rev 0.4)**

Aufgrund eines Layoutfehlers auf dem PCB mit der Bezeichnung Rev. 0.3 sind Kurzschlüsse an der Stelle aufgetreten, an der Lötstopplack auf der Rückseite der Platine entfernt wurde, um die Montage des Kühlkörpers zu ermöglichen. Sobald ein Kühlkörper auf der Rückseite angebracht wurde, stellte sich ein Kurzschluss zwischen der blau markierten Leiterbahn VCC und Masse ein. Dieser Fehler wurde in der aktuellen Version (Rev. 0.4) überarbeitet. Abbildung 50 veranschaulicht die fehlerhafte Positionierung (Abbildung 50 rechts) einer Leiterbahn sowie die vorgenommene Korrektur (Abbildung 50 links).

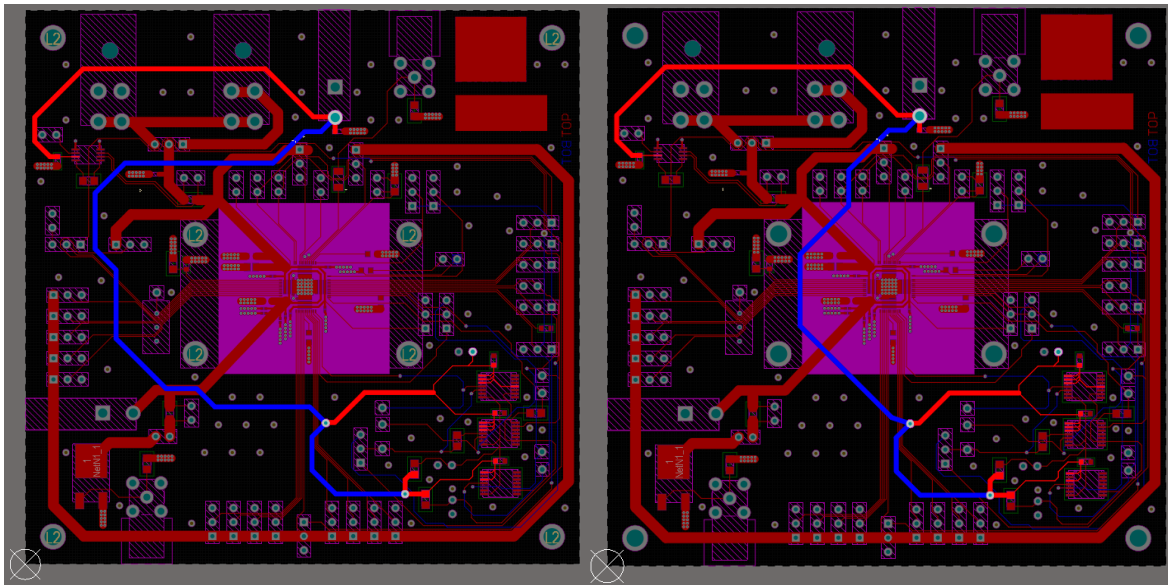


Abbildung 50: hervorgehobene Leitung VCC. Links: fehlerhafte Version Rev. 0.3.  
Rechts: Überarbeitete Version Rev. 0.4.

### **3.8 - Messaufbau**

Das gesamte Messsystem für die Validierung des Testchips ist in Abbildung 51 zu sehen. Der Messplatz besteht aus zwei Sourcemetern, einem Netzteil mit drei Ausgangskanälen, sieben Multimetern, einem Oszilloskop, einem Waveformgenerator sowie einem Linux PC für die Kommunikation mit einigen der Messgeräte.

Die Sourcemeter sowie ein Messgerät werden über GPIB mit dem Computer verbunden. Alle weiteren Multimeter kommunizieren über die USB-Schnittstelle mit dem PC. Über das Shuldo-Test-Tool werden die Messgeräte angesteuert und die Ergebnisse ausgelesen und gespeichert. Das in dieser Thesis vorgestellte neue Feature ermöglicht die Einstellung der Widerstandswerte über digitale Potentiometer für  $R_{ext}$ ,  $V_{ref}$ ,  $V_{ofsH}$  und  $V_{ofsL}$ .

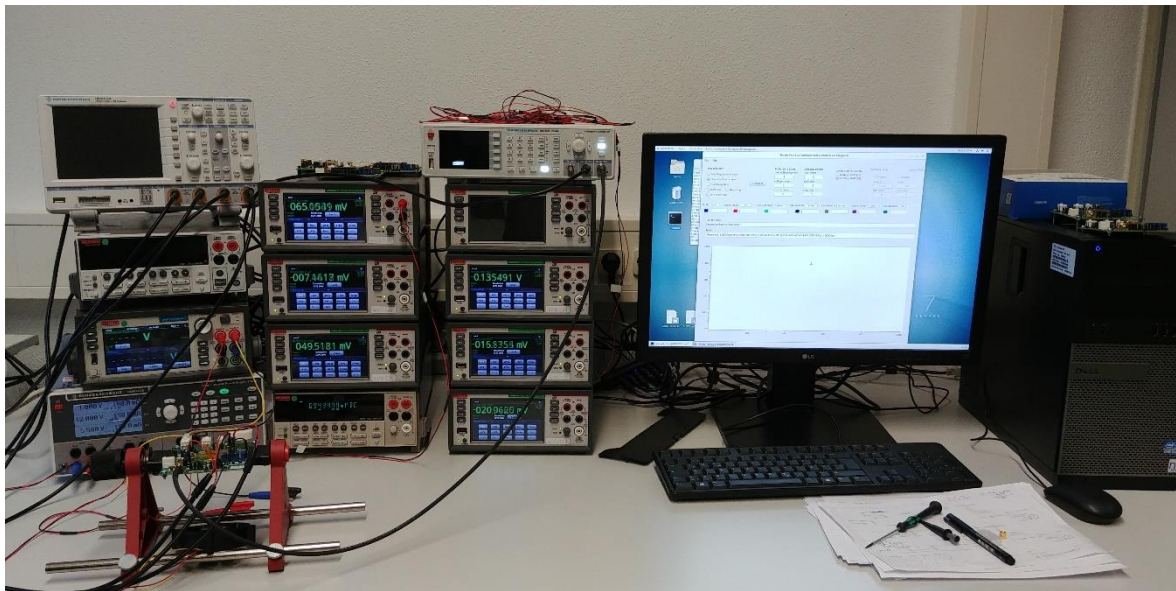


Abbildung 51: Messaufbau inkl. der Messgeräte und des PCs



Die Keithley Multimeter in Abbildung 51 sind in dem ShuLDO-Testtool bereits mit Beschriftungen für die jeweiligen Spannungen zugeordnet. Die Tabelle 4 zeigt eine Übersicht der Anordnung. Der Benutzer hat jedoch die Möglichkeit, die voreingestellten Zuweisungen anzupassen.

Tabelle 4: Anordnung der Multimeter

|  |  |
|--|--|
| Keithley Multimeter<br><br>Messgröße: <b>Iin_Sense</b>   |  |
| Keithley Multimeter<br><br>Messgröße: <b>Vout_Sense</b>  | Keithley Multimeter<br><br>Messgröße: <b>Vpre reg (Vout pre)</b> |
| Keithley Multimeter<br><br>Messgröße: <b>Vref</b>        | Keithley Multimeter<br><br>Messgröße: <b>Vref_pre</b>            |
| Keithley Multimeter<br><br>Messgröße: <b>Vofs (half)</b> | Keithley Multimeter<br><br>Messgröße: <b>Iout_Sense</b>          |

### **3.9 - Messgeräte**

Der nachfolgende Abschnitt gibt eine kurze Einführung in die verwendeten Messinstrumente.

#### 3.9.1 - Sourcemeter Keithley 2460 und Keithley 2401

Keithley 2460: Das Sourcemeter besitzt einen Spannungsbereich von bis zu 105 V und einen Strombereich von bis zu 7 A DC und eignet sich daher optimal zur Versorgung des Reglers. Die Quelle wird hierbei an den Anschluss PWR\_IN angeschlossen sowie im 4-Wire-Modus an den Sense-Anschluss. Hierdurch können der Strom und die Spannung über das Sourcemeter gemessen und gegebenenfalls limitiert werden. Das Sourcemeter wird über die SCPI- und TSP-Programmiermodi angesteuert und kann bei Bedarf mittels der Messsteuerungs-Software unterschiedliche Sweeps für Versorgungsspannung oder Versorgungsstrom ausführen. Zudem können über Trigger-Modelle Rampenfunktionen zwischen zwei Punkten, mit einer variablen Anstiegszeit von mindestens 50  $\mu$ s erstellt werden. [KEI3][ZOR]



Abbildung 52: Sourcemeter Keithley 2460 [KEI3]

Keithley 2401: Das Sourcemeter besitzt einen Spannungsbereich von bis zu 200 V und einen Strombereich von bis 1 A DC. Hierdurch ist das Sourcemeter nur bedingt als Versorgungsstromquelle für den Regler geeignet. Jedoch kann das Sourcemeter als konfigurierbare Last im Sink-Betrieb verwendet werden. Hierfür wird das Sourcemeter über den Anschluss V<sub>out</sub> mit der Platine verbunden. [KEI2][ZOR]



Abbildung 53: Sourcemeter Keithley 2401 [DLC]

### 3.9.2 - Labornetzgerät Rohde & Schwarz HMP4030

Das Labornetzgerät bietet als Konstant-Spannungsquelle eine Spannung von maximal 32 V mit drei Ausgangskanälen und einem Ausgangsstrom von bis zu 10 A pro Kanal. Es besteht zudem die Möglichkeit mit dem Labornetzgerät eine externe Referenz- und Offsetspannung auf den Testchip zu geben. Zudem wird das Labornetzgerät über den Platinen Anschluss VCC zur Versorgung der Prepower-Schaltung verwendet und zum Betrieb eines Lüfters, der den Chip während einer Messung kühlt. Die Kanäle des Netzteils Rohde & Schwarz HMP4030 sind galvanisch getrennt. Dies ist für die Messung einer seriellen Kette erforderlich. [RAS3][NEK][ZOR]



Abbildung 54: Netzteil Rohde & Schwarz HMP4030 [NEK]

### 3.9.3 - Oszilloskop Rohde & Schwarz HMO1024

Das Oszilloskop mit vier Ausgangskanälen wird für die Aufnahme transienter Signale verwendet. Die Kanäle besitzen einen Eingangswiderstand von 1 M $\Omega$ . Über den Trigger Ausgang besteht die Möglichkeit ein externes Trigger Signal anzuschließen. Dies ist hilfreich für Messungen, bei denen der Waveformgenerator zum Einsatz kommt.

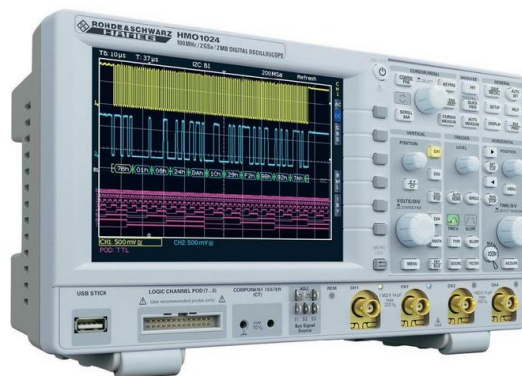


Abbildung 55: Oszilloskop Rohde & Schwarz HMO1024 [AZN]

Das Oszilloskop Rohde & Schwarz HMO1024 ist nicht in der Lage massefrei zu messen. Um Kurzschlüsse zwischen den einzelnen Reglern bei der Aufnahme von Spannungen für serielle Messungen zu vermeiden, kann die Mathematik-funktion des Oszilloskops verwendet werden. Anhand der Analysefunktion können unterschiedliche Berechnungsarten (z.B. Subtraktion, SUB) mit dem Formeleditor durchgeführt werden. Es besteht die Möglichkeit, die Spannungen einzelnen Regler in einer seriellen Kette über die Mathematikfunktion zu ermitteln. Die Tastköpfe werden bei den Messungen an die globale Masse angeschlossen. [AZN][RAS2][ZOR]



## 3.10 - Die Shuldo-Test-Messsteuerungs-Software

### 3.10.1 - Einleitung

Die Abbildung 59 zeigt die Bedienoberfläche der Software „Shuldo-Test“, die im Rahmen einer Bachelorarbeit [MAC] programmiert wurde. Für die Programmierung wird die Programmierumgebung Qt-Creator verwendet, welche auf der Programmiersprache C++ basiert und plattformunabhängig ist. Qt-Creator erleichtert die Entwicklungsarbeit für die GUI-Programmierung. Der Computer des Messplatzes läuft auf dem Betriebssystem CentOS, bei dem es sich um ein Linux-Derivat handelt und im Wesentlichen auf GNU-Software basiert.

Über die Messsteuerungs-Software lassen sich die verwendeten Sourcemeter ansteuern, die Multimeter auslesen und mit dem neuen Feature auch die Widerstände der Potentiometer einstellen. Der grafische Verlauf der statischen Messergebnisse wird im unteren Bereich des Programms dargestellt. Die Messergebnisse werden als Text-Datei gespeichert und können anschließend mit einem Tabellenkalkulationsprogramm weiterverarbeitet werden. [MAC][ZOR]



Abbildung 59: GUI der SLDO-Messsteuerungs-Software [MAC]

### Start der Software:

Für die Verwendung des Messtools benötigt der Benutzer Root-Rechte. Bevor das Tool gestartet werden kann, müssen für den GPIB-Treiber entsprechende Kernelmodule geladen werden. Hierzu müssen in der Kommandozeile (Terminal) folgende Befehle eingegeben werden:

→ `modprobe ni_usb_gpib`

→ `gpib_cong`

Im nächsten Schritt muss der Benutzer in das Verzeichnis wechseln, in dem sich die Messsoftware befindet. Über die nachfolgenden Befehle wird die Software gestartet:

→ `cd measure_2019`

→ `./measure`

### 3.10.2 - Starten einer Messung:

Nach dem Start des Programms, sollte zunächst die Checkbox „test multimeter“ im Bereich der „test selection“ ausgewählt werden, um anschließend durch Auswahl von "start test" zu überprüfen, ob eine Verbindung zwischen den Messgeräten und dem Computer besteht. Des Weiteren werden in diesem Schritt einmalig alle Widerstandswerte der digitalen Potentiometer gesetzt. Sobald alle Geräte miteinander synchronisiert sind, werden die aktuellen Messwerte in den entsprechenden Textfeldern ausgegeben. Jedem Messgerät wurde in der Benutzeroberfläche eine unterschiedliche Farbe zugeordnet, so besitzt z.B. das Messgerät „ID=16“ die Farbe Blau (siehe Abbildung 59).

Über die Schaltfläche „start test“ können die zur Verfügung stehenden Testdurchläufe ausgeführt werden. Die verschiedenen Felder im Fensterbereich „test selection“ stellen die ausführbaren Tests dar:

→ *Line Regulation voltage: Messung der Line Regulation im LDO-Modus*

→ *Line Regulation current: Messung der Line Regulation im ShuntLDO-Modus*

→ *Load Regulation: Messung der Load Regulation im LDO- und ShuntLDO-Modus*

→ *Line Regulation current, sweep  $R_{ext}$ : Automatisierte Messung der Line Regulation im ShuntLDO-Modus, bei dem der Widerstand  $R_{ext}$  automatisiert erhöht wird (Kapitel 3.4.3)*

→ *Line Regulation current, sweep  $V_{ofsH}$ : Automatisierte Messung der Line Regulation im ShuntLDO-Modus, bei dem der Widerstand  $V_{ofsH}$  automatisiert erhöht wird (Kapitel 3.4.3)*

Über die Auswahl „Which SMU for Sweep?“ kann zwischen beiden Sourcemetern aus dem Messaufbau gewählt werden. Für die Messung einer Line Regulation wird das Keithley 2460 ausgewählt und für eine Load Regulation das Keithley 2400. Der Start- bzw. Endpunkt (start value, end value) und die Schrittweite (step size) der Messung lassen sich im Bereich „settings sweep“ eintragen. Im Auswahlbereich der digitalen Potentiometer kann ebenfalls ein Sweep ausgeführt werden. Hierbei lassen sich für die Potis  $R_{ext}$  (sweep Rslope) und  $V_{ofsH}$  (sweep Offsetspannung) ein ohmscher Startwert, Endwert und die

Schrittweite des Sweeps einstellen. Im Bereich „limits und ranges“ werden die Grenzwerte der Ausgabewerte sowie der Messbereiche über die Textfelder eingestellt. Bei einem Wechsel zwischen den Tests ist es ratsam, die Einstellungen manuell am Sourcemeter zu überprüfen.

Für die Potis  $V_{ref}$  und  $V_{ofsL}$  lässt sich im Bereich der digitalen Potentiometer jeweils ein ohmscher Wert auswählen. Für eine Einzelübergabe der Ohmwerte steht zudem ein separater Button zur Verfügung, bei dem kein Sweep ausgeführt wird.

Zusätzlich bietet das Tool die Möglichkeit, eine Rampenfunktion für die Sourcemeter zu konfigurieren. Über die Auswahl von „start ramp“ lässt sich nach Festlegung des Start- und Endpunkts, der Anzahl an Schritten (steps) und der Zeit zwischen den Schritten (delay) ein Trigger Model aus verschiedenen Blöcken auf dem Sourcemeter ausführen, welches so lange läuft, bis die Checkbox „stop ramp“ ausgewählt und ein weiterer Klick auf „start test“ ausgeführt wird. Bei Bedarf lässt sich das Trigger-Modell auf dem Sourcemeter im Menü unter Trigger → Configure öffnen und manuell bearbeiten.

Zur Speicherung der Messergebnisse wird ein Pfad samt Dateiname benötigt, der im Textfeld „Log File Path“ eingetragen wird. Der Name der Textdatei sollte standardmäßig mit der Endung „.txt“ enden. Messungen werden in dieselbe Ausgabedatei gespeichert, wenn zwischen den Messungen der Dateiname bzw. Dateipfad nicht verändert wird. Existiert der gewählte Pfad nicht, wird keine Datei erstellt. Zudem können weitere Informationen zur Messung in das Textfeld „Note“ eingetragen werden, wie z.B. die gesetzten Trimm-Bits, die verwendeten Referenzspannungen oder weitere gewünschte Informationen. [ZOR]

### 3.11. Erweiterung der Software zur Ansteuerung digitaler Potentiometer

#### 3.11.1 - Übersicht

Die Shuldo-Test-Messsteuerungs-Software wurde mit dem Programm „Qt-Creator“ erstellt und führt nach der Erweiterung um die Potentiometer-Steuerung, die Bezeichnung Shuldo-Test 1.1b. Abbildung 60 zeigt die Ansicht des Programms Qt-Creator.

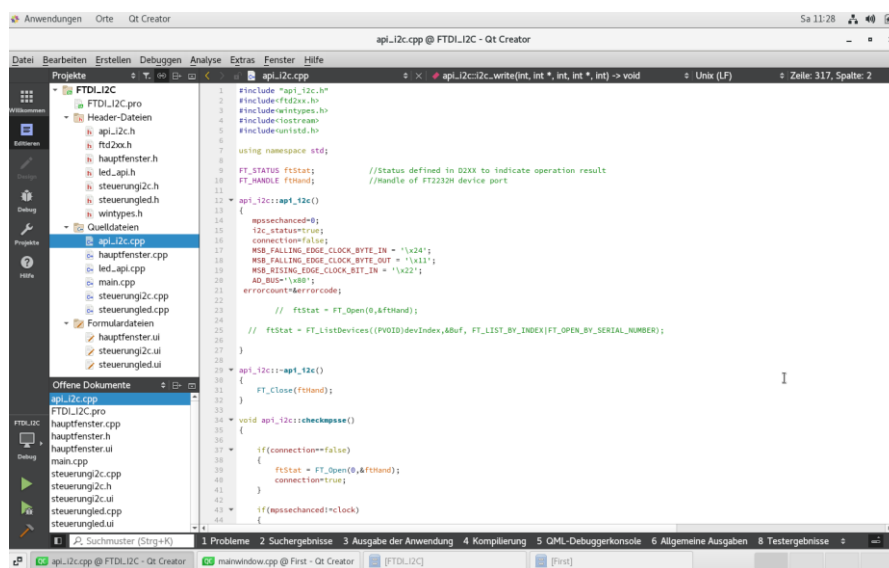


Abbildung 60: Ansicht des Programms Qt-Creator

Ein wesentlicher Bestandteil der Programmierung bzw. Weiterentwicklung der Shuldo-Test-Messsteuerungs-Software, ist die Einbindung der API aus der Arbeit [ÖZK2]. Die API beinhaltet die von FTDI zur Verfügung gestellten Methoden, mit deren Hilfe eine I<sup>2</sup>C-Kommunikation zwischen dem FTDI Mini Modul (Master) und den Potis (Slave) durchgeführt werden kann.



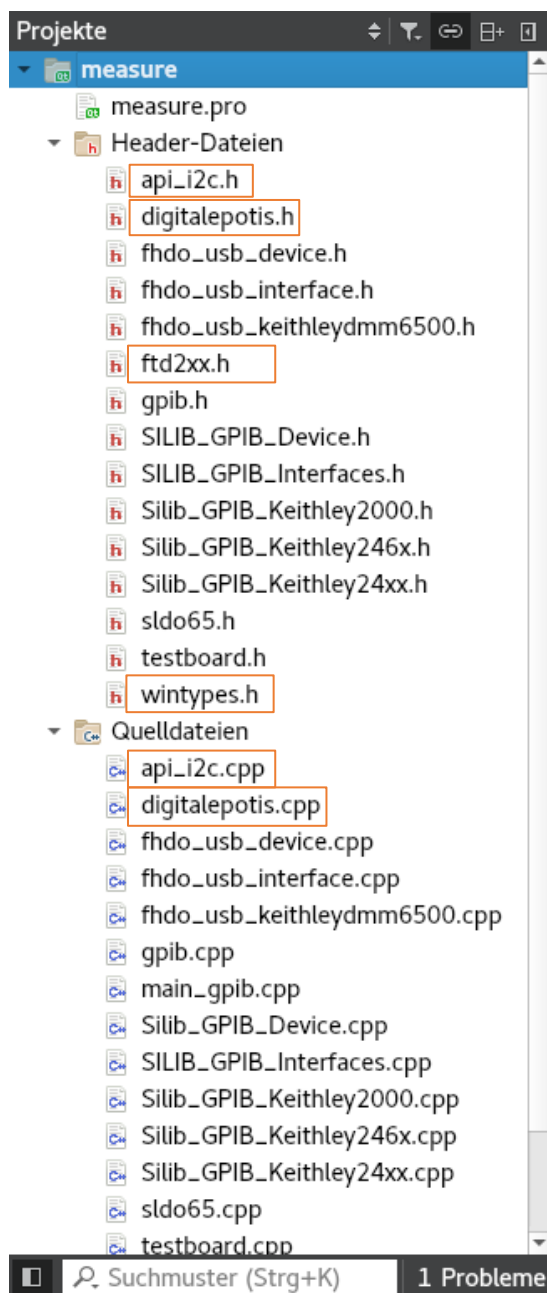


Abbildung 61: Übersicht der eingebundenen Header- und Quell-Dateien

Zum Aufruf der API und den einhergehenden Funktionen müssen die Header-Dateien `api_i2c.h`, `wintypes.h` und `ftd2xx.h` eingebunden werden. Die Header-Dateien `wintypes.h` und `ftd2xx.h` beinhalten den Treiber des FTDI 2232H Mini Moduls. Ebenfalls muss die Quelldatei `api_i2c.cpp` eingefügt werden, welche die Methoden zur Steuerung der Kommunikation beinhaltet. Die Abbildung 61 zeigt die eingefügten Header- und Quell-Dateien.

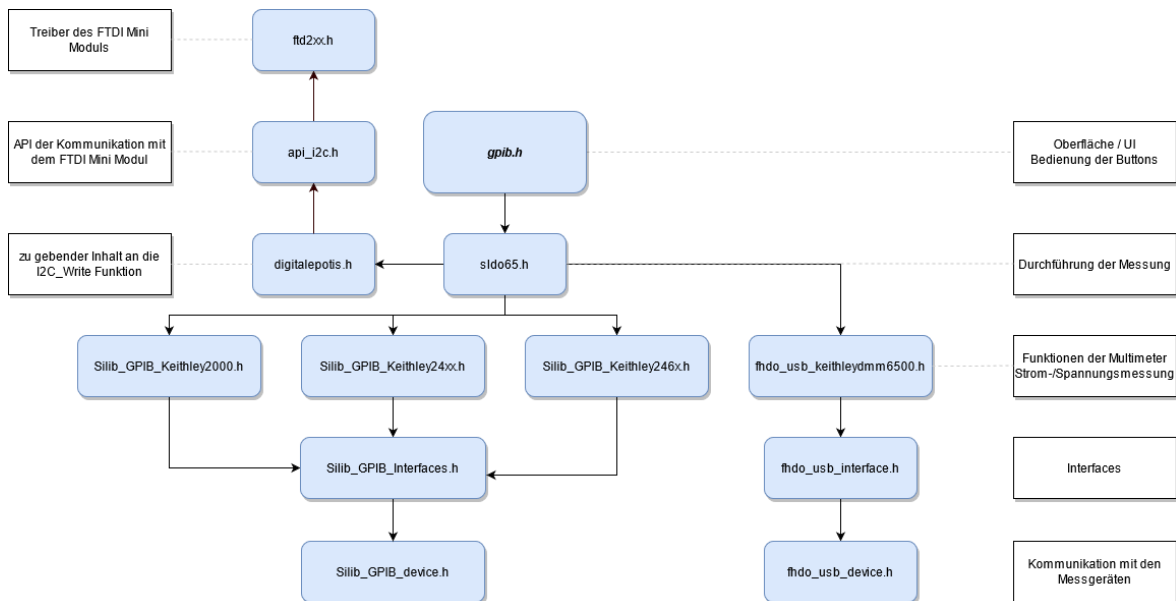


Abbildung 62: Klassendiagramm

Abbildung 62 zeigt das gesamte Klassendiagramm der Testsystem-Software. Der Treiber des FTDI Mini Moduls, welcher sich in der Klasse `ftd2xx` befindet, wird in die Klasse `api_i2c` eingebunden, der die Kommunikation des I<sup>2</sup>C's beinhaltet. Die API wiederum wird in die Klasse der digitalen Potentiometer (`digitalepotis`) eingearbeitet. Die Methoden aus der Poti-Klasse werden anschließend an die Hauptklasse `slido65` vererbt, in der alle Klassen zusammengeführt und aufgerufen werden. Durch die Vererbung der Poti-Methoden kann nachfolgend jede einzelne Funktion zur Übergabe der Informationen des Potis mittels gewünschter Aktion direkt in der Klasse `gpib` aufgerufen werden.

### 3.11.2 - Quellcode

Nachfolgend werden die Hauptbestandteile des erweiterten Quellcodes erläutert. Hierzu werden die einzeln erstellten Funktionen beschrieben. Doppelt ausgeführte Methoden werden nicht abgebildet und können im kompletten Programmcode, im Anhang nachgeschlagen werden.

In Abbildung 63 werden die Dropdown-Menüs („comboBoxen“, siehe Abbildung 59) mit den ohmschen Werten gefüllt. Durch den Aufruf `z->poti_widerstand_1k` bzw. `100k` (z.B. Code-Zeile 56) werden die Methoden aus der Klasse „`digitalepotis`“ ausgeführt und in die jeweilige Variable gespeichert und anschließend in die Boxen geladen. Des Weiteren werden die Standartwerte der Potis als Startwert gesetzt, so dass nach dem Start der Software reguläre Werte vorkonfiguriert sind (Code-Zeile 88).

```

49 // Banko : DROPDOWN MENÜS DER BUTTONS, ZUM ÜBERGEBEN DER OHMSCHENWERTE .....
50
51 int ohm_Rext, ohm_Rext_End, ohm_Vofsh, ohm_Vofsh_End, ohm_Vofsl, ohm_Vref;
52
53 for (int Schrittweite = 0; Schrittweite < 256; Schrittweite++) // Schleife bis 255 zum Erstellen der Widerstandswerte
54 {
55     ohm_Rext = z->poti_widerstand_1k(Schrittweite); // Aufrufen der Funktion poti_widerstand aus der Klasse digitalepotis
56
57     ohm_Rext_End = z->poti_widerstand_1k(Schrittweite);
58
59     ohm_Vofsh = z->poti_widerstand_100k(Schrittweite);
60
61     ohm_Vofsh_End = z->poti_widerstand_100k(Schrittweite);
62
63     ohm_Vofsl = z->poti_widerstand_100k(Schrittweite);
64
65     ohm_Vref = z->poti_widerstand_100k(Schrittweite);
66
67
68
69 ui->comboBox_Rext->addItem(QString::number(ohm_Rext)); // Laden des Wertes in die ComboBox
70 ui->comboBox_Rext->setCurrentIndex(132); // Startwert setzen
71
72 ui->comboBox_Rext_end->addItem(QString::number(ohm_Rext_End));
73 ui->comboBox_Rext_end->setCurrentIndex(144);
74
75
76 ui->comboBox_Vofsh->addItem(QString::number(ohm_Vofsh + 75000));
77 ui->comboBox_Vofsh->setCurrentIndex(76);
78
79 ui->comboBox_Vofsh_end->addItem(QString::number(ohm_Vofsh_End + 75000));
80 ui->comboBox_Vofsh_end->setCurrentIndex(77);
81
82
83 ui->comboBox_Vofsl->addItem(QString::number(ohm_Vofsl));
84 ui->comboBox_Vofsl->setCurrentIndex(18);
85
86
87 ui->comboBox_Vref->addItem(QString::number(ohm_Vref + 75000));
88 ui->comboBox_Vref->setCurrentIndex(163);
89
90 }

```

Abbildung 63: Erstellen der Dropdown-Menüs

```

21
22 int digitalepotis::poti_widerstand_1k(int Schrittweite) // Methode für die Formel zum Erstellen der Widerstandswerte
23 {
24
25     int OhmWert;
26     int PotiGrundwert = 75;
27
28     OhmWert = ((Schrittweite * 4) + PotiGrundwert);
29
30     //OhmWert = (((Schrittweite * 1000) / 256) + PotiGrundwert);
31
32     return OhmWert;
33
34 }
35
36
37
38 int digitalepotis::poti_widerstand_100k(int Schrittweite) // Methode für die Formel zum Erstellen der Widerstandswerte
39 {
40
41     int OhmWert;
42     int PotiGrundwert = 75;
43
44     OhmWert = ((Schrittweite * 390) + PotiGrundwert);
45
46     //OhmWert = (((Schrittweite * 100000) / 256) + PotiGrundwert);
47
48     return OhmWert;
49
50 }
51

```

Abbildung 64: Formel zur Berechnung der Widerstandswerte aus der Klasse „digitalepotis“

In Abbildung 64 sind die Methoden zur Generierung der Widerstandswerte gelistet. Da das 1 k $\Omega$  Poti eine Schrittweite von 3.9  $\Omega$  pro Schritt entspricht und auf Grund der Bauteil-Toleranz ein derart genauer Wert nicht erreicht wird, wurde eine vereinfachte Formel angewendet, bei der 4  $\Omega$  pro Schritt festgelegt wurde. Das 100 k $\Omega$  Poti besitzt eine Schrittweite von 390  $\Omega$ .

```
1395 // Banko : AB HIER INDEX WECHSEL .....
1396
1397
1398
1399 void gpib::on_comboBox_Rext_currentIndexChanged(const QString &arg1)
1400 {
1401     int index_Rext, schrittweiten_subtraktion_Rext, schrittweite_ende; // Variable für die Schrittweite
1402     char hexa_Rext[3]; // Variable für das umwandeln in HEX
1403
1404
1405     index_Rext = ui->comboBox_Rext->currentIndex(); // in die Variable a wird der aktuelle Index geladen : Index = Schrittweite
1406     ui->label_Rext_schritte->setNum(index_Rext); // in das Label wird die ausgewählte Schrittweite als Zahl gesetzt
1407     sprintf(hexa_Rext, "%x", index_Rext); // umwandeln der Schrittweite in HEX
1408
1409     ui->hex_Rext->setText(hexa_Rext); // HEX Wert der Variable ins Label setzen
1410
1411
1412     schrittweite_ende = ui->comboBox_Rext_end->currentIndex(); // Schrittweite Endwert laden
1413     schrittweiten_subtraktion_Rext = schrittweite_ende - index_Rext; // Subtraktion : Endwert - Startwert
1414
1415
1416     if (ui->checkBox_sweep_Rext->isChecked()) // wenn Sweep Rext ausgewählt ist, ...
1417     {
1418         if (schrittweiten_subtraktion_Rext <= 0) // Schrittweiten Subtraktion kleiner gleich null ?
1419         {
1420             ui->comboBox_Rext->setCurrentIndex(schrittweite_ende-1); // setze den Indexwert auf Endwert -1, damit kein negativer Wert möglich ist
1421         }
1422     }
1423     else // wenn nicht, ...
1424     {
1425         ui->spinBox_Rext->setMaximum((schrittweiten_subtraktion_Rext*4)); // setze das Maximum der SpinBox auf den Wert der Subtraktion
1426     }
1427
1428     ui->spinBox_Rext->setSingleStep(4); // 4 Ohm pro Schrittweite
1429     ui->spinBox_Rext->setMinimum(4); // Minimum 4 Ohm
1430 }
1431
1432
1433
1434
1435
1436
1437
1438
1439 }
```

Abbildung 65: Index Wechsel der „ComboBox“ Rext

Durch den Code aus Abbildung 65 wird verhindert, dass bei der Konfiguration eines Sweeps in der GUI der ohmsche Startwert (start value) größer als der ohmsche Endwert (end value) gesetzt wird. Hierdurch kann die Schrittweite (step size) keinen negativen Wert erreichen, wodurch ein fehlerhafter Sweep verhindert wird. Der Startwert bei einem Sweep besitzt jederzeit die Stop-Schrittweite minus einem Schritt, siehe Code-Zeile 1425. Des Weiteren wird die Schrittgröße auf 4  $\Omega$  gesetzt (Code-Zeile 1435). In der Zeile 1437 wird der Startwert des Menüs auf 4  $\Omega$  festgelegt.

In gleicher Weise wird die Stop-Schrittweite behandelt, so dass die Stop-Schrittweite nicht kleiner als die Start-Schrittweite gesetzt wird, um eine negative Schrittweite zu verhindern.

