



MASTERARBEIT

Design, Aufbau und Programmierung der Regelelektronik für einen Klimaschrank

vorgelegt von

Jürgen Gasparitz, BSc

am

Institut für Elektronik

Vorstand: Univ.-Prof. Dipl.-Ing. Dr.techn. Wolfgang Pribyl

Technische Universität Graz

Betreuer:

Ass.Prof. Dipl.-Ing. Dr.techn. Gunter Winkler

Graz, im November 2010

Danksagung

Ich möchte mich an dieser Stelle besonders bei meinem Betreuer **Ass.Prof. Dipl.-Ing. Dr.techn. Gunter Winkler** bedanken, der mir mit seinem umfassenden Fachwissen immer mit Rat und Tat zur Seite stand. Für viele hilfreiche Tipps bezüglich der praktischen Durchführung meiner Masterarbeit sei herzlichst Herrn **Dipl.-Ing. Dr.techn. Harald Hartl** gedankt. Weiters möchte ich mich stellvertretend bei Herrn **Eduard Dorner** von der Werkstätte des Instituts für Elektronik bedanken, der mich mit seinen Mitarbeitern bei der Fertigung der Elektronik bzw. den mechanischen Baugruppen des Klimaschranks unterstützte.

Außerdem möchte ich mich bei meiner Familie bedanken, ohne deren Unterstützung ich mich wohl nie für ein Studium entschieden hätte. Mein ganz besonderer Dank richtet sich aber an meine Freundin, die mich während des Studiums moralisch unterstützte und im Zuge der Masterarbeit bei vielen Formulierungen und auch bei der Korrektur sehr hilfreich zur Seite stand.

Zum Abschluss möchte ich nicht darauf vergessen, mich bei meinen langjährigen Studienkollegen zu bedanken, ohne diese die erfolgreiche Absolvierung dieses Studiums nicht möglich gewesen wäre.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, am 16. November 2010

Kurzfassung

Diese Masterarbeit dokumentiert den Umbau eines gewöhnlichen Weinkühlschranks zu einem Klimaschrank, der in seiner Anwendung elektronische Schaltungen auf deren Temperaturabhängigkeit überprüft. Der Klimaschrank, der nach dem elektrothermischen Prinzip arbeitet, wurde mit leistungsfähigeren elektronischen Komponenten ausgestattet, wodurch die bestehenden mechnischen Baugruppen angepasst werden mussten. Die analoge Regelelektronik wurde durch ein Mikrocontrollersystem ersetzt, wodurch eine digitale Regelung samt adaptiver Reglereinstellung ermöglicht wurde. Durch die Anbindung eines Displays und eines Navigationskits, konnte ein Bedienelement zur Steuerung des Klimaschranks realisiert werden. Für die Temperaturmessung wurde ein Pt1000-Messwiderstand und ein digitaler Feuchte- und Temperatursensor SHT15 an den Mikrocontroller angebunden. Im Falle des Pt1000-Messwiderstandes wurde ein spezielles Messverfahren, bestehend aus Präzisionsstromspiegel und Ladungsausgleichsintegrator angewandt, welches am Institut für Elektronik entworfen wurde. Um auch eine softwaremäßige Fernsteuerung zu ermöglichen (z. B. für Temperaturläufe), wurde ein Protokoll auf Basis der seriellen Datenübertragung implementiert, welches die Übertragung aktueller Temperaturwerte sowie die Definition neuer Solltemperturen ermöglichte.

Abstract

This master thesis describes the rebuilding of a normal refrigerator into a climate box so that the temperature dependencies of electronic circuits can be analyzed. The climate box, which operates according to the principle of thermoelectric cooling, is fitted with powerful electronic devices. As a result, mechanical components had to be changed. The analog control circuit was replaced by a microcontroller system, which enables the digital control and adaptive adjustment. By connecting the microcontroller to a display and a navigation kit it was possible to telecontrol the climate box. To measure the temperature a Pt1000 measuring resistor and an SHT15 digital temperature and humidity sensor were used. In the case of the Pt1000 resistor a specialized measurement procedure was adopted, which required the combination of a precision current mirror and a charge-balance ADC. This specialized measurement procedure was invented at the Institute of Electronics at Graz University of Technology. In order to implement a software-based remote control (e.g. for temperature measurement procedures), a serial connection based protocol was designed, which allows transmitting the current temperature values as well as defining new nominal temperatures.

Inhaltsverzeichnis

Danksagung	i
Eidesstattliche Erklärung	ii
Kurzfassung	iii
Abstract	iv
1. Ziele	1
1.1. Istzustand	2
1.2. Sollzustand	3
2. Klimaschrank	5
2.1. Definitionen	5
2.2. Physikalische Effekte	6
2.2.1. Verdunstungskühlung	6
2.2.2. Magnetische Kühlung	6
2.2.3. Joule-Thomson-Effekt	6
2.2.4. Thermoelektrischer Effekt	7
2.3. Elektrothermische Kühlung	7
2.3.1. Seebeck-Effekt	7
2.3.2. Peltier-Effekt	8
2.4. Peltier-Element	9
2.4.1. Aufbau	9
2.4.2. Eigenschaften	10
2.4.3. Verwendetes Modell	10
2.5. Peltier-Verstärker	10
2.6. Kühlverfahren	13
2.7. Temperaturmessung	15
2.7.1. Pt1000	15
2.7.2. SHT15	16

3. Hardwareentwurf	17
3.1. Mechanische Komponenten	17
3.1.1. Peltier-Lüfter-Stack	18
3.1.2. Sensorpositionierung	19
3.1.3. Aufbau Peltier-Verstärker/Versorgungsplatine	20
3.1.4. Rückwandabdeckung	21
3.1.5. Kommunikationsport	21
3.1.6. Bedienteil	22
3.2. Elektronische Komponenten	23
3.2.1. Spannungsversorgung	23
3.2.2. Mikrocontroller und -beschaltung	26
3.2.3. Serielle Datenübertragung	27
3.2.4. MOSFET-Schalter	29
3.2.5. Messschaltung	29
3.2.6. Displayansteuerung	29
3.2.7. Navigationskit	31
4. Digitale Regelung	33
4.1. Einleitung	33
4.2. Standardregler	34
4.2.1. P-Regler	34
4.2.2. I-Regler	34
4.2.3. PI-Regler	35
4.2.4. PID-Regler	35
4.2.5. Vergleich der Reglertypen	36
4.3. Einstellregeln für Standardregler	36
4.3.1. Open-Loop-Method	37
4.3.2. Closed-Loop-Method	39
4.3.3. Zweipunktregler	40
4.3.4. Beurteilung	41
4.4. Adaptive und selbsteinstellende Regler (Autotuning)	41
4.5. Reglerüberlauf (Windup-Effekt)	42
4.6. Gewähltes Konzept	43
4.6.1. Digitales Reglermodell	44
4.6.2. Reglerauswahl	45
4.6.3. Reglerdimensionierung	45
4.6.4. Autotuning	46

5. Messungen und Berechnungen	47
5.1. Zweipunktregelung	47
5.2. Berechnung der Verstärkungsfaktoren	50
5.3. Sollwertsprünge	51
6. Firmware	53
6.1. Einleitung	53
6.2. Hauptprogramm	54
6.2.1. State-Machine	54
6.2.2. Unterprogramme	56
6.2.3. Messungen bzw. Berechnungen	57
6.3. Implementierung	57
6.3.1. Timer B	58
6.3.2. Digitaler Regler	60
6.3.3. Autotuning	62
6.3.4. Serielle Datenübertragung	63
6.3.5. Fernsteuerung	65
6.3.6. Flash-Speicherung	66
7. Ausblick	69
7.1. Klimaschrank/Peltier-Lüfter-Stack	69
7.2. Display	70
7.3. MOSFET-Schalter	70
7.4. Kaskadenregelung	70
7.5. Bumpless Transfer	70
7.6. Modellbildung	71
A. Anhang	80
A.1. Entwicklungsumgebung	80
A.2. Serielle Datenübertragung	80
A.2.1. RS232	80
A.2.2. FT232RL	81
A.3. Remote Control	82
A.4. Alte Regelelektronik	83
A.5. Schaltpläne	85
A.5.1. Messschaltung	85
A.5.2. Mikrocontroller und -beschaltung	86

Inhaltsverzeichnis

A.5.3. Serielle Datenübertragung	87
A.6. Layouts	88
A.6.1. Versorgung	88
A.6.2. Mikrocontrollerplatine	89
A.6.3. Peripherieplatine	90
A.7. Anbaupläne	91
A.7.1. Aufbau Peltier-Lüfter-Stack	91
A.7.2. Bodenplatte Peltier-Verstärker	92
A.7.3. Aufbau Versorgungsplatine	93
A.7.4. Bedienteildeckel	94
A.7.5. Abdeckung Bedienteildeckel	95
A.7.6. Bedienteil Seitenteile	96
A.7.7. Rückwandabdeckung	97
A.8. Code-Dokumentation	98
A.9. Bauteilliste	98
A.10. Datenblätter	98
A.11. Datenträger	98

1. Ziele

Ziel dieser Masterarbeit ist es, einen gewöhnlichen handelsüblichen Weinkühlschrank zu einem Klimaschrank mit einer digitalen Temperaturregelung umzubauen.

Bevor in den folgenden Abschnitten durch eine Gegenüberstellung des Ist- bzw. Sollzustandes ein Überblick über die Anforderungen an diese Masterarbeit gegeben wird, soll das Zusammenspiel der verwendeten Komponenten durch eine Systemübersicht (siehe Abbildung 1.1) dargestellt werden.

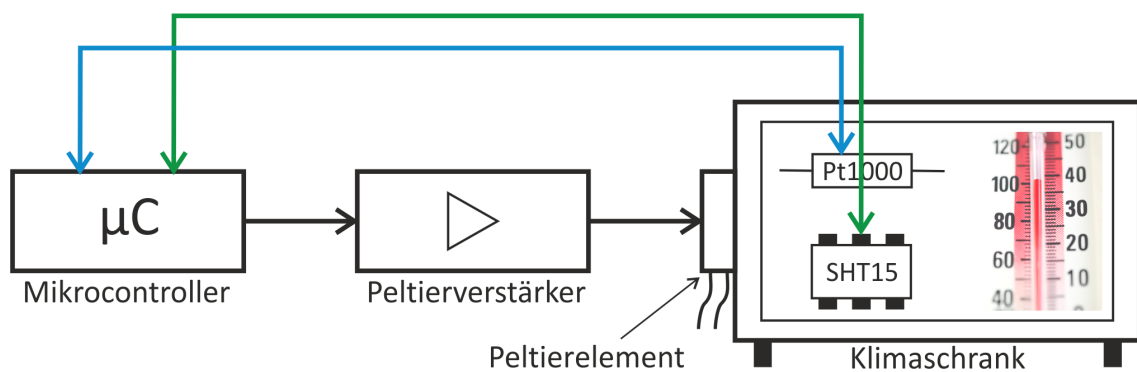


Abbildung 1.1.: Systemübersicht

Der Klimaschrank arbeitet prinzipiell mit einem elektrothermischen Kühlsystem (Peltier-Element), welches von einem Peltier-Verstärker versorgt wird. Dieser wiederum soll von meinem Mikrocontrollersystem angesteuert werden, welches neben der Kommunikation mit dem beiden Temperatursensoren (Pt1000, SHT15) die digitale Temperaturregelung ausführt.

1. Ziele

1.1. Istzustand

Der zu verwendende Weinkühlschrank (siehe Abbildung 1.2) weist laut Datenblatt (vgl. [Eur10]) folgende Spezifikationen auf:

- **Inhalt:** maximal 40 Liter
- **Modellnummer:** WICL40HBL
- **Hersteller:** Euro-Asia Electric Manufacturing (Holdings) Ltd.
- **Herstellernummer:** EA23-40H
- **Kühlsystem:** elektrothermisches Kühlsystem (Peltier-Element)
- **Stromaufnahme:** maximal 70W
- **Temperaturbereich:** maximal 10°C unter Umgebungstemperatur
- **Abmessungen:** 410mm x 460mm x 540mm (Höhe x Breite x Tiefe)



Abbildung 1.2.: Weinkühlschrank EA23-40H ©

1. Ziele

Die elektronischen Komponenten des Weinkühlschranks setzen sich aus einer **Verorgungselektronik** und einer **Regelungselektronik** zusammen. Ein Blockschaltbild dieser beiden Komponenten ist in Abbildung 1.3 dargestellt.

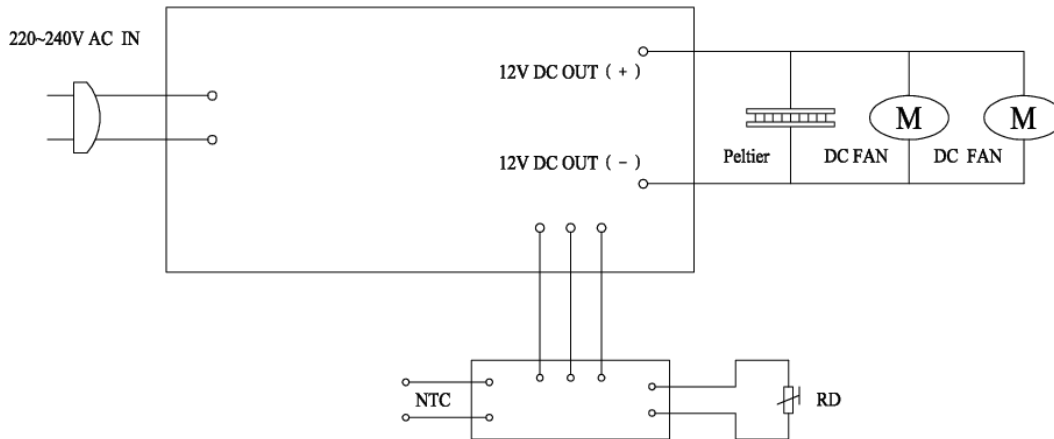


Abbildung 1.3.: Blockdiagramm Elektronik ©

Die analoge Regelelektronik arbeitet dabei in Brückenschaltung, bei der ein NTC¹ als Temperatursensor und ein Potentiometer R_D für die Festlegung der Solltemperatur verwendet wird. Für genauere Details sind im Abschnitt A.4 die Schaltpläne, die von Allan Chow der Euro-Asia Electric Manufacturing Ltd. zur Verfügung gestellt wurden, dargestellt.

1.2. Sollzustand

Der Klimaschrank soll im Zuge der Masterarbeit derart umfunktioniert werden, sodass folgende prinzipielle Vorgaben erfüllt werden können:

- Verwendung des leistungsfähigeren Peltier-Verstärkers (vgl. [Plattner, 2009])
- Auswahl eines leistungsfähigeren elektrothermischen Elementes
- Temperaturmessung mit Pt1000 (Luftstrom)
- Temperatur- bzw. Feuchtemessung mit SHT15 (variable Innenraummessung)
- Temperaturregelbereich von 10°C bis 50°C

¹Negative Temperature Coefficient

1. Ziele

Dazu soll ein, auf ein Bakkalaureatsprojekt von [Frech, 2009] basierendes, Mikrocontrollersystem entworfen werden, welches die digitale Regelung ausführt. Folgende Anforderungen werden prinzipiell an die **Temperaturregelung** gestellt:

- einfache Reglerstruktur
- Festwert- bzw. Endwertregelung
- adaptive Reglereinstellung

Die Aufnahme der Temperatur soll dabei einerseits durch den digitalen Temperatur- und Feuchtesensor SHT15 und andererseits durch ein vom Institut für Elektronik entworfenes Messverfahren für einen Pt1000-Widerstand (vgl. [Hartl et al., 2008]) ermöglicht werden. Weiters soll das Mikrocontrollersystem zu einem Bedienelement erweitert werden, mit dem eine Steuerung des Klimaschranks möglich gemacht werden soll. Für eine softwaremäßige Fernsteuerung soll ein einfaches Protokoll auf Basis der seriellen Schnittstelle implementiert werden.

2. Klimaschrank

Um die Funktion eines Klimaschranks besser verstehen zu können, sollen in diesem Abschnitt notwendige Definitionen, die zugrundeliegenden physikalischen Effekte und die verwendeten Komponenten genauer erläutert werden.

2.1. Definitionen

Grundsätzlich kann unter einem **Klimaschrank** ein technisches Gerät zur Beeinflussung des Klimas in einem geschlossenen System verstanden werden. Für die Definition des **Klimas** gibt es in der Klimaforschung, abhängig vom jeweiligen Entwicklungsstand, verschiedenste Definitionen. In der meteorologischen Klimatologie beispielsweise lautet die Definition:

„Klima ist die örtlich charakteristische Häufigkeitsverteilung atmosphärischer Zustände und Vorgänge während eines hinreichend langen Bezugszeitraums, der so zu wählen ist, dass die Häufigkeitsverteilung der atmosphärischen Zustände und Vorgänge den typischen Verhältnissen am Bezugsort gerecht wird.“ ([Bernhardt and Hupfer, 1991, o.S.])

Weiters kann die räumliche Ausdehnung des Klimas durch die Begriffe **Mikroklima**, **Mesoklima** und **Makroklima** charakterisiert werden. Im Falle eines Klimaschranks trifft am ehesten das „**Mikroklima**“ zu, welches mit „*Kleinklima*“ bzw. „*Klima in bodennahen Luftschichten*“ definiert wird (vgl. [Dud10]).

Messbare bzw. beeinflussbare Eigenschaften eines Klimasystems werden **Klimaelemente** genannt. Solche Klimaelemente sind u. a.:

- **Lufttemperatur** (Messgerät: Thermometer)
- **Luftfeuchtigkeit** (Messgerät: Hygrometer)
- **Luftdruck** (Messgerät: Barometer)

Im Bezug auf den vorhandenen Klimaschrank und den vorhandenen Sensoren könnte das Klima im Innenraum durch die Regelung der Klimaelemente Temperatur bzw. Luft-

feuchtigkeit beeinflusst werden. Diese Masterarbeit wird sich in weiterer Folge ausschließlich mit der Regelung der Temperatur befassen¹.

2.2. Physikalische Effekte

Um ein bestimmtes Klima in einem Klimaschrank einzustellen, werden verschiedenste physikalische Effekte genutzt. Ein kleiner Auszug an praxisrelevanten Methoden, ohne Anspruch auf Vollständigkeit, sei in den folgenden Abschnitten angeführt.

2.2.1. Verdunstungskühlung

Dieses Verfahren stellt eines der ältesten aber auch bewährtesten Methoden dar. Durch Verdunsten von Wasser an der Luft entsteht ein Kühlpotential, welches unterhalb der Umgebungstemperatur liegt. Dabei können, abhängig von den klimatischen Bedingungen der Luft, Untertemperaturen von bis zu 10°C erreicht werden. Bis auf den Transport von Luft und Wasser wird bei diesem Verfahren keinerlei zusätzliche elektrische, mechanische oder thermische Energie benötigt (vgl. [Wik10e]).

2.2.2. Magnetische Kühlung

Diese Kühlmethode beruht auf den magnetischen Eigenschaften bestimmter Stoffe, die bei Magnetisierung Wärme freisetzen. Führt man einen Stoff, der mit diesen Eigenschaften ausgestattet ist, in ein Magnetfeld ein und leitet die so entstehende Wärme mit einer Kühlflüssigkeit ab, so kann dieser Stoff beim Entmagnetisieren² Wärme aufnehmen. Es wird somit thermische Energie beim Entfernen aus dem Magnetfeld zugeführt. Verfahren nach diesem Prinzip sind sehr effizient aber auch teurer, da geeignete Stoffe dafür benötigt werden (vgl. [Wik10e], [Wik10g]).

2.2.3. Joule-Thomson-Effekt

Wird ein Gas oder Gasgemisch durch Drosselung (Druckänderung) verändert, so expandiert bzw. kontrahiert es, d. h. das Volumen des Gases vergrößert oder verkleinert sich. Es verändert sich dabei der mittlere Teilchenabstand, wodurch sich die Temperatur des Gases ändert. Dieser Effekt, der auch mehrstufig ausgeführt werden kann, erlaubt eine Abkühlung bis in die Nähe des absoluten Nullpunktes (vgl. [Wik10e], [Wik10c]).

¹Der Zusammenhang zwischen Temperatur und Luftfeuchtigkeit bleibt somit unberücksichtigt.

²Beim Entfernen des Stoffes aus dem Magnetfeld.

2. Klimaschrank

Eine wichtige Anwendung des Joule-Thomson-Effekt im industriellen Bereich ist das „Linde-Verfahren“, bei dem durch Verflüssigung großer Mengen von Luft, Sauerstoff, Stickstoff und Argon ein Kühleffekt auftritt (vgl. [Wik10f]).

2.2.4. Thermoelektrischer Effekt

Zur Kühlung (oder Heizung) kann auch der thermoelektrische Effekt ausgenutzt werden. Dabei werden zwei elektrische Leiter mit unterschiedlichen elektronischen Wärmekapazitäten³ in Kontakt gebracht und durch einen von außen angelegten elektrischen Strom durchflossen. So wird, abhängig von der Anordnung der Materialien, eine Verringerung oder Erhöhung der Temperatur erreicht.

2.3. Elektrothermische Kühlung

Der verwendete Klimaschrank arbeitet (laut Spezifikationen im Abschnitt 1.1) mit einem elektrothermischen Kühlsystem. Um das Funktionsprinzip besser verstehen zu können, sollen in diesem Abschnitt die zugrundeliegenden physikalischen Effekte erklärt werden.

2.3.1. Seebeck-Effekt

Der Seebeck-Effekt beschreibt das Auftreten der Thermospannung. Werden dabei zwei unterschiedlich homogene Metalle verbunden und deren Kontaktstelle erhitzt (es entsteht eine Temperaturdifferenz $\Delta\vartheta = \vartheta_2 - \vartheta_1$ zwischen den beiden Metallen), so kann an den offenen Enden eine elektrische Spannung U bzw. ein Strom I gemessen werden (siehe Abbildung 2.1).

Die auftretende Spannung wird, wie bereits erwähnt, Thermospannung genannt und ist von der Temperaturdifferenz bzw. dem Temperaturgradienten abhängig. Diesem Effekt liegt zugrunde, dass sich Ladungsträger unter Einfluß eines Temperaturunterschiedes neu gruppieren. Die Thermospannung wird mit positiven Vorzeichen gezählt, wenn der Strom von der kalten zur warmen Kontaktstelle fließt.

Entdeckt wurde dieser physikalische Effekt im Jahre 1821 von Thomas Johann Seebeck. Er ermöglichte damit die bis heute bekannte Temperaturmessung mit Thermoelementen (vgl. [Ins10], [uni10b], [Qui10], [v.Ardenne et al., 2005]).

³Unter elektronischer Wärmekapazität, auch elektronische Wärmeleitung genannt, versteht man die Bewegungsenergie der Elektronen.

2. Klimaschrank

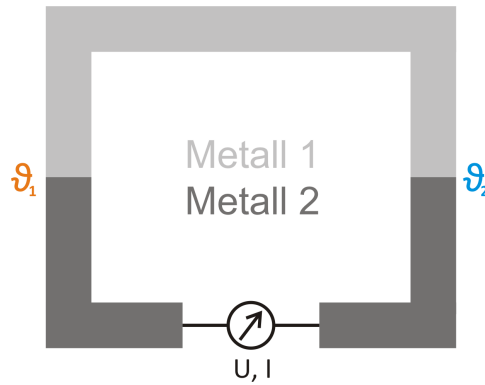


Abbildung 2.1.: Seebeck-Effekt

2.3.2. Peltier-Effekt

Der französische Physiker Jean Charles Athanase Peltier berichtete im Jahre 1834 über Temperaturbesonderheiten, die an Berührungsstellen zweier verschiedener Leiter bei Stromfluss auftraten. Der von ihm entdeckte Peltier-Effekt entspricht grundsätzlich einem elektrothermischer Wandler, der somit die Umkehrung des Seebeck-Effektes darstellt (siehe Abbildung 2.2) (vgl. [v.Ardenne et al., 2005], [Qui10]).

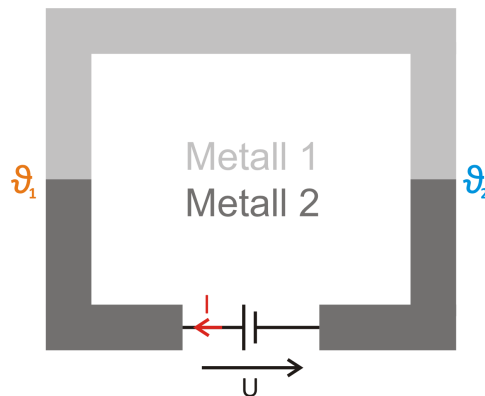


Abbildung 2.2.: Peltier-Effekt

Wird dabei ein Thermopaar (Kontaktstelle zweier unterschiedlicher homogener Metalle) von einem elektrischen Strom durchflossen, können zwei Fälle auftreten (vgl. [Wik10h]):

1. Fließt der Strom vom Material mit dem höherenergetischen Elektronen in den Leiter mit den niederenergetischen Elektronen, so erwärmt sich die Kontaktstelle. Es geben dabei die höherenergetischen Elektronen ihre Energie an die niederenergetischen Elektronen ab, wodurch diese selbst an Energie zunehmen.

2. Hier fließen niederenergetische Elektronen in den Leiter mit den höherenergetischen Elektronen, wodurch die höherenergetischen Elektronen an Energie verlieren. Dieser Effekt führt zu einer Verringerung der Temperatur.

2.4. Peltier-Element

2.4.1. Aufbau

Erst mit der Entwicklung hoch dotierter Halbleitermaterialien konnte der Wirkungsgrad des Peltier-Effekts derart gesteigert werden, sodass praktische Anwendungen aktuell wurden. Ein prinzipieller Aufbau eines Peltier-Elementes ist in Abbildung 2.3 dargestellt.

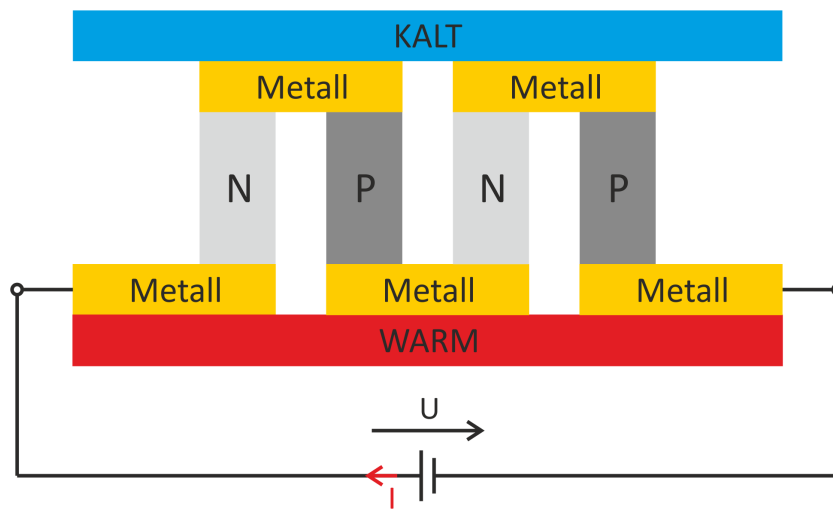


Abbildung 2.3.: Peltier-Element

Es sind dabei p- bzw. n-dotierte Halbleiterquader abwechselnd durch Metallbrücken zu einer Reihenschaltung verbunden. Die Metallbrücken dienen dabei als thermische Kontaktfläche, die nach außen hin durch eine Keramikplatte elektrisch isoliert ist. Der angelegte elektrische Gleichstrom fließt durch alle Halbleiterquader, sodass sich in Abhängigkeit der Stromrichtung und -stärke beispielsweise die oberen Verbindungsstellen abkühlen, während sich die unteren erwärmen. Es wird somit Wärme von der einen Seite des Elementes zur anderen Seite transportiert, sodass eine Temperaturdifferenz zwischen den Platten entsteht. Durch Abkühlen der warmen Seite kann die Temperaturdifferenz zwischen den Platten noch weiter gesteigert werden, da sich die kühlende Seite weiter abkühlen kann (vgl. [Wik10h], [v.Ardenne et al., 2005]).

2. Klimaschrank

Es werden für Peltier-Elemente Materialien verwendet, die eine **hohe** elektrische Leitfähigkeit und eine **niedrige** Wärmeleitfähigkeit besitzen. Da die meisten elektrischen Leiter auch eine hohe Wärmeleitfähigkeit haben, wählt man dafür, wie bereits erwähnt, dotierte Halbleiter. Als Halbleitermaterialien werden u. a. Bismuth Tellurid (Bi_2Te_3), Antimon Tellurid (Sb_2Te_3), Bismuth Selenide (Bi_2Se_3), etc. eingesetzt (vgl. [Qui10]).

2.4.2. Eigenschaften

Die größten Vorteile von Peltier-Elementen sind die geringe Größe, die Vermeidung jeglicher mechanischer Teile und die Vermeidung von Trägermedien. Eine Wärmepumpe im Vergleich benötigt immer ein Trägermedium (Kühlflüssigkeit) und einen Kompressor. Weiters kann durch eine Umkehr der Stromrichtung mit Peltier-Elementen sowohl gekühlt als auch geheizt werden. Nachteilig ist der zu niedrige Wirkungsgrad, der zu hoher Energieaufnahme bei vergleichsweise geringer Kühlleistung führt. Aus diesen Gründen werden Peltier-Elemente meist dazu eingesetzt, elektronische Baugruppen (z. B. spezielle Chips) zu kühlen (vgl. [Wik10h], [uni10a]).

