



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Bachelorthesis

Sebastian Heisch

Entwicklung und Implementierung einer
ausfallsicheren
Hochvakuum-Vorpumpensteuerung auf Basis
eines Raspberry Pi

Sebastian Heisch

Entwicklung und Implementierung einer
ausfallsicheren Hochvakuum-Vorpumpensteuerung
auf Basis eines Raspberry Pi

Bachelorthesis eingereicht im Rahmen der Bachelorprüfung
im Studiengang Informations- und Elektrotechnik
am Department Informations- und Elektrotechnik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr. Klaus Jünemann
Zweitgutachter : Prof. Dr.-Ing. Ulfert Meiners

Abgegeben am 29. August 2016

Sebastian Heisch

Thema der Bachelorthesis

Entwicklung und Implementierung einer ausfallsicheren Hochvakuum-Vorpumpensteuerung auf Basis eines Raspberry Pi

Stichworte

Steuerung, Raspberry Pi, Python, Turbopumpe, Scrollpumpe, Hochvakuum, ausfallsicher, UHV

Kurzzusammenfassung

Diese Arbeit behandelt den gesamten Entwicklungsprozess einer zentralen ausfallsicheren Vorpumpensteuerung für ein Hochvakuumssystem. Als Steuerung wird dabei ein Raspberry Pi eingesetzt, welcher in Python programmiert wird. Durch den Einsatz von redundanten Netzteilen, einer Unterbrechungsfreien Stromversorgung und redundanten Transistoren für die Ventile, kann das System im Fehlerfall in einen sicheren Zustand zurückkehren und gilt damit als ausfallsicher. Weiterhin wird ein Atmel AVR verwendet, um die Aktivität des Raspberry Pi und der USV zu überwachen.

Sebastian Heisch

Title of the paper

Development and construction of a Raspberry Pi based failsafe pump control system

Keywords

Control, Pump, Raspberry Pi, Python, Turbopump, High Vacuum, failsafe, UHV

Abstract

This thesis describes the development process of a pump control system for a high vacuum system. The system is built around a Raspberry Pi computer, which is being programmed in Python 2.7. By using redundant power supplies, a battery backup for the Raspberry Pi and redundant transistors for the valves, the system can return to a safe state after the occurrence of an error and is considered to be failsafe. Furthermore a coprocessor based on an Atmel AVR is integrated to monitor the Raspberry Pi and its battery backup.

Inhaltsverzeichnis

Abkürzungsverzeichnis	6
Tabellenverzeichnis	7
Abbildungsverzeichnis	8
1. Einführung	10
1.1. Über den Auftraggeber	10
1.2. Motivation	11
1.3. Aufgabenstellung	12
2. Grundlagen der verwendeten Technik	15
2.1. Raspberry Pi	15
2.1.1. Serial Peripheral Interface	16
2.1.2. Universal Asynchronous Receiver and Transmitter	17
2.1.3. Watchdog Timer	18
2.2. Vakuumsysteme	18
2.2.1. KF Verbindungssystem	18
2.2.2. CF Verbindungssystem	19
2.2.3. Swagelok Verbindungssystem	20
2.2.4. Scrollpumpen	21
2.2.5. Turbomolekularpumpen	22
2.2.6. KF-Eckventile	23
2.2.7. Swagelok Membranventile	24
2.2.8. Dosierventil	26
2.2.9. Thermovac Drucksensoren	26
2.2.10. Ionivac Drucksensoren	27
2.2.11. Ceravac Drucksensoren	28
2.2.12. Festo Drucksensor	29
2.2.13. Wika Drucksensoren	29
3. Entwurf der Pumpensteuerung	30
3.1. Anforderungen an das System	30

3.2. Spezifikation des Systems	31
4. Hardware Design	36
4.1. Digitale Schnittstellen	37
4.2. Analoge Schnittstellen	41
4.3. Ausfallsicherheit	44
4.3.1. Spannungsversorgung	44
4.3.2. Unterbrechungsfreie Stromversorgung	49
4.3.3. Koprozessor	51
5. Software Design	53
5.1. Raspberry Pi - Code	53
5.1.1. Programmablauf Überblick	53
5.1.2. Programmablauf Abzweigsteuerung	55
5.1.3. Programmablauf Pumpensteuerung	56
5.1.4. Programmablauf Koprozessor Interface	57
5.1.5. Programmablauf User Interface	60
5.2. Koprozessor - Code	64
5.2.1. Programmablauf Überblick	64
5.2.2. Programmablauf USV Steuerung	65
5.2.3. Programmablauf Pi Power Control	66
5.2.4. Status-LED	67
5.2.5. Programmablauf Watchdog Timer	68
5.2.6. Raspberry Pi Power State	70
6. Zusammenfassung und Ausblick	71
7. Literatur	72
A. Anhang	73

Abkürzungsverzeichnis

PETRA	Positronen-Elektronen-Tandem-Ring-Anlage
HASYLAB	Hamburger Synchrotron Strahlungslabor
DESY	Deutsches Elektronen-Synchrotron
HERA	Hadronen-Elektronen-Ring-Anlage
GPIO	General Purpose Input Output
IO	Input-Output
IC	Integrated Circuit
SPI	Serial Peripheral Interface
I2C	Inter-Integrated Circuit
ADC	Analog to Digital Converter
MOSI	Master Out Slave In
MISO	Master In Slave Out
KF	Kleinflansch
CF	CONFLAT (Markenname)
HV	Hochvakuum
UHV	Ultrahochvakuum
NO	Normally Open (bei Schaltern)
NC	Not Connected (bei Schaltungen/ICs) oder Normally Closed (bei Schaltern)
UPS	Uninterruptible Power Supply
USV	Unterbrechungsfreie Spannungsversorgung
PSU	Power Supply Unit
LED	Light Emitting Diode
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
PAP	Programmablaufplan
FSM	Finite State Machine
WDT	Watchdog Timer

Tabellenverzeichnis

3.1. Benötigte Ein- und Ausgänge und Stromaufnahme der Bauteile	31
4.1. Stromaufnahme der 3V und 5V Komponenten	48
5.1. Zustände der Raspberry Pi Stromversorgung nach Systemzuständen von 'Pi Power Control' (PPC) und Watchdog Timer	70

Abbildungsverzeichnis

1.1. Aufbau des Vorvakuumsystems für den Einsatz bei PETRAIII	13
2.1. Bild eines Raspberry Pi 2	16
2.2. Schematische Darstellung des SPI Busses	17
2.3. Zusammensetzung einer KF-Verbindung	19
2.4. Aufbau einer CF Verbindung	20
2.5. Teile einer Swagelok Verschraubung	21
2.6. Bild einer Turbopumpe	22
2.7. Schnittzeichnung VAT Eckventil	23
2.8. Bild Softpump Ventil	24
2.9. Aufbau eines Membranventils	25
2.10. Bild eines Thermovac	26
2.11. Schema Bayard-Alpert-Vakuummeter	27
2.12. Bild Bayard-Alpert-Vakuummeter von Ionivac	28
3.1. Blockschaltbild der Platine	34
4.1. 3D-Rendering der Platine	36
4.2. Open Collector Schaltung für Interlock-Signal	37
4.3. Open Collector Schaltung für Steuerventile	38
4.4. Links: Eingangssignal ohne Schutzschaltung (CH1), Versorgungsspannung (CH2); Rechts: verwendete Schaltung	39
4.5. Schutzschaltung basierend auf einer Zener-Diode	40
4.6. Eingangssignal mit Zener-Dioden Schutzschaltung (CH1), Versorgungsspannung (CH2)	40
4.7. Signalaufbereitungsschaltung der analogen 0...10V Eingänge	41
4.8. Signalaufbereitungsschaltung der analogen 4...20mA Eingänge	42
4.9. Signalaufbereitungsschaltung der analogen 0...10V Ausgänge	43
4.10. Automatische Umschaltung der 24V Versorgung	44
4.11. Aufteilung der 24V Versorgung	45
4.12. Automatische Umschaltung der 5V Versorgung	46
4.13. Laderegler der Unterbrechungsfreien Stromversorgung	50
4.14. Aufwärtswandler der Unterbrechungsfreien Spannungsversorgung	51

4.15. Schalter für Raspberry Pi Stromversorgung	52
5.1. Programmablaufplan Raspberry Pi Überblick	54
5.2. Programmablaufplan Raspberry Pi Abzweigsteuerung	55
5.3. Programmablaufplan Raspberry Pi Vorpumpensteuerung	57
5.4. Programmablaufplan Raspberry Pi Empfang serielle Daten	58
5.5. Programmablaufplan Raspberry Pi Timeout serielle Daten	59
5.6. Programmablaufplan Raspberry Pi Watchdog Timer für AVR	60
5.7. Programmablaufplan Raspberry Pi Taster der Nutzerschnittstelle	61
5.8. Programmablaufplan Raspberry Pi LEDs der Nutzerschnittstelle	63
5.9. Programmablaufplan Koprozessor im Überblick	64
5.10. Programmablaufplan Koprozessor Steuerung der unterbrechungsfreien Stromversorgung	66
5.11. Programmablaufplan Koprozessor Steuerung der Stromversorgung des Raspberry Pi	67
5.12. Programmablaufplan Koprozessor Watchdog Timer Überwachung des Raspberry Pi	69

1. Einführung

1.1. Über den Auftraggeber

Im Jahr 1958 wurde das erste Gebäude auf einem ehemaligen Exerzierplatz in Hamburg errichtet. Erst 1959 wurde das 'Deutsche Elektronen-Synchrotron' von Sigfried Balke (ehem. Bundesminister für Atomfragen) und Max Brauer (ehem. Bürgermeister) im Rathaus offiziell gegründet. Seitdem befassen sich dort internationale Teams mit der Erforschung von physikalischen Grundlagen. Der Fokus liegt dabei auf folgenden Gebieten:

1. Forschung mit Synchrotronstrahlung und Freie-Elektronen-Lasern
2. Konstruktion und Betrieb von Teilchenbeschleunigern
3. (Astro-) Teilchenphysik

Dabei war DESY der erste im Forschungszentrum eingesetzte Beschleuniger (erste Experimente im Jahre 1964) und ist der Namensursprung des Forschungszentrums. Der Beschleuniger besitzt einen Umfang von fast 300 m und war damit zum Eröffnungszeitpunkt der größte seiner Art.

Von 1984 bis 1990 wurde die Hadronen-Elektronen-Ring-Anlage gebaut. Dies ist mit 6,3 km der größte Beschleuniger des DESY, mit der Hauptaufgabe, Elektronen und Protonen mit hohen Energien kollidieren zu lassen, um näheres über den Aufbau der Protonen zu erfahren zu können. Das internationale Interesse an einem Beschleuniger wie HERA führte dazu, dass sich auch andere Staaten an den Baukosten beteiligten. Trotz Abschaltung im Jahr 2007, dauert die Auswertung der bei HERA gesammelten Daten bis heute an.

PETRA wurde 1978 in Betrieb genommen und zunächst für die Teilchenphysik genutzt. Im Jahre 1987 wurde der 2,3 km lange PETRA Speicherring in PETRA II umbenannt und umgebaut, um als Vorbeschleuniger für den HERA-Ring zu dienen. Als HERA 2007 abgeschaltet wurde, wurde PETRA II in PETRA III umbenannt und nach einem Umbau durch das Hamburger Synchrotronstrahlungslabor genutzt. PETRA III zählt zu den brilliantesten Röntgenstrahlungsquellen der Welt und verwendet das DESY als Vorbeschleuniger. Derzeit werden die 14 vorhandenen Messplätze um weitere 11 Stück ergänzt, um den Bedürfnissen der vielen Forscher gerecht werden zu können und der Bau von PETRA IV ist bereits in Planung.

Da die Bachelorarbeit in der Abteilung Photon-Science bei PETRA III durchgeführt wurde,

soll nachfolgend kurz auf den Hintergrund von PETRA eingegangen werden:

DESY arbeitet als Vorbeschleuniger und bringt die Elektronen auf annähernd Lichtgeschwindigkeit. Diese Elektronen werden in den Speicherring von PETRA gegeben. Dabei fliegen sie bei jedem Umlauf auch durch die Undulatoren, die in ihrer Öffnung verändert werden können. Der Undulator besteht aus starken Permanentmagneten mit wechselnder Polarität, diese Magnetfelder zwingen die Elektronen zu schnellen Richtungsänderungen. Durch diese Richtungsänderungen verringert sich die kinetische Energie und wird als Synchrotronstrahlung emittiert.

Diese Strahlung besitzt ein schmales Spektrum, welches durch die Wahl der Undulatorparameter auf einen Spektralbereich eingestellt werden kann (bei PETRAIII zwischen UV und harter Röntgenstrahlung). Weiterhin besitzt die Strahlung eine hohe Brillanz, ist sehr intensiv und weist in tangentialer Richtung zur Elektronenbewegung bereits eine hohe Bündelung auf. Außerdem wird sie in sehr kurzen Pulsen (< 1 ns) abgegeben, was eine hohe zeitliche Auflösung ermöglicht. Synchrotronstrahlung kann aufgrund dieser Eigenschaften dazu verwendet werden, um Details auf molekularer Ebene zu untersuchen und sehr schnelle Prozesse zu beobachten.

Durch diese Möglichkeiten gibt es eine Vielzahl von Anwendungen für Einrichtungen wie PETRA, einige dem Autor bekannten Beispiele sind nachfolgend aufgeführt:

- Entwicklung moderner Pharmazeutika durch Beobachtung lebender Viren, Proteine und Krebszellen
- Optimierung von technischen Prozessen, z.B. durch hochgenaue Betrachtung von Werkstoffen unter Belastung oder das Wachstum von Kristallen
- Beobachtung von Umwelteinflüssen durch Chemikalien auf Mikroben
- Erstellung hochauflösender Röntgenbilder von Fundstücken durch Paläontologen
- Analyse von Kulturgütern wie Gemälden oder Steinzeitwerkzeugen

Seit 2009 befindet sich das DESY unter der Leitung von Dr. Helmut Dosch und beschäftigt etwa 2000 Menschen an den Standorten Hamburg und Zeuthen.

1.2. Motivation

Für die Durchführung von Experimenten mit Teilchenbeschleunigern und Synchrotronstrahlung ist ein Vakuum mit einem Druck von weniger als 10^{-10} mBar notwendig, dies wird als Ultrahochvakuum bezeichnet.

Das Ultrahochvakuum wird in mehreren Stufen erzeugt:

1. Scrollpumpe erzeugt ein Vorvakuum mit $< 10^{-1}$ mBar

2. Turbomolekularpumpe erzeugt ein Ultrahochvakuum mit $< 10^{-8}$ mBar
3. Ionengetterpumpe verbessert das Ultrahochvakuum zu $< 10^{-10}$ mBar

Diese Arbeit befasst sich hauptsächlich mit der Erzeugung des Vorvakuums.

Die Experimente werden bei PETRA an sogenannten Beamlines durchgeführt, in denen die Photonen zu den Experimenten geleitet werden. Dabei handelt es sich um tangential vom Speicherring abgehende Stränge, in denen die ausgekoppelten Elektronen zur Emission von Synchrotronstrahlung angeregt werden. Anschließend wird diese Strahlung entlang der Beamline gelenkt und aufbereitet, bis sie schließlich das Messobjekt trifft und Detektoren die Auswirkungen erfassen. Bei der Aufbereitung werden Energie, spektraler Anteil sowie die Polarisation des 'Lichtes' auf die Bedürfnisse des Experiments angepasst.

Der Strahl legt dabei eine Strecke von ca. 80 Metern in Rohren zurück. Damit der Strahl möglichst verlustfrei auf dem Messobjekt auftrifft, ist bei einigen Wellenlängen ein Ultrahochvakuum notwendig. Dieses Ultrahochvakuum wird entlang der Beamline an mehreren Stellen von Ionengetterpumpen erzeugt. Um zu funktionieren benötigen Ionengetterpumpen bereits ein hohes Vakuum, dies wird durch die Turbomolekularpumpen (auch Turbopumpen genannt) sichergestellt.

Für die Verwendung von Turbomolekularpumpen ist ein Vorvakuum nötig, dieses wird bisher von einer dedizierten Scrollpumpe pro Turbopumpe erzeugt. Dieses Vorvakuum soll nun von einer einzelnen (größeren) Scrollpumpe kommen und mit einem Rohrsystem zu den Turbopumpen geleitet werden. Mit diesem Ansatz ist es möglich, die Stromkosten zu senken, die Lärmbelastung des Arbeitsplatzes sinkt und die Wartung wird durch die Zentralisierung vereinfacht. Außerdem wird es, durch die Verwendung eines Computers zur Steuerung des Pumpensystems, möglich über das Netzwerk Kontrolle über das Vorvakuum auszuüben und die Druckgrenzen festzulegen.

Ein weiterer Vorteil dieses Systems ist das Vorhandensein von Abzweigen für Nutzer. Bisher bringt jeder Nutzer eigene Pumpen für die Erzeugung des Vorvakuums mit. Mit dem neuen System wäre es möglich, den Nutzern eine Anschlussmöglichkeit für ein Vorvakuum zu bieten. Damit vereinfacht sich auch der logistische Aufwand und die Zeit zum Aufbau neuer Experimente verkürzt sich.