Der in Abbildung 66 gezeigte Quellcode ist verantwortlich dafür, dass während der Ausführung der Funktion „Test-Multimeter“ bzw. während der Initialisierung der Multimeter, alle Potentiometer einmalig, gleichzeitig gesetzt werden. Hierdurch wird gewährleistet, dass die Potentiometer mit sinnvollen Werten initialisiert wurden.

```

163 // Banko : einmaliges Übergeben der Werte, zum Start bzw. beim initialisieren der Multimeter
164 int ohmwert_Rext_inHex, ohmwert_VofSH_inHex, ohmwert_Vref_inHex, ohmwert_VofSL_inHex; // integer Variablen für die Slave Adresse und den Ohmwert in HEX
165 OString ohmwert_Rext, ohmwert_VofSH, ohmwert_Vref, ohmwert_VofSL; // String für das Umwandeln des ohmschen Wertes
166
167 ohmwert_Rext = ui->hex_Rext->text(); // in den String den Text des Feldes hex_Rext laden
168 ohmwert_Rext_inHex = ohmwert_Rext.toInt(60k,16); // umwandeln in INT
169
170 ohmwert_VofSH = ui->hex_VofSH->text(); // in den String den Text des Feldes hex_VofSH laden
171 ohmwert_VofSH_inHex = ohmwert_VofSH.toInt(60k,16); // umwandeln in INT
172
173 ohmwert_Vref = ui->hex_Vref->text(); // in den String den Text des Feldes hex_Vref laden
174 ohmwert_Vref_inHex = ohmwert_Vref.toInt(60k,16); // umwandeln in INT
175
176 ohmwert_VofSL = ui->hex_VofSL->text(); // in den String den Text des Feldes hex_Rext laden
177 ohmwert_VofSL_inHex = ohmwert_VofSL.toInt(60k,16); // umwandeln in INT
178
179 // std::cout<<std::hex<<ohmwert_Rext_inHex<<"TEST"<<std::endl; // hierdurch könnte man den Zwischenschritt überprüfen, AÜsgabe auf der Konsole
180
181 z->widerstandswert_poti_uebergabe_Rext(ohmwert_Rext_inHex); // Übergeben der HEX Werte
182 z->widerstandswert_poti_uebergabe_VofSH_Vref(ohmwert_VofSH_inHex, ohmwert_Vref_inHex);
183 z->widerstandswert_poti_uebergabe_VofSL(ohmwert_VofSL_inHex);
184 }
    
```

Abbildung 66: setzen aller Poti-Werte gleichzeitig, in der Test-Multimeter Auswahl

Abbildung 67 zeigt die Funktion zur Konfiguration des ohmschen Wertes der digitalen Potentiometer über das FTDI-Modul. Die Slave-Adresse der jeweiligen Potis ist fest durch das Layout festgesetzt (z.B. Code-Zeile 59). Für die Übergabe wurde ein Array mit drei Feldern erstellt, in denen die festen Werte 00 und 01 gesetzt werden, da diese vom Datenblatt so vorgegeben werden (Code-Zeilen 64 und 65). Als drittes Feld wird der gewünschte Widerstandswert übergeben (Code-Zeile 66). Für den Poti  $V_{ofSH}$ ,  $V_{ref}$  werden fünf Array-Felder benötigt, da hier zwei Widerstandswerte übergeben werden. Diese Daten werden ebenfalls in der Konsole über „cout“ ausgegeben, so dass diese während eines Schreibvorgangs überprüft werden können (Code-Zeilen 68 und 69). Über die Funktion `i2c_write` wird das gesamte Array-Paket übergeben (Code-Zeile 71). [POT1]

```

54 void digitalepotis::widerstandswert_poti_uebergabe_Rext(int ohmwert_Rext_inHex) // Methode zum aufrufen der Schreibfunktion des Potis
55 {
56
57     int slaveaddress_Rext;
58
59     slaveaddress_Rext = 0X58;
60
61
62     int outputdata_Rext[3]; // Array mit 3 Feldern
63
64     outputdata_Rext[0] = 0X00; // 00 ins erste Feld
65     outputdata_Rext[1] = 0X01; // 01 ins zweite Feld
66     outputdata_Rext[2] = ohmwert_Rext_inHex; // Laden des HEX Wertes in das dritte Feld
67
68     cout<<"\n<<" Rext Poti " <<"SlaveAdresse:"<<slaveaddress_Rext<<" / Festwert:"<<outputdata_Rext[0]<<outputdata_Rext[1]<<endl; // Ausgabe der Werte über die Konsole;
69     std::cout<<std::hex<<"HEX: " <<outputdata_Rext[2]<<std::endl;
70
71     i2c.i2c_write(slaveaddress_Rext, outputdata_Rext, 3); // Funktion i2c_write aus der API ausführen
72
73 }
74
75
76 void digitalepotis::widerstandswert_poti_uebergabe_VofSH_Vref(int ohmwert_VofSH_inHex, int ohmwert_Vref_inHex)
77 {
78
79     int slaveaddress_VofSH_Vref;
80
81     slaveaddress_VofSH_Vref = 0X5A;
82
83
84     int outputdata_VofSH_Vref[5];
85
86     outputdata_VofSH_Vref[0] = 0X00;
87     outputdata_VofSH_Vref[1] = 0X01;
88     outputdata_VofSH_Vref[2] = ohmwert_VofSH_inHex;
89     outputdata_VofSH_Vref[3] = 0X00;
90     outputdata_VofSH_Vref[4] = ohmwert_Vref_inHex;
91
92     cout<<"\n<<" VofSH & Vref Poti " <<"SlaveAdresse:"<<slaveaddress_VofSH_Vref<<" / Festwert:"<<outputdata_VofSH_Vref[0]<<outputdata_VofSH_Vref[1]<<endl;
93     std::cout<<std::hex<<"HEX: " <<outputdata_VofSH_Vref[2]<<std::endl;
94     cout<<" - Festwert:"<<outputdata_VofSH_Vref[3]<<endl;
95     std::cout<<std::hex<<"HEX: " <<outputdata_VofSH_Vref[4]<<std::endl;
96
97     i2c.i2c_write(slaveaddress_VofSH_Vref, outputdata_VofSH_Vref, 5);
98
99 }
100
    
```

Abbildung 67: Übergabe Funktion der digitalen Potentiometer

```

243 // Banko : ELSE IF ABFRAGE DER SWEEP FUNKTIONEN/CHECKBOXEN
244 else if(ui->checkBox_sweep_Rext ->isChecked())
245 {
246     QString s_sweep_Rext, e_sweep_Rext, st_sweep_Rext;
247     int s_sweep_Rext_inHex, e_sweep_Rext_inHex, st_sweep_Rext_inHex;
248
249     s_sweep_Rext = ui->hex_Rext->text(); // Startwert Sweep Rext laden
250     e_sweep_Rext = ui->hex_Rext_end->text(); // Endwert Sweep Rext laden
251     st_sweep_Rext = ui->spinBox_Rext->text(); // Schrittweite Sweep Rext laden
252
253     s_sweep_Rext_inHex = s_sweep_Rext.toInt(&ok,16); // umwandeln in HEX
254     e_sweep_Rext_inHex = e_sweep_Rext.toInt(&ok,16);
255     st_sweep_Rext_inHex = st_sweep_Rext.toInt(&ok,16);
256
257     ui->plot->setAxisTitle(QwtPlot::xBottom, "I(A)"); //Beschriftung der X-Achse mit I für Strom
258     ui->plot->setAxisTitle(QwtPlot::yLeft, "U(V)"); //Beschriftung der Y-Achse mit U für Spannung
259
260     while ( s_sweep_Rext_inHex <= e_sweep_Rext_inHex ) // Schleife : Startwert kleiner Endwert ?
261     {
262         if (ui->cb_k2400->isChecked()) // Auswahl zwischen 2400 und 2460
263         {
264             cout<<"\n"<<endl;
265             z->widerstandswert_poti_uebergabe_Rext(s_sweep_Rext_inHex);
266             a=z->lineregcur(s, e, st, limit); //K2400 //per Memberfunktion lineregcur von z(code in sldo65.cpp) soll der Vector b für die /
267             // cout<<"lineregcur aus der Klasse sldo60 ausführen"<<endl; //K2400 //per Memberfunktion lineregcur von z(code in sldo65.cpp) soll der Vector b für die /
268         }
269         else
270         {
271             cout<<"\n"<<endl;
272             z->widerstandswert_poti_uebergabe_Rext(s_sweep_Rext_inHex);
273             a=z->lineregcur2460(s, e, st, limit); //K2460 //per Memberfunktion lineregcur von z(code in sldo65.cpp) soll der Vector b für die A
274             // cout<<"lineregcur2460 aus der Klasse sldo60 ausführen"<<endl; //K2460 //per Memberfunktion lineregcur von z(code in sldo65.cpp) soll der Vector b für die A
275         }
276
277         // std::cout<<std::hex<<s_sweep_Rext_inHex<<"start"<<std::endl;
278         // std::cout<<std::hex<<e_sweep_Rext_inHex<<"ende"<<std::endl;
279         // std::cout<<std::hex<<st_sweep_Rext_inHex/4<<"step"<<std::endl;
280
281         s_sweep_Rext_inHex = s_sweep_Rext_inHex + (st_sweep_Rext_inHex/4); // Erhöhe um Schrittweite : Schrittweite / 4
282     }
283 }
284 }
285 }

```

Abbildung 68: Auswahl der CheckBox „sweep Rext“ ; automatisierter Vorgang

Die Kernfunktionen der ergänzten Aufgabe der Test-Software-Erweiterung ist die Auswahl zwischen den Sweeps für Rslope und der Offsetspannung und ist in Abbildung 68 abgebildet. Zuerst wird über eine if-Abfrage überprüft, welche Messung ausgewählt ist (Code-Zeile 243). Anschließend werden die aktuellen Widerstandswerte (entsprechend der Schrittweite, in HEX) geladen und die Beschriftung der Achsen in der GUI gesetzt (Code-Zeile 249 bis 259). Nachfolgend wird eine while-Schleife durchlaufen, bei der untersucht wird, ob der gesetzte „sweep-Widerstands-Startwert“ kleiner-gleich dem „sweep-Widerstands-Endwert“ ist (Code-Zeile 261 bis 278). Solange diese Bedingung erfüllt ist, wird eine if-Abfrage durchlaufen, in der festgestellt wird, welche SMU ausgewählt worden ist, um die entsprechende Methode auszuführen. Für die SMU 2400 wird die Methode „lineregcur“ und für die SMU 2460 wird die Methode „lineregcur2460“ ausgeführt. Zuvor wird noch der Widerstandswert übergeben. Die Methoden lineregcur und lineregcur2460 sind bereits vorhandene Methoden, die bei einer regulären „LineRegulation Current“ genutzt werden. Nach jedem Durchlauf der while-Schleife, wird der Startwert um die ausgewählte Schrittweite erhöht (Code-Zeile 285).

Für die Erstellung der Textdatei bei einem Sweep ist es wichtig, dass die Messwerte nach jeder Messung in dem String gelöscht werden. In einer regulären Messung werden die einzelnen Messungen hintereinander in einer Datei aufgelistet. Wird der String-Inhalt nicht gelöscht, wird jede Messung vor jeder weiteren Messung aufgelistet, so dass beispielsweise die sechste Messung auch die vorherigen fünf Messungen beinhaltet. Das Löschen des String-Inhalts wird durch den Befehl „g.clear“ ausgelöst und ist in der Abbildung 69, Zeile 665 zu finden.

```

641         if (ui->cb_loadreg->isChecked())
642         {
643         }
644         out << "I(A)\t" + names << endl;           //Stiller : Ausgabe Loadreg
645     }
646     else if(ui-> cb_linereg_curr ->isChecked())
647     {
648     out << "I(A)\t" + names << endl;           //Stiller : Ausgabe LineReg Current
649     }
650     else if(ui-> cb_linereg_vol ->isChecked())
651     {
652     out << "U(V)\t" + names << endl;           //Stiller : Ausgabe Line Reg Voltage
653     }
654     else if(ui-> checkBox_sweep_Rext ->isChecked())
655     {
656     out << "I(A)\t" + names << endl;           //Banko : Ausgabe LineReg Current und Sweep
657     }
658     else if(ui-> checkBox_sweep_VofsH ->isChecked())
659     {
660     out << "I(A)\t" + names << endl;           //Banko : Ausgabe LineReg Current und Sweep
661     }
662
663
664     out << g << endl; }           // Die Messwerte werden in der Datei hinterlegt
665     g.clear();                   // String leeren, ansonsten wird die Ausgabe der Daten doppelt angelegt
666     data.close();               // Die Datei wird geschlossen
667     check = 0;
668
669     }
670
671 }
672
673

```

Abbildung 69: String leeren, um doppeltes Anlegen der Dateien zu vermeiden

```

1353 // Banko : AB HIER CLICK DER BUTTONS ZUM ÜBERGEBEN DER SCHRITTWEITE EINZELN .....
1354
1355 void gpib::on_pushButton_Rext_clicked()           // bei Betätigung des Buttons wird die nachfolgende Funktion ausgeführt
1356 {
1357     int ohmwert_Rext_inHex;                       // integer Variablen für die Slave Adresse und den OhmWert in HEX
1358     QString ohmwert_Rext;                         // String für das umwandeln des ohmschen Wertes
1359
1360     ohmwert_Rext = ui->hex_Rext->text();           // in den String den Text des Feldes hex_Rext laden
1361     ohmwert_Rext_inHex = ohmwert_Rext.toInt(&ok,16); // umwandeln in INT
1362     std::cout<<std::hex<<"HEX: " <<ohmwert_Rext_inHex<<std::endl;
1363
1364     z->widerstandswert_poti_uebergabe_Rext(ohmwert_Rext_inHex); // Übergabe des Wertes
1365 }
1366
1367
1368 void gpib::on_pushButton_VofsH_Vref_clicked()    // bei Betätigung des Buttons wird die nachfolgende Funktion ausgeführt
1369 {
1370     int ohmwert_VofsH_inHex, ohmwert_Vref_inHex; // integer Variablen für die Slave Adresse und den OhmWert in HEX
1371     QString ohmwert_VofsH, ohmwert_Vref;         // String für das umwandeln des ohmschen Wertes
1372
1373     ohmwert_VofsH = ui->hex_VofsH->text();        // in den String den Text des Feldes hex_VofsH laden
1374     ohmwert_VofsH_inHex = ohmwert_VofsH.toInt(&ok,16); // umwandeln in INT
1375
1376     ohmwert_Vref = ui->hex_Vref->text();          // in den String den Text des Feldes hex_Vref laden
1377     ohmwert_Vref_inHex = ohmwert_Vref.toInt(&ok,16); // umwandeln in INT
1378
1379     z->widerstandswert_poti_uebergabe_VofsH_Vref(ohmwert_VofsH_inHex, ohmwert_Vref_inHex); // Übergabe der Werte
1380 }
1381
1382
1383 void gpib::on_pushButton_VofsL_clicked()         // bei Betätigung des Buttons wird die nachfolgende Funktion ausgeführt
1384 {
1385     int ohmwert_VofsL_inHex;                       // integer Variablen für die Slave Adresse und den OhmWert in HEX
1386     QString ohmwert_VofsL;                         // String für das umwandeln des ohmschen Wertes
1387
1388     ohmwert_VofsL = ui->hex_VofsL->text();          // in den String den Text des Feldes hex_Rext laden
1389     ohmwert_VofsL_inHex = ohmwert_VofsL.toInt(&ok,16); // umwandeln in INT
1390
1391     z->widerstandswert_poti_uebergabe_VofsL(ohmwert_VofsL_inHex); // Übergabe des Wertes
1392 }
1393

```

Abbildung 70: Übergabe der Einzelwerte

Für die Ausführung einer Messung ohne Sweep, z.B. bei einer Line Regulation, wird eine einfache Übergabe der ohmschen Werte der Potentiometer benötigt. Diese Funktion ist in Abbildung 70 dargestellt. Nach Auswahl des gewünschten Wertes wird dieser mittels des jeweiligen Buttons (siehe Abbildung 59) zum Setzen der Potis übergeben. Hierzu werden die Schrittweiten aus den einzelnen Variablen geladen und in HEX konvertiert sowie übergeben. Die Methodik ist dieselbe wie in dem Sweep zuvor.

```
1157 void gpib::on_checkBox_sweep_Rext_clicked() // Banko :
1158 {
1159     if (ui->checkBox_sweep_Rext->isChecked())
1160     {
1161
1162         ui->cb_linereg_vol->setChecked(0);
1163         ui->cb_linereg_curr->setChecked(0);
1164         ui->cb_test_mult->setChecked(0);
1165         ui->cb_loadreg->setChecked(0);
1166         ui->cb_start_ramp->setChecked(0);
1167         ui->cb_stop_ramp->setChecked(0);
1168         ui->checkBox_sweep_Vofsh->setChecked(0);
1169         ui->label_Rext_endValue->setEnabled(1);
1170         ui->comboBox_Rext_end->setEnabled(1);
1171         ui->label_Rext_6->setEnabled(1);
1172         ui->label_Rext_schritte_end->setEnabled(1);
1173         ui->hex_Rext_end->setEnabled(1);
1174         ui->label_Rext_stepSize->setEnabled(1);
1175         ui->spinBox_Rext->setEnabled(1);
1176         this->on_comboBox_Rext_end_currentIndexChanged("");
1177
1178         // Vofsh Sweep Felder deaktivieren
1179
1180         ui->label_Vofsh_endValue->setDisabled(1);
1181         ui->comboBox_Vofsh_end->setDisabled(1);
1182         ui->label_Vofsh_7->setDisabled(1);
1183         ui->label_Vofsh_schritte_end->setDisabled(1);
1184         ui->hex_Vofsh_end->setDisabled(1);
1185         ui->label_Vofsh_stepSize->setDisabled(1);
1186         ui->spinBox_Vofsh->setDisabled(1);
1187
1188         //Ramp Felder aktivieren
1189
1190         ui->label_23->setDisabled(1);
1191         ui->label_28->setDisabled(1);
1192         ui->label_24->setDisabled(1);
1193         ui->label_27->setDisabled(1);
1194         ui->label_26->setDisabled(1);
1195         ui->cb_ramp_current->setDisabled(1);
1196         ui->cb_ramp_voltage->setDisabled(1);
1197         ui->ramp_delay_le->setDisabled(1);
1198         ui->ramp_step_le->setDisabled(1);
1199         ui->ramp_start_le->setDisabled(1);
1200         ui->ramp_end_le->setDisabled(1);
1201
1202         //Sweep Felder aktivieren
1203
1204         ui->startLE->setEnabled(1);
1205         ui->endLE->setEnabled(1);
1206         ui->stepLE->setEnabled(1);
1207         ui->cb_k2460->setEnabled(1);
1208         ui->cb_k2400->setEnabled(1);
1209         ui->label_9->setEnabled(1);
1210         ui->label_29->setEnabled(1);
1211         ui->label->setEnabled(1);
1212         ui->label_2->setEnabled(1);
1213         ui->label_3->setEnabled(1);
1214
1215         // andere Sachen aktivieren
```

Abbildung 71: aktivieren/deaktivieren einzelner Labels, comboBoxen und Textfelder

Damit eine Mehrfachauswahl der Methoden ausgeschlossen ist, werden diese, wie in Abbildung 71 zu sehen, in Abhängigkeit der gewählten Funktion aktiviert bzw. deaktiviert. Wird beispielsweise ein Sweep des Widerstands durchgeführt, so werden die Felder des Sweeps der Offsetspannung nicht benötigt und dementsprechend deaktiviert. Dies wird durch die Funktion „setDisabled“ ausgeführt und mit einer „1“ aktiviert (z.B Code-Zeile 1200).



```
1 #ifndef SLD065_H
2 #define SLD065_H
3
4 #include <vector>
5 #include <iostream>
6 #include <string>
7 #include "Silib_GPIB_Keithley2000.h"
8 #include "Silib_GPIB_Keithley246x.h"
9 #include "Silib_GPIB_Keithley24xx.h"
10 #include "SILIB_GPIB_Interfaces.h"
11 #include "SILIB_GPIB_Device.h"
12 #include "QString"
13 #include "qwt_plot_curve.h"
14 #include "QTextEdit"
15
16 //Neueingebundene Klassen für die Multimeter DMM6500
17 #include "fhdo_usb_interface.h"
18 #include "fhdo_usb_device.h"
19 #include "fhdo_usb_keithleydmm6500.h"
20
21 #include "digitalepotis.h"
22
23
24
25 class sldo65 : public digitalepotis
26 {
27     public:
28         sldo65 ();
29         ~sldo65();
30         /*$PUBLIC_FUNCTIONS$*/
```

Abbildung 72: Vererbung der Potis-Klasse in die sldo65-Klasse

Durch den Code `class sldo65 : public digitalepotis` (Code-Zeile 25) und `#include „digitalepotis.h“` (Code-Zeile 21) wird die Klasse der digitalen Potentiometer an die Hauptklasse `sldo65` vererbt. Hierdurch werden alle Methoden eingebracht und als Bestandteil der `Sldo`-Klasse aufgerufen. Hierdurch wird eine übersichtliche Klassenstruktur etabliert.

## 4. Messergebnisse

In diesem Kapitel werden die unterschiedlichen Messungen für die Charakterisierung des ShuntLDO-Reglers wie z.B. eine Line Regulation vorgestellt. Die Messungen beinhalten unterschiedliche Revisionen des Reglers. Vor den jeweiligen Messungen wird jeweils eine kurze Erläuterung über die Einstellungen sowie der Methodik gegeben.

Eine ausführliche Analyse der Messergebnisse kann in den Arbeiten [ZOR], [BAN2] und [BAN3] nachgelesen werden.

### 4.1 - Spezifikation

Für den Produktionschip aus dem Projekt RD53B gelten die folgenden technischen Spezifikationen. Diese Anforderungen definieren die Qualität des Testchips und müssen eingehalten werden.

*Tabelle 5: Spezifikationen des Chips*

| <b>Technische Spezifikationen</b>                           |                |
|---|----------------|
| Eingangsspannung (LDO-Modus)                                | 1.4V - 2V      |
| Eingangsstrom (SLDO-Modus)                                  | Max. 2 A       |
| Laststrom   | Max. 1 A       |
| Ausgangsspannung  | 0.7 V - 1.32 V |
| Line Regulation   | < 10 mV        |
| Load Regulation   | < 10 mV        |
| Variation der Ausgangsspannung bei transienten Lastwechseln | ± 30 mV        |

## **4.2 - Statische Messungen**

Im nachfolgenden werden die wichtigsten statischen Messungen für die Charakterisierung des ShuntLDO-Reglers vorgestellt. Diese Messungen werden mit Hilfe des ShuLDO-Test-Tools durchgeführt.

### 4.2.1 Line Regulation

Die Line Regulation ist ein Messverfahren, das die Fähigkeit des Reglers charakterisiert, die gewünschte Ausgangsspannung unter einem variierenden Eingangssignal konstant zu halten. Die Formel 4.2\_1 definiert die Line Regulation als Messgröße.

$$Line\ Regulation = \frac{\Delta V_{OUT}}{\Delta V_{IN}} \quad (4.2\_1)$$

Hierbei wird das Verhältnis zwischen der Veränderung der Ausgangsspannung  $\Delta V_{OUT}$  bei sich verändernder Eingangsspannung  $\Delta V_{IN}$  bzw. Eingangsstrom  $\Delta I_{in}$  betrachtet. Der Spannungswert, bei dem sich alle Transistoren in Sättigung befinden, wird als „Startwert“ für die Berechnung der Line Regulation verwendet. Sobald die Eingangsspannung auf ihrem maximalen Wert liegt, ist der Endpunkt der Messung erreicht. Im Idealfall hat die Eingangsspannung keinen Einfluss auf die Ausgangsspannung, daher sind sehr kleine Werte von  $< 0,1\%$  für die Line Regulation gewünscht. Hierbei ist ein Wert von  $< 10\text{ mV}$  durch die Spezifikation vorgegeben.

Die Line Regulation wird gemessen, indem der Regler mit einer linear steigenden Spannung im LDO-Modus und mit einem linear steigenden Strom im SLDO-Modus versorgt wird. Für die Ergebnisse werden alle relevanten Größen des ShuntLDO-Reglers, wie in Abbildung 73 zu sehen, aufgenommen. [KAM][ZOR]

Das Messergebnis zeigt eine Line Regulation im SLDO-High Power Modus. Der Versorgungsstrom wird schrittweise von 0 A bis auf 2 A erhöht. Es ist zu erkennen, dass die Eingangsspannung, sobald alle Transistoren gesättigt sind, einen nahezu linearen Verlauf aufzeigt. Auch die Spannungen  $V_{ref}$  und  $V_{ofsHalf}$  der Bandgap erreichen ihre zu erwartenden Werte von 600 mV und 400 mV. Bei einem ausreichend großem Versorgungsstrom, erreicht auch die Ausgangsspannung den gewünschten Wert von 1.2 V und hält diesen für höhere Versorgungsströme konstant.

Für den Faktor der Line Regulation konnte ein Wert von 21.8 mV ermittelt werden, der für diesen Testchips etwas über der Spezifikation liegt.

Line Regulation - SLDO - HPM -  $I_{in} = 0 \text{ A bis } 2 \text{ A}$

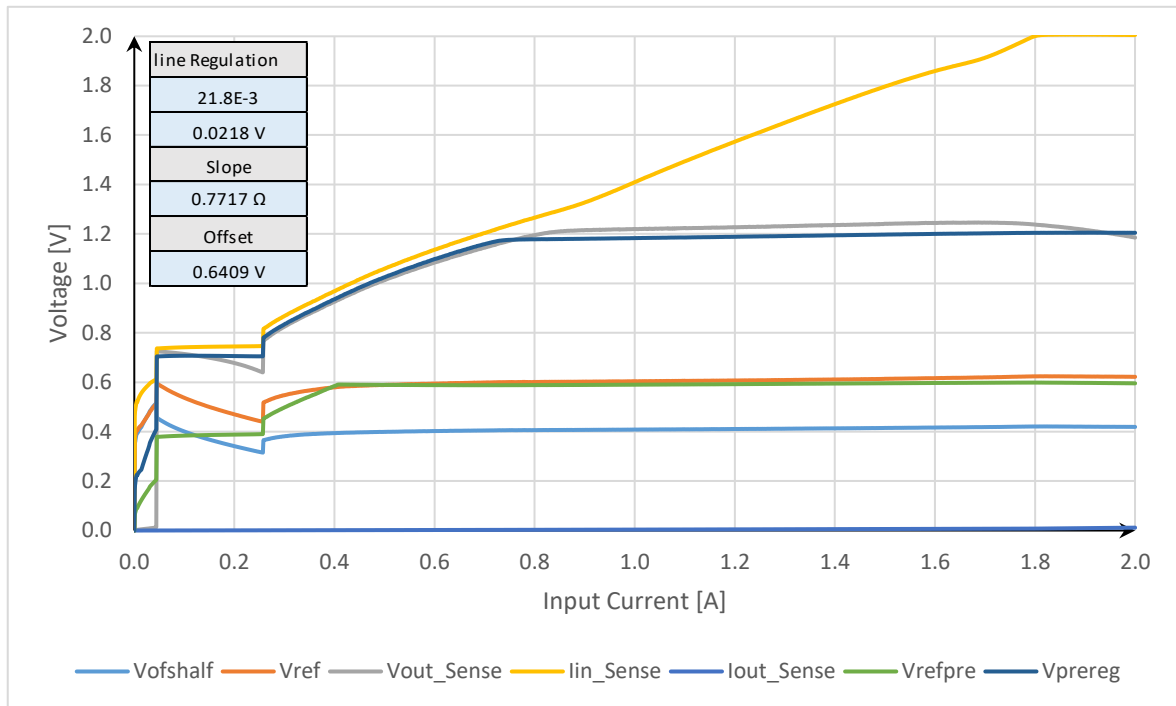


Abbildung 73: Line Regulation im SLDO - HPM Modus für  $I_{in} = 0 \text{ A bis } 2 \text{ A}$

#### 4.2.2 Load Regulation

Die Load Regulation beschreibt die Fähigkeit eines Reglers, trotz Änderungen der Last, einen konstanten Spannungspegel oder Strompegel am Ausgangskanal zu gewährleisten. Im Idealfall ist die Ausgangsspannung unabhängig von der Ausgangslast und dementsprechend sollte die Load Regulation null betragen. Die Formel 4.2\_2 definiert die Load Regulation [KAM]:

$$\text{Load Regulation} = \frac{V_{\text{minLoad}} - V_{\text{maxLoad}}}{\Delta I_{\text{Load}}} \quad (4.2_2)$$

Die Load Regulation wird gemessen, indem der Regler mit einer konstanten Spannung (LDO-Modus) bzw. einem konstanten Strom (SLDO-Modus) versorgt wird. Der Laststrom wird sukzessiv von 0 A auf einen maximalen Strom (1 A) erhöht.

Die Load Regulation sollte unter einem Wert von 10 mV liegen. Abbildung 74 zeigt das Messergebnis für den SLDO-Modus. Da sich der Regler im eingeschwungenem Zustand befindet, verlaufen alle Spannungen nahezu konstant. Für die Load Regulation ergibt sich ein Wert von 17.3 mV, der auch in diesem Fall etwas über der Spezifikation liegt.

#### Load Regulation - SLDO - HPM - $I_{\text{in}} = 1.05 \text{ A}$ , $I_{\text{load}} = 0 \text{ A bis } 1 \text{ A}$

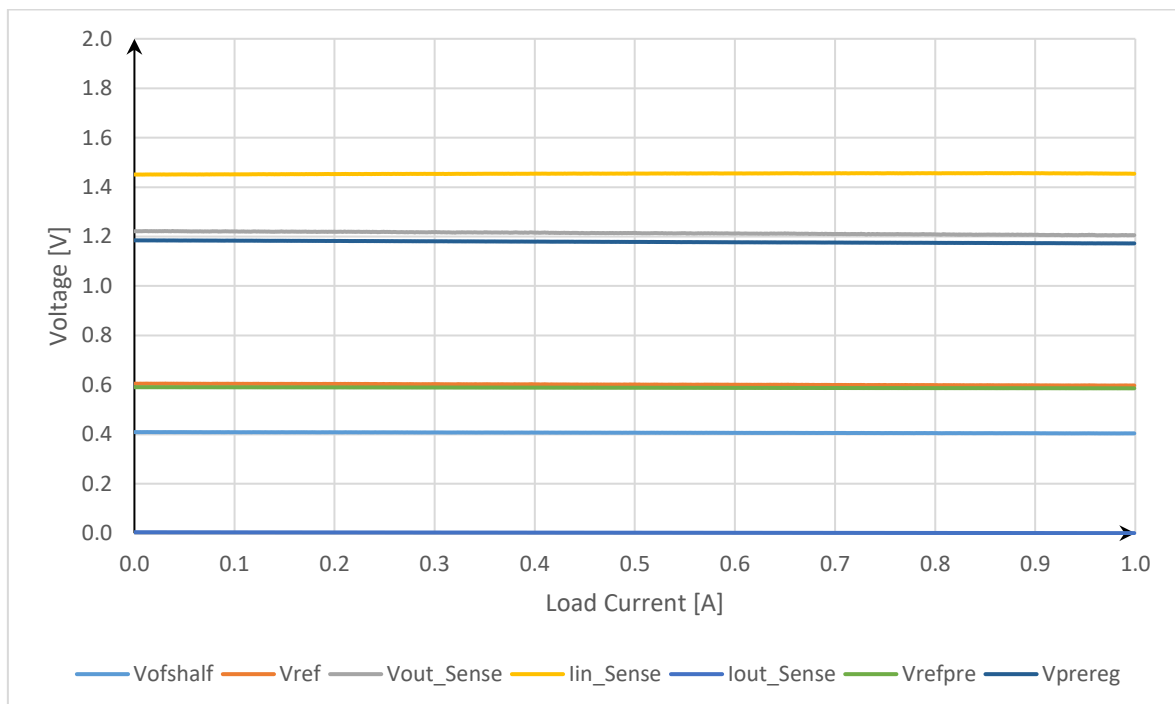


Abbildung 74: Load Regulation im SLDO - HPM Modus für  $I_{\text{in}} = 1,05 \text{ A}$ ,  $I_{\text{load}} = 0 \text{ A bis } 1 \text{ A}$

## **4.3 - Transiente Messungen**

### 4.3.1 Einfluss von Variationen der Eingangsspannung

Für die Messungen im SLDO-Modus wird die Prepower-Platine (siehe Kapitel 3.3.2) zwischen dem Regler und der Spannungsquelle geschaltet, um den Eingangsstrom mit sehr schnellen Flankenzeiten variieren zu können. Die Prepower-Schaltung ermöglicht, dass ein Teil des Eingangsstromes mit Flankenzeiten im Nanosekundenbereich am Regler vorbei nach Masse geleitet wird. [ZOR]

In dieser Messung wird überprüft, wie stark die Eingangs- und Ausgangsspannung auf plötzlich auftretende Signalwechsel am Eingang reagieren. Im Optimalfall sollten sich beide Größen möglichst schnell zurück auf ihren nominellen Wert einpendeln.

Auf der Prepower-Platine wird das Gate eines NMOS Transistors mit einem rechteckförmigem Signal angesteuert. Sobald der Transistorkanal leitfähig wird, wird ein Teil des Versorgungsstroms nach Masse geleitet. Damit ergibt sich durch das Ein- und Ausschalten des Transistors ein rechteckförmiges Eingangssignal. Am Ausgang des Instrumentenverstärkers kann eine Spannung abgegriffen werden, die sich proportional zum Versorgungsstrom verhält.

Abbildung 75 zeigt, dass im SLDO-HPM-Modus zunächst Spannungsspitzen für  $V_{in}$  und  $V_{out}$  in die entgegengesetzte Richtung zu erkennen sind, bevor sich der Nennwert einstellt. In Abbildung 76 wurde auf die steigende Flanke mit einer höheren zeitlichen Auflösung von  $20 \mu\text{s} / \text{div}$  hereingezoomt. Die negative Spannungsspitze ist hierbei nicht mehr zu erkennen. Zudem fällt auf, dass der Verlauf wellig ist. Die Spannungsspitze bei der fallenden Flanke fällt mit einer höheren zeitlichen Auflösung von  $20 \mu\text{s} / \text{div}$  (Abbildung 77) kleiner aus als auf der Abbildung 75. Der wellige Verlauf ist erneut zu erkennen.

Transiente Eingangswchsel - SLDO - HPM -  $I_{in} = 1 \text{ A}$  bis  $1,6 \text{ A}$

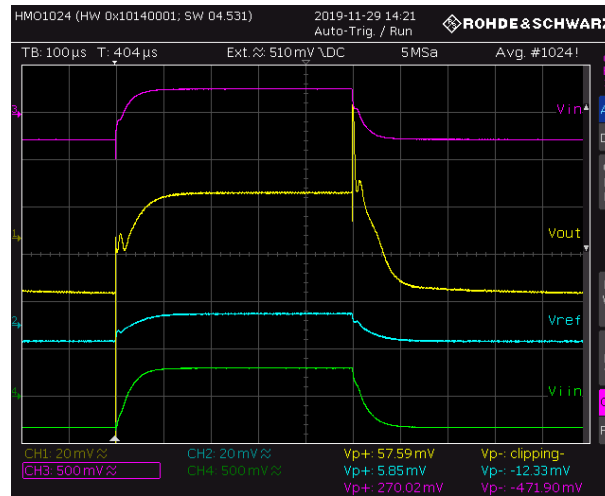


Abbildung 75: Transiente Eingangswchsel im SLDO - HPM Modus für  $I_{in} = 1 \text{ A}$  bis  $1,6 \text{ A}$

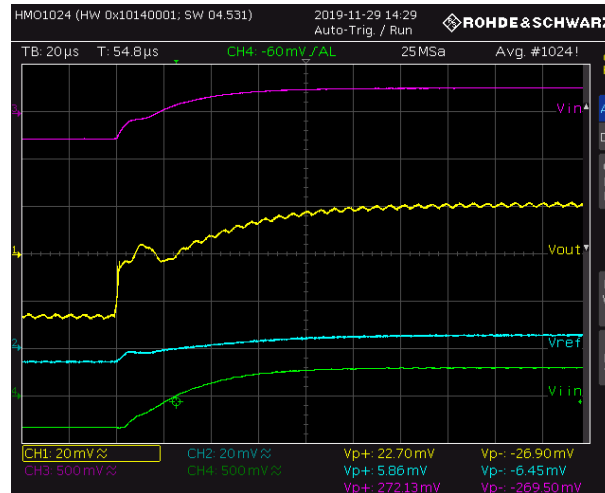


Abbildung 76: Transiente Eingangswchsel im SLDO - HPM Modus für  $I_{in} = 1 \text{ A}$  bis  $1,6 \text{ A}$  (steigende Flanke)

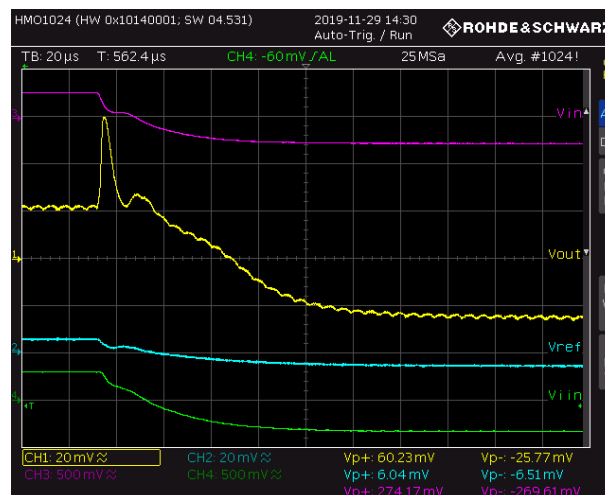


Abbildung 77: Transiente Eingangswchsel im SLDO - HPM Modus für  $I_{in} = 1 \text{ A}$  bis  $1,6 \text{ A}$  (fallende Flanke)

### 4.3.2 Einfluss von Lastwechseln

In diesem Kapitel wird das Verhalten des Reglers bei schnellen Lastwechseln untersucht. Plötzliche Lastwechsel beeinflussen die Ausgangsspannung und führen kurzzeitig zu einem Anstieg bzw. einem Abfall am Ausgang. Untersucht werden soll die Einschwingdauer der Ausgangsspannung nach einem Lastwechsel sowie die Amplituden der Über- bzw. Unterschwinger.

Für die Generierung von Lastwechseln wird der NMOS Transistor (N1) auf der Testplatine mit einem rechteckförmigem Signal aus einem Waveformgenerator angesteuert. Damit wird der Transistor ein- und ausgeschaltet und zieht im geschlossenen Zustand einen Laststrom aus dem Chip. Der Transistor erzeugt Lastvariationen, die beispielsweise zwischen 0 A und 500 mA liegen können. Wird über das Sourcemeter (Keithley 2400) am Ausgang  $V_{out}$  ein zusätzlicher konstanter Laststrom gezogen, so können auch Lastströme generiert werden, die zwischen 450 mA und 950 mA liegen.

Um das schnelle Schaltverhalten eines durch den Regler versorgten Digitalteils nachzubilden, liegen die Anstiegs- und Abfallzeiten des Laststromes im Nanosekundenbereich. An den Eingang des Reglers wird ein konstanter Versorgungsstrom mit dem Keithley 2460 Sourcemeter angelegt. [ZOR]

Die Messergebnisse im SLDO-Modus zeigen, dass bei einem Anstieg des Laststromes die Ein- sowie die Ausgangsspannung zunächst einbrechen, bevor beide Spannungen wieder auf ihren nominellen Wert einschwingen. Sobald sich der Laststrom verringert, kann ein kurzer Überschwinger beobachtet werden. Die Einschwingdauer liegt sowohl bei einer Lastzunahme als auch bei einer Lastabnahme bei etwa 20  $\mu$ s. Die Amplituden der Spannungsspitzen liegen im Bereich zwischen 6.8 mV bis 8 mV, siehe Tabelle 6.