2.4.3. Verwendetes Modell

Da das bestehende Peltier-Element aus dem Klimaschrank ($I_{MAX} \approx 4\text{A}$) den Anforderungen des Peltier-Verstärkers (siehe Abschnitt 2.5) nicht entsprechen konnte, wurde ein leistungsfähigeres Element der CP12/715 Serie ausgewählt. Da kein Datenblatt für das gewählte Element verfügbar war, mussten relevante Parameter ermittelt bzw. durch Vergleich von verwandten Modellen abgeschätzt werden:

- Hot Side Temperature $\approx 50^\circ\text{C}$
- $I_{MAX} \approx 15\text{A}$
- $V_{MAX} \approx 10\text{V}$

Anmerkung: Bei einem Strom von $I = \pm 10\text{A}$ konnte eine Gegenspannung von $U \approx 8\text{V}$ gemessen werden.

2.5. Peltier-Verstärker

Unter einem Peltier-Verstärker versteht man grundsätzlich einen Stromregler, der einen Laststrom durch eine ohmsche Last (Peltier-Element) treibt. Als Eingangsgröße dient

ein pulswertenmoduliertes Signal, dessen Tastverhältnis die Größe und Richtung des Stromes einstellt.

Wie bereits erwähnt, wurde der Peltier-Verstärker im Zuge eines Masterprojektes (vgl. [Plattner, 2009]) am Institut für Elektronik entworfen. Ein Blockschaltbild der notwendigen Komponenten ist in Abbildung 2.4 dargestellt.

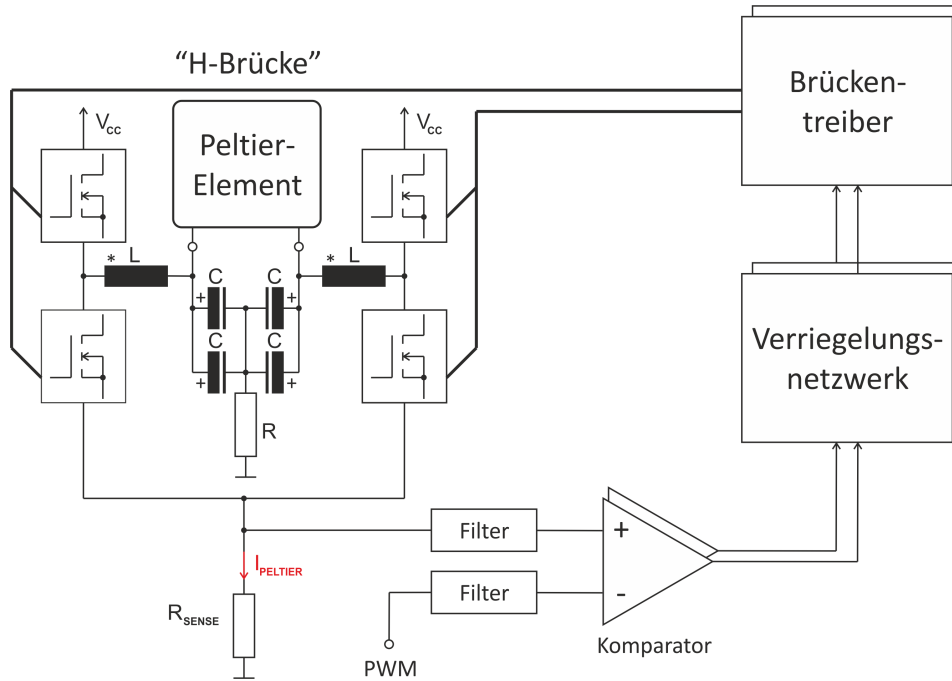


Abbildung 2.4.: Blockschaltbild Peltier-Verstärker

Funktionsprinzip: Für die Regelung des Stromes durch das Peltier-Element wird eine hysteretische Regelung verwendet. Dabei pendelt der Strom in einem dreieckförmigen Verlauf zwischen zwei Schwellen. Dieser Strom wird durch einen Messwiderstand R_{SENSE} erfasst, durch Tiefpässe gefiltert und als Messgröße an den Komparator geführt⁴. An den anderen Eingang des Komparators wird ein gefiltertes pulswertenmoduliertes Signal geleitet, wodurch der Vergleich der Messgröße mit der Sollgröße erfolgt. In Abhängigkeit der Mess- und Sollgröße entsteht am Ausgang des Komparators wieder ein pulswertenmoduliertes Signal, welches über das Verriegelungsnetzwerk⁵ an die Brückentreiber geleitet wird. Die H-Brücke besteht aus 4 Leistungs-MOSFET's, welche

⁴Eigentlich wird für jede Schwelle (obere und untere) ein Komparator benötigt. Der Einfachheit halber wird hier nur ein „Zweig“ erklärt. Der zweite Zweig funktioniert analog.

⁵Das Verriegelungsnetzwerk besteht aus 4 NAND-Gattern, einem MOSFET und einem Transistor-Entprellnetzwerk.

2. Klimaschrank

durch die zwei Brückentreiber (inkl. Ladungspumpe) angesteuert werden. Die Steuerungssignale dafür werden vom Verriegelungsnetzwerk jeweils gekreuzt an die Brückentreiber gelegt, um Kurzschlüsse durch das gleichzeitige Öffnen von MOSFET's einer Seite zu verhindern. Genauere Informationen bezüglich der Regelung sind in [Plattner, 2009, S. 15ff.] nachzulesen.

Implementierungsdetail: Wie bereits erwähnt, liefert der Peltier-Verstärker in Abhängigkeit des Tastverhältnisses des Eingangssignales einen Strom bestimmter Größe und Richtung (siehe Tabelle 2.1).

Duty-Cycle	$I_{PELTIER}$
[%]	[A]
≈ 0	-10
50	0
≈ 100	+10

Tabelle 2.1.: Tastverhältnis

Es ist zu berücksichtigen, dass lange ungewollte High- bzw. Low- Phasen der Eingangsgröße hohe Ausgangströme des Peltier-Verstärkers bewirken können. Aus diesem Grund wurde bei der Entwicklung darauf geachtet, entsprechende Schutzmaßnahmen zu implementieren. Eine statische Eingangsgröße ohne Flanken bewirkt beispielsweise, dass das nachgeschaltete Monoflop nicht mehr getriggert wird und dadurch das Transmission-Gate einen neutralen Pegel (z. B. GND) an den nachgeschalteten Komparator weitergibt. Relevante Parameter für das pulswertenmodulierte Eingangssignal sind (vgl. [Plattner, 2009, S. 12; 26; 107]):

- **Versorgungsspannungsbereich:** 24 - 100V DC (bis 64V getestet)
- **Eingangspegel:** CMOS⁶
- **Frequenz:** $> 70\text{ms}$ ($\approx 15\text{Hz}$)
- **Impulsbreiten:** $100\mu\text{s} < \tau_{duty} < 3.9\text{ms}$
- **Ausgangsstrom:** $I_{max} \approx \pm 10\text{A}$

⁶Obwohl der High-Pegel im Falle von CMOS in einem Bereich von 3.5V - 5V liegen muss, konnte ein Pegel von 3.3V erfolgreich getestet werden.

2.6. Kühlverfahren

Um das Funktionsprinzip zu erklären, wird davon ausgegangen, dass der Innenraum des Klimaschranks gekühlt werden soll. Der umgekehrte Fall (Beheizung des Innenraumes) funktioniert analog - lediglich die Seiten des Peliter-Elementes drehen sich um.

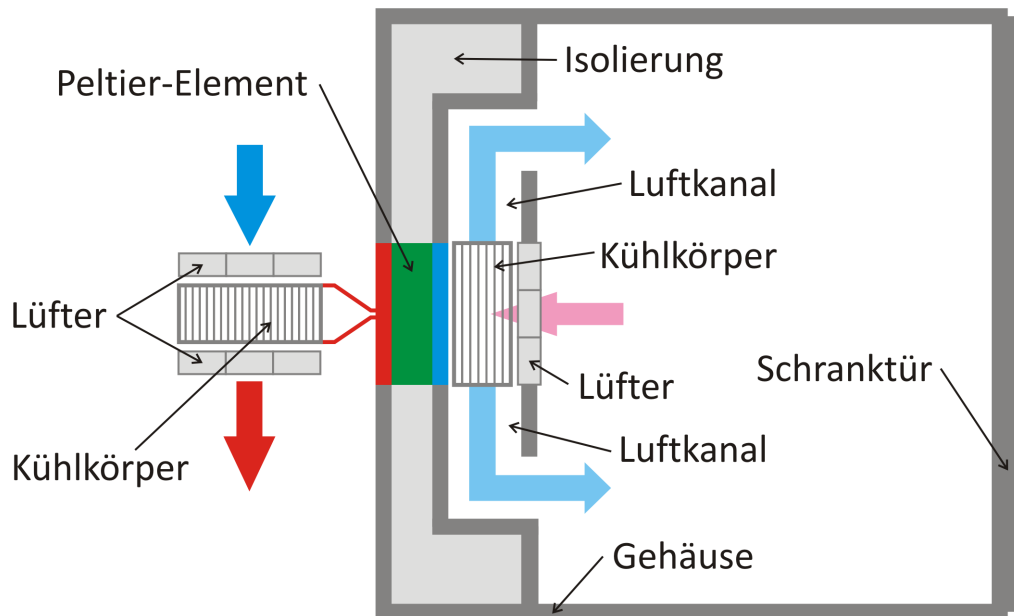


Abbildung 2.5.: Kühlverfahren (Ansicht von oben)

Wie in Abbildung 2.5 dargestellt, befindet sich das Peltier-Element in einem Kanal zwischen Innenraum und Rückseite des Klimaschranks. Wird das Peltier-Element für die Kühlung des Innenraumes versorgt, sinkt die Temperatur auf der Innenseite bzw. steigt die Temperatur auf der Außenseite des Elementes. Je höher der angelegte Strom, desto kälter wird die Innenseite bzw. desto wärmer wird die Außenseite des Peltier-Elementes.

Auf der inneren Seite des Peltier-Elements (kalte Seite) ist ein Kühlkörper montiert, deren Kühlrippen durch einen Lüfter mit Luft aus dem Innenraum versorgt wird. Die dadurch abgekühlte Luft kann durch die beiden seitlich liegenden Luftkanäle in den Innenraum des Klimaschranks strömen, wodurch ein Kühlkreislauf entsteht. Durch ein weiteres Kühlkörpersystem an der thermischen Gegenmasse (warme Seite des Peltier-Elements), kann die Wärmeenergie nach Außen abtransportiert werden. Das verwendete Kühlkörpersystem ist dabei mit Heatpipes ausgestattet, um einen optimalen Transport der Wärmeenergie von der Oberfläche des Peltier-Elements zum Kühlkörper zu gewährleisten.

Heatpipe: Das Funktionsprinzip einer Heatpipe beruht auf der physikalische Eigenschaft, dass durch Verdampfen und Kondensieren einer Flüssigkeit große Energiemengen transportiert werden können. Dabei bestimmt die Wahl der Flüssigkeit den jeweiligen Temperaturbereich. Zusätzlich herrscht im Inneren der Heatpipe ein Unterdruck, der z. B. Wasser schon bei einigen Plusgraden verdampfen lässt.

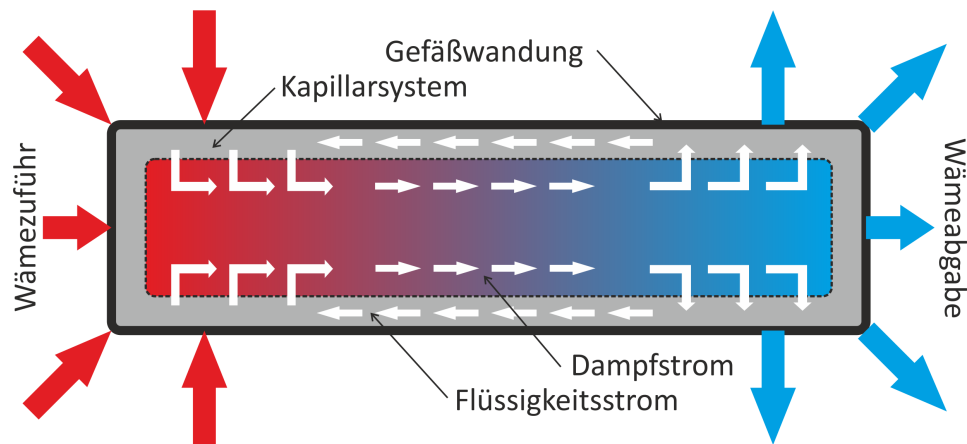


Abbildung 2.6.: Funktion einer Heatpipe

Wie in Abbildung 2.6 dargestellt, verdampft das Arbeitsmedium infolge der Wärmezufuhr an der einen Seite und kondensiert auf der anderen Seite (Dampfstrom). Aufgrund der Kapillarkräfte der inneren Struktur wird das Arbeitsmedium wieder zurück an den Ort der Verdampfung transportiert (Flüssigkeitsstrom) (vgl. [Qui10]).

Eine korrekte Auslegung der folgenden Parameter ist für eine optimale Funktion einer Heatpipe notwendig:

- zu übertragende Leistung
- Verdampfungstemperatur
- Kondensationstemperatur
- Einbaulage

Um ein Kühlkörpersystem mit Heatpipes für den Abtransport der Wärme des Peltier-Elementes verwenden zu können, müssen die eben erwähnten Auslegungsparameter übereinstimmen. Genauere Erläuterungen dazu sind im Abschnitt 3.1.1 zusammengefasst.

2.7. Temperaturmessung

Die Temperatur kann grundsätzlich mit zwei verschiedenen Sensoren gemessen werden. Einerseits mit dem Pt1000-Sensor, der im inneren Kühlkörper des Peltier-Lüfter-Stacks die Temperatur des Luftstromes aufnimmt und andererseits mit dem SHT15-Sensor, der im Innenraum des Klimaschranks individuell positioniert werden kann. Die genaue Positionierung ist im Abschnitt 3.1.2 dokumentiert.

2.7.1. Pt1000

Wie bereits erwähnt, wird für die Temperaturmessung u. a. ein vom Institut für Elektronik entworfenes Pt1000-Thermometer verwendet (vgl. [Hartl et al., 2008, S. 701ff]). Dieses Messverfahren stellt grundsätzlich eine Kombination aus folgenden Komponenten dar (siehe Abbildung 2.7):

- Sensorinterface
 - Dreileiteranschluss (Brückenschaltung)
 - Transistor-Präzisionsstromquelle
- Ladungsausgleichsintegrator
 - Tastverhältnismultiplizierer
 - Differenzintegrator
 - Komparator
 - D-Flip-Flop

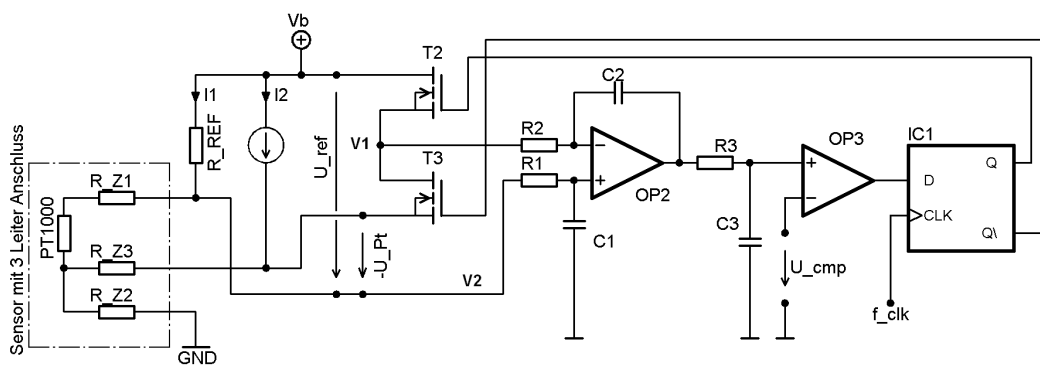


Abbildung 2.7.: Pt1000-Messverfahren (vgl. [Frech, 2009, S. 16])

Funktionsprinzip: Um eine genaue Widerstandsmessung zu ermöglichen, wird der Widerstandswert des Pt1000 relativ zu einem Vergleichs- oder Referenzwiderstand (Brückenschaltung) gemessen. Um dabei die Zuleitungswiderstände, die bei der Dreileitermethode als identisch anzusehen sind, zu kompensieren bzw. um die Ströme in den Brückenzweigen identisch zu halten, wurde eine Transistor-Präzisionsstromquelle verwendet. Die Spannungsabfälle an den Widerständen (U_{ref} , U_{Pt}) können dadurch an einen Differenzintegrator geschaltet werden, der das Integral der Differenzspannung ermittelt ($U_a = -\frac{1}{RC} \int (V_1 - V_2) dt$). Der nachgeschaltete Komparator vergleicht nun die Ausgangsspannung des Differenzintegrators mit einer vorgegebenen Schaltschwelle (U_{cmp}). Je nachdem ob die Ausgangsspannung des Differenzintegrators kleiner oder größer als die Komparatorspannung ist, aktiviert ein taktgesteuertes D-Flip-Flop die Transistoren T_2 und T_3 , die abwechselnd die Spannung U_{Pt} bzw. $U_{Pt} + U_{ref}$ an den Eingang des Differenzintegrators legen. Es wird somit ein Tastverhältnis eingestellt, für welches folgender Zusammenhang gilt:

$$d = \frac{U_{Pt}}{U_{Pt} + U_{ref}} = \frac{t_{einT_2}}{t_{einT_2} + t_{einT_3}} = \frac{t_{einT_2}}{T}$$

Durch eine entsprechende Logik bzw. Taktung des D-Flip-Flops können die gezählten Taktflanken während T_2 aktiv ist ($= t_{einT_2}$) von einem Mikrocontroller ausgelesen werden. Der Widerstandswert kann nun mit folgender Formel berechnet werden, wobei N die Anzahl der gezählten Impulse während eine Messperiode ($= N_{Periode}$) darstellt:

$$\frac{N}{N_{Periode}} = \frac{R_{PT}}{R_{PT} + R_{REF}}$$

$$R_{PT} = R_{REF} \cdot \frac{N}{N_{Periode} - N}$$

Das Verfahren erlaubt eine Messung der Temperatur in einem Bereich von -50 bis $+250^\circ\text{C}$ mit einer Messunsicherheit von $< 0.1^\circ\text{C}$. Die Auflösung beträgt dabei 0.01°C .

2.7.2. SHT15

Für die Messung der Temperatur im Inneren des Klimaschranks wird der digitale Temperatur- und Feuchtesensor SHT15 verwendet. Der kapazitive Sensor kann die Temperatur in einem Bereich von -40 bis $+125^\circ\text{C}$ (Genauigkeit $\pm 0.3^\circ\text{C}$) und die relative Feuchte von 0 bis 100% (Genauigkeit $\pm 2.0\%$) bestimmen. Weiters liefert er ein digitales Ausgangssignal, welches mit einer seriellen 2-Wire-Schnittstelle an den Mikrocontroller angeschlossen und ausgelesen werden kann. Weitere Details können aus dem Datenblatt des Herstellers (vgl. [Sen10]) entnommen werden.

3. Hardwareentwurf

Hardwaremäßig wurde der Klimaschrank um mechanische Komponenten und elektronische Baugruppen erneuert bzw. erweitert, auf welche in den folgenden Kapitel näher eingegangen wird.

3.1. Mechanische Komponenten

In diesem Abschnitt sind mechanische Komponenten dokumentiert, die aufgrund neuer elektronischer Baugruppen für den Klimaschrank entworfen und gefertigt werden mussten. In der Abbildung 3.1 sind die entsprechenden Komponenten dargestellt. Dazu zählen:

- Peltier-Lüfter-Stack
- Aufbau Peltier-Verstärker/Versorgungsplatine
- Rückwandabdeckung inkl. Kommunikationsport
- Bedienteil

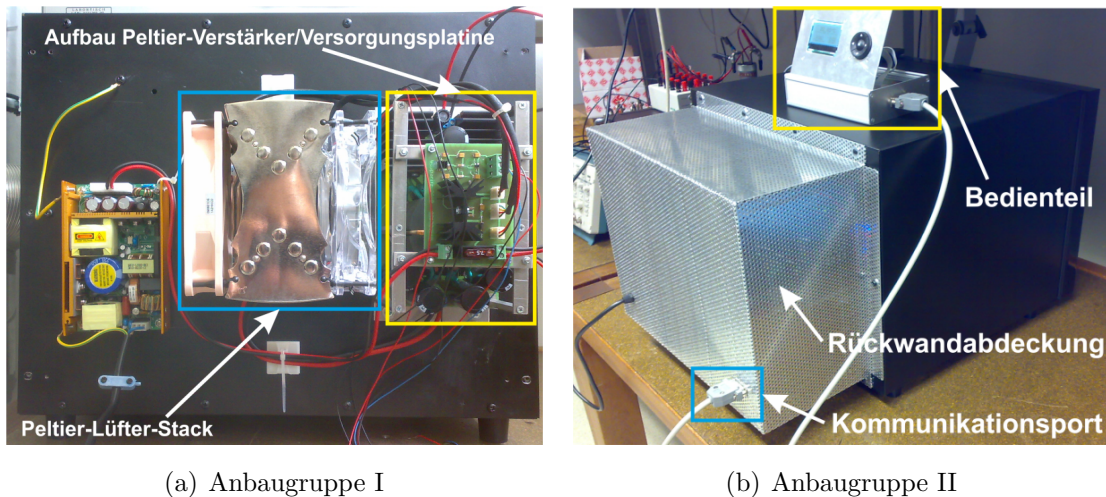


Abbildung 3.1.: Mechanische Anbaugruppen

3.1.1. Peltier-Lüfter-Stack

Die Verwendung des stärkeren Peltier-Elementes brachte eine derartige Erhöhung der Temperatur der thermischen Gegenmasse mit sich, sodass sich der bestehende Peltier-Lüfter-Stack als zu schwach für den Abtransport der Wärme herausstellte (im Speziellen die Seite der thermischen Gegenmasse). Aus diesem Grund wurde nach einem leistungsstarken Kühlkörpersystem gesucht, welches mit den auftretenden Wärmeleistung umgehen kann. Aufgrund der mechanischen Gegebenheiten bzw. den wirtschaftlichen Mitteln wurde das CPU-Kühlersystem „Nordwand Rev. B“ des Herstellers „Alpenföhn“¹ ausgewählt (siehe Abbildung 3.2).

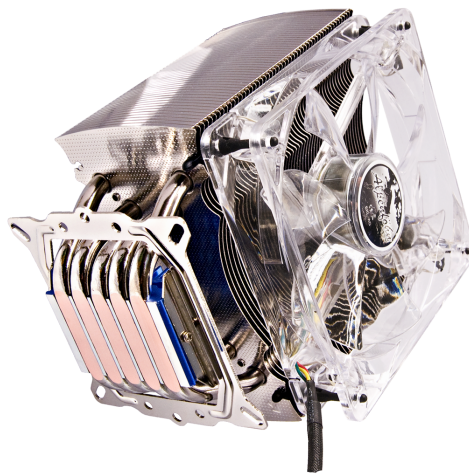


Abbildung 3.2.: Nordwand Rev. B

Dieses CPU-Kühlkörpersystem, welches mit Heatpipes ausgestattet ist, eignet sich laut Spezifikationen² für Prozessoren der Intel® Pentium® 4 Familie. Prozessoren dieser Kategorie sind für eine maximale Gehäusetemperatur von $T_{CASE} \approx 70^{\circ}\text{C}$ und eine maximale TDP³ von 115 Watt ausgelegt.

Messungen an der warmen Seite des Peltier-Elementes (thermische Gegenmasse) lieferten eine maximale Temperatur von etwa 50°C . Für die Abschätzung der Wärmeleistung wurde bei einem maximalen Strom von $\pm 10\text{A}$ die Gegenspannung am Peltier-Element ermittelt und aus folgendem Zusammenhang die maximale Wärmeleistung näherungsweise berechnet:

¹<http://www.alpenfoehn.de>

²<http://www.intel.com>

³Thermal Design Power

3. Hardwareentwurf

$$\Rightarrow P_{max} = U_{gem} \cdot I_{max} = \pm 8V \cdot \pm 10A = \pm 80\text{Watt}$$

Da sich die Auslegungsparameter des Peltier-Elementes mit den Spezifikationen des CPU-Kühlkörpersystem deckten, wurde das CPU-Kühlkörpersystem für die Kühlung der thermischen Gegenmasse verwendet. Um die Wärme optimal abtransportieren zu können, wurden als Erweiterung ein weiterer Lüfter mit einem maximalen Luftdurchflussvolumen von ca. $92\text{m}^3/\text{h}$ montiert (bei einer Drehzahl von 1300 U/m). Der Peltier-Lüfter-Stack ist in der Abbildung 3.3 dargestellt.

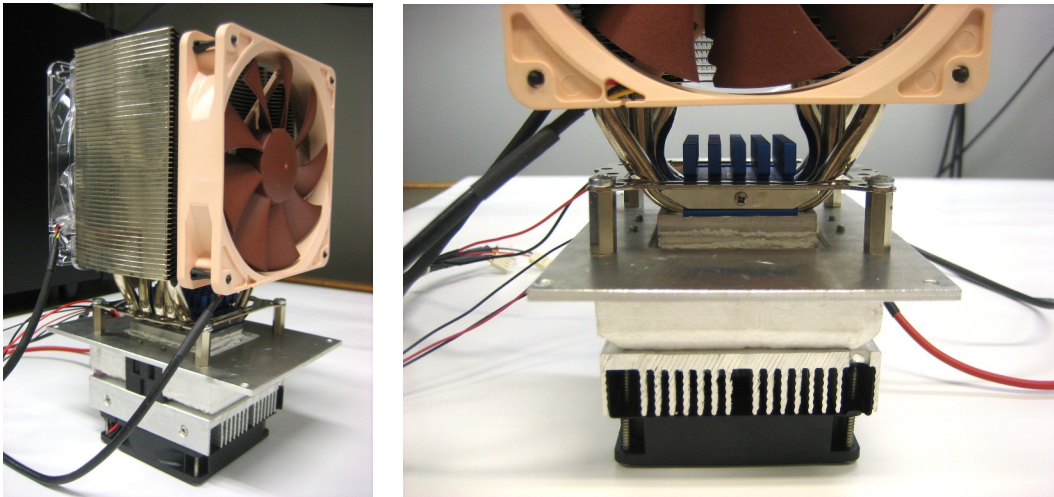


Abbildung 3.3.: Peltier-Lüfter-Stack

3.1.2. Sensorpositionierung

Wie bereits erwähnt, wurde der Pt1000-Sensor am inneren Kühlkörper des Peltier-Lüfter-Stacks montiert, um eine Temperaturmessung des Luftstromes zu ermöglichen (siehe Abbildung 3.4). Es wurde bei der Montage genau darauf geachtet, dass der Sensor keinen direkten Kontakt mit dem Kühlkörpermaterial aufweist, um die Temperatur des Luftstromes zu messen und nicht die des Kühlkörpermaterials.

Der digitale Temperatur- und Feuchtesensor SHT15 kann, wie bereits erwähnt, beliebig im Innenraum positioniert werden. Die Anbindung erfolgt dabei über ein Kabel, welches direkt zum Bedienteil geleitet wird.

3. Hardwareentwurf

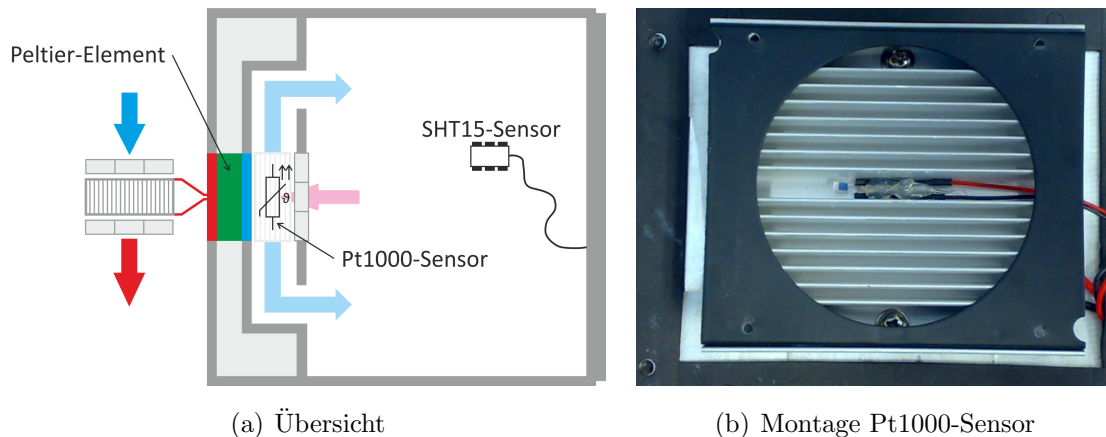


Abbildung 3.4.: Sensorpositionierung

3.1.3. Aufbau Peltier-Verstärker/Versorgungsplatine

Der Peltier-Verstärker und die Versorgungsplatine wurden durch einen mechanischen Aufbau übereinander montiert und rechts neben dem CPU-Kühlkörpersystem positioniert (siehe Abbildung 3.5). Grund dafür war die Wärmeentwicklung der Leistungsbau-elemente, die bei voller Belastung beider Baugruppen eine Kühlung durch das CPU-Kühlkörpersystem⁴ benötigten.