1.3. Aufgabenstellung

Die nachfolgende Abbildung 1.1 zeigt den schematischen Aufbau des anzusteuernenden Vakuumsystems:

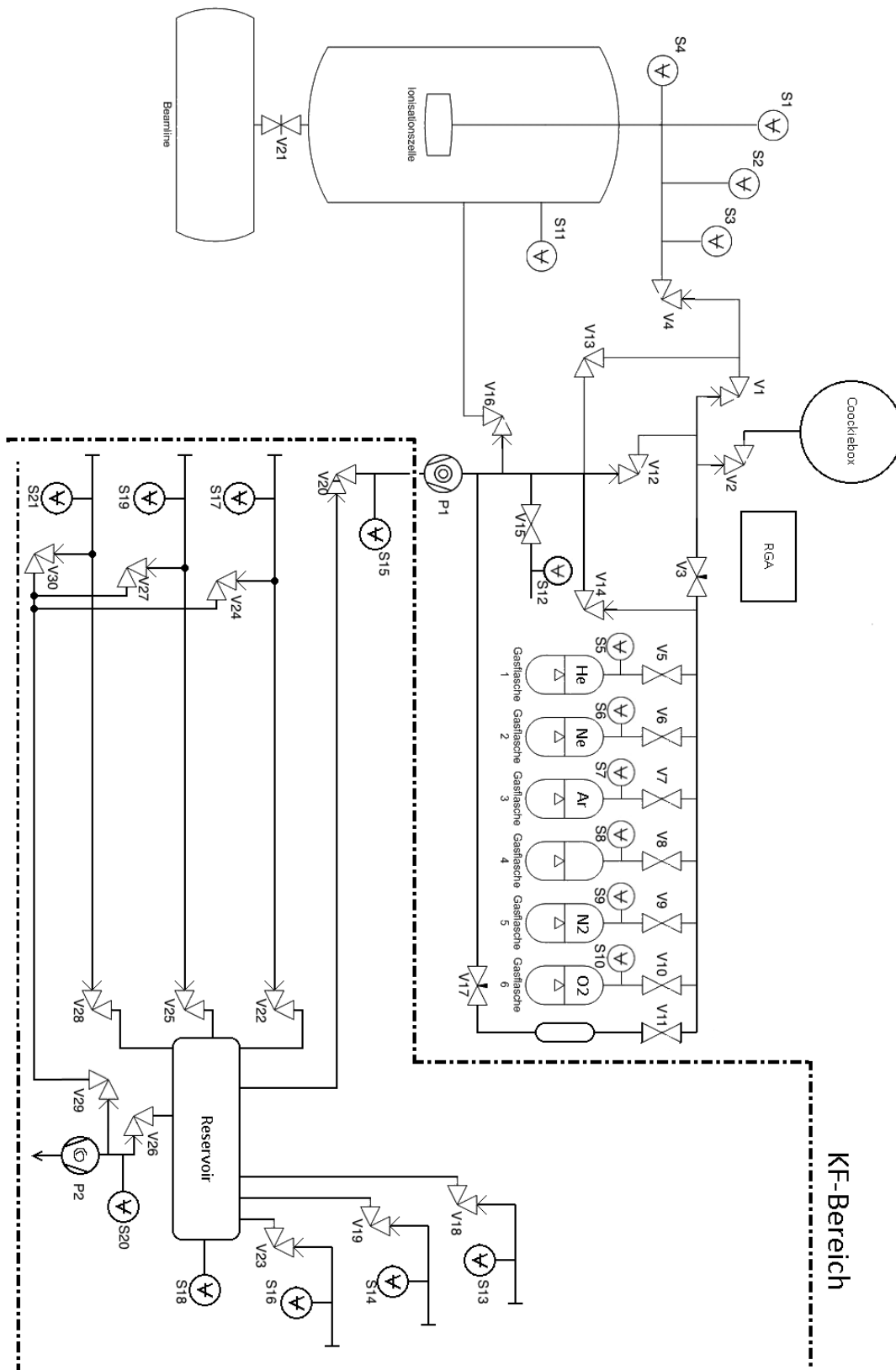


Abbildung 1.1.: Aufbau des Vorvakuumsystems für den Einsatz bei PETRAIII (gezeichnet von Jörn Seltmann [DESY])

Dabei handelt es sich bei Teilen mit S<x> um Drucksensoren, mit P<x> um Pumpen und mit V<x> um Ventile. Insgesamt müssen 34 Ventile, 2 Pumpen sowie 21 Sensoren überwacht und angesteuert werden, dabei sind 4 der Ventile nicht eingezeichnet. Die vier nicht eingezeichneten Ventile arbeiten ähnlich wie die Ventile V24, V27, V29 und V30 in den rechten Abzweigen als möglicher Umgehungspfad für das Reservoir.

Ziel dieser Bachelorarbeit ist es eine Hardware-Plattform zu schaffen, welche die Ansteuerung aller Einzelteile ermöglicht. Weiterhin soll in dieser Arbeit bereits die Software-Steuerung der Komponenten innerhalb der gestrichelten Linie realisiert werden. Die Ansteuerung muss dabei ausfallsicher sein und soll später über das lokale Netzwerk erreichbar sein. Ausfallsicher bedeutet hierbei die Fähigkeit auch bei Absturz des verwendeten Controllers schnell in einen sicheren Zustand überzugehen und nach einem kurzzeitigen Stromausfall schnellstmöglich die Handlungsfähigkeit des Systems wiederherzustellen. Es ist weiterhin wichtig die Ausgänge der Komponenten zu überwachen, um sicherzustellen, dass sie fehlerfrei arbeiten. Damit setzt sich diese Bachelorarbeit aus den folgenden Aufgabenpunkten zusammen:

- Erschaffen einer Hardware-Plattform für die Ansteuerung der Vakuumkomponenten
- Entwicklung eines Software-Interfaces zur Ansteuerung eines Teils der Komponenten
- Entwicklung eines Programmes zur Druckregelung des Vorvakuums

2. Grundlagen der verwendeten Technik

Für ein besseres Verständnis dieser Arbeit soll im folgenden zunächst auf die Grundlagen der Systeme eingegangen werden.

2.1. Raspberry Pi

Der erste Raspberry Pi wurde 2012 von der Raspberry Pi Foundation vermarktet. Dabei handelt es sich um eine Platine, auf der die Kernkomponenten eines Computers vereint sind. Diese Kernkomponenten enthalten einen Hauptprozessor, einen Grafikprozessor, Hardware um Audiosignale zu erzeugen, Arbeitsspeicher sowie eine SD-Karte als Festspeicher.

Die Grundidee des Raspberry Pi ist es, eine günstige Lernplattform für Schüler zu erschaffen, da große Computer teurer werden und daher weniger mit deren Hardware experimentiert wird. Da beim Raspberry Pi, statt der x86 oder x64-Architektur eines großen Computers, ein ARM basierter Prozessor eingesetzt wird, ist die Rechenleistung erheblich geringer, jedoch sinkt damit auch die Verlustleistung des Chips wesentlich und der Preis ist geringer.

Zusätzlich zu den Kernkomponenten eines Computers bietet der Raspberry Pi eine Steckerleiste mit bis zu 48 digitalen Ein- und Ausgängen (auch als GPIO-Pins bezeichnet) und eine Ethernet Schnittstelle für eine LAN-Verbindung. Einige der GPIO-Pins bieten Sonderfunktionen, beispielsweise serielle Schnittstellen. Diese Verbindungsmöglichkeiten machen den Raspberry Pi zu einer idealen Lösung für Projekte, die gleichzeitig mit primitiverer Elektronik wie Transistoren oder ICs und Computern oder Servern im Netzwerk verbunden sein sollen. Als Rechner für diese Arbeit wurde der Raspberry Pi 2 ausgewählt, da dieser zu den aktuellsten zählt und zugleich auf einer bewährten 32-Bit Architektur basiert. Das Compute Module und der Raspberry Pi Zero weisen zwar einen praktischen Formfaktor auf, bieten jedoch keine nativen Netzwerkanbindungen und sind mitunter problematisch zu beschaffen. Der Raspberry Pi 2 bietet einen 4 Kern Prozessor mit 900 MHz Taktfrequenz, integriertes Ethernet und 26 GPIO-Pins. Damit bietet der Raspberry Pi 2 alle benötigten Funktionen.

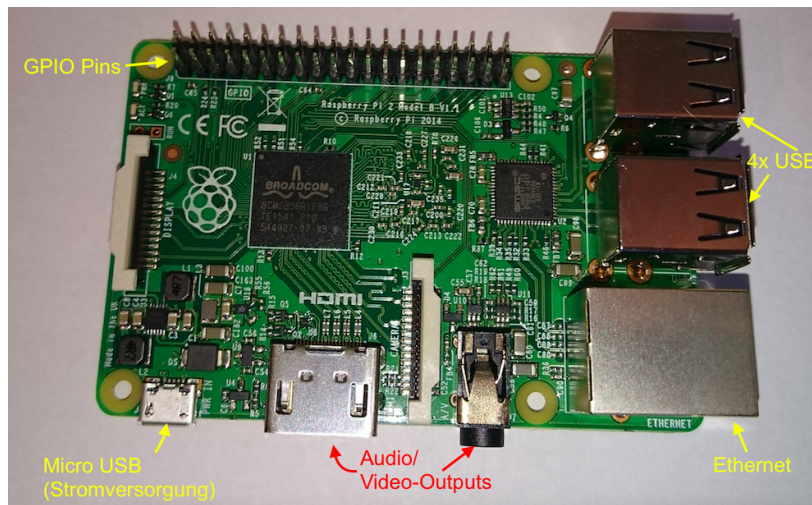


Abbildung 2.1.: Bild eines Raspberry Pi 2

Auf Abbildung 2.1 ist ein Raspberry Pi 2 gezeigt. Oben links befindet sich die 40-polige GPIO Steckerleiste, davon können 26 Pins als GPIO-Pins genutzt werden, übrige Pins dienen entweder der Stromversorgung oder sind Produkten der Raspberry Pi Foundation vorbehalten. Unten links ist der Micro-USB Anschluss gezeigt, an diese wird das Netzteil für den Raspberry Pi angeschlossen. Rechts davon befinden sich die Ausgänge für Audio- und Video-Signale. An der rechten Außenkante sind die Ethernet-Schnittstelle sowie die USB-Anschlüsse des Computers zu sehen.

2.1.1. Serial Peripheral Interface

Wie bereits im vorhergehenden Kapitel beschrieben, bietet der Raspberry Pi auf seinen GPIO-Pins serielle Schnittstellen. Eine dieser Schnittstellen ist das Serial Peripheral Interface. SPI wird in dieser Arbeit dazu verwendet, um Daten aus dem Analog-Digital-Wandler auszulesen und eine Verbindung zu den verwendeten Port-Expandern herzustellen. Ein ADC wird dazu verwendet, die als analoge Spannung ausgegebenen Sensormesswerte in ein vom Computer verarbeitbares digitales Format umzusetzen. Mit einem Port-Expander lassen sich weitere einzeln adressierbare Ein- und Ausgänge zu einem Mikrocontroller hinzufügen. Das Serial Peripheral Interface ist ein von Motorola entwickeltes Bussystem für die Kommunikation zwischen einem Master und einem oder mehreren Slave-Geräten. Dafür werden 3 Leitungen verwendet: eine für die Datenübertragung vom Master zum Slave (MOSI), eine für die Übertragung vom Slave zum Master (MISO) und eine Taktleitung für die Synchronisation der Übertragung. Diese drei Leitungen werden von allen Teilnehmern geteilt, zusätzlich

besitzt jeder Slave eine Adressleitung. Diese Adressleitung ist notwendig, um einzelne Teilnehmer für die Kommunikation auszuwählen und wird als Chip Select (CS) bezeichnet. Der Aufbau des Interface ist im nachfolgenden Bild dargestellt:

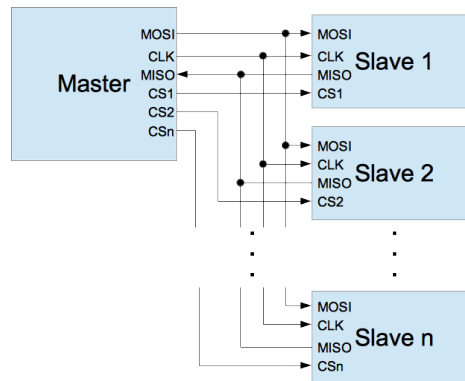


Abbildung 2.2.: Schematische Darstellung des SPI Busses

Die Taktleitung CLK wird vom Master getaktet. Dies veranlasst den ausgewählten Slave die an MOSI anliegenden Daten einzulesen, beziehungsweise seine eigenen Daten auf die MISO-Leitung zu geben. Es gibt 4 Modi nach denen das SPI arbeiten kann, diese legen den Zustand der Taktleitung bei Inaktivität und die für die Synchronisation verwendete Taktflankenrichtung fest.

2.1.2. Universal Asynchronous Receiver and Transmitter

Neben dem SPI bietet der Raspberry Pi auch Anschlüsse für einen Universal Asynchronous Receiver and Transmitter (UART). Der UART verwendet zwei Datenleitungen und verzichtet im Gegensatz zum SPI auf eine Taktleitung. Die Synchronisation der Datenübertragung erfolgt, indem zuerst ein Startbit übertragen wird. Die Übertragungsgeschwindigkeit (auch als Baudrate bezeichnet) muss dabei auf beiden Geräten übereinstimmen. Die Kommunikation mittels UART ist in den meisten Fällen auf zwei gleichberechtigte Teilnehmer begrenzt und kommt daher ohne Chip Select Leitung aus. Beim UART werden die Kommunikationsleitungen als Tx und Rx bezeichnet, Tx ist dabei die Sendeleitung des Gerätes und Rx bildet die Empfangsleitung. Zwischen den Teilnehmern werden die Leitungen überkreuzt und Tx mit Rx verbunden. Neben der Baudrate ist die Anzahl der zu übertragenden Bits, ob auf Parität geprüft werden soll sowie die Anzahl der Stoppbits festzulegen, damit die Endgeräte miteinander kommunizieren können. Eine häufig verwendete Kombination ist 8N1, d.h. 8 Datenbits, keine Paritätsprüfung und 1 Stoppbit. Für die Übertragung sendet das Gerät zuerst

ein Startbit zur Synchronisation, darauf folgen die Datenbits, ein Paritätsbit und schliesslich das Stoppbit. In dem für die Bachelorarbeit konstruierten System wird der UART verwendet, um die Kommunikation zwischen dem Raspberry Pi und dem Koprozessor zu realisieren. Dieser Koprozessor ist für die Überwachung des Raspberry Pi und der Stromversorgung zuständig.

2.1.3. Watchdog Timer

Da der Raspberry Pi aufgrund von Fehlern bei der Programmierung oder externen Einflüssen möglicherweise seine Aufgaben nicht korrekt erfüllen kann, ist es notwendig, dessen Aktivität zu überwachen. Diese Überwachung geschieht mittels eines Watchdog Timers. Wird der Watchdog Timer gestartet, muss der überwachte Prozessor innerhalb von festgelegten Zeitabständen ein Signal senden, um seine Funktionalität zu signalisieren und den Timer zurückzusetzen. Bleibt dieses Signal aus, so erzwingt der Watchdog Timer einen Reset des Prozessors und ermöglicht diesem durch einen Neustart aus dem fehlerhaften Zustand zurückzukehren.

2.2. Vakuumsysteme

Da Vakuumsysteme am DESY eine zentrale Rolle einnehmen, soll im Folgenden auf die Funktionsweise einiger Teile eingegangen werden.

2.2.1. KF Verbindungssystem

Das KF-Verbindungssystem ist weit verbreitet und einfach zu verwenden. Für die Verbindung von zwei Komponenten, beispielsweise zweier Rohrleitungen, werden ein Dichtungsring mit Zentrierring sowie ein Spannring benötigt, um die Flansche zu verbinden.

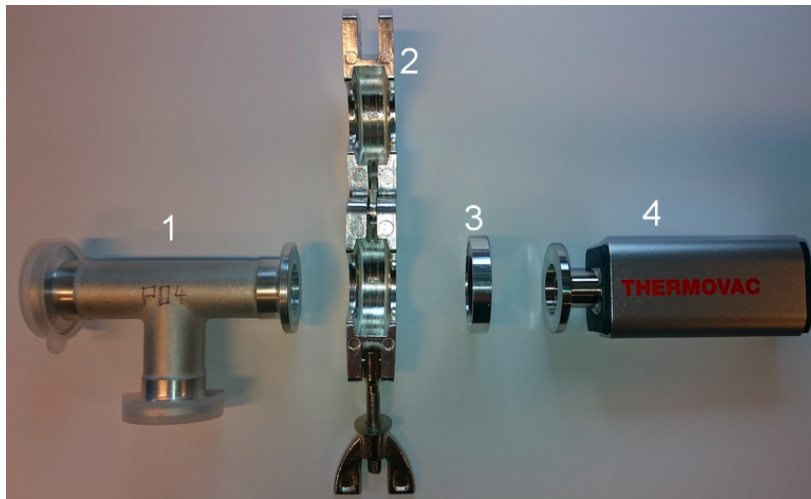


Abbildung 2.3.: Beispielaufbau einer KF Verbindung zwischen einem T-Stück und einem Thermovac-Sensor

Abbildung 2.3 zeigt beispielhaft die Zusammensetzung einer KF-Verbindung. Die zu verbindenden Teile (hier T-Stück (1) und Thermovac (4)) liegen Außen, dazwischen wird eine Dichtung (3) gesetzt. Abschließend wird der Spannring (2) um die Flansche gelegt und mit der Flügelmutter festgezogen. Durch den Zentrierring der Dichtung wird sichergestellt, dass die Flansche mittig aufeinandertreffen. Aufgrund der angeschrägten Oberflächen des Spannrings und der Flansche, drückt der Spannring diese auf den Dichtungsring. Die so entstehende Verbindung weist eine ausreichende Dichtigkeit für Grob- beziehungsweise Vorvakuumssysteme auf.

2.2.2. CF Verbindungssystem

Für Anwendungen im UHV-Bereich sind KF-Verbindungen nicht dicht genug, da das verwendete Dichtungsmaterial nicht ausreichend gegen Diffusion schützt. Stattdessen werden CF-Verbindungen genutzt. Bei CF-Verbindungen wird auf einen Spannring verzichtet und die Flansche werden mithilfe von mehreren Schrauben zusammengehalten. Die bei KF verwendeten Dichtungen werden bei CF durch Kupferringe ersetzt. An den Flanschen befindet sich eine Schneidkante, welche sich beim Zusammenfügen in das Kupfer eindrückt und die Verbindung damit abdichtet. Abbildung 2.4 zeigt beispielhaft eine CF-Verbindung zwischen einem Blindflansch und einem drehbaren Flansch. Nach Entnehmen der Dichtung aus der Schutzverpackung wird diese zwischen die Schneidkanten der Flansche gelegt. Die Flansche werden dann mit Schrauben und Muttern versehen, welche kreuzweise angezogen werden.



Abbildung 2.4.: Beispielaufbau einer CF Verbindung (Blindflansch mit drehbarem Flansch an Reservoir)

Neben der im Vergleich zu KF höheren Druckfestigkeit bietet CF auch andere Vorteile: so kann die Kupferdichtung keine Gase freisetzen und das Vakuumsystem kann mit höheren Temperaturen ausgeheizt werden. Die bei CF-Flanschen von der Dichtung radial abgehenden Nuten können dafür verwendet werden, Flüssigkeiten mit niedrigem Siedepunkt (beispielsweise Alkohole) oder Gase wie Helium nahe an die Dichtung heranzusprühen, um Undichtigkeiten ermitteln zu können.

2.2.3. Swagelok Verbindungssystem

Das Gaseinlasssystem (Abbildung 1.1, Mitte oben) verwendet Swagelok-Verbindungen zwischen den Gasflaschen, den Ventilen und dem restlichen System. KF- und CF-Verbindungen werden hier nicht eingesetzt, da sie undicht (KF) beziehungsweise zu teuer (CF) sind.



Abbildung 2.5.: Teile einer Swagelok Verschraubung

Abbildung 2.5 zeigt die Einzelteile einer Swagelok-Verschraubung. Diese bestehen aus dem anzuschließenden Rohr (nicht gezeigt), einer Überwurfmutter (1), einem hinteren Metallring (2), einem vorderen Metallring (3) und einem Verschraubungskörper (4). Die Metallringe sind keilförmig ausgeführt, werden ineinander gelegt und dann in die Überwurfmutter eingesetzt. Die Überwurfmutter wird mit den Metallringen auf den Verschraubungskörper aufgesetzt und das anzuschließende Rohr eingeschoben, danach wird die Mutter festgezogen. Durch die konische Form wird der vordere Ring beim Festziehen in das Rohr eingedrückt und dichtet die Verbindung ab. Der hintere Metallring sitzt am Kopfende der Mutter und klemmt das Rohr ein, dadurch entsteht eine feste mechanische Verbindung der Rohres mit der Verschraubung.