Tabelle 6: Spannungsbereich und Einschwingzeit von  $V_{out}$  für Lastwechsel bei  $I_{in} = 1$  A im HPM

|            |   |               | $\Delta V_{out}$ | $\Delta t$ |            |
|------------|---|---------------|------------------|------------|------------|
| SLDO-Modus | $I_{in} = 1$ A<br>$I_{load} = 0$ A - 500 mA | Testchip 3.D2 | Steigende Flanke | - 7,94 mV  | 20 $\mu$ s |
|            |   |               | Fallende Flanke  | 6,79 mV    | 20 $\mu$ s |



Transiente Lastw. - SLDO - HPM -  $I_{in} = 1\text{ A}$ ,  $I_{load} = 0\text{ A}$  bis  $500\text{ mA}$

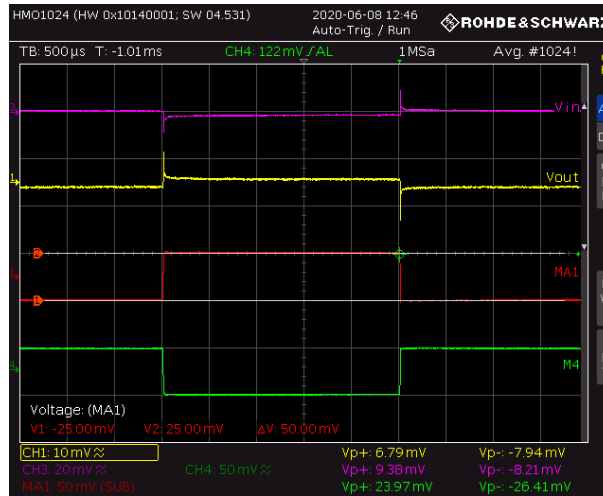


Abbildung 78: Transiente Lastwechsel im SLDO - HPM Modus für  $I_{in} = 1\text{ A}$ ,  $I_{load} = 0\text{ A}$  bis  $500\text{ mA}$  ( $500\text{ μs/Div}$ )

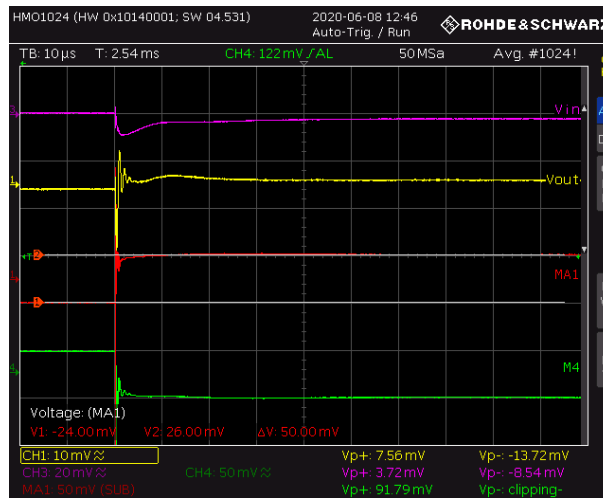


Abbildung 79: Transiente Lastwechsel im SLDO - HPM Modus für  $I_{in} = 1\text{ A}$ ,  $I_{load} = 0\text{ A}$  bis  $500\text{ mA}$  (steigende Flanke) ( $10\text{ μs/Div}$ )

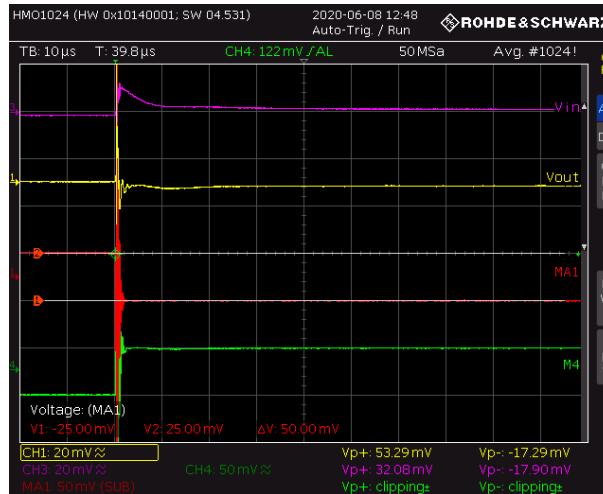


Abbildung 80: Transiente Lastwechsel im SLDO - HPM Modus für  $I_{in} = 1\text{ A}$ ,  $I_{load} = 0\text{ A}$  bis  $500\text{ mA}$  (fallende Flanke) ( $10\text{ μs/Div}$ )

### 4.3.3 Untersuchung des Einschaltverhaltens

Eine Untersuchung des Reglerverhaltens während der Power-Up-Phase wird ermöglicht, indem die Schaltung mit einem rampenförmigem Signal versorgt wird. Das ShuntLDO-Test-Tool bietet die Möglichkeit ein solches Eingangssignal mit einer frei konfigurierbaren Anstiegszeit zu generieren. Das Einschaltverhalten wird sowohl im LDO- auch als im SLDO-Modus betrachtet. Für den SLDO-Modus ist der Verlauf des Versorgungsstromes sowie der Eingangs- und Ausgangsspannung interessant.

Die Messung des Einschaltverhaltens werden im HPM durchgeführt. Für die Messungen zur Untersuchung des Einschaltverhaltens im SLDO-Modus muss der Instrumentenverstärker versorgt werden, um  $V_{in}$  bzw.  $I_{in}$  messen zu können. Die transienten Ergebnisse werden an einem Oszilloskop abgegriffen und über das ShuntLDO-Test-Tool werden die Parameter der Rampenfunktion festgelegt. Es ist ratsam, nach einem Durchlauf des Rampensignals einen Verzögerungsblock einzubauen. Hierfür müssen folgende Einstellungen in den Trigger-Menü am Sourcemeter durchgeführt werden:

Menü öffnen → Trigger Configure → Block 3 anklicken → Insert After → Action → Constant Delay → 1 ms

Hierdurch wird eine Verzögerung von 1 ms eingestellt, so dass die nächste Rampe mit einer Verzögerung beginnt.

Des Weiteren muss darauf geachtet werden, dass die „settings ramp“ Einstellungen aus dem Steuerprogramm am PC korrekt im Sourcemeter übertragen worden sind, da es beim Überspielen der Parameter zu Fehlern kommen kann. Die Parameter können in den Trigger Einstellungen überprüft werden. Falls diese nicht übernommen worden sind, können sie manuell neu geladen werden, indem der Trigger für 1-2 Sekunden gedrückt gehalten und dann neu starten ausgewählt wird.

Für den SLDO-HPM-Modus wurden die Messungen sowohl mit als auch ohne StartUp-Schaltung ausgeführt. Die Messergebnisse zeigen für den SLDO-HPM-Modus mit aktiver StartUp-Schaltung, dass die Ausgangsspannung einen kurzen Einbruch hat, bevor sich der Verlauf abrupt wieder einpendelt. Zeitgleich weist die Eingangsspannung ebenfalls einen leichten Knick auf, siehe Abbildung 81. Abbildung 82 weist einen sauberen Verlauf aus.

Einschaltverhalten - SLDO - HPM -  $I_{in} = 0$  A bis 1 A

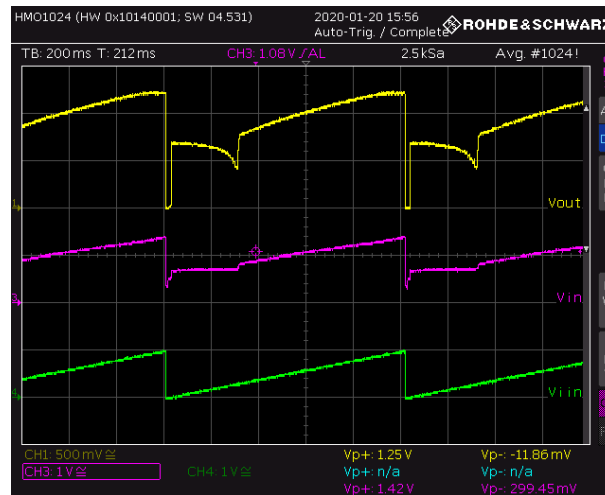


Abbildung 81: Einschaltverhalten im SLDO - HPM Modus für  $I_{in} = 0$  A bis 1 A,  $t_{rise} = 1,05$  s (mit Start-Up)

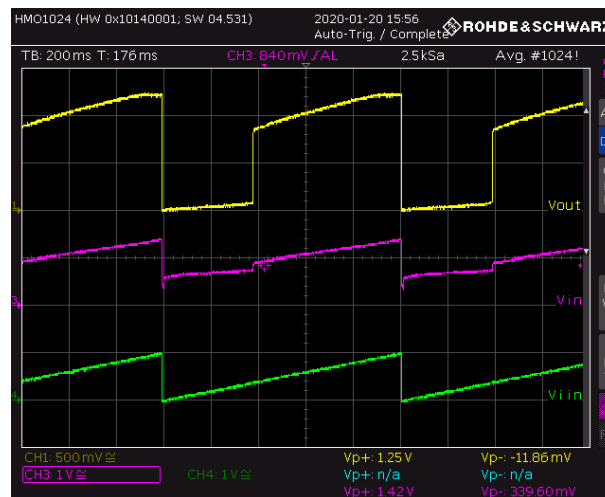


Abbildung 82: Einschaltverhalten im SLDO - HPM Modus für  $I_{in} = 0$  A bis 1 A,  $t_{rise} = 1,05$  s (ohne Start-Up)

#### 4.3.4 Validierung der Overvoltage Protection

Die Aufgabe der Overvoltage Protection (OVP) ist es, den Regler vor zu hohen Spannungen zu schützen.

Untersucht wird die Funktionalität des Überspannungsschutzes mit einer plötzlichen Erhöhung des Eingangsstromes. Dabei wird der Strom so gewählt, dass die Eingangsspannung Werte von 2 V überschreiten würde und die OVP aktiv wird.

Zur Ausführung der Messung wird die Prepower Platine an die Testplatine angeschlossen. Das Sourcemeter versorgt den Chip mit einem konstanten Strom von 2 A. Zusätzlich wird der Transistor auf der Prepower Platine angesteuert, um ein rechteckförmiges Versorgungssignal zu generieren. Für die Messung in Abbildung 83 stellt sich ein Versorgungsstrom ein, der zwischen 1 A und 2 A variiert. Sobald der Versorgungsstrom auf 2 A ansteigt, nimmt auch die Eingangsspannung zu.

Die transienten Ergebnisse im SLDO-Modus zeigen, dass bei einem schnellen Anstieg des Eingangsstromes  $I_{in}$  die Eingangsspannung für einen kurzen Moment minimal über den von der OVP begrenzten Wert steigt, bis die OVP reagiert und die Eingangsspannung begrenzt wird. In denselben Momenten kommt es auch zu erhöhten Werten in Höhe von ca. 130 mV der Ausgangsspannung. Dieses Ergebnis würde eine schlechte Line Regulation bedeuten, so dass eine neue Messung mit einem eventuell größer gewählten  $I_{in\_min}$  verglichen werden sollte. [ZOR]

Tabelle 7: Übersicht zur Messung der Overvoltage Protection (transient)

| transientes Verhalten   |                        |   |               |              |
|-------------------------|------------------------|---|---------------|--------------|
| SLDO-Modus (High Power) | $I_{in} = 2 \text{ A}$ | $I_{iin} = \pm 1 \text{ A}$<br>$t_{rise} = 10 \text{ ns}$ | Testchip 3.D2 | Abbildung 83 |

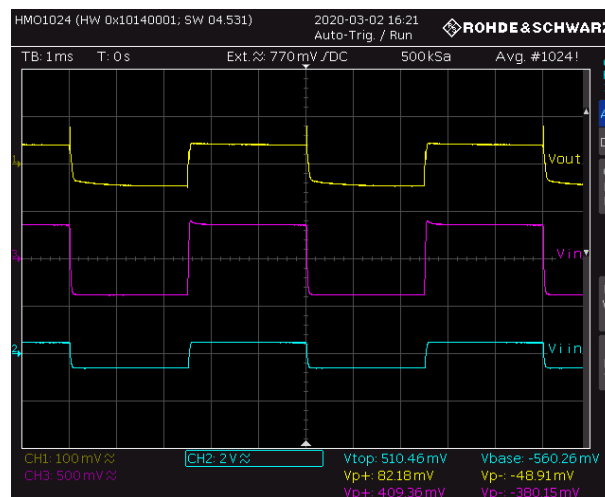


Abbildung 83: OVP - SLDO - HPM -  $I_{in} = 1 \text{ A} - 2 \text{ A}$ ,  $t_{rise} = 10 \text{ ns}$

#### 4.3.5 Validierung der Undershunt Current Protection

Mit Hilfe der Under Shunt Current Protection (USCP) wird der Regler vor zu großen Lastströmen geschützt. Ohne die Schutzschaltung bricht bei einem zu hohen Laststrom die Eingangsspannung ein und führt dazu, dass der gesamte Regler ausfällt. Für Messungen im seriellen Betrieb konnte zudem festgestellt werden, dass der Einbruch der Eingangsspannung eines Reglers zu Überspannungen auf die weiteren Regler führt. Die USCP wird auf der Testplatine über den Anschluss P<sub>19</sub> aktiviert.

Die Funktionalität der USCP kann entweder mit Hilfe einer statischen Messung [ZOR] oder einer transienten Messung untersucht werden. Für die transienten Untersuchung wird ein periodischer rechteckförmiger Laststrom aus dem Chip gezogen. Die maximale Amplitude wird so gewählt, dass ein Überlastfall eintritt.

Für die Umsetzung der Messung wird mit dem Sourcemeter, welches als Last fungiert, ein konstanter Strom am Anschluss V<sub>out</sub> aus dem Chip gezogen. Außerdem zieht der MOSFET einen zusätzlichen periodischen Laststrom. Sobald die USCP eine zu hohe Last detektiert, wird die Referenzspannung gesenkt und als Konsequenz sinkt auch die Ausgangsspannung. Die Eingangsspannung bleibt weiterhin konstant. Sobald der Überlastfall endet, steigen die Referenz- und Ausgangsspannung wieder auf ihre nominellen Werte.

Die transiente Messung der USCP wird mit dem Oszilloskop durchgeführt. Zudem wird über den Waveform-Generator ein Rechtecksignal eingespeist, welches den Stromfluss durch den Transistor steuert.

Abbildung 84 zeigt vier unterschiedliche Messergebnisse mit zunehmendem Laststrom. In der ersten Messung tritt kein Überlastfall ein. Alle Spannungen bleiben unverändert. In der zweiten und dritten Messung wurde ein Überlastfall detektiert und als Folge die Referenz- sowie die Ausgangsspannung gesenkt. Es ist zu erkennen, dass die Eingangsspannung V<sub>in</sub> nicht von dem Überlastfall beeinflusst wird. Für den Testaufbau wurde der Laststrom weiter vergrößert, bis auch die Eingangsspannung einbricht. Im realen Betrieb würde die Reduktion der Ausgangsspannung zu einer Abnahme des Laststroms führen und damit den Regler vor einem Ausfall schützen.

#### USCP (transient) - SLDO - HPM - I<sub>in</sub> = 1 A, I<sub>load</sub> = 0 A - 1 A

Tabelle 8: Lastströme der Messungen bei I<sub>in</sub> = 1 A im HPM

|            |  |               | I <sub>load</sub>        | I <sub>load, wave</sub> | I <sub>load, ges</sub> |         |
|------------|--|---------------|--------------------------|-------------------------|------------------------|---------|
| SLDO-Modus | I <sub>in</sub> = 1 A<br>V <sub>wave</sub> = 2,1 V<br>f = 100 Hz | Testchip 3.D2 | USCP inaktiv             | 430 mA                  | 557 mA                 | 987 mA  |
|            |  |               | USCP aktiv               | 435 mA                  | 557 mA                 | 992 mA  |
|            |  |               |                          | 515 mA                  | 536 mA                 | 1051 mA |
|            |  |               | Einbruch V <sub>in</sub> | 520 mA                  | 533 mA                 | 1053 mA |

Die nachfolgenden Abbildungen zeigen die Messergebnisse mit aktivierter USCP-Schaltung bei einem Versorgungsstrom von  $I_{in} = 1\text{ A}$  für einen Lastwechsel von  $I_{load} = 0\text{ A} - 1\text{ A}$ . In den einzelnen Aufnahmen wird der Laststrom stetig in 5 mA Schritte vergrößert, bis die Ausgangsspannung abgesenkt ist, bis zum Einbruch.

Da  $I_{load, ges}$  größer ist als  $I_{in}$  mit 1 A, sind hier Ungenauigkeiten bei Ablesung zu vermuten, da es nicht möglich ist mehr Strom zu ziehen, als eingespeist wird.

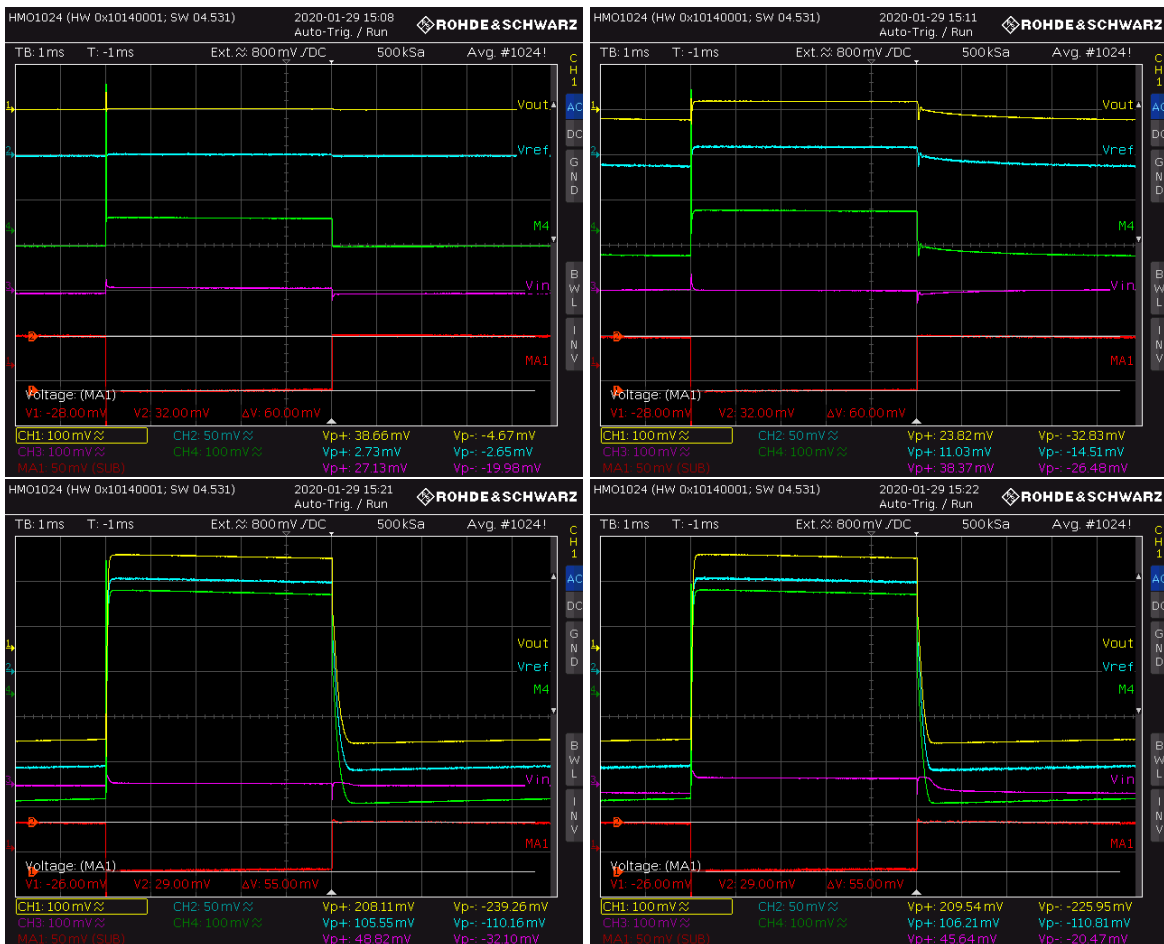


Abbildung 84: USCP (transient) - SLDO - HPM -  $I_{in} = 1\text{ A}$ ,  $I_{load} = 0\text{ A} - 1\text{ A}$  (Erhöhung der Last bis Einbruch)



## 4.4 - Messungen im seriellen Betrieb

In diesem Kapitel wird das Verhalten der Regler in einer seriellen Verschaltung untersucht. Alle Messungen der seriellen Verschaltung wurden mit einer Line Regulation (SLDO HPM – Line Regulation –  $I_{in} = 0 \text{ A}$  bis  $1.5 \text{ A}$ ) im 4-Wire Modus, mit aktiver StartUp-Schaltung und OVP durchgeführt.

Die Messungen haben gezeigt, dass die Ergebnisse dem zu erwartenden Verlauf mit geringen Abweichung entsprechen, die voraussichtlich auf Temperatureinflüsse oder Variationen der Bauteile zurückzuführen sind.

### Serielle Verschaltung

Zur Implementierung einer seriellen Versorgungskette muss die Konfiguration des Reglers angepasst werden, was nachfolgend detailliert erläutert wird. Hierzu wird die Konfiguration des Jumpers P8 anhand der Schaltpläne sowie des PCBs beschrieben.

Abbildung 86 zeigt einen Auszug des Schematics mit der zu steckenden Verbindung am Jumper P8. Durch den Jumper P8 kann zwischen dem Einzelbetrieb (Abbildung 87) und dem seriellen Betrieb (Abbildung 88) gewählt werden.

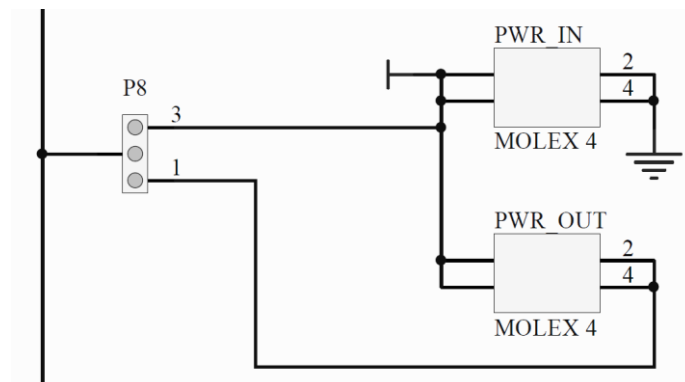


Abbildung 86: Schematic Auszug des Jumpers P8 und der Molex Stecker Verbindungen PWR\_IN & PWR\_OUT

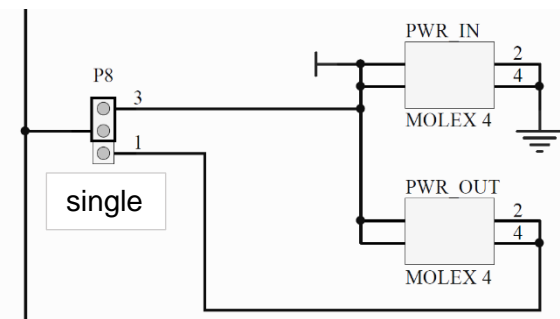


Abbildung 87: Gesetzter Jumper P8 auf single

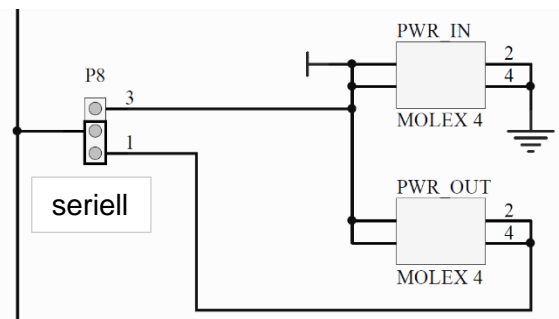


Abbildung 88: Gesetzter Jumper P8 auf seriell

Abbildung 87 zeigt die Jumperstellung P8 auf Position 3. In diesem Fall fließt der Strom zum oberen Pfad, in den Anschluss 3 und somit durch PWR\_IN über die Anschlüsse 2 und 4 nach Masse. Diese Verbindung ist im Einzelbetrieb, bei der parallelen Verschaltung oder als letzte Verbindung in einer Kette zu stecken.



Die Abbildung 88 zeigt die Jumper-Konfiguration für den seriellen Betrieb. Durch den Jumper P8 auf Position 1 kann demnach eine Kette in gewünschter Länge aufgebaut werden.

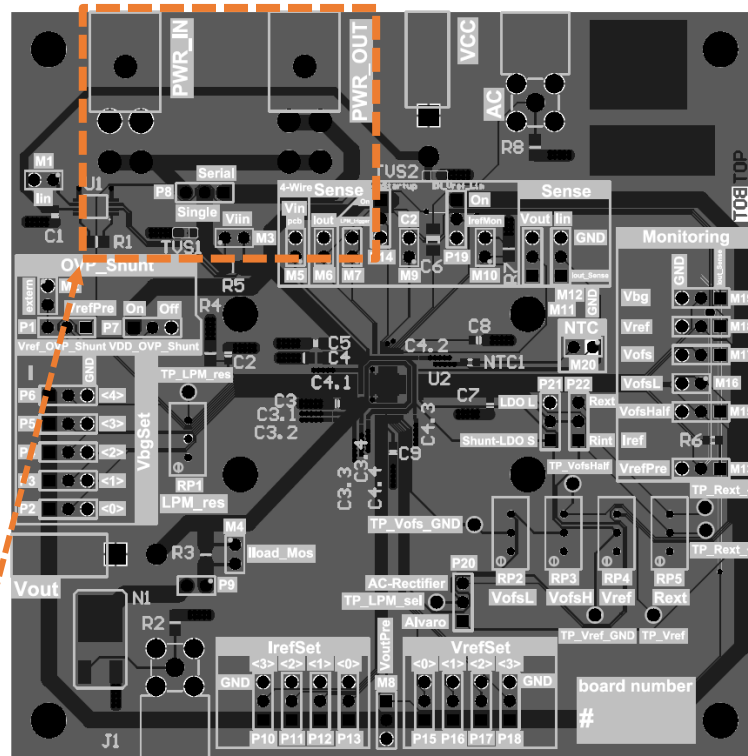


Abbildung 89: PCB RD53B

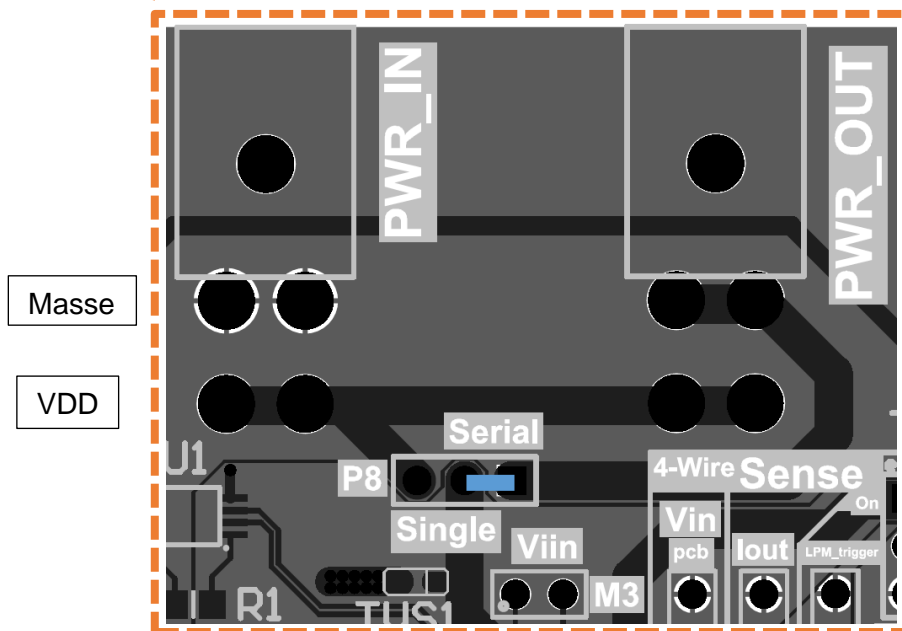


Abbildung 90: Nahaufnahme an den gesetzten Jumper P8 auf seriell

Abbildung 90 zeigt eine Nahaufnahme des gesetzten Jumpers P8 auf der PCB RD53B. Hierdurch wird ersichtlich, dass durch die serielle Verbindung an P8 die Verbindung zu PWR\_OUT hergestellt wird.

Für die serielle Verschaltung existieren zwei Varianten.

Variante 1: Bei der ersten Variante sind alle Regler auf P8.1 (seriell) gesteckt. Wie in Abbildung 91 dargestellt muss am Ende der Kette eine Brücke gesetzt werden, damit der Strom weiter zum Chip geleitet wird.

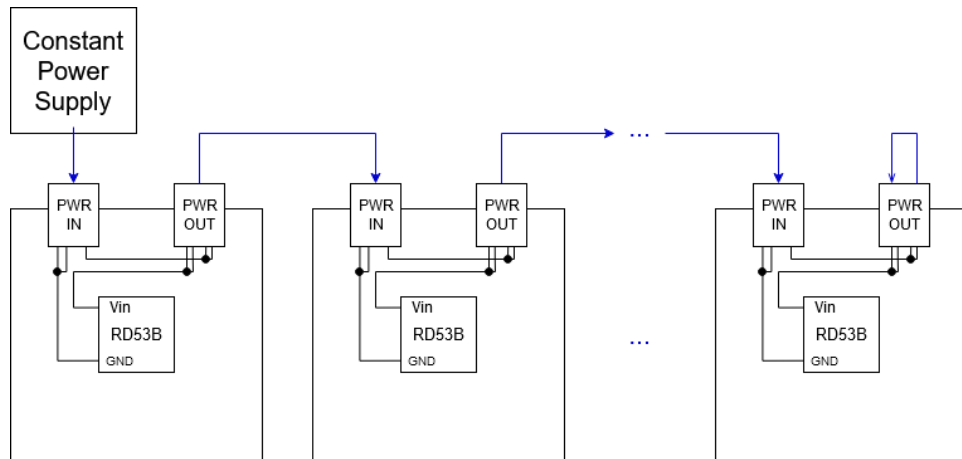


Abbildung 91: Veranschaulichung der seriellen Kette / Variante 1

Variante 2: Wie in Abbildung 88 erläutert und in Abbildung 92 dargestellt, kann ebenfalls eine Kette aufgebaut werden, indem alle Regler auf P8.1 gesteckt werden und der letzte Regler auf P8.3 gesteckt wird.

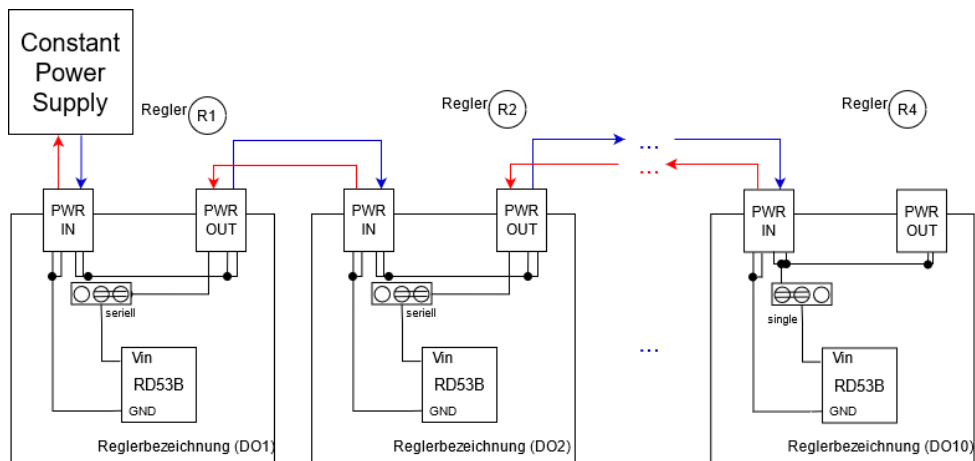


Abbildung 92: Serielle Verschaltung anhand der Molex Stecker und Jumper P8 / Variante 2

Das Sourcemeter Keithley 2460 wird als Quelle betrieben und an den Anschluss PWR\_IN der ersten Platine angeschlossen. Die serielle Kette wird dadurch implementiert, indem das Kabel von PWR\_OUT der ersten Platine mit dem Anschluss PWR\_IN der zweiten Platine verbunden wird. Diese Verbindung kann bei den nachfolgenden Reglern entsprechend fortgeführt werden, um eine Kette in beliebiger Länge aufzubauen (siehe Abbildung 92).

### Massen-Potentialunterschiede von 6 Reglern in Reihe (Anschluss M12)

In der nachfolgenden Abbildung wird der Massen-Potentialunterschied vom Regler R1 hin zu jedem weiteren Regler abgebildet, d.h. die Masse vom Regler R1 zur Masse des Reglers R2, Masse des Reglers R1 zur Masse des Reglers R3, usw. Es kann beobachtet werden, dass die Spannungen linear ansteigen und bei einem Eingangsstrom von 1.5 A eine Differenz von ca. 2 V zu jedem Regler vorliegt.  $V_{sup}$  wird bezogen auf die globale Masse gemessen und alle weiteren Werte bezogen auf die lokale Chip-Masse. Der Offset wurde über  $V_{ofs}$  auf 800 mV eingestellt. Die Spannung über den Reglern wurde über den Anschluss  $I_{in}$  (M12) abgegriffen.

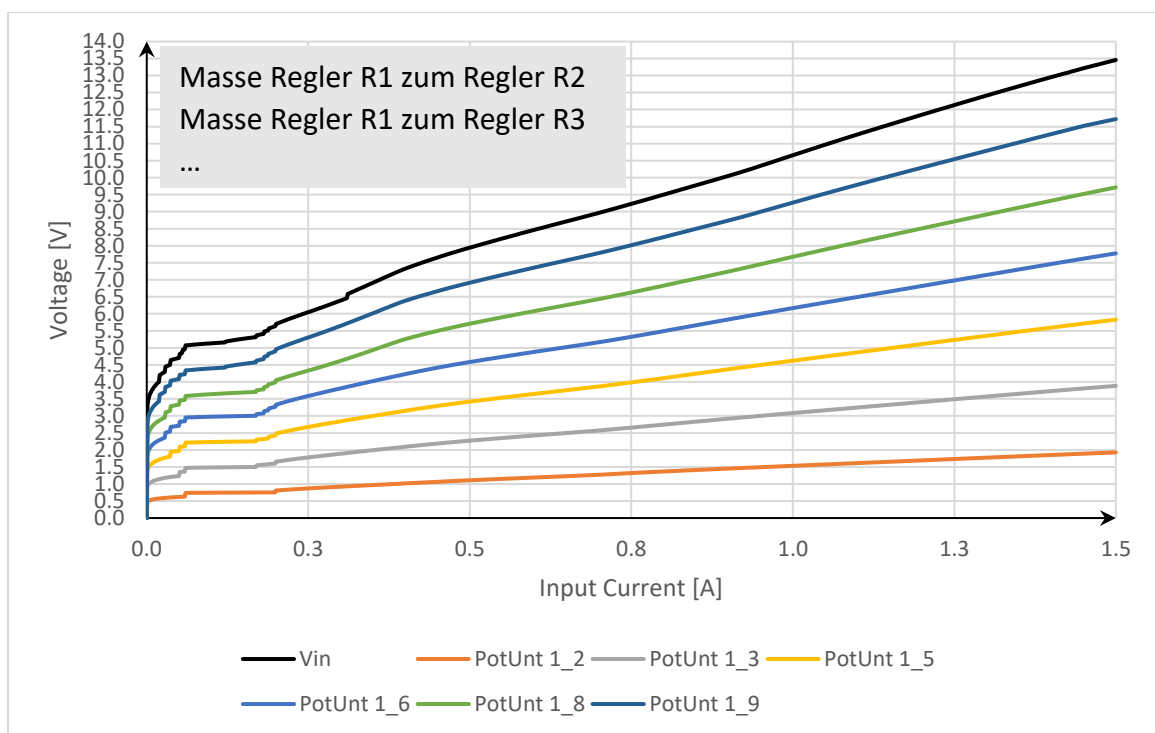


Abbildung 93: Massen-Potentialunterschiede der Regler R1 bis R6

## 4.5 - Messung im redundanten Betrieb

Die Untersuchung des redundanten Betriebs wird mit einer Line Regulation im SLDO-Modus durchgeführt. Die SMU erhöht den Versorgungsstrom schrittweise und über die Multimeter werden auf den Leiterplatten am Anschluss  $V_{in}$  die Reglerströme abgegriffen.

Für die Überprüfung des redundanten Betriebes wurden bis zu vier Regler parallel verschaltet werden. Die Verschaltung der einzelnen Regler wird mit Hilfe der Leitung aus Kapitel 9.1.1.5 ermöglicht und nach Abbildung 94 umgesetzt. Das Ziel dieser Messung ist es, zu untersuchen, ob sich der Versorgungsstrom nahezu gleichmäßig auf alle parallel geschalteten Regler verteilt, damit ein zuverlässiger redundanter Betrieb gewährleistet werden kann.