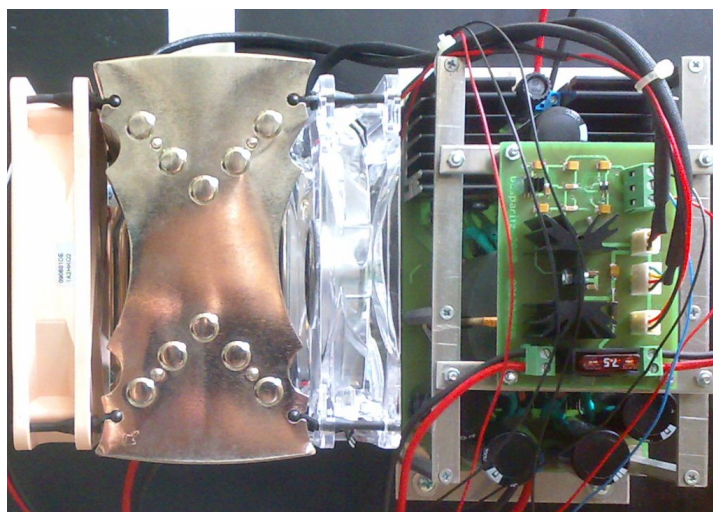


Abbildung 3.5.: Aufbau Peltier-Verstärker/Versorgungsplatine

⁴Die Luft strömt aufgrund der Drehrichtung der Lüfter von links nach rechts (Sicht von hinten).

3.1.4. Rückwandabdeckung

Die Rückwandabdeckung wurde aus Lochblech (siehe Abbildung 3.1(b)) gefertigt, um eine optimale Luftzufuhr/-abfuhr zu bzw. von den eben besprochenen Baugruppen zu ermöglichen. Das Netzkabel wurde durch eine Kabeldurchführung mit Zugentlastung zum Netzteil im Inneren der Abdeckung geführt.

3.1.5. Kommunikationsport

Um eine Spannungsversorgung des Bedienteils bzw. eine Kommunikation mit der an der Rückwand des Klimaschranks montierten Baugruppen und Sensoren zu ermöglichen, wurde an der Rückwandabdeckung ein 15-poliger D-Sub Stecker (Pinbelegung siehe Abbildung 3.6) mit folgender Steckerbelegung (siehe Tabelle 3.1) montiert:

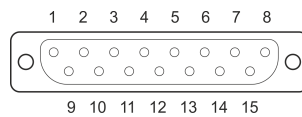


Abbildung 3.6.: Pinbelegung D-Sub-Stecker 15-polig

Pin	Baugruppe	Bezeichnung	Anmerkung
1	Bedienteil	+3.3V	positive Versorgungsspannung
2	Bedienteil	-3.3V	negative Versorgungsspannung
3	Bedienteil	GND	Masse
4	Pt1000-Sensor	Pt1000+	Dreileiteranschluß Pt1000
5	Pt1000-Sensor	Pt1000-	Dreileiteranschluß Pt1000
6	Pt1000-Sensor	Pt1000-	Dreileiteranschluß Pt1000
7	Peltier-Verstärker	PWM	pulsweitenmodulierte Stellgröße
8	Peltier-Serstärker	GND	Signalmasse
9	SHT15-Sensor	GND	Signalmasse
10	SHT15-Sensor	DATA	Bidirectional Serial Data (I ² C)
11	SHT15-Sensor	SCK	Serial Clock (I ² C)
12	SHT15-Sensor	+3.3V (VDD)	Source Voltage (I ² C)

Tabelle 3.1.: Pinbelegung Rückwandsteckers

Anmerkung: Die Pins 9-12 sind nur beim Stecker des Bedienteils belegt, da ein separates Kabel vom Bedienteil zum variablen SHT15-Innenraumsensor geführt wurde.

3.1.6. Bedienteil

Das Bedienteil (siehe Abbildung 3.7) stellt im Prinzip die Fernsteuerung des Klimaschranks dar. Wie bereits im vorigen Abschnitt erwähnt, ist es durch einen 15-poligen D-Sub-Stecker mit dem Klimaschrank verbunden, wodurch einerseits die Stromversorgung der Mikrocontroller- und Peripherieplatine gewährleistet und andererseits die Kommunikation mit den Sensoren bzw. Peltier-Verstärker hergestellt wird. Weiters verfügt das Bedienteil über eine RS232- und USB-Schnittstelle (Details siehe Abschnitt 3.2.3), wodurch eine Kommunikation bzw. Fernsteuerung durch eine Software (Details siehe Abschnitt 6.3.5) implementiert werden konnte.



Abbildung 3.7.: Bedienteil

3.2. Elektronische Komponenten

In den folgenden Abschnitten ist die Auswahl bzw. die Entwicklung der elektronischen Komponenten dokumentiert. Schaltpläne und Layouts der entsprechenden Baugruppen sind im Abschnitt A des Anhangs angeführt.

3.2.1. Spannungsversorgung

Als globale Spannungsversorgung wurde ein XP Power Open-Frame-Netzteil zugekauft. Dieses Schaltnetzteil weist folgende Spezifikationen auf (siehe Tabelle 3.2):

Model	SDS180PS24B
Operating Temperature	-20°C to +70°C
Input Voltage	90 - 260VAC
Input Frequency	47 - 63Hz
Power Factor	0.97 typical
Input Protection	Internal T3.15A/250VAC Fuse
Maximum Output Power	180W
Output Voltage	24V
Maximum Output Current	7.5A

Tabelle 3.2.: Spezifikationen Open-Frame-Netzteil

Durch eine nachgeschaltete Versorgungsplatine wurden alle notwendigen Versorgungsspannungen erzeugt und für alle weiteren Baugruppen durch entsprechende Stecker zur Verfügung gestellt. Die Entscheidung für eine gesonderte Versorgungsplatine wurde durch folgende Überlegungen bekräftigt:

- Übersichtlichkeit
- Ausreichend Platz unter der Rückwandabdeckung
- Schutz vor Berührung durch die Positionierung unter der Rückwandabdeckung
- Indirekte Kühlung der Leistungsbaulemente bzw. deren Kühlkörper durch die Abluft des Peltier-Lüfter-Stacks

Details zur Implementierung: Die Versorgungsplatine wird mit 24V aus dem Schalt-
netzteil versorgt. Nach einer 7.5A Kfz-Sicherung wurde ein Leitersplattensteckverbin-
der zur Versorgung des Peltier-Verstärkers geschaltet. Im Fehlerfall ($I_{Amplifier} > 7.5A$)
würde die KFZ-Sicherung ansprechen, die danach manuell ersetzt werden müsste. An-
schließend wurden die 24V des Schaltnetzteils über eine Schutzdiode⁵ ($1N4001$) zum
12V-Spannungsregler ($L78S12CV$) geführt. Zur Schwingungsunterdrückung wurden
vom Eingang- und Ausgangspin 100nF Keramikkondensatoren gegen Masse geschaltet.
Weiters diente ein 10uF Tantal-Elektrolytkondensator am Eingang des Spannungsreglers
gegen Masse zur Glättung.

Für die Kühlkörperdimensionierung des 12V-Spannungsreglers musste die maximale Ver-
lustleistung abgeschätzt werden. Folgende Überlegungen wurden getroffen:

- Eingangsspannung des 12V-Spannungsreglers⁶:

$$U_E = U_{A,Netzteil} - U_{F,Diode} = 24V - 1.1V = 22.9V$$

- Spannungsabfall am 12V-Spannungsregler:

$$\Delta U = U_E - U_A = 22.9V - 12V = 10.9V$$

- Stromaufnahme der 3 Lüfter des Peltier-Lüfter-Stacks:

$$I_{Cooler} \approx 500mA$$

- Stromaufnahme der Mikrocontrollerplatine (inkl. Peripherieplatine):

$$I_{Controller} \approx 100mA$$

Aus den entsprechenden Datenblättern konnten mit Hilfe der folgenden Wärmewi-
derstände⁷ die maximale Temperatur des 12V-Spannungsreglers inkl. Kühlkörper ab-
geschätzt werden:

- 12V-Spannungsregler ($L78S12CV$):

$$R_{th,JC} = 3^{\circ}C/W$$

- Kühlkörper ($SK\ 129-25,4-STS-2$):

$$R_{th,HA} = 6.5^{\circ}C/W$$

$$P = \Delta U \cdot I_{ges} = 10.9V \cdot 0.6A = 6.54W$$

$$T_{JA} = P \cdot (R_{th,JC} + R_{th,HA}) = 6.54W \cdot (3^{\circ}C/W + 6.5^{\circ}C/W) \cong 62^{\circ}C$$

⁵Die Forwardspannung der Schutzdiode beträgt laut Datenblatt $U_F = 1.1V$.

⁶ $U_{E,max} = 30V$ laut Datenblatt

⁷Der Wärmewiderstand der Wärmeleitpaste $R_{th,CH}$ ist vernachlässigbar klein und wurde somit nicht weiter berücksichtigt.

3. Hardwareentwurf

Aus diesen Ergebnissen kann geschlossen werden, dass der Kühlkörper auch ohne der indirekten Belüftung durch den Peltier-Lüfter-Stacks ausreichend groß dimensioniert wurde, da der Spannungsregler laut Datenblatt einer maximale Temperatur⁸ von 150°C standhalten könnte.

Die stabilisierte Spannung von 12V wurde nun auf drei Leiterplattensteckverbinder geleitet, die als Anschluss für die Lüfter des Peltier-Lüfter-Stacks dienten.

Zur Erzeugung der 3.3V wurde ein weiterer Spannungsregler (*MC33269D-3.3G*) verwendet. Zur Stabilisierung wurden zwei 10uF Tantal-Elektrolytkondensatoren von Eingangs- und Ausgangspin gegen Masse geschaltet.

Für die Kühlkörperdimensionierung des 3.3V-Spannungsregler musste wieder die Verlustleistung am verwendeten Spannungsregler abgeschätzt werden:

- Eingangsspannung des 3.3V-Spannungsreglers⁹:

$$U_E = 12V$$

- Spannungsabfall am 3.3V-Spannungsregler:

$$\Delta U = U_E - U_A = 12V - 3.3V = 8.7V$$

- Stromaufnahme der Mikrocontrollerplatine (inkl. Peripherieplatine):

$$I_{Controller} \approx 100mA$$

Aus den entsprechenden Datenblättern konnten wieder die erforderlichen Wärmewiderstände abgelesen werden, um daraus die maximale Temperatur des 3.3V- Spannungsreglers inkl. Kühlkörper abzuschätzen:

- 3.3V-Spannungsregler (*MC33269D-3.3G*):

$$R_{th,JC} = 25^\circ C/W$$

- Kühlkörper (*ICK SMD A 5 SA*):

$$R_{th,HA} = 123^\circ C/W$$

$$P = \Delta U \cdot I_{ges} = 8.7V \cdot 0.1 = 0.87W$$

$$T_{JA} = P \cdot (R_{th,JC} + R_{th,HA}) = 0.87W \cdot (25^\circ C/W + 123^\circ C/W) \cong 129^\circ C$$

⁸Absolute Maximum Ratings

⁹ $U_{E,max} = 20V$ laut Datenblatt

3. Hardwareentwurf

In diesem Fall wäre der Betrieb des Spannungsreglers inkl. Kühlkörper ohne der indirekten Belüftung durch den Peltier-Lüfter-Stacks problematisch, da der kritische Temperaturbereich des Spannungsreglers laut Datenblatt zwischen 125°C und 150°C liegt.

Zur Generierung der -3.3V wurde ein Spannungsinverter (*MAX828*) verwendet, der nach dem Switched-Capacitor Prinzip arbeitet und die an den Eingang angelegte Spannung mit Hilfe von zwei 10uF Tantal-Elektrolytkondensatoren invertiert. Diese zwei Spannungen wurden wieder auf einen Leiterplattensteckverbinder gelegt, der für die Versorgung der Mikrocontrollerplatine diente.

3.2.2. Mikrocontroller und -beschaltung

Grundsätzlich handelt es sich beim verwendeten Mikrocontroller (*MSP430F1612*) um einen Mixed-Signal-Prozessor der Firma Texas Instruments (vgl. [Tex10]), der folgende Features bietet:

- Versorgungsspannung 1.8V - 3.6V
- 16-Bit RISC Architektur
- 12-Bit ADC
- Dual 12-Bit DAC
- Timer A (16-Bit) mit 3 Capture/Compare Einheiten
- Timer B (16-Bit) mit 3 oder 7 Capture/Compare Einheiten
- USART0 (UART, SPI oder I²C)
- USART1 (UART oder SPI)
- JTAG Interface

Die Beschaltung des Mikrocontrollers erfolgte prinzipiell laut Datenblatt. Zur standardmäßigen Beschaltung gehörten folgende Baugruppen, wobei eine externe ADC-Referenz und ein weiterer Quarz schaltungstechnisch vorgesehen aber hardwaremäßig nicht bestückt wurden:

- Stromversorgung
- Resetbeschaltung

- Quarzbeschaltung
- Beschaltung des JTAG-Interfaces

Der Controller verfügt über 6 Ports, von denen 5 laut Tabelle 3.3 beschaltet wurden¹⁰. Nicht verwendete Mikrocontrollerpins vom Port 4 und Port 5 wurden auf eigene Steckerleisten gelegt, um bei softwaremäßigen Erweiterungen abgreifbare Punkte zur Verfügung zu stellen.

3.2.3. Serielle Datenübertragung

Prinzipiell bietet die Mikrocontrollerplatine zwei Möglichkeiten für die serielle Kommunikation - einerseits mit einem klassischen *MAX232* und andererseits mit einem *FT232RL*. Durch zwei Jumper (*TxD_JMP*, *RxD_JMP*) kann zwischen den beiden Möglichkeiten gewählt werden.

MAX232: Die Beschaltung des Bausteins erfolgte grundsätzlich laut Datenblatt. Da aber die Möglichkeit zur Umschaltung zwischen dem *MAX232* und dem *FT232RL* geboten wurde, musste beim Transmit-Eingang des Bausteins (*T1IN* - Pin 11) ein Pull-Down-Widerstand geschaltet werden. Weiters wurde der Transmit-Ausgang (*T1OUT* - Pin 14) bzw. der Receive-Eingang (*R2IN* - Pin 8) des *MAX232* mit einem Pi-Filter¹¹ versehen, um mögliche Störungen auf Signalleitungen zu reduzieren.

FT232RL: Die zweite Möglichkeit zur seriellen Übertragung wurden durch den *FT232RL* ermöglicht, der im Prinzip einen USB/Seriell-Umsetzer darstellt. Das Bauteil kann direkt vom Mikrocontroller seriell angesteuert werden (*RxD* und *TxD* mit TTL-Pegel), wobei der Chip das komplette USB-Handling (elektrisches Interface, Protokoll) übernimmt. Die Beschaltung erfolgte in der „*Self Powered Configuration*“ aus dem Datenblatt, wobei zwei Serienwiderstände bei den USB-Datenleitungen zur Pegelanpassung hinzugefügt wurden. Auch hier musste beim Transmit-Eingang des Bausteins (*RXD* - Pin 5) ein Pull-Down-Widerstand geschaltet werden, um einem möglichen offenen Eingang bei entsprechender Jumperstellung entgegenzuwirken. Weiters wurde das LED-Interface beschaltet, um auf Aktivitäten auf den Datenleitungen hinzuweisen.

¹⁰Portpins ohne Verwendung sind in der Tabelle nicht angeführt.

¹¹Das Filter wurde empirisch aufgrund von Messungen an den Signalleitungen eines ungeschirmten Kabels, welche durch genormte Störgrößen (Burst) beeinflusst wurden, dimensioniert.

3. Hardwareentwurf

Port	Funktion	Anmerkung
1.0	TACLK	externer ADC (TACLK Input)
1.1	GATE.TIME	externer ADC (Timer A Capture-Register)
1.2	FF \overline{Q}	externer ADC (Timer A Capture-Register)
2.0	Push Button	Navigationskit Input
2.1	Down Button	Navigationskit Input
2.2	Right Button	Navigationskit Input
2.3	Up Button	Navigationskit Input
2.4	Left Button	Navigationskit Input
2.5	SCK	SHT15 Output
2.6	DATA	SHT15 (Input/Output)
2.7	CS	Display
3.0	RES	Display (USART0/SPI Mode)
3.1	MOSI	Display (USART0/SPI Mode)
3.2	MISO	Display (USART0/SPI Mode)
3.3	SCLK	Display (USART0/SPI Mode)
3.6	TxD	RS232 (USART1/UART Mode)
3.7	RxD	RS232 (USART1/UART Mode)
4.0	PORT4.1	Steckerleiste Port 4 (Input/Output)
4.1	PORT4.2	Steckerleiste Port 4 (Input/Output)
4.2	PWM	Stellgröße für Peltier-Verstärker (Timer B Register)
4.3	STAT	Status-LED für Regelung (Timer B Register)
4.4	PORT4.3	Steckerleiste Port 4 (Input/Output)
4.5	PORT4.4	Steckerleiste Port 4 (Input/Output)
4.6	OUTPUT_1	Ausgang für 12V MOSFET-Schalter 1
4.7	OUTPUT_2	Ausgang für 12V MOSFET-Schalter 2
5.0	PORT5.1	Steckerleiste Port 5 (Input/Output)
5.1	PORT5.2	Steckerleiste Port 5 (Input/Output)
5.2	PORT5.3	Steckerleiste Port 5 (Input/Output)
5.3	PORT5.4	Steckerleiste Port 5 (Input/Output)

Tabelle 3.3.: Mikrocontroller Portbelegung

3.2.4. MOSFET-Schalter

Um die Möglichkeit zu schaffen, induktive 12V Lasten (z. B. Lüfter des Peltier-Lüfter-Stacks) mit dem Mikrocontroller zu schalten, wurden zwei MOSFET-Schalter implementiert, die durch Setzen der entsprechenden Ports (siehe Tabelle 3.3) aktiviert werden können. Auch hier wurden hochohmige Pull-Down-Widerstände (zwischen Gate und Source) geschaltet, um einen definierten Pegel bei Inaktivität der Ansteuerung sicherzustellen. Parallel zur induktiven Last wurde eine Freilaufdiode geschaltet, die die Spannungsspitze beim Abschalten einer induktiven Last auf die Durchlassspannung der Diode begrenzt.

3.2.5. Messschaltung

Die Messschaltung zur Verarbeitung der Pt1000-Werte (Präzisionsstromspiegel, Charge-Balance-ADC, etc.) wurde vollständig von [Hartl et al., 2008] übernommen.

3.2.6. Displayansteuerung

Beim verwendeten Display (*64128K-FC-BW-3*) handelt es sich um ein LCD-Modul der Firma Displaytech. Die wichtigsten Spezifikationen (vgl. [Dis10]) lauten:

- Modules: Graphic COG Panels
- Resolution: 128 x 64
- Viewing Area (mm): 50.0 x 25.0
- Module Size (mm): 56.0 x 39.0
- Pinout: Pins
- LCD Module: FSTN¹²
- Polarizer Type: Transflective
- Viewing Angle: 6 o'clock
- Operating Temperature: Wide
- Backlight: LED (white)
- LCD Controller/Driver: ST7565R-G (vgl. [Sit10])

¹²Film Super Twisted Nematic

3. Hardwareentwurf

Prinzipiell wurde das Display über die entsprechenden SPI¹³-Ports des Mikrocontrollers (siehe Tabelle 3.3) angesteuert. Die Beschaltung wurde laut Datenblatt durchgeführt. Da das Display aufgrund von Designentscheidungen nicht auf der Mikrocontrollerplatine verschaltet werden konnte, musste es über einen gesonderten 10-poligen Leiterplattensteckverbinder verbunden werden. Die Pinbelegung des Steckers ist in der Tabelle 3.4 zusammengefasst.

Pin	Funktion	Anmerkung
1	CS	Cable Select
2	RES	Reset
3	MISO	Master In Slave Out
4	SCLK	Serial Clock
5	MOSI	Master Out Slave In
6	-	-
7	-	-
8	-	-
9	+3.3V	Versorgungsspannung
10	GND	Signalmasse

Tabelle 3.4.: Display Steckerbelegung

Für die Ansteuerung der Hintergrundbeleuchtung, die im Prinzip eine Parallelschaltung dreier LEDs ($U_F = 3.2V \pm 0.2V$; $I_F = 60mA$) darstellte, konnte ein theoretischer Wert für einen Vorwiderstand errechnet werden:

$$R_V = \frac{U_B - U_F}{I_F} = \frac{3.3V - (3.2V - 0.2V)}{3 \cdot (20 \cdot 10^{-3})} = 5\Omega$$

Die Anwendung zeigte aber, dass ein Strom $I_{F,gew} \approx 6mA$ genügt, um eine ausreichend helle Hintergrundbeleuchtung zu gewährleisten. Die dafür nötige Erhöhung des Vorwiderstandes unterstützte die Dimensionierung des 3.3V-Spannungsreglers, da eine geringere Stromaufnahme auch eine niedrigere Verlustleistung bedeutete:

$$R_V = \frac{U_B - U_F}{I_F} = \frac{3.3V - (3.2V - 0.2V)}{3 \cdot (2 \cdot 10^{-3})} = 50\Omega \Rightarrow R_{V,gew} = 68\Omega$$

¹³Beim „*Serial Peripheral Interface*“ handelt es sich um einen synchronen seriellen Bus, bei dem digitale Schaltungen nach dem Master/Slave-Prinzip miteinander kommunizieren. Jeder Teilnehmer ist dabei an drei gemeinsame Leitungen (MISO, MOSI, SCLK) angeschlossen. Details können unter [Wik10i] nachgelesen werden.

3.2.7. Navigationskit

Um Benutzereingaben beim Bedienteil zu ermöglichen, wurde ein Navimec™ Module (vgl. [MEC10]) ausgewählt, welches wieder aufgrund von Designentscheidungen mit einem 10-poligen Leiterplattensteckverbinder mit der Mikrocontrollerplatine verbunden wurde. Die Steckerbelegung dafür ist in der Tabelle 3.5 zusammengefasst:

Pin	Funktion	Anmerkung
1	GND	Signalmasse
2	Right	SW1 - Button Right (Port 2.2)
3	GND	Signalmasse
4	Push	SW2 - Button Push (Port 2.0)
5	GND	Signalmasse
6	Down	SW5 - Button Down (Port 2.1)
7	GND	Signalmasse
8	Up	SW3 - Button Down (Port 2.3)
9	GND	Signalmasse
10	Left	SW4 - Button Left (Port 2.4)

Tabelle 3.5.: Navigationskit Steckerbelegung

Um ein Prellen der Taster des Navigationskits zu verhindern bzw. zu reduzieren, wurde eine hardwaremäßige Entprellung durch RC-Tiefpässe schaltungstechnisch vorgesehen.

Prinzip der Entprellung: Unter Prellen versteht man grundsätzlich mechanische Vibrationen, die durch schnelles Ein- und Ausschalten von mechanischen Schalter auftreten (vgl. [www10]). Um diesem Effekt entgegenzuwirken, kann bei einfachen Tastern ein simpler RC-Tiefpass verwendet werden (siehe Abbildung 3.8).

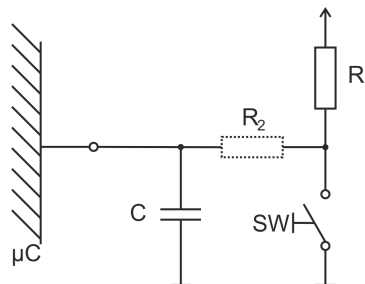


Abbildung 3.8.: Entprellung (hardwaremäßig)

3. Hardwareentwurf

Bei geöffnetem Schalter lädt sich der Kondensator langsam über den Widerstand R_1 auf die Betriebsspannung auf. Wird der Taster SW geschlossen, entlädt sich der Kondensator schnell (über R_2). Ein Prellen des Schalters (auf die Betriebsspannung) wird verhindert, da sich der Kondensator nur langsam über den Widerstand R_1 wieder aufladen kann (die Spannung am Kondensator kann sich nicht sprunghaft ändern). Die Zeitkonstanten für die Auf- bzw. Entladung des Kondensators werden durch die Werte der beiden Widerstände bestimmt. Da die Zeitkonstante für das Entladen (R_2C -Tiefpass) wesentlich kleiner als die Zeitkonstante für das Aufladen (R_1C -Tiefpass) sein muss, weist natürlich auch R_2 einen wesentlich kleineren Wert als R_1 auf.

4. Digitale Regelung

Dieses Kapitel beinhaltet eine Einleitung in den rechnerunterstützten Entwurf von Regelkreisen, definiert die für die Anwendung wichtigsten Standardregler bzw. deren Dimensionierung und diskutiert die Möglichkeit der adaptiven Reglereinstellung. Zum Abschluss werden die für den Klimaschrank gewählten Konzepte ausführlich begründet und auf Implementierungsdetails hingewiesen.

4.1. Einleitung

Grundsätzlich versteht man unter einem Regelkreis die zyklische Verarbeitung der Schritte **Messen**, **Vergleichen** und **Stellen**. Die prinzipielle Aufgabe eines Reglers ist es, eine Regelstrecke derart zu beeinflussen (durch Erzeugung einer Stellgröße $y(t)$), sodass eine Regelgröße $x(t)$ einer gegebenen Führungsgröße $w(t)$ mit einer möglichst kleinen Regelabweichung $e(t)$ folgt. Nicht zu vergessen sind Störgrößen, die in einem realen System auf die Regelstrecke einwirken. Das Strukturbild eines typischen Regelkreises ist in Abbildung 4.1 dargestellt.

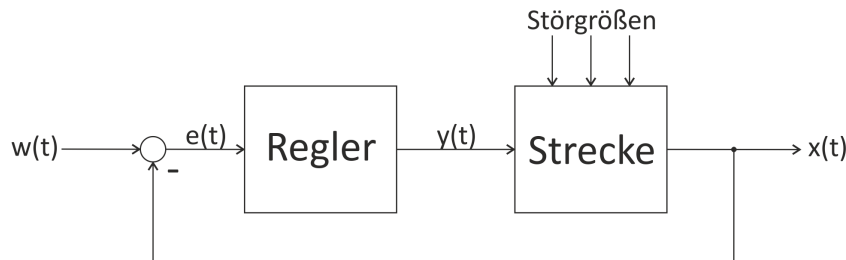


Abbildung 4.1.: Standardregelkreis (Strukturbild)

Im Falle der Regelung mit einem digitalen Rechnersystem ist das Strukturbild des Standardregelkreises um die Komponenten „A/D-Wandler“, „D/A-Wandler“ und „Sensoren“ zu erweitern. Wie im Strukturbild eines typischen digitalen Regelkreises (siehe Abbildung 4.2) dargestellt, übernimmt der Digitalrechner die Aufgaben des Reglers, indem er einen Regelalgorithmus ausführt. Er benötigt zusätzlich A/D- bzw. D/A-Wandler als Koppelgeräte, die dann Teile des Digitalrechners sind. Es ist zu berücksichtigen, dass diese Wandler nur zu bestimmten Zeitpunkten (Abtastperiode T_a) ihre

Eingangsgrößen verarbeiten können. Weiters benötigt der Digitalrechner noch geeignete Sensoren, um die Regelgröße aufzunehmen (vgl. [Rob10], [Horn and Dourdoumas, 2004], [Gausch et al., 1991]).

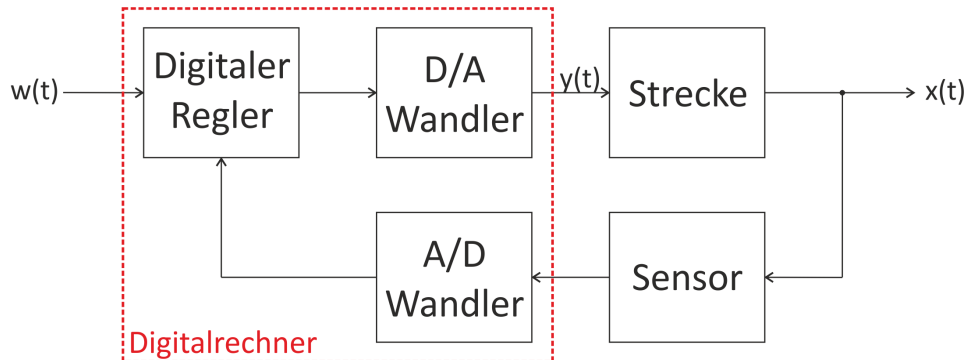


Abbildung 4.2.: Digitaler Regelkreis (Strukturbild)

4.2. Standardregler

In diesem Abschnitt soll ein kurzer Überblick über die wichtigsten Standardregler gegeben werden. Dabei soll eine Gegenüberstellung der analogen und digitalen Zusammenhänge vorgenommen werden.

4.2.1. P-Regler

Bei einem Proportionalregler wird die Regelabweichung $e(t)$ mit einem reellen Verstärkungsfaktor K_P multipliziert. Zwischen analoger und digitaler Betrachtung gilt folgender Zusammenhang:

$$e(t) = w(t) - x(t) \iff e[t] = w[t] - x[t]$$

$$y(t) = K_P \cdot e(t) \iff y[t] = K_P \cdot e[t]$$

Der P-Regler ist in der Anwendung sehr einfach und schnell, es tritt jedoch eine bleibende Regelabweichung auf.

4.2.2. I-Regler

Der Integralregler summiert die Regelabweichung $e(t)$ über die Zeit auf und skaliert diese Summe mit einem reellen Faktor K_I . Der Zusammenhang zwischen analoger und digitaler Betrachtung lautet:

4. Digitale Regelung

$$e(t) = w(t) - x(t) \iff e[t] = w[t] - x[t]$$
$$y(t) = K_I \int_0^t e(t) dt \iff \begin{cases} e_{sum}[t] = e_{sum}[t-1] + e[t] \\ y[t] = K_I \cdot T_a \cdot e_{sum}[t] \end{cases}$$

Je länger die Regelabweichung aufsummiert wird, desto größer wird die Stellgröße des I-Reglers. Im Vergleich zum P-Regler ist der I-Regler sehr langsam, er hat jedoch den Vorteil, dass die bleibende Regelabweichung vollständig eliminiert wird.