2.2.4. Scrollpumpen

Die Erzeugung der Grob- bzw. Vorvakua erfolgt mithilfe von Scrollpumpen. In diesem System wird eine 'xDS35i' von Edwards über eine KF-Verbindung an das (Vor-) Vakuumsystem angeschlossen. Die Pumpe kann über einen Steckverbinder gesteuert und der aktuelle Fehlerzustand ausgelesen werden. Eine Scrollpumpe besitzt einen spiralförmigen Rotor und Stator, welche ineinander greifen. Durch die Verbindung zum Vakuumsystem wird das Prozessgas (hier Luft) in die Pumpe eingelassen. Ein Teil des Gases gelangt dabei in den zeitweise offenen Zwischenraum der Spiralen. Beim Pumpvorgang wird der Rotor der Pumpe durch den Motor nicht gedreht, stattdessen wird durch Verschiebung des Rotors eine Abschnürung der Luft zwischen den Spiralen bewirkt. Durch Fortsetzen der Verschiebung wird die Luft weiter komprimiert und in das Innere der Spiralen gezwungen, bis sie den Gasauslass im Mittelpunkt der Spiralen erreicht und durch ein Rückschlagventil aus der Pumpe austreten kann. Die kontinuierliche Fortsetzung dieser Bewegung saugt bis zum Entstehen des Endvakuum (hier 10^{-2} mBar) Gase aus dem System heraus.

2.2.5. Turbomolekularpumpen

Um das benötigte Ultrahochvakuum zu erreichen, wird eine 'Turbovac SL80' Turbopumpe von Oerlikon verwendet. Diese Pumpe wird über einen CF Anschluss an das (Hoch-) Vakuumsystem angeschlossen. Der Aufbau ähnelt dem einer Flugzeugturbine, es sind mehrere Rotoren mit angewinkelten Blättern verbaut und zwischen Rotoren sind ebenfalls angewinkelte Statorblätter eingesetzt. Von einem Motor wird der Rotor mit 72.000 U/min gedreht, damit bewegen sich die Rotorblätter mit einer ähnlichen oder höheren Geschwindigkeit, als die vorhandenen Restgas-Moleküle. Diese Moleküle stoßen mit den Rotorblättern zusammen und werden dabei entweder umgelenkt oder von der Oberfläche der Rotorblätter zeitweise adsorbiert. In beiden Fällen bewegen sich die Moleküle nach dem Zusammenstoß, bzw. nach Verlassen der Rotoroberfläche, weiter in die Pumpe hinein, bis sie an dem Gasauslass der Pumpe ausgestoßen werden. Auf der nachfolgenden Abbildung 2.6 ist links (Bildabschnitt a) eine Turbopumpe gezeigt. Erkennbar sind der CF-Flansch für die Hochvakuumseite (oben) und der KF-Flansch für das Vorvakuum (unten rechts). Der rechte Abschnitt b) zeigt den Blick von oben auf den CF Flansch, erkennbar sind dabei die Rotorblätter der Pumpe.

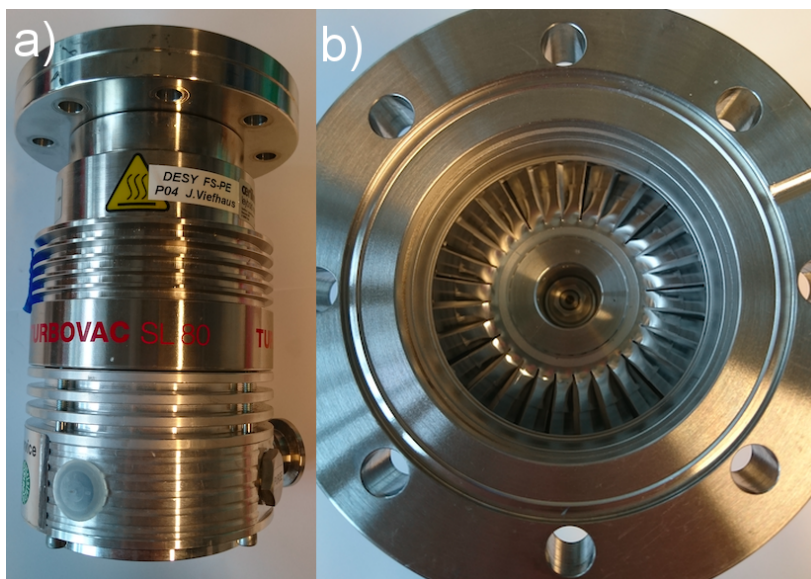


Abbildung 2.6.: a) Gesamtansicht Turbopumpe b) Rotorblätter einer Turbopumpe

Aufgrund der hohen Umdrehungsgeschwindigkeit, muss diese Pumpe mit einem Vorvakuum betrieben werden, um ein Überhitzen der Rotorblätter aufgrund von Luftreibung zu verhindern.

2.2.6. KF-Eckventile

Einzelne Abschnitte des HV Systems werden durch pneumatische KF-Eckventile getrennt. Die Funktionsweise soll beispielhaft an einem KF-Eckventil des Typs '26432-KA41' beschrieben werden. Auf der folgenden Abbildung ist in der rechten Bildhälfte eine Schnittzeichnung des verwendeten Ventils zu sehen.

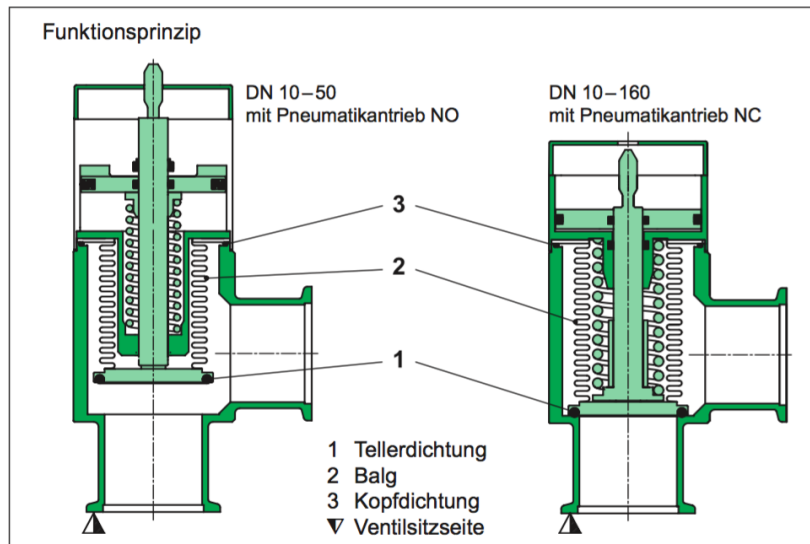


Abbildung 2.7.: Schnittzeichnung eines VAT Eckventils, verwendeter Typ: NC (rechts) [Quelle: VAT¹]

Das Ventil wird mithilfe von Druckluft geöffnet (rechts) bzw. geschlossen (links), dabei trennt ein stählerner Balg den Hochdruckbereich und das Hochvakuum voneinander. Für die weitere Erklärung wird sich auf das verwendete Ventil (rechts) beschränkt. Die Druckluft wird extern zugeführt und mittels eines am Ventil angebrachten Steuerventils zu- oder abgeschaltet. Bei Aktivierung der Druckluft wird der mittlere Zapfen mit der umliegenden Dichtung hochgedrückt und das Ventil öffnet. Wird das Steuerventil abgeschaltet, entlüftet der Bereich unterhalb der Dichtung und das Ventil wird von der eingebauten Feder geschlossen. Im geschlossenen Zustand wird die Tellerdichtung durch die Feder fest gegen das Gehäuse gedrückt, damit ist das Ventil dicht geschlossen. Das Ventil ist so einzubauen, dass der rechts eingezeichnete Flansch mit der Seite mit dem höheren Relativdruck gegenüber dem unteren Flansch verbunden ist. So kann verhindert werden, dass das Ventil durch den Druckunterschied aufgedrückt wird.

¹VAT Vakuumentile AG: VAT-Katalog 2016, S. 183, www.vatvalve.com/docs/default-source/informations-and-downloads/katalog2016_d_web_b.pdf (Zugriff: 03.08.2016)

Eine Erweiterung des KF-Eckventils ist das Softpump-Ventil. Dieses wird um ein weiteres Steuerventil ergänzt und besitzt im Inneren ein zweites Ventil mit einer wesentlich geringeren Ventilöffnung. Ein Ventil dieser Bauart verfügt über drei Schaltzustände: 'geschlossen', 'soft-pump open' und 'offen'. Durch den zusätzlichen 'soft-pump open' Zustand, ist es möglich einen langsameren Druckausgleich als bei gewöhnlichen Ventilen zu erreichen und die sprunghafte Druckänderung auf sensible Teile zu mindern.

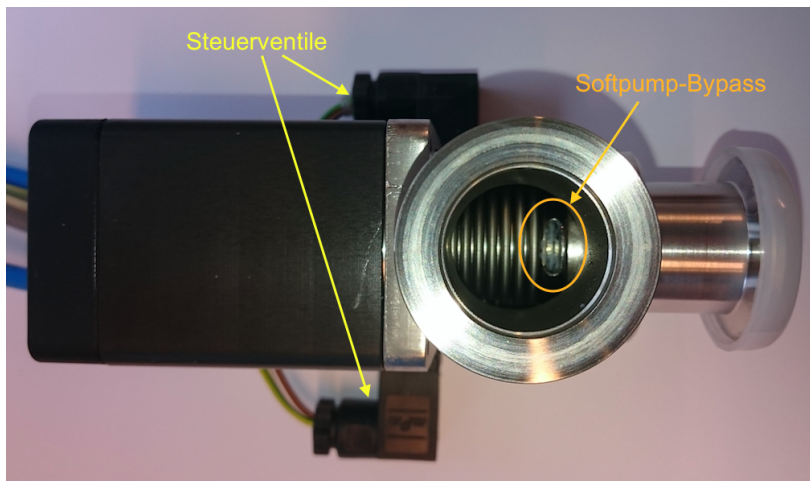


Abbildung 2.8.: Foto eines VAT Softpump Ventils

Abbildung 2.8 zeigt eines der Softpump-Ventile. Innerhalb der Markierung ist die kleine Softpump Öffnung des Ventils zu erkennen, diese ist bei Ansteuerung des Softpump Steuerventils durchlässig.

2.2.7. Swagelok Membranventile

Ein Membranventil führt einen der Prozess-Anschlüsse zur Mitte des Ventils und dort senkrecht nach oben. Auf dem offenen Rohrende sitzt die Membran. Soll das Ventil geschlossen werden, drückt die Membran auf das Rohrende und dichtet dieses ab. Zum Öffnen des Ventils wölbt sich die Membran vom Rohrende weg und ermöglicht dem Medium, in die umliegende Kammer zu fließen. Diese Kammer ist mit dem anderen Prozess-Anschluss direkt verbunden und erlaubt damit einen Fluß des Mediums zwischen den Anschlüssen. Abbildung 2.9 zeigt die Teile eines Membranventils des Herstellers Swagelok.



Abbildung 2.9.: Aufbau eines Swagelok-Membranventils [Quelle: Swagelok²]

Auf der Oberseite des Körpers ist das Rohrende eines der Endanschlüsse sowie die beschriebene umliegende Kammer des anderen Endanschlusses zu erkennen. Im geschlossenen Zustand dichtet die Membran das Rohrende ab und unterbindet damit den Fluß des Mediums zwischen den Endanschlüssen. Der pneumatische Steuerkopf wird für die Operation des Ventils mit Druckluft versorgt. Diese wird genutzt, um der Kraft einer Feder (diese legt fest, ob es sich um ein NC oder NO Ventil handelt) entgegenzuwirken und das Ventil zu öffnen bzw. zu schließen. Über das Gleitstück wird die Kraft des Steuerkopfes mechanisch an die Membran weitergeleitet, welche entsprechend verformt wird.

²Swagelok: Swagelok-Katalog, Membranventile - Serie DP, S. 524, www.swagelok.com/downloads/webcatalogs/de/MS-01-165.PDF (Zugriff: 08.08.2016)

2.2.8. Dosierventil

2.2.9. Thermovac Drucksensoren

Bei den meisten verwendeten Sensoren handelt es sich um 'Thermovac' von Oerlikon (Abbildung 1.1, S13...S21), diese messen in einem Bereich von $5 \cdot 10^{-5}$ bis 1000 mBar und wandeln den erfassten Druck in eine analoge Spannung von 0,61 bis 10 V um. Eine Spannung von ca. 0,5 V weist auf eine fehlerhafte Sensoreinheit hin. Fällt die Spannung darunter ist der Sensor nicht angeschlossen oder das Kabel ist brüchig. Auf der nachfolgenden Abbildung ist ein Thermovac Sensor gezeigt, dieser wird über den Kleinflansch (links) an das Rohrsystem angebunden:



Abbildung 2.10.: Foto eines Thermovac

Auf der rechten Seite (hier nicht sichtbar) befindet sich der elektrische Anschluss des Thermovac. Die Druckmessung basiert auf einem Pirani-Messrohr und erfolgt mit einer Präzision von $\pm 25\%$ ($p < 1000$ mBar), $\pm 5\%$ ($p < 100$ mBar) oder $\pm 10\%$ ($p < 10^{-3}$ mBar). Ein Draht innerhalb des Messrohres wird durch einen Strom auf eine konstante Temperatur (hier 35° C über Umgebungstemperatur) erhitzt und das Gas innerhalb des Rohres leitet die Wärme von dem Draht weg. Die Wärmeleitfähigkeit hängt dabei innerhalb des Messbereiches vom Gasdruck im Inneren des Rohres ab. Indem die Temperatur (damit auch der Widerstand) des Drahtes konstant gehalten wird, kann durch Messung des benötigten Stromes bestimmt werden, wie stark das Messelement durch die Wärmeleitung abgekühlt wird. Da auch die Art des Gases die Wärmeleitfähigkeit beeinflusst, muss der Sensormesswert durch einen Korrekturfaktor angepasst werden.

2.2.10. Ionivac Drucksensoren

Ergänzend zu den Thermovac Sensoren, werden 'Ionivac' Sensoren (ebenfalls von Oerlikon) mit einer Genauigkeit von 15% verwendet (Abbildung 1.1, S4, S11). Diese bieten einen größeren Messbereich von $5 \cdot 10^{-10}$ bis 1000 mBar, indem die Druckmessung bei niedrigen Drücken durch ein Bayard-Alpert-Vakuummeter übernommen wird. Der prinzipielle Aufbau eines solchen Sensors ist in Abbildung 2.11 dargestellt:

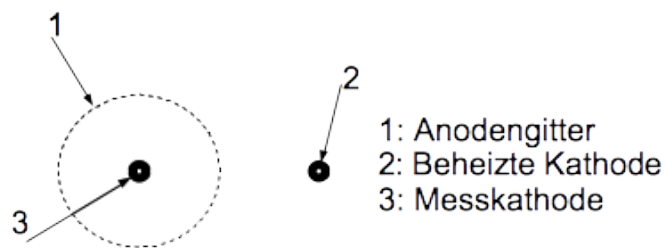


Abbildung 2.11.: Schema-Zeichnung eines Bayard-Alpert-Vakuummeters

Die ergänzende Messmethode basiert auf einer beheizten Kathode (2), welche durch den Edison-Richardson-Effekt Elektronen freisetzt. Diese werden durch das elektrische Feld zwischen der Kathode (2) und der gitterförmigen Anode (1) beschleunigt und lösen beim Zusammenstoß mit Molekülen des verbliebenen Gases Elektronen aus diesen heraus und ionisieren diese dadurch. Da die Messkathode (3) über ein negativeres Potential als die beheizte Kathode verfügt, bewegen sich die Ionen bevorzugt zu der Messkathode. Aus dem Stromfluss an der Messkathode kann bestimmt werden, wieviele Moleküle ionisiert werden und daraus kann geschlossen werden, wieviele Moleküle in dem System noch vorhanden sind.

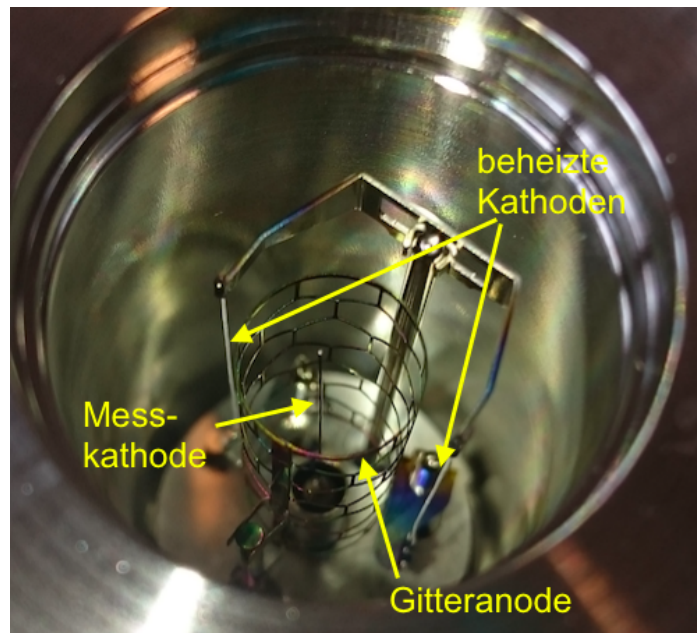


Abbildung 2.12.: Foto des Bayard-Alpert-Vakuummeter im inneren des Ionivac Sensors

Auf Abbildung 2.12 ist in der Mitte die Gitteranode zu erkennen, im Zentrum dieser Anode ist die Messkathode gezeigt und außerhalb der Anode sind die beheizten Kathoden sichtbar. Auch hier wird das analoge 0 bis 10V Ausgangssignal von der Gasart beeinflusst und muss kalibriert werden. Die durch die freigesetzten Elektronen ionisierten Moleküle des Gases werden hauptsächlich von der Messkathode angezogen. Jedoch werden Moleküle auch von der beheizten Kathode angezogen und verunreinigen diese über längere Zeitspannen. Diese Verunreinigungen können durch den verwendeten Sensor selbstständig bereinigt werden, indem die Kathode von einem hohen Strom beheizt wird, was ein Verdampfen der Gasmoleküle bewirkt. Dieser Vorgang wird als 'Degas' (dt. 'Entgasen') bezeichnet.

2.2.11. Ceravac Drucksensoren

Um genauere Messwerte (0,003%) zu erhalten, werden 'Ceravac' Sensoren (Oerlikon) eingesetzt (Abbildung 1.1, S1, S2, S3). Die Ceravac-Sensoren erfassen den Druck durch Messung der Kapazität zwischen einer Metallmembran und einer Elektrode. Aufgrund des Druckunterschiedes wird die Membran verformt, was zu einer veränderten Nähe von Membran und Elektrode führt. Die Kapazität zwischen diesen Oberflächen hängt antiproportional mit der Distanz zusammen. Durch Messung der Kapazität, kann die Verformung der Membran und damit der Druck an dem Ventil bestimmt werden. Das Ceravac bestimmt sein 0 bis 10V

Ausgangssignal als einziger Vakuumsensor, ohne dabei von der Art des Gases abhängig zu sein.