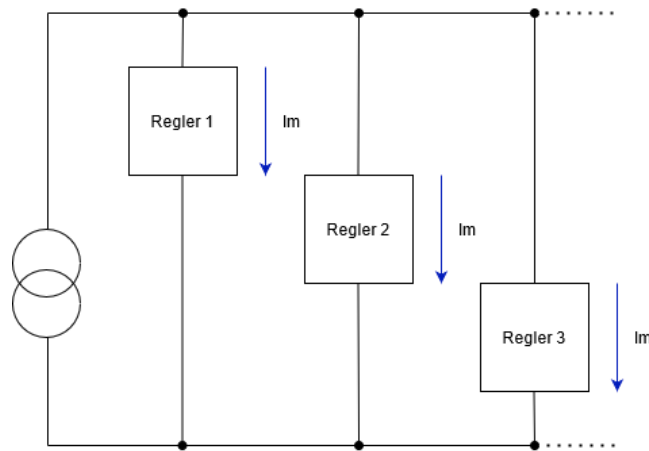


Abbildung 94: Konzept für die parallele Verschaltung der Regler

In Abbildung 95 ist ein Messergebnis mit drei parallel geschalteten Reglern dargestellt. Auf der x-Achse ist der Versorgungsstrom und auf der y-Achse der Strom durch die einzelnen Regler aufgetragen. Für die Messung wird der Versorgungsstrom sukzessiv von 0 A auf 4.3 A mit einer Schrittweite von 1 mA erhöht. Für hohe Versorgungsströme ab etwa 4 A ist zu erkennen, dass Temperatureinflüsse auftreten, welche die Messergebnisse verfälschen und dafür sorgen, dass die Stromverteilung ungleichmäßiger wird. In dem Bereich zwischen 0 A - 4 A verteilt sich der Gesamtstrom hingegen nahezu gleichmäßig auf die Regler. Die Stromabweichungen zwischen den einzelnen Reglern liegt hier etwa im Bereich zwischen 10 mA - 33 mA.

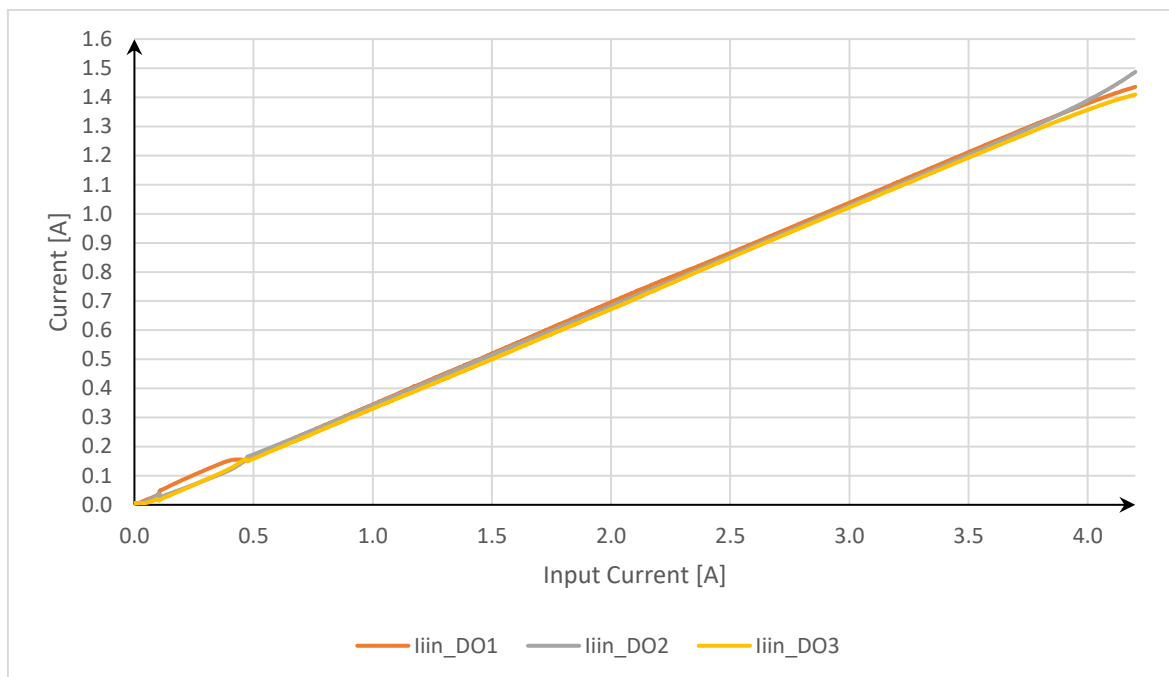


Abbildung 95: Messung von drei parallelen Reglern zur Überprüfung der Stromaufteilung (DO1,DO2,DO3)

## 4.6 - Validierung der OVP im seriellen Betrieb

Im folgenden Versuchsaufbau wird die Funktionalität der OVP im seriellen Betrieb überprüft. Hierfür wird die Stromquelle so eingestellt, dass die Eingangsspannung jedes Reglers bei etwa 1.9 V liegt.

Analysiert werden die transienten Messungen aller Reglereingangsspannungen, in dem Moment, bei dem die Spannung des überlasteten Reglers zusammenbricht.

Für die Überprüfung der OVP gibt es zwei unterschiedliche Varianten. Bei einer seriellen Kette von vier Reglern wird der erste Regler durch eine hohe Last zum Einbruch gebracht. Die zweite Variante ist, dass zwischen den Reglern R1 und R2 die PrePower Platine geschaltet wird und hierdurch ein weiter Strompfad erzeugt wird, um einen Teil des Versorgungsstromes am Regler R1 vorbeizuführen und somit seine Eingangsspannung zum Einbruch zu bringen. Bei beiden Varianten wird jeweils das hohe Potenzial des überprüften Reglers und das lokale Massepotenzial (GND) mit dem Oszilloskop aufgenommen. Über die Mathematikfunktion wird dann die Spannung des jeweiligen Reglers ermittelt (Formel 4.6\_1). Beide Varianten werden nachfolgend näher erläutert und ihre Messergebnisse analysiert.

$$\text{Mathematikfunktion} = \text{Regler } X_{\text{hohes Potenzial}} - \text{Regler } X_{\text{Massepotenzial}}$$

(4.6\_1)

### Vergleich der OVP mit interner oder externer Referenz

Um die Messergebnisse auf Plausibilität zu prüfen, werden diese bei eingeschalteter und ausgeschalteter OVP verglichen. Für die Deaktivierung der OVP wird ihre Referenzspannung  $V_{\text{refPre}}$  hochgesetzt, sodass die Schutzschaltung im relevanten Spannungsbereich keinen Einfluss hat bzw. nicht aktiv wird. Die interne Referenz  $V_{\text{refPre}}$  ist so gewählt, dass die OVP bei einem Spannungswert von 2 V aktiv wird. Wird eine externe Referenz von 0.7 V an  $V_{\text{ref\_OVP\_Shunt\_ext}}$  angelegt, so sollte die OVP erst bei einer Spannung von 2.33 V greifen.

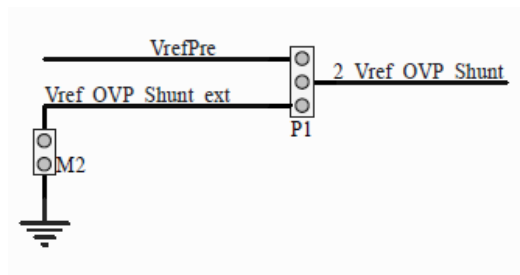


Abbildung 96: Schematic Auszug des Jumpers P1 zur Auswahl der Referenz

Aufbau zur Überprüfung der OVP mit der PrePower-Platine

Bei den Messungen wird die PrePower-Platine zwischen den Reglern R1 und R2 geschaltet, um einen Teil des Versorgungsstromes am Regler R1 vorbeizuführen und somit seine Eingangsspannung zum Einbruch zu bringen. Das Gate des Transistors wird über den Anschluss J1 angesteuert, wodurch der Transistor einen Teil des Stromes an R1 vorbei führt.

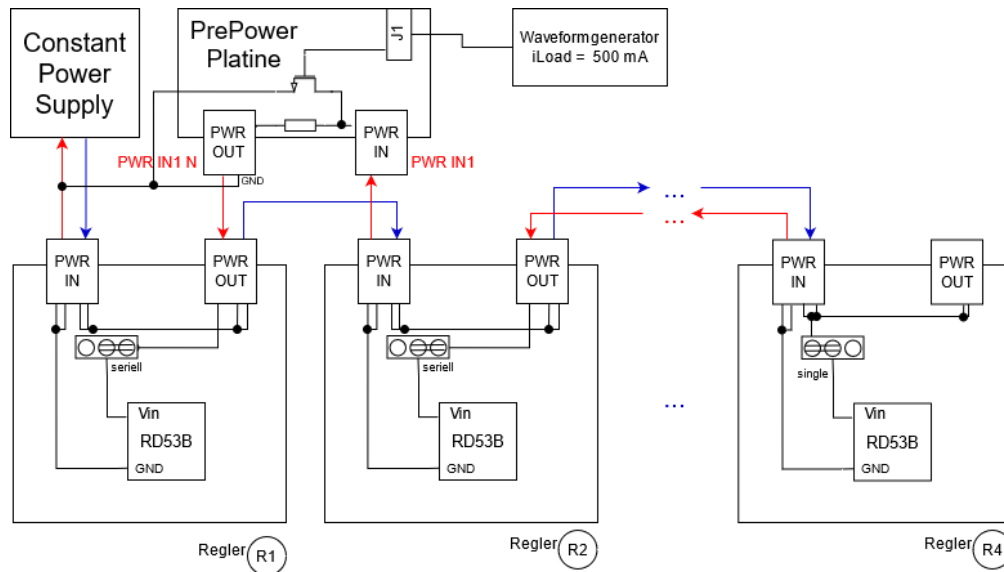


Abbildung 97: serielle Verschaltung mit vorgeschalteter PrePower Platine

Abbildung 97 zeigt das Konzept für die Verschaltung mit vorgeschalteter PrePower-Platine. Die Platinen werden folgendermaßen miteinander verbunden:

Die lokale Masse (GND Anschluss) des Reglers R2 wird an den Anschluss PWR IN1 der PrePower Platine angeschlossen. PWR IN1 N vom Stecker PWR OUT der PrePower Platine wird mit dem GND des PWR OUT Steckers vom Regler R1 verbunden. Das GND der PrePower Platine wird an die globale Masse angeschlossen.

Dementsprechend wird im regulären Betrieb, wenn der NMOS Transistor inaktiv ist, der Versorgungsstrom über den Widerstand auf der PrePower Platine direkt an den Regler weitergeleitet, wird der NMOS angesteuert, fließt ein Teil des Stromes über den Transistor vorbei am Regler in die globale Masse und ein zusätzlicher Strompfad wird gebildet. Durch den reduzierten Stromfluss durch den Regler bricht die Eingangsspannung abrupt ein.

Die Abbildungen 98 bis 100 zeigen die Messergebnisse des Reglers R2 sowie der PrePower-Platine bei einem Eingangsstrom von 1.5 A und einer Waveformgenerator-Spannung  $V_{\text{wave}}$  von 5 V. Der Strom über der PrePower-Platine weist bei allen Messungen einen hohen Anstieg auf, bevor dieser abrupt abfällt und ein erneuter konstanter Anstieg sowie Abfall stattfindet. In Abbildung 100 bei der Messung mit einer externen Referenz von 0.7 V weist der Strom zudem einen oszillierenden Verlauf auf. In Abbildung 98, bei der internen Referenz zeigt die Eingangsspannung  $V_{\text{in}2}$  des zweiten Reglers einen leichten Knick auf und fällt nachfolgend konstant ab. Bei der externen Referenz von 0.6 V zeigt die Eingangsspannung  $V_{\text{in}2}$  einen oszillierenden Verlauf auf.

## Regler R2



Abbildung 98: **Regler R2** – Sourcemeter 1.5 A, interne Referenz,  $V_{wave} = 5 V$ , PrePower



Abbildung 99: **Regler R2** – Sourcemeter 1.5 A, externe Referenz 0.6 V,  $V_{wave} = 5 V$ , PrePower

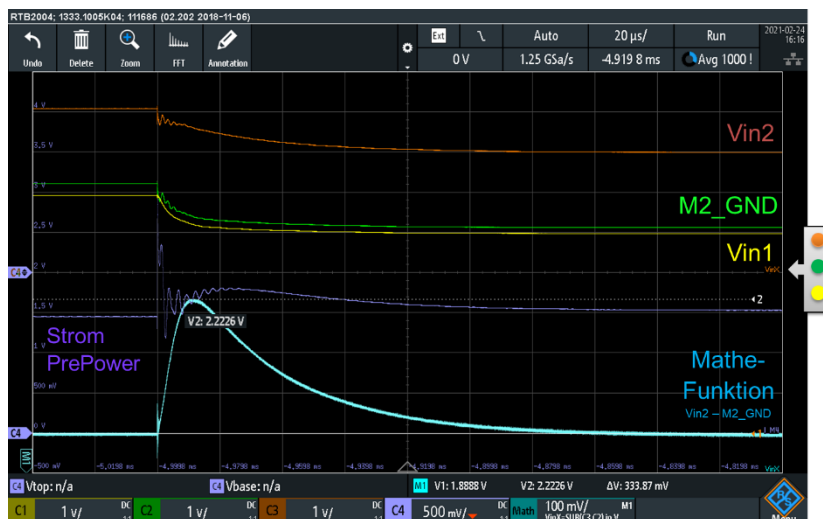


Abbildung 100: **Regler R2** – Sourcemeter 1.5 A, externe Referenz 0.7 V,  $V_{wave} = 5 V$ , PrePower



## Serielle Kette ohne PrePower Platine

In diesem Abschnitt wird eine weitere Methode vorgestellt, die es möglich macht, die Funktionalität der OVP in einer transienten Messung zu untersuchen.

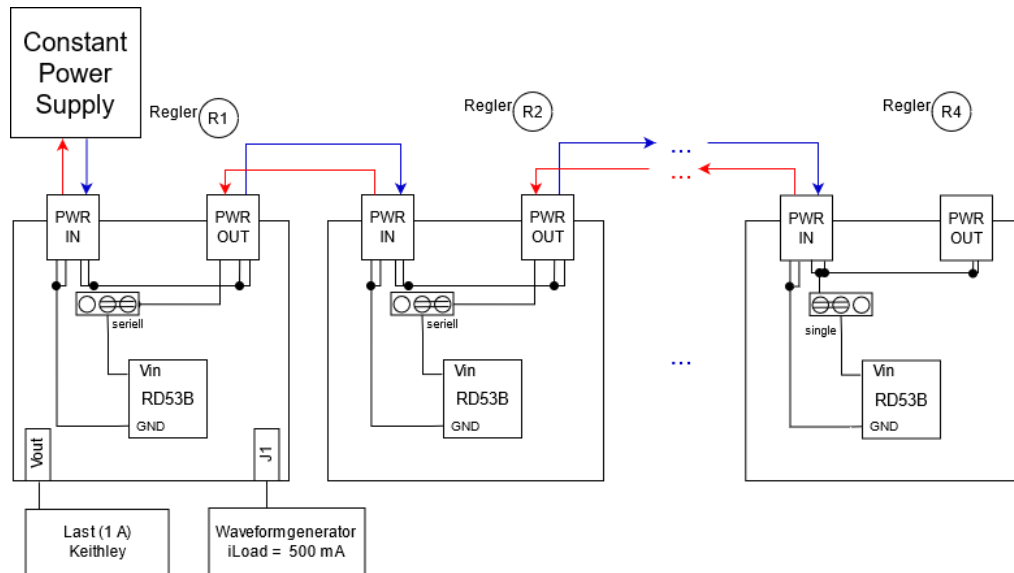


Abbildung 101: Schaltplan zur seriellen Verschaltung

Abbildung 101 zeigt einen Schaltplan der seriellen Verschaltung mit vier Reglern. Die Stromquelle wird an den Regler R1 angeschlossen und der Strom fließt durch alle Regler der Kette. Am Anschluss  $V_{out}$  wird eine konstante Last von 1 A gezogen. Über den NMOS Transistors, wird der Regler mit einem zusätzlichem periodischem Laststrom mit einer Amplitude von ca. 500 mA belastet. Hierdurch wird die Spannung des Reglers R1 durch eine Überlastsituation zum Einbruch gebracht und anschließend wird überprüft, ob sich wie in den vorherigen Messungen Überspannungen an den weiteren Reglern in der Kette einstellen.

Am Sourcemeter wird ein Strom von 1.5 A eingestellt, sodass über jedem Regler eine Spannung von etwa 1.9 V abfällt.

### Regler R2

Abbildung 102 zeigt neben den zuvor erwähnten Messgrößen die Mathematikfunktion zur Berechnung der Eingangsspannung  $V_{in2}$  des Reglers R2, bezogen auf die lokale Chip-Masse bei einer externen Referenz von 0.6 V.  $V_{in}$  weist nach einen Anstieg auf 2.1192 V einen kleinen weiteren Peak auf, bevor sich die Eingangsspannung nachfolgend linear absenkt und auf den Ursprungswert einpendelt.

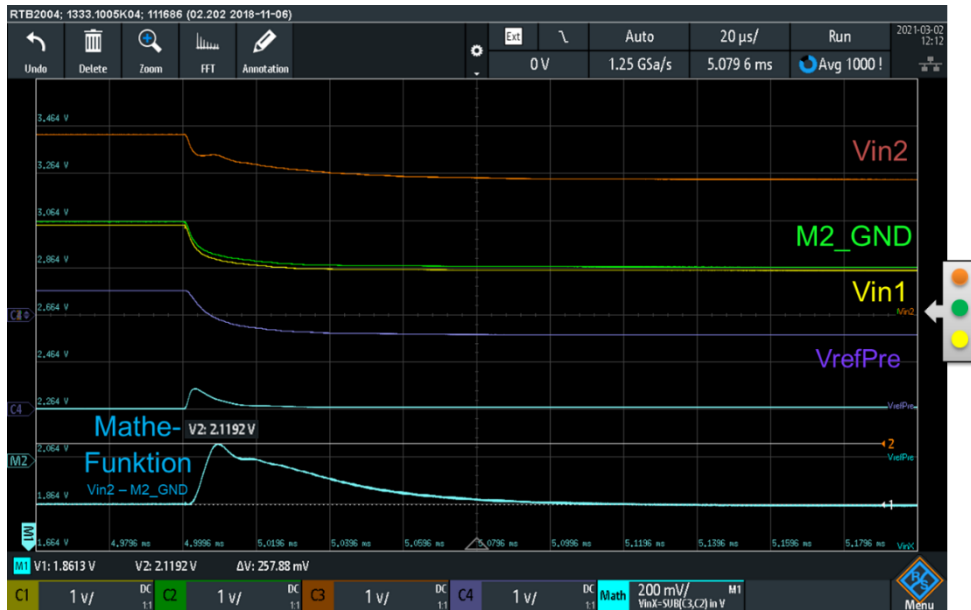


Abbildung 102: Regler R2 – Sourcemeter 1.5 A, externe Referenz 0.6 V,  $V_{wave} = 5 V$ , ohne PrePower

In Abbildung 103 wird die Mathematikfunktion zur Berechnung der Eingangsspannung  $V_{in2}$  des Reglers R2, bezogen auf die lokale Chip-Masse bei einer externen Referenz von 0.7 V gezeigt. Hierbei ist kein auffälliger Verlauf zu erkennen.

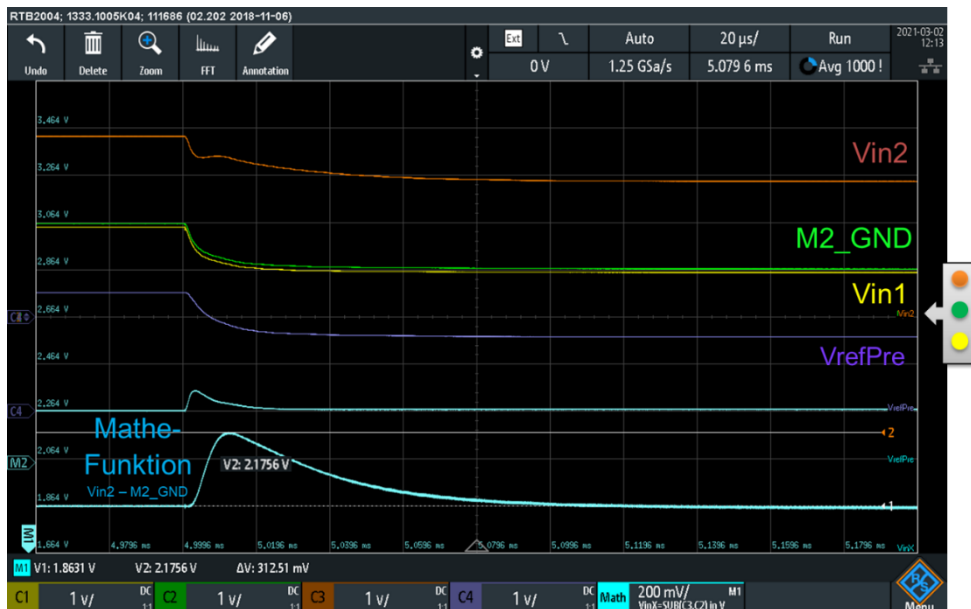


Abbildung 103: Regler R2 – Sourcemeter 1.5 A, externe Referenz 0.7 V,  $V_{wave} = 5 V$ , ohne PrePower

2-er Kette, ohne PrePower, Regler mit externer oder interner REF

Überprüft wird die Eingangsspannung  $V_{in1}$  des Reglers R1 sowie die Eingangsspannung  $V_{in2}$  und das Massenpotenzial  $M_{2\_GND}$  des Reglers R2. Weitere aufgenommene Messgrößen sind  $V_{refPre}$  und eine Mathematikfunktion zur Berechnung der Eingangsspannung  $V_{in2}$  auf dem Regler R2, bezogen auf die lokale Chip-Masse. Hierbei wird der überprüfte Regler mit der internen oder einer externen Referenz von 0.7 V betrieben. Die Waveformgeneratorspannung beträgt für alle Messungen 5 V. Das Sourcemeter wird auf einen Strom von 1.5 A eingestellt.

|                     | 2er-Kette mit PrePower             |               |                 | 2er-Kette ohne PrePower            |               |
|---------------------|------------------------------------|---------------|-----------------|------------------------------------|---------------|
|                     | Modul 2 angesteuert, intern/extern |               |                 | Modul 2 angesteuert, intern/extern |               |
| Vwave               | 5 V                                |               |                 | 5 V                                |               |
| Referenz            | intern                             | 0.6 V         | 0.7 V           | 0.6 V                              | 0.7 V         |
| <b>M2, Vin Peak</b> | <b>2.1726 V</b>                    | <b>2.1704</b> | <b>2.2226 V</b> | <b>2.1192</b>                      | <b>2.1756</b> |
| <b>M3, Vin Peak</b> | <b>X</b>                           | <b>X</b>      | <b>X</b>        | <b>X</b>                           | <b>X</b>      |
| <b>M4, Vin Peak</b> | <b>X</b>                           | <b>X</b>      | <b>X</b>        | <b>X</b>                           | <b>X</b>      |

Vin Peak = Spitzenwert der Mathefunktion (Vin - GND)

Abbildung 104: Messergebnisse der 2er-Kette ohne PrePower Platine, ein Regler angesteuert

Die Messergebnisse zeigen, dass die Peaks der Eingangsspannungen bei der Messung, bei dem der Regler R1 durch Überlast zum Einbruch gebracht wird (ohne PrePower Platine), kleiner ausfallen als bei der Variante, bei dem durch die PrePower Platine der Regler R1 überbrückt wird (siehe Abbildung 104). Dies könnte an der kürzeren Gesamtlänge der Leitungen und dem einhergehenden Spannungsabfall liegen. Beim Anschluss der PrePower-Platine werden zusätzliche Leitungen benötigt. Des Weiteren muss beachtet werden, dass die Verschaltungsvarianten 1 und 2 zwei komplett unterschiedliche Methoden sind, welche auch unterschiedliche Zeitkonstanten besitzen. Daher werden lediglich die Spitzenwerte betrachtet.

## 4.7 - Auswertung der Messergebnisse

Dieses Kapitel gibt einen Überblick darüber, wie die Messergebnisse ausgewertet wurden.

### Darstellung der statischen Messung mit Hilfe von Excel

Abbildung 105 zeigt die statische Messung einer Line Regulation aus dem Kapitel 4.2.1. Die Messergebnisse werden von der Messsteuerungssoftware in einer Textdatei abgelegt und werden anschließend manuell in eine Excel-Tabelle eingepflegt.

Die eingepflegten Ergebnisse werden anschließend grafisch ausgewertet und können analysiert werden.

Des Weiteren werden Berechnungen für die Line Regulation, des Offsets und des Slopes durchgeführt, siehe Tabelle 10.

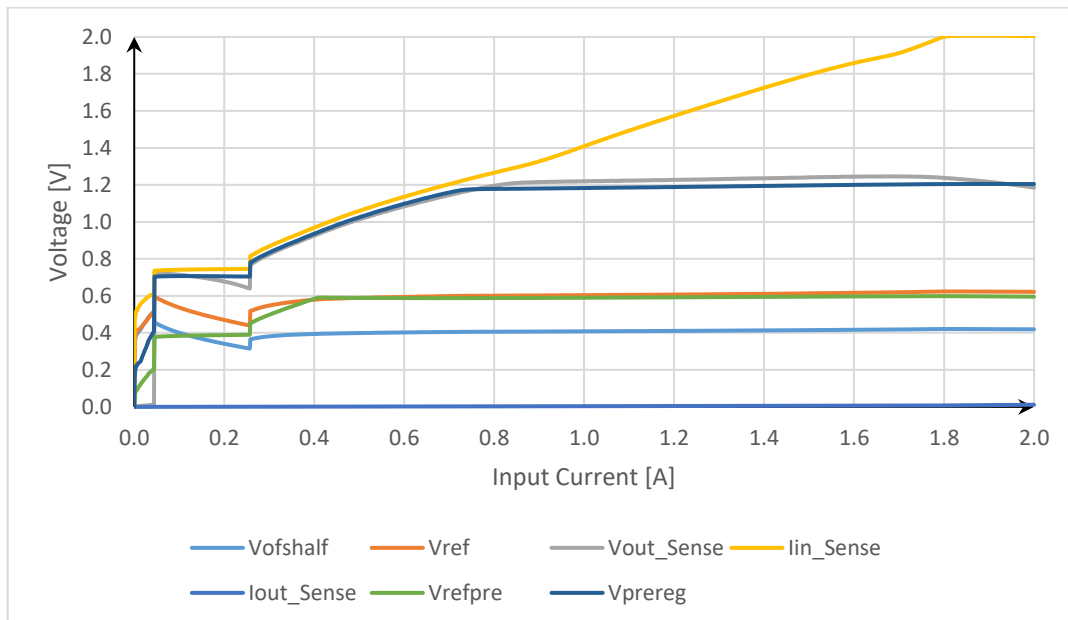


Abbildung 105: Line Regulation im SLDO - HPM Modus für  $I_{in} = 0$  A bis 2 A

Tabelle 10: Messergebnisse Line Regulation im SLDO - HPM Modus für  $I_{in} = 0$  A bis 2 A

|   | $\Delta$    | min         | max         | $\emptyset$ | Anstieg/Abfall<br>(1.8A - 1A) | line Regulation |
|---|-------------|-------------|-------------|-------------|-------------------------------|-----------------|
| $\Delta V_{ofs}$                                  | 12.7040E-3  | 408.2750E-3 | 420.9790E-3 | 413.7612E-3 | 12.2400E-3                    | 21.8E-3         |
| $\Delta V_{ref}$                                  | 20.2540E-3  | 603.5300E-3 | 623.7840E-3 | 611.9583E-3 | 19.5420E-3                    | 0.0218 V        |
| $\Delta V_{out\_Sense}$                           | 27.1300E-3  | 1.2191E+0   | 1.2462E+0   | 1.2347E+0   | 16.9600E-3                    | Slope           |
| $\Delta I_{in\_Sense}$                            | 590.8100E-3 | 1.4094E+0   | 2.0002E+0   | 1.7142E+0   | 591.1800E-3                   | 0.7717 $\Omega$ |
| $\Delta I_{out\_Sense}$                           | 4.0900E-3   | 3.7536E-3   | 7.8436E-3   | 5.6960E-3   | 4.1022E-3                     | Offset          |
| $\Delta V_{refpre}$                               | 8.6930E-3   | 589.8630E-3 | 598.5560E-3 | 594.4600E-3 | 8.6380E-3                     | 0.6409 V        |
| $\Delta V_{prereg}$                               | 21.5700E-3  | 1.1829E+0   | 1.2044E+0   | 1.1941E+0   | 21.3900E-3                    |                 |
| gewählter Wertebereich für $I_{in}$ : 1A bis 1.8A |             |             |             |             |                               |                 |

### Darstellung der transienten Messung mit Hilfe eines Oszilloskops

In Abbildung 106 ist die transiente Messung aus dem Kapitel 4.3.1 abgebildet. Hierbei werden die einzelnen Oszilloskop-Eingänge anhand unterschiedlicher Farben dargestellt. Die Kanäle können manuell umbenannt werden. Für die Darstellung der transienten Ergebnisse ist eine geeignete Auflösung am Oszilloskop notwendig. Nachfolgend können die Messergebnisse grafisch abgelesen und gespeichert werden.

Abbildung 106 zeigt beispielsweise den gelben Eingangskanal  $V_{out}$ , der die Auflösung 100  $\mu\text{s}$  / Div und 20 mV / Div besitzt. An jedem Kanal kann eine eigene Einstellung eingestellt werden.

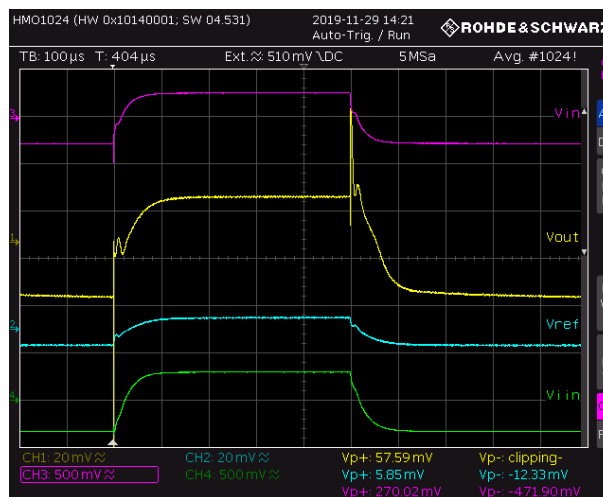


Abbildung 106: Transiente Eingangswechsel im SLDO - HPM Modus für  $I_{in} = 1 \text{ A}$  bis  $1,6 \text{ A}$

### Berechnung von Offset und Steigung zur Überprüfung der Funktionalität

Analog zur Auswertung der Line- oder zur Load Regulation wird die Berechnung von Offset und die Berechnung der Steigung zur Überprüfung der Funktionalität durchgeführt. Durch Variation von  $R_{ext}$  ändert sich bei gegebener Eingangsspannung der Referenzstrom in diesem Pfad, was sich auch auf die Steigung der Eingangsspannung auswirkt. Des Weiteren werden verschieden Einflüsse auf die Offsetspannung überprüft. Dementsprechend wird wie im Kapitel 4.8 eine grafische Auswertung durchgeführt sowie eine Tabelle mit Eckdaten ausgegeben.

Überprüft wird, ob sich die Eingangsspannung nach Eintritt in die Sättigung als lineare Funktion in Form von

$$y = R_{slope} \cdot x + V_{ofs} \quad (4.7\_1)$$

annähert. Der Offset wird aus dem y-Achsenabschnitt bestimmt.

## 4.8 - Messergebnisse aktuelle Testplatine (Rev. 0.3)

In diesem Kapitel werden die Messergebnisse der Revision 0.3 abgebildet, welche die Einbindung der digitalen Potentiometer beinhaltet. Die Erläuterung der Potis ist im Kapitel 3.4 verschriftlicht.

### Line Regulation LDO

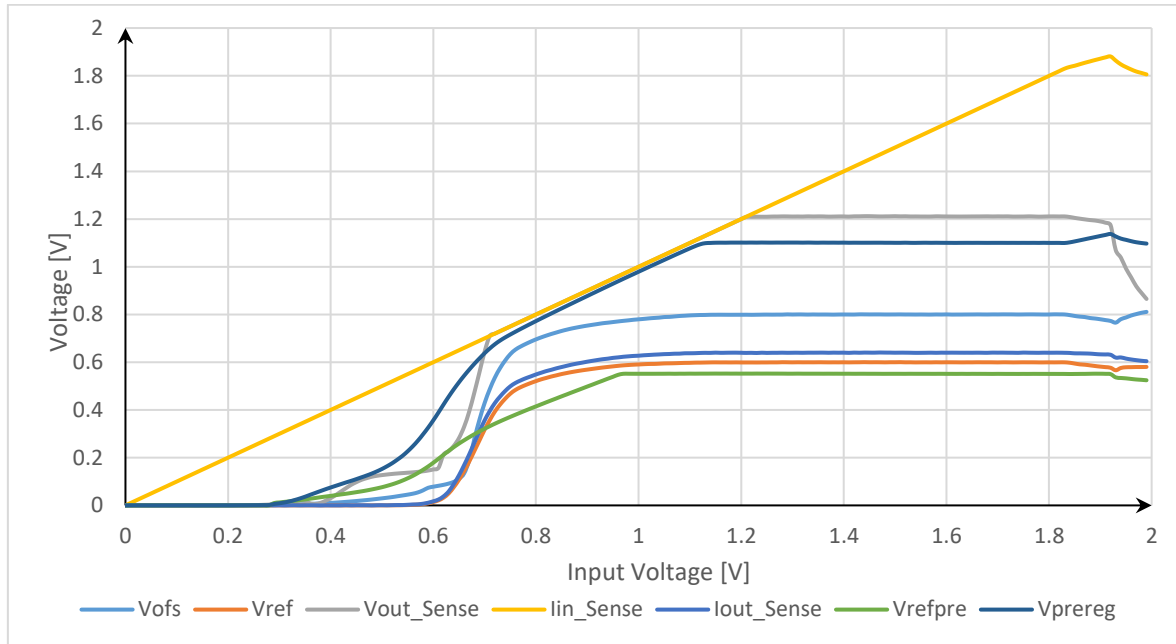


Abbildung 107: Line Regulation im LDO-Modus, Rev.0.3, ohne StartUp sowie ohne USCP

Tabelle 11: Messergebnisse der Line Regulation im LDO-Modus, Rev.0.3 Gewählter Wertebereich für  $V_{in}$ : 1.2 V bis 1.8 V, ohne StartUp sowie ohne USCP

|                         | $\emptyset$ | Min      | max      |
|-------------------------|-------------|----------|----------|
| $\Delta V_{ofs}$        | 0.80        | 765.8E-3 | 810.8E-3 |
| $\Delta V_{ref}$        | 0.60        | 566.3E-3 | 600.2E-3 |
| $\Delta V_{out\_Sense}$ | 1.21        | 865.1E-3 | 1.2E+0   |
| $\Delta l_{in\_Sense}$  | 1.50        | 1.2E+0   | 1.9E+0   |
| $\Delta l_{out\_Sense}$ | 0.64        | 604.2E-3 | 640.6E-3 |
| $\Delta V_{refpre}$     | 0.55        | 524.4E-3 | 552.4E-3 |
| $\Delta V_{prereg}$     | 1.10        | 1.1E+0   | 1.1E+0   |

In Abbildung 107 ist die Messung der Line Regulation im LDO-Modus zur Überprüfung der Revision 0.3 abgebildet. Die Tabelle 11 zeigt die Messergebnisse der Rev. 0.3. Aufgrund der starken Eigenerwärmung ab ca. 1.9 V wurde der Wertebereich angepasst. Anhand der Messergebnisse kann gesagt werden, dass die Funktionalität des Reglers gegeben ist.