4.2.3. PI-Regler

Ein PI-Regler stellt eine Kombination aus einem Proportional- und Integralregler dar. Die schnelle Reaktion des Proportionalreglers und die exakte Ausregelung des Integralreglers zeichnet diesen Regler aus. Zwischen analoger und digitaler Betrachtung gelten folgende Zusammenhänge:

$$e(t) = w(t) - x(t) \iff e[t] = w[t] - x[t]$$
$$y(t) = K_P \cdot e(t) + K_I \int_0^t e(t) dt \iff \begin{cases} e_{sum}[t] = e_{sum}[t-1] + e[t] \\ y[t] = K_P \cdot e[t] + K_I \cdot T_a \cdot e_{sum}[t] \end{cases}$$

4.2.4. PID-Regler

Der PID-Regler fügt zum PI-Regler noch einen D-Anteil hinzu, der die Änderung der Regelabweichung bewertet. Der D-Anteil reagiert nicht auf die Höhe, sondern auf die Änderungsgeschwindigkeit der Regelabweichung. Auch hier gibt es wieder einen reellen Skalierungskoeffizienten K_D . Die analogen und digitalen Zusammenhänge lauten:

$$e(t) = w(t) - x(t) \iff e[t] = w[t] - x[t]$$
$$y(t) = K_P \cdot e(t) + K_I \int_0^t e(t) dt + K_D \frac{de(t)}{dt} \iff \begin{cases} e_{sum}[t] = e_{sum}[t-1] + e[t] \\ y[t] = K_P \cdot e[t] + K_I \cdot T_a \cdot e_{sum}[t] + K_D \frac{e[t] - e_{alt}[t]}{T_a} \\ e_{alt}[t] = e[t] \end{cases}$$

Ein PID-Regler (siehe Abbildung 4.3) stellt den klassischen Regler für die meisten Regelstrecken dar, indem er alle positiven Eigenschaften der einzelnen Komponenten kombiniert. Er ist somit sehr schnell und sehr genau.

4. Digitale Regelung

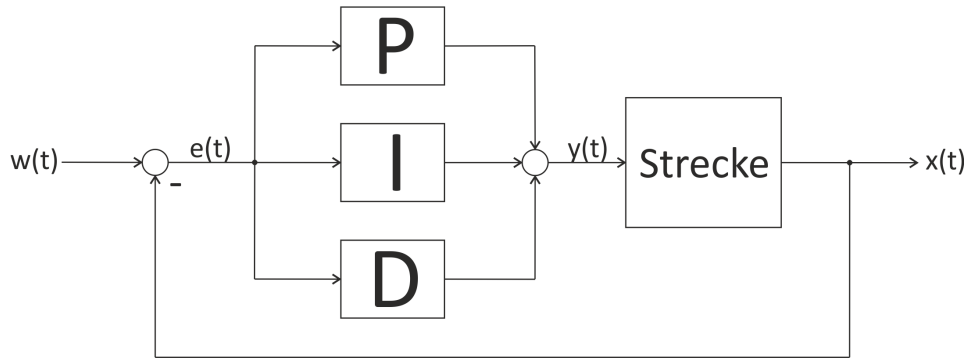


Abbildung 4.3.: PID-Regler (Strukturbild)

4.2.5. Vergleich der Reglertypen

Zusammenfassend kann gesagt werden, dass P-geregelte Kreise schnell sind, aber eine bleibende Regelabweichung aufweisen. Durch eine Kombination von P- und I-Regler kann die Regelabweichung eliminiert werden, jedoch darf die Begrenzung der Stellgröße bei realen Reglern nicht unterschätzt werden (vgl. Abschnitt 4.5). Die Verwendung eines D-Anteils erfolgt in Fällen schneller Dynamik. Für Strecken mit geringer Dynamik kann der D-Anteil entfallen.

4.3. Einstellregeln für Standardregler

Um für eine bestehende Regelstrecke einen Regler zu dimensionieren, kann einerseits aufgrund physikalischer Gegebenheiten ein **mathematisches Modell** abgebildet oder andererseits mit Hilfe von **empirischen Verfahren** eine Dimensionierung durchgeführt werden. Anstatt der Modellbildung liefern Experimente am System die notwendige Informationen über die Regelstrecke. Voraussetzung dafür ist die **Stabilität** des bestehenden Systems¹ (vgl. [Wik10b]).

Neben der „*T-Summenregel*“ stellen die „*Einstellregeln nach Ziegler-Nichols*“ eines der gängigsten empirischen Verfahren zur Dimensionierung von Standardreglern dar (vgl. [Horn and Dourdoumas, 2004, S. 223]). Dieses Verfahren ist durch die Festlegung der Begriffe „*Stabilität*“, „*Dynamisches Verhalten*“ und „*Stationäres Verhalten*“ wie folgt definiert:

¹Instabiles Verhalten darf keinen Schaden am System anrichten können.

4. Digitale Regelung

Stabilität: Die Voraussetzung für die Anwendung dieser Einstellregeln, ist eine Strecke mit bestimmten Merkmalen. Durch einfache Experimente an der Regelstrecke sollen relevante Kenngrößen ermittelt werden, die dann einen reduzierten Parametersatz (Ersatzgrößen) der Regelstrecke darstellen.

Dynamisches Verhalten: Die Idee der Einstellregeln nach „Ziegler-Nichols“ beruhen darauf, dass die Dämpfung des Regelkreises so eingestellt wird, dass die ersten beiden Maxima der Sprungantwort in einem Verhältnis von 4:1 stehen (siehe Abbildung 4.4):

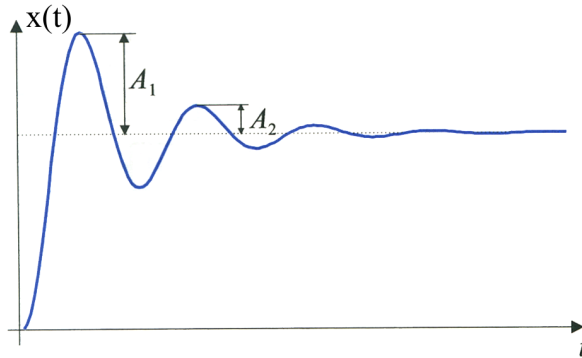


Abbildung 4.4.: Gewünschter Verlauf der Sprungantwort des Regelkreises (vgl. [Horn and Dourdoumas, 2004, S. 260])

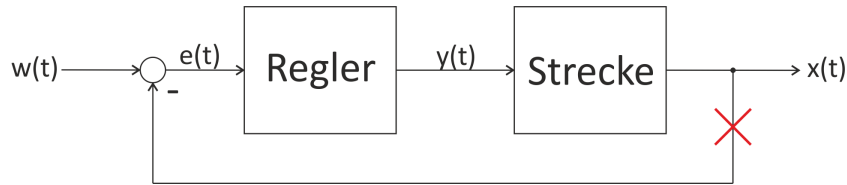
Stationäres Verhalten: Durch die Wahl des geeigneten Standardreglers kann das stationäre Verhalten des Regelkreises beeinflusst werden.

4.3.1. Open-Loop-Method

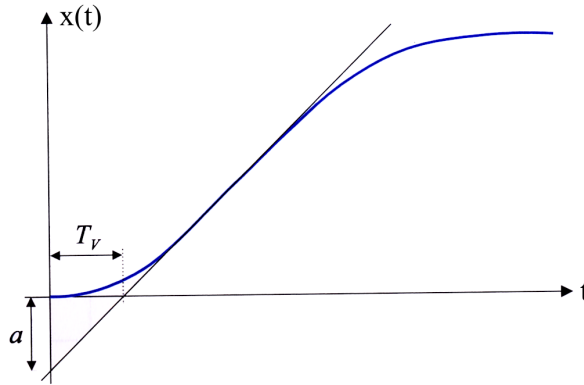
Diese Methode, die auch „*step response method*“ genannt wird, analysiert die Sprungantwort einer Strecke. Dafür muss der Regelkreis geöffnet werden (siehe Abbildung 4.5). Anwendung findet diese Methode beispielsweise bei Strecken, die nicht zum Schwingen gebracht werden können (vgl. [Horn and Dourdoumas, 2004, S. 260f]).

Durch das Anlegen der Wendetangente an die Sprungantwort, kann mit Hilfe der gewonnenen Kenngrößen a und T_V (Verzugszeit) der Regler dimensioniert werden (siehe Tabelle 4.1).

4. Digitale Regelung



(a) Strukturbild



(b) Sprungantwort des offenen Kreises (vgl. [Horn and Dourdoumas, 2004, S. 261])

Abbildung 4.5.: Open-Loop-Method

Reglertyp	K_P	T_I	T_D
P-Regler	a^{-1}	-	-
PI-Regler	$0.9 \cdot a^{-1}$	$3 \cdot T_V$	-
PID-Regler	$1.2 \cdot a^{-1}$	$2 \cdot T_V$	$0.5 \cdot T_V$

Tabelle 4.1.: Berechnungstabelle Open-Loop-Method

Für die Berechnung der Verstärkungsfaktoren (aus den obigen Kenngrößen) gelten folgende Zusammenhänge:

$$\begin{aligned}
 K_I &= \frac{K_P}{T_I} \\
 K_D &= K_P \cdot T_D
 \end{aligned}
 \tag{4.1}$$

Anmerkung: In der Literatur bezeichnet man K_P als den Proportionalbeiwert, T_I als die Nachstellzeit und T_D als die Vorhaltezeit.

4.3.2. Closed-Loop-Method

Bei dieser Methode, die auch „*frequency response method*“ genannt wird, wird der geschlossene Regelkreis untersucht und als Regler ein einfaches P-Glied verwendet. Es wird der Verstärkungsfaktor bei einer sprunghaften Änderung der Führungsgröße so lange erhöht, bis die Stabilitätsgrenze erreicht wird und der Regelkreis ungedämpft zu schwingen beginnt² (siehe Abbildungen 4.6) (vgl. [Horn and Dourdoumas, 2004, S. 263f]).

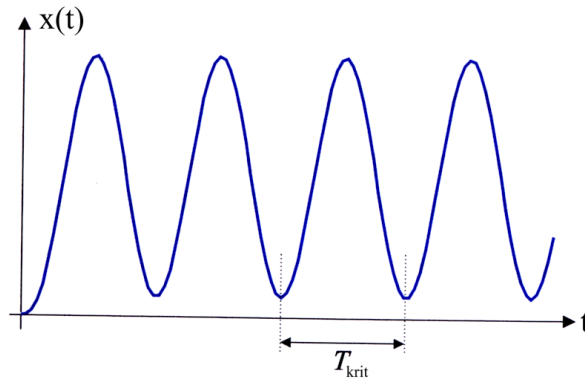
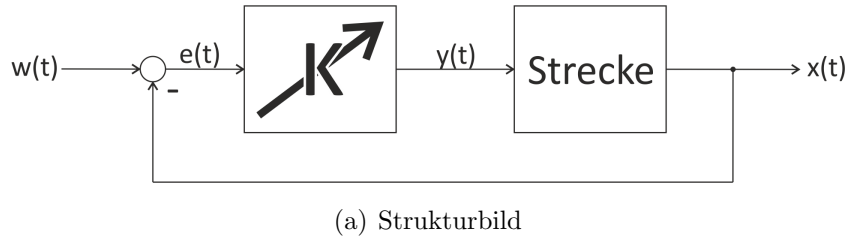


Abbildung 4.6.: Closed-Loop-Method

Diesen kritischen Verstärkungsfaktor bezeichnet man als K_{krit} . Die kritische Periodendauer der Dauerschwingung nennt man T_{krit} . Auch bei dieser Methode können durch die Tabelle 4.2 und den Zusammenhängen 4.1 die Dimensionierung der Reglerparameter vorgenommen werden.

²Voraussetzung dafür ist, dass die Strecke gefahrlos an ihre Stabilitätsgrenze gebracht und auch überschritten werden kann.

4. Digitale Regelung

Reglertyp	K_P	T_I	T_D
P-Regler	$0.5 \cdot K_{krit}$	-	-
PI-Regler	$0.4 \cdot K_{krit}$	$0.8 \cdot T_{krit}$	-
PID-Regler	$0.6 \cdot K_{krit}$	$0.5 \cdot T_{krit}$	$0.12 \cdot T_{krit}$

Tabelle 4.2.: Berechnungstabelle Closed-Loop-Methode

4.3.3. Zweipunktregler

Eine weitere Methode, die kritischen Parameter K_{krit} und T_{krit} zu ermitteln, besteht darin, als Regler ein „Zweipunktglied“ zu verwenden (siehe Abbildung 4.7) (vgl. [Yu, 2006], [Åström and Hägglund, 1995]).

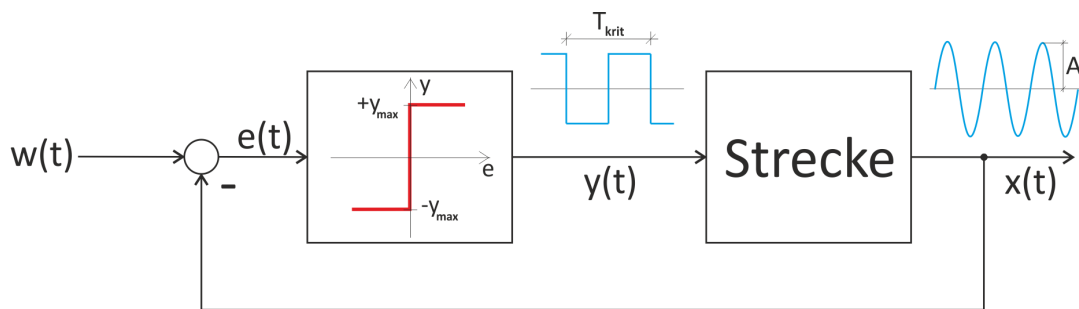


Abbildung 4.7.: Regelkreis mit Zweipunktglied (Strukturbild)

Das Ein- und Ausgangsverhalten eines Zweipunktgliedes ist durch folgende Relation charakterisiert:

$$y = \begin{cases} +y_{max} & \text{für } e \geq 0 \\ -y_{max} & \text{für } e < 0 \end{cases}$$

Die Maßnahme bewirkt eine Dauerschwingung um den gewählten Arbeitspunkt, deren Periodendauer näherungsweise T_{krit} ist. Die kritische Verstärkung K_{krit} kann aus der Amplitude A der Dauerschwingung und durch folgenden Zusammenhang³ ermittelt werden (vgl. [Horn and Dourdoumas, 2004, S. 265f]):

$$K_{krit} \approx \frac{4 \cdot y_{max}}{\pi \cdot A} \tag{4.2}$$

³Dieser Zusammenhang kann mit Hilfe der „Methode der Beschreibungsfunktion“ hergeleitet werden.

4.3.4. Beurteilung

Die Erfahrung zeigt, dass Dimensionierungen anhand der Einstellregeln nach *Ziegler-Nichols* zu einem schwach gedämpften Regelkreis mit geringer Stabilitätsreserve führen. Typischerweise ist die Verstärkung der geschlossenen Schleife um zwei- bis dreimal zu hoch. Weiters liefert die „*closed loop method*“ bessere Ergebnisse als die „*open loop method*“. Gründe dafür sind, dass die kritische Verstärkung klar definiert ist, es aber für die Verzögerungszeit viele verschiedene Definitionen gibt (vgl. [Åström and Hägglund, 1995, S. 142]).

4.4. Adaptive und selbsteinstellende Regler (Autotuning)

Automatisiert man den Vorgang der Beaufschlagung des Systems mit Testsignalen, Ermittlung der Ersatzgrößen und der anschließenden Berechnung der Reglerparameter, dann kann von „*Autotuning von Reglern*“ gesprochen werden. Genauere Details zum Thema „*Autotuning von PID-Reglern*“ können unter [Yu, 2006] nachgelesen werden. Das Prinzip der adaptiven Reglerauslegung ist in der Abbildung 4.8 veranschaulicht (vgl. [Schulz, 2008, S. 324f]).

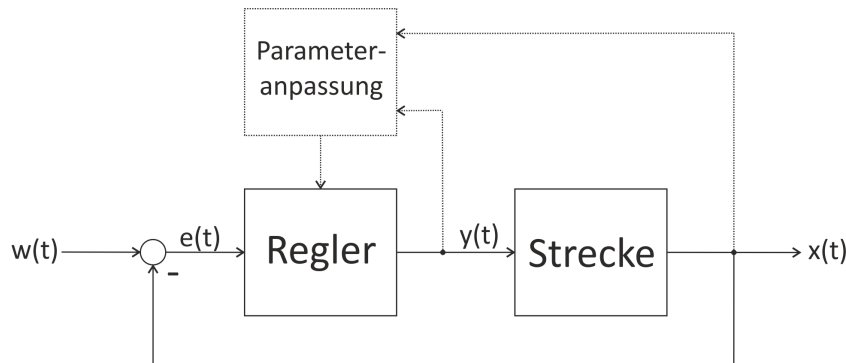


Abbildung 4.8.: Prinzip der adaptiven Regelung (Strukturbild)

Das System erhält eine weitere Rückführungsschleife, die für die Einstellung der Reglerparameter dient. Die Konfigurationsphasen für die adaptive bzw. selbsteinstellende Reglerdimensionierung lauten (vgl. [Horn and Dourdoumas, 2004, S. 265f]):

1. System in gewünschten Arbeitspunkt fahren
2. Regelkreis mit Zweipunktglied betreiben
3. Ersatzgrößen ermitteln, Reglerparameter nach „*Ziegler-Nichols*“ berechnen und Regler mit aktualisierten Parametern zuschalten

4. Digitale Regelung

In der Abbildung 4.9 ist die Anwendung der Konfigurationsphasen grafisch dargestellt.

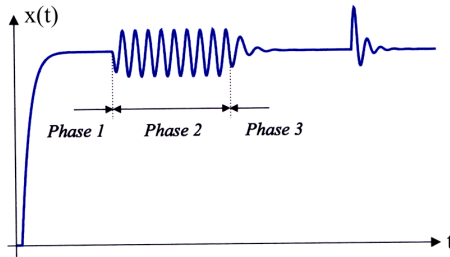


Abbildung 4.9.: Verlauf der adaptiven Reglereinstellung (vgl. [Horn and Dourdoumas, 2004, S.266])

4.5. Reglerüberlauf (Windup-Effekt)

Bei realen Reglern mit Integralanteil tritt bei Sollwertsprüngen praktisch immer eine Stellgrößenbeschränkung auf, d. h. dass der Stellgrößen-Ausgang des Reglers die physikalischen Grenzen des Stellglieds überschreitet. Tritt beispielsweise nach einem positiven Sollwertsprung zuerst eine positive Regelabweichung auf, so muss diese durch eine entsprechende negative Regelabweichung kompensiert werden. Es muss sprichwörtlich die „*Fläche unter der Kurve*“ zu Null werden, um die Regelabweichung zu eliminieren. Dieser Effekt kann dazu führen, dass der I-Anteil alleine einen Wert oberhalb der maximalen Stellgröße verursacht. Die Folge wären somit lange Ausgleichsvorgänge, d. h. größere und länger andauernde Regeldifferenzen (siehe Abbildung 4.10). Dieser Effekt wird als „*Windup*“ bezeichnet (vgl. [Schulz, 2008, S. 277]).

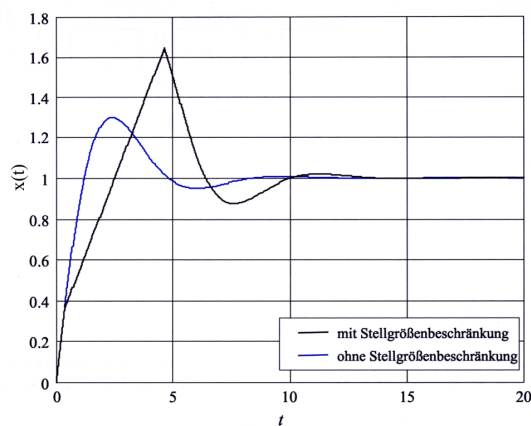


Abbildung 4.10.: Stellgrößenbeschränkung (vgl. [Horn and Dourdoumas, 2004, S. 199])

4. Digitale Regelung

Maßnahmen, um diesem unerwünschten Effekt entgegenzuwirken, werden als „*Anti-Windup-Maßnahmen*“ bezeichnet. Diese werden in den meisten Fällen realisiert, indem man den Integrator am Integrieren hindert, solange eine Stellgrößenbegrenzung vorliegt. Da die sprunghafte Abschaltung des Integrators im Prinzip eine unstetige Maßnahme darstellt, soll im Strukturbild der Abbildung 4.11 eine andere einfache Möglichkeit dargestellt werden.

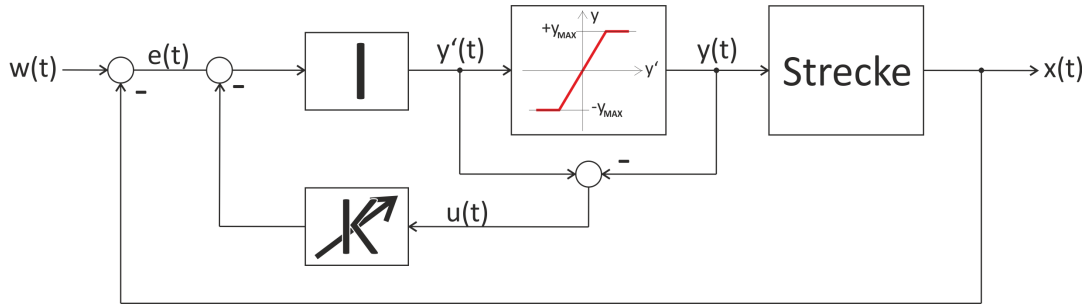


Abbildung 4.11.: Regler mit Anti-Windup-Maßnahmen (Strukturbild)

Es wird dabei die Differenz der Reglerausgangsgröße $y'(t)$ und dem beschränkten Signal $y(t)$ mit einem geeigneten Faktor K gewichtet und an den Eingang des Integrators zurückgeführt (vgl. [Horn and Dourdoumas, 2004, S. 198], [Zacher, 2000, S. 80]).

4.6. Gewähltes Konzept

Da es sich im Falle des Klimaschranks um ein **nichtlineares System** handelte, konnten die in den vorigen Abschnitten diskutierten Einstellregeln für Standardregler nicht ohne weitere Überlegungen verwendet werden. Es musste eine Art „*Linearisierung*“ des Systems durchgeführt werden, um die empirischen Verfahren nach „*Ziegler-Nichols*“ dennoch nützen zu können. Durch die Einführung von **Temperaturbereichen**, wurde das nichtlineare System in den Arbeitspunkten für die Berechnung der Reglerparameter als linear betrachtet:

Temperaturbereich	Arbeitspunkt
10°C - 20°C	15°C
20°C - 30°C	25°C
30°C - 40°C	35°C
40°C - 50°C	45°C

Tabelle 4.3.: Temperaturbereiche

Anmerkung: Da in den Spezifikationen des digitalen Reglers (vgl. Abschnitt 1.2) grundsätzlich nur die Stabilität des stationären Zustandes (Festwert- bzw. Endwertregelung) gefordert war und dafür eine möglichst einfache Reglerstruktur (mit adaptiver Einstellmöglichkeit) verwendet werden sollte, erwies sich das gewählte Konzept der Linearisierung als völlig ausreichend.

4.6.1. Digitales Reglermodell

Der, für den Klimaschrank gültige, vereinfachte digitale Regelkreis ist in Abbildung 4.12 dargestellt.

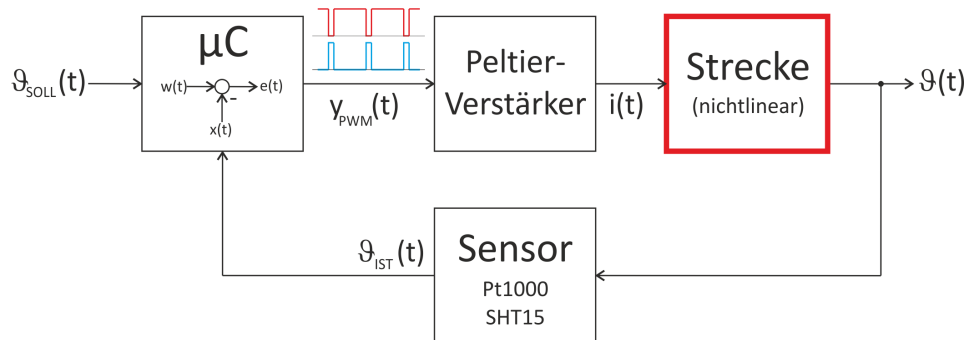


Abbildung 4.12.: Digitale Regelkreis des Klimaschranks (Strukturbild)

Im Vergleich zum digitalen Standardregelkreis in Abbildung 4.2 können die relevanten Parameter des Regelkreises definiert werden:

- Führungsgröße $w(t)$: Solltemperatur $\vartheta_{SOLL}(t)$
- Stellgröße $y(t)$: pulswertenmoduliertes Signal $y_{PWM}(t)$ bzw. $i(t)$ des Peltier-Verstärkers
- Regelgröße $x(t)$: Isttemperatur $\vartheta_{IST}(t)$

Kurzbeschreibung: Die Funktion des Digitalrechners übernimmt ein Mikrocontroller, der intern den Regelalgorithmus ausführt. Er bildet die Regelabweichung aus Soll- und Isttemperatur und erzeugt ein zur Regelabweichung proportionales pulswertenmoduliertes Signal. Dieses Signal wandelt der Peltier-Verstärker in einen proportionalen Strom um, der das Peltier-Element versorgt. In Abhängigkeit der Stromrichtung erzeugt das Peltier-Element inklusive Kühlkörper und Lüfter (siehe Abbildung 2.5) kalte oder warme Luft, die sich im Inneren des Klimaschranks verbreitet. Mit Hilfe der beiden Temperatursensoren (vgl. Abschnitt 3.1.2) kann der Mikrocontroller die aktuelle Temperatur im

Innenraum bzw. im Luftstrom des Peltier-Lüfter-Stacks aufnehmen und weiterverarbeiten.

4.6.2. Reglerauswahl

Für die Regelung der Temperatur des Klimaschranks wurde ein **PI-Regler mit Anti-Windup** (siehe Abbildung 4.13) ausgewählt.

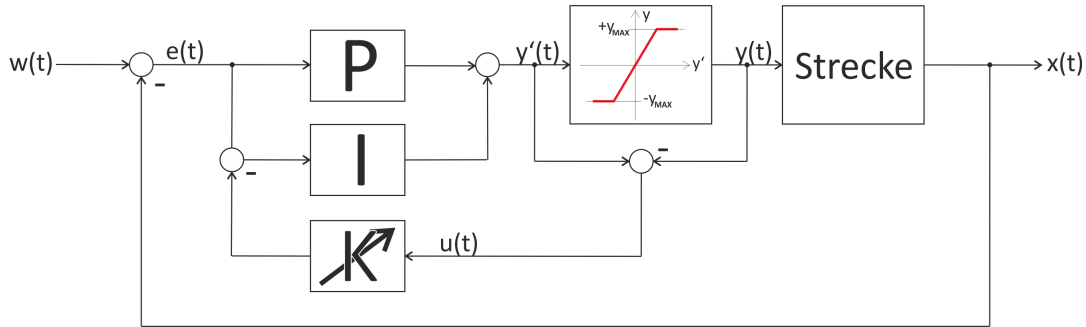


Abbildung 4.13.: PI-Regler mit Anti-Windup (Strukturbild)

Anmerkung: Da der Klimaschrank ein System mit einer hohen Totzeit bzw. sehr geringer Dynamik darstellte, konnte der D-Anteil entfallen (vgl. Abschnitt 4.2.5).

4.6.3. Reglerdimensionierung

Für die Bestimmung der Reglerparameter wurden folgende Schritte durchgeführt:

- **Zweipunktregelung** konfiguriert
- Analyse der Stell- und Regelgröße in den Arbeitspunkten (siehe Abschnitt 5.1)
- Ermittlung der kritischen Parameter K_{krit} und T_{krit} und Berechnung der Verstärkungsfaktoren
- **PI-Regler mit Anti-Windup** konfiguriert
- Analyse der Regelgröße (siehe Abbildungen 5.3 und 5.4)
- Auswahl der „optimalen“ Reglerparameter (siehe Tabelle 5.5)

Anmerkung: Im Zuge der Firmwareentwicklung (siehe Abschnitt 6.3.2) dienen die berechneten Reglerparameter für die Bereitstellung der Default-Reglereinstellung in der Temperaturbereichen.

4.6.4. Autotuning

Beim Modus der adaptiven Reglereinstellung wird der PI-Regler mit Anti-Windup auf einen **Regelkreis mit Zweipunktglied** umgeschaltet und das im Abschnitt 4.4 vorgestellte Verfahren angewandt. Da bei der adaptiven Reglereinstellung keine genaue Analyse der Stell- und Regelgröße möglich ist, wurden folgende Dimensionierungsvorschriften festgelegt:

- Für die Berechnung der kritischen Periodendauer T_{krit} wurde die mittlere auftretende Periodendauer der Stellgröße im gewählten Arbeitspunkt implementiert.
- Zur Berechnung der kritischen Verstärkung K_{krit} wurde der arithmetische Mittelwert aus der maximalen Über- und Unterschwingungsamplitude im gewählten Arbeitspunkt (virtuelle Nulllinie) ausgewählt. Aufgrund der im Abschnitt 4.3.4 erwähnten Problematik der Reglerdimensionierung nach *Ziegler-Nichols*, wonach die errechnete Verstärkung in den meisten Fällen zu hoch ausfällt, wurden nur 50% des errechneten Wertes für die Berechnung der Verstärkung des P-Anteils weiterverarbeitet⁴

Anmerkung: Die arithmetischen Mittelwerte der Periodendauer und der kritische Verstärkung werden aus 5 aufeinanderfolgenden Perioden der Regelgröße ermittelt (empirischer Wert).