2.2.12. Festo Drucksensor

Für die Überwachung der Druckluftleitung wird ein 'SDE1' Drucksensor von Festo (Abbildung 1.1, S12) verwendet. Dieser bestimmt den Druck in einem Bereich von 0 bis 10 Bar mithilfe eines piezoresistiven Sensors. Dabei wird in ein Silizium Trägermaterial eine Membran eingätzt, welche auf der einen Seite dem Außendruck und auf der anderen Seite dem Druck des zu messenden Systems ausgesetzt ist. Durch die Druckdifferenz wird die Membran verformt, was zu einer Verformung von eingearbeiteten Dehnmessstreifen führt. Durch die Veränderung des Widerstands dieser Dehnmessstreifen, kann bestimmt werden, wie groß die Druckdifferenz ist.

2.2.13. Wika Drucksensoren

Später sollen für die Messzwecke Gase in die Vakuumkammer (siehe Abbildung 1.1, links) eingelassen werden. Der Druck in den dafür verwendeten Gasflaschen wird mithilfe von 'WU-15' Sensoren von Wika (Abbildung 1.1, S5...S10) erfasst. Die Messung erfolgt durch einen Dünnfilm-Drucksensor, dabei handelt es sich um eine Metallmembran mit eingearbeiteten Dünnfilmwiderständen. Durch den Druck wird die Membran verformt und die Leiterbahnen der Dünnfilmwiderstände ändern wie ein Dehnmessstreifen ihre Länge bzw. Dicke und damit ihren Widerstandswert.

3. Entwurf der Pumpensteuerung

3.1. Anforderungen an das System

Die Anforderungen an das zu konstruierende System lauten wie folgt:

- Programmierung in Python
- Lage- bzw. Statusmeldungen der Ventile, Pumpen und Drucksensoren lesen und Fehlererkennung versuchen
- 'Degas' Funktion der Ionivac Sensoren ermöglichen
- Steuerbar über Netzwerk
- Hot-Swap-fähige Spannungsversorgung mit je zwei Netzteilen für 5V und 24V
- Überwachung der Spannungsversorgung und des Druckluftdruckes
- Bei Ausfall der 5V Versorgung über integrierte USV mindestens 10 Minuten aktiv bleiben
- automatischer Wiederanlauf nach Wiederherstellen der Spannungsversorgung
- Status-Pin für Interlock-System
- Display integrierbar
- Absicherung gegen Absturz des Raspberry Pi
- 7 steuerbare Abzweige, davon 3 für die Beamline, 3 für Gäste und 1 für die interne Verwendung
- Taster und Status-LEDs für die Gäste-Abzweige

3.2. Spezifikation des Systems

Basierend auf Abbildung 1.1 wurde mithilfe der Datenblätter der Komponenten eine Tabelle der anzusteuernenden Ein- und Ausgänge sowie der benötigten Stromversorgung erstellt, diese ist im Folgenden angegeben:

Tabelle 3.1.: Benötigte Ein- und Ausgänge und Stromaufnahme der Bauteile

Bauteil	Typ	D. In	D. Out	A. In	A. Out	Strom (A)
V1	KF-Eckventil	1	1	0	0	0,15
V2	KF-Eckventil	1	1	0	0	0,15
V3	Nadelventil	2	0	0	1	0,25
V4	KF-Eckventil	1	1	0	0	0,15
V5	Swagelok Membranv.	1	1	0	0	0,08
V6	Swagelok Membranv.	1	1	0	0	0,08
V7	Swagelok Membranv.	1	1	0	0	0,08
V8	Swagelok Membranv.	1	1	0	0	0,08
V9	Swagelok Membranv.	1	1	0	0	0,08
V10	Swagelok Membranv.	1	1	0	0	0,08
V11	Swagelok Membranv.	1	1	0	0	0,15
V12	KF-Eckventil	1	1	0	0	0,15
V13	KF-Eckventil	1	1	0	0	0,15
V14	KF-Eckventil	1	1	0	0	0,15
V15	KF-Eckventil	1	1	0	0	0,15
V16	KF-Eckventil	1	1	0	0	0,15
V17	Nadelventil	2	0	0	1	0,25
V18	KF-Eckventil	1	1	0	0	0,15
V19	KF-Eckventil	1	1	0	0	0,15
V20	KF Softpump	2	2	0	0	0,3
V21	Schieberventil	2	2	0	0	0,3
V22	KF-Eckventil	1	1	0	0	0,15
V23	KF-Eckventil	1	1	0	0	0,15
V24	KF-Eckventil	1	1	0	0	0,15
V25	KF-Eckventil	1	1	0	0	0,15
V26	KF-Eckventil	1	1	0	0	0,15
V27	KF-Eckventil	1	1	0	0	0,15
V28	KF-Eckventil	1	1	0	0	0,15
V29	KF-Eckventil	1	1	0	0	0,15
V30	KF-Eckventil	1	1	0	0	0,15

Fortsetzung auf folgender Seite.

Bauteil	Typ	D. In	D. Out	A. In	A. Out	Strom (A)
V31	KF-Eckventil	1	1	0	0	0,15
V32	KF-Eckventil	1	1	0	0	0,15
V33	KF-Eckventil	1	1	0	0	0,15
V34	KF-Eckventil	1	1	0	0	0,15
P1	Turbopumpe	2	1	0	0	0
P2	Scrollpumpe	1	1	0	0	0,3
S1	Ceravac	0	0	1	0	0,05
S2	Ceravac	0	0	1	0	0,05
S3	Ceravac	0	0	1	0	0,05
S4	Ionivac	0	1	1	0	1,2
S5	Wika-Wu	0	0	1	0	0,2
S6	Wika-Wu	0	0	1	0	0,2
S7	Wika-Wu	0	0	1	0	0,2
S8	Wika-Wu	0	0	1	0	0,2
S9	Wika-Wu	0	0	1	0	0,2
S10	Wika-Wu	0	0	1	0	0,2
S11	Ionivac	0	1	1	0	1,2
S12	Festo SDE1	0	0	1	0	0,07
S13	Thermovac	0	0	1	0	0,2
S14	Thermovac	0	0	1	0	0,2
S15	Thermovac	0	0	1	0	0,2
S16	Thermovac	0	0	1	0	0,2
S17	Thermovac	0	0	1	0	0,2
S18	Thermovac	0	0	1	0	0,2
S19	Thermovac	0	0	1	0	0,2
S20	Thermovac	0	0	1	0	0,2
S21	Thermovac	0	0	1	0	0,2
Gesamt		41	38	17	2	11,1

In der Tabelle sind die Spalten für die digitalen Ein- bzw. Ausgänge mit einem 'D.' und für die analogen Ein- bzw. Ausgänge mit einem 'A.' in der Spaltenüberschrift gekennzeichnet. Aus der Tabelle geht hervor, dass 79 digitale IO-Pins, 21 analoge Eingänge sowie 2 analoge Ausgänge benötigt werden. Da der Raspberry Pi weder über genügend digitale Pins, noch über analoge Pins verfügt, müssen diese durch externe Massnahmen geschaffen werden.

Die Vielzahl an digitalen Schnittstellen wird mithilfe von Port Expandern hergestellt, dabei handelt es sich um integrierte Schaltungen mit mehreren GPIO-Pins. Der Port Expander wird über ein primitives Bussystem wie SPI oder I2C angesteuert und bietet neben einfacher IO-Funktionalität weitere Optionen, darunter integrierte zuschaltbare Pull-Up- bzw. Pull-Down-Widerstände und einstellbare Interrupt-Pins. Mithilfe von Pull-Up- / Pull-Down-Widerständen ist es möglich, unbeschaltete hochohmige Eingänge mit einem festgelegten Potential zu versehen ohne dabei im beschalteten Zustand den angelegten Pegel zu überschreiben. Durch Interrupt-Pins kann der steuernde Prozessor aus dem Ruhezustand in den aktiven Modus umgeschaltet oder eine asynchrone Befehlsausführung ausgelöst werden, wenn es zu einer Pegeländerung an den Digitaleingängen des Port-Expanders kommt.

Für die analogen Eingänge wird ein einzelner externer Analog-Digital-Wandler verwendet, dieser lässt sich über SPI auslesen. Da dieser ADC nur einen einzigen Eingang bietet, wird dieser durch 4 analoge Multiplexer mit den 21 analogen Kanälen verbunden. Bei Multiplexern handelt es sich um analoge Umschalter, über die vorhandenen 3-Bit Adressleitungen kann ausgewählt werden, welcher Kanal des Multiplexers an den Eingang des Umsetzers weitergeleitet wird.

Das System wird durch einen externen 2-Kanal Digital-Analog-Umsetzer ergänzt. Dem DAC werden über SPI Daten zugeführt, welche intern in analoge Ausgangsspannungen umgewandelt und auf die Ausgänge gegeben werden. Die generierte Spannung kann nach einer Aufbereitung zur Steuerung der Dosierventile verwendet werden.

Zusätzlich zu den genannten Schnittstellen müssen noch weitere Ein- und Ausgänge, um die Versorgungsspannung zu überwachen, Verbindungen zum User Interface herzustellen oder zur internen Verwendung, geschaffen werden. Die folgende Abbildung [3.1](#) zeigt ein Blockschaltbild der Platine:

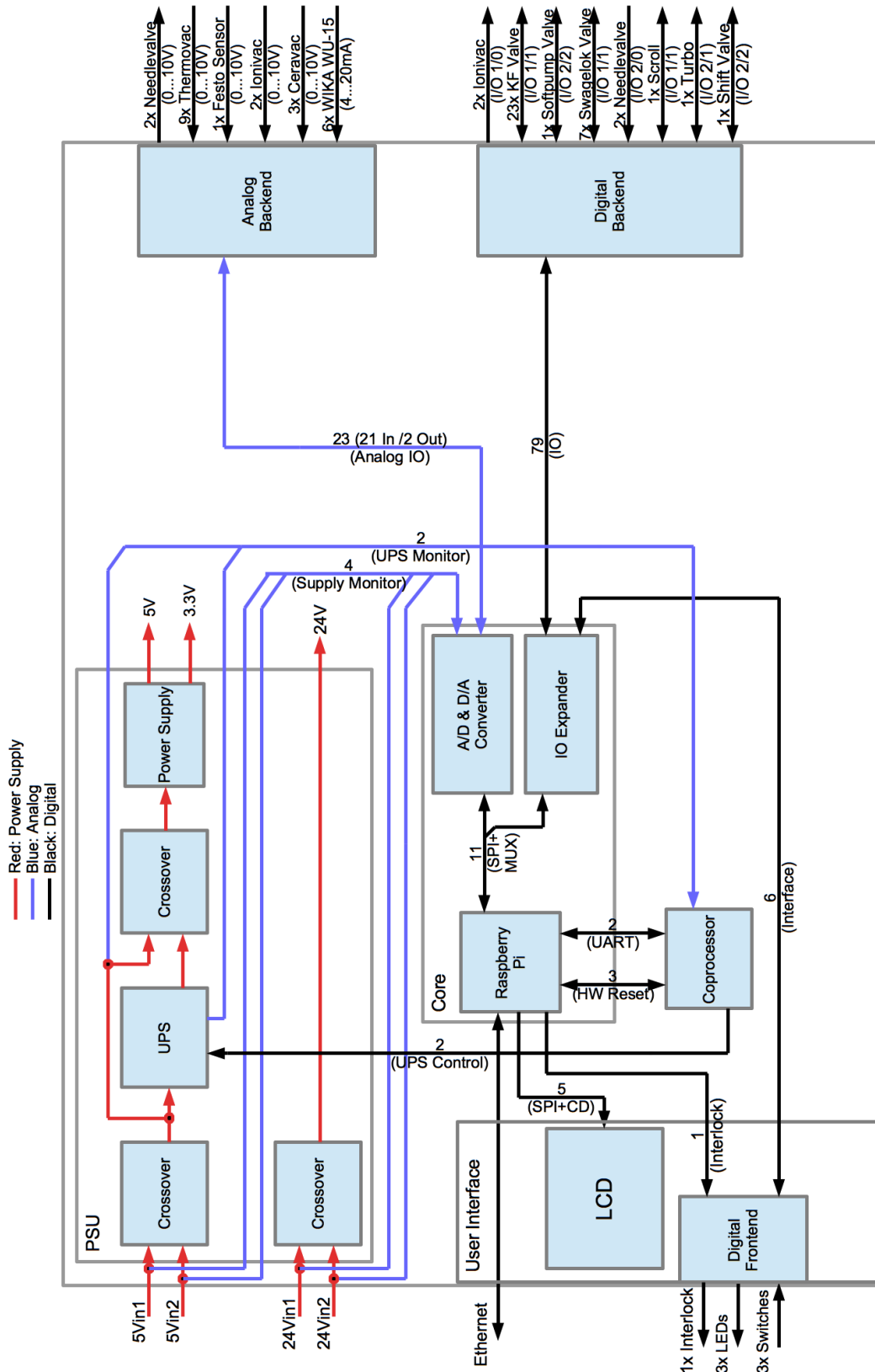


Abbildung 3.1.: Blockschaltbild der Platine

Auf dem Blockschaltbild sind die Verbindungen zwischen den Funktionsblöcken dargestellt, dabei sind direkte Verbindungen von zwei Leitungen durch einen Punkt auf der Verbindung markiert. Leiterbündel sind durch ihre Beschriftung gekennzeichnet, die Zahl oberhalb der Linie gibt dabei an, um wie viele Leitungen es sich handelt. Unterhalb der Linie ist der Bezeichner der Leitung angegeben. Ein- bzw. Austritte aus Leiterbündeln sind in einem Winkel von 45° eingezeichnet.

Aus dem Blockschaltbild geht hervor, dass für die Überwachung der Versorgungsspannung weitere 4 analoge Eingänge und damit insgesamt 25 benötigt werden.

Die Hot-Swap-Fähigkeit der Netzteile wird durch die 'Crossover'-Blöcke erreicht, diese überwachen die Eingangsspannung und schalten selbstständig auf den anderen Kanal um, wenn ein Fehler vorliegt (mehr über die Funktionsweise in Abschnitt 4).

Für die unterbrechungsfreie Spannungsversorgung (UPS-Block) wird ein Akkumulator sowie die dafür benötigten Lade- und Überwachungsschaltungen in das System integriert.

Der eingezeichnete Koprozessor dient der Überwachung des Raspberry Pi und muss über den UART regelmässig von der vollständigen Funktionsbereitschaft in Kenntnis gesetzt werden, bleibt dieses Signal aus sendet der Mikrocontroller eine Anfrage an den Computer, sollte eine Antwort ausbleiben wird dieser zurückgesetzt und neu gestartet. Neben der Funktion als Watchdog Timer ist der Koprozessor dafür zuständig, die unterbrechungsfreie Spannungsversorgung zu überwachen und zu steuern, um sicherzustellen, dass der verwendete Akku eine bestmögliche Lebensdauer erreicht.

Die beiden Backend-Blöcke (Analog Backend und Digital Backend) sind für die Aufbereitung der eingehenden Signale zuständig und sollen die Steuer- und Messelektronik vor Überspannungen schützen.

Das Interlock-System verwendet ein 24V-Signal, dieses muss im Betrieb auf Masse gezogen werden, im Fehlerfall steigt das Signal auf ca. 24V und signalisiert dem Überwachungssystem einen Fehlerzustand.

4. Hardware Design

In diesem Kapitel wird auf Ausschnitte der Schaltpläne eingegangen, der vollständige Schaltplan befindet sich auf der angehängten DVD (Anhang/Hardware/Schaltplan.pdf). Für den Schaltplan wurde ein hierarchisches Design gewählt. Das bedeutet, dass es ein Hauptblatt gibt (S. 1 in Schaltplan.pdf) welches die Subblätter als Blöcke und ihre Verbindungen untereinander aufzeigt. Auf den Subblättern ist die tatsächliche Hardware gezeigt. Verbindungen zwischen dem Hauptblatt und den Subblättern werden durch gelbe Pfeile (sog. 'Sheet-Entries') dargestellt. Eine step-Datei sowie die Gerber-Dateien der Platine befinden sich auf der Anhangs-DVD (Anhang/Hardware/PCB/*). Abbildung 4.1 zeigt ein 3D-Rendering der fertigen Platine:

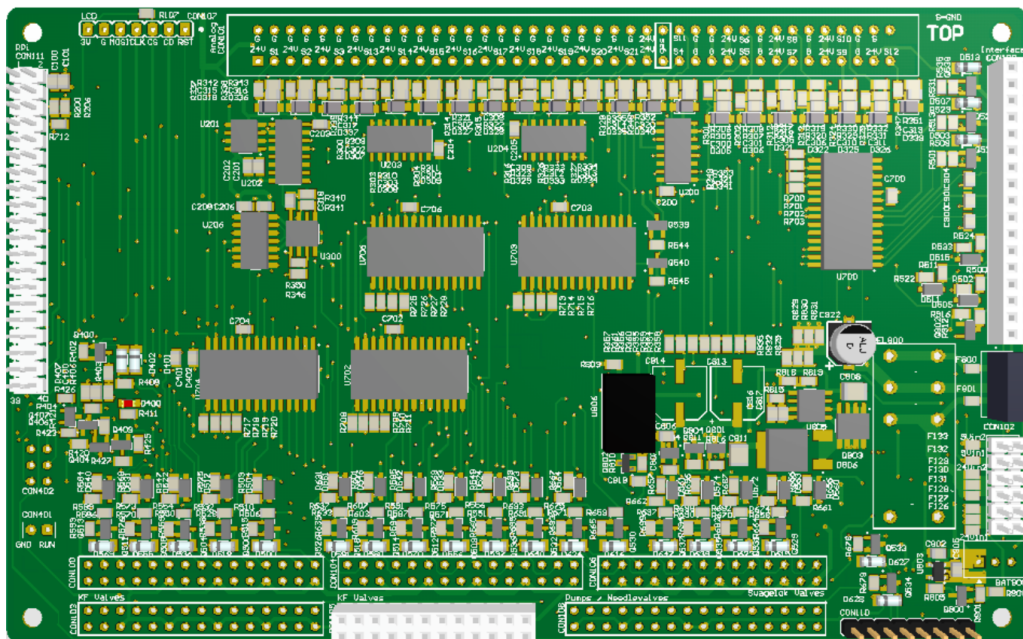


Abbildung 4.1.: 3D-Rendering der Platine

Links sind die Steckerleiste für den Raspberry Pi sowie der Programmieranschluss für den Koprozessor gezeigt. Die unteren Steckerleisten dienen dem Anschluss der Ventile.

Unten rechts befinden sich die Anschlüsse für den Akku der unterbrechungsfreien Stromversorgung sowie die Stromversorgung durch die Netzteile. Darüber liegt die Anschlussleiste für die Nutzerschnittstelle und weitere Ventile. Die obere Steckerleiste stellt die Verbindungen zu den Sensoren her. Es handelt sich um eine 4-lagige Platine mit doppelseitiger Bestückung.