Line Regulation SLDO

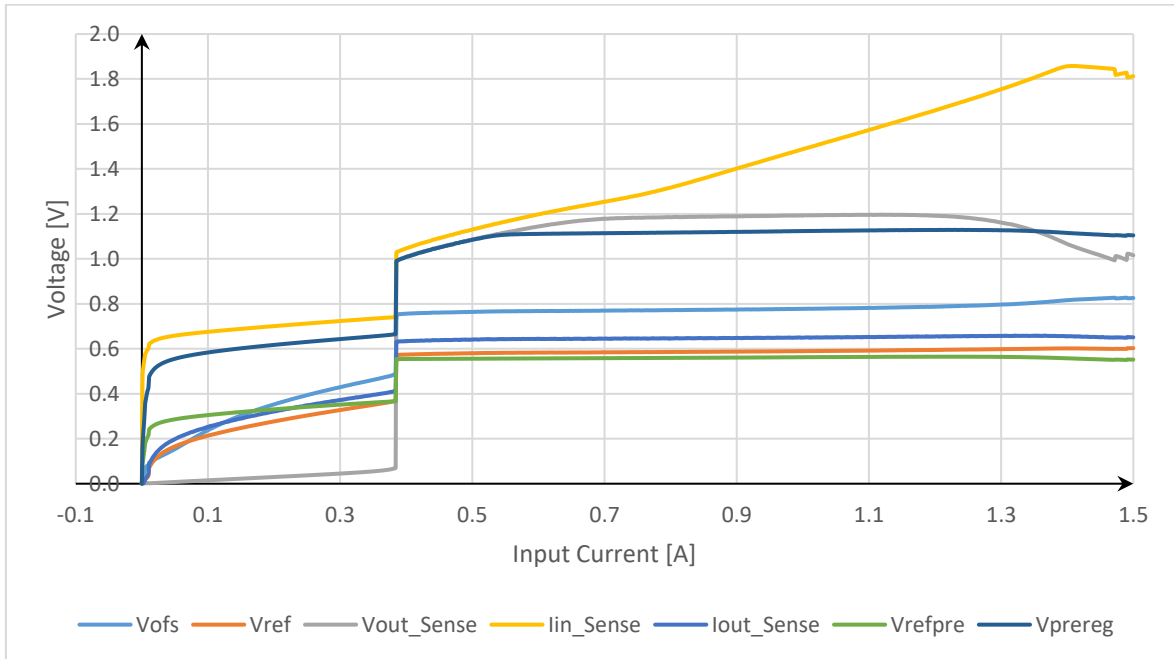


Abbildung 108: Line Regulation im SLDO-Modus, Rev.0.3

Tabelle 12: Messergebnisse der Line Regulation im SLDO-Modus, Rev.0.3

|   | $\Delta$  | min       | max       | $\emptyset$ | Anstieg/Abfall (1.2 A - 0.7 A) |
|---|-----------|-----------|-----------|-------------|--------------------------------|
| $\Delta V_{ofs}$                                      | 17.41E-3  | 769.55E-3 | 786.96E-3 | 776.76E-3   | 17.24E-3                       |
| $\Delta V_{ref}$                                      | 10.63E-3  | 584.19E-3 | 594.81E-3 | 588.79E-3   | 10.27E-3                       |
| $\Delta V_{out\_Sense}$                               | 18.14E-3  | 1.178E+0  | 1.196E+0  | 1.191E+0    | 13.18E-3                       |
| $\Delta I_{in\_Sense}$                                | 405.98E-3 | 1.25E+0   | 1.66E+0   | 1.44E+0     | 405.98E-3                      |
| $\Delta I_{out\_Sense}$                               | 11.09E-3  | 644.18E-3 | 655.28E-3 | 649.20E-3   | 9.03E-3                        |
| $\Delta V_{refpre}$                                   | 6.54E-3   | 558.23E-3 | 564.78E-3 | 561.63E-3   | 6.22E-3                        |
| $\Delta V_{prereg}$                                   | 14.75E-3  | 1.11E+0   | 1.128E+0  | 1.121E+0    | 14.49E-3                       |
| Gewählter Wertebereich für $I_{in}$ : 0.7 A bis 1.2 A |           |           |           |             |                                |

|                 |                 |
|-----------------|-----------------|
| line Regulation | 26.4E-3         |
|                 | - 0.026 V       |
| Slope           | 0.8369 $\Omega$ |
| Offset          | 0.6515 V        |

Abbildung 108 zeigt die Messung der Line Regulation im SLDO-Modus zur Überprüfung der dritten Revision und die Tabelle 12 zeigt die dazugehörigen Messergebnisse. Aufgrund der starken Eigenerwärmung ab ca. 1.2 A wurde der Wertebereich angepasst. Die Messergebnisse liegen näher an den geforderten Werten als der im vorherigen Kapitel überprüfte LDO-Modus.

Abschließend kann gesagt werden, dass die Funktionalität des Reglers gegeben ist. Durch weitere Messungen kann der Regler charakterisiert werden. Hierbei ist darauf zu achten, dass durch den fehlenden Kühlkörper eine starke Eigenerwärmung vorhanden ist.

Sweep des externen Widerstands  $R_{ext}$ 

Der Referenzstrom für den Spannungsregler wird über den Widerstand  $R_3$  eingestellt. Auf der Testplatine kann zwischen einem internen Widerstand  $R_{int}$  und einem externen Widerstand  $R_{ext}$  gewählt werden. Der interne Widerstand besitzt einen festen Wert von  $600 \Omega$ , während der externe Widerstand frei konfigurierbar ist. Durch Variation von  $R_{ext}$  ändert sich bei gegebener Eingangsspannung der Referenzstrom in diesem Pfad, was sich auch auf die Steigung der Eingangsspannung auswirkt. Der Eingangsstrom des Reglers wird durch die Verwendung der Line Regulation im SLDO-Modus gesweept. Die Eingangsspannung wird im eingeschwungenem Zustand als lineare Funktion betrachtet und über die Formel 4.8\_1 angenähert (x-Achsenabschnitt). Der Offset wird aus dem y-Achsenabschnitt bestimmt.

$$y = R_{slope} \cdot x + V_{ofs}$$

(4.8\_1)

Die ermittelten Kennlinien der Eingangsspannungen finden sich in Abbildung 109. Hierdurch wird ersichtlich, dass von  $V_{in}$  mit zunehmenden Wert für  $R_{ext}$  ebenfalls größer wird. Für hohe Versorgungsströme zeigt sich erneut ein starker Temperatureffekt auf Grund des fehlenden Kühlkörpers.

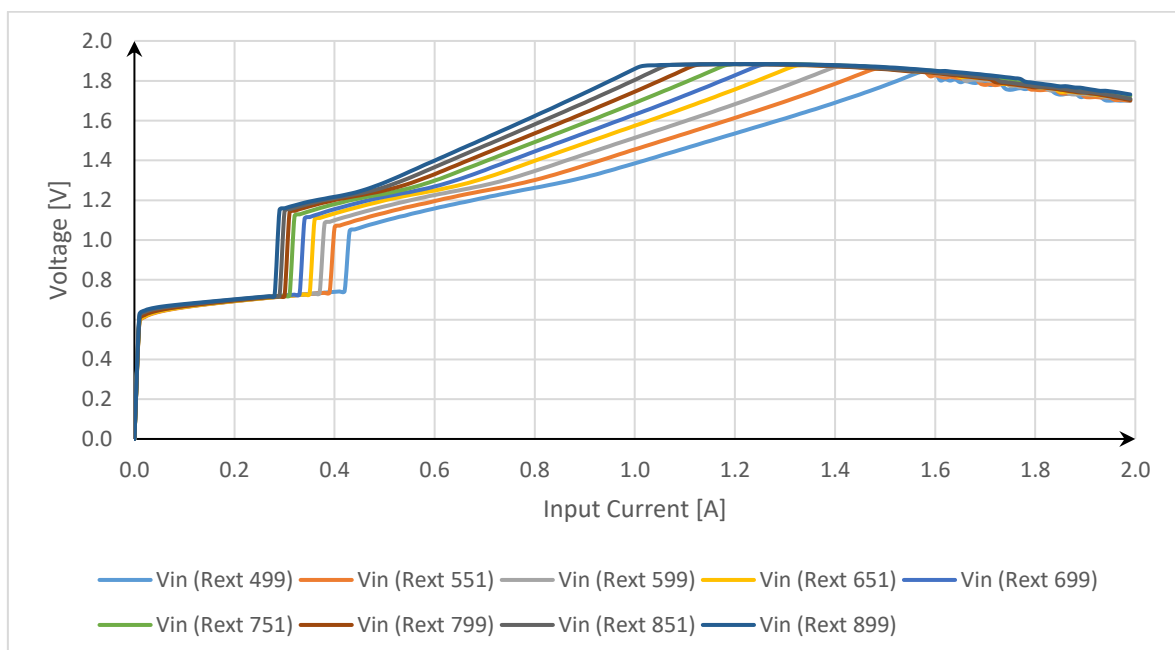


Abbildung 109: Eingangsspannung der Revision 0.3 für verschiedene Widerstandswerte  $R_{ext}$

Die exakten Werte für den Slope und den Offset finden sich in der Tabelle 13. Die ermittelten Werte werden mit den Ergebnissen aus der Arbeit von Herrn Zorn [ZOR] verglichen. Die Abweichungen liegen für die Steigung maximale bei  $0,08 \Omega$  und  $0,1363 \text{ V}$  für den Offset. Die Messungen liegen demnach nah an der vorherigen Revision und schlussfolgernd kann festgehalten werden, dass die Einbindung und Umsetzung des Potentiometers für  $R_{ext}$  wie gewünscht funktioniert. Des Weiteren wird ersichtlich, dass die Abweichung des Offsets mit zunehmenden Wert  $R_{ext}$  abnimmt.



Tabelle 13: Slope- und Offset-Werte bei verschiedenen Widerstandswerten  $R_{ext}$

| $R_{ext}$ [ $\Omega$ ] | Slope [ $\Omega$ ] | Offset [V] | Slope [ $\Omega$ ]<br>Differenz<br>[ZOR] | Offset [V]<br>Differenz<br>[ZOR] |
|------------------------|--------------------|------------|--|----------------------------------|
| 500                    | 0.624              | 0.7745     | -0.08                                    | 0.1363                           |
| 550                    | 0.6933             | 0.7701     | -0.0586                                  | 0.1111                           |
| 600                    | 0.733              | 0.7811     | -0.0682                                  | 0.1051                           |
| 650                    | 0.8045             | 0.7702     | -0.0498                                  | 0.0836                           |
| 700                    | 0.8792             | 0.7525     | -0.0244                                  | 0.0547                           |
| 750                    | 0.9068             | 0.7749     | -0.0462                                  | 0.0672                           |
| 800                    | 0.9782             | 0.7613     | -0.0241                                  | 0.0464                           |
| 850                    | 1.0345             | 0.7564     | -0.0247                                  | 0.04                             |
| 900                    | 1.1289             | 0.7235     | 0.0198                                   | 0.0023                           |

### Sweep der Offsetspannung $V_{ofsH}$

In diesem Kapitel soll der Einfluss verschiedener Werte der Offsetspannung  $V_{ofs}$  auf die Eingangsspannung untersucht werden. Der Offset wird aus dem y-Achsenabschnitt der Formel 4.8\_1 errechnet.

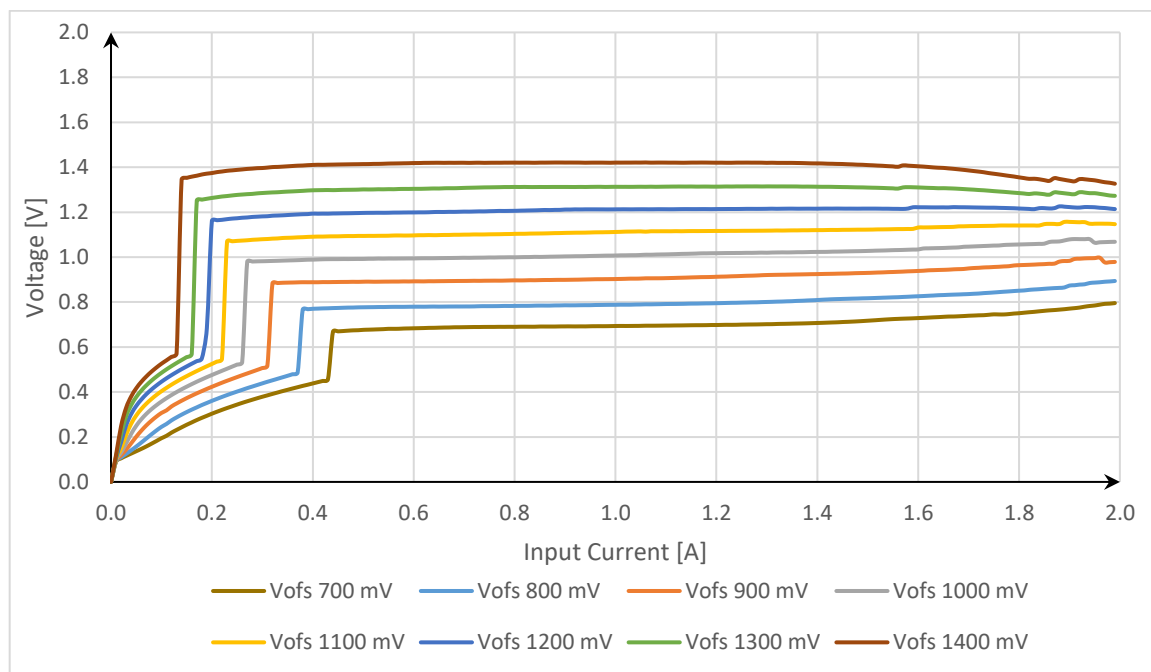


Abbildung 110: Übersicht der eingestellten Offsetspannungen  $V_{ofs}$

In Abbildung 110 ist ersichtlich, dass die Offsetspannung  $V_{ofs}$  im Bereich von 700 mV bis 1.4 V variiert. Ab einem Eingangsstrom  $I_{in}$  von 1.5 A driften die Kennlinien aufgrund der Eigenerwärmung auseinander.

Abbildung 111 zeigt die Eingangsspannungen für verschiedene Offsetspannungen  $V_{ofs}$ . Es ist erkennbar, dass die Kennlinien relativ parallel verlaufen. Bei steigender Versorgungsspannung sind Schwankungen aufgrund von Eigenerwärmung zu erkennen. Mit steigendem Widerstand von  $V_{ofs}$  verringert sich auch der Stromwert bei der die Eingangsspannung zu steigen beginnt.

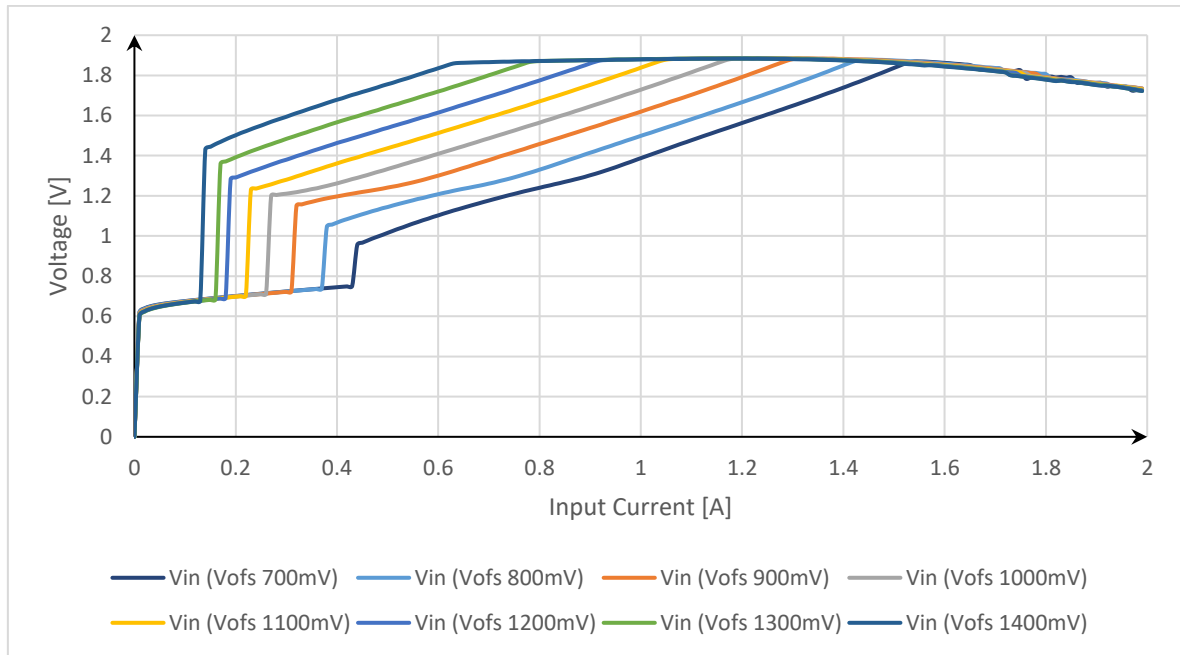


Abbildung 111: Eingangsspannung der Revision 0.3 für verschiedene Offsetspannungen  $V_{ofs}$

Die Tabelle 14 zeigt die exakten Messergebnisse für den Slope und den Offset sowie die Differenz zur [ZOR] Messung. Die maximalen Differenzen sind 0.1  $\Omega$  für den Slope und 0.133 V für den Offset. Die Differenz beim Slope verläuft konstant und die Differenz des Offset wird kleiner, je größer der Offset gewählt wird.

Tabelle 14: Messergebnisse für den Slope und den Offset sowie der Differenz

| Vofs [V] | Slope [ $\Omega$ ] | Offset [V] | Slope [ $\Omega$ ]<br>Differenz<br>[ZOR] | Offset [V]<br>Differenz<br>[ZOR] |
|----------|--------------------|------------|--|----------------------------------|
| 700      | 0.9309             | 0.573      | 0.1                                      | 0.0493                           |
| 800      | 0.7467             | 0.625      | -0.0542                                  | -0.0457                          |
| 900      | 0.7549             | 0.6745     | -0.0242                                  | -0.133                           |
| 1000     | 0.7898             | 0.9371     | 0.0291                                   | 0.0085                           |
| 1100     | 0.7901             | 1.0406     | 0.0468                                   | -0.004                           |
| 1200     | 0.7869             | 1.1452     | 0.045                                    | -0.0026                          |
| 1300     | 0.7855             | 1.25       | 0.0431                                   | 0.003                            |
| 1400     | 0.8192             | 1.3469     | 0.0715                                   | 0.0022                           |

## 5. Fazit

Trotz Verwendung eines Lüfters, der positiven Einfluss auf die Messungen hatte, brach bei Messungen bei Raumtemperatur mit Versorgungsströmen im Bereich von knapp 2 A bei einigen Testchips die Funktionalität zusammen. Insbesondere die Revision 0.3 hat aufgrund des fehlenden Kühlkörpers starke Einbrüche aufgezeigt. Auch bei Reglern, bei denen sich kein kompletter Zusammenbruch der Reglerfunktion bei hohen Strömen eingestellt hat, ist ein Einfluss der Temperatur auf die Messgrößen zu beobachten. Daher wurde der Wertebereich des Eingangsstromes so gewählt, dass Einbrüche möglichst ausgeschlossen werden konnten.

Die Messungen mit seriell verschalteten Modulen lieferten optimale Ergebnisse und liegen innerhalb einer geringen Abweichung zur Spezifikation, die durch eventuelle Eigen erwärmung oder durch Bauteilvarianzen erzeugt werden. Für eine Parallelverschaltung von zwei Reglern und einem Versorgungsstrom von 2 A ergeben sich die Differenzen im Eingangsstrom von 16 mA und 29 mA. Für die Differenzen aus der Messung mit drei parallelen Reglern ergibt sich ein Mittelwert von 22 mA. Die Messung mit vier Reglern wird für die Versorgungsströme von 4.2 A und 5.6 A durchgeführt. Hierbei ergibt sich ein Mittelwert von 12.7 mA für die Messung bei 4.2 A, wobei die Werte bei 4 A aufgenommen wurden, da ein Temperatureffekt beobachtet wurde. Für die Messung bei 5.2 A ergibt sich ein Mittelwert von 36.8 mA.

Aus den Messungen des Kapitels 4.6 zur Validierung der OVP im seriellen Betrieb geht hervor, dass die OVP dazu führt, dass Eingangsspannungs-Peaks schneller reduziert werden.

Durch einen Versuchsaufbau der digitalen Potentiometer konnten die offene Frage bezüglich der Write Protection und die Zusammenarbeit der unterschiedlichen Potis frühzeitig geklärt werden, so dass die Konzeptionierung auf der PCB und der spätere Aufbau reibungslos verlief. Anhand des Versuchsaufbaus ergab sich, dass die erste Auswahl der Potentiometer nicht miteinander kompatibel waren. Die Einbindung der Potis bzw. der Klassen in die vorhandene Messsteuerungssoftware war komplex, konnte jedoch durch die Wiederverwendung der vorangegangenen Bachelor-Thesis [ÖZK1] ohne große Probleme gut umgesetzt werden.

Die Messungen des automatisierten Messvorgangs lieferten ähnliche Kennwerte wie die Messungen in [ZOR], welche die vorherige Platinen-Revision überprüft hat, so dass die Implementierung der digitalen Potentiometer als erfolgreich angesehen werden kann. Nichtsdestotrotz sollte eine komplette Charakterisierung der dritten bzw. vierten Revision durchgeführt werden. Hierzu können die Arbeiten [BAN1], [BAN2], [BAN3] und diese Master-Thesis als Handbuch hinzugezogen werden.

## 6. Abbildungsverzeichnis

|   |    |
|---|----|
| ABBILDUNG 1: AUFBAU ATLAS-DETEKTORS [CIM1] .....  | 3  |
| ABBILDUNG 2: PIXELDETEKTOR UND PIXELDETEKTOR INKL. (BUMP-BOND-KONTAKT) [KAR5].....  | 4  |
| ABBILDUNG 3: HYBRIDER PIXELDETEKTOR [KAR5] .....  | 5  |
| ABBILDUNG 4: PIXEL DES DETEKTORS [KAR5] .....   | 5  |
| ABBILDUNG 5: VERSORGUNGSSCHEMA MIT PARALLEL VERSCHALTETEN MODULEN [KAR5] .....  | 6  |
| ABBILDUNG 6: VERSORGUNGSSCHEMA MIT SERIELL VERSCHALTETEN MODULEN [KAR5] .....   | 8  |
| ABBILDUNG 7: MODUL MIT VIER PARALLELEN REGLERN [KAR5].....  | 9  |
| ABBILDUNG 8: SCHALTBILD EINES SHUNT-REGLERS [KAR5] .....  | 10 |
| ABBILDUNG 9: STROM/SPANNUNGSSCHARAKTERISTIK FÜR DREI PARALLELE SHUNT-REGLER [KAR5] .....  | 11 |
| ABBILDUNG 10: SHUNT-REGLER MIT EINEM EINGANGSWIDERSTAND PARALLEL ZU EINEM LDO-REGLER<br>[KAR5] .....  | 12 |
| ABBILDUNG 11: STROM/SPANNUNGSSCHARAKTERISTIK FÜR EINEN SHUNT-REGLER MIT EINEM WIDERSTAND<br>AM EINGANG [KAR5] .....   | 12 |
| ABBILDUNG 12: LDO-REGLER [KAR4] .....   | 14 |
| ABBILDUNG 13: SHUNT-REGLER [KAR4].....  | 15 |
| ABBILDUNG 14: STROMSPIEGEL [KAR4].....  | 16 |
| ABBILDUNG 15: VERBESSERUNG DER SPIEGELGENAUIGKEIT [KAR4] .....  | 17 |
| ABBILDUNG 16: REFERENZWERT FÜR DEN INVERTIERENDEN EINGANG DES VERSTÄRKERS A <sub>3</sub> [KAR4] .....   | 18 |
| ABBILDUNG 17: GESAMTES SCHALTBILD DES SHUNT-LDO-REGLERS [KAR4] .....  | 19 |
| ABBILDUNG 18: SHUNT-LDO-ERSATZSCHALTUNG MIT EINEM WIDERSTAND IN REIHE ZU EINER<br>SPANNUNGSQUELLE (LINKS) UND MIT EINER ZENER-DIODE IN REIHE ZU EINEM WIDERSTAND (RECHTS)<br>[KAR4] ..... | 20 |
| ABBILDUNG 19: KURVENVERLAUF SHUNT-LDO V/I CHARAKTERISTIK [KAR4] .....   | 21 |
| ABBILDUNG 20 SPANNUNGSVERLAUF VON ZWEI PARALLEL GESCHALTETEN SHUNT-LDO-REGLERN [KAR4] .....   | 22 |
| ABBILDUNG 21: BANDGAP SCHALTUNG [KAR4].....   | 23 |
| ABBILDUNG 22: SCHALTUNG DER UNDER SHUNT PROTECTION [KAR4].....  | 24 |
| ABBILDUNG 23: SCHALTUNG DER OVERVOLTAGE PROTECTION [KAR4] .....   | 26 |
| ABBILDUNG 24: KURVENVERLAUF ÜBERSPANNUNGSSCHUTZ [KAR4] .....  | 27 |
| ABBILDUNG 25: AUFBAU DER AC-GLEICHRICHTER-SCHALTUNG .....   | 28 |
| ABBILDUNG 26: AUFBAU LEITERPLATTE SLDO (RD53B) TESTCHIPS III, 2019 REVISION 0.3 . ANSICHT VON<br>OBEN, MIT KAPPE .....  | 30 |
| ABBILDUNG 27: ANSICHT DER LEITERPLATTE VON UNTEN (CAD OPTIMIERTE BELEUCHTUNG AUSGEWÄHLT)<br>.....   | 31 |
| ABBILDUNG 28: RD53B 3D-ANSICHT SEITLICH (CAD OPTIMIERTE BELEUCHTUNG AUSGEWÄHLT) .....   | 32 |
| ABBILDUNG 29: RD53B 3D-ANSICHT FRONTAL (CAD OPTIMIERTE BELEUCHTUNG AUSGEWÄHLT) .....  | 32 |
| ABBILDUNG 30: AUFBAU LEITERPLATTE SLDO .....  | 35 |
| ABBILDUNG 31: AUFBAU LEITERPLATTE SLDO BOTTOM-ANSICHT (CAD OPTIMIERTE BELEUCHTUNG<br>AUSGEWÄHLT) .....  | 35 |
| ABBILDUNG 32: 3D-ANSICHT SEITLICH (CAD OPTIMIERTE BELEUCHTUNG AUSGEWÄHLT).....  | 36 |
| ABBILDUNG 33: 3D-ANSICHT FRONTAL (CAD OPTIMIERTE BELEUCHTUNG AUSGEWÄHLT) .....  | 36 |
| ABBILDUNG 34: SCHALTUNGS-AUSSCHNITT DES INSTRUMENTENVERSTÄRKERS .....   | 38 |
| ABBILDUNG 35: SCHALTUNG DES NMOS-TRANSISTORS AUF DEM TESTCHIP .....   | 39 |
| ABBILDUNG 36: SCHALTUNG DES NMOS-TRANSISTORS AUF DER PREPOWER-PLATINE .....   | 40 |
| ABBILDUNG 37: WIPER/SCHRITTWEITE DES DIGITALEN POTENTIOMETERS [POT1].....   | 41 |
| ABBILDUNG 38: BEISPIELRECHNUNG EINES DIGITALPOTENTIOMETERS MIT EINEM WERT VON 10 KΩ [POT1]<br>.....   | 41 |
| ABBILDUNG 39: PIN-KONFIGURATION DES AD5252 [POT1] .....   | 42 |
| ABBILDUNG 40: AUFBAU LEITERPLATTE SLDO (RD53B) TESTCHIPS III, 2019 REVISION 0.3 OBEN-ANSICHT MIT<br>KAPPE.....  | 43 |
| ABBILDUNG 41: VERSCHALTUNGSKONZEPT DER DIGITALEN POTENTIOMETER LAUT HERSTELLER [POT1] ...   | 44 |

|  |    |
|--|----|
| ABBILDUNG 42: WIDERSTANDSWERTE FÜR DIE POTENTIOMETER IM NOMINELLEN FALL .....  | 45 |
| ABBILDUNG 43: VERSCHALTUNGSKONZEPT DER DIGITALEN POTENTIOMETER .....   | 46 |
| ABBILDUNG 44: VERSUCHSAUFBAU ZUR ÜBERPRÜFUNG DER DIGITALEN POTENTIOMETER UND DES FTDI<br>2232H MINI MODULS .....                                     | 46 |
| ABBILDUNG 45: DIGITALPOTI IC1 (RP5, ERSETZTES, ANALOGES POTI) FÜR $R_{EXT}$ .....  | 47 |
| ABBILDUNG 46: DIGITALPOTI IC2 (ERSETZT DIE ANALOGEN POTENTIOMETER RP3 UND RP4) FÜR $V_{OFSHALF}$<br>UND $V_{REF}$ .....                              | 47 |
| ABBILDUNG 47: DIGITALPOTI IC3 (RP2, ERSETZTES, ANALOGES POTI) FÜR $V_{OFSL}$ .....   | 48 |
| ABBILDUNG 48: SCHALTPLAN DER LEITUNGEN SDA, SCL.....   | 48 |
| ABBILDUNG 49: SCHALTPLAN DES ANSCHLUSSES M21 ZUM MESSEN DER VERSORGUNGSSPANNUNG DER<br>DIGITALEN POTIS .....   | 48 |
| ABBILDUNG 50: HERVORGEHOBENE LEITUNG VCC. LINKS: FEHLERHAFTES VERSION REV. 0.3. RECHTS:<br>ÜBERARBEITETE VERSION REV. 0.4.....                       | 49 |
| ABBILDUNG 51: MESSAUFBAU INKL. DER MESSGERÄTE UND DES PCS .....  | 50 |
| ABBILDUNG 52: SOURCEMETER KEITHLEY 2460 [KEI3] .....   | 52 |
| ABBILDUNG 53: SOURCEMETER KEITHLEY 2401 [DLC].....   | 52 |
| ABBILDUNG 54: NETZTEIL ROHDE & SCHWARZ HMP4030 [NEK].....  | 53 |
| ABBILDUNG 55: OSZILLOSKOP ROHDE & SCHWARZ HMO1024 [AZN] .....  | 53 |
| ABBILDUNG 56: KEITHLEY DMM6500 [CPS]                      ABBILDUNG 57: KEITHLEY 2000 [IMX] .....  | 54 |
| ABBILDUNG 58: R&S HMF2525 [RAS4].....  | 54 |
| ABBILDUNG 59: GUI DER SLDO-MESSSTEUERUNGS-SOFTWARE [MAC] .....   | 55 |
| ABBILDUNG 60: ANSICHT DES PROGRAMMS QT-CREATOR .....   | 58 |
| ABBILDUNG 61: ÜBERSICHT DER EINGEBUNDENEN HEADER- UND QUELL-DATEIEN .....  | 59 |
| ABBILDUNG 62: KLASSENDIAGRAMM .....  | 60 |
| ABBILDUNG 63: ERSTELLEN DER DROPDOWN-MENÜS.....  | 61 |
| ABBILDUNG 64: FORMEL ZUR BERECHNUNG DER WIDERSTANDSWERTE AUS DER KLASSE „DIGITALEPOTIS“<br>.....   | 61 |
| ABBILDUNG 65: INDEX WECHSEL DER „COMBOBOX“ $R_{EXT}$ .....   | 62 |
| ABBILDUNG 66: SETZEN ALLER POTI-WERTE GLEICHZEITIG, IN DER TEST-MULTIMETER AUSWAHL.....  | 63 |
| ABBILDUNG 67: ÜBERGABE FUNKTION DER DIGITALEN POTENTIOMETER .....  | 63 |
| ABBILDUNG 68: AUSWAHL DER CHECKBOX „SWEEP $R_{EXT}$ “ ; AUTOMATISIERTER VORGANG .....  | 64 |
| ABBILDUNG 69: STRING LEEREN, UM DOPPELTES ANLEGEN DER DATEIEN ZU VERMEIDEN.....  | 65 |
| ABBILDUNG 70: ÜBERGABE DER EINZELWERTE .....   | 65 |
| ABBILDUNG 71: AKTIVIEREN/DEAKTIVIEREN EINZELNER LABELS, COMBOBOXEN UND TEXTFELDER .....  | 66 |
| ABBILDUNG 72: VERERBUNG DER POTIS-KLASSE IN DIE SLDO65-KLASSE .....  | 67 |
| ABBILDUNG 73: LINE REGULATION IM SLDO - HPM MODUS FÜR $I_{IN} = 0$ A BIS 2 A .....   | 70 |
| ABBILDUNG 74: LOAD REGULATION IM SLDO - HPM MODUS FÜR $I_{IN} = 1,05$ A, $I_{LOAD} = 0$ A BIS 1 A.....   | 71 |
| ABBILDUNG 75: TRANSIENTE EINGANGSWECHSEL IM SLDO - HPM MODUS FÜR $I_{IN} = 1$ A BIS 1,6 A .....  | 73 |
| ABBILDUNG 76: TRANSIENTE EINGANGSWECHSEL IM SLDO - HPM MODUS FÜR $I_{IN} = 1$ A BIS 1,6 A<br>(STEIGENDE FLANKE) .....                                | 73 |
| ABBILDUNG 77: TRANSIENTE EINGANGSWECHSEL IM SLDO - HPM MODUS FÜR $I_{IN} = 1$ A BIS 1,6 A (FALLENDE<br>FLANKE) .....                                 | 73 |
| ABBILDUNG 78: TRANSIENTE LASTWECHSEL IM SLDO - HPM MODUS FÜR $I_{IN} = 1$ A, $I_{LOAD} = 0$ A BIS 500 MA<br>(500 $\mu$ S/DIV) .....                  | 75 |
| ABBILDUNG 79: TRANSIENTE LASTWECHSEL IM SLDO - HPM MODUS FÜR $I_{IN} = 1$ A, $I_{LOAD} = 0$ A BIS 500 MA<br>(STEIGENDE FLANKE) (10 $\mu$ S/DIV)..... | 75 |
| ABBILDUNG 80: TRANSIENTE LASTWECHSEL IM SLDO - HPM MODUS FÜR $I_{IN} = 1$ A, $I_{LOAD} = 0$ A BIS 500 MA<br>(FALLENDE FLANKE) (10 $\mu$ S/DIV).....  | 75 |
| ABBILDUNG 81: EINSCHALTVERHALTEN IM SLDO - HPM MODUS FÜR $I_{IN} = 0$ A BIS 1 A, $T_{RISE} = 1,05$ S (MIT<br>START-UP).....                          | 77 |
| ABBILDUNG 82: EINSCHALTVERHALTEN IM SLDO - HPM MODUS FÜR $I_{IN} = 0$ A BIS 1 A, $T_{RISE} = 1,05$ S (OHNE<br>START-UP).....                         | 77 |
| ABBILDUNG 83: OVP - SLDO - HPM - $I_{IN} = 1$ A - 2 A, $T_{RISE} = 10$ NS .....  | 78 |

|   |     |
|---|-----|
| ABBILDUNG 84: USCP (TRANSIENT) - SLDO - HPM - $I_{IN} = 1\text{ A}$ , $I_{LOAD} = 0\text{ A} - 1\text{ A}$ (ERHÖHUNG DER LAST BIS EINBRUCH).....  | 80  |
| ABBILDUNG 85: AC-GLEICHRICHTER / WECHSEL ZWISCHEN HPM UND LPM - $I_{IN} = 1,4\text{ A}$ , $V_{AC} = 0\text{ V} - 1,2\text{ V}$ , 600 CYCLES ..... | 81  |
| ABBILDUNG 86: SCHEMATIC AUSZUG DES JUMPERS P8 UND DER MOLEX STECKER VERBINDUNGEN PWR_IN & PWR_OUT.....  | 82  |
| ABBILDUNG 87: GESETZTER JUMPER P8 AUF SINGLE SERIELL .....  | 82  |
| ABBILDUNG 88: GESETZTER JUMPER P8 AUF SERIELL .....   | 82  |
| ABBILDUNG 89: PCB RD53B.....  | 83  |
| ABBILDUNG 90: NAHAUFNAHME AN DEN GESETZTEN JUMPER P8 AUF SERIELL .....  | 83  |
| ABBILDUNG 91: VERANSCHAULICHUNG DER SERIELLEN KETTE / VARIANTE 1.....   | 84  |
| ABBILDUNG 92: SERIELLE VERSCHALTUNG ANHAND DER MOLEX STECKER UND JUMPER P8 / VARIANTE 2. ....   | 84  |
| ABBILDUNG 93: MASSEN-POTENTIALUNTERSCHIEDE DER REGLER R1 BIS R6 .....   | 85  |
| ABBILDUNG 94: KONZEPT FÜR DIE PARALLELE VERSCHALTUNG DER REGLER .....   | 86  |
| ABBILDUNG 95: MESSUNG VON DREI PARALLELEN REGLERN ZUR ÜBERPRÜFUNG DER STROMAUFTEILUNG (DO1,DO2,DO3) .....   | 87  |
| ABBILDUNG 96: SCHEMATIC AUSZUG DES JUMPERS P1 ZUR AUSWAHL DER REFERENZ .....  | 88  |
| ABBILDUNG 97: SERIELLE VERSCHALTUNG MIT VORGESCHALTETER PREPOWER PLATINE.....   | 89  |
| ABBILDUNG 98: <b>REGLER R2</b> – SOURCEMETER 1.5 A, INTERNE REFERENZ, $V_{WAVE} = 5\text{ V}$ , PREPOWER .....                                    | 90  |
| ABBILDUNG 99: <b>REGLER R2</b> – SOURCEMETER 1.5 A, EXTERNE REFERENZ 0.6 V, $V_{WAVE} = 5\text{ V}$ , PREPOWER.....                               | 90  |
| ABBILDUNG 100: <b>REGLER R2</b> – SOURCEMETER 1.5 A, EXTERNE REFERENZ 0.7 V, $V_{WAVE} = 5\text{ V}$ , PREPOWER... ..                             | 90  |
| ABBILDUNG 101: SCHALTPLAN ZUR SERIELLEN VERSCHALTUNG .....  | 91  |
| ABBILDUNG 102: <b>REGLER R2</b> – SOURCEMETER 1.5 A, EXTERNE REFERENZ 0.6 V, $V_{WAVE} = 5\text{ V}$ , <b>OHNE</b> PREPOWER.....                  | 92  |
| ABBILDUNG 103: <b>REGLER R2</b> – SOURCEMETER 1.5 A, EXTERNE REFERENZ 0.7 V, $V_{WAVE} = 5\text{ V}$ , <b>OHNE</b> PREPOWER.....                  | 92  |
| ABBILDUNG 104: MESSERGEBNISSE DER 2ER-KETTE OHNE PREPOWER PLATINE, EIN REGLER ANGESTEUERT .....   | 93  |
| ABBILDUNG 105: LINE REGULATION IM SLDO - HPM MODUS FÜR $I_{IN} = 0\text{ A}$ BIS 2 A .....  | 94  |
| ABBILDUNG 106: TRANSIENTE EINGANGSWECHSEL IM SLDO - HPM MODUS FÜR $I_{IN} = 1\text{ A}$ BIS 1,6 A.....  | 95  |
| ABBILDUNG 107: LINE REGULATION IM LDO-MODUS, REV.0.3, OHNE STARTUP SOWIE OHNE USCP .....  | 96  |
| ABBILDUNG 108: LINE REGULATION IM SLDO-MODUS, REV.0.3 .....   | 97  |
| ABBILDUNG 109: EINGANGSSPANNUNG DER REVISION 0.3 FÜR VERSCHIEDENE WIDERSTANDSWERTE $R_{EXT}$ .....  | 98  |
| ABBILDUNG 110: ÜBERSICHT DER EINGESTELLTEN OFFSETSPANNUNGEN $V_{OFS}$ .....   | 99  |
| ABBILDUNG 111: EINGANGSSPANNUNG DER REVISION 0.3 FÜR VERSCHIEDENE OFFSETSPANNUNGEN $V_{OFS}$ .....  | 100 |
| ABBILDUNG 112: PCB MIT AUFGETRAGENER LÖTPASTE .....   | 113 |
| ABBILDUNG 113: PLATINE AUF DER BESTÜCKUNGSMASCHINE.....   | 114 |
| ABBILDUNG 114: ARBEITSPLATZ INKLUSIVE DES PCS, KLIMAKAMMER, ABSAUGUNG UND LÖTSTATION ...  | 115 |
| ABBILDUNG 115: PLATINE MIT DEN BESTÜCKTEN, GEFESTIGTEN SMD BAUTEILEN .....  | 115 |
| ABBILDUNG 116: SCHEMATIC DER TESTPCB-RD53B, TEIL1.....  | 117 |
| ABBILDUNG 117: SCHEMATIC DER TESTPCB-RD53B, TEIL2 .....   | 117 |
| ABBILDUNG 118: ÜBERSICHT DER SCHEMATIC DER PREPOWER-PLATINE.....  | 118 |