⁴Diese Annahme wurde durch die Analyse der Regelvorgänge des 15°C Arbeitspunktes bestätigt, da sich dieser bei einer zu hohen Verstärkung als schwinganfällig herausstellte (vgl. Abschnitt 5.3).

5. Messungen und Berechnungen

In diesem Kapitel sind Messungen am Gesamtsystem bzw. daraus resultierende Berechnungen protokolliert. Als Temperatursensor wurde der am inneren Kühlkörper angebrachte Pt1000-Sensor ausgewählt.

Anmerkung: Prinzipiell könnten die gleichen Messungen auch mit dem SHT15-Sensor durchgeführt werden, doch liefert dieser Sensor, der im Inneren des Klimaschranks beliebig positioniert werden kann, keine reproduzierbaren Ergebnisse. Weiters sind die sehr geringe Dynamik und die extrem hohe Totzeit erschwerende Faktoren bei der Ermittlung der Kennlinien.

5.1. Zweipunktregelung

Durch die Anwendung der **Zweipunktregelung** (siehe Abbildung 4.7), konnten die Verläufe der Stell- (siehe Abbildung 5.1) und Regelgröße (siehe Abbildung 5.2) in den verschiedenen Temperaturarbeitspunkten aufgenommen werden:

5. Messungen und Berechnungen

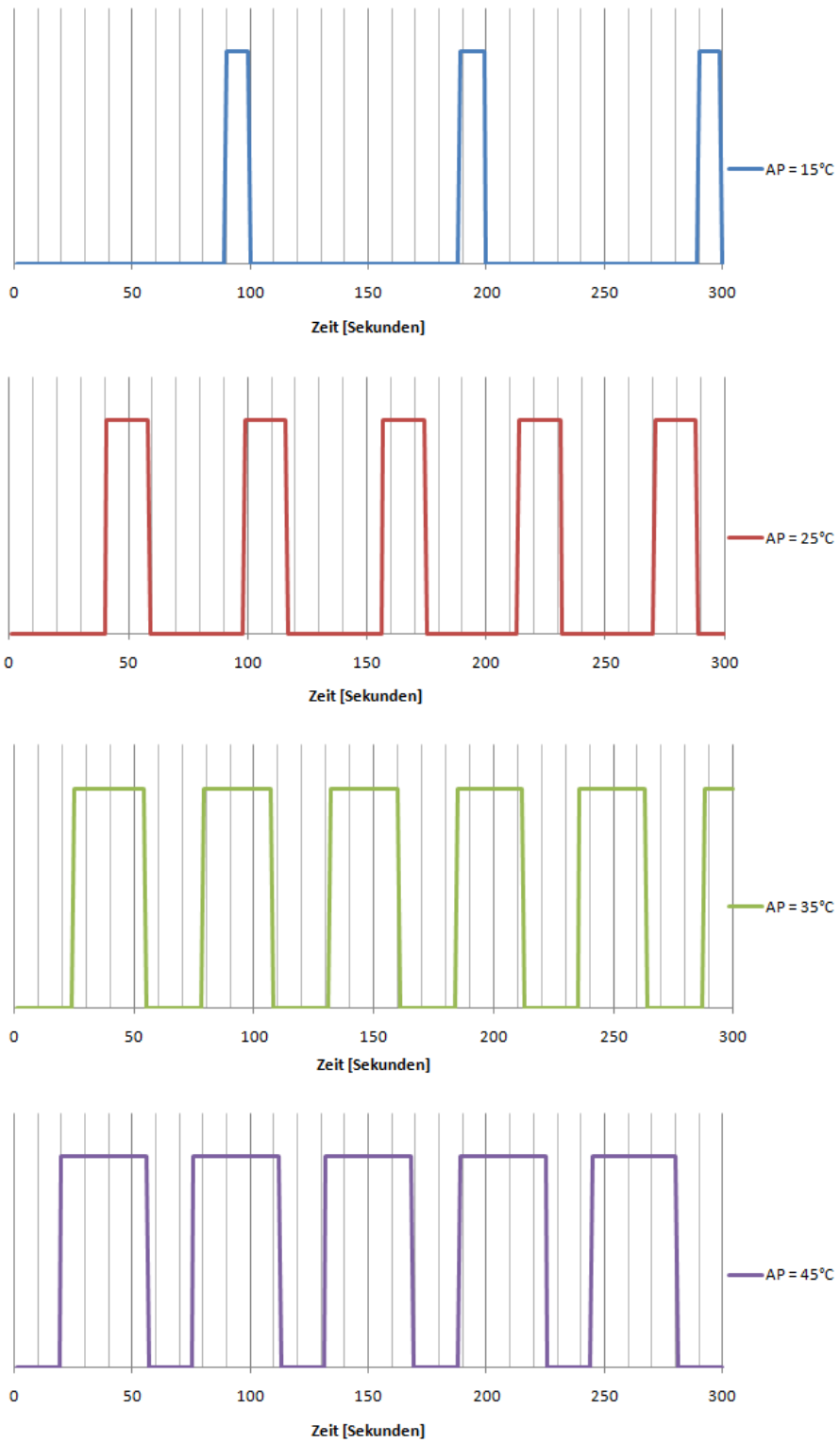


Abbildung 5.1.: Verlauf der Stellgröße in den Arbeitspunkten des Zweipunktreglers

5. Messungen und Berechnungen

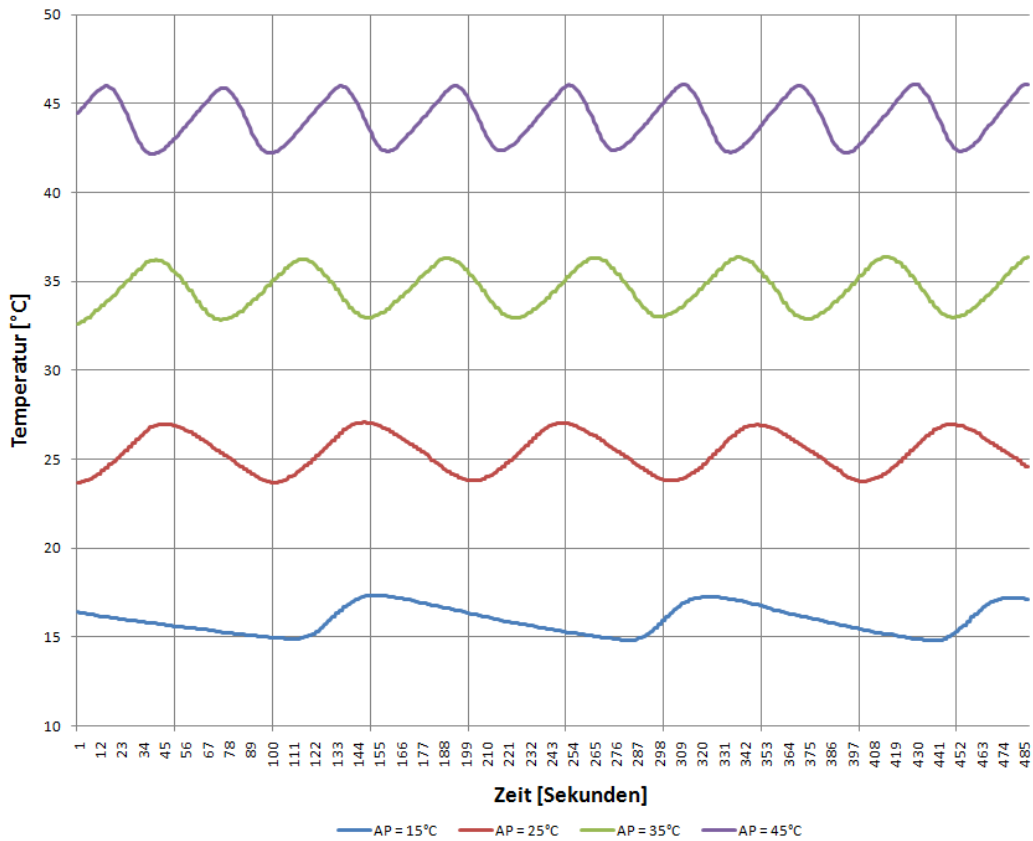


Abbildung 5.2.: Verlauf der Regelgröße in den Arbeitspunkten des Zweipunktreglers

Die aus den Verläufen der Stell- und Regelgröße ermittelten Werte für die kritische Periodendauer (T_{krit}) und der kritischen Amplitude (A_+ , A_- und \bar{A}) sind in folgender Tabelle zusammengefasst:

Arbeitspunkt	T_{krit}	A_+	A_-	\bar{A}
[°C]	[s]	[°C]	[°C]	[°C]
15	99	2.35	0.20	1.28
25	58	2.07	1.35	1.71
35	52	1.39	2.40	1.90
45	56	1.09	2.82	1.96

Tabelle 5.1.: Kritische Parameter der Stell- und Regelgrößen

5.2. Berechnung der Verstärkungsfaktoren

Aufgrund der Messergebnisse (siehe Tabelle 5.1), dem Zusammenhang 4.2 sowie der Berechnungstabelle 4.2 konnten die Reglerkoeffizienten K_P und K_I für die verschiedenen Temperaturarbeitspunkte errechnet werden. Jede Tabelle zeigt dabei die errechneten Koeffizienten bei der Verwendung einer bestimmten kritischen Amplitude (A_+ , A_- oder \bar{A}):

Arbeitspunkt	K_P	K_I
15°C	25.46	0.322
25°C	3.77	0.081
35°C	2.12	0.051
45°C	1.81	0.040

Tabelle 5.2.: Reglerkoeffizienten (Amplitude A_-)

Arbeitspunkt	K_P	K_I
15°C	3.98	0.050
25°C	2.98	0.064
35°C	2.68	0.064
45°C	2.60	0.058

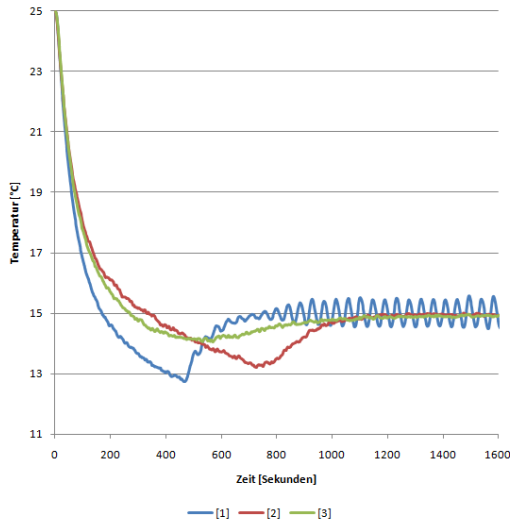
Tabelle 5.3.: Reglerkoeffizienten (Amplitude \bar{A})

Arbeitspunkt	K_P	K_I
15°C	2.17	0.027
25°C	2.46	0.053
35°C	3.66	0.088
45°C	4.67	0.104

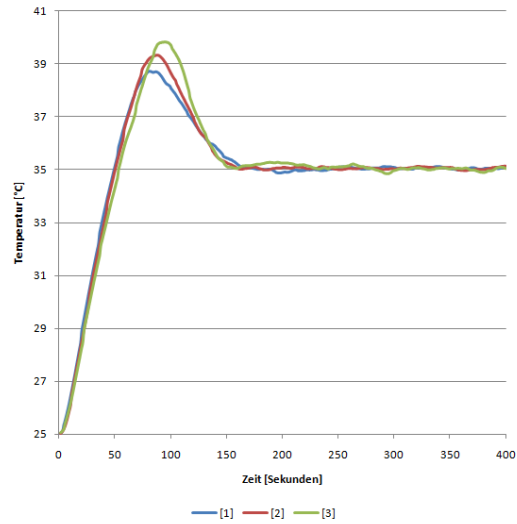
Tabelle 5.4.: Reglerkoeffizienten (Amplitude A_+)

5.3. Sollwertsprünge

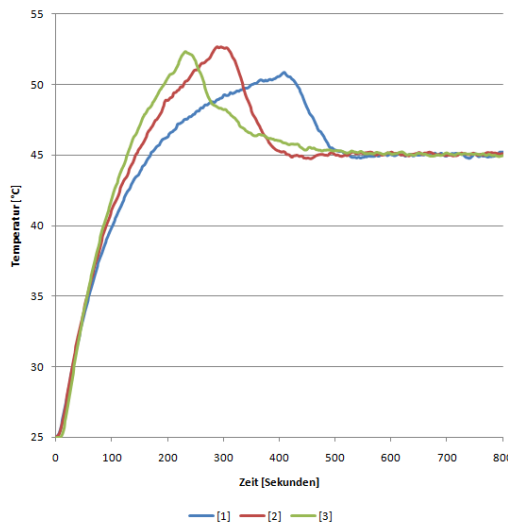
Dieser Abschnitt zeigt Antworten des **PI-Reglers mit Anti-Windup** auf Sollwertsprünge. Für die Konfiguration des Reglers wurden dabei die Reglerkoeffizienten einer bestimmten kritischen Amplitude (A_- siehe Tabelle 5.2, \bar{A} siehe Tabelle 5.3 und A_+ siehe Tabelle 5.4) in den Temperaturarbeitspunkten verwendet.



(a) Regelverlauf 15°C



(b) Regelverlauf 35°C



(c) Regelverlauf 45°C

Abbildung 5.3.: Regelverlauf in den Arbeitspunkten (Reglerkoeffizienten bei A_- [1], \bar{A} [2] und A_+ [3])

5. Messungen und Berechnungen

Die Antwort der Regelstrecke für den Temperaturarbeitspunkt von 25°C sind in der Abbildung 5.4 dargestellt. Es wurde dabei die „Richtung des Verlaufes“¹ gesondert untersucht.

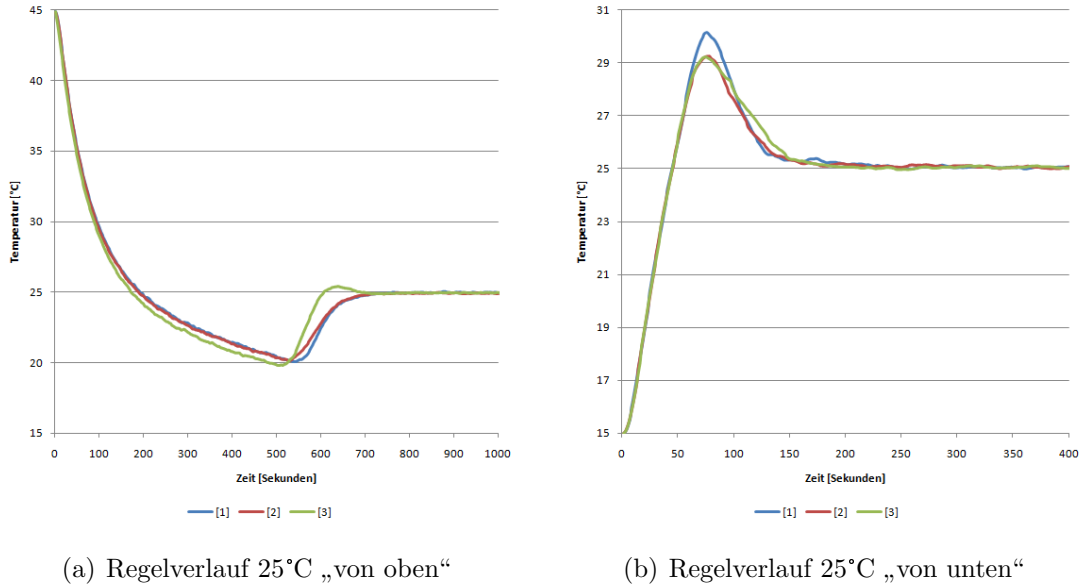


Abbildung 5.4.: Regelverlauf bei 25°C (Reglerkoeffizienten bei A_- [1], \bar{A} [2] und A_+ [3])

Durch eine Analyse der eben dargestellten Regelverläufe sind in der Tabelle 5.5 die gewählten Reglerkoeffizienten zusammengefasst.

Arbeitspunkt	K_P	K_I	Anmerkung
15°C	2.17	0.027	Amplitude A_+ [3]
25°C	2.98	0.064	Amplitude \bar{A} [2]
35°C	2.68	0.064	Amplitude \bar{A} [2]
45°C	2.60	0.058	Amplitude \bar{A} [2]

Tabelle 5.5.: Gewählte Reglerkoeffizienten (Pt1000-Sensor)

¹sprichwörtlich „von oben“ oder „von unten“

6. Firmware

In den folgenden Abschnitten soll ein Überblick über die verschiedenen Hardware- bzw. Software-Komponenten der Firmware und deren Implementierung gegeben werden.

6.1. Einleitung

Der MSP430 Mikrocontroller soll im Wesentlichen mit folgenden Komponenten¹ (siehe Abbildung 6.1) kommunizieren können:

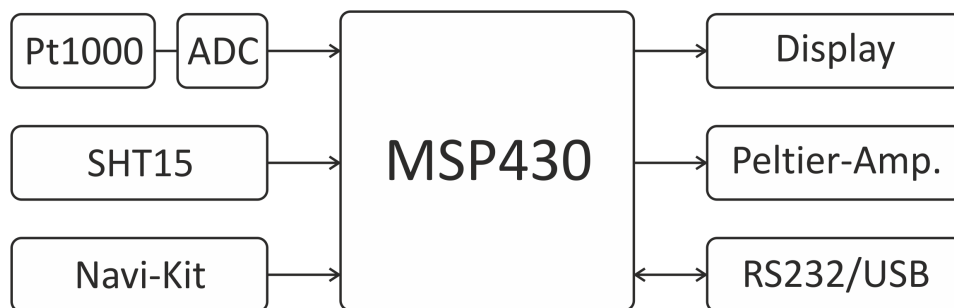


Abbildung 6.1.: Blockschaltbild der Hardwarekomponenten

Neben der Kommunikation mit der Hardware umfasst die Firmware noch weitere Komponenten, auf die in den folgenden Abschnitten näher eingegangen wird:

- Menüführung (User-Interface)
- Messungen bzw. Berechnungen
- Digitaler Regler (inkl. Autotuning)
- Kommunikationsprotokoll
- Flash-Speicherung

¹Die Eingabekomponenten sind in der Abbildung links bzw. die Ausgabekomponenten rechts dargestellt.

6.2. Hauptprogramm

Um eine „gleichzeitige“ Bearbeitung aller eben genannten integrierten Funktionen zu gewährleisten, musste für das Hauptprogramm der Firmware ein spezielles Konzept entwickelt werden, welches in den folgenden Abschnitten genauer erläutert wird.

6.2.1. State-Machine

Die Benutzereingaben des User-Interfaces werden durch eine State-Machine² (siehe Abbildung 6.2) verarbeitet. Der Wechsel von einem State in einen anderen State erfolgt jeweils beim Durchlauf der Hauptschleife. Um die Abfolge der Zustände besser verstehen zu können, sollen diese im Folgenden kurz erklärt werden:

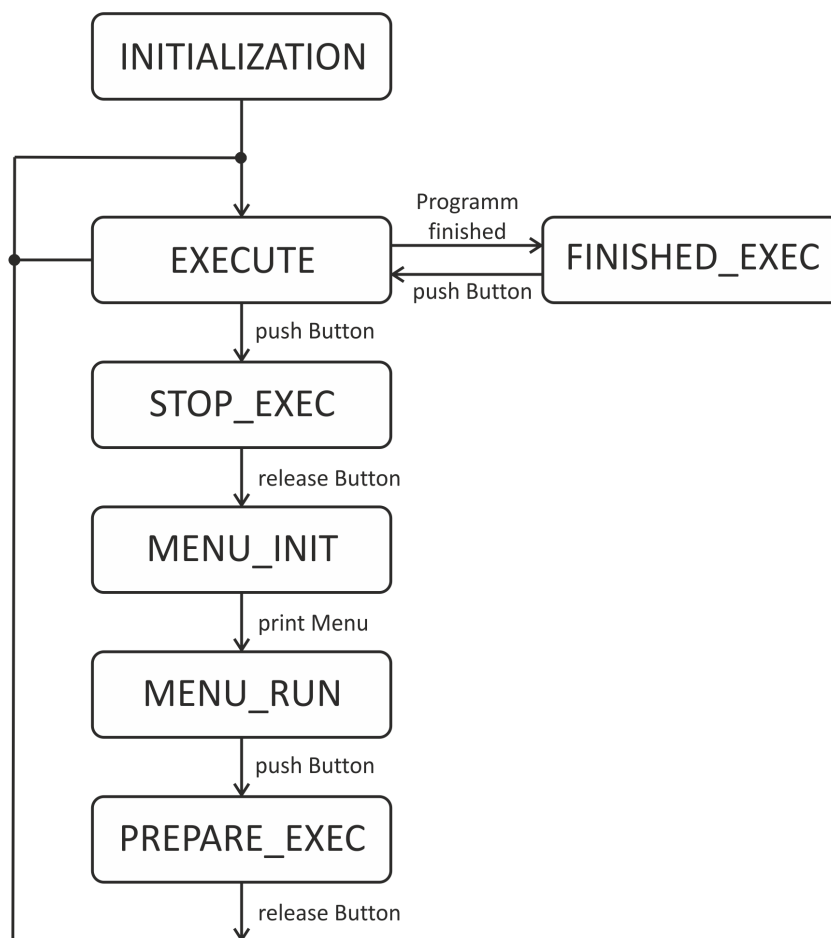


Abbildung 6.2.: State Machine

²Als Vorlage dafür diente [Frech, 2009].

INITIALIZATION: Hier werden folgende globale Einstellungen getroffen:

- Oszillator initialisieren
- Ports initialisieren
- SPI initialisieren (für Display)
- ADC initialisieren (für Pt1000 Messung)
- Flash Speicher initialisieren
- Timer B initialisieren (PWM)
- LCD initialisieren
- RS232 initialisieren
- aktuellen Regler (mit entsprechenden Parametern) festlegen
- aktuellen Sensor festlegen (Pt1000 oder SHT15)
- Menüpunkt festlegen (aktuelle Temperatur des aktuellen Sensors anzeigen)
- initialen Zustand festlegen (**EXECUTE**)

EXECUTE: Hier wird das ausgewählte Unterprogramm so lange ausgeführt, bis der Push-Button gedrückt wird (Wechsel in den **STOP_EXEC**-Zustand) oder selbst terminiert (Wechsel in den **FINISHED_EXEC**-Zustand). Eine Liste der möglichen Unterprogramme sind im Abschnitt 6.2.2 angeführt.

FINISHED_EXEC: Terminiert ein Programm selbsttätig, wird in diesen Zustand gewechselt. Nach dem Drücken des Push-Buttons wird wieder in den **EXECUTE**-Zustand gewechselt.

STOP_EXEC: Hier wird auf das Loslassen des Push-Buttons gewartet und danach in den **MENU_INIT**-Zustand gewechselt.

MENU_INIT: In diesem Zustand wird das Konfigurationsmenü ausgegeben und in den Zustand **MENU_RUN** gewechselt.

MENU_RUN: Hier werden die Aktionen des Navigationskits verarbeitet. Wird die Auswahl mit dem Push-Button bestätigt, wechselt die Hauptschleife in den **PREPARE_EXEC**-Zustand.

PREPARE_EXEC: Hier wird wieder auf das Loslassen des Push-Buttons gewartet, um in den **EXECUTE**-Zustand wechseln zu können.

6.2.2. Unterprogramme

Im Zustand **EXECUTE** des Hauptprogramms können verschiedene Unterprogramme ausgeführt werden. Diese sind:

displayPt1000Temperature: Dieses Unterprogramm zeigt die aktuelle Temperatur des Pt1000-Sensors auf $\frac{1}{100}$ °C genau am Hauptbildschirm an. Die Werte dafür werden im Hintergrund jede halbe Sekunde aktualisiert (vgl. Abbildung 6.3).

displaySHT15Temperature: Hier wird die aktuelle Temperatur des SHT15-Sensors auf $\frac{1}{100}$ °C genau auf dem Hauptbildschirm angezeigt. Der Wert wird im Hintergrund jede Sekunde aktualisiert (vgl. Abbildung 6.3).

displaySHT15Humid: Die aktuelle Luftfeuchte des SHT15-Sensors wird auf dem Hauptbildschirm als Zusatzinformation angezeigt. Die Aktualisierung des Wertes findet jede Sekunde statt (vgl. Abbildung 6.3).

displayDefinedTemp: Hier wird die gewünschte Solltemperatur auf dem Hauptbildschirm als Zusatzinformation angezeigt.

displayInformation: Mit Hilfe dieses Unterprogramms wird auf dem Hauptbildschirm der aktuell verwendete Sensor angezeigt. Zusätzlich wird während der Aktivität des Autotuning-Modus eine Statusmeldung darüber ausgegeben.

climaConfiguration: Der Benutzer definiert in diesem Unterprogramm mit Hilfe des Navigationskits die gewünschte Solltemperatur bzw. den gewünschten Modus für die digitale Regelung. Nachdem diese Informationen im Informationsspeicher (Segment A) des Mikrocontrollers abgelegt wurden, kann anschließend ein passender Regler ausgewählt werden. Nähere Informationen darüber können im Abschnitt 6.3.2 nachgelesen werden.

pt1000Calibration: Durch die Angabe eines Offsets³ und des tatsächlichen Wertes des Referenzwiderstandes kann eine Fehlerkorrektur des Pt1000-ADCs vorgenommen werden. Nähere Details zur Fehlerkorrektur können unter [Frech, 2009, S. 26f] nachgelesen werden. Zusätzlich werden diese beiden Informationen im Informationsspeicher (Segment B) des Mikrocontrollers abgespeichert.

³Korrekturwert, der bei der Berechnung der Summe der Impulse des externen ADCs abgezogen wird.

6.2.3. Messungen bzw. Berechnungen

Durch die Funktion `RunSelectedProgramm()` werden in der Hauptschleife des Hauptprogramms Messungen bzw. Berechnungen gestartet. Da diese nicht gleichzeitig stattfinden können, wurde ein dem Multitasking von Prozessen ähnliches Zeitscheiben-Verfahren entwickelt. Zentraler Bestandteil dieses Verfahrens ist ein globaler Zähler, der pro Durchlauf der Hauptschleife inkrementiert wird. Der Überlauf des Counters wurde dabei so gewählt, dass ein globales Intervall von einer Sekunde ($\hat{=}$ 22000 Durchläufe der Hauptschleife) entstand. Nun konnten zu bestimmten Zeitpunkten (= Zählerständen) Messungen bzw. Berechnungen durchgeführt werden, die sich in einem Intervall von einer halben Sekunde ($\hat{=}$ 11000 Durchläufe) oder einer ganzen Sekunde ($\hat{=}$ 22000 Durchläufe) wiederholten. Um einen Überblick über die zeitliche Abfolge der verschiedenen Messungen bzw. Berechnungen zu bekommen, sollen diese in einem Zeitstrahl (siehe Abbildung 6.3) dargestellt werden.

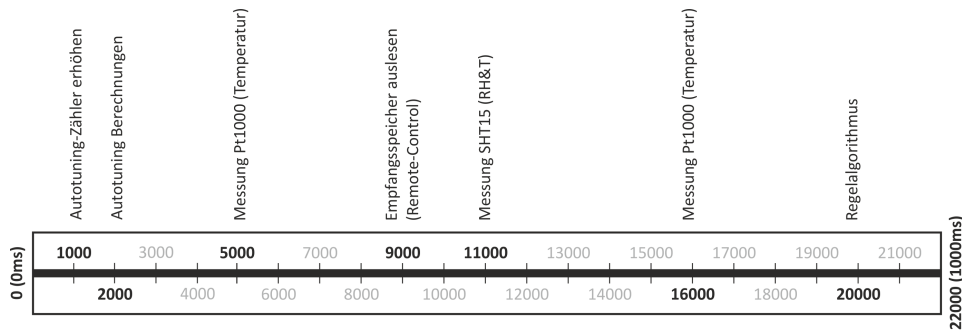


Abbildung 6.3.: Zeitstrahl

6.3. Implementierung

Bei der Implementierung konnten bestehende Bibliotheken anderer Projekte für die Entwicklung der Firmware verwendet werden:

Bibliothek	Autor	Files
Displayansteuerung	[Posch, 2009]	<code>lcd_new.c(.h)</code> , <code>fonts.c(.h)</code>
Timer A	[Frech, 2009]	<code>extern_adc.c(.h)</code> , <code>calibration.c(.h)</code>
SHT15	[Frech, 2009]	<code>sht.c(.h)</code>
Flashzugriff	[Frech, 2009]	<code>flash.c(.h)</code>

Tabelle 6.1.: Übernommene Bibliotheken

6.3.1. Timer B

Files: functions.c und functions.h

Da der Timer A bereits für die Ermittlung des Pt1000-Widerstandswertes seine Anwendung fand, wurde für die Generierung der pulswertenmodulierten Stellgröße der Timer B ausgewählt. Als Modus wurde der „Up-Mode“ definiert, bei dem die Periodendauer und das Tastverhältnis durch das Beschreiben der Register⁴ TBCCR0 und TBCCR2 festgelegt werden konnte (siehe Abbildung 6.4).