4.1. Digitale Schnittstellen

Die digitalen Ein- und Ausgänge werden durch Port Expander bereitgestellt. Aufgrund der einfachen Ansteuerbarkeit und der hohen Verfügbarkeit werden MCP23S17 von Microchip verwendet. Der MCP23S17 wird über das integrierte SPI des Raspberry Pi angesprochen und bietet zwei Ports mit jeweils 8 Ein- bzw. Ausgängen. Diese Anschlüsse können frei konfiguriert werden und verfügen über integrierte Pull-Up-Widerstände. Ein weiterer Vorteil ist das integrierte Adressiersystem, dieses erlaubt es 8 ICs dieses Typen an einer Chip-Select Leitung zu betreiben und damit insgesamt 128 separate GPIOs zur Verfügung stellen zu können.

Mit Ausnahme des Interlock-Signals und der internen Signale werden alle IO-Signale durch die Port Expander bereitgestellt. Das Interlock-Signal ist ausgenommen, um dieses direkt vom steuernden Computer abhängig zu machen und im Fehlerfall diesen schnellstmöglich melden zu können.

Das Interlock-System gibt über einen Widerstand eine Spannung von 24 V aus, dieses Signal wird im Betrieb durch den Raspberry Pi mithilfe eines Transistors in 'open-collector' Schaltung auf ein niedrigeres Spannungspotential gesetzt. Abbildung 4.2 zeigt diese Schaltung im Detail:

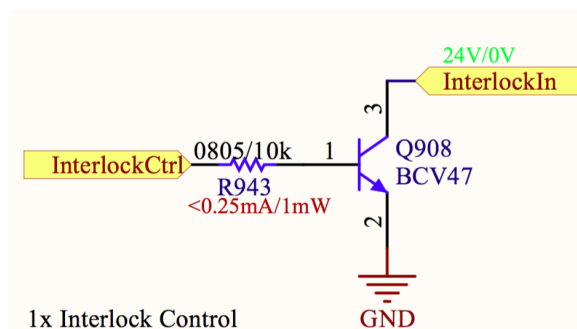


Abbildung 4.2.: Open Collector Schaltung für Interlock-Signal
(InterlockCtrl: Signal von Raspberry Pi; InterlockIn: Interlock-Signalleitung)

Da der Kollektorstrom des Transistors gering ist und der verwendete Transistor einen Verstärkungsfaktor von 40000 aufweist, kann dieser mit einem geringen Basisstrom von $250 \mu A$ gesättigt werden. Im gesättigten Zustand beträgt die Kollektor-Emitter-Spannung maximal 1 V, diese Spannung ist gering genug für das Interlock-System.

Dieselbe Schaltung wird auch für die Steuerung der Feedback-LEDs verwendet, zusätzlich wurden die für den Betrieb der LEDs notwendigen Vorwiderstände am Kollektor ergänzt. Durch den Vorwiderstand von 200Ω wird der Strom durch die blauen Dioden auf $10 mA$ begrenzt.

Die Taster für die Gäste-Abzweige werden über einen Entprell-Kondensator direkt an die Port Expander angeschlossen. Solange der Taster nicht betätigt ist, wird die Spannung am Eingang über den internen Pull-Up-Widerstand auf 3 V gehalten, bei Betätigung schliesst der Taster den Kondensator kurz und die Spannung sinkt auf 0 V.

Die pneumatische Betätigung der Ventile erfolgt über mehrere Steuerventile, diese basieren auf Elektromagneten, welche bei Ansteuerung den Luftfluss freigeben. Ähnlich wie das Interlock werden die Ventile mittels Transistoren in 'open-collector'-Schaltung betrieben. Dafür ist ein Anschluss des steuernden Elektromagneten mit den 24 V der Stromversorgung verbunden und der zweite Anschluss wird über den Transistor nach Masse geschaltet. Da die Steuerventile induktiv arbeiten, kommt es beim schnellen Abschalten des Steuerventils zu einer Spannungsüberhöhung. Um eine Zerstörung des Transistors zu verhindern, muss eine Freilaufdiode integriert werden. Die resultierende Schaltung ist nachfolgend gezeigt:

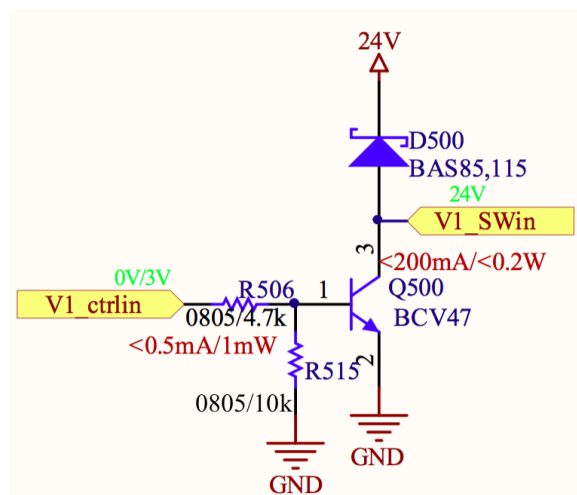


Abbildung 4.3.: Open Collector Schaltung für Steuerventile

(V1_ctrlin: Ausgang von Port Expander; V1_SWin: Low-Side des Ventils)

Die Freilaufdiode (hier D500) ist im Normalbetrieb in Sperrrichtung, beim Abschalten des

Ventils kommt es durch das zusammenbrechende Magnetfeld zu einer Erhöhung der Spannung am Kollektor. Damit befindet sich die Diode in Flußrichtung und limitiert damit die Spannungsüberhöhung auf ca. 24,7 V. Durch R515 kann statt des NPN Bipolar Transistors auch ein N-Channel MOSFET verwendet werden, dies erlaubt eine geringere Verlustleistung. R515 stellt dann sicher, dass der MOSFET bei ausbleibender Ansteuerung abgeschaltet ist.

Die Ventile verfügen über eine Positionsmeldung, diese basiert auf einem elektrischen Schalter, der durch den pneumatischen Kolben betätigt wird. Da diese Ventile in großer Entfernung von der Steuerplatine angebracht sein können, kann es aufgrund der Induktivität der Leitungen beim Umschalten des Positionsmelders zu Schwingungen und Spannungsüberhöhungen kommen. Abbildung 4.4 zeigt links ein Oszilloskopbild eines ungeschützten Eingangs und rechts die verwendete Messschaltung. Erkennbar ist ein deutliches Überspringen der Eingangsspannung auf eine Spannung von 6,6 Volt, diese würde zu einer Zerstörung des Eingangs führen.

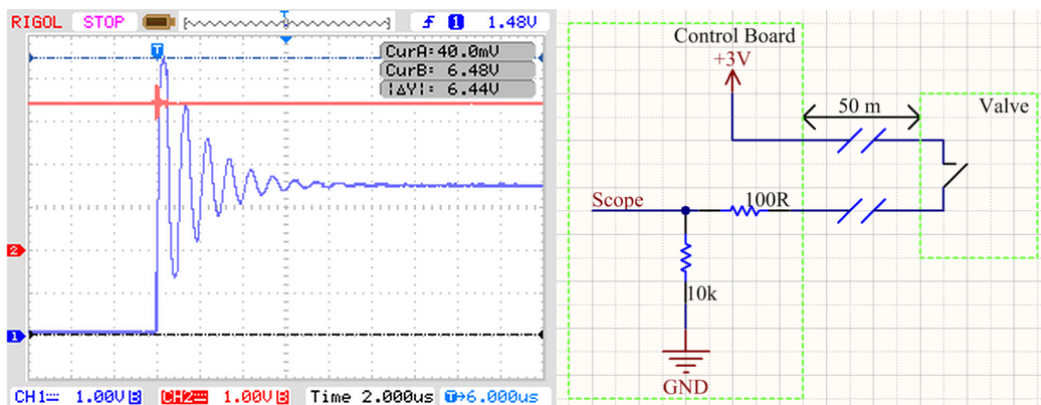


Abbildung 4.4.: Links: Eingangssignal ohne Schutzschaltung (CH1), Versorgungsspannung (CH2); Rechts: verwendete Schaltung

Um eine Zerstörung des digitalen Eingangs zu verhindern, wird eine Zener-Diode verwendet. Dies ist in der folgenden Abbildung 4.5 dargestellt:

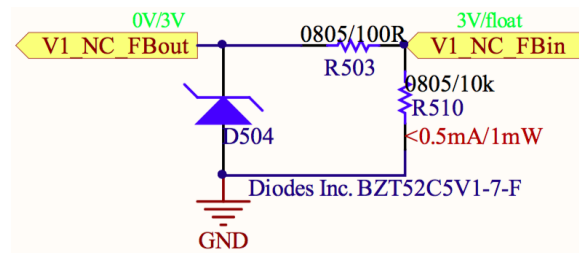


Abbildung 4.5.: Schutzschaltung basierend auf einer Zener-Diode
(V1_NC_FBout: Eingang von Port Expander; V1_NC_FBin: Meldeleitung von Ventil)

Durch Verschieben des $10\text{ k}\Omega$ Pull-Down Widerstandes vor den Strombegrenzungswiderstand, kann der Spannungsabfall verringert werden. Steigt die über die Zener-Diode anliegende Spannung über die Durchbruchspannung, kommt es zum Zener-Durchbruch. Beim Zener-Durchbruch ändert sich der differentielle Widerstand der Diode, dadurch regelt sich die Spannung auf einen Wert nahe der Durchbruchspannung ein.

Nachfolgend ist dargestellt, wie sich der Signalverlauf durch die neue Schaltung verändert:

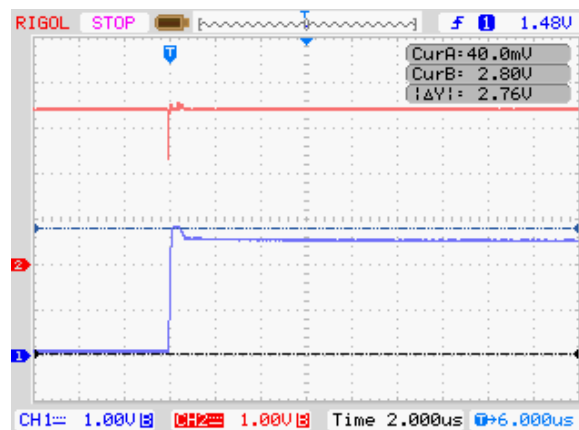


Abbildung 4.6.: Eingangssignal mit Zener-Dioden Schutzschaltung (CH1), Versorgungsspannung (CH2)

Es kommt weiter zu einem Überschwingen der Eingangsspannung, dieses ist jedoch wesentlich geringer und befindet sich innerhalb der Grenzen der Eingangsschaltung. Die Vorteile dieser Schaltung bestehen in der kurzen Reaktionszeit, dem geringen Bauteileaufwand und der hohen Zuverlässigkeit, nachteilig ist die Ungenauigkeit der resultierenden Spannung und der notwendige, durch die Diode fließende, Strom.

4.2. Analoge Schnittstellen

Für das System werden 21 externe und 4 interne (Überwachung der Versorgungsspannungen) analoge Eingänge benötigt. Außerdem sind zwei externe analoge Ausgänge notwendig.

Die analogen Eingänge der Platine werden durch einen MCP3201 Analog-Digital-Wandler geschaffen. Dieser verfügt über eine Auflösung von 12-Bit und wird, wie die Port Expander, über SPI ausgelesen. Da der AD-Umsetzer nur über einen einzelnen Kanal verfügt, müssen die benötigten Kanäle über Multiplexer auf diesen Kanal des ADC gelegt werden. Dafür werden vier CD4051 Multiplexer verwendet, einer davon, um zwischen den Ausgangskanälen der anderen drei umzuschalten. Jeder der Multiplexer (auch MUX) verfügt über 8 Eingänge, welche auf einen durch die Adressleitungen bestimmten Ausgang durchgeschaltet werden. Über 5 Adressleitungen können damit die benötigten 25 analogen Eingangskanäle durch den Raspberry Pi adressiert werden.

Die meisten Sensoren geben ihren Messwert als analoge Spannung zwischen 0 und 10 Volt an das System weiter, daher soll zunächst beispielhaft auf die Aufbereitung eines dieser Signale eingegangen werden. Da der AD-Wandler nur Eingangssignale zwischen 0 V und der Referenzspannung (hier: 3,3 V) verarbeiten kann, muss die Eingangsspannung durch einen Spannungsteiler angepasst werden. Abbildung 4.7 zeigt die verwendete Eingangsschaltung:

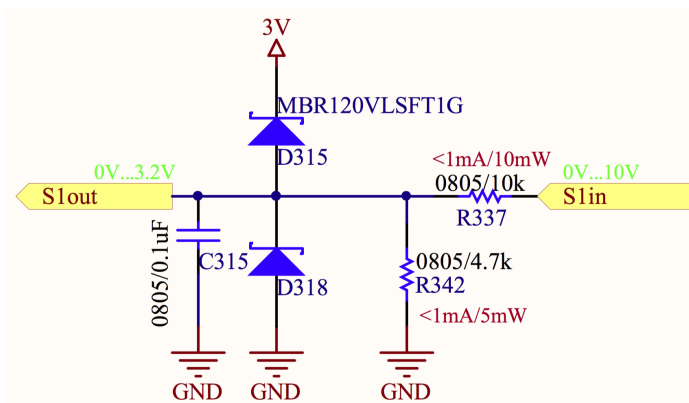


Abbildung 4.7.: Signalaufbereitungsschaltung der analogen 0...10V Eingänge
(S1out: Eingang des ADC über MUX; S1in: Signal von Sensor)

R337 und R342 teilen die eingehende Spannung durch einen Faktor von 3,13 auf einen sicheren Wert. Der Kondensator C315 bildet in Verbindung mit dem Spannungsteiler einen

Tiefpass ($f_g \approx 500$ Hz) und ermöglicht damit eine genauere Messung der Eingangsspannung mit geringem Rauschen. An den analogen Schnittstellen der Platine kommt es nicht zu regelmässigen Schaltvorgängen, dennoch kann es beim Einstecken der Sensoren zu ähnlichen Überschwingungen wie bei den digitalen Anschlüssen kommen. Neben den Einsteckvorgängen, kann es auch durch fehlerhaftes Anschließen zu einer Überspannung kommen. Um eine Zerstörung des analogen Eingangs zu verhindern, ist auch hier eine Schutzschaltung notwendig. Durch die Dioden D315 und D318 wird die am MUX anliegende Spannung auf einen sicheren Bereich begrenzt. Aufgrund der zuvor genannten Nachteile kann die für die digitalen Eingänge verwendete Schutzschaltung mit einer Zener-Diode nicht verwendet werden, der hohe Stromfluß durch die Diode würde zu einer Verfälschung der Messwerte führen. Die verwendeten Schottky-Dioden besitzen einen geringen Sperrstrom, damit ist die Verfälschung der Messwerte minimal. Sinkt die anliegende Spannung unter 0 V (beispielsweise durch Verpolung) beginnt D318 zu leiten und begrenzt die Unterspannung auf ca. $-0,3$ V, bei einer Überspannung leitet D315, durch den abfließenden Strom wird die Überspannung auf ca. $+3,6$ Volt limitiert. Die verwendete Klemmschaltung reagiert sehr schnell, ist sehr zuverlässig und hat einen geringen Leckstrom, nachteilig ist der erhöhte Bauteileaufwand.

Die Drucksensoren des Gaseinlasssystems geben den Druck als Strom zwischen 4 und 20 mA aus. Für die Verarbeitung der Messwerte durch den AD-Wandler muss dieser Strom in eine Spannung zwischen 0,6 V und 3 V umgewandelt werden. Abbildung 4.8 zeigt die Umsetzung dieses Eingangssignals:

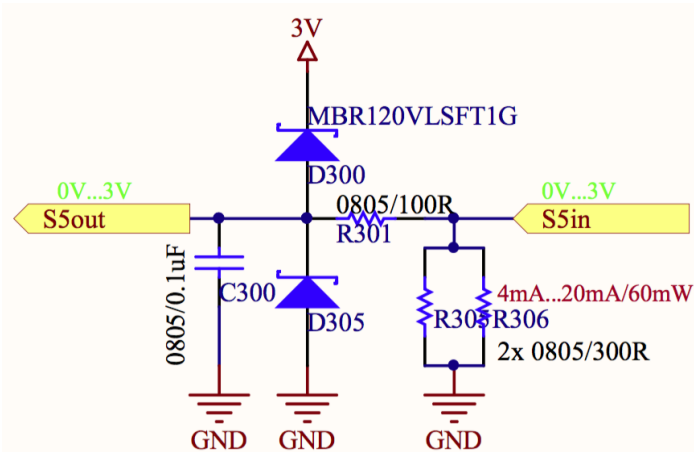


Abbildung 4.8.: Signalaufbereitungsschaltung der analogen 4...20mA Eingänge (S5out: Eingang des ADC über MUX; S5in: Signal von Sensor)

Die notwendige Umwandlung erfolgt durch die Widerstände R305 und R306. Auch hier wur-

de die Zener-Dioden-Schutzschaltung durch zwei Schottky-Dioden ersetzt. Die Filterung des Signals erfolgt durch R301 und C300 mit einer Grenzfrequenz von ungefähr 16 kHz. Der tatsächliche Wert von R301 muss im realen System optimiert werden, um eine Zerstörung durch eine zu hohe Eingangsspannung zu verhindern.

Die Überwachung der Spannungsversorgung benötigt keine hohe Präzision, daher werden die Spannungen der Netzteile durch Spannungsteiler für die Verarbeitung herabgesenkt. Eine weitere Absicherung ist nicht notwendig, da hochwertige Netzteile verwendet werden und die Netzteileingänge durch Suppressordioden vor Spannungsschwankungen geschützt werden.

Mithilfe eines MCP4922 Digital-Analog-Umsetzers werden die analogen Ausgänge zur Verfügung gestellt. Der MCP4922 verfügt über zwei Kanäle, eine Auflösung von 12-Bit und wird ebenfalls über SPI mit Daten versorgt.

Für die Dosierventile des Gaseinlasssystems werden zwei analoge Spannungen zwischen 0 V und 10 V benötigt. Der DAC liefert jeweils eine analoge Ausgangsspannung von 0 V bis ca. 3,3 Volt, diese Spannung wird durch Operationsverstärker angehoben:

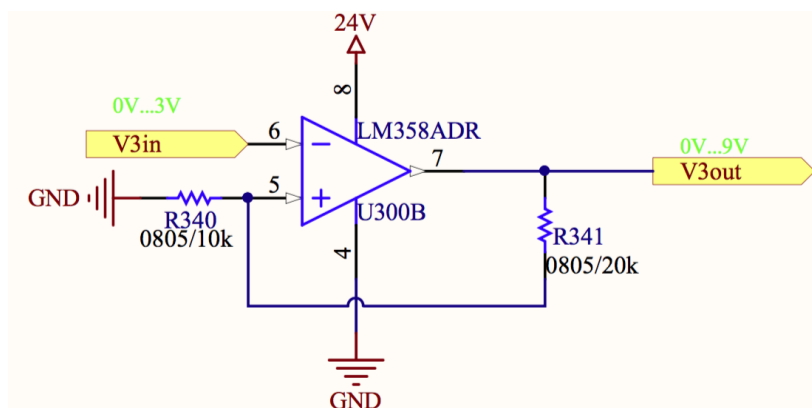


Abbildung 4.9.: Signalaufbereitungschaltung der analogen 0...10V Ausgänge (V3in: Signal von DAC; V3out: Eingang des Dosierventils)

Der Operationsverstärker hebt die Spannung um einen Faktor von 3 auf maximal ca. 9,9 V an, dieser Wert genügt für die Öffnung des Ventils. Eine weitere Absicherung der Ausgangskanäle ist nicht notwendig, da der Operationsverstärker kurzschlussfest ist und mit einem großen Spannungsbereich am Ausgang arbeiten kann.