## 7. Quellenverzeichnis

|       |  |
|-------|--|
| [FES] | H. Ibach und H. Lüth<br>Festkörperphysik / Einführung in die Grundlagen<br>Berlin, (Springer) 2002   |
| [HBT] | S.M. Sze and K.K. Ng<br>Halbleiterbauteile / Physics of semiconductor devices (3rd ed.)<br>Hoboken (Wiley) 2007.   |
| [HDT] | G. Lutz<br>Halbleiterdetektoren / Semiconductor radiation detectors. Device physics<br>Berlin (Springer) 1999  |
| [AGI] | Agilent (Hrsg.)<br>Agilent 33500 Series Waveform Generator<br><a href="https://electrica.uniandes.edu.co/images/manual_33500_Op_and_Svc_US.pdf">https://electrica.uniandes.edu.co/images/manual_33500_Op_and_Svc_US.pdf</a><br>Online Zugriff am 13.08.2019  |
| [ICY] | Icyer, Ömer F.<br>USB basierte Einbindung von Multimetern in einen Messplatz zur automatisierten Charakterisierung von Spannungsreglern<br><a href="https://www.fh-dortmund.de/de/fb/3/personen/lehr/karagounis/medien/Bachelor_Thesis_Oemer_Icyer.pdf">https://www.fh-dortmund.de/de/fb/3/personen/lehr/karagounis/medien/Bachelor_Thesis_Oemer_Icyer.pdf</a><br>Fachhochschule Dortmund, Bachelorarbeit, 2019  |
| [GSM] | Garcia-Sciveres, Maurice<br>The RD53A Integrated Circuit / Cern, Genf.<br><a href="https://cds.cern.ch/record/2287593/files/RD53A_Manual_V3-5.pdf">https://cds.cern.ch/record/2287593/files/RD53A_Manual_V3-5.pdf</a><br>2019 Forschungsbericht  |
| [WIN] | Winkler, Florian<br>Verification of the Shunt-Low-Dropout voltage regulator for the current based supply of the serially connected pixel detector modules of the ATLAS- and CMS-experiments at the High-Luminosity Large Hadron Collider<br><a href="https://www.fh-dortmund.de/de/fb/3/personen/lehr/karagounis/medien/Masterthesis_Winkler.pdf">https://www.fh-dortmund.de/de/fb/3/personen/lehr/karagounis/medien/Masterthesis_Winkler.pdf</a><br>Fachhochschule Dortmund, Masterarbeit, 2019 |

|        |  |
|--------|--|
| [ZOR]  | <p>Zorn, Jendrik</p> <p>Messtechnische Validierung eines Shunt-Low-Dropout-Spannungsreglers zur strombasierten Versorgung der seriell verschalteten Pixel-Detektormodule des ATLAS- und CMS-Experiments am High-Luminosity Large Hadron Collider</p> <p><a href="https://www.fh-dortmund.de/de/fb/3/personen/lehr/karagounis/medien/Masterthesis_Zorn.pdf">https://www.fh-dortmund.de/de/fb/3/personen/lehr/karagounis/medien/Masterthesis_Zorn.pdf</a></p> <p>Fachhochschule Dortmund, Masterthesis, 2019</p> |
| [MAC]  | <p>Macha, Karl</p> <p>Einrichtung eines automatisierten Messplatzes zur Charakterisierung von Spannungsreglern</p> <p><a href="https://www.fh-dortmund.de/de/fb/3/personen/lehr/karagounis/medien/BachelorthesisMacha.pdf">https://www.fh-dortmund.de/de/fb/3/personen/lehr/karagounis/medien/BachelorthesisMacha.pdf</a></p> <p>Fachhochschule Dortmund, Bachelorarbeit, 2017</p>   |
| [KAM]  | <p>Kampkötter, Jeremias</p> <p>ShuntLDO Verification,</p> <p>Fachhochschule Dortmund, Projektarbeit, 2018</p>  |
| [KAR1] | <p>Karagounis, Michael</p> <p>Analog Integrated CMOS Circuits for the Readout and Powering of Highly Segmented Detectors in Particle Physics Applications</p> <p><a href="https://ub-deposit.fernuni-hagen.de/servlets/MCRFileNodeServlet/mir_derivate_00000025/Diss_Karagounis_CMOS_2010.pdf">https://ub-deposit.fernuni-hagen.de/servlets/MCRFileNodeServlet/mir_derivate_00000025/Diss_Karagounis_CMOS_2010.pdf</a></p> <p>Fernuniversität Hagen, Diss., 2010</p>   |
| [KAR4] | <p>Karagounis, Michael</p> <p>Shunt-LDO Regulator - Internal Review</p> <p>Fachhochschule Dortmund, Präsentation, 2019</p>   |
| [KAR5] | <p>Karagounis, Michael</p> <p>Shunt-LDO Regulator - Einführung</p> <p>Fachhochschule Dortmund, Präsentation, 2016</p>  |
| [SFS]  | <p>Karagounis, Michael / Arutinov, David / Barbero, Marlon / Huegging, Fabian / Krueger, Hans / Wermes, Norbert</p> <p>An Integrated Shunt-LDO Regulator for Serial Powered Systems</p> <p><a href="https://indico.cern.ch/event/72160/attachments/1036621/1477145/Shunt-LDO_Regulator.pdf">https://indico.cern.ch/event/72160/attachments/1036621/1477145/Shunt-LDO_Regulator.pdf</a></p> <p>University of Bonn - Physics Department</p>  |
| [EFI]  | <p>Schnabel, Patrick</p> <p>Elektronik-Fibel 5. Aufl.</p> <p>Schnabel Verlag, 2010</p>   |



|        |   |
|--------|---|
| [ORF1] | <p>Orfanelli, Stella</p> <p>Serial Powering Optimization for CMS and ATLAS Pixel Detectors within RD53 Collaboration for HL-LHC</p> <p>System Level Simulations and Testing / Santa Cruz, California 2017. Paper <a href="http://inspirehep.net/record/1665020/files/CR2017_385.pdf?version=1">http://inspirehep.net/record/1665020/files/CR2017_385.pdf?version=1</a></p>  |
| [ORF2] | <p>Orfanelli, Stella</p> <p>Serial Powering for Pixel Detectors System Level Simulations and Testing / Santa Cruz, California</p>   |
| [KEI1] | <p>Keithley (Hrsg.)</p> <p>Model 2000 6½-Digit Multimeter</p> <p><a href="https://www.instrumex.de/datasheet/2000.pdf">https://www.instrumex.de/datasheet/2000.pdf</a>: Keithley.</p> <p>Online Zugriff am 13.08.2019</p>   |
| [KEI2] | <p>Keithley (Hrsg.)</p> <p>Model 2401 SourceMeter® SMU Instrument</p> <p><a href="http://research.physics.illinois.edu/bezryadin/labprotocol/Keithley2400Manual.pdf">http://research.physics.illinois.edu/bezryadin/labprotocol/Keithley2400Manual.pdf</a></p> <p>Online Zugriff am 13.08.2019</p>  |
| [KEI3] | <p>Keithley (Hrsg.)</p> <p>Model 2460 SourceMeter®SMU Instrument</p> <p><a href="https://docs-emea.rs-online.com/webdocs/1347/0900766b81347571.pdf">https://docs-emea.rs-online.com/webdocs/1347/0900766b81347571.pdf</a></p> <p>Online Zugriff am 13.08.2019</p>   |
| [KEI4] | <p>Keithley (Hrsg.)</p> <p>Model DMM6500 6½-Digit Bench/System Digital Multimeter</p> <p><a href="https://docs-emea.rs-online.com/webdocs/1634/0900766b81634895.pdf">https://docs-emea.rs-online.com/webdocs/1634/0900766b81634895.pdf</a></p> <p>Online Zugriff am 13.08.2019</p>  |
| [RAS1] | <p>Rohde &amp; Schwarz (Hrsg.)</p> <p>R&amp;S®HMF25xx Arbitrary Function Generator</p> <p><a href="https://scdn.rohde-schwarz.com/ur/pws/dl_downloads/dl_common_library/dl_brochures_and_datasheets/pdf_1/service_support_30/170728_HMF_Fact_Sheet.pdf">https://scdn.rohde-schwarz.com/ur/pws/dl_downloads/dl_common_library/dl_brochures_and_datasheets/pdf_1/service_support_30/170728_HMF_Fact_Sheet.pdf</a></p> <p>Online Zugriff am 13.08.2019</p> |
| [RAS2] | <p>Rohde &amp; Schwarz (Hrsg.)</p> <p>R&amp;S®HMO1024 4 Kanal Digital-Oszilloskop</p> <p><a href="https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuals/gb_1/h/hmo72x_202x/HMOcompact_DigitalOscilloscope_UserManual_en_03.pdf">https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuals/gb_1/h/hmo72x_202x/HMOcompact_DigitalOscilloscope_UserManual_en_03.pdf</a></p> <p>Online Zugriff am 13.08.2019</p>    |

|        |   |
|--------|---|
| [RAS3] | <p>Rohde &amp; Schwarz (Hrsg.)</p> <p>R&amp;S®HMP4030 Series Power Supply</p> <p><a href="https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuels/gb_1/h/hmp_serie/HMPSeries_UserManual_en_02.pdf">https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuels/gb_1/h/hmp_serie/HMPSeries_UserManual_en_02.pdf</a></p> <p>Online Zugriff am 13.08.2019</p>                                      |
| [RAS4] | <p>HMF2525 - Arbitrary Function Generator - Benutzerhandbuch</p> <p><a href="https://www.datatec.de/media/pdf/e6/0b/ef/db628693-446f-447b-9f43-c12d3dedb77c_hmf2550_1_d8hHPqhQzZ4SC.pdf">https://www.datatec.de/media/pdf/e6/0b/ef/db628693-446f-447b-9f43-c12d3dedb77c_hmf2550_1_d8hHPqhQzZ4SC.pdf</a></p> <p>Online Zugriff am 03.03.2020</p>   |
| [RAS5] | <p>Power supply, HMP Series, Benutzerhandbuch, User Manual</p> <p><a href="https://cdn.rohde-schwarz.com/hameg-archive/HAMEG_MAN_DE_EN_HMPSeries.pdf">https://cdn.rohde-schwarz.com/hameg-archive/HAMEG_MAN_DE_EN_HMPSeries.pdf</a></p> <p>(Online Zugriff am 05.03.2021)</p>   |
| [RAS7] | <p>100MHz 2, 4 Kanal Digital-Oszilloskop HMO1022 [HMO1024]</p> <p><a href="http://www.farnell.com/datasheets/1808893.pdf">http://www.farnell.com/datasheets/1808893.pdf</a></p> <p>(Online Zugriff am 07.05.2020)</p>   |
| [RAS8] | <p>HMO Kompakt Serie Digital Oszilloskop Benutzerhandbuch</p> <p><a href="https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuels/gb_1/h/hmo72x_202x/HMOcompact_DigitalOscilloscope_UserManual_de_03.pdf">https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuels/gb_1/h/hmo72x_202x/HMOcompact_DigitalOscilloscope_UserManual_de_03.pdf</a></p> <p>(Online Zugriff am 07.05.2020)</p>      |
| [MUE]  | <p>Tobias Stockmanns</p> <p>Multi-Chip-Modul-Entwicklung für den ATLAS-Pixeldetektor / Analyse der Front-End-Chip-Elektronik in strahlenharter 0,25-<math>\mu</math>m-Technologie sowie Entwicklung und Realisierung eines <i>Serial-Powering</i>-Konzeptes</p> <p><a href="http://hss.ulb.uni-bonn.de/2004/0418/0418.pdf">http://hss.ulb.uni-bonn.de/2004/0418/0418.pdf</a></p> <p>Universität Bonn Diss. veröffentlicht: 2004</p> |
| [FUZ]  | <p>Duc Bao Ta</p> <p>Entwicklung und Demonstration des "Serial Powering" an ATLAS Pixelmodulen und Moduleitern (Funktionalitäts- und Zuverlässigkeitstests)</p> <p><a href="https://www.hep1.physik.uni-bonn.de/results/data/internal/Ta.pdf">https://www.hep1.physik.uni-bonn.de/results/data/internal/Ta.pdf</a></p> <p>Universität Bonn Diss. veröffentlicht: 2005</p>   |
| [PIL1] | <p>Andreas Pille</p> <p>Analog CMOS Design</p> <p>Teil 1: Grundlagen der CMOS-Technologie</p> <p>Fachhochschule Dortmund, Projektarbeit 1, veröffentlicht 2019</p>  |

|        |  |
|--------|--|
| [BAN1] | <p>Maurice Bankowsky</p> <p>Einführung in die Funktionalität des Shunt-LDO-Reglers für die serielle Versorgung der Pixelmodule der Atlas und CMS Detektoren am Cern)</p> <p><a href="https://1drv.ms/w/s!AuQqnut2S7FfxQ_2hN8r3DSKH5F1?e=VPTcMs">https://1drv.ms/w/s!AuQqnut2S7FfxQ_2hN8r3DSKH5F1?e=VPTcMs</a></p> <p>Fachhochschule Dortmund, Projektarbeit 1, veröffentlicht 2019</p>   |
| [BAN2] | <p>Maurice Bankowsky</p> <p>Messtechnische Validierung des Shunt-Low-Dropout-Spannungsreglers zur strombasierten Versorgung der seriell verschalteten Pixel-Detektormodule des ATLAS- und CMS-Experiments am High-Luminosity Large Hadron Collider (Testchip III, 2019 Revision 0.1)</p> <p><a href="https://1drv.ms/w/s!AuQqnut2S7FfyGTmoi7HIUJCGyH4?e=Nbdlrl">https://1drv.ms/w/s!AuQqnut2S7FfyGTmoi7HIUJCGyH4?e=Nbdlrl</a></p> <p>Fachhochschule Dortmund, Projektarbeit 2, veröffentlicht 2020</p> |
| [BAN3] | <p>Maurice Bankowsky</p> <p>Serielle und parallele Verschaltung der SLDO-Regler für die Pixeldetektoren des ATLAS- und CMS-Experiments am High-Luminosity Large Hadron Collider</p> <p>Fachhochschule Dortmund, Masterstudienarbeit, veröffentlicht 2021</p>   |
| [ÖZK1] | <p>Nurullah Özkan</p> <p>Entwicklung einer API für die I2C Schnittstelle des FTDI 2232H Mini Moduls</p> <p>Fachhochschule Dortmund, Master-Projektarbeit 1, veröffentlicht 2019</p>  |
| [ÖZK2] | <p>Nurullah Özkan</p> <p>Konfiguration und Inbetriebnahme des FTDI 2232H Mini Moduls als I2C Schnittstelle</p> <p>Fachhochschule Dortmund, Bachelorarbeit, veröffentlicht 2019</p>   |
| [POT1] | <p>Dual 64-/256-Position, I2C Nonvolatile Memory Digital Potentiometers, Data Sheet AD5251/AD5252</p> <p><a href="https://www.analog.com/media/en/technical-documentation/data-sheets/AD5251_5252.pdf">https://www.analog.com/media/en/technical-documentation/data-sheets/AD5251_5252.pdf</a></p> <p>(Online Zugriff am 20.04.2021)</p>   |
| [POT2] | <p>Dual 64-/256-Position, I2C Nonvolatile Memory Digital Potentiometers, Data Sheet AD5251/AD5252</p> <p><a href="https://www.analog.com/media/en/technical-documentation/data-sheets/AD5251_5252.pdf">https://www.analog.com/media/en/technical-documentation/data-sheets/AD5251_5252.pdf</a></p> <p>(Online Zugriff am 20.04.2021)</p>   |
| [POT3] | <p>Dual 64-/256-Position, I2C Nonvolatile Memory Digital Potentiometers, Data Sheet AD5241/AD5242</p> <p><a href="https://www.analog.com/media/en/technical-documentation/data-sheets/AD5241_5242.pdf">https://www.analog.com/media/en/technical-documentation/data-sheets/AD5241_5242.pdf</a></p> <p>(Online Zugriff am 20.04.2021)</p>   |

|       |   |
|-------|---|
| [MOU] | Keysight Technologies 33500B Series Waveform Generators<br><a href="https://www.mouser.com/pdfdocs/dt5991-0692EN-2.pdf">https://www.mouser.com/pdfdocs/dt5991-0692EN-2.pdf</a><br>(Online Zugriff am 14.10.2019)  |
| [FTD] | Future Technology Devices International Ltd.<br>FT2232H Mini Module, USB Hi-Speed FT2232H Evaluation Module, Datasheet<br><a href="https://ftdichip.com/wp-content/uploads/2020/07/DS_FT2232H_Mini_Module.pdf">https://ftdichip.com/wp-content/uploads/2020/07/DS_FT2232H_Mini_Module.pdf</a><br>(Online Zugriff am 14.10.2019) |
| [IMV] | Single and Dual-Supply, Rail-to-Rail, Low Cost Instrumentation Amplifier<br><a href="https://www.analog.com/media/en/technical-documentation/data-sheets/AD623.pdf">https://www.analog.com/media/en/technical-documentation/data-sheets/AD623.pdf</a><br>(Online Zugriff am 23.08.2021)   |

### **Webverzeichnis:**

[CER1] <https://atlas.cern/>  
(Online Zugriff am 02.08.2019)

[CER2] <https://atlas.cern/discover/about>  
(Online Zugriff am 02.08.2019)

[CER3] <https://atlas.cern/discover/detector>  
(Online Zugriff am 02.08.2019)

[CER4] <https://home.cern/about/who-we-are/our-mission>  
(Online Zugriff am 02.08.2019)

[CER5] <https://home.cern/science/experiments/cms>  
(Online Zugriff am 02.08.2019)

[CIM1] <https://cds.cern.ch/images/CERN-GE-0803012-01>  
(Online Zugriff am 02.08.2019)

[EKM1] <https://www.elektronik-kompodium.de/sites/slt/0210141.htm>  
(Online Zugriff am 10.10.2019)

[EKM2] <https://www.elektronik-kompodium.de/sites/slt/0204134.htm>  
(Online Zugriff am 10.10.2019)

[ETT] <https://et-tutorials.de/8033/mosfet-aufbau-und-funktionsweise/>  
(Online Zugriff am 11.10.2019)

[MIC] <https://www.mikrocontroller.net/topic/345023>  
(Online Zugriff am 11.10.2019)

[DLC] <https://www.distrelec.de/de/quellmessgeraet-1x20w-keithley-2401/p/11038621?queryFromSuggest=true>  
(Online Zugriff am 14.10.2019)

[CPS] <https://www.calplus.de/keithley-dmm6500.html>  
(Online Zugriff am 14.10.2019)

[IMX] <http://imex.co.uk/KEITHLEY-2000-20/E-65-DIGITAL-MULTIMETER-AT-220V>  
(Online Zugriff am 14.10.2019)

[NEK] <https://www.newark.com/rohde-schwarz/hmp4030-03/bench-power-supply-prog-3ch-10a/dp/51AC6946>  
(Online Zugriff am 14.10.2019)

[AZN] <https://www.amazon.de/Rohde-and-Schwarz-HMO1024-Oszilloskop/dp/B00D02QQIK>  
(Online Zugriff am 14.10.2019)

[RAS4] [https://www.rohde-schwarz.com/de/produkt/hmf-produkt-startseite\\_63493-43456.html](https://www.rohde-schwarz.com/de/produkt/hmf-produkt-startseite_63493-43456.html)  
(Online Zugriff am 14.10.2019)

[TEK1] <https://uk.tek.com/tektronix-and-keithley-digital-multimeter/dmm6500-manual/model-dmm6500-6-1-2-digit-multimeter-3>  
(Online Zugriff am 01.03.2021)

[RAS6] <https://de.farnell.com/en-DE/rohde-schwarz/r-s-hmp4030/labornetzteil-prog-3-kanle-10a/dp/2850101>  
(Online Zugriff am 05.03.2021)

[UKT] <https://uk.tek.com/tektronix-and-keithley-digital-multimeter/dmm6500-manual/model-dmm6500-6-1-2-digit-multimeter-3>  
(Online Zugriff am 09.11.2020)

[DGK1] <https://www.digikey.de/de/articles/the-fundamentals-of-digital-potentiometers>  
(Online Zugriff am 03.06.2021)

**Bestellübersicht:**

<https://www.digikey.de/short/97i7fn3d>

## 8. Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass die von mir vorgelegte Arbeit selbstständig und ohne unzulässige fremde Hilfe erstellt worden ist. Alle verwendeten Quellen sind in der Arbeit so aufgeführt, dass Art und Umfang der Verwendung nachvollziehbar sind.

Dortmund, 26.08.2021

---

Unterschrift – Maurice Bankowsky

## 9. Anhang

### 9.1 - Montage- und Lötarbeiten

#### 9.1.1 - Arbeitsschritte

In den nachfolgenden Unterkapiteln wird der Bestückungsvorgang der Testplatine schrittweise beschrieben. Die angefertigten Leitungen werden im Kapitel 9.1.1.5 erläutert.

##### 9.1.1.1 - Pasten der Platine

Im ersten Schritt wird eine Lötpaste mit Hilfe einer Schablone auf die zu bestückenden Kontakte der Leiterplatte aufgetragen. Die Schablone gewährleistet, dass die Lötpaste gleichmäßig aufgetragen wird. Die nachfolgende Abbildung zeigt die Testplatine mit der aufgetragenen Lötpaste.

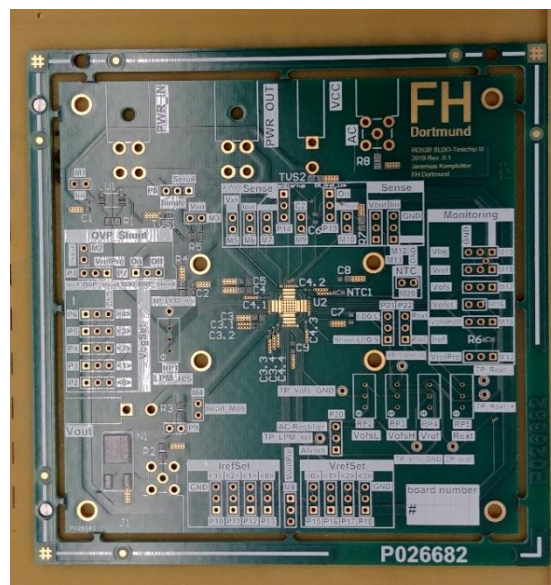


Abbildung 112: PCB mit aufgetragener Lötpaste

##### 9.1.1.2 - Bestückung der Platine und Lötvorgang

Im nächsten Arbeitsschritt werden die SMD-Bauteile (engl. Surface-Mounted Device, deutsch: oberflächenmontiertes Bauelement) aufgebracht.

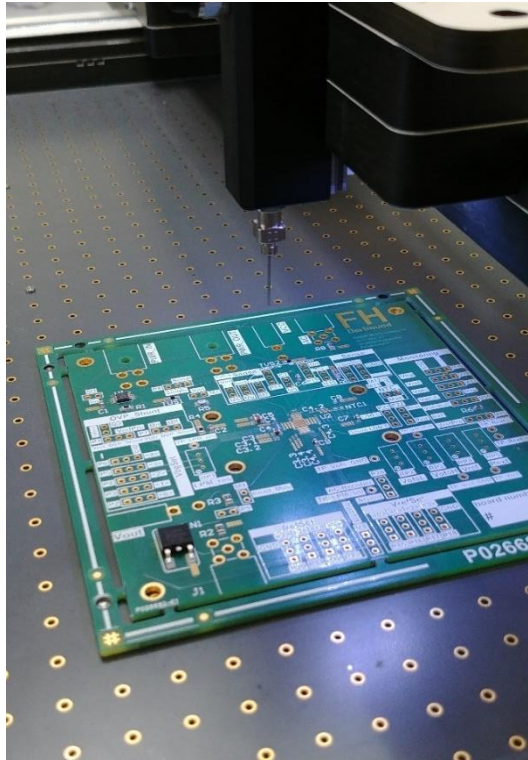


Abbildung 113: Platine auf der Bestückungsmaschine

Wegen der geringen Größe der SMD-Bauteile wird für die Bestückung der Platine eine Bestückungsvorrichtung (eC-placer) der Firma Euro Circuits zur Hilfe genommen. Die Bauteile werden durch Druckluft in Kombination mit einem kleinen Saugkopf angesogen und anschließend manuell an die vorgesehene Stelle abgesetzt. Hierbei ist darauf zu achten, dass die Bauteile ohne großen Versatz aufgebracht werden, um Kurzschlüsse oder Ausfälle durch unerwünschtes Verrutschen zu vermeiden. Zudem muss beachtet werden, dass insbesondere die Dioden TVS1 und TVS2 sowie der Instrumentenverstärker U1 korrekt ausgerichtet werden. Nach dem Bestückungsvorgang werden die Platinen erhitzt, bis sich die Lötpaste verflüssigt und die SMD Bauteile mit der Platine verbunden sind.

Abbildung 114 zeigt die Klimakammer, wobei hier ein handelsüblicher Backofen zum Einsatz kommt, in der die Platinen auf eine Temperatur von etwa 250 °C erhitzt wird. Der Temperaturverlauf wird über die Software des Computers mithilfe eines Temperatursensors im Ofen gesteuert und läuft ein typisches Temperaturprofil für ein Reflow-Lötverfahren ab. Das Reflow-Verfahren wird hierbei bestmöglich nachgestellt, da gängige Reflow-Öfen sehr teuer sind und die Anschaffung für das Projekt nicht wirtschaftlich wäre. Des Weiteren werden durch eine Lüftungsvorrichtung alle Dämpfe während des Erhitzungsvorgangs abgesaugt und durch ein offenes Fenster die Frischluftzufuhr stets gewährleistet. Nach dem Heizvorgang werden die Platinen langsam abgekühlt.





Abbildung 114: Arbeitsplatz inklusive des PCs, Klimakammer, Absaugung und Lötstation

Die Abbildung 115 zeigt eine bestückte Platine nach einem erfolgreichen Reflow-Lötvorgang.

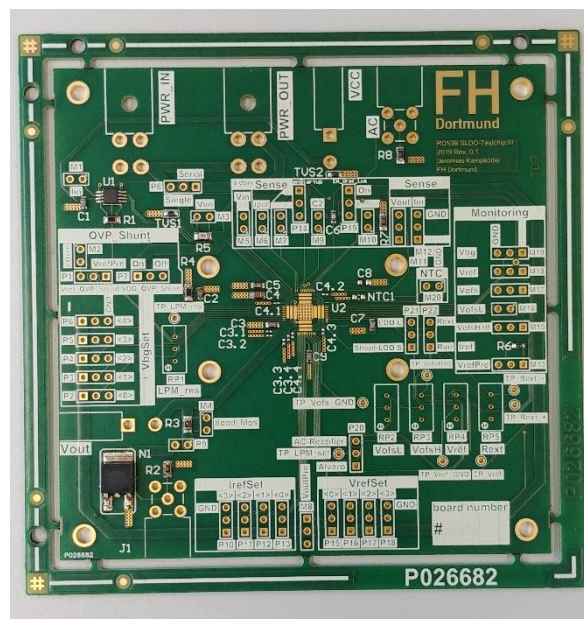


Abbildung 115: Platine mit den bestückten, gefestigten SMD Bauteilen

### 9.1.1.3 - Bonden

Der Testchip wird auf die Platine geklebt und mittels eines Bondprozesses mit den Anschlüssen auf der Platine leitend verbunden. Das sogenannte „bonden“ (engl. bond – „Verbindung oder Haftung“) ist ein Verfahren, bei dem mittels dünner Bonddrähte die Anschlüsse eines Halbleiters wie Transistoren oder Leuchtdioden mit den elektrischen Anschlüssen eines zweiten Bauteils oder eines Gehäuses verbunden werden.

Nachdem die SMD Bauteile auf die Platine gelötet worden sind, wurden die Platinen an der TU Dortmund „gebondet“.

#### 9.1.1.14 - THT-Bauteile löten

Die Durchsteckmontage (through-hole technology, THT oder pin-in-hole technology, PIH) wurde nach Fertigstellung des Bondvorgangs durchgeführt, so dass die Bauteile während des Bondprozesses nicht stören.

Die Durchsteckmontage ist dadurch gekennzeichnet, dass die Bauelemente Drahtanschlüsse haben. Die bedrahteten Bauelemente werden bei der Montage durch Kontaktlöcher in die Leiterplatte gesteckt und anschließend durch Löten mit der Leiterbahn verbunden.

Das Löten wurde hierbei mittels eines LötKolben durchgeführt.

#### 9.1.1.5 - Leitungserstellung

##### Einzelleitung

Für die serielle Verschaltung werden Leitungen zur Verbindung der einzelnen Regler benötigt. Die Leitungsenden werden zunächst abisoliert und nachfolgend mittels Crimpzange eine sogenannte „CONN SOCKET“ aufgebracht. Die beiden Enden einer Leitung werden abschließend in die Molex Stecker „CONN RECEPT 4POS DUAL“ gesteckt.

##### Y-Leitung

Für die parallele Verschaltung der Regler wird eine Y-Leitung benötigt. Ein Leitungsende der Y-Leitung besitzt 4 Leitungen zum Anschluss von 4 parallelen Reglern. Diese werden zunächst abisoliert und nachfolgend wird mittels Crimpzange eine sogenannte „CONN SOCKET“ aufgebracht. Die CONN SOCKET wird nachfolgend in die Molex Stecker „CONN RECEPT 4POS DUAL“ gesteckt. Das andere Ende der Y-Leitung besitzt einen Bananenstecker-Anschluss zum Anschluss der Spannungsversorgung.



### 9.3 - Schematic der PrePower-Platine

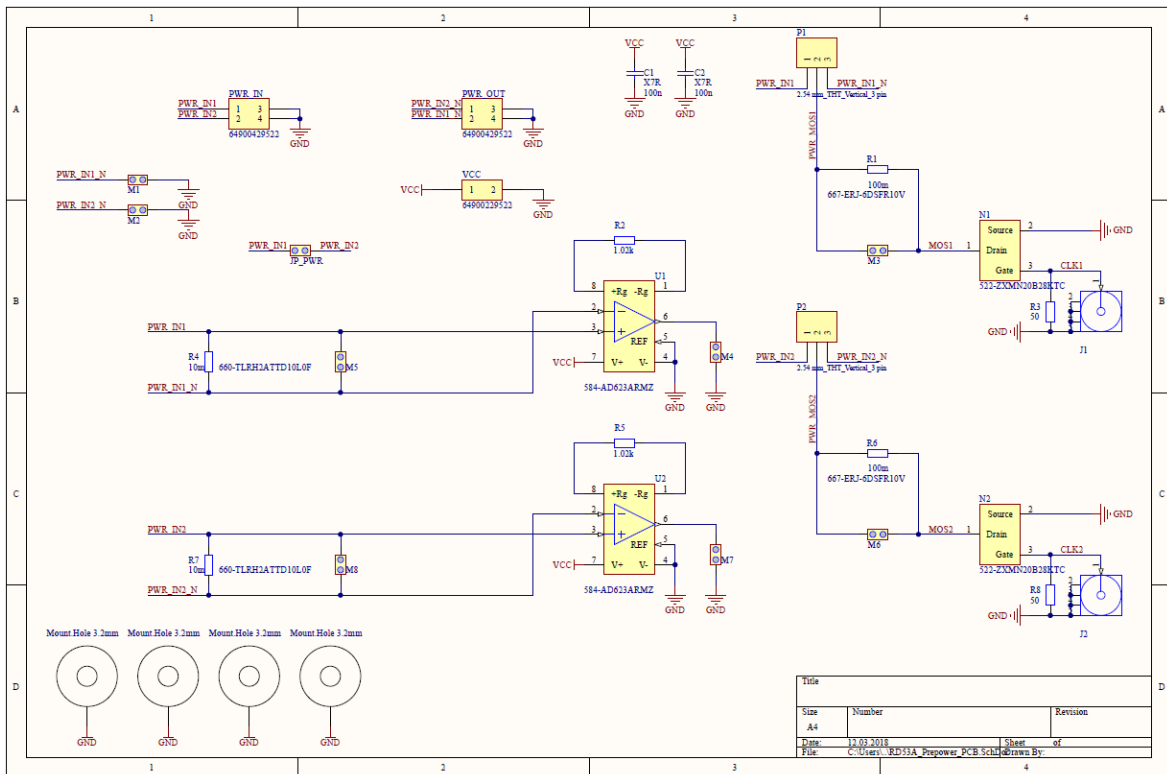


Abbildung 118: Übersicht der Schematic der PrePower-Platine

## **9.4 - Programmcode Qt**

### 9.4.1 - Header „digitalepotis.h“

```
#ifndef DIGITALEPOTIS_H
#define DIGITALEPOTIS_H

class digitalepotis
{

public:

    digitalepotis();

    int poti_widerstand_1k(int Schrittweite);

    int poti_widerstand_100k(int Schrittweite);

    void widerstandswert_poti_uebergabe_Rext(int ohmwert_Rext_inHex);

    void widerstandswert_poti_uebergabe_VofsH_Vref(int ohmwert_VofsH_inHex,
int ohmwert_Vref_inHex);

    void widerstandswert_poti_uebergabe_VofsL(int ohmwert_VofsL_inHex);

};

#endif // DIGITALEPOTIS_H
```

### 9.4.2 - Klasse „digitalepotis“

```
#include "digitalepotis.h"
#include<iostream>
#include "api_i2c.h"
#include<ftd2xx.h>
#include<wintypes.h>

#include<string>
#include<QString>
#include<unistd.h>

api_i2c i2c; // einbinden der i2c funktion
n aus der API (Arbeit Özkan)

using namespace std; // wird für die Funktion Cou
t benötigt
bool hex_converter; // wird für die Konvertierun
g HEX benötigt

digitalepotis::digitalepotis()
{
}

int digitalepotis::poti_widerstand_1k(int Schrittweite) // Methode
für die Formel zum Erstellen der Widerstandswerte
{
    int OhmWert;
    int PotiGrundwert = 75;

    OhmWert = ((Schrittweite * 4) + PotiGrundwert);

    //OhmWert = (((Schrittweite * 1000) / 256 ) + PotiGrundwert);

    return OhmWert;
}

int digitalepotis::poti_widerstand_100k(int Schrittweite) // Methode
für die Formel zum Erstellen der Widerstandswerte
{
    int OhmWert;
```

```
int PotiGrundwert = 75;

OhmWert = ((Schrittweite * 390 ) + PotiGrundwert);

//OhmWert = (((Schrittweite * 100000) / 256 ) + PotiGrundwert);

return OhmWert;
}

void digitalepotis::widerstandswert_poti_uebergabe_Rext(int ohmwert_Rext_inHex)
// Methode zum aufrufen der Schreibfunktion des Potis
{

int slaveaddress_Rext;

slaveaddress_Rext = 0X58;

int outputdata_Rext[3];
// Array mit 3 Feldern

outputdata_Rext[0] = 0X00;
// 00 ins erste Feld
outputdata_Rext[1] = 0X01;
// 01 ins zweite Feld
outputdata_Rext[2] = ohmwert_Rext_inHex;
// Laden des HEX Wertes in das dritte Feld

cout<<"\n"<<"_Rext Poti_ " <<"SlaveAdresse:"<<slaveaddress_Rext<<" / Fest
wert:"<<outputdata_Rext[0]<<outputdata_Rext[1]<<endl; // Ausgabe der
Werte über die Konsole; zur Kontrolle
std::cout<<std::hex<<"HEX: " <<outputdata_Rext[2]<<std::endl;

i2c.i2c_write(slaveaddress_Rext, outputdata_Rext, 3);
// Finktion i2c_write aus der API ausführen

}

void digitalepotis::widerstandswert_poti_uebergabe_VofsH_Vref(int ohmwert_VofsH_inHex, int ohmwert_Vref_inHex)
{

int slaveaddress_VofsH_Vref;

slaveaddress_VofsH_Vref = 0X5A;
```

```
int outputdata_VofsH_Vref[5];

outputdata_VofsH_Vref[0] = 0X00;
outputdata_VofsH_Vref[1] = 0X01;
outputdata_VofsH_Vref[2] = ohmwert_VofsH_inHex;
outputdata_VofsH_Vref[3] = 0X00;
outputdata_VofsH_Vref[4] = ohmwert_Vref_inHex;

cout<<"\n"<<"_VofsH & Vref Poti_ "<<"SlaveAdresse:"<<slaveaddress_VofsH_
Vref<<" / Festwert:"<<outputdata_VofsH_Vref[0]<<outputdata_VofsH_Vref[1]<<en
dl;
std::cout<<std::hex<<"HEX: "<<outputdata_VofsH_Vref[2]<<std::endl;
cout<<" - Festwert:"<<outputdata_VofsH_Vref[3]<<endl;
std::cout<<std::hex<<"HEX: "<<outputdata_VofsH_Vref[4]<<std::endl;

i2c.i2c_write(slaveaddress_VofsH_Vref, outputdata_VofsH_Vref, 5);

}

void digitalepotis::widerstandswert_poti_uebergabe_VofsL(int ohmwert_VofsL_
inHex)
{
int slaveaddress_VofsL;

slaveaddress_VofsL = 0X5E;

int outputdata_VofsL[3];

outputdata_VofsL[0] = 0X00;
outputdata_VofsL[1] = 0X01;
outputdata_VofsL[2] = ohmwert_VofsL_inHex;

cout<<"\n"<<"_VofsL Poti_ "<<"SlaveAdresse:"<<slaveaddress_VofsL<<" / Fe
stwert:"<<outputdata_VofsL[0]<<outputdata_VofsL[1]<<endl; // Ausgabe der
Werte über die Konsole; zur Kontrolle
std::cout<<std::hex<<"HEX: "<<"HEX: "<<outputdata_VofsL[2]<<std::endl;

i2c.i2c_write(slaveaddress_VofsL, outputdata_VofsL, 3);

}
```