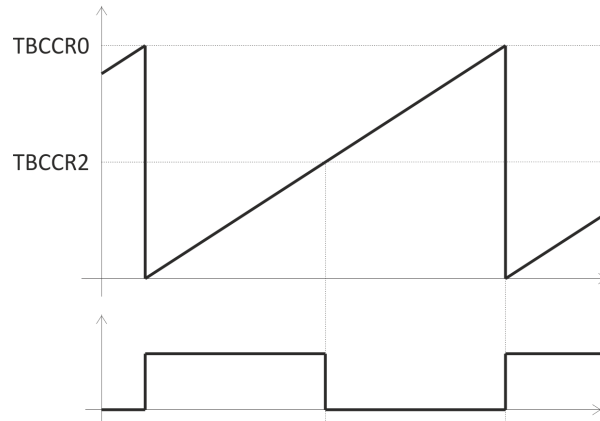


Abbildung 6.4.: Timer B Up-Mode

Die Initialisierung des Timers wurde in der Funktion `initTimerB()` implementiert:

Listing 6.1: Initialisierung Timer B

```

1 void initTimerB(void)
2 {
3     TBCTL = MC0;           // Stop timer B
4     TBCTL = TBSSEL_1 + TBCLR; // ACLK = 4MHz
5     TBCCR0 = DIGITS_RANGE; // PWM Period
6     TBCCTL2 = OUTMOD_7;   // CCR2 reset/set
7     P4DIR |= BIT2;        // P4 output 0x04 (Port 4.2)
8     P4SEL |= BIT2;        // P4 option 0x04 (Port 4.2)
9     TBCTL |= MC0;         // Start Timer_B in Up-Mode
10 }

```

⁴Dieses Register ist hardwaremäßig mit dem Port 4.2 verknüpft.

Um das Tastverhältnis der pulsweitenmodulierten Stellgröße „*online*“ ändern zu können, wurde die Funktion `setDutyCycle(int duty)` implementiert:

Listing 6.2: Timer B Duty-Cycle

```

1 void setDutyCycle(int duty)
2 {
3     TBCCR2 = duty;
4 }

```

Berechnung: Für die Frequenz der pulsweitenmodulierten Stellgröße wurden 250Hz ⁵ gewählt. Aus dem gewählten Wert ergab sich eine Periodendauer von $T_{PWM} = \frac{1}{f_{PWM}} = \frac{1}{250} = 4\text{ms}$. Um das Register TBCCR0 für die Periodendauer entsprechend setzen zu können, wurde folgende Berechnung durchgeführt ($T_{PWM} = 4\text{ms}$, $f_{CLK} = 4\text{MHz}$):

$$T_{PWM} = \frac{1}{f_{CLK}} \cdot N_{Impulse}$$

$$\Rightarrow N_{Impulse} = T_{PWM} \cdot f_{CLK} = 4 \cdot 10^{-3} \cdot 4 \cdot 10^6 = 16000 \text{ Impulse}$$

Wie bereits erwähnt, muss für die Erzeugung eines Tastverhältnisses das TBCCR2-Registers entsprechend gesetzt werden. Für die mathematische Berechnung wurden folgende Überlegungen angestellt:

- Der Wert kann nur in einem Bereich von 0 bis 16000 Impulse liegen (vgl. Tabelle 2.1).
- Der Arbeitsbereich des Klimaschranks wurde in einem Bereich von 10°C bis 50°C definiert (vgl. Abschnitt 1.2).
- Die minimale und maximale Impulsbreite für den Peltier-Verstärker muss berücksichtigt werden.

$$D = \frac{N_{Impulse}}{\Delta\vartheta} = \frac{16000}{50^\circ\text{C} - 10^\circ\text{C}} = 400 \text{ Impulse}/^\circ\text{C}$$

Anmerkung: Wie bereits erwähnt, muss das Tastverhältnis der pulsweitenmodulierten Stellgröße, um vom Peltier-Verstärker verarbeitet werden zu können, Flanken beinhalten. 0% bzw. 100% Tastverhältnis würde der Peltier-Verstärker als Fehler interpretieren.

⁵Diese Annahme beruht auf die Angabe von [Plattner, 2009, S. 13], der dafür eine minimale Frequenz von 15Hz definierte.

Aufgrund dieser Überlegungen wurden folgende Grenzen (durch Messung) festgelegt:

$I_{PELTIER}$	ΔT_{Impuls}	D
[A]	[μ s]	[Impulse]
-10A	100	400
+10A	3900	15600

Tabelle 6.2.: Grenzen der Stellgröße

6.3.2. Digitaler Regler

Files: `control.c` und `control.h`

Als Modell für die digitale Regelung diente der **PI- Regler mit Anti-Windup** (siehe Abbildung 4.13). Die vereinfachte Implementierung des Reglers lautet:

Listing 6.3: Implementierung PI-Controller

```

1 void PIController(struct PConfiguration* pi)
2 {
3     pi->e = w - x;
4     pi->esum = pi->esum + pi->e;
5
6     pPart = pi->kp * pi->e;
7     iPart = (pi->ki * pi->Ta * pi->esum) - pi->ka * pi->u;
8
9     yStrich = calculatePwmValue(pPart + iPart);
10    pi->y = limitRange(yStrich);
11    setDutyCycle( pi->y );
12
13    pi->u = calculateTempValue(yStrich - pi->y);
14 }

```

Anmerkungen:

- Um die pulswertenmodulierte Stellgröße des Reglers generieren zu können, muss aus der Summe des P- und I-Anteils ein entsprechendes Tastverhältnis erzeugt werden. Dies wird in der Funktion `calculatePwmValue()` (siehe Code 6.4) durchgeführt.

6. Firmware

- Durch Aufruf der Funktion `setDutyCycle(duty)` wird die pulsweitenmodulierte Stellgröße aus den errechneten P- und I-Anteilen des digitalen Reglers gesetzt.
- Die Funktion `limitRange()` dient dazu, die Grenzen des pulsweitenmodulierten Signals (vgl. Grenzen 6.2) einzuhalten.

Listing 6.4: Erzeugung der pulsweitenmodulierten Stellgröße

```
1 float calculatePwmValue(float value)
2 {
3     float yPwm = 0;
4
5     yPwm = DIGITS_RANGE/2 + (value * DIGITS_PER_DEGREE);
6
7     return yPwm;
8 }
```

Die Reglerfunktion `PIController()` wird in der Hauptschleife des Hauptprogramms einmal pro Sekunde ausgeführt (vgl. Abbildung 6.3), d. h. pro Sekunde wird vom digitalen Regler eine neue Stellgröße ermittelt.

Wie bereits im Abschnitt 6.2.2 erwähnt, kann der Benutzer durch das User-Interface (Unterprogramm `climaConfiguration`) den Regler durch die Definition einer neuen Solltemperatur und eines entsprechenden Modus konfigurieren. Der Modus legt dabei den Temperatursensor und den Regelalgorithmus fest (siehe Tabelle 6.3).

Modus	Temperatursensor	Regelalgorithmus
1	Pt1000	PI- Regler mit Anti-Windup
2	SHT15	PI- Regler mit Anti-Windup
3	Pt1000	Zweipunktregler (Autotuning)
4	SHT15	Zweipunktregler (Autotuning)

Tabelle 6.3.: Reglermodi

Im Falle der ersten beiden Modi ist für jede gewünschte Solltemperatur anhand der festgelegten Temperaturbereiche (vgl. Tabelle 4.3) eine passende Reglerkonfigurationen zugeordnet. Diese werden zur Compilezeit mit den errechneten Werten anhand der Tabelle 5.5 gesetzt.

Grundsätzlich besteht eine Reglerkonfiguration aus den folgenden Parametern:

- Abtastzeit: T_a
- Verstärkungsfaktor P-Anteil: k_p
- Verstärkungsfaktor I-Anteil: k_i
- Beschränkungsfaktor für den Integrator: k_a
- Regelabweichung: e
- Summe der Regelabweichung über die Zeit: e_{sum}
- Stellgröße: y
- Ausgangsgröße des Stellgrößenbeschränkers: u

6.3.3. Autotuning

Files: `control.c`, `control.h`, `autotuning.c` und `autotuning.h`

Definiert der Benutzer im User-Interface eine Solltemperatur und einen Modus für Autotuning (vgl. Tabelle 6.3), wird der digitale Regler vom PI-Regler mit Anti-Windup auf einen **Regelkreis mit Zweipunktglied** umgeschaltet. Die Implementierung dieses Reglers wird in der Funktion `twoStepController()` durchgeführt:

Listing 6.5: Zweipunktregler

```

1 void twoStepController(void)
2 {
3     e = w - x;
4
5     if(e >= 0)
6         y = DIGITS_RANGE;    // heating
7
8     else if(e < 0)
9         y = 0;                // cooling
10
11     setDutyCycle( limitRange(y) );
12 }
```

Nun wird anstatt des PI-Regler mit Anti-Windup der Regelkreis mit Zweipunktglied in der Hauptschleife des Hauptprogramms einmal pro Sekunde ausgeführt. Gleichzeitig wird zum globalen Zähler ein zusätzlicher Sekundenzähler für die Periodendauermessung hinzugeschaltet, der bei jedem Überlauf des Hauptzählers inkrementiert wird (vgl. Zeitstrahl 6.3). Nach der Ermittlung des ersten Nulldurchgangs⁶ werden aus den nächsten 5 Perioden der Dauerschwingung durch die Funktion `measureValues()` folgende Parameter ermittelt:

- **amplitude**: Mittelwert aus den Amplituden der minimalen (`minTemp`) und maximalen Abweichung (`maxTemp`) um den Temperaturarbeitspunkt.
- **period**: Arithmetischer Mittelwert aus kleinster (`minPeriod`) und größter (`maxPeriod`) Periodendauer der Dauerschwingung.

Durch die Funktion `calculateParameters()` können nach der Ermittlung der eben genannten Parameter die aktuellen, für die gewünschte Solltemperatur errechneten, Reglerparameter errechnet und in den Reglerkonfigurationen gespeichert werden. Als Vorlage dafür dienen die im Abschnitt 4.6.4 definierten Dimensionierungsvorschriften. Um nicht die vorkonfigurierten Reglerkonfigurationen zu überschreiben, stehen für jeden Temperaturbereich und Sensor leere Reglerkonfigurationen zur Verfügung, die durch das Autotuning gesetzt werden können.

Nach Abschluss dieses Verfahrens wird anstelle des Zweipunktreglers wieder der PI-Regler mit Anti-Windup ausgeführt, der die aktuell ermittelten Reglerparameter für die Regelung anwendet.

6.3.4. Serielle Datenübertragung

Files: `rs232.c` und `rs232.h`

Für die Kommunikation mit dem Mikrocontroller wurde die serielle Schnittstelle im USART1-Modus⁷ implementiert. Die Funktion `initRS232()` führt die Initialisierung durch:

⁶Darunter versteht man das Erreichen der Solltemperatur „von oben herab“, d. h. es wird gewartet, bis der fallende Verlauf der Dauerschwingung die Solltemperatur unterschreitet.

⁷USART0 wurde für die SPI-Kommunikation mit den Display verwendet.

Listing 6.6: Initialisierung der seriellen Schnittstelle

```

1 void initRS232(void)
2 {
3     P3SEL |= BIT6;           // P3.6 = USART1 (TxD)
4     P3SEL |= BIT7;           // P3.7 = USART1 (RxD)
5     P3DIR |= BIT6;           // Tx Pin (21) Output direction
6     P3DIR &= ~BIT7;          // Rx Pin (22) Input direction
7
8     U1CTL |= CHAR + SWRST;    // 8-bit character + Software reset enable
9     U1TCTL |= SSEL0;          // ACLK (4MHz)
10
11    // Baud Rate: 9600 Baud from ACLK
12    U1BR0 = 0xA0;             // Baud Rate LSW
13    U1BR1 = 0x01;             // Baud Rate MSW
14    U1MCTL = 0x6D;            // Modulation Register
15
16    ME2 |= UTXE1 + URXE1;     // Enable USART1 TXD/RXD
17    IE2 |= URXIE1;            // Enable USART1 RXD Interrupt
18
19    U1CTL &= ~SWRST;          // Software reset disable
20
21    IE2 = 0x10;                // URXIE1 enabled
22 }

```

Anmerkung: Bei der Konfiguration der Baudrate wurde für die Berechnung der Registerwerte der *MSP430 UART Register Calculator* verwendet (vgl. [Day10]).

Für die Übertragung der Daten wurden zwei unterschiedliche Funktionen implementiert. Während die Funktionen `transmitChar(char data)` einzelne Characters überträgt, können mit der Funktion `transmitFloat(float value)` Float-Messwerte übertragen werden. Der wesentliche Teil der Implementierung beider Funktionen lautet:

Listing 6.7: Prinzip des seriellen Sendevorgangs

```

1 for(i = 0; i < size; i++)
2 {
3     U1TXBUF = buffer[i];
4     while((U1TCTL & TXEPT) == 0);
5 }

```

Für den Datenempfang wurde die ISR⁸ des USART1-Modus implementiert:

Listing 6.8: ISR der seriellen Schnittstelle

```

1 void rs232_rx_isr(void) __interrupt [USART1RX_VECTOR]
2 {
3     receivedChar = U1RXBUF;
4 }

```

6.3.5. Fernsteuerung

Files: `remoteControl.c` und `remoteControl.h`

Um eine Art Fernsteuerung für zukünftige Erweiterungen zu ermöglichen, wurde ein Miniaturprotokoll auf Basis der seriellen Übertragung implementiert. Damit können aktuelle Messwerte übertragen und neue Solltemperaturen bzw. Modi konfiguriert werden. Eine Übersicht über die implementierten Befehle ist in der Tabelle 6.4 dargestellt.

Befehl	Beschreibung
*1	Die aktuelle Temperatur des Pt1000-Sensors wird retourniert.
*2	Die aktuelle Temperatur des SHT15-Sensors wird retourniert.
#[_][_][_][_]1	Es wird eine neue Solltemperatur gesetzt und der Regler in den Modus 1 geschaltet.
#[_][_][_][_]2	Es wird eine neue Solltemperatur gesetzt und der Regler in den Modus 2 geschaltet.
#[_][_][_][_]3	Es wird eine neue Solltemperatur gesetzt und der Regler in den Modus 3 geschaltet.
#[_][_][_][_]4	Es wird eine neue Solltemperatur gesetzt und der Regler in den Modus 4 geschaltet.

Tabelle 6.4.: Befehle der Fernsteuerung

Anmerkung: Um eine neue Solltemperatur zu definieren, muss zwischen dem Startzeichens (#) und dem Endzeichen, welches den Reglermodus darstellt, der neue Sollwert als Gleitkommazahl⁹ übertragen werden. Es ist darauf zu beachten, dass die 4 Bytes der Gleitkommazahl hintereinander gesendet werden.

⁸Interrupt Service Routine

⁹Wird je nach Programmiersprache anderes bezeichnet - in den meisten Fällen aber *Float* oder *Single*.

Implementierungsdetails: Wie bereits im vorigen Abschnitt erwähnt, ist eine ISR für den Empfang der Befehle des seriellen Interfaces zuständig. Da die komplette Behandlung der Befehle in der ISR die zeitkritische Verarbeitung der Messschaltung (ISR des Timer A) beeinflusste, wurde ein Empfangsspeicher („Ringbuffer“) implementiert. Dieser wird durch die ISR des seriellen Interfaces mit den empfangenen Character beschrieben. Eine Variable, die die Anzahl der im Ringbuffer gespeicherten Character anzeigt, signalisiert der Hauptschleife des Hauptprogrammes neue empfangene Befehle. Durch die Funktion `receiveData()`, die in der Hauptschleife jede Sekunde ausgeführt wird (vgl. Zeitstrahl 6.3), können die Daten aus dem globalen Empfangsspeicher verarbeitet werden. Diese Lösung erlaubt, dass auch schnell aufeinanderfolgende Befehle problemlos entgegengenommen werden können.

Anmerkung: Die Größe des Ringbuffers wurde mit 20 Einträgen festgelegt. Da die Verarbeitung der empfangenen Characters in der Hauptschleife nur jede Sekunde durchgeführt wird, dürfen pro Sekunde nicht mehr als 20 Characters übertragen werden.

6.3.6. Flash-Speicherung

Files: `flash.c` und `flash.h`

Da die Parameter der Pt1000-Kalibrierung bzw. die Parameter der Regelung auch nach einem Neustart des Systems erhalten bleiben sollen, wurden Funktionen implementiert, die eine Kommunikation mit dem Flash-Speicher ermöglichen.

Grundsätzlich besteht der Flash-Speicher des verwendeten Mikrocontrollers aus 256 Segmenten Hauptspeicher (à 512 bytes) und aus zwei Segmenten Informationsspeicher (à 128 bytes). Der Informationsspeicher des verwendeten Mikrocontrollers ist folgendermaßen aufgebaut (siehe Tabelle 6.5):

Segment	von	bis	Größe
B	01000h	0107Fh	128 bytes
A	01080h	010FFh	128 bytes

Tabelle 6.5.: Aufbau des Informationsspeichers

Da der Informationsspeicher prinzipiell byteweise gelesen, doch immer nur segmentweise beschrieben bzw. gelöscht werden kann, wurde folgende Speicherzuordnung gewählt (siehe Tabelle 6.6). Es werden somit in das Segment B die Parameter der Kalibrierung des Pt1000 und in das Segment A die Parameter der Regelung gespeichert.

6. Firmware

Segment	Adresse	Länge	Beschreibung
B	01000h	4 Byte	Offset des ADCs (long)
B	01020h	4 Byte	Wert des Referenzwiderstandes (float)
A	01080h	4 Byte	Solltemperatur der Regelung (float)
A	010A0h	1 Byte	Aktueller Temperatursensor (unsigned char)

Tabelle 6.6.: Speicherzuweisung des Informationsspeichers

Eine grundlegende Implementierung eines Flash- Schreibzugriffes bzw. eines Flash-Lesezugriffes sind in den folgenden Codeteilen dargestellt:

Listing 6.9: Flash-Schreibzugriff

```

1 void flash_write_float(unsigned int address, float value)
2 {
3     float *Flash_ptr;           // Flash pointer
4     Flash_ptr = (float *) address; // Initialize Flash pointer
5
6     FCTL1 = FWKEY + ERASE;      // Set Erase bit
7     FCTL3 = FWKEY;             // Clear Lock bit
8     *Flash_ptr = 0;            // Dummy write to erase segment
9
10    while (! (FCTL3 & WAIT) );
11
12    FCTL1 = FWKEY + WRT;        // Set WRT bit for write operation
13    *Flash_ptr = value;         // Write value to flash
14
15    while (! (FCTL3 & WAIT) );
16
17    FCTL1 = FWKEY;              // Clear WRT bit
18    FCTL3 = FWKEY + LOCK;       // Reset LOCK bit
19 }

```


6. Firmware

Listing 6.10: Flash- Lesezugriff

```
1 float flash_read_float(unsigned int address, float min, float max,  
2 float initial)  
3 {  
4     float value = 0.0;  
5     value = *((float*)address);  
6     if(value < min || value > max)  
7         value = initial;  
8     return value;  
9 }
```

7. Ausblick

Im Zuge der Entwicklung der einzelnen Komponenten bzw. beim Verfassen dieser schriftlichen Masterarbeit konnten Erweiterungen bzw. Verbesserungen der bestehenden Lösungen gefunden werden. Dieses Kapitel soll einige Punkte konkret ansprechen, die nicht mehr oder nur zum Teil umgesetzt werden konnten.

7.1. Klimaschrank/Peltier-Lüfter-Stack

Um den Wirkungsgrad bzw. den Temperaturbereich des verwendeten Klimaschranks zu verbessern, müsste einerseits der Klimaschrank besser abgedichtet und andererseits die Leistungsfähigkeit des Peltier-Lüfter-Stacks erhöht werden. Eine Leistungssteigerung könnte prinzipiell durch ein stärkeres Peltier-Element erzielt werden. Laut [Plattner, 2009, S.107] wäre ein Tuning des Verstärkers, d. h. eine Vergrößerung des Peltier-Stromes durch eine Verkleinerung des Messwiderstandes (siehe R_{SENSE} bei Abbildung 2.4), durchaus möglich.

Weiters wäre es denkbar, mehrere Peltier-Elemente übereinander in einer Kaskade zu betreiben. Beispielsweise könnte man durch diese Variante die warme Seite des Peltier-Elementes (thermische Gegenmasse) mit einem „parallel“ geschalteten Element kühlen (siehe Abbildung 7.1).

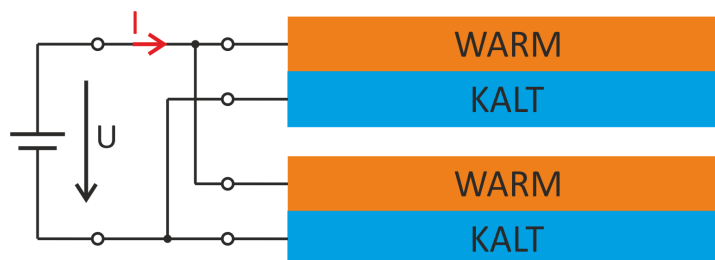


Abbildung 7.1.: Kaskade

7.2. Display

Um die Lebensdauer des Displays bzw. dessen Hintergrundbeleuchtung zu erhöhen, könnte auf der Peripherieplatine ein MOSFET-Schalter implementiert werden, der die Hintergrundbeleuchtung durch einen zusätzlichen Ausgang des Mikrocontrollers ein- bzw. ausschaltet. Die dafür nötigen freien Ports am Mikrocontroller bzw. am Leiterplattensteckverbinder wären vorhanden.

7.3. MOSFET-Schalter

Wie bereits im Abschnitt 3.2.4 erwähnt, können zwei bereits implementierte MOSFET-Schalter der Mikrocontrollerplatine dazu verwendet werden, die Lüfter der thermischen Gegenmasse zu schalten. Diese Möglichkeit könnte beim Vorgang des „*Heizens*“ Sinn ergeben, da in diesem Fall die thermische Gegenmasse vom Prinzip her nicht zusätzlich gekühlt werden müsste.

Anmerkung: Für die praktische Umsetzung müsste zuvor die Mikrocontrollerplatine mit 12V der Versorgungsplatine über das Kommunikationsport (siehe Abschnitt 3.1.5) versorgt werden. Weiters wären dann auch noch die MOSFET-Ausgänge an den Kommunikationsport zu koppeln, um sie über das Verbindungskabel zum Peltier-Lüfter-Stack führen zu können.

7.4. Kaskadenregelung

Durch eine Kaskadenregelung wäre es softwaretechnisch möglich, beide Temperatursensoren in die Regelung miteinzubeziehen. Es erfolgt somit eine Kaskadierung, d. h. eine Hintereinanderschaltung bzw. Verkettung mehrerer Regelkreise mit dem Ziel, die Regelgenauigkeit zu erhöhen (vgl. [Wik10d]).

7.5. Bumpless Transfer

Wie im Abschnitt 5.1 ersichtlich, konnten unsymmetrische Verläufe der Regelgröße bei der Verwendung des Zweipunktreglers für die Ermittlung der kritischen Parameter beobachtet werden. Im Prinzip würden sich, in Abhängigkeit des gewählten Arbeitspunktes, daraus unterschiedliche Verstärkungsfaktoren für „*Kühlen*“ und „*Heizen*“ ergeben, da die Periodendauer und Amplitude der Dauerschwingung unterschiedlich sind. Würde man diese unterschiedlichen Faktoren in der Regelung berücksichtigen wollen, müsste

man eine Umschaltung der Verstärkungsfaktoren während der Regelung implementieren. Das Reduzieren der Unstabilitäten bei der Umschaltung wird in der Literatur mit dem Begriff „*bumpless transfer*“ bezeichnet. Ergänzende Literatur kann unter [Zheng et al., 1996, S. 3ff] und [Hippe, 2006, S. 217] gefunden werden.

7.6. Modellbildung

Um eine Optimierung der Regelvorgänge zu erreichen, müsste ein mathematisches Modell der nichtlinearen Regelstrecke erstellt werden. Je genauer dabei die Realität (Klimaschrank) mit all ihren physikalischen Effekten abgebildet wird, desto besser wären die erzielten Verbesserungen gegenüber der Linearisierung mit den Einstellregeln nach „*Ziegler-Nichols*“ (vgl. Abschnitt 4.3).

Internetquellen

- [Day10] DAYCOUNTER INC. ENGINEERING SERVICES: *MSP430 UART Register Calculator*. <http://www.daycounter.com/Calculators/MSP430-Uart-Calculator.phtml>. Version: 2010. – [Online; Stand 14. Oktober 2010]
- [Dis10] DISPLAYTECH LTD.: *LCD Module Product Specification (64128K COG Series LCD Module)*. http://www.displaytech.com.hk/upload/fast_find/attachment/661-64128k%20cog%20series-v11.pdf. Version: 2010. – [Online; Stand 4. Oktober 2010]
- [Dud10] DUDEN: *Mikroklima*. <http://www.duden.de/definition/mikroklima>. Version: 2010. – [Online; Stand 1. Juli 2010]
- [Eur10] EURO-ASIA ELECTRIC MANUFACTURING (HOLDINGS) LTD.: *EA23-40H-SV*. <http://www.ea-electric.com/en/displayproduct.html?proID=2522795>. Version: 2010. – [Online; Stand 15. Oktober 2010]
- [Fut10] FUTURE TECHNOLOGY DEVICES INTERNATIONAL LTD.: *Virtual COM Port Drivers*. <http://www.ftdichip.com/Drivers/VCP.htm>. Version: 2010. – [Online; Stand 4. Oktober 2010]
- [Ins10] INSTITUT FÜR MIKROELEKTRONIK AN DER TU WIEN: *SEEBECK-Effekt, PELTIER-Effekt*. <http://www.iue.tuwien.ac.at/phd/knaipp/node18.html>. Version: 2010. – [Online; Stand 3. Juni 2010]
- [MEC10] MEC SWITCHES: *Navimec with multimec® 3A anf 3F switches or illumec 4F™ switches*. <http://www.mec.dk/files/webnavimec.pdf>. Version: 2010. – [Online; Stand 4. Oktober 2010]
- [Qui10] QUICK-OHM KÜPPER UND CO. GMBH: *Heat Pipes and Peltier Elements*. <http://www.quick-cool.com/bibliothek.html>. Version: 2010. – [Online; Stand 1. Juli 2010]
- [Rob10] ROBOTERNETZ: *Regelungstechnik*. <http://www.rn-wissen.de/index.php/Regelungstechnik>. Version: 2010. – [Online; Stand 8. Juli 2010]

Internetquellen

- [Row10] ROWLEY ASSOCIATES LTD.: *CrossWorks for MSP430*. <http://www.rowley.co.uk/crossworks/Downloads.htm>. Version: 2010. – [Online; Stand 4. Oktober 2010]
- [Sen10] SENSIRION - THE SENSOR COMPANY: *SHT15 - Digitaler Feuchtesensor (RH&T)*. http://www.sensirion.com/de/01_humidity_sensors/03_humidity_sensor_sht15.htm. Version: 2010. – [Online; Stand 20. Oktober 2010]
- [Sit10] SITRONIX: *65 x 132 Dot Matrix LCD Controller/Driver (ST7565R)*. http://www.displaytech-us.com/pdf/application/Graphic_Module/Sitronix/ST7565R_V17_960601.pdf. Version: 2010. – [Online; Stand 4. Oktober 2010]
- [Tex10] TEXAS INSTRUMENTS: *MSP430F15x, MSP430F16x, MSP430F161x Mixed Signal Microcontroller*. <http://focus.ti.com/lit/ds/symlink/msp430f1612.pdf>. Version: 2010. – [Online; Stand 4. Oktober 2010]
- [uni10a] UNI-PROTOKOLLE.DE: *Peltier-Element*. <http://www.uni-protokolle.de/Lexikon/Peltier-Element.html>. Version: 2010. – [Online; Stand 3. Juni 2010]
- [uni10b] UNI-PROTOKOLLE.DE: *Seebeck-Effekt*. <http://www.uni-protokolle.de/Lexikon/Seebeck-Effekt.html>. Version: 2010. – [Online; Stand 3. Juni 2010]
- [Wik10a] WIKIPEDIA: *EIA-232*. <http://de.wikipedia.org/w/index.php?title=EIA-232&oldid=79693393>. Version: 2010. – [Online; Stand 6. Oktober 2010]
- [Wik10b] WIKIPEDIA: *Faustformelverfahren (Automatisierungstechnik)*. [http://de.wikipedia.org/w/index.php?title=Faustformelverfahren_\(Automatisierungstechnik\)&oldid=79780582](http://de.wikipedia.org/w/index.php?title=Faustformelverfahren_(Automatisierungstechnik)&oldid=79780582). Version: 2010. – [Online; Stand 19. Oktober 2010]
- [Wik10c] WIKIPEDIA: *Joule-Thomson-Effekt*. <http://de.wikipedia.org/w/index.php?title=Joule-Thomson-Effekt&oldid=74478723>. Version: 2010. – [Online; Stand 3. Juni 2010]
- [Wik10d] WIKIPEDIA: *Kaskadenregelung*. <http://de.wikipedia.org/w/index.php?title=Kaskadenregelung&oldid=71867418>. Version: 2010. – [Online; Stand 14. Oktober 2010]
- [Wik10e] WIKIPEDIA: *Kältemaschine*. <http://de.wikipedia.org/w/index.php?title=K%C3%A4ltemaschine&oldid=73847496>. Version: 2010. – [Online; Stand 3. Juni 2010]

Internetquellen

- [Wik10f] WIKIPEDIA: *Linde-Verfahren*. <http://de.wikipedia.org/w/index.php?title=Linde-Verfahren&oldid=73460982>. Version: 2010. – [Online; Stand 3. Juni 2010]
- [Wik10g] WIKIPEDIA: *Magnetische Kühlung*. http://de.wikipedia.org/w/index.php?title=Magnetische_K%C3%BChlung&oldid=65385882. Version: 2010. – [Online; Stand 3. Juni 2010]
- [Wik10h] WIKIPEDIA: *Peltier-Element*. <http://de.wikipedia.org/w/index.php?title=Peltier-Element&oldid=77707941>. Version: 2010. – [Online; Stand 15. Oktober 2010]
- [Wik10i] WIKIPEDIA: *Serial Peripheral Interface*. http://de.wikipedia.org/w/index.php?title=Serial_Peripheral_Interface&oldid=77279100. Version: 2010. – [Online; Stand 1. Oktober 2010]
- [www10] WWW.MIKROCONTROLLER.NET: *Entprellung*. <http://www.mikrocontroller.net/articles/Entprellung>. Version: 2010. – [Online; Stand 4. Oktober 2010]

Literaturverzeichnis

- [Åström and Hägglund, 1995] Åström, K. and Hägglund, T. (1995). *PID Controllers: Theory, Design and Tuning*. International Society for Measurement and Con, 2nd edition.
- [Bernhardt and Hupfer, 1991] Bernhardt, K.-H. and Hupfer, P. (1991). *Das Klimasystem der Erde: Diagnose und Modellierung, Schwankungen und Wirkungen*. Akademie Verlag, Berlin.
- [Frech, 2009] Frech, M. (2009). 1/10°C Thermometer mit 1/100° C Auflösung. Bakkalaureatsarbeit, Institut für Elektronik, Technische Universität Graz.
- [Gausch et al., 1991] Gausch, F., Hofer, A., and Schlacher, K. (1991). Digitale Regelkreise. Technical report. Institut für Regelungstechnik, Technische Universität Graz.
- [Hartl et al., 2008] Hartl, H., Krasser, E., Pribyl, W., Söser, P., and Winkler, G. (2008). *Elektronische Schaltungstechnik. Mit Beispielen in PSpice*. Pearson Studium, München u.a.
- [Hippe, 2006] Hippe, P. (2006). *Windup in Control. Its Effects and Their Prevention*. Springer Verlag, London.
- [Horn and Dourdoumas, 2004] Horn, M. and Dourdoumas, N. (2004). *Regelungstechnik. Rechnerunterstützter Entwurf zeitkontinuierlicher und zeitdiskreter Regelkreise*. Pearson Studium, München.
- [Plattner, 2009] Plattner, R. (2009). Regelelektronik für einen Klimaschrank. Masterarbeit, Institut für Elektronik, Technische Universität Graz.
- [Posch, 2009] Posch, C. (2009). Messgerät zur Messung des Anstellwinkels bei Modellhubschraubern. Masterprojekt, Institut für Elektronik, Technische Universität Graz.
- [Schulz, 2008] Schulz, G. (2008). *Regelungstechnik 2: Mehrgrößenregelung, Digitale Regelungstechnik, Fuzzy-Regelung*. Oldenbourg Verlag, München, 2. auflage edition.