4.3. Ausfallsicherheit

Der Einsatz am DESY setzt Ausfallsicherheit voraus, für dieses System bedeutet Ausfallsicherheit, dass Fehler der Sensoren und Ventile erkannt werden und das System im Falle eines internen Fehlers in einen sicheren Zustand übergehen kann. In diesem sicheren Zustand soll die Vorpumpe P2 (wenn möglich) eingeschaltet sein und alle Ventile sollen geschlossen sein. Außerdem soll der Raspberry Pi bei Stromausfall für einige Minuten weiter aktiv bleiben können, um bei schnellem Wiederherstellen der Spannungsversorgung schnell wieder arbeiten zu können oder andernfalls kontrolliert herunterzufahren. Um sicherstellen zu können, dass die Ventile bei einem vollständigen Stromausfall der 24 V Versorgung geschlossen werden, sind alle Ventile so gewählt, dass ihre Ruhestellung geschlossen (NC) ist.

4.3.1. Spannungsversorgung

Die Versorgung der Ventile und Sensoren erfolgt durch zwei redundante 24 V Netzteile, der Strom fließt zuerst durch eine Sicherung und an einer Suppressordiode vorbei und erreicht dann folgende Schaltung:

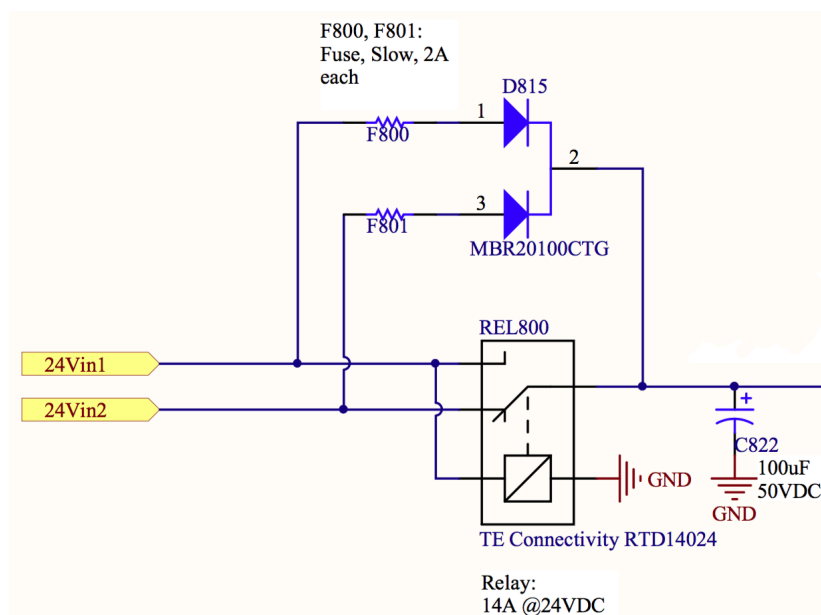


Abbildung 4.10.: Automatische Umschaltung der 24V Versorgung
(24Vin1: Anschluß erstes Netzteil; 24Vin2: Anschluß zweites Netzteil)

Das Relais REL800 arbeitet als Umschalter zwischen den Eingangsspannungen, liegt an '24Vin1' eine Spannung an, wird die Spule des Relais von Strom durchflossen und es schließt. Damit ist der Ausgang mit '24Vin1' verbunden. Liegt an '24Vin1' keine Spannung an, wird '24Vin2' automatisch auf den Ausgang durchgeschaltet. Gemäß Datenblatt hat REL800 eine Umschaltzeit von etwa 8 Millisekunden. C822 dient dazu die Ausgangsspannung zu glätten und beim Umschalten kurzzeitig die Schaltung mit Strom zu versorgen. Ergänzend zu C822 wird D815 verwendet, um die Ausgangsspannung während der kurzen Umschaltphase stabil zu halten. Sollte ein Fehlerfall zu einem ständigen Stromfluß durch D815 führen (z.B. Verschweissen der Relaiskontakte), verhindern die Sicherungen F800 oder F801 eine Beschädigung der Diode.

Danach wird die Versorgungsleitung nach Sensorversorgung und Ventilversorgung aufgeteilt:

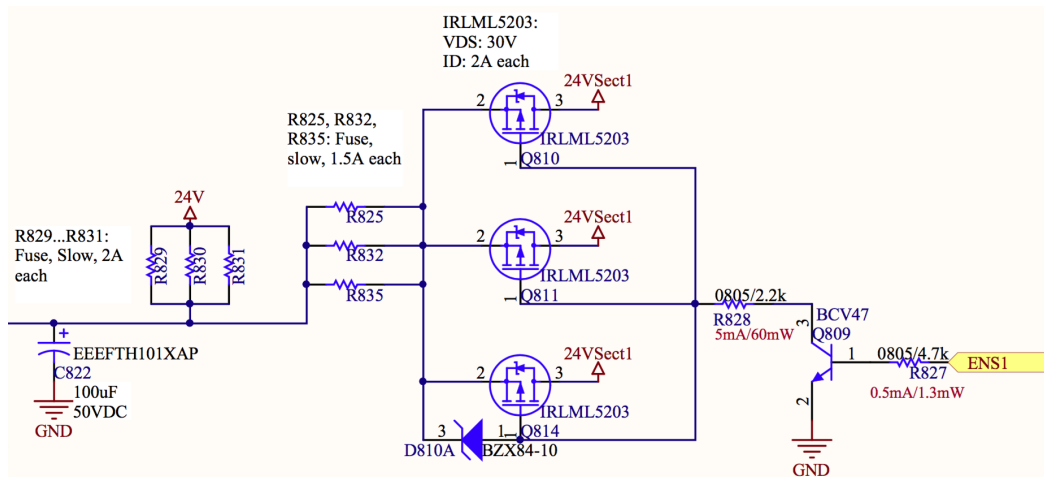


Abbildung 4.11.: Aufteilung der 24V Versorgung

(24V: Sensoren und Scrollpumpensteuerung; 24Vsect1: Abschaltbare Versorgung der Ventile; ENS1: Enable Signal von Raspberry Pi)

Die Ventile werden über drei parallele Sicherungen versorgt. Zusätzlich zu der Sicherung befinden sich parallel geschaltete MOSFETs in Reihe mit den Ventilen, durch diese Vorkehrung können die Ventile im Fehlerfall global deaktiviert werden. Folgende Fehlerfälle lassen sich damit kontrollierbar machen:

1. Zerstörung/Latch-up eines Port Expanders
2. Zerstörung/Latch-up eines für Ventilsteuerung verwendeten Transistors
3. Absturz des Raspberry Pi

Während des Betriebes führt das Versagen von Halbleiter-Bauteilen in den häufigsten Fällen zu einem Kurzschluss des Halbleiters [Quelle: Panasonic¹]. Daher werden die MOSFETs Q810 bis Q814 eingesetzt, um bei Zerstörung eines der anderen Bauteile das Netz '24Vsect1' und damit die Ventile abschalten zu können. Im Falle eines Absturzes des Raspberry Pi, setzt der Koprozessor den Raspberry Pi zurück und die Feldeffekttransistoren öffnen die Verbindung zu den Ventilen.

Die Sensoren sowie die Steuerleitung der Scrollpumpe werden über drei parallele 2 Ampere Sicherungen angeschlossen. Damit sind die Sensoren sowie die Scrollpumpe immer verfügbar, solange die Netzteile 24 V zur Verfügung stellen.

Der Raspberry Pi und die Schaltkreise der Platine werden aus zwei redundanten Netzteilen mit 5 V versorgt. Da diese Stromversorgung mit einer geringeren Leistung arbeitet, kann die Umschaltung zwischen den Netzteilen durch ideale Dioden statt eines Relais erfolgen:

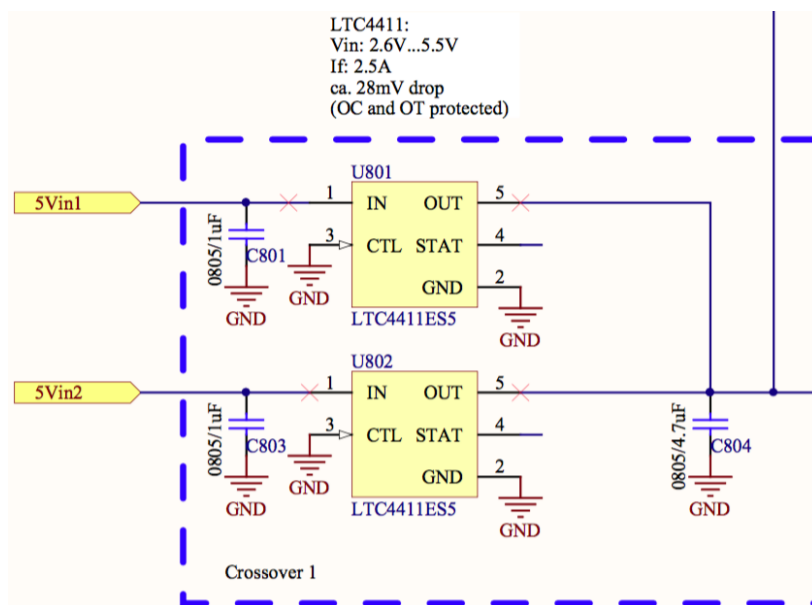


Abbildung 4.12.: Automatische Umschaltung der 5V Versorgung
(5Vin1: Anschluß erstes Netzteil; 5Vin2: Anschluß zweites Netzteil)

Die LTC4411 imitieren das Verhalten von Dioden: Liegt an Pin 1 eine höhere Spannung als an Pin 5 an, beginnt das IC einen Feldeffekttransistor zwischen den Pins durchzusteuern. Andernfalls wird der Transistor gesperrt (die 'Diode' arbeitet in Sperrrichtung). Während des Betriebs wird die Spannungsdifferenz zwischen den Pins auf einen Wert von etwa 28 mV

¹Panasonic: Failure Mechanism of Semiconductor Devices, T04007BE-3 2009.4, Tabelle 3.1 S. 3-4, www.semicon.panasonic.co.jp/en/aboutus/pdf/t04007be-3.pdf (Zugriff: 12.08.2016)

geregelt. Diese Spannung entspricht der Flußspannung der Diode, ist dabei jedoch deutlich geringer als die 300 mV einer standard Schottky-Diode. Diese Eigenschaften machen dieses Bauteil ideal für die Parallelschaltung von Netzteilen.

Die stabilisierte Eingangsspannung wird, wie in Abbildung 3.1 dargestellt, an die Unterbrechungsfreie Stromversorgung (siehe Abschnitt 4.3.2) weitergeleitet. Danach wird mit einer Schaltung wie in Abbildung 4.12, zwischen dem Ausgang der USV und der stabilisierten Eingangsspannung umgeschaltet. Die Ausgangsspannung wird von einem Schaltregler auf die für die Schaltkreise benötigte 3,3 V Gleichspannung herabgesetzt und an die Port Expander, Wandler und Multiplexer verteilt.

Für die Auslegung der Netzteile, der USV und der Platine wurde eine Tabelle angelegt, um einen Überblick über den Stromverbrauch der Baugruppen zu bekommen:

Tabelle 4.1.: Stromaufnahme der 3V und 5V Komponenten

Blatt	Bauteilgruppe	I(5V) in mA	I(3V) in mA
main			
	Raspberry Pi	600	0
	Display	0	10
Power supply & UPS			
	Crossover 1	0,5	0
	Batterie Laderegler	110	1
	Crossover 2	0,5	0
	DCDC Wandler UPS	0	1
	Raspberry Pi Enable	0	1
	Spannungsregler 5V zu 3V	1	0
	Enable 24V Sector	0	0
Digital Backend			
	KF Ventile	0	22
	Swagelok Ventile	0	7
	Softpump Ventil	0	2
	Turbopumpe	0	1,5
	Scrollpumpe	0	0,5
	Ionivac Degas	0	1
	Nadelventil Feedback	0	0
	Schieberventil	0	2
IO Expander			
	SPI CS Pullup Widerstand	0	0,5
	Port Expander Chips	0	6
Analog Backend			
	-keine 3V/5V Verbraucher-	0	0
A/D & D/A Converters			
	SPI CS Pullup Widerstaende	0	1
	Analog-Digital-Converter	0	0,5
	Multiplexer	0	0,5
	Digital-Analog-Converter	0	1
User Interface			
	Buttons	0	0,5
	LEDs	30	1
	Interlock	0	0,5
Coprocessor			
	Analog inputs	1	0,5
	LED	0	3
	Reset Pull-Up Widerstand	0	0,5
	Atmega328p	0	4
	Programmieradapter Levelshifter	0	2
		743	70,5

Auf den Stromverbrauch der 5V-Versorgung müssen die 70,5 mA der 3V-Versorgung addiert werden, da die 3,3 Volt durch einen Spannungswandler aus den 5 Volt generiert werden. Damit ergibt sich ein Strom von 813,5 mA (ab hier aufgerundet auf 1 A bzw. 5W) aus der 5V-Versorgung.

4.3.2. Unterbrechungsfreie Stromversorgung

Um bei Ausfall der beiden Netzteile (Beispielsweise durch Netzverlust) betriebsbereit zu bleiben und nach einer Zeitdauer sicher herunterzufahren, wird eine USV verwendet. Diese besteht aus drei wesentlichen Teilen:

- Laderegler
- Energiespeicher
- Ausgangsspannungswandler

In diesem System wird ein Lithium-Ionen Akku verwendet. Die Vorteile dieses Akkumulator-Typs sind die geringe Selbstentladung und die hohe Strombelastbarkeit. Der Nachteil dieser Akkus ist die geringe Toleranz gegenüber Fehlbehandlung. Der verwendete Speicher hat eine Kapazität von 4,5 Ah mit einer nominalen Spannung von 3,7 Volt, diese Kapazität genügt, um den Computer für ca. 2,5 Stunden, statt der geforderten 10 Minuten, lauffähig zu erhalten. Die zusätzliche Kapazität hilft, die Entladetiefe zu verringern und die Lebensdauer des Akkus zu erhöhen. Wie zuvor erwähnt, sind die verwendeten Speicher sehr empfindlich, die Zellspannung muss zwischen 2,5 V und 4,2 V und die Temperatur zwischen 0 °C und 45 °C liegen. Werden diese Grenzen nicht eingehalten, führt dies zur Zerstörung des Akkus. Um die Sicherheit zu gewährleisten, wird ein Akkumulator mit integrierter Schutzschaltung verwendet, darüber hinaus wird der Ladezustand durch den Koprozessor überwacht.

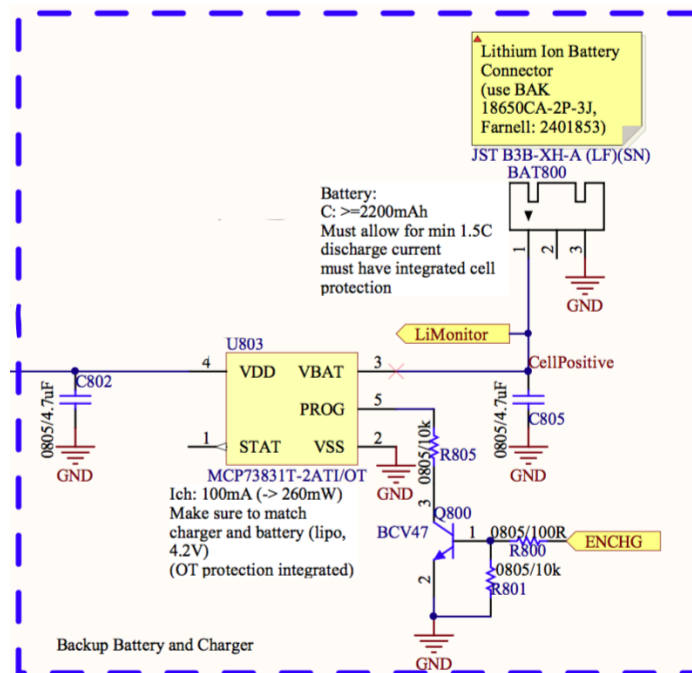


Abbildung 4.13.: Laderegler der Unterbrechungsfreien Stromversorgung

(ENCHG: Enable-Signal des Koprozessors; LiMonitor: Analoger Eingang des Koprozessors; links: Stabilisierte Eingangsspannung nach Crossover 1)

Geladen wird die Batterie durch einen MCP73831 Laderegler, welcher durch einen Widerstand auf einen Ladestrom von 100 mA eingestellt wird. Lithium Akkus werden zuerst mit einem konstanten Strom bis zu einer Zellspannung von 4,2 V geladen und dann durch anlegen einer konstanten Spannung vollständig geladen.

Da der Akku eine unregulierte Spannung zwischen 2,5 V und 4,2 V liefert, ist ein DCDC Aufwärtswandler notwendig, um den Raspberry Pi mit einer geregelten Spannung von 5 Volt zu versorgen:

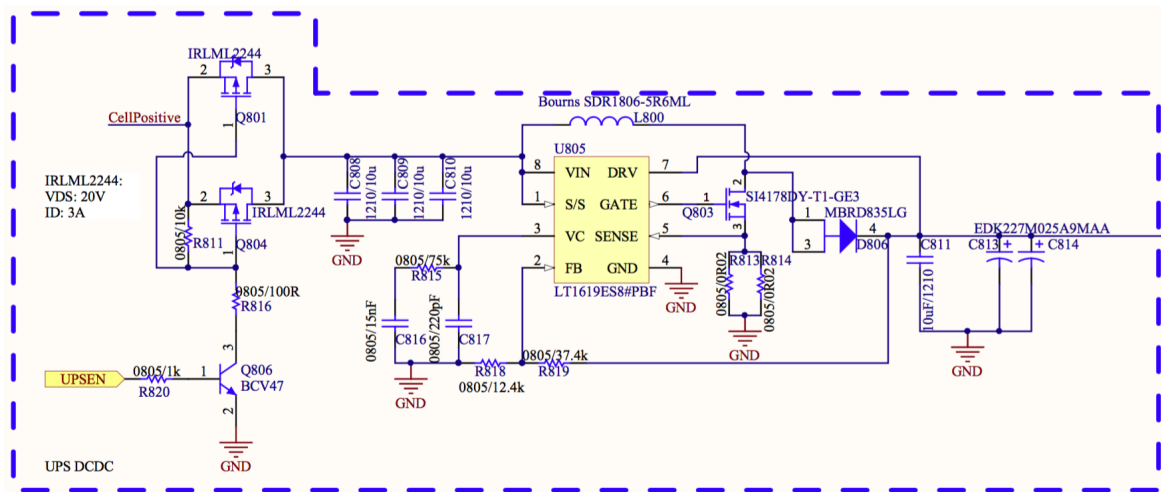


Abbildung 4.14.: Aufwärtswandler der Unterbrechungsfreien Spannungsversorgung
(UPSEN: Enable-Signal des Koprozessors; CellPositive: Positiver Anschluss des Energiespeichers)

Das gezeigte Design des Wandlers entspricht dem Referenzdesign aus dem Datenblatt des LT1619. Zur Erzeugung der höheren Ausgangsspannung schaltet der Schaltregler die Eingangsspannung durch eine Induktivität nach Masse, dabei wird ein Magnetfeld um die Induktivität aufgebaut. Beim Abschalten entsteht durch das einbrechende Magnetfeld eine höhere Spannung, welche nach Glättung durch den Raspberry Pi verwendet werden kann. Durch die vorgeschalteten Transistoren Q801 und Q804 kann der Wandler vollständig abgeschaltet und eine Tiefenentladung des Akkus verhindert werden.