### 9.4.3 - Klasse „gpib“

```
/* KARL MACHA/ Bachelorthesis/ Sommersemester 2017/ Fachhochschule Dortmund
 * In diesem Programm wird die Grapgische Oberfläche in gpib.ui codiert. Die
 * einzelnen Elemente sind mit Name und einer
 * Endung des Widgets hinterlegt. Beispielsweise könnte ein PushButton, mit
 * welchem das Programm beendet wird ClosePB heißen.
 * Den Code kann man den Memberfunktions entnehmen.
 */
// Erweiterung um die Potis durch Bankowsky

//Eingebundene Bibliotheken für das GUI
#include "gpib.h" //Einbindung des Headers vom der Klasse GPIB
#include "ui_gpib.h" //Verknüpfung mit dem UserInterface
#include "QString" //Einbindung von QString, um den Datentyp String
nutzen zu können
#include "qvector.h" //Bibliothek, mit welcher der Datentyp Vektor
eingebunden wird
#include "qfile.h" //Hiermit kann der Datentyp File eingelesen
und kopiert werden
#include <QTime> //Stadartfunktion, um mit der aktuellen Zeit
und Datum arbeiten zu können
#include "qcolor.h"
#include "qwt_plot.h" // Ermöglicht die Einbindung von QWT-
Plot Widgets
#include "qwt_plot_curve.h" //Ermöglicht die Erstellung von Kurven im Zu
sammenhang mit Zahlenwerten
#include "qwt_plot_grid.h" //Bindet das Tool ein, mit welchem R
aster für Diagramme entwickelt werden können
#include "SILIB_GPIB_Interfaces.h" //Einbindung des GPIB Adapters und d
ie Nutzung mit einem Interface
#include "SILIB_GPIB_Device.h" //Einbindung des GPIB- Adapters
#include "Silib_GPIB_Keithley2000.h" //Einbindung des Multimeters Keithley
2000
#include "Silib_GPIB_Keithley24xx.h" //Einbindung des Sourcemeters Keithle
y 2401
#include "sldo65.h" //Einbindung der Funktionen aus SLDO
65

#include "Silib_GPIB_Keithley246x.h"
#include <QMessageBox>
#include <QFileDialog>
#include <QFileInfo>
#include "testboard.h"

#include "fhdo_usb_interface.h"
#include "fhdo_usb_device.h"
#include "fhdo_usb_keithleydmm6500.h"
```

```
// Für das UserInterface (UI) zu erzeugende Objekte zusammen mit den zugehörigen Destruktoren

using namespace std;          // Banko : wird für die Funktion Cout benötigt
bool ok;                      // Banko : wird für die HEX konvertierung benötigt
int z;

gpib::gpib(QWidget *parent) : // Constructor
    QMainWindow(parent),
    ui(new Ui::gpib)
{
    ui->setupUi(this);
    z = new sldo65();

    // Banko : DROPDOWN MENÜS DER BUTTONS, ZUM ÜBERGEBEN DER OHMSCHENWERTE :
    ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
    :::::::::::::::

    int ohm_Rext, ohm_Rext_End, ohm_VofsH, ohm_VofsH_End, ohm_VofsL, ohm_Vref;

    for (int Schrittweite = 0; Schrittweite < 256; Schrittweite++)
        // Schleife bis 255 zum Erstellen der Widerstandswerte
        {

            ohm_Rext = z->
>poti_widerstand_1k(Schrittweite);          // Aufrufen der
Funktion poti_widerstand aus der Klasse digitalepotis

            ohm_Rext_End = z->poti_widerstand_1k(Schrittweite);

            ohm_VofsH = z->poti_widerstand_100k(Schrittweite);

            ohm_VofsH_End = z->poti_widerstand_100k(Schrittweite);

            ohm_VofsL = z->poti_widerstand_100k(Schrittweite);

            ohm_Vref = z->poti_widerstand_100k(Schrittweite);

            ui->comboBox_Rext->
>addItem(QString::number(ohm_Rext));        // Laden des Wertes
in die ComboBox
            ui->comboBox_Rext->
>setCurrentIndex(132);                      // Startwert setzen

            ui->comboBox_Rext_end->addItem(QString::number(ohm_Rext_End));
```

```
        ui->comboBox_Rext_end->setCurrentIndex(144);

        ui->comboBox_VofsH->addItem(QString::number(ohm_VofsH + 75000));
        ui->comboBox_VofsH->setCurrentIndex(76);

        ui->comboBox_VofsH_end->
>addItem(QString::number(ohm_VofsH_End + 75000));
        ui->comboBox_VofsH_end->setCurrentIndex(77);

        ui->comboBox_VofsL->addItem(QString::number(ohm_VofsL));
        ui->comboBox_VofsL->setCurrentIndex(18);

        ui->comboBox_Vref->addItem(QString::number(ohm_Vref + 75000));
        ui->comboBox_Vref->setCurrentIndex(163);

    }
```

// Banko : unsichtbar machen der nicht benötigten label, bei Weiterentwicklung oder bedarf können diese als Speicher der Zwischenschritte angezeigt werden

```
        ui->label_Rext_schritte->setVisible(0);
        ui->label_VofsH_schritte->setVisible(0);
        ui->label_Vref_schritte->setVisible(0);
        ui->label_VofsL_schritte->setVisible(0);

        ui->label_Rext_schritte_end->setVisible(0);
        ui->label_VofsH_schritte_end->setVisible(0);

        ui->hex_Rext->setVisible(0);
        ui->hex_VofsH->setVisible(0);
        ui->hex_Vref->setVisible(0);
        ui->hex_VofsL->setVisible(0);

        ui->hex_Rext_end->setVisible(0);
        ui->hex_VofsH_end->setVisible(0);

    }

gpib::~gpib() //Destructor
{
    delete ui;
    delete z;
}
```

```
void gpib::on_start_button_clicked()
{
    double a1,a2,a3,a4,a5,a6,a7;
    QString y1,y2,y3,y4,y5,y6,y7;

    double s, e, st, r1,r2,r3,r4,r5,r6,r7,r8,delay,limit,crange,vrange;
        //Die verwendeten reelen Zahlen vom Typ Double
    int i;
    std::vector<double> a, h1,h2,h3,h4,h5,h6,h7,h8;
        //Deklaration der Vektoren: In b wird der Vektor mit Messwerten
        //gefüllt, in h1 und h2 werden zwei Vektoren für die Zeichung der Kurve erstellt
    QString x1,x2,x3,x4,x5,x6,x7,x8,y,x,w,g,notiz;
        // Stiller Notiz eingefügt //Deklaration der Benötigten String Dateien
    QString ch_name_16,ch_name_17,ch_name_18,ch_name_19,ch_name_20,ch_name_21,
ch_name_22,names; //Zorn Kanalnamen eingefügt //Ausgaben der Multimeter
    QwtPlotCurve series1,series2,series3,series4,series5,series6,series7;
        //Deklaration der Kurve für das Diagramm
    QwtPlotGrid raster;
        //Deklaration des Rasters für das Diagramm

    if (ui->cb_test_mult->isChecked())
    {
        a1 = z->testvoltage();
        a2 = z->testvoltage2();
        a3 = z->testvoltage3();
        a4 = z->testvoltage4();
        a5 = z->testvoltage5();
        a6 = z->testvoltage6();
        a7 = z->testvoltage7();

        y1= QString::number(a1);
        y2= QString::number(a2);
        y3= QString::number(a3);
        y4= QString::number(a4);
        y5= QString::number(a5);
        y6= QString::number(a6);
        y7= QString::number(a7);

        ui->outkeith->setText(y1);
        ui->outkeith2->setText(y2);
        ui->outkeith3->setText(y3);
        ui->outkeith4->setText(y4);
        ui->outkeith5->setText(y5);
        ui->outkeith6->setText(y6);
        ui->outkeith7->setText(y7);
    }
}
```

```
        // Banko : einmaliges Übergeben der Werte, zum Start bzw. beim i
nitialisieren der Multimeter
        int ohmwert_Rext_inHex, ohmwert_VofsH_inHex, ohmwert_Vref_inHex,
        ohmwert_VofsL_inHex; // integer Variablen für die Slave Adresse und den
Ohmwert in HEX
        QString ohmwert_Rext, ohmwert_VofsH, ohmwert_Vref, ohmwert_VofsL
;
        // String für das Umwandeln des ohmschen Wertes

        ohmwert_Rext = ui->hex_Rext-
>text(); // in den String
den Text des Feldes hex_Rext laden
        ohmwert_Rext_inHex = ohmwert_Rext.toInt(&ok,16);
        // umwandeln in INT

        ohmwert_VofsH = ui->hex_VofsH-
>text(); // in den String
den Text des Feldes hex_VofsH laden
        ohmwert_VofsH_inHex = ohmwert_VofsH.toInt(&ok,16);
        // umwandeln in INT

        ohmwert_Vref = ui->hex_Vref-
>text(); // in den String
den Text des Feldes hex_Vref laden
        ohmwert_Vref_inHex = ohmwert_Vref.toInt(&ok,16);
        // umwandeln in INT

        ohmwert_VofsL = ui->hex_VofsL-
>text(); // in den String
den Text des Feldes hex_Rext laden
        ohmwert_VofsL_inHex = ohmwert_VofsL.toInt(&ok,16);
        // umwandeln in INT

        // std::cout<<std::hex<<ohmwert_Rext_inHex<<"TEST"<<std::endl;
        // hierdurch könnte man den Zwischenschritt überprüf
en, AUSgabe auf der Konsole

        z-
>widerstandswert_poti_uebergabe_Rext(ohmwert_Rext_inHex);
        // Übergeben der HEX Werte
        z-
>widerstandswert_poti_uebergabe_VofsH_Vref(ohmwert_VofsH_inHex, ohmwert_Vref
_inHex);
        z->widerstandswert_poti_uebergabe_VofsL(ohmwert_VofsL_inHex);

    }

    else if (ui->cb_loadreg->isChecked() || ui-> cb_linereg_curr -
>isChecked() || ui-> cb_linereg_vol ->isChecked())
```

```
{
    raster.setPen(Qt::gray); //Setzt die Farbe des
Rasters in Grau
    raster.attach(ui-
>plot); //Einfügen der Raster im Diagramm

    //Deklaration der Eingaben
    s= ui->startLE-
>text().toDouble(); //Startwert der Messung
    e=ui->endLE->text().toDouble(); //Endwert der Messung
    st=ui->stepLE-
>text().toDouble(); //Schrittweite der Messung
    limit=ui->limitLE->text().toDouble();
    w= ui->n-
>text(); //Name für das File muss hier im ui ein
getippt werden (Name.txt)

    if (ui->cb_loadreg->isChecked())
    {
        ui->plot-
>setAxisTitle(QwtPlot::xBottom, "I(A)"); //Beschriftung der X-
Achse mit I für Strom
        ui->plot-
>setAxisTitle(QwtPlot::yLeft, "U(V)"); //Bechriftung der Y-
Achse mit U für Spannung
        if (ui->cb_k2400->isChecked())
        {
            a=z-
>loadreg(s, e, st, limit); //K2400 //per Memberfunktion
loadreg von z(code in sldo65.cpp) soll der Vector b für die Ausgabe im UI
erstellt werden
        }
        else
        {
            a=z-
>loadreg2460(s, e, st, limit); //K2460 //per Memberfunktion
loadreg von z(code in sldo65.cpp) soll der Vector b für die Ausgabe im UI
erstellt werden
        }
    }
    else if(ui-> cb_linereg_curr ->isChecked())
    {
        ui->plot-
>setAxisTitle(QwtPlot::xBottom, "I(A)"); //Beschriftung der X-
Achse mit I für Strom
```

```
        ui->plot-
>setAxisTitle(QwtPlot::yLeft, "U(V)");    //Beschriftung der Y-
Achse mit U für Spannung

        if (ui->cb_k2400->isChecked())
        {
            a=z-
>lineregcur(s, e, st, limit);            //K2400 //per Memberfunktion line
regcur von z(code in sldo65.cpp) soll der Vector b für die Ausgabe im UI
erstellt werden
        }
        else
        {
            a=z-
>lineregcur2460(s, e, st,limit);        //K2460 //per Memberfunktion line
regcur von z(code in sldo65.cpp) soll der Vector b für die Ausgabe im UI
erstellt werden
        }
    }
    else if(ui-> cb_linereg_vol ->isChecked())
    {
        ui->plot-
>setAxisTitle(QwtPlot::xBottom, "Uin(V)");    //Beschriftung der X-
Achse mit U für Spannung
        ui->plot-
>setAxisTitle(QwtPlot::yLeft, "Uout(V)");    //Stiller Spannugnen Bechrift
ung der Y-Achse

        if (ui->cb_k2400->isChecked())
        {
            a=z-
>lineregvol(s, e, st,limit);            //Stiller K2400//per Member
funktion lineregvol von z(code in sldo65.cpp) soll der Vector a für die
Ausgabe im UI erstellt werden
        }
        else
        {
            a=z-
>lineregvol2460(s, e, st,limit);        //Stiller K2460 //per Membe
rfunktion lineregvol von z(code in sldo65.cpp) soll der Vector a für die
Ausgabe im UI erstellt werden
        }
    }
}
```

```

// Banko : ELSE IF ABFRAGE DER SWEEP FUNKTIONEN/CHECKBOXEN
else if(ui-> checkBox_sweep_Rext ->isChecked())
{
    QString s_sweep_Rext, e_sweep_Rext, st_sweep_Rext;
    int s_sweep_Rext_inHex, e_sweep_Rext_inHex, st_sweep_Rext_inHex;

    s_sweep_Rext = ui->hex_Rext-
>text(); // Startwert Sweep Rext lade
n
    e_sweep_Rext = ui->hex_Rext_end-
>text(); // Endwert Sweep Rext laden
    st_sweep_Rext = ui->spinBox_Rext-
>text(); // Schrittweite Sweep Rext laden

    s_sweep_Rext_inHex = s_sweep_Rext.toInt(&ok,16);
    // umwandeln in HEX
    e_sweep_Rext_inHex = e_sweep_Rext.toInt(&ok,16);
    st_sweep_Rext_inHex = st_sweep_Rext.toInt(&ok,10);

    ui->plot-
>setAxisTitle(QwtPlot::xBottom, "I(A)"); //Beschriftung
    der X-Achse mit I für Strom
    ui->plot-
>setAxisTitle(QwtPlot::yLeft, "U(V)"); //Bechriftung
    der Y-Achse mit U für Spannung

    while ( s_sweep_Rext_inHex <= e_sweep_Rext_inHex )
    // Schleife : Startwert kleiner Endwert ?
    {

        if (ui->cb_k2400-
>isChecked() // Auswahl zwischen
    2400 und 2460
        {

            cout<<"\n"<<endl;
            z->widerstandswert_poti_uebergabe_Rext(s_sweep_Rext_inHex);
            a=z-
>lineregcur(s, e, st, limit); //K2400
            //per Memberfunktion lineregcur von z(code in sldo65.cpp) soll der Vector
            b für die Ausgabe im UI erstellt werden
            // cout<<"lineregcur aus der Klasse sldo60 ausführen"<<endl
            ; //K2400 //per Memberfunktion lineregcur von z(code in sldo65.cp
            p) soll der Vector b für die Ausgabe im UI erstellt werden
        }
    }
else

```



```

        {
            cout<<"\n"<<endl;
            z->widerstandswert_poti_uebergabe_Rext(s_sweep_Rext_inHex);
            a=z-
>lineregcur2460(s, e, st,limit); //K2460 /
/per Memberfunktion lineregcur von z(code in sldo65.cpp) soll der Vector b
für die Ausgabe im UI erstellt werden
            // cout<<"lineregcur2460 aus der Klasse sldo60 ausführen"<<
endl; //K2460 //per Memberfunktion lineregcur von z(code in sldo65.cpp
) soll der Vector b für die Ausgabe im UI erstellt werden
        }

        // std::cout<<std::hex<<s_sweep_Rext_inHex<<"start"<<std::endl;
        // std::cout<<std::hex<<e_sweep_Rext_inHex<<"ende"<<std::endl;
        // std::cout<<std::hex<<st_sweep_Rext_inHex/4<<"step"<<std::end
l;

        s_sweep_Rext_inHex = s_sweep_Rext_inHex + (st_sweep_Rext_in
Hex/4); // Erhöhe um Schrittweite : Schrittweite / 4

    }

}
else if(ui-> checkBox_sweep_VofsH ->isChecked())
{

    QString s_sweep_VofsH, e_sweep_VofsH, st_sweep_VofsH;
    int s_sweep_VofsH_inHex, e_sweep_VofsH_inHex, st_sweep_VofsH_inH
ex;

    s_sweep_VofsH = ui->hex_VofsH-
>text(); // Startwert Sweep VofsH laden
    e_sweep_VofsH = ui->hex_VofsH_end-
>text(); // Endwert Sweep VofsH laden
    st_sweep_VofsH = ui->spinBox_VofsH-
>text(); // Schrittweite Sweep VofsH laden

    s_sweep_VofsH_inHex = s_sweep_VofsH.toInt(&ok,16);
// Werte in HEX wandeln
    e_sweep_VofsH_inHex = e_sweep_VofsH.toInt(&ok,16);
    st_sweep_VofsH_inHex = st_sweep_VofsH.toInt(&ok,16);

    ui->plot-
>setAxisTitle(QwtPlot::xBottom, "I(A)"); //Beschriftung
der X-Achse mit I für Strom

```

```
        ui->plot-
>setAxisTitle(QwtPlot::yLeft, "U(V)"); //Bechriftung
    der Y-Achse mit U für Spannung

        while (s_sweep_VofsH_inHex <= e_sweep_VofsH_inHex)
// Schleife : Startwert kleiner gleich Endwert ?
    {

        if (ui->cb_k2400->isChecked())
    {

            cout<<"\n"<<endl;
            z-
>widerstandswert_poti_uebergabe_Rext(s_sweep_VofsH_inHex);
            a=z-
>lineregcur(s, e, st, limit); //K2400 /
//per Memberfunktion lineregcur von z(code in sldo65.cpp) soll der Vector b
für die Ausgabe im UI erstellt werden
            // cout<<"lineregcur aus der Klasse sldo60 ausführen"<<endl
;            //K2400 //per Memberfunktion lineregcur von z(code in sldo65.cpp
) soll der Vector b für die Ausgabe im UI erstellt werden
        }
        else
        {
            cout<<"\n"<<endl;
            z-
>widerstandswert_poti_uebergabe_Rext(s_sweep_VofsH_inHex);
            a=z-
>lineregcur2460(s, e, st,limit); //K2460 /
//per Memberfunktion lineregcur von z(code in sldo65.cpp) soll der Vector b
für die Ausgabe im UI erstellt werden
            // cout<<"lineregcur2460 aus der Klasse sldo60 ausführen"<<
endl; //K2460 //per Memberfunktion lineregcur von z(code in sldo65.cpp
) soll der Vector b für die Ausgabe im UI erstellt werden
        }

            s_sweep_VofsH_inHex = s_sweep_VofsH_inHex + st_sweep_VofsH_
inHex; // Erhöhe um Schrittweite : Schrittweite / 390

    }

}
```

```

        for (i=0; i<a.size()+1; i=i+8) //Stiller Änderung
ng von i<a.size() da sonst letzter Wert in Diagramm fehlte //Schleifendurchl
auf für die Erstellung der Ausgabe, des Graphes sowie der Hinterlegung in
einer Datei
    {
        r1=a[i]; //r1-
r5 sind Hilfsvariablen, welchen die einzelnen Werte des Vektors a zugewiesen
werden, um diese später bei x1, x2 in einen String zu konvertieren
        r2=a[i+1];
        r3=a[i+2];
        r4=a[i+3];
        r5=a[i+4];
        r6=a[i+5];
        r7=a[i+6];
        r8=a[i+7];

        h1.push_back(a[i]); //h1, h2 sind die
Vektoren für das Diagramm, welche den Verlauf des Graphen angeben. Denen
werden die einzelnen Werte von a zugewiesen
        h2.push_back(a[i+1]); //Bsp: a[0]=h1[0
], a[1]=h2[0], usw.
        h3.push_back(a[i+2]);
        h4.push_back(a[i+3]);
        h5.push_back(a[i+4]);
        h6.push_back(a[i+5]);
        h7.push_back(a[i+6]);
        h8.push_back(a[i+7]);

        x1=QString::number(r1); //Umwandlung der
Variablen r1,r2 in eine QSTRINGvariable x1,x2
        x2=QString::number(r2);
        x3=QString::number(r3);
        x4=QString::number(r4);
        x5=QString::number(r5);
        x6=QString::number(r6);
        x7=QString::number(r7);
        x8=QString::number(r8);
        x=x1+"\t"+x2; //Addition der
Strings x1-x5 in x um die Ausgabe durchzuführen
        x=x+"\t"+x3;
        x=x+"\t"+x4;
        x=x+"\t"+x5;
        x=x+"\t"+x6;
        x=x+"\t"+x7;
        x=x+"\t"+x8;
        if(i<a.size()) //Stiller ansonsten
wird ein Wert zuviel ausgegeben
    {

```

```

        g=g+x+"\n"; // g ist der String
, welcher die Ausgabe für die gesamten Messswerte für das File Speichert
    }
    series1.setRawSamples(&h1[0],&h2[0],i/8); //die Kurve series1
wird gezeichnet
    series2.setRawSamples(&h1[0],&h3[0],i/8);
    series3.setRawSamples(&h1[0],&h4[0],i/8);
    series4.setRawSamples(&h1[0],&h5[0],i/8);
    series5.setRawSamples(&h1[0],&h6[0],i/8);
    series6.setRawSamples(&h1[0],&h7[0],i/8);
    series7.setRawSamples(&h1[0],&h8[0],i/8);

};
    series1.setPen(Qt::blue); //Setzt die
Linie der Kurve series1 in die Farbe Blau
    series2.setPen(Qt::red); //Setzt die
Linie der Kurve series1 in die Farbe Rot
    series3.setPen(Qt::green); //Setzt die
Linie der Kurve series1 in die Farbe Grün
    series4.setPen(Qt::black); //Setzt die
Linie der Kurve series1 in die Farbe Gelb
    series5.setPen(Qt::darkYellow);
    series6.setPen(Qt::darkMagenta);
    series7.setPen(Qt::darkCyan);

    series1.attach(ui-
>plot); //Die Kurven werden in das Diagramm
Plot eingefügt
    series2.attach(ui->plot);
    series3.attach(ui->plot);
    series4.attach(ui->plot);
    series5.attach(ui->plot);
    series6.attach(ui->plot);
    series7.attach(ui->plot);

    ui->plot-
>replot(); //Die Kurve wird im Plot eingezeichnet

    this->repaint();

    notiz= ui->Notiz_LE-
>text(); //Stiller: Auslesen des Notizfeldes
und speichern in Variable notiz
    ch_name_16 = ui->channel_names_le_16-
>text(); //Stiller: Auslesen der Kanalnamen Variable 16
    ch_name_17 = ui->channel_names_le_17-
>text(); //Stiller: Auslesen der Kanalnamen Variable 17

```

```
        ch_name_18 = ui->channel_names_le_18-
>text();          //Stiller: Auslesen der Kanalnamen Variable 18
        ch_name_19 = ui->channel_names_le_19-
>text();          //Stiller: Auslesen der Kanalnamen Variable 19
        ch_name_20 = ui->channel_names_le_20-
>text();          //Stiller: Auslesen der Kanalnamen Variable 20
        ch_name_21 = ui->channel_names_le_21-
>text();          //Stiller: Auslesen der Kanalnamen Variable 21
        ch_name_22 = ui->channel_names_le_22-
>text();          //Stiller: Auslesen der Kanalnamen Variable 22

        names=ch_name_16+"\t"+ch_name_17;
//Addition der Strings der Kanalnamen
        names=names+"\t"+ch_name_18;
        names=names+"\t"+ch_name_19;
        names=names+"\t"+ch_name_20;
        names=names+"\t"+ch_name_21;
        names=names+"\t"+ch_name_22;

        QFile data(w);
//Eine *.txt wird mit dem Namen aus dem String w (Z.214) erstellt
        {
            //for writing line by line to text file
            if (data.open(QIODevice::WriteOnly | QIODevice::Append))
            {
                QTextStream out(&data);
//In der Variable out werden die *FILE Ausgaben gespeichert

                if (ui->cb_loadreg->isChecked())
                {
//Load Reg wird im *File hinterlegt
                    out << "Load Regulation ";
                }
                else if(ui-> cb_linereg_curr ->isChecked())
                {
//Line Reg Cur wird im *File hinterlegt
                    out << "Line Regulation Current ";
                }
                else if(ui-> cb_linereg_vol ->isChecked())
                {
//Line Reg Vol wird im *File hinterlegt
                    out << "Line Regulation Voltage ";
                }
                else if(ui-> checkBox_sweep_Rext ->isChecked())
                {
//Banko: Line Reg Cur und Sweep wird im *File hinterlegt
                    out << "Sweep Rext / Line Regulation Current ";
                }
                else if(ui-> checkBox_sweep_Vofsh ->isChecked())
```

```
{
    out << "Sweep VofsH / Line Regulation Current ";
//Banko:Line Reg Cur und Sweep wird im *File hinterlegt
}

//Stiller Anmerken welche SMU läuft
if (ui->cb_k2400->isChecked())
{
    out << "using Keithley 2400 ";
}
else
{
    out << "using Keithley 2460 ";
}
out << QDateTime::currentTime().toString();
//Uhrzeit der Messung wird in der Datei hinterlegt
out << " ";
//Tabulatorleerschritt wird im file hinterlegt
out << QDateTime::currentDate().toString() << endl;
//Datum der Messung wird in der Datei hinterlegt
out << "" << endl;
out << notiz << endl;
//Stiller : Ausgabe der Notiz aus der GUI im Logfile für Anmerkungen was gerade gemessen wird.
out << "" << endl;

if (ui->cb_loadreg->isChecked())
{
    out << "I(A)\t" + names << endl;
//Stiller : Ausgabe Loadreg
}
else if(ui-> cb_linereg_curr ->isChecked())
{
    out << "I(A)\t" + names << endl;
//Stiller : Ausgabe LineReg Current
}
else if(ui-> cb_linereg_vol ->isChecked())
{
    out << "U(V)\t" + names << endl;
//Stiller : Ausgabe Line Reg Voltage
}
else if(ui-> checkBox_sweep_Rext ->isChecked())
{
    out << "I(A)\t" + names << endl;
//Banko : Ausgabe LineReg Current und Sweep
}
else if(ui-> checkBox_sweep_VofsH ->isChecked())
```

```
        {
            out << "I(A)\t" + names << endl;
//Banko : Ausgabe LineReg Current und Sweep
        }

        out << g <<endl;
//Die Messwerte werden in der Datei hinterlegt
        data.close();
//Die Datei wird geschlossen
    }
}

else if (ui->cb_start_ramp->isChecked())
{

    s= ui->ramp_start_le-
>text().toDouble();           //Startwert der Rampe
    e=ui->ramp_end_le-
>text().toDouble();           //Endwert der Rampe
    st=ui->ramp_step_le-
>text().toDouble();           //Schritte der Rampe
    delay=ui->ramp_delay_le-
>text().toDouble();           //Delay zwischen den Rampen
    limit=ui->limitLE->text().toDouble();
    crange=ui->crangeLE->text().toDouble();
    vrange=ui->vrangeLE->text().toDouble();

    if (ui->cb_ramp_current->isChecked())
    {
        double res;
        res=z->CSweep(delay, st, s, e, limit, crange, vrange);
    }
    else if (ui->cb_ramp_voltage->isChecked())
    {
        double res;
        res=z->VSweep(delay, st, s, e, limit, crange, vrange);
    }
}
else if (ui->cb_stop_ramp->isChecked())
{
    double res;
    res=z->StopSweep();
}
else
{
```

```
        QMessageBox msgBox;
        msgBox.setText("Bitte wählen Sie einen Test aus!");
        msgBox.exec();
    }
}

void gpib::on_actionAbout_triggered()
{
    QMessageBox msgBox;
    msgBox.setText("Testsoftware for SLDO characterization. Created by M.Eng
. Andreas Stiller. Dortmund 2017-
2019. Special thanks to B.Eng. Karl Macha and B.Eng. Jendrik Zorn for helpin
g improve the software.");
    msgBox.exec();
}

void gpib::on_actionExit_triggered()
{
    this->close();
}

void gpib::on_actionOpen_triggered()
{
    QString str_fileName = QFileDialog::getOpenFileName(this,tr("Select Simu
lation Data"),"/",tr("All (*)"));
    QFileInfo fileNameInfo (str_fileName);

    QString str_fileName_short = fileNameInfo.fileName();           // return
s filename
    QString str_absoluteFilePath = fileNameInfo.absoluteFilePath();
// returns path

    ui->n->setText(str_fileName);
}

void gpib::on_actionTestchips_triggered() //neues UI soll angezeigt werden
mit Testchip A-Eigenschaften
{
    testboard =new TestBoard(this);
    testboard->show();
}
}
```



```
// Buttonabfragen ab hier, damit nie mehr als ein Button geklickt ist
```

```
void gpib::on_cb_k2400_clicked()
{
    if (ui->cb_k2400->isChecked())
    {
        ui->cb_k2460->setChecked(0);
    }
    else
    {
        ui->cb_k2460->setChecked(1);
    }
}
```

```
void gpib::on_cb_k2460_clicked()
{
    if (ui->cb_k2460->isChecked())
    {
        ui->cb_k2400->setChecked(0);
    }
    else
    {
        ui->cb_k2400->setChecked(1);
    }
}
```

```
void gpib::on_cb_ramp_current_clicked()
{
    if (ui->cb_ramp_current->isChecked())
    {
        ui->cb_ramp_voltage->setChecked(0);
    }
    else
    {
        ui->cb_ramp_voltage->setChecked(1);
    }
}
```

```
void gpib::on_cb_ramp_voltage_clicked()
{
    if (ui->cb_ramp_voltage->isChecked())
    {
        ui->cb_ramp_current->setChecked(0);
    }
    else
```

```
{
    ui->cb_ramp_current->setChecked(1);
}
}

void gpib::on_cb_linereg_vol_clicked()
{
    if (ui->cb_linereg_vol->isChecked())
    {
        ui->cb_linereg_curr->setChecked(0);
        ui->cb_start_ramp->setChecked(0);
        ui->cb_test_mult->setChecked(0);
        ui->cb_loadreg->setChecked(0);
        ui->cb_stop_ramp->setChecked(0);
        ui->checkBox_sweep_Rext->setChecked(0);
        ui->checkBox_sweep_VofsH->setChecked(0);

        //Sweep Felder aktivieren

        ui->startLE->setEnabled(1);
        ui->endLE->setEnabled(1);
        ui->stepLE->setEnabled(1);
        ui->cb_k2460->setEnabled(1);
        ui->cb_k2400->setEnabled(1);
        ui->label_9->setEnabled(1);
        ui->label_29->setEnabled(1);
        ui->label->setEnabled(1);
        ui->label_2->setEnabled(1);
        ui->label_3->setEnabled(1);

        //Ramp Felder deaktivieren

        ui->label_23->setDisabled(1);
        ui->label_28->setDisabled(1);
        ui->label_24->setDisabled(1);
        ui->label_27->setDisabled(1);
        ui->label_26->setDisabled(1);
        ui->label_25->setDisabled(1);
        ui->cb_ramp_current->setDisabled(1);
        ui->cb_ramp_voltage->setDisabled(1);
        ui->ramp_delay_le->setDisabled(1);
        ui->ramp_step_le->setDisabled(1);
        ui->ramp_start_le->setDisabled(1);
        ui->ramp_end_le->setDisabled(1);

        // VofsH Sweep Felder deaktivieren

        ui->label_VofsH_endValue->setDisabled(1);
```

```
ui->comboBox_VofsH_end->setDisabled(1);
ui->label_VofsH_7->setDisabled(1);
ui->label_VofsH_schritte_end->setDisabled(1);
ui->hex_VofsH_end->setDisabled(1);
ui->label_VofsH_stepSize->setDisabled(1);
ui->spinBox_VofsH->setDisabled(1);

// Rext Sweep Felder deaktivieren

ui->label_Rext_endValue->setDisabled(1);
ui->comboBox_Rext_end->setDisabled(1);
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);

// andere Sachen aktivieren

ui->pushButton_VofsL->setEnabled(1);
ui->pushButton_Rext->setEnabled(1);
ui->pushButton_VofsH_Vref->setEnabled(1);
ui->label_Vref_title->setEnabled(1);
ui->label_Rext_title->setEnabled(1);
ui->label_VofsH_title->setEnabled(1);
ui->label_VofsL_title->setEnabled(1);
ui->label_Rext_startValue->setEnabled(1);
ui->label_VofsH_startValue->setEnabled(1);
ui->label_Vref_value->setEnabled(1);
ui->label_VofsL_value->setEnabled(1);
ui->comboBox_Rext->setEnabled(1);
ui->comboBox_VofsH->setEnabled(1);
ui->comboBox_VofsL->setEnabled(1);
ui->comboBox_Vref->setEnabled(1);
ui->label_Rext_4->setEnabled(1);
ui->label_VofsH_4->setEnabled(1);
ui->label_Vref_4->setEnabled(1);
ui->label_VofsL_4->setEnabled(1);

}

}

void gpib::on_cb_linereg_curr_clicked()
{
    if (ui->cb_linereg_curr->isChecked())
    {
        ui->cb_linereg_vol->setChecked(0);
        ui->cb_start_ramp->setChecked(0);
    }
}
```

```
    ui->cb_test_mult->setChecked(0);
    ui->cb_loadreg->setChecked(0);
    ui->cb_stop_ramp->setChecked(0);
    ui->checkBox_sweep_Rext->setChecked(0);
    ui->checkBox_sweep_VofsH->setChecked(0);

//Sweep Felder aktivieren

ui->startLE->setEnabled(1);
ui->endLE->setEnabled(1);
ui->stepLE->setEnabled(1);
ui->cb_k2460->setEnabled(1);
ui->cb_k2400->setEnabled(1);
ui->label_9->setEnabled(1);
ui->label_29->setEnabled(1);
ui->label->setEnabled(1);
ui->label_2->setEnabled(1);
ui->label_3->setEnabled(1);

//Ramp Felder deaktivieren

ui->label_23->setDisabled(1);
ui->label_28->setDisabled(1);
ui->label_24->setDisabled(1);
ui->label_27->setDisabled(1);
ui->label_26->setDisabled(1);
ui->label_25->setDisabled(1);
ui->cb_ramp_current->setDisabled(1);
ui->cb_ramp_voltage->setDisabled(1);
ui->ramp_delay_le->setDisabled(1);
ui->ramp_step_le->setDisabled(1);
ui->ramp_start_le->setDisabled(1);
ui->ramp_end_le->setDisabled(1);

// VofsH Sweep Felder deaktivieren

ui->label_VofsH_endValue->setDisabled(1);
ui->comboBox_VofsH_end->setDisabled(1);
ui->label_VofsH_7->setDisabled(1);
ui->label_VofsH_schritte_end->setDisabled(1);
ui->hex_VofsH_end->setDisabled(1);
ui->label_VofsH_stepSize->setDisabled(1);
ui->spinBox_VofsH->setDisabled(1);

// Rext Sweep Felder deaktivieren

ui->label_Rext_endValue->setDisabled(1);
ui->comboBox_Rext_end->setDisabled(1);
```

```
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);

// andere Sachen aktivieren

ui->pushButton_Vofsl->setEnabled(1);
ui->pushButton_Rext->setEnabled(1);
ui->pushButton_Vofsh_Vref->setEnabled(1);
ui->label_Vref_title->setEnabled(1);
ui->label_Rext_title->setEnabled(1);
ui->label_Vofsh_title->setEnabled(1);
ui->label_Vofsl_title->setEnabled(1);
ui->label_Rext_startValue->setEnabled(1);
ui->label_Vofsh_startValue->setEnabled(1);
ui->label_Vref_value->setEnabled(1);
ui->label_Vofsl_value->setEnabled(1);
ui->comboBox_Rext->setEnabled(1);
ui->comboBox_Vofsh->setEnabled(1);
ui->comboBox_Vofsl->setEnabled(1);
ui->comboBox_Vref->setEnabled(1);
ui->label_Rext_4->setEnabled(1);
ui->label_Vofsh_4->setEnabled(1);
ui->label_Vref_4->setEnabled(1);
ui->label_Vofsl_4->setEnabled(1);

}
}

void gpib::on_cb_start_ramp_clicked()
{
    if (ui->cb_start_ramp->isChecked())
    {
        ui->cb_linereg_vol->setChecked(0);
        ui->cb_linereg_curr->setChecked(0);
        ui->cb_test_mult->setChecked(0);
        ui->cb_loadreg->setChecked(0);
        ui->cb_stop_ramp->setChecked(0);
        ui->checkBox_sweep_Rext->setChecked(0);
        ui->checkBox_sweep_Vofsh->setChecked(0);

        //Sweep Felder deaktivieren

        ui->startLE->setDisabled(1);
        ui->endLE->setDisabled(1);
        ui->stepLE->setDisabled(1);
    }
}
```

```
ui->cb_k2460->setDisabled(1);
ui->cb_k2400->setDisabled(1);
ui->label_9->setDisabled(1);
ui->label_29->setDisabled(1);
ui->label->setDisabled(1);
ui->label_2->setDisabled(1);
ui->label_3->setDisabled(1);

//Ramp Felder aktivieren

ui->label_23->setEnabled(1);
ui->label_28->setEnabled(1);
ui->label_24->setEnabled(1);
ui->label_27->setEnabled(1);
ui->label_26->setEnabled(1);
ui->label_25->setEnabled(1);
ui->cb_ramp_current->setEnabled(1);
ui->cb_ramp_voltage->setEnabled(1);
ui->ramp_delay_le->setEnabled(1);
ui->ramp_step_le->setEnabled(1);
ui->ramp_start_le->setEnabled(1);
ui->ramp_end_le->setEnabled(1);

// VofsH Sweep Felder deaktivieren

ui->label_VofsH_endValue->setDisabled(1);
ui->comboBox_VofsH_end->setDisabled(1);
ui->label_VofsH_7->setDisabled(1);
ui->label_VofsH_schritte_end->setDisabled(1);
ui->hex_VofsH_end->setDisabled(1);
ui->label_VofsH_stepSize->setDisabled(1);
ui->spinBox_VofsH->setDisabled(1);
ui->label_VofsH_endValue->setDisabled(1);
ui->comboBox_VofsH_end->setDisabled(1);
ui->label_VofsH_7->setDisabled(1);
ui->label_VofsH_schritte_end->setDisabled(1);
ui->hex_VofsH_end->setDisabled(1);
ui->label_VofsH_stepSize->setDisabled(1);
ui->spinBox_VofsH->setDisabled(1);

// Rext Sweep Felder deaktivieren

ui->label_Rext_endValue->setDisabled(1);
ui->comboBox_Rext_end->setDisabled(1);
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);
```

```
ui->label_Rext_endValue->setDisabled(1);
ui->comboBox_Rext_end->setDisabled(1);
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);

// andere Sachen aktivieren

ui->pushButton_Vofsl->setEnabled(1);
ui->pushButton_Rext->setEnabled(1);
ui->pushButton_Vofsh_Vref->setEnabled(1);
ui->label_Vref_title->setEnabled(1);
ui->label_Rext_title->setEnabled(1);
ui->label_Vofsh_title->setEnabled(1);
ui->label_Vofsl_title->setEnabled(1);
ui->label_Rext_startValue->setEnabled(1);
ui->label_Vofsh_startValue->setEnabled(1);
ui->label_Vref_value->setEnabled(1);
ui->label_Vofsl_value->setEnabled(1);
ui->comboBox_Rext->setEnabled(1);
ui->comboBox_Vofsh->setEnabled(1);
ui->comboBox_Vofsl->setEnabled(1);
ui->comboBox_Vref->setEnabled(1);
ui->label_Rext_4->setEnabled(1);
ui->label_Vofsh_4->setEnabled(1);
ui->label_Vref_4->setEnabled(1);
ui->label_Vofsl_4->setEnabled(1);

}
}

void gpib::on_cb_test_mult_clicked()
{
    if (ui->cb_test_mult->isChecked())
    {
        ui->cb_linereg_vol->setChecked(0);
        ui->cb_linereg_curr->setChecked(0);
        ui->cb_start_ramp->setChecked(0);
        ui->cb_loadreg->setChecked(0);
        ui->cb_stop_ramp->setChecked(0);
        ui->checkBox_sweep_Rext->setChecked(0);
        ui->checkBox_sweep_Vofsh->setChecked(0);

        //Sweep Felder deaktivieren

        ui->startLE->setDisabled(1);
        ui->endLE->setDisabled(1);
```

```
ui->stepLE->setDisabled(1);
ui->cb_k2460->setDisabled(1);
ui->cb_k2400->setDisabled(1);
ui->label_9->setDisabled(1);
ui->label_29->setDisabled(1);
ui->label->setDisabled(1);
ui->label_2->setDisabled(1);
ui->label_3->setDisabled(1);

//Ramp Felder deaktivieren

ui->label_23->setDisabled(1);
ui->label_28->setDisabled(1);
ui->label_24->setDisabled(1);
ui->label_27->setDisabled(1);
ui->label_26->setDisabled(1);
ui->label_25->setDisabled(1);
ui->cb_ramp_current->setDisabled(1);
ui->cb_ramp_voltage->setDisabled(1);
ui->ramp_delay_le->setDisabled(1);
ui->ramp_step_le->setDisabled(1);
ui->ramp_start_le->setDisabled(1);
ui->ramp_end_le->setDisabled(1);

// VofsH Sweep Felder deaktivieren

ui->label_VofsH_endValue->setDisabled(1);
ui->comboBox_VofsH_end->setDisabled(1);
ui->label_VofsH_7->setDisabled(1);
ui->label_VofsH_schritte_end->setDisabled(1);
ui->hex_VofsH_end->setDisabled(1);
ui->label_VofsH_stepSize->setDisabled(1);
ui->spinBox_VofsH->setDisabled(1);
ui->label_VofsH_title->setDisabled(1);
ui->pushButton_VofsH_Vref->setDisabled(1);

// Rext Sweep Felder deaktivieren

ui->label_Rext_endValue->setDisabled(1);
ui->comboBox_Rext_end->setDisabled(1);
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);
ui->label_Rext_title->setDisabled(1);
ui->pushButton_Rext->setDisabled(1);

// andere Sachen deaktivieren
```



```
    ui->pushButton_Vofsl->setDisabled(1);
    ui->label_Vref_title->setDisabled(1);
    ui->label_Vofsl_title->setDisabled(1);
    ui->label_Rext_startValue->setDisabled(1);
    ui->label_Vofsh_startValue->setDisabled(1);
    ui->label_Vref_value->setDisabled(1);
    ui->label_Vofsl_value->setDisabled(1);
    ui->comboBox_Rext->setDisabled(1);
    ui->comboBox_Vofsh->setDisabled(1);
    ui->comboBox_Vofsl->setDisabled(1);
    ui->comboBox_Vref->setDisabled(1);
    ui->label_Rext_4->setDisabled(1);
    ui->label_Vofsh_4->setDisabled(1);
    ui->label_Vref_4->setDisabled(1);
    ui->label_Vofsl_4->setDisabled(1);
}
}
```

```
void gpib::on_cb_loadreg_clicked()
{
    if (ui->cb_loadreg->isChecked())
    {
        ui->cb_linereg_vol->setChecked(0);
        ui->cb_linereg_curr->setChecked(0);
        ui->cb_start_ramp->setChecked(0);
        ui->cb_test_mult->setChecked(0);
        ui->cb_stop_ramp->setChecked(0);
        ui->checkBox_sweep_Rext->setChecked(0);
        ui->checkBox_sweep_Vofsh->setChecked(0);
```