- [v.Ardenne et al., 2005] v.Ardenne, M., Musiol, G., and Klemradt, U. (2005). *Effekte der Physik und ihre Anwendungen*. Wissenschaftlicher Verlag Harri Deutsch, Frankfurt. 3., überarbeitete, neu strukturierte und wesentlich erweiterte Auflage.
- [Yu, 2006] Yu, C.-C. (2006). *Autotuning of PID Controllers. A Relay Feedback Approach*. Springer Verlag, 2nd edition.
- [Zacher, 2000] Zacher, S. (2000). *Automatisierungstechnik kompakt. Theoretische Grundlagen, Entwurfsmethoden, Anwendungen*. Vieweg, Braunschweig u.a.
- [Zheng et al., 1996] Zheng, K., Lee, A.-H., Bentsman, J., and Taft, C.-W. (1996). Steady-state bumpless transfer under controller uncertainty using the state/output feedback topology. *IEEE Control Systems Magazine*.

Abbildungsverzeichnis

1.1. Systemübersicht	1
1.2. Weinkühlschrank EA23-40H ©	2
1.3. Blockdiagramm Elektronik ©	3
2.1. Seebeck-Effekt	8
2.2. Peltier-Effekt	8
2.3. Peltier-Element	9
2.4. Blockschaltbild Peltier-Verstärker	11
2.5. Kühlverfahren (Ansicht von oben)	13
2.6. Funktion einer Heatpipe	14
2.7. Pt1000-Messverfahren (vgl. [Frech, 2009, S. 16])	15
3.1. Mechanische Anbaugruppen	17
3.2. Nordwand Rev. B	18
3.3. Peltier-Lüfter-Stack	19
3.4. Sensorpositionierung	20
3.5. Aufbau Peltier-Verstärker/Versorgungsplatine	20
3.6. Pinbelegung D-Sub-Stecker 15-polig	21
3.7. Bedienteil	22
3.8. Entprellung (hardwaremäßig)	31
4.1. Standardregelkreis (Strukturbild)	33
4.2. Digitaler Regelkreis (Strukturbild)	34
4.3. PID-Regler (Strukturbild)	36
4.4. Gewünschter Verlauf der Sprungantwort des Regelkreises (vgl. [Horn and Dourdoumas, 2004, S. 260])	37
4.5. Open-Loop-Method	38
4.6. Closed-Loop-Method	39
4.7. Regelkreis mit Zweipunktglied (Strukturbild)	40
4.8. Prinzip der adaptiven Regelung (Strukturbild)	41
4.9. Verlauf der adaptiven Reglereinstellung (vgl. [Horn and Dourdoumas, 2004, S.266])	42

Abbildungsverzeichnis

4.10. Stellgrößenbeschränkung (vgl. [Horn and Dourdoumas, 2004, S. 199]) . . .	42
4.11. Regler mit Anti-Windup-Maßnahmen (Strukturbild)	43
4.12. Digitale Regelkreis des Klimaschranks (Strukturbild)	44
4.13. PI-Regler mit Anti-Windup (Strukturbild)	45
5.1. Verlauf der Stellgröße in den Arbeitspunkten des Zweipunktreglers	48
5.2. Verlauf der Regelgröße in den Arbeitspunkten des Zweipunktreglers . . .	49
5.3. Regelverlauf in den Arbeitspunkten (Reglerkoeffizienten bei A_- [1], \bar{A} [2] und A_+ [3])	51
5.4. Regelverlauf bei 25°C (Reglerkoeffizienten bei A_- [1], \bar{A} [2] und A_+ [3]) .	52
6.1. Blockschaltbild der Hardwarekomponenten	53
6.2. State Machine	54
6.3. Zeitstrahl	57
6.4. Timer B Up-Mode	58
7.1. Kaskade	69
A.1. Registerkarten Userprogramm	82
A.2. Schaltplan Versorgungselektronik ©	83
A.3. Schaltplan Regelelektronik ©	84

Tabellenverzeichnis

2.1. Tastverhältnis	12
3.1. Pinbelegung Rückwandsteckers	21
3.2. Spezifikationen Open-Frame-Netzteil	23
3.3. Mikrocontroller Portbelegung	28
3.4. Display Steckerbelegung	30
3.5. Navigationskit Steckerbelegung	31
4.1. Berechnungstabelle Open-Loop-Method	38
4.2. Berechnungstabelle Closed-Loop-Method	40
4.3. Temperaturbereiche	43
5.1. Kritische Parameter der Stell- und Regelgrößen	49
5.2. Reglerkoeffizienten (Amplitude A_-)	50
5.3. Reglerkoeffizienten (Amplitude \bar{A})	50
5.4. Reglerkoeffizienten (Amplitude A_+)	50
5.5. Gewählte Reglerkoeffizienten (Pt1000-Sensor)	52
6.1. Übernommene Bibliotheken	57
6.2. Grenzen der Stellgröße	60
6.3. Reglermodi	61
6.4. Befehle der Fernsteuerung	65
6.5. Aufbau des Informationsspeichers	66
6.6. Speicherzuweisung des Informationsspeichers	67
A.1. Pinbelegung RS232-Buchse	80
A.2. CD-ROM Verzeichnisse	98

A. Anhang

A.1. Entwicklungsumgebung

Als Entwicklungsumgebung wurde Rowley Crossworks (vgl. [Row10]) in der Version 2.0.8 für MSP430-Prozessoren ausgewählt. Für die Kommunikation mit der JTAG-Schnittstelle des Mikrocontrollers wurde ein USB-Debugging-Interface (*MSP-FET430UIF*) verwendet.

A.2. Serielle Datenübertragung

Wie bereits erwähnt, bietet der Bedienteil neben einer klassischen RS232-Schnittstelle noch eine USB-Schnittstelle zur Kommunikation. Durch zwei Jumper am Bedienteil (*TxD_JMP*, *RxD_JMP*) kann entweder eine Kommunikation über die RS232- oder USB-Schnittstelle konfiguriert werden. Standardmäßig ist eine Übertragung über die USB-Schnittstelle konfiguriert.

A.2.1. RS232

Für die Kommunikation des Rechners mit dem Bedienteil wird grundsätzlich ein 1:1 Kabel benötigt, da es sich um eine Rechner (DTE¹) zu Modem (DCE²) Verbindung handelt (Details siehe [Wik10a]). Die relevanten Leitungsbezeichnungen sind in der Tabelle A.1 zusammengefasst. Obwohl die Belegung der Buchsen im Falle des DTE und DCE identisch sind, unterscheiden sie sich in der Übertragungsrichtung.

Pin	Name	Beschreibung	Richtung beim DTE	Richtung beim DCE
2	RxD	Receive Data	In	Out
3	TxD	Transmit Data	Out	In
5	GND	Ground	-	-

Tabelle A.1.: Pinbelegung RS232-Buchse

¹Data Terminal Equipment

²Data Communication Equipment

Für die Datenübertragung über die serielle Schnittstelle wurde eine klassische 8-N-1 Verbindung gewählt. Das Rechnerinterface muss in diesem Fall folgendermaßen konfiguriert sein (vgl. Einstellungen der Firmware im Abschnitt 6.3.4):

- **Bits pro Sekunde:** 9600
- **Datenbits:** 8
- **Parität:** keine
- **Stoppbits:** 1
- **Flusssteuerung:** keine

A.2.2. FT232RL

Die Kommunikation über die USB-Schnittstelle wurde über einen FT232RL-Baustein realisiert, der im Prinzip einen USB/Seriell-Umsetzer darstellt. Am Rechner wird dabei ein virtueller serieller Anschluss installiert, der in weiterer Folge wie ein „normaler“ serieller Anschluss verwendet werden kann.

Grundsätzlich wurde die Installation der Treiber bzw. Kommunikation über die virtuellen seriellen Anschlüsse mit folgenden Betriebssystemen erfolgreich getestet:

- Microsoft Windows XP Professional 32-Bit
- Microsoft Windows 7 Professional 32-Bit
- Microsoft Windows 7 Professional 64-Bit

Bei allen getesteten Betriebssystemen erfolgte eine automatische Treiberinstallation, da der aktuelle Treiber der Version 2.08.02 durch WHQL³ zertifiziert wurde. Durch diese digitale Signatur von Microsoft konnten die aktuellen Treiber über das Microsoft Update bereit gestellt werden. Sollte dennoch eine manuelle Treiberinstallation von Nöten sein, können aktuelle VCP⁴-Treiber für Windows, Linux und Macintosh sowohl für x86- als auch für x64-Systeme von der Herstellerseite (vgl. [Fut10]) geladen werden.

Anmerkung: Bei einer erfolgreichen automatischen oder manuellen Treiberinstallation müssen im Gerätemanager zwei neue Geräte mit den Bezeichnungen „*USB Serial Port (COMx)*“ und „*USB Serial Converter*“ gefunden werden.

³Windows Hardware Quality Labs

⁴Virtual COM Port

A.3. Remote Control

Wie bereits im Abschnitt 6.3.5 erwähnt, wurde, um die Funktionalität der implementierten Befehle des Kommunikationsprotokolls zu verifizieren, ein Userprogramm entworfen. Als Entwicklungsumgebung wurde Microsoft Visual Basic 2010 Express Edition gewählt.

Kurzbeschreibung: Beim Start der Software müssen auf der Registerkarte „*Configuration*“ die Einstellungen für die serielle Datenübertragung getroffen werden. Nach der erfolgreichen Konfiguration und Bestätigung durch „*Save*“ können auf den beiden weiteren Registerkarten aktuelle Temperaturwerte gelesen („*Read Data*“) und neue Solltemperaturen bzw. Modi geschrieben („*Write Data*“) werden. Eine erfolgreiche Übermittlung der Befehle wird durch ein Acknowledge (aktuell gelesene oder geschriebene Temperatur) bestätigt (siehe Abbildungen A.1).

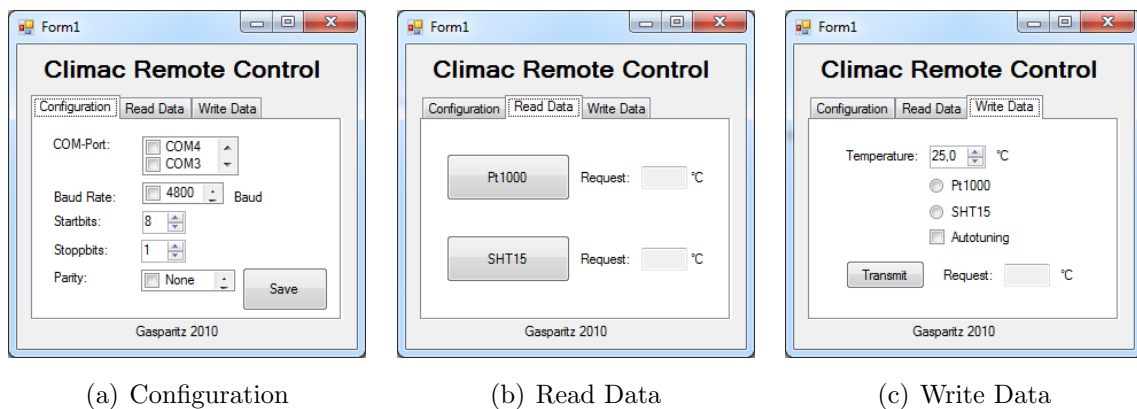


Abbildung A.1.: Registerkarten Userprogramm

Anmerkung: Da die Empfangsroutine der Firmware nur jede Sekunde ausgeführt wird, kann eine ebenso lange Verzögerung zwischen Request und Acknowledge auftreten.

A.4. Alte Regelelektronik

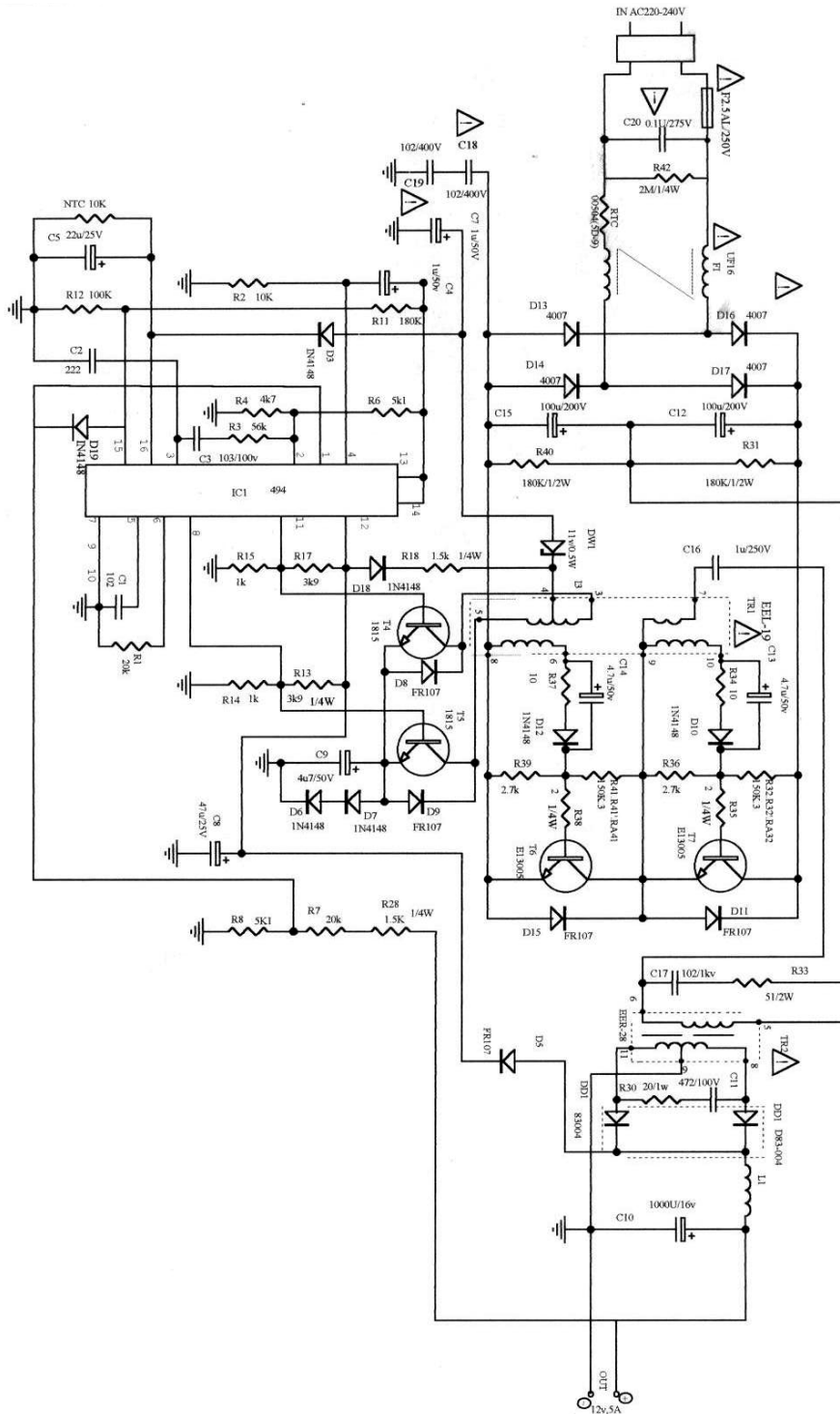


Abbildung A.2.: Schaltplan Versorgungselektronik ©

A. Anhang

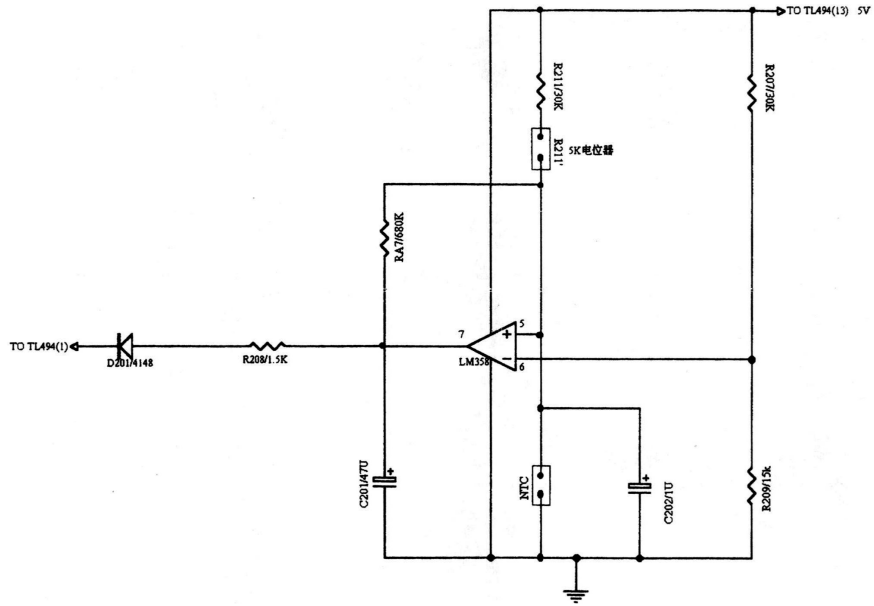
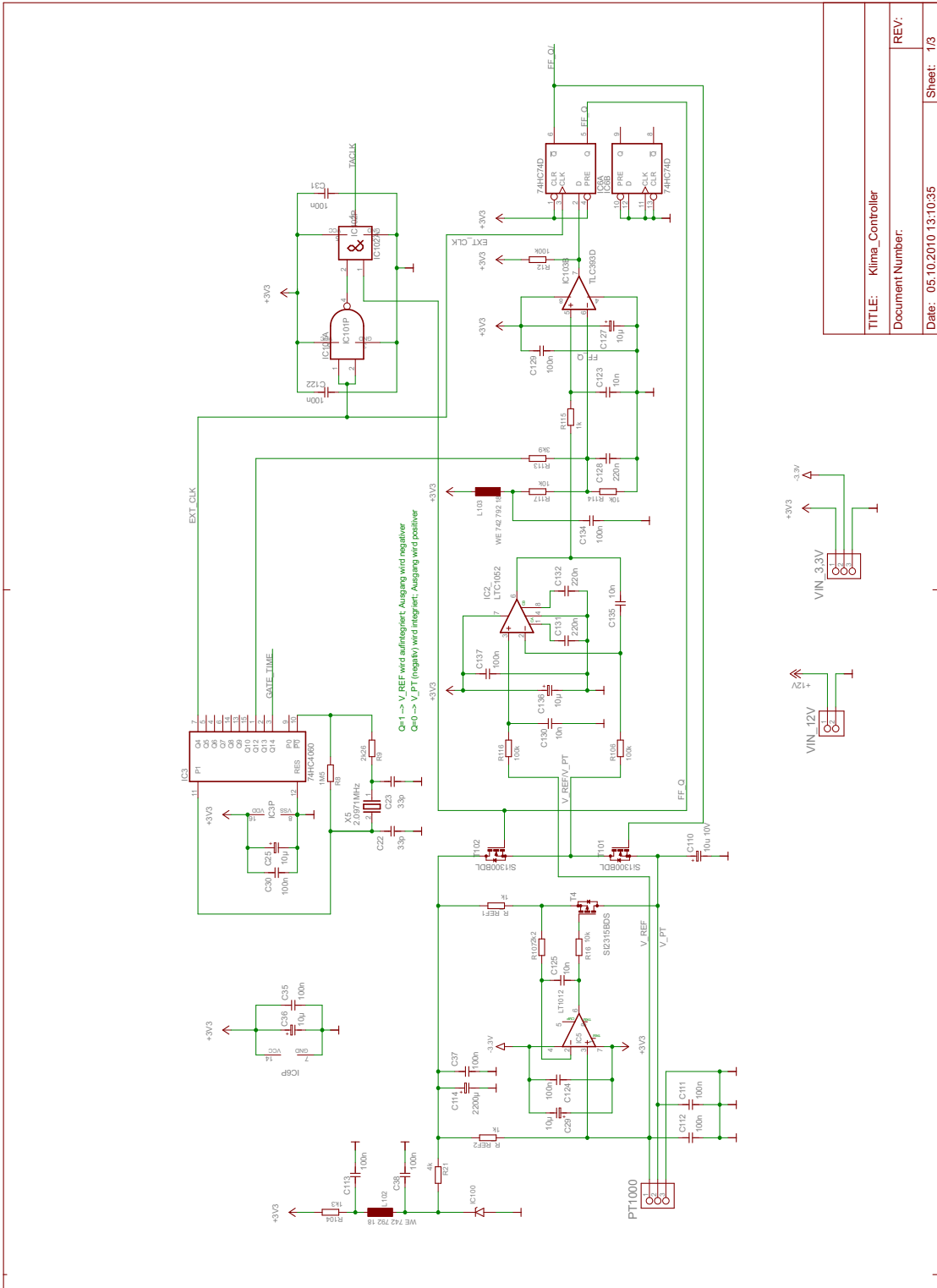


Abbildung A.3.: Schaltplan Regelelektronik ©

A.5. Schaltpläne

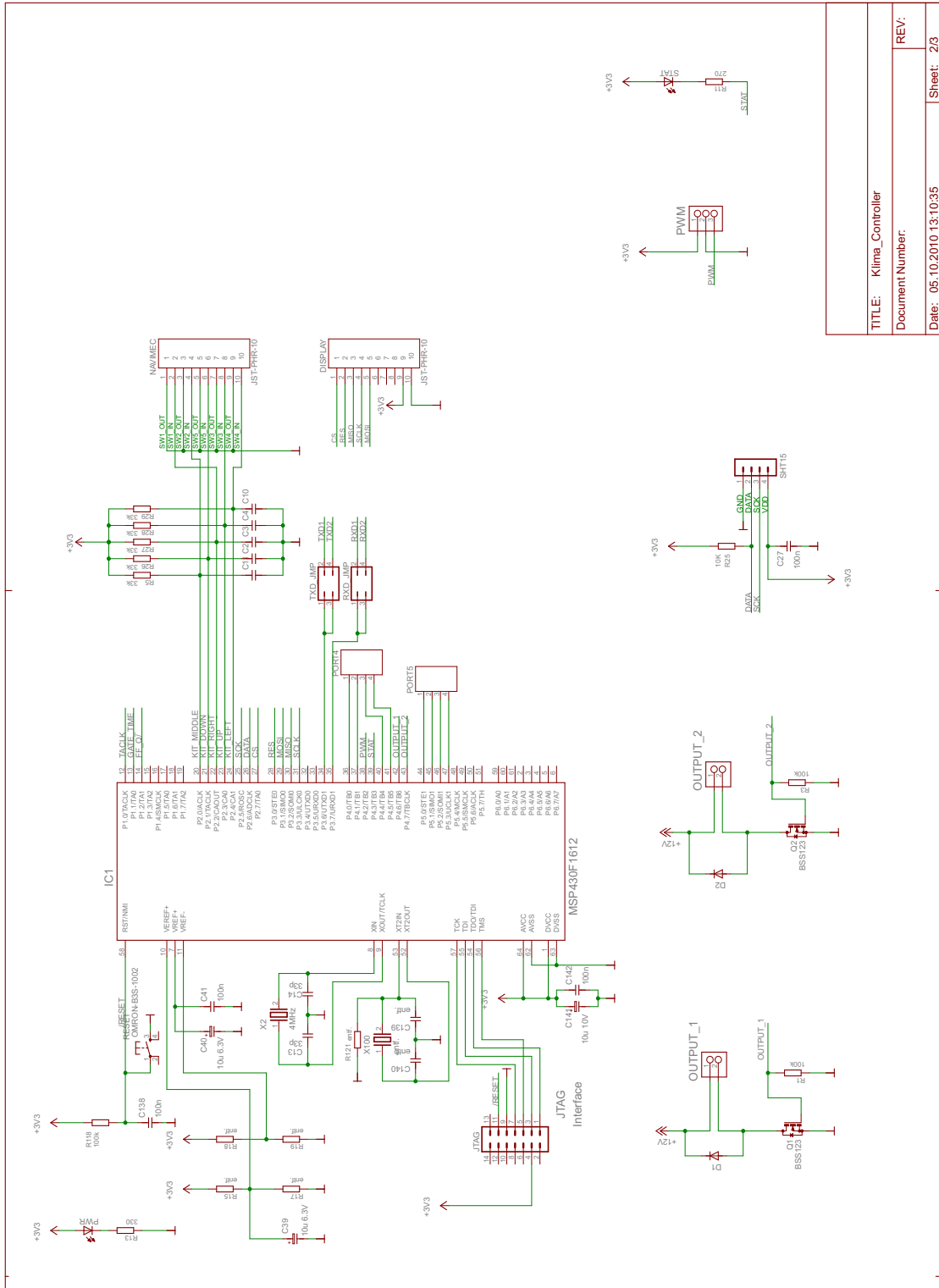
A.5.1. Messschaltung



TITLE: Klima_Controller	
Document Number:	
Date: 05.10.2010 13:10:35	Sheet: 1/3

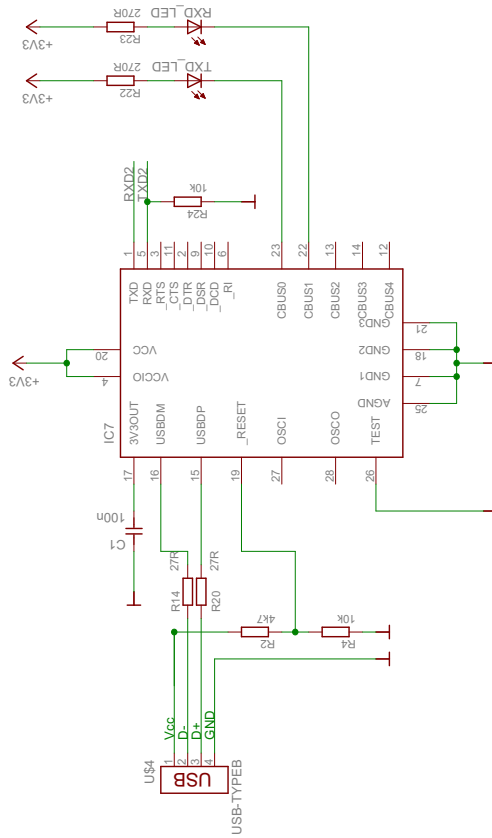
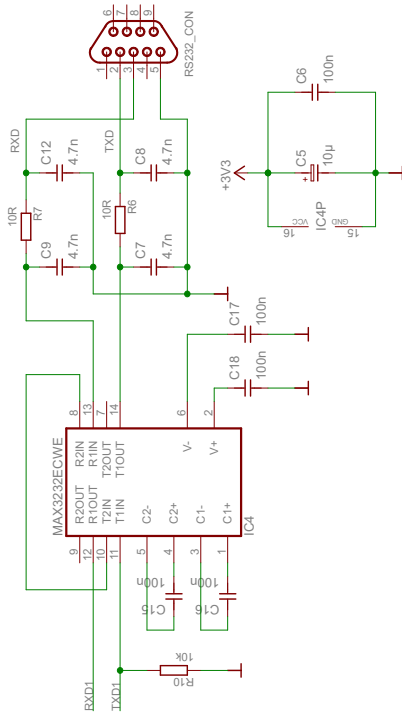