4.3.3. Koprozessor

Auf dem Board ist ein Koprozessor in Form eines Atmel AVR Atmega328 integriert, wie bereits erklärt ist dieser für die Kontrolle der USV zuständig und steuert dafür den Laderegler und den Aufwärtswandler an. Dafür liest der AVR einen Spannungsteiler die Spannung der Versorgungsleitung sowie die Zellspannung des Akkumulators. Um den Standby-Belastung des Akkus zu verringern, wird dessen Spannungsteiler über einen Transistor geschaltet.

Neben der Überwachung der Unterbrechungsfreien Stromversorgung, ist der Koprozessor für die Überwachung der netzgebundenen 5 Volt Stromversorgung zuständig. Bei Ausfall der Netzversorgung schaltet die ideale Diode automatisch auf die USV um, nach einer Zeit von 10 Minuten wird der Raspberry Pi über den UART aufgefordert herunterzufahren. Nach einer

weiteren Wartezeit wird dann durch einen MOSFET die Verbindung von Raspberry Pi und Stromversorgung getrennt (siehe Abbildung 4.15).

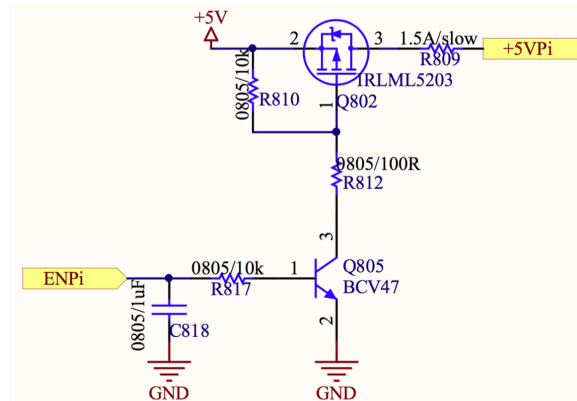


Abbildung 4.15.: Schalter für Raspberry Pi Stromversorgung

(ENPi: Enable-Signal des Koprozessors; +5VPi: Schaltbare Versorgungsspannung des Raspberry Pi)

Diese Möglichkeit die Stromversorgung des Raspberry Pi zu unterbrechen wird ebenfalls genutzt, um diesen nach Wiederherstellung der Netzverbindung erneut zu starten oder nach einem Fehlerfall einen vollständigen Reset auszuführen. Durch den vollständigen Reset können auch mögliche Probleme in der Hardware des Raspberry Pi behoben werden und das Gerät wird komplett neu initialisiert.

Weiterhin arbeitet der Koprozessor als flexibler Watchdog Timer. Dieser stellt durch regelmäßige Kommunikation über den UART sicher, dass der Raspberry Pi aktiv ist und seine Aufgaben erfüllen kann. Bleibt eine UART Meldung des Raspberry Pi aus, sendet der Koprozessor eine Aktivitäts-Anfrage über den UART. Wenn auch diese Anfrage unbeantwortet bleibt, schaltet der AVR den Raspberry Pi ab und lässt diesen einen Neustart ausführen.

5. Software Design

In diesem Kapitel wird näher auf die Software für den Betrieb des Systems eingegangen. Zunächst wird die Software des Raspberry Pi behandelt und im zweiten Abschnitt, dann der Code für den Koprozessor. Beschrieben wird der Code mithilfe von Programmablaufplänen. Der geschriebene Raspberry Pi Programmcode befindet sich auf der Anhangs-CD unter 'Anhang/Software/RaspberryPi/*'. Für den Koprozessor befindet sich der Code im Ordner 'Anhang/Software/Coprocessor/'. In beiden Fällen handelt es sich bei den mit 'Utils' endenden Dateien um begleitende Dateien, das Hauptprogramm befindet sich in 'pumpPi.py' beziehungsweise 'coprocessor.c'.

5.1. Raspberry Pi - Code

In den nachfolgenden Abschnitten werden die Programmabläufe des Python Codes auf dem Raspberry Pi näher erläutert.

5.1.1. Programmablauf Überblick

Nachfolgend ist der Programmablauf des Raspberry Pi Codes im Überblick dargestellt:

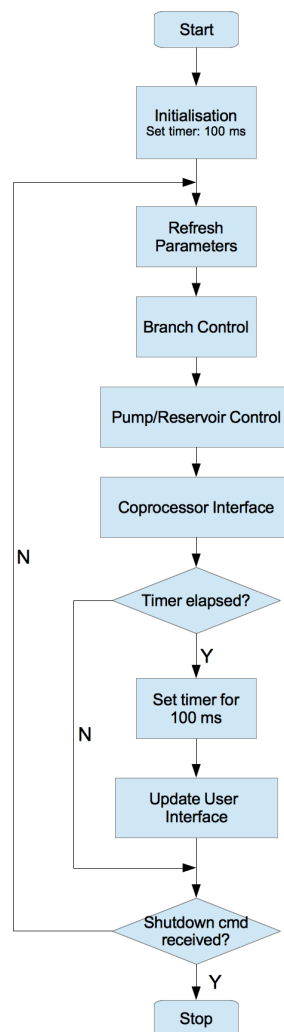


Abbildung 5.1.: Programmablaufplan Raspberry Pi Überblick

Zuerst wird das System initialisiert, dabei werden die Systemzustände sowie die Timer für die Nutzerschnittstelle aufgesetzt. Außerdem werden die Objekte für die Steuerung der einzelnen Abzweige, die Steuerung der Scrollpumpe und für das Interface zu dem Koprozessor erzeugt. Zeitgleich werden die Variablen, die Ein- bzw. Ausgänge der Objekte und die serielle Schnittstelle zum Koprozessor konfiguriert und dem Koprozessor über die serielle Schnittstelle der erfolgreiche Systemstart signalisiert. Anschließend geht das Programm in eine Schleife über, in dieser Schleife werden die Drücke und Ventilzustände der Abzweige, der Pumpe und des Reservoirs aktualisiert und auf Fehler überprüft. Danach evaluiert das Programm die Drücke und behandelt den Abzweig in der Abzweigsteuerung. Im Anschluss überwacht und steuert das Programm das Verhalten der Vorvakuumsystems und behandelt

dann mögliche Anfragen und Signale vom Koprozessor. In einem Zeitraster von 100 Millisekunden wird die Nutzerschnittstelle eingelesen und angesteuert. Wenn vom Koprozessor ein Befehl zum Herunterfahren des Systems ausgegeben wurde, wird das System dann heruntergefahren, anderenfalls beginnt ein neuer Zyklus.

5.1.2. Programmablauf Abzweigsteuerung

Die Abzweigsteuerung befindet sich nach der Initialisierung im Zustand BRANCH_IDLE (BranchState = 1), in diesem Zustand wird der abzapfende Abzweig ausgewählt. Gibt es Abzweige, in denen der Istdruck den Solldruck übersteigt, wird davon der Abzweig mit der höchsten Priorität ausgewählt. Anderenfalls wird geprüft, ob es einen neuen Abzweig gibt und dieser gegebenenfalls ausgewählt. Nach Auswahl eines Abzweiges geht das System in den Zustand BRANCH_PREPARE (BranchState = 2) über. Es wird geprüft, ob der Vordruck des Reservoirs ausreicht, um den Abzweig zu evakuieren. Sofern notwendig, wird die Pumpe zugeschaltet und der Reservoir-Zieldruck angepasst. Zuletzt befindet das System sich im Zustand BRANCH_PUMPING (BranchState = 3). Das zum ausgewählten Abzweig gehörende Ventil wird geöffnet und der Abzweig wird abgepumpt. Dieser Zustand wird beibehalten bis der Zieldruck erreicht ist oder ein anderer Abzweig mit höherer Priorität das Solldruckfenster verlässt. Abbildung 5.2 illustriert den Verlauf des Programmabschnitts:

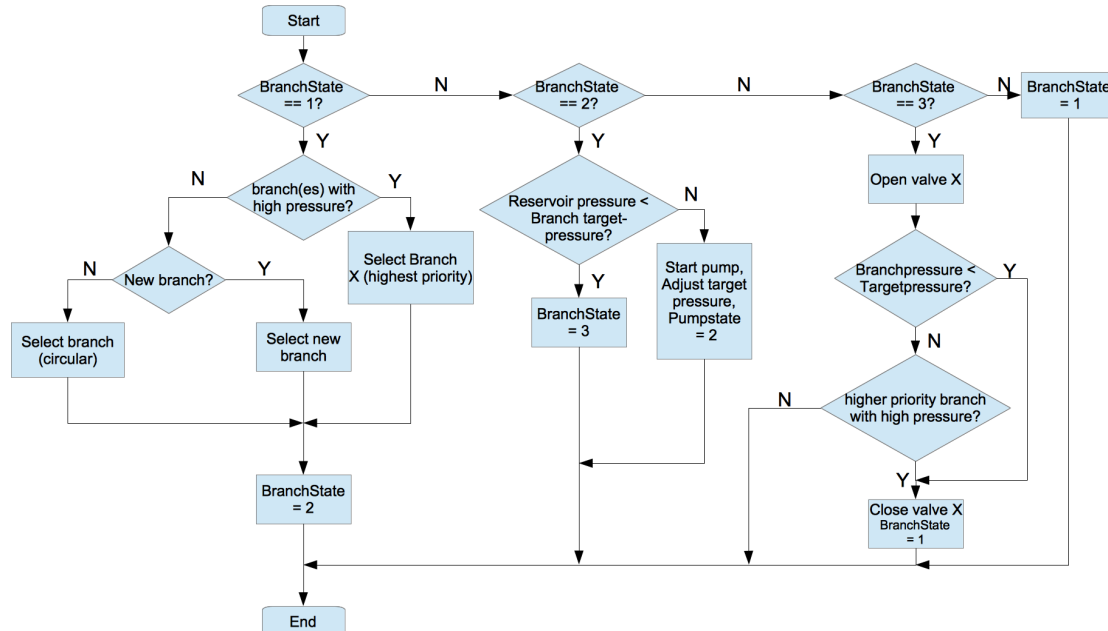


Abbildung 5.2.: Programmablaufplan Raspberry Pi Abzweigsteuerung

5.1.3. Programmablauf Pumpensteuerung

Die Pumpensteuerung liest in jedem Durchlauf die Messwerte der Thermovacs erneut ein und prüft auf Fehler der Sensoren, Ventile und Pumpe. Beim ersten Durchlauf befindet sich die Pumpe im 'Idle' Zustand, dieser wird beibehalten, bis der Druck des Reservoirs das Soll-druckfenster verlässt. Dann geht die Pumpensteuerung in den Zustand 'Prepare' über, darin wird die Pumpe zugeschaltet und diese pumpt den Bereich vor dem ersten Ventil (V26 auf Abbildung 1.1) ab, bis der Druck dieses Bereiches geringer ist, als der Reservoirdruck. Damit kann ein weiterer Anstieg des Reservoirdruckes verhindert werden. Nachdem der Druck angepasst wurde, wird der Zustand 'Pumping' eingenommen, das Ventil (V26) wird geöffnet und das Reservoir wird evakuiert. Solange der Reservoirdruck größer als die untere Grenze des Soll-druckfensters ist, wird dieser Zustand beibehalten. Sobald die Soll-druckgrenze unterschritten wird, wird das Ventil geschlossen, die Pumpe angehalten, (wenn notwendig) ein neuer Zieldruck angesetzt und das System geht wieder in den Ruhezustand 'Idle' über.

Der Verlauf der Pumpensteuerung ist in Abbildung 5.3 grafisch dargestellt:

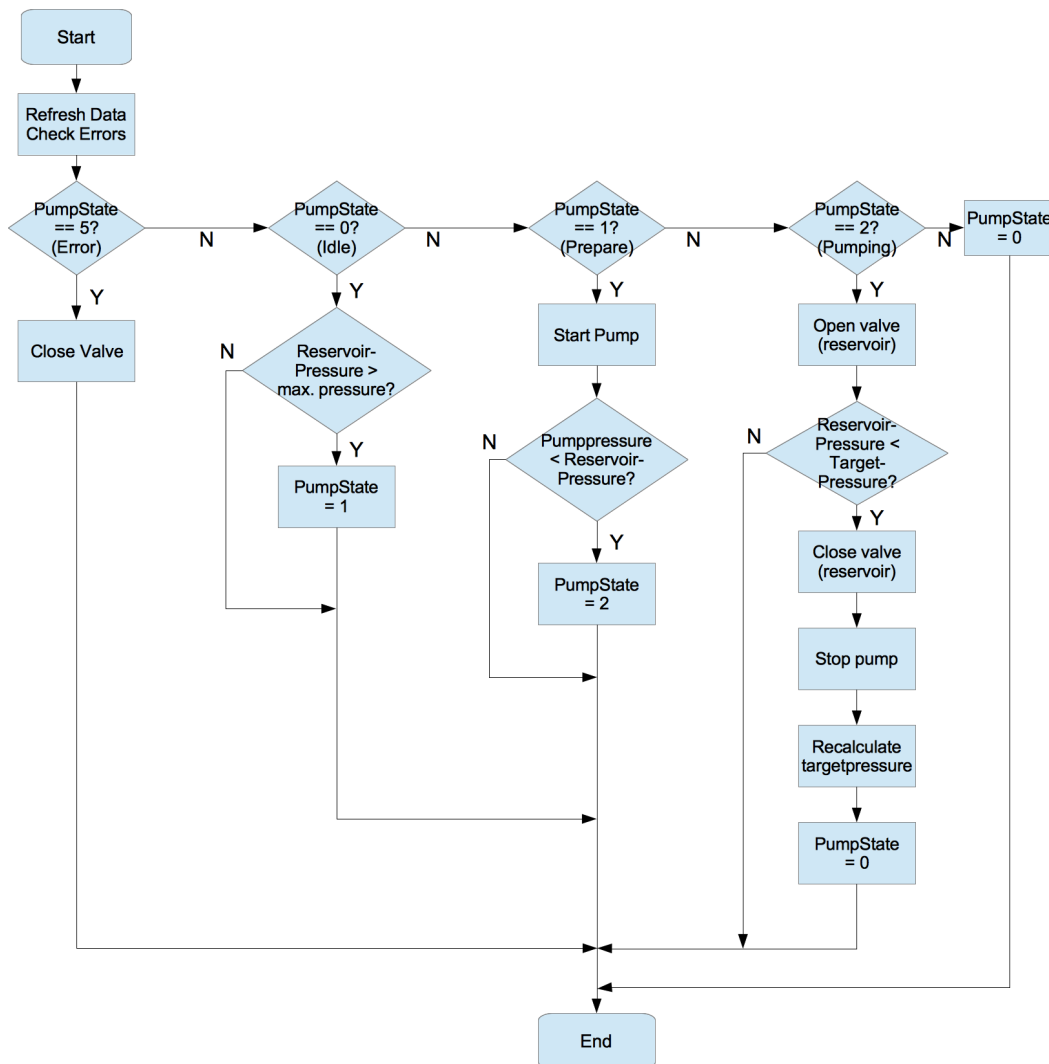


Abbildung 5.3.: Programmablaufplan Raspberry Pi Vorpumpensteuerung

5.1.4. Programmablauf Koprozessor Interface

Das Interface mit dem Koprozessor beinhaltet die Behandlung der seriellen Kommunikation, mit Auswertung der empfangenen Daten sowie die Watchdog Timer Steuerung des Raspberry Pi für den AVR (Raspberry Pi überwacht AVR Controller). Dafür prüft der Raspberry Pi zunächst, ob sich Daten im Puffer des UART Moduls befinden, um eine Blockade des Programmes durch den Empfang beziehungsweise durch das Warten auf Daten zu verhindern. Befinden sich Daten im Puffer werden diese byteweise eingelesen und in einen programminternen Speicher eingeschrieben. Wenn dieser Speicher mit 8 Zeichen gefüllt wurde, beginnt

die Dekodierung der empfangenen Nachricht. Entsprechend der dekodierten Nachricht antwortet das System und/oder ändert seinen Zustand. Dieser Ablauf ist hier visualisiert:

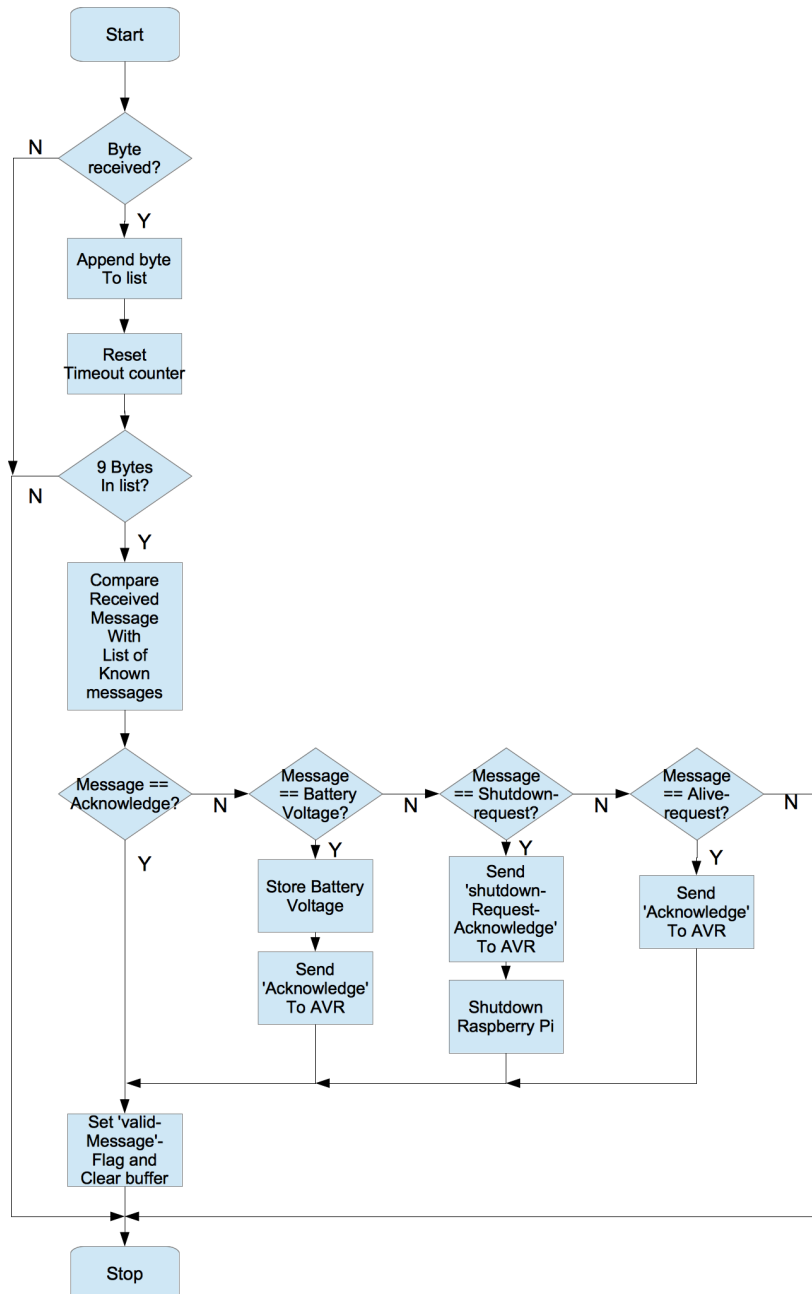


Abbildung 5.4.: Programmablaufplan Raspberry Pi Empfang serielle Daten

Eine Timeout-Funktion wurde hinzugefügt, um eine Störung der Übertragung durch unvoll-

ständige oder fehlerhafte Nachrichten zu verhindern. Dazu wird in einem Abstand von 20 Millisekunden geprüft, ob sich Daten im internen Empfangsspeicher befinden. Wenn Daten vorhanden sind, wird ein Zähler inkrementiert. Beim Empfang von Daten wird dieser Zähler auf Null zurückgesetzt. So wird der Zähler alle 20 Millisekunden erhöht. Wenn unvollständige Daten empfangen wurden und nach ca. 500 Millisekunden werden Zähler und Empfangspuffer zurückgesetzt.