//Sweep Felder aktivieren

```
    ui->startLE->setEnabled(1);
    ui->endLE->setEnabled(1);
    ui->stepLE->setEnabled(1);
    ui->cb_k2460->setEnabled(1);
    ui->cb_k2400->setEnabled(1);
    ui->label_9->setEnabled(1);
    ui->label_29->setEnabled(1);
    ui->label->setEnabled(1);
    ui->label_2->setEnabled(1);
    ui->label_3->setEnabled(1);
```

//Ramp Felder deaktivieren

```
    ui->label_23->setDisabled(1);
```

```
ui->label_28->setDisabled(1);
ui->label_24->setDisabled(1);
ui->label_27->setDisabled(1);
ui->label_26->setDisabled(1);
ui->label_25->setDisabled(1);
ui->cb_ramp_current->setDisabled(1);
ui->cb_ramp_voltage->setDisabled(1);
ui->ramp_delay_le->setDisabled(1);
ui->ramp_step_le->setDisabled(1);
ui->ramp_start_le->setDisabled(1);
ui->ramp_end_le->setDisabled(1);

// VofsH Sweep Felder deaktivieren

ui->label_VofsH_endValue->setDisabled(1);
ui->comboBox_VofsH_end->setDisabled(1);
ui->label_VofsH_7->setDisabled(1);
ui->label_VofsH_schritte_end->setDisabled(1);
ui->hex_VofsH_end->setDisabled(1);
ui->label_VofsH_stepSize->setDisabled(1);
ui->spinBox_VofsH->setDisabled(1);

// Rext Sweep Felder deaktivieren

ui->label_Rext_endValue->setDisabled(1);
ui->comboBox_Rext_end->setDisabled(1);
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);

// andere Sachen aktivieren

ui->pushButton_VofsL->setEnabled(1);
ui->pushButton_Rext->setEnabled(1);
ui->pushButton_VofsH_Vref->setEnabled(1);
ui->label_Vref_title->setEnabled(1);
ui->label_Rext_title->setEnabled(1);
ui->label_VofsH_title->setEnabled(1);
ui->label_VofsL_title->setEnabled(1);
ui->label_Rext_startValue->setEnabled(1);
ui->label_VofsH_startValue->setEnabled(1);
ui->label_Vref_value->setEnabled(1);
ui->label_VofsL_value->setEnabled(1);
ui->comboBox_Rext->setEnabled(1);
ui->comboBox_VofsH->setEnabled(1);
ui->comboBox_VofsL->setEnabled(1);
ui->comboBox_Vref->setEnabled(1);
```

```
    ui->label_Rext_4->setEnabled(1);
    ui->label_VofsH_4->setEnabled(1);
    ui->label_Vref_4->setEnabled(1);
    ui->label_Vofsl_4->setEnabled(1);

    }
}

void gpib::on_cb_stop_ramp_clicked()
{
    if (ui->cb_stop_ramp->isChecked())
    {
        ui->cb_linereg_vol->setChecked(0);
        ui->cb_linereg_curr->setChecked(0);
        ui->cb_test_mult->setChecked(0);
        ui->cb_loadreg->setChecked(0);
        ui->cb_start_ramp->setChecked(0);
        ui->checkBox_sweep_Rext->setChecked(0);
        ui->checkBox_sweep_VofsH->setChecked(0);

        //Sweep Felder deaktivieren

        ui->startLE->setDisabled(1);
        ui->endLE->setDisabled(1);
        ui->stepLE->setDisabled(1);
        ui->cb_k2460->setDisabled(1);
        ui->cb_k2400->setDisabled(1);
        ui->label_9->setDisabled(1);
        ui->label_29->setDisabled(1);
        ui->label->setDisabled(1);
        ui->label_2->setDisabled(1);
        ui->label_3->setDisabled(1);

        //Ramp Felder aktivieren

        ui->label_23->setEnabled(1);
        ui->label_28->setEnabled(1);
        ui->label_24->setEnabled(1);
        ui->label_27->setEnabled(1);
        ui->label_26->setEnabled(1);
        ui->label_25->setEnabled(1);
        ui->cb_ramp_current->setEnabled(1);
        ui->cb_ramp_voltage->setEnabled(1);
        ui->ramp_delay_le->setEnabled(1);
        ui->ramp_step_le->setEnabled(1);
        ui->ramp_start_le->setEnabled(1);
        ui->ramp_end_le->setEnabled(1);

        // VofsH Sweep Felder deaktivieren
```

```
ui->label_VofsH_endValue->setDisabled(1);
ui->comboBox_VofsH_end->setDisabled(1);
ui->label_VofsH_7->setDisabled(1);
ui->label_VofsH_schritte_end->setDisabled(1);
ui->hex_VofsH_end->setDisabled(1);
ui->label_VofsH_stepSize->setDisabled(1);
ui->spinBox_VofsH->setDisabled(1);
ui->label_VofsH_endValue->setDisabled(1);
ui->comboBox_VofsH_end->setDisabled(1);
ui->label_VofsH_7->setDisabled(1);
ui->label_VofsH_schritte_end->setDisabled(1);
ui->hex_VofsH_end->setDisabled(1);
ui->label_VofsH_stepSize->setDisabled(1);
ui->spinBox_VofsH->setDisabled(1);
```

```
// Rext Sweep Felder deaktivieren
```

```
ui->label_Rext_endValue->setDisabled(1);
ui->comboBox_Rext_end->setDisabled(1);
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);
ui->label_Rext_endValue->setDisabled(1);
ui->comboBox_Rext_end->setDisabled(1);
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);
```

```
// andere Sachen aktivieren
```

```
ui->pushButton_VofsL->setEnabled(1);
ui->pushButton_Rext->setEnabled(1);
ui->pushButton_VofsH_Vref->setEnabled(1);
ui->label_Vref_title->setEnabled(1);
ui->label_Rext_title->setEnabled(1);
ui->label_VofsH_title->setEnabled(1);
ui->label_VofsL_title->setEnabled(1);
ui->label_Rext_startValue->setEnabled(1);
ui->label_VofsH_startValue->setEnabled(1);
ui->label_Vref_value->setEnabled(1);
ui->label_VofsL_value->setEnabled(1);
ui->comboBox_Rext->setEnabled(1);
ui->comboBox_VofsH->setEnabled(1);
ui->comboBox_VofsL->setEnabled(1);
```

```
    ui->comboBox_Vref->setEnabled(1);
    ui->label_Rext_4->setEnabled(1);
    ui->label_VofsH_4->setEnabled(1);
    ui->label_Vref_4->setEnabled(1);
    ui->label_VofsL_4->setEnabled(1);

}

}

void gpib::on_checkBox_sweep_Rext_clicked() // Banko :
{
    if (ui->checkBox_sweep_Rext->isChecked())
    {

        ui->cb_linereg_vol->setChecked(0);
        ui->cb_linereg_curr->setChecked(0);
        ui->cb_test_mult->setChecked(0);
        ui->cb_loadreg->setChecked(0);
        ui->cb_start_ramp->setChecked(0);
        ui->cb_stop_ramp->setChecked(0);
        ui->checkBox_sweep_VofsH->setChecked(0);
        ui->label_Rext_endValue->setEnabled(1);
        ui->comboBox_Rext_end->setEnabled(1);
        ui->label_Rext_6->setEnabled(1);
        ui->label_Rext_schritte_end->setEnabled(1);
        ui->hex_Rext_end->setEnabled(1);
        ui->label_Rext_stepSize->setEnabled(1);
        ui->spinBox_Rext->setEnabled(1);
        this->on_comboBox_Rext_end_currentIndexChanged("");

        // VofsH Sweep Felder deaktivieren

        ui->label_VofsH_endValue->setDisabled(1);
        ui->comboBox_VofsH_end->setDisabled(1);
        ui->label_VofsH_7->setDisabled(1);
        ui->label_VofsH_schritte_end->setDisabled(1);
        ui->hex_VofsH_end->setDisabled(1);
        ui->label_VofsH_stepSize->setDisabled(1);
        ui->spinBox_VofsH->setDisabled(1);

        //Ramp Felder aktivieren

        ui->label_23->setDisabled(1);
        ui->label_28->setDisabled(1);
        ui->label_24->setDisabled(1);
        ui->label_27->setDisabled(1);
        ui->label_26->setDisabled(1);
        ui->cb_ramp_current->setDisabled(1);
```

```
    ui->cb_ramp_voltage->setDisabled(1);
    ui->ramp_delay_le->setDisabled(1);
    ui->ramp_step_le->setDisabled(1);
    ui->ramp_start_le->setDisabled(1);
    ui->ramp_end_le->setDisabled(1);

    //Sweep Felder aktivieren

    ui->startLE->setEnabled(1);
    ui->endLE->setEnabled(1);
    ui->stepLE->setEnabled(1);
    ui->cb_k2460->setEnabled(1);
    ui->cb_k2400->setEnabled(1);
    ui->label_9->setEnabled(1);
    ui->label_29->setEnabled(1);
    ui->label->setEnabled(1);
    ui->label_2->setEnabled(1);
    ui->label_3->setEnabled(1);

    // andere Sachen aktivieren

    ui->pushButton_Vofsl->setEnabled(1);
    ui->pushButton_Rext->setEnabled(1);
    ui->pushButton_Vofsh_Vref->setEnabled(1);
    ui->label_Vref_title->setEnabled(1);
    ui->label_Rext_title->setEnabled(1);
    ui->label_Vofsh_title->setEnabled(1);
    ui->label_Vofsl_title->setEnabled(1);
    ui->label_Rext_startValue->setEnabled(1);
    ui->label_Vofsh_startValue->setEnabled(1);
    ui->label_Vref_value->setEnabled(1);
    ui->label_Vofsl_value->setEnabled(1);
    ui->comboBox_Rext->setEnabled(1);
    ui->comboBox_Vofsh->setEnabled(1);
    ui->comboBox_Vofsl->setEnabled(1);
    ui->comboBox_Vref->setEnabled(1);
    ui->label_Rext_4->setEnabled(1);
    ui->label_Vofsh_4->setEnabled(1);
    ui->label_Vref_4->setEnabled(1);
    ui->label_Vofsl_4->setEnabled(1);

}

else
{
    ui->label_Rext_endValue->setEnabled(0);
    ui->comboBox_Rext_end->setEnabled(0);
    ui->label_Rext_6->setEnabled(0);
```

```
    ui->label_Rext_schritte_end->setEnabled(0);
    ui->hex_Rext_end->setEnabled(0);
    ui->label_Rext_stepSize->setEnabled(0);
    ui->spinBox_Rext->setEnabled(0);
}

}

void gpib::on_checkBox_sweep_VofsH_clicked() // Banko :
{
    if (ui->checkBox_sweep_VofsH->isChecked())
    {

        ui->cb_linereg_vol->setChecked(0);
        ui->cb_linereg_curr->setChecked(0);
        ui->cb_test_mult->setChecked(0);
        ui->cb_loadreg->setChecked(0);
        ui->cb_start_ramp->setChecked(0);
        ui->cb_stop_ramp->setChecked(0);
        ui->checkBox_sweep_Rext->setChecked(0);

        // VofsH Sweep Felder aktivieren

        ui->label_VofsH_endValue->setEnabled(1);
        ui->comboBox_VofsH_end->setEnabled(1);
        ui->label_VofsH_7->setEnabled(1);
        ui->label_VofsH_schritte_end->setEnabled(1);
        ui->hex_VofsH_end->setEnabled(1);
        ui->label_VofsH_stepSize->setEnabled(1);
        ui->spinBox_VofsH->setEnabled(1);
        this->on_comboBox_VofsH_end_currentIndexChanged("");

        //Ramp Felder deaktivieren

        ui->label_23->setDisabled(1);
        ui->label_28->setDisabled(1);
        ui->label_24->setDisabled(1);
        ui->label_27->setDisabled(1);
        ui->label_26->setDisabled(1);
        ui->cb_ramp_current->setDisabled(1);
        ui->cb_ramp_voltage->setDisabled(1);
        ui->ramp_delay_le->setDisabled(1);
        ui->ramp_step_le->setDisabled(1);
        ui->ramp_start_le->setDisabled(1);
        ui->ramp_end_le->setDisabled(1);

        // Rext Sweep Felder deaktivieren

        ui->label_Rext_endValue->setDisabled(1);
```

```
ui->comboBox_Rext_end->setDisabled(1);
ui->label_Rext_6->setDisabled(1);
ui->label_Rext_schritte_end->setDisabled(1);
ui->hex_Rext_end->setDisabled(1);
ui->label_Rext_stepSize->setDisabled(1);
ui->spinBox_Rext->setDisabled(1);
```

```
//Sweep Felder aktivieren
```

```
ui->startLE->setEnabled(1);
ui->endLE->setEnabled(1);
ui->stepLE->setEnabled(1);
ui->cb_k2460->setEnabled(1);
ui->cb_k2400->setEnabled(1);
ui->label_9->setEnabled(1);
ui->label_29->setEnabled(1);
ui->label->setEnabled(1);
ui->label_2->setEnabled(1);
ui->label_3->setEnabled(1);
```

```
// andere Sachen aktivieren
```

```
ui->pushButton_Vofsl->setEnabled(1);
ui->pushButton_Rext->setEnabled(1);
ui->pushButton_Vofsh_Vref->setEnabled(1);
ui->label_Vref_title->setEnabled(1);
ui->label_Rext_title->setEnabled(1);
ui->label_Vofsh_title->setEnabled(1);
ui->label_Vofsl_title->setEnabled(1);
ui->label_Rext_startValue->setEnabled(1);
ui->label_Vofsh_startValue->setEnabled(1);
ui->label_Vref_value->setEnabled(1);
ui->label_Vofsl_value->setEnabled(1);
ui->comboBox_Rext->setEnabled(1);
ui->comboBox_Vofsh->setEnabled(1);
ui->comboBox_Vofsl->setEnabled(1);
ui->comboBox_Vref->setEnabled(1);
ui->label_Rext_4->setEnabled(1);
ui->label_Vofsh_4->setEnabled(1);
ui->label_Vref_4->setEnabled(1);
ui->label_Vofsl_4->setEnabled(1);
```

```
}
```

```
else
```

```
{
```

```
ui->label_Vofsh_endValue->setEnabled(0);
ui->comboBox_Vofsh_end->setEnabled(0);
```



```
        ui->label_VofsH_7->setEnabled(0);
        ui->label_VofsH_schritte_end->setEnabled(0);
        ui->hex_VofsH_end->setEnabled(0);
        ui->label_VofsH_stepSize->setEnabled(0);
        ui->spinBox_VofsH->setEnabled(0);
    }
}

// Banko : AB HIER CLICK DER BUTTONS ZUM ÜBERGEBEN DER SCHRITTWEITE EINZELN
// .....
// .....

void gpib::on_pushButton_Rext_clicked()
// bei Betätigung des Buttons wird die nachfolgende Funktion ausgeführt
{
    int ohmwert_Rext_inHex;
// integer Variablen für die Slave Adresse und den OhmWert in HEX
    QString ohmwert_Rext;
// String für das umwandeln des ohmschen Wertes

    ohmwert_Rext = ui->hex_Rext-
>text(); // in den String den Text des Feldes hex
_Rext laden
    ohmwert_Rext_inHex = ohmwert_Rext.toInt(&ok,16);
// umwandeln in INT
    std::cout<<std::hex<<"HEX: " <<ohmwert_Rext_inHex<<std::endl;

    z-
>widerstandswert_poti_uebergabe_Rext(ohmwert_Rext_inHex); // Übergabe
des Wertes
}

void gpib::on_pushButton_VofsH_Vref_clicked()
// bei Betätigung des Buttons wird die nachfolgende Funktion ausgeführt
{
    int ohmwert_VofsH_inHex, ohmwert_Vref_inHex;
// integer Variablen für die Slave Adresse und den OhmWert in HEX
    QString ohmwert_VofsH, ohmwert_Vref;
// String für das umwandeln des ohmschen Wertes

    ohmwert_VofsH = ui->hex_VofsH-
>text(); // in den String den Text des Feldes
hex_VofsH laden
    ohmwert_VofsH_inHex = ohmwert_VofsH.toInt(&ok,16);
// umwandeln in INT
```

```
    ohmwert_Vref = ui->hex_Vref-
>text(); // in den String den Text des Feldes hex
_Vref laden
    ohmwert_Vref_inHex = ohmwert_Vref.toInt(&ok,16);
// umwandeln in INT

    z-
>widerstandswert_poti_uebergabe_VofsH_Vref(ohmwert_VofsH_inHex, ohmwert_Vref
_inHex); // Übergabe der Werte
}

void gpib::on_pushButton_VofsL_clicked()
// bei Betätigung des Buttons wird die nachfolgende Funktion ausgeführt
{
    int ohmwert_VofsL_inHex;
// integer Variablen für die Slave Adresse und den OhmWert in HEX
    QString ohmwert_VofsL;
// String für das umwandeln des ohmschen Wertes

    ohmwert_VofsL = ui->hex_VofsL-
>text(); // in den String den Text des Feldes hex_Rext
laden
    ohmwert_VofsL_inHex = ohmwert_VofsL.toInt(&ok,16);
// umwandeln in INT

    z->widerstandswert_poti_uebergabe_VofsL(ohmwert_VofsL_inHex);
// Übergabe des Wertes
}

// Banko : AB HIER INDEX WECHSEL ::::::::::::::::::::::::::::::::::::::::::::
::::::::::::::::::::::::::::::::::::::::::::::::::
::::::::::::

void gpib::on_comboBox_Rext_currentIndexChanged(const QString &arg1)
{
    int index_Rext, schrittweisen_subtraktion_Rext, schrittweite_ende;
// Variable für die Schrittweite
    char hexa_Rext[3];
// Variable für das umwandeln in HEX

    index_Rext = ui->comboBox_Rext-
>currentIndex(); // in die Variable a wird der
aktuelle Index geladen : Index = Schrittweite
```

```
    ui->label_Rext_schritte->setNum(index_Rext);
// in das label wird die ausgewählte Schrittweite als Zahl gesetzt

    sprintf(hexa_Rext, "%x", index_Rext);
// umwandeln der Schrittweite in HEX

    ui->hex_Rext->setText(hexa_Rext);
// HEX Wert der Variable ins label setzen

    schrittweite_ende = ui->comboBox_Rext_end-
>currentIndex();          // Schrittweite Endwert laden

    schrittweiten_subtraktion_Rext = schrittweite_ende - index_Rext;
// Subtraktion : Endwert - Startwert

if (ui->checkBox_sweep_Rext->isChecked())
// wenn Sweep Rext ausgewählt ist, ...
{
    if (schrittweiten_subtraktion_Rext <= 0)
// Schrittweiten Subtraktion kleiner gleich null ?
    {
        ui->comboBox_Rext->setCurrentIndex(schrittweite_ende-
1);          // setze den Indexwert auf Endwert -
1, damit kein negativer Wert möglich ist
    }

    else
// wenn nicht, ...
    {
        ui->spinBox_Rext-
>setMaximum((schrittweiten_subtraktion_Rext*4));
// setze das Maximum der SpinBox auf den Wert der Subtraktion
    }
}

    ui->spinBox_Rext->setSingleStep(4);
// 4 Ohm pro Schrittweite

    ui->spinBox_Rext->setMinimum(4);
// Minimum 4 Ohm
}
```

```
void gpib::on_comboBox_Rext_end_currentIndexChanged(const QString &arg1)
{

    char hexa_Rext_End[3];
        // Variable für das umwandeln in HEX
    int schrittweite_start_int, schrittweiten_subtraktion_Rext, index_Rext_End;
    // Variable für die Schrittweite
    QString schrittweite_start, schrittweite_ende;

    index_Rext_End = ui->comboBox_Rext_end-
>currentIndex(); // in die Variable a wird der aktuelle
    Index geladen : Index = Schrittweite

    ui->label_Rext_schritte_end-
>setNum(index_Rext_End); // in das label wird die
    ausgewählte Schrittweite als Zahl gesetzt

    sprintf(hexa_Rext_End, "%x", index_Rext_End);
        // umwandeln der Schrittweite in HEX
        // hierdurch kann man sich den Wert auf der Console ausgeben lassen

    ui->hex_Rext_end-
>setText(hexa_Rext_End);
// setzen des HEX Wertes ins Label

    schrittweite_start = ui->label_Rext_schritte-
>text(); // laden der Schrittweite

    schrittweite_start_int = schrittweite_start.toInt();
        // umwandeln in int

    schrittweiten_subtraktion_Rext = index_Rext_End - schrittweite_start_int
; //Endwert - Startwert = 5 - 4 = 1

    if (schrittweiten_subtraktion_Rext <= 0)
        // if Abfrage : Schrittweitendifferenz kleiner gleich null ?
    {
        ui->comboBox_Rext_end-
>setCurrentIndex(schrittweite_start_int+1); // dann Endwert auf
        Startwert + 1 setzen, damit kein negativer Wert erreicht werden kann
    }

    else
        // sonst
    {
```

```
        ui->spinBox_Rext-
>setMaximum((schrittweiten_subtraktion_Rext*4));
// Maximum gleich Schrittweite
    }

    ui->spinBox_Rext-
>setSingleStep(4);
// Schrittweite 4 Ohm pro Step

    ui->spinBox_Rext-
>setMinimum(4);
// Minimum auf 4 Ohm setzen

}

void gpib::on_comboBox_VofsH_currentIndexChanged(const QString &arg1)
{

    char hexa_VofsH[3];
        // Variable für das umwandeln in HEX
    int index_VofsH, schrittweiten_subtraktion_VofsH, schrittweite_ende;

    index_VofsH = ui->comboBox_VofsH-
>currentIndex(); // in die Variable a wird der
aktuelle Index geladen : Index = Schrittweite

    ui->label_VofsH_schritte-
>setNum(index_VofsH); // in das label wird
die ausgewählte Schrittweite als Zahl gesetzt

    sprintf(hexa_VofsH, "%x", index_VofsH);
        // umwandeln der Schrittweite in HEX

    ui->hex_VofsH-
>setText(hexa_VofsH);
// HEX Wert der Variable ins label setzen

    schrittweite_ende = ui->comboBox_VofsH_end-
>currentIndex(); // Schrittweite Endwert setzen

    schrittweiten_subtraktion_VofsH = schrittweite_ende - index_VofsH;
        // Endwert minus Startwert
```

```
if (ui->checkBox_sweep_VofsH-
>isChecked())
// wenn Sweep VofsH gewählt, ...
{
    if (schrittweiten_subtraktion_VofsH <= 0)
        // wenn Schrittweite VofsH kleiner gleich null ?, ...
        {
            ui->comboBox_VofsH->setCurrentIndex(schrittweite_ende-
1); // setze Box gleich Endwert - 1, damit kein negativer
Wert erreicht werden kann
        }

    else
        // sonst, ...
        {
            ui->spinBox_VofsH-
>setMaximum((schrittweiten_subtraktion_VofsH*390));
// setze Maximum = Schrittweite
        }
}

ui->spinBox_VofsH-
>setSingleStep(390);
// setze 390 Ohm pro Schritt

ui->spinBox_VofsH-
>setMinimum(390);
// setze Minimum = 390 Ohm
}

void gpib::on_comboBox_VofsH_end_currentIndexChanged(const QString &arg1)
{
    int schrittweite_start_int, schrittweiten_subtraktion_VofsH_end, index_V
ofsH_end; // Variable für die Schrittweite
    char hexa_VofsH_end[3]; // Variable für das umwandeln in HEX
    QString schrittweite_start, schrittweite_ende;

    index_VofsH_end = ui->comboBox_VofsH_end-
>currentIndex();
// in die Variable a wird der aktuelle Index geladen : Index = Schrittweite

    ui->label_VofsH_schritte_end-
>setNum(index_VofsH_end);
// in das label wird die ausgewählte Schrittweite als Zahl gesetzt
```

```
sprintf(hexa_VofsH_end, "%x", index_VofsH_end);
        // umwandeln der Schrittweite in HEX

    ui->hex_VofsH_end-
>setText(hexa_VofsH_end);
    // HEX Wert der Variable ins label setzen

    schrittweite_start = ui->label_VofsH_schritte-
>text();
// Schrittweite Start laden

    schrittweite_start_int = schrittweite_start.toInt();
        // in Int wandeln

    schrittweiten_subtraktion_VofsH_end = index_VofsH_end - schrittweite_start_int;
        // Endwert - Startwert

    if (schrittweiten_subtraktion_VofsH_end <= 0)
        // wenn Schrittweite kleiner gleich null, dann
    {
        ui->comboBox_VofsH_end-
>setCurrentIndex(schrittweite_start_int+1);
// setze Box = Startwert +1, damit kein negativer Wert erreicht werden kann
    }

    else
        // sonst
    {
        ui->spinBox_VofsH-
>setMaximum((schrittweiten_subtraktion_VofsH_end*390));
        // setze Maximum = Schrittweite
    }

    ui->spinBox_VofsH-
>setSingleStep(390);
    // setze 390 Ohm pro Schritt

    ui->spinBox_VofsH-
>setMinimum(390);
    // setze Minimum = 390 Ohm
}

void gpib::on_comboBox_Vref_currentIndexChanged(const QString &arg1)
{
```

```
int a;
    // Variable für die Schrittweite
char hexa_Vref[3];
    // Variable für das umwandeln in HEX

a = ui->comboBox_Vref-
>currentIndex(); // in die Variable
a wird der aktuelle Index geladen : Index = Schrittweite

ui->label_Vref_schritte-
>setNum(a); // in das label wird
die ausgewählte Schrittweite als Zahl gesetzt

sprintf(hexa_Vref, "%x", a);
    // umwandeln der Schrittweite in HEX

ui->hex_Vref-
>setText(hexa_Vref);
// HEX Wert der Variable ins label setzen
}

void gpib::on_comboBox_VofsL_currentIndexChanged(const QString &arg1)
{

int a;
    // Variable für die Schrittweite
char hexa_VofsL[3];
    // Variable für das umwandeln in HEX

a = ui->comboBox_VofsL-
>currentIndex(); // in die Variable
a wird der aktuelle Index geladen : Index = Schrittweite

ui->label_VofsL_schritte-
>setNum(a); // in das label wird
die ausgewählte Schrittweite als Zahl gesetzt

sprintf(hexa_VofsL, "%x", a);
    // umwandeln der Schrittweite in HEX

ui->hex_VofsL-
>setText(hexa_VofsL);
// HEX Wert der Variable ins label setzen
}
```



#### 9.4.4 - Header „sldo65.h“

```
#ifndef SLD065_H
#define SLD065_H

#include <vector>
#include <iostream>
#include <string>
#include "Silib_GPIB_Keithley2000.h"
#include "Silib_GPIB_Keithley246x.h"
#include "Silib_GPIB_Keithley24xx.h"
#include "SILIB_GPIB_Interfaces.h"
#include "SILIB_GPIB_Device.h"
#include "QString"
#include "qwt_plot_curve.h"
#include "QTextEdit"

//Neueingebundene Klassen für die Multimeter DMM6500
#include "fhdo_usb_interface.h"
#include "fhdo_usb_device.h"
#include "fhdo_usb_keithleydmm6500.h"

#include "digitalepotis.h"

class sldo65 : public digitalepotis
{
public:
    sldo65 ();
    ~sldo65();
    /*$PUBLIC_FUNCTIONS$*/

    //Karl Macha
    double testvoltage();
    double testvoltage2();
    double testvoltage3();
    double testvoltage4();
    double testvoltage5();
    double testvoltage6();
    double testvoltage7();

    std::vector<double> lineregv01(double start,double stop,double step,
double limit);
    std::vector<double> lineregv012460(double start,double stop,double s
tep, double limit);

```

```
        std::vector<double> lineregcur(double start,double stop,double step,
double limit);
        std::vector<double> lineregcur2460(double start,double stop,double s
tep, double limit);
        std::vector<double> loadreg(double start,double stop,double step, do
uble limit);
        std::vector<double> loadreg2460(double start,double stop,double step
, double limit);
        double VSweep(double delay, double steps, double start, double end,
double limit, double vrange, double crange);
        double CSweep(double delay, double steps, double start, double end,
double limit, double vrange, double crange);
        double StopSweep();

        double vec1(double a);
        double vec2 (double a);
        // Karl Macha Ende

//    virtual void changeMode();

protected:
    /*$PROTECTED_FUNCTIONS$*/
    void *hUSB;
    TGPIB_Interface_USB *gpib;
    TGPIB_Keithley2000 *keith;

    //Neue Geräte USB als Interface
    USB_Interface *usb;
    KeithleyDMM6500 *keith2;
    KeithleyDMM6500 *keith3;
    KeithleyDMM6500 *keith4;
    KeithleyDMM6500 *keith5;
    KeithleyDMM6500 *keith6;
    KeithleyDMM6500 *keith7;
    TGPIB_Keithley240x *source;
    TGPIB_Keithley246X *source2;
    TGPIB_Keithley240x *source3;

};

#endif // SLD065_H
```