A.5.2. Mikrocontroller und -beschaltung



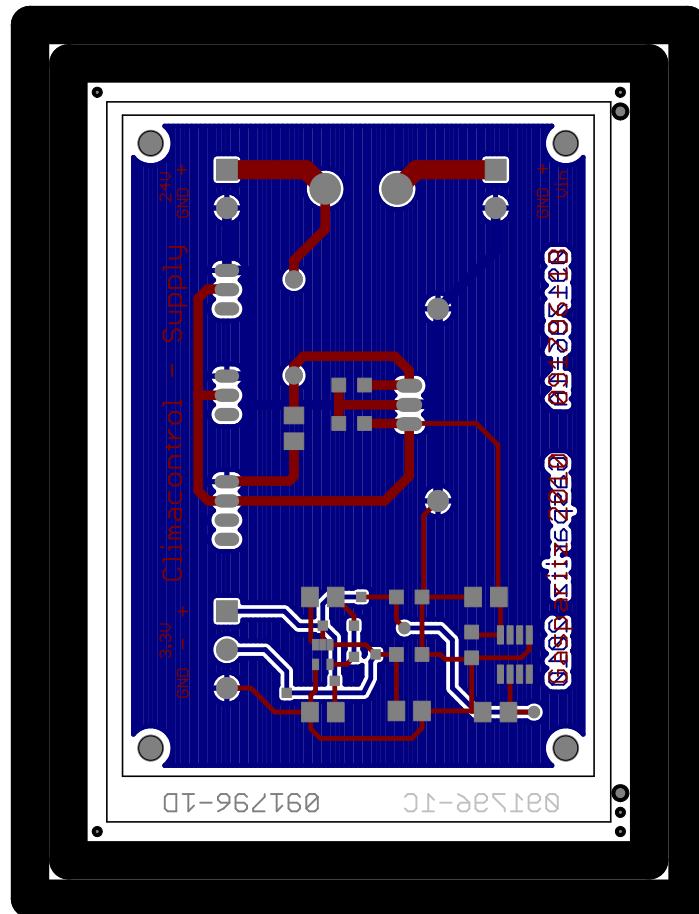
TITLE: Klima_Controller	REV:
Document Number:	
Date: 05.10.2010 13:10:35	Sheet: 2/3

A.5.3. Serielle Datenübertragung

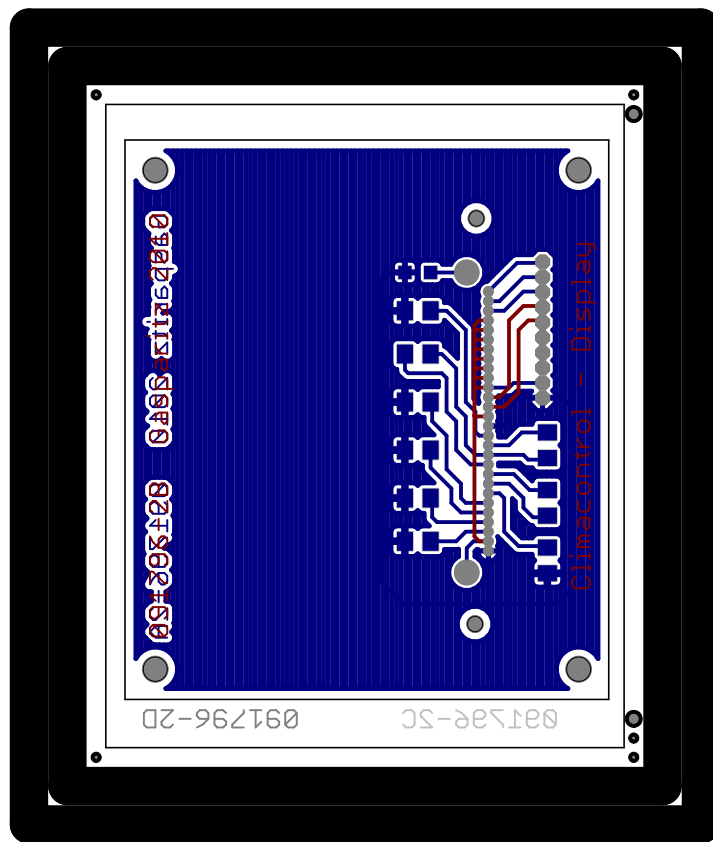


A.6. Layouts

A.6.1. Versorgung

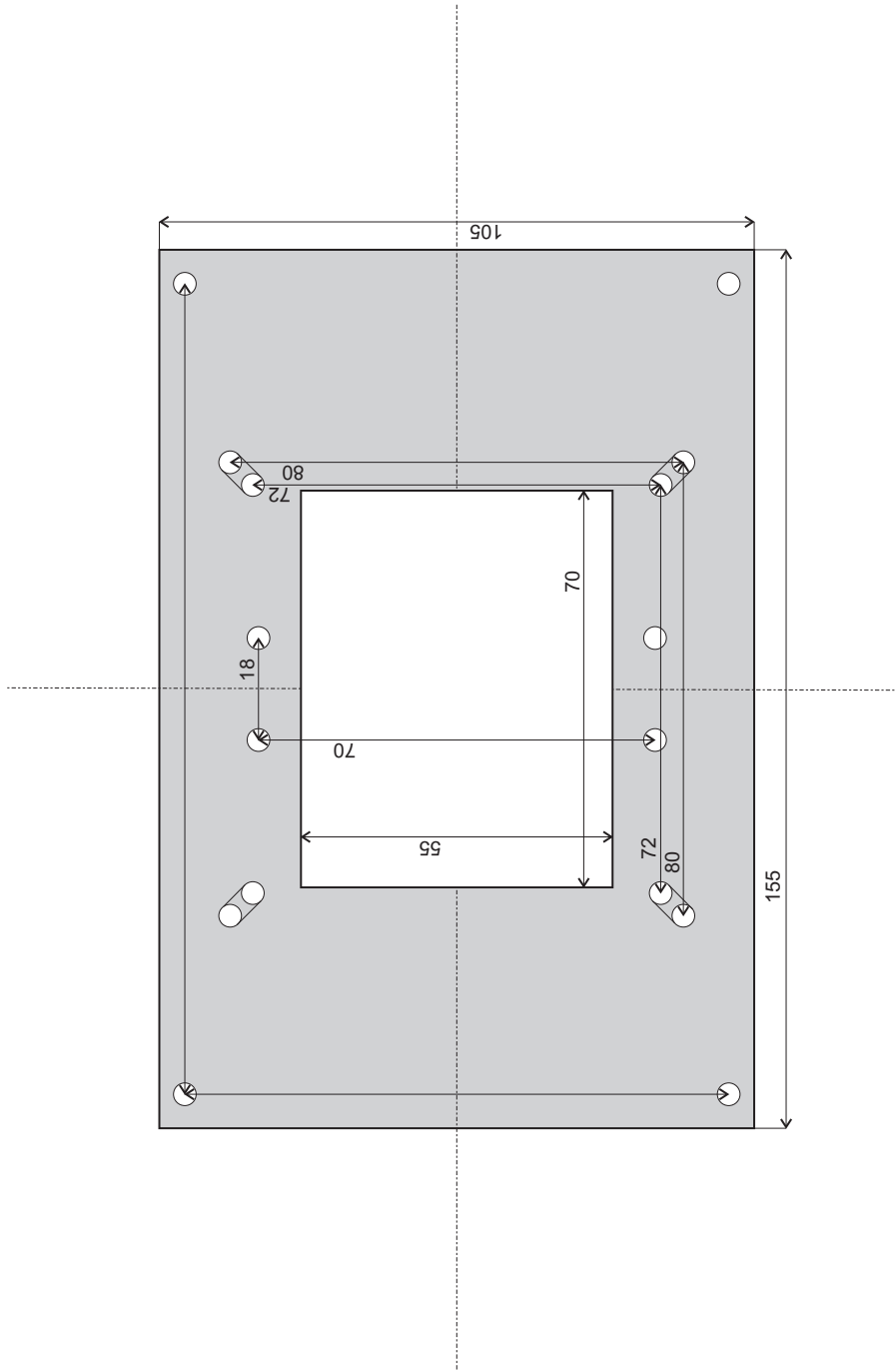


A.6.3. Peripherieplatine

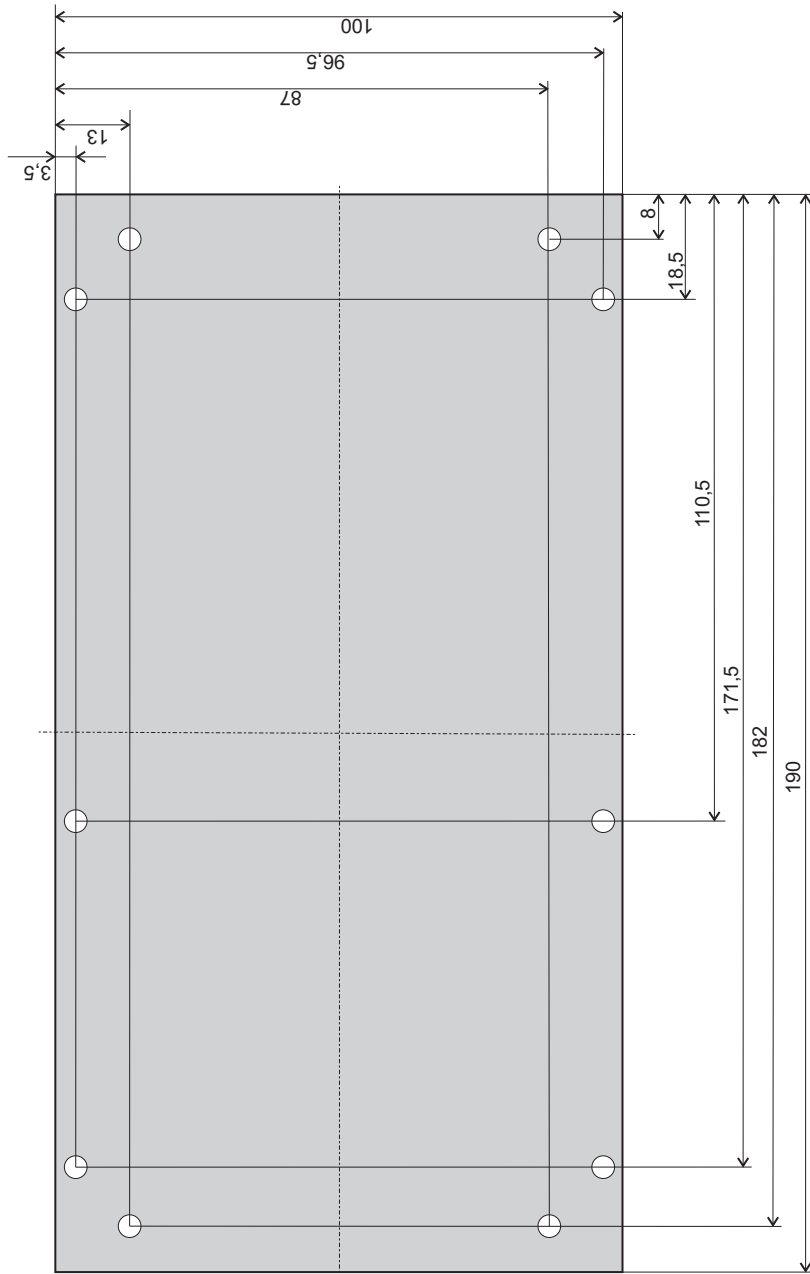


A.7. Anbaupläne

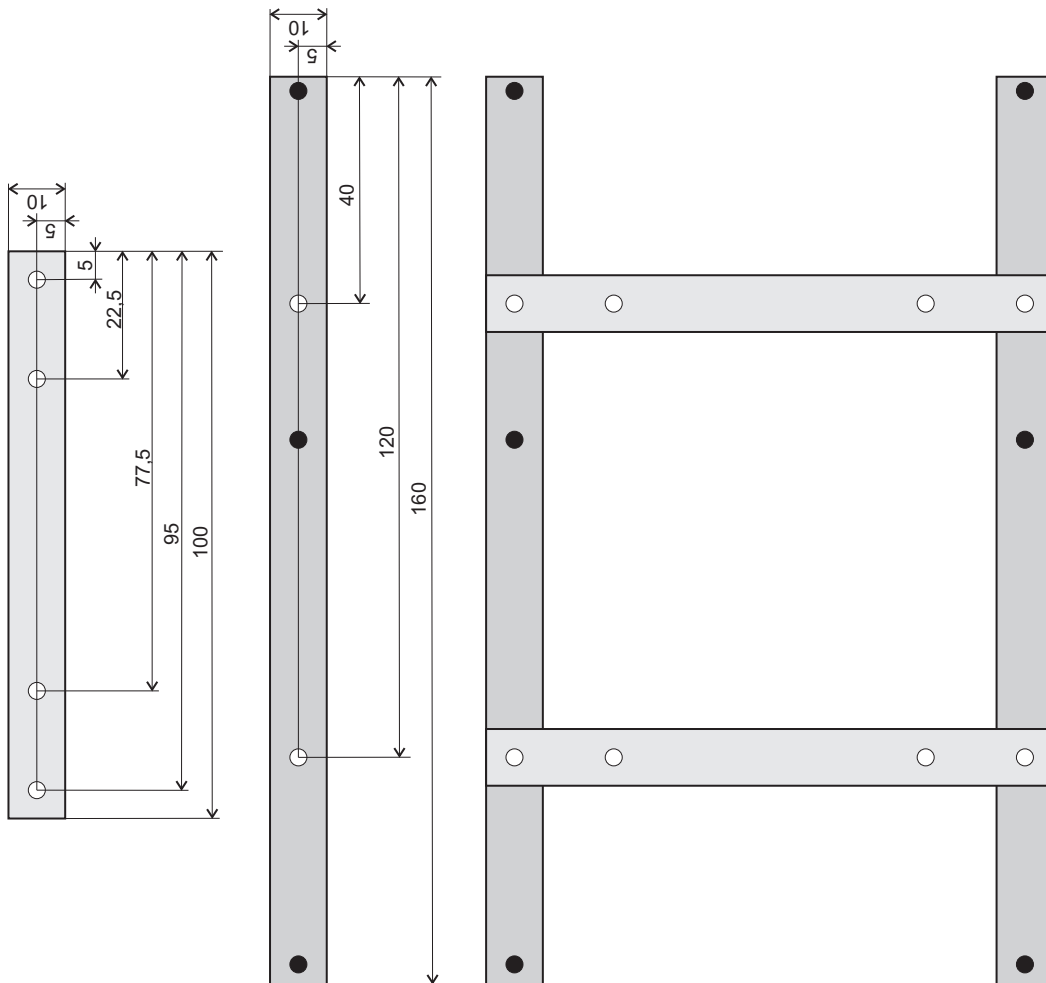
A.7.1. Aufbau Peltier-Lüfter-Stack



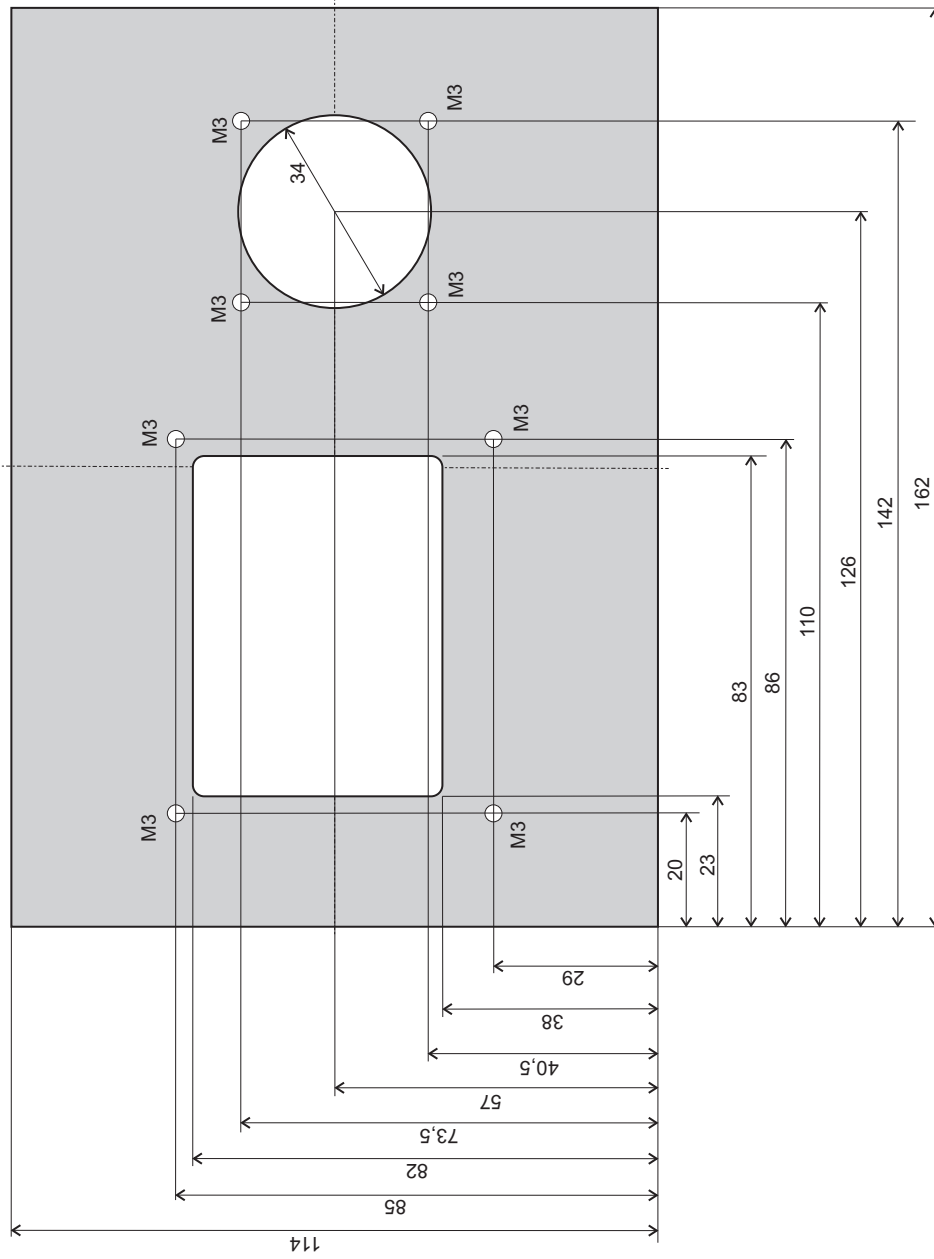
A.7.2. Bodenplatte Peltier-Verstärker



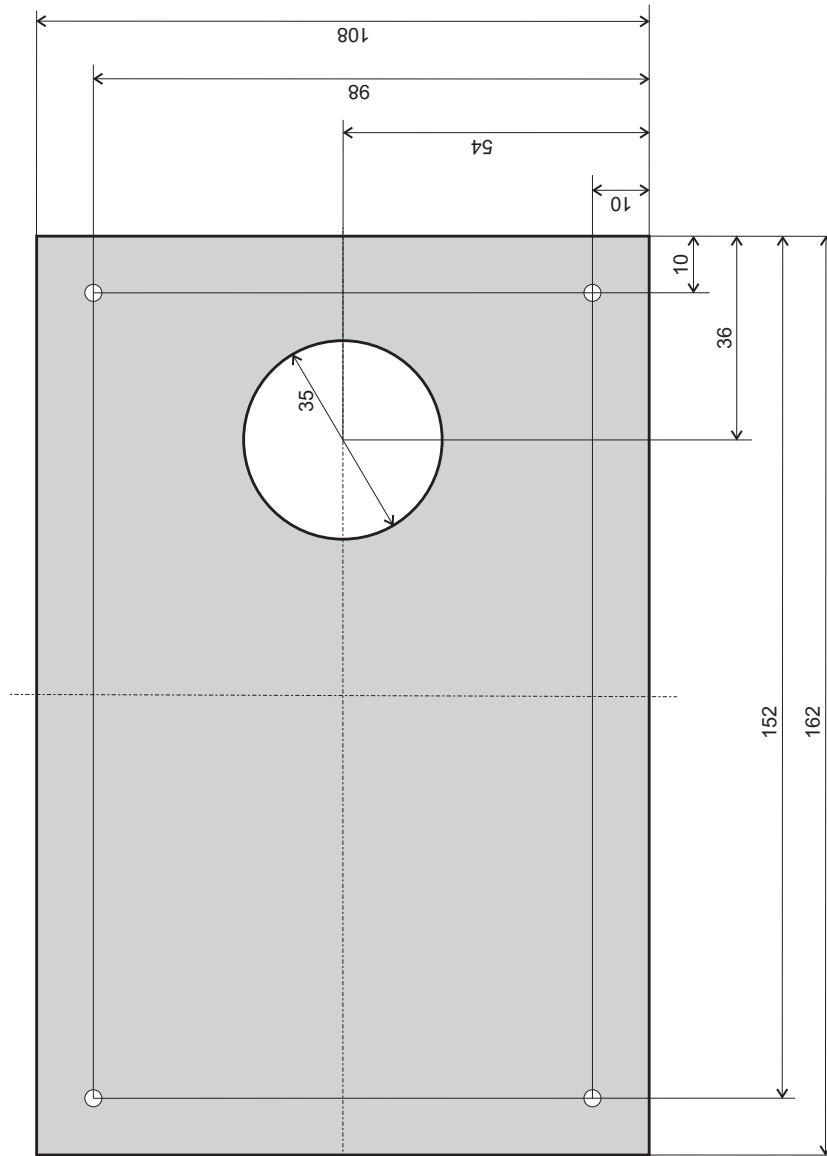
A.7.3. Aufbau Versorgungsplatine



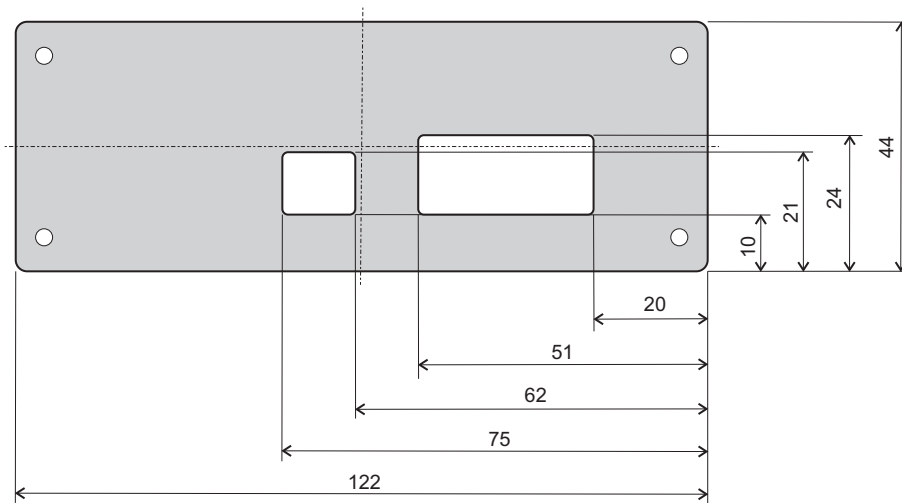
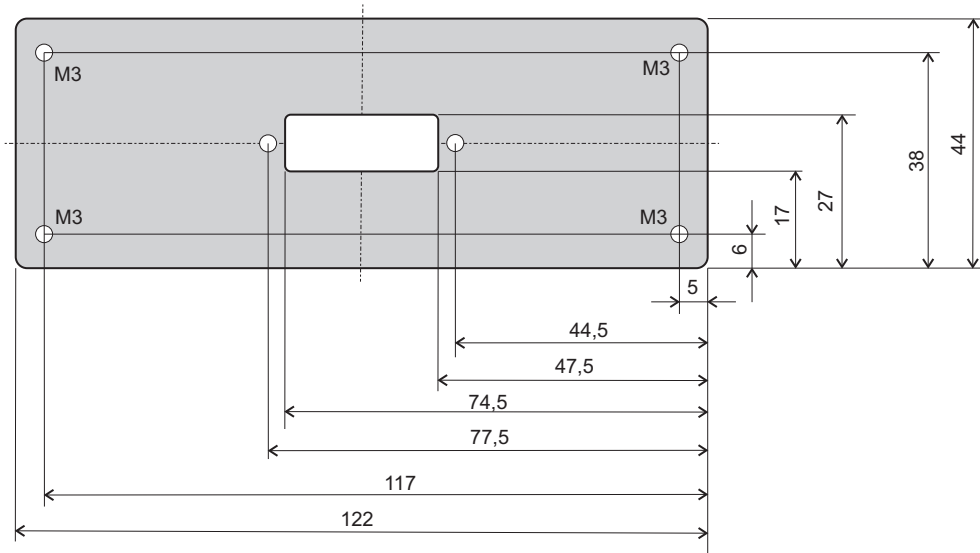
A.7.4. Bedienteildeckel



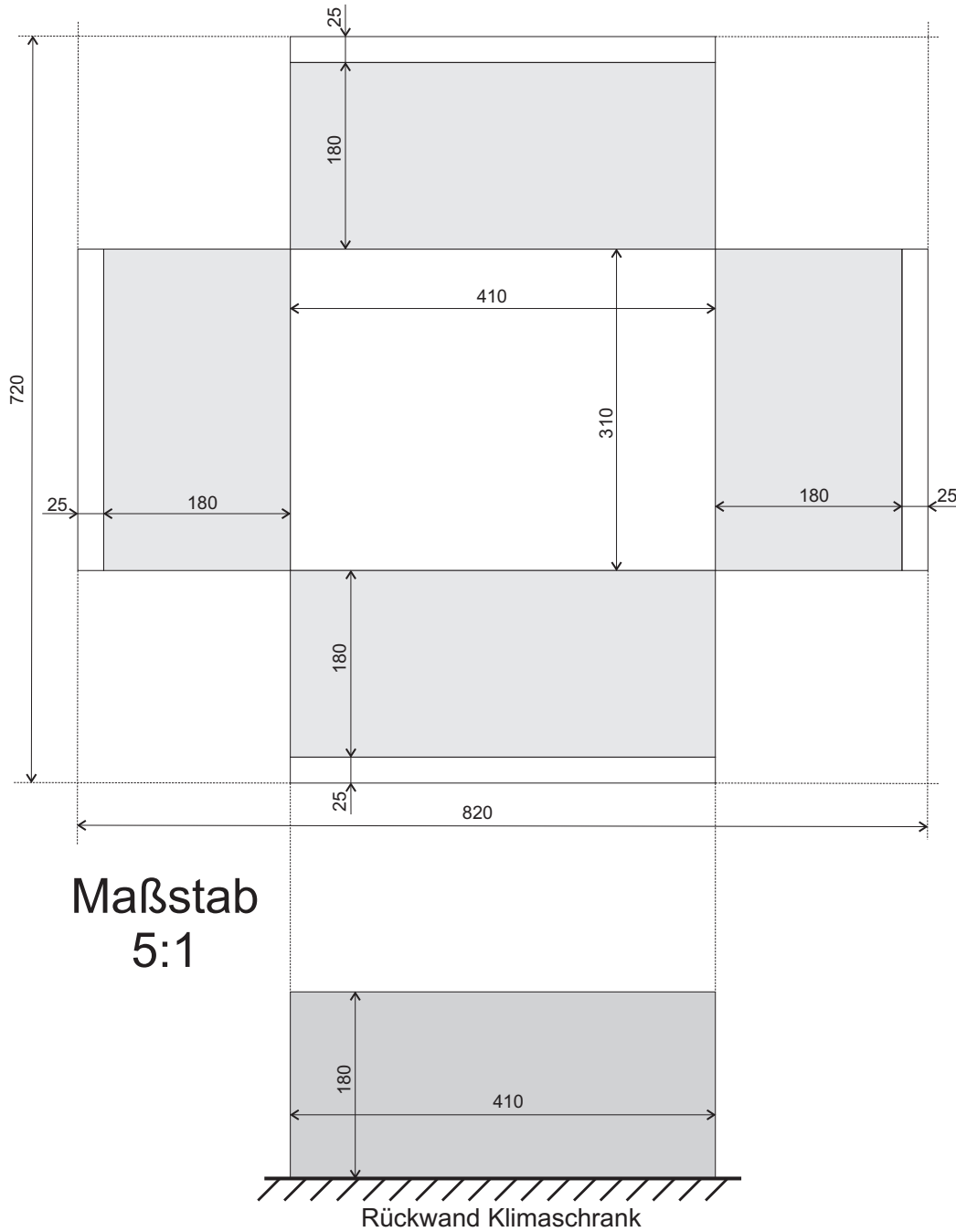
A.7.5. Abdeckung Bedienteildeckel



A.7.6. Bedienteil Seitenteile



A.7.7. Rückwandabdeckung



A.8. Code-Dokumentation

Eine ausführliche Code-Dokumentation (Doxygen) der Firmware ist auf dem Datenträger zu dieser Masterarbeit zu finden.

A.9. Bauteilliste

Eine Liste, der bei der Entwicklung der Platinen verwendeten elektronischen Bauelementen, ist auf dem beigelegten Datenträger gespeichert.

A.10. Datenblätter

Die Datenblätter der verwendeten mechanischen Baugruppen bzw. elektronischen Bauelementen sind auf dem beigelegten Datenträger abgelegt.

A.11. Datenträger

Dieser Masterarbeit ist eine CD-ROM beigelegt, auf der folgende relevante Informationen zur Masterarbeit gespeichert sind (siehe Tabelle A.2):

Verzeichnis	Beschreibung
... \Code-Dokumentation	Doxygen Code-Dokumentation der Firmware
... \Dokumentation	Masterarbeit
... \Firmware	Rowley CrossWorks Firmware-Projekt
... \Literatur	Projektarbeiten, Digitale Regelung, Windup, Auto-tuning, etc.
... \Messungen	Messdaten (Regelverläufe, etc.)
... \Platinenentwurf	Schaltpläne, Layouts, Datenblätter, Bauteilliste
... \Präsentation	Präsentation der Masterarbeit
... \Software & Treiber	Eagle, CrossWorks, TeraTerm, USB-JTAG Interface, FTDI, etc.
... \Userprogramm	Remote Control (Visual Basic 2010 Express Edition Projekt)

Tabelle A.2.: CD-ROM Verzeichnisse