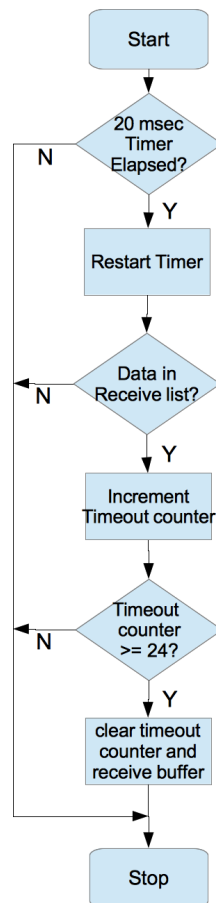


Abbildung 5.5.: Programmablaufplan Raspberry Pi Timeout serielle Daten

Abbildung 5.5 bildet den beschriebenen Programmverlauf des seriellen Timeouts ab.

Weiterhin wurde zur Erhöhung der Sicherheit ein Watchdog Timer (WDT) integriert, dieser überwacht den Betrieb des Koprozessors über die serielle Schnittstelle. Der WDT startet im Zustand 'Running', da der Koprozessor bereits läuft, wenn der Raspberry Pi startet. Dieser Zustand wird beibehalten, solange in regelmäßigen Abständen valide Nachrichten empfangen werden. Bleiben diese Nachrichten aus, sendet der Raspberry Pi einen 'Alive-Request'

über den UART und wechselt in den Zustand 'Prewarn'. Wird diese Nachricht beantwortet, geht das System in den Zustand 'Running' über. Anderenfalls wird der Koprozessor nach einer kurzen Zeit neu gestartet und das System wechselt in den Zustand 'Running'. Der detaillierte Ablauf ist nachfolgend dargestellt:

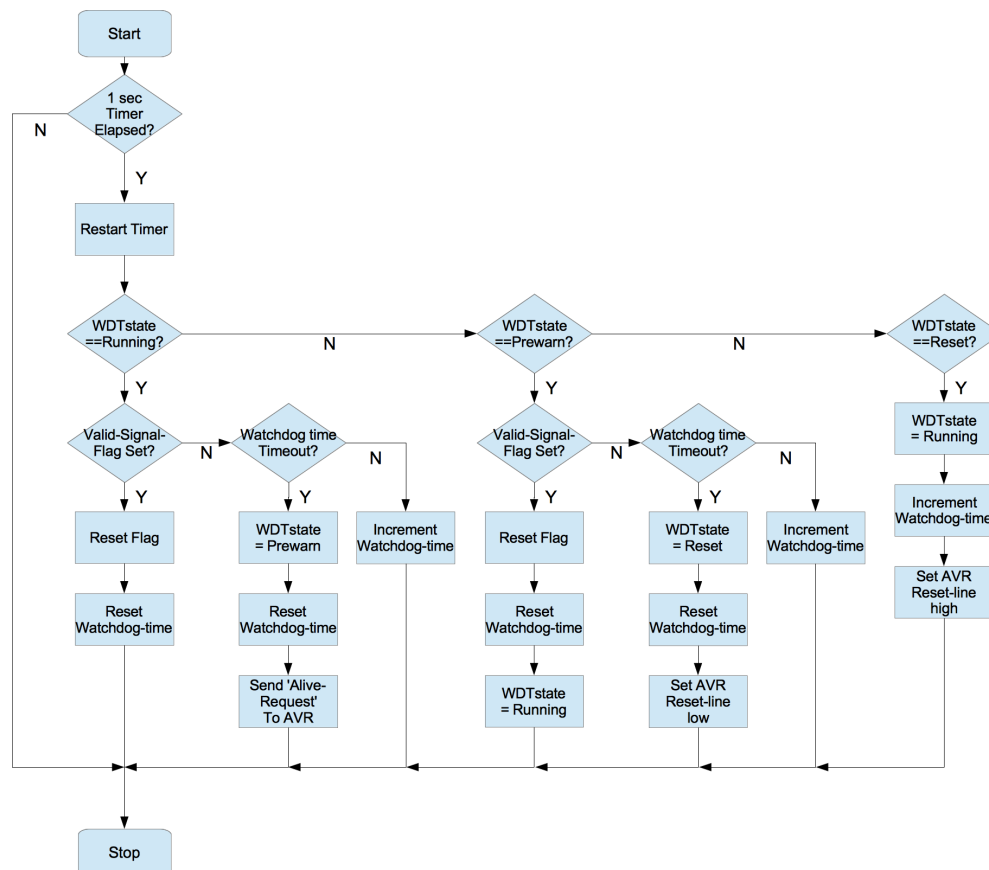


Abbildung 5.6.: Programmablaufplan Raspberry Pi Watchdog Timer für AVR

5.1.5. Programmablauf User Interface

Das User Interface besteht aus Tastern und Feedback LEDs. Um fehlerhafte Eingaben zu verhindern, soll das System erst auf die Betätigung eines Tasters reagieren, wenn dieser für 3 Sekunden gehalten wird. Der Taster wird in einem Abstand von 100 Millisekunden eingelesen. Ist der Taster gedrückt wird ein Zähler dekrementiert. Wenn dieser Zähler Null erreicht wird der zugehörige Abzweig ein bzw. aus geschaltet. Ist der Taster nicht betätigt wird der Zählerwert zurückgesetzt. Weiterhin verfügen nur die Gästeabzweige über Taster,

da nur diese von den Nutzern zu bedienen sind. Der implementierte Programmablauf sieht wie folgt aus:

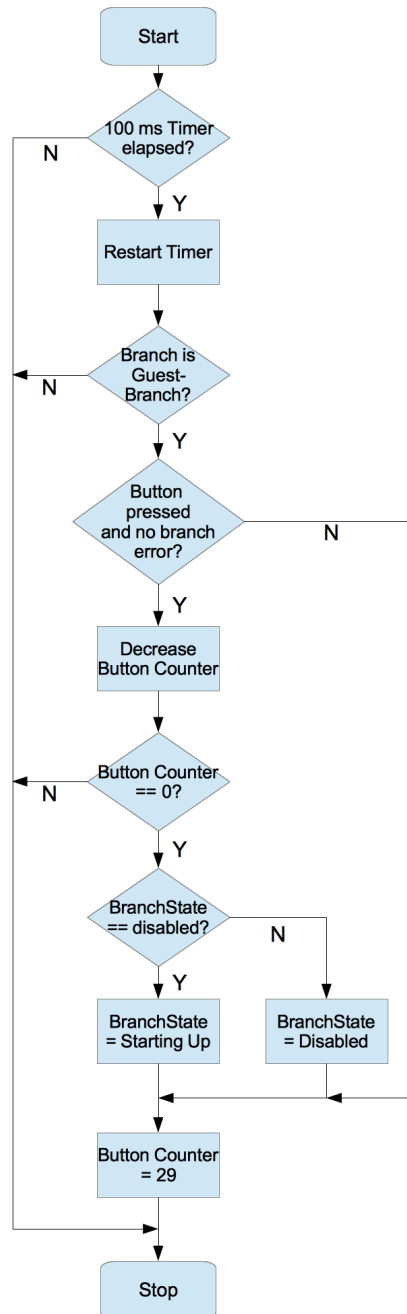


Abbildung 5.7.: Programmablaufplan Raspberry Pi Taster der Nutzerschnittstelle

Auch die Feedback-LEDs sind nur bei Gästeabzweigen implementiert. Die Feedback-LEDs zeigen den Zustand des jeweiligen Abzweiges an, diese sollen wie folgt kodiert werden:

LED aus: Abzweig abgeschaltet

LED konstant an: Abzweig aktiv

LED blinkt (1 Hz): Abzweig startet

LED blinkt (4 Hz): Fehler in Abzweig erkannt

LED blinkt (kurz): Leck in Abzweig erkannt (Erkennung in diesem Release nicht implementiert)

Dafür wird eine Tabelle mit binären (8-Bit) Representationen der Signalmuster der LEDs angelegt, über den Systemzustand wird die aktuelle Position in der Tabelle adressiert. In einem Zeitraster von 100 Millisekunden über einzelne Stellen des adressierten Musters iteriert und die LED entsprechend dem Wert der Position gesetzt. Dieser Vorgang ist nachfolgend als Ablaufplan dargestellt:

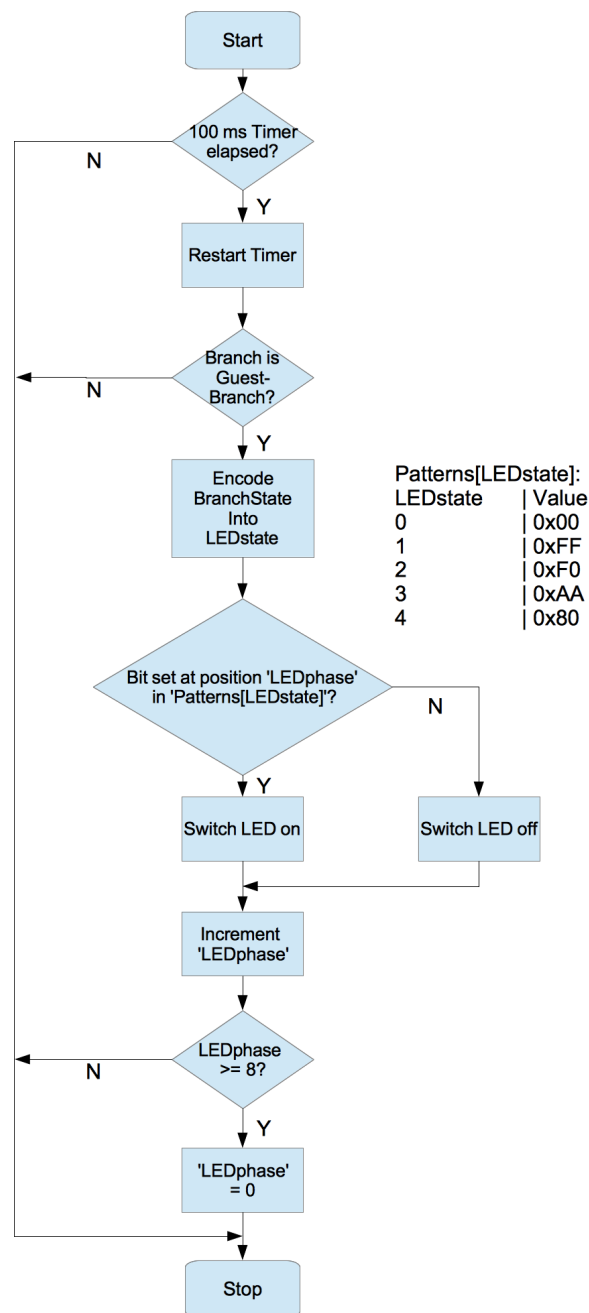


Abbildung 5.8.: Programmablaufplan Raspberry Pi LEDs der Nutzerschnittstelle

5.2. Koprozessor - Code

Dieses Kapitel beschreibt den für den AVR entwickelten C-Code. Der AVR verwaltet als Koprozessor die Stromversorgung durch die USV und überwacht die Betriebsbereitschaft des Raspberry Pi.

5.2.1. Programmablauf Überblick

Der auf dem AVR ausgeführte Code ist in der nachfolgenden Abbildung 5.9 im Überblick dargestellt:

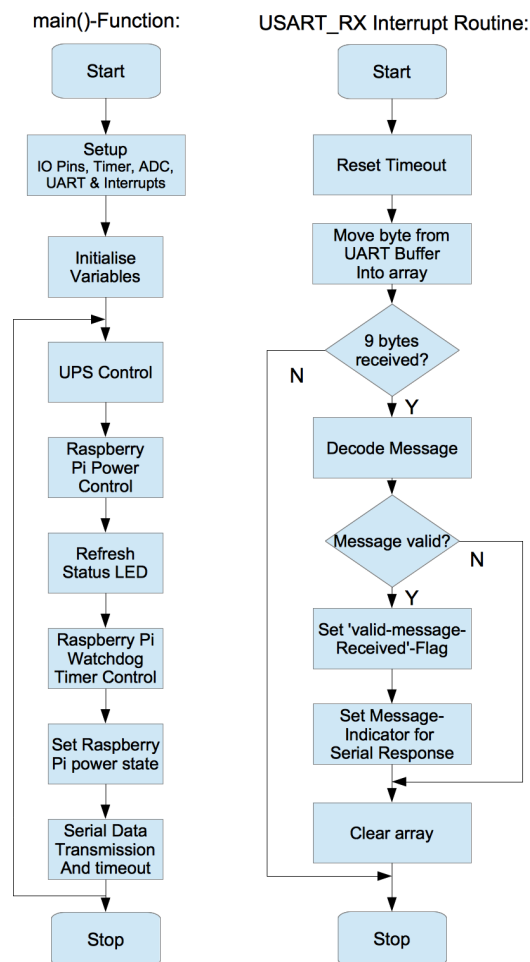


Abbildung 5.9.: Programmablaufplan Koprozessor im Überblick

Auf der linken Seite ist der Ablauf der `main()`-Funktion abgebildet, der Prozessor initialisiert zuerst die Ein- und Ausgänge sowie die internen Module (UART, ADC und Timer). Danach tritt das Programm in eine Schleife ein. In dieser Schleife wird zuerst die USV auf Basis der vom ADC gemessenen Spannungswerte für Batterie und Versorgungsspannung angesteuert. Danach verwaltet der AVR im Programmabschnitt 'Pi Power Control', basierend auf dem Zustand der Stromversorgung, die Stromzufuhr zum Raspberry Pi und schaltet die Status-LED entsprechend dem Systemzustand. Anschließend prüft das Programm den Betriebszustand des Raspberry Pi als Watchdog Timer. Im Abschnitt 'Set Raspberry Pi Power State' steuert der AVR den Feldeffekttransistor in der Stromversorgungsleitung des Raspberry Pi an. Dies erfolgt in einem eigenen Abschnitt, da der Zustand des FETs von den Zuständen der 'Pi Power Control' und des Watchdog Timers abhängt. Abschließend beantwortet der AVR mögliche eingehende serielle Nachrichten.

Rechts ist die implementierte Interrupt-Service-Routine für den seriellen Empfang dargestellt. Diese wird automatisch von dem Mikrocontroller aufgerufen, wenn ein Byte im Empfangspuffer des UART zur Verfügung steht und speichert die Daten in einem C String statt einer Python Liste. Da sich die Interrupt-Service-Routine bis auf die Auslösung (Interrupt statt Polling) und der Speicherung prinzipiell nur wenig von der Funktion auf dem Raspberry Pi unterscheidet, wird auf eine detaillierte Beschreibung verzichtet und auf [Abbildung 5.4](#) und den zugehörigen Abschnitt verwiesen.

5.2.2. Programmablauf USV Steuerung

Damit der Lithium-Akku eine größtmögliche Lebensdauer erreicht, steuert der Koprozessor den Lade- und Entladevorgang des Energiespeichers. Dazu misst der AVR über den eingebauten ADC die Versorgungs- und die Zellspannung und schaltet den Laderegler und den Ausgangs-DCDC-Wandler ein oder aus. Der Akku soll in einem Bereich von 3,4 bis 4 Volt betrieben werden, damit die Zelle nicht voll entladen oder geladen und eine Ladestand bedingte Alterung oder Schädigung der Zelle vermindert. Der Ablauf ist nachfolgend dargestellt:

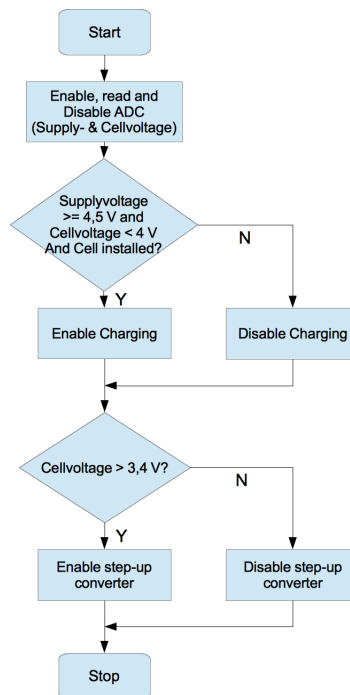


Abbildung 5.10.: Programmablaufplan Koprozessor Steuerung der unterbrechungsfreien Stromversorgung

5.2.3. Programmablauf Pi Power Control

Bei der Pi Power Control handelt es sich um einen Zustandsautomaten. Dieser befasst sich mit der Steuerung des Raspberry Pi bei Ausfall der Netzversorgung. Fällt die Versorgungsspannung aus den Netzteilen unter einen Wert von 4,8 Volt, arbeitet der Raspberry Pi nicht mehr zuverlässig, ab diesem Punkt wechselt das System in den 'Backup' Modus und der Computer wird durch die USV versorgt. Diese Versorgung durch die USV soll maximal 10 Minuten anhalten, damit der Raspberry Pi die letzten Aktionen abschließen und dann sicher herunterfahren kann. Daher wechselt das System nach 10 Minuten in den 'Shutdown' Zustand und sendet über den UART eine shutdown-Anfrage an den Raspberry Pi. Nach weiteren 30 Sekunden wechselt das System zurück in den 'Off' Zustand, der Raspberry Pi ist deaktiviert und der Zustandsautomat wartet auf einen Anstieg der Versorgungsspannung auf einen Wert von über 4,8 V, um den Computer dann erneut zu starten. Der Ablauf ist nachfolgend genauer visualisiert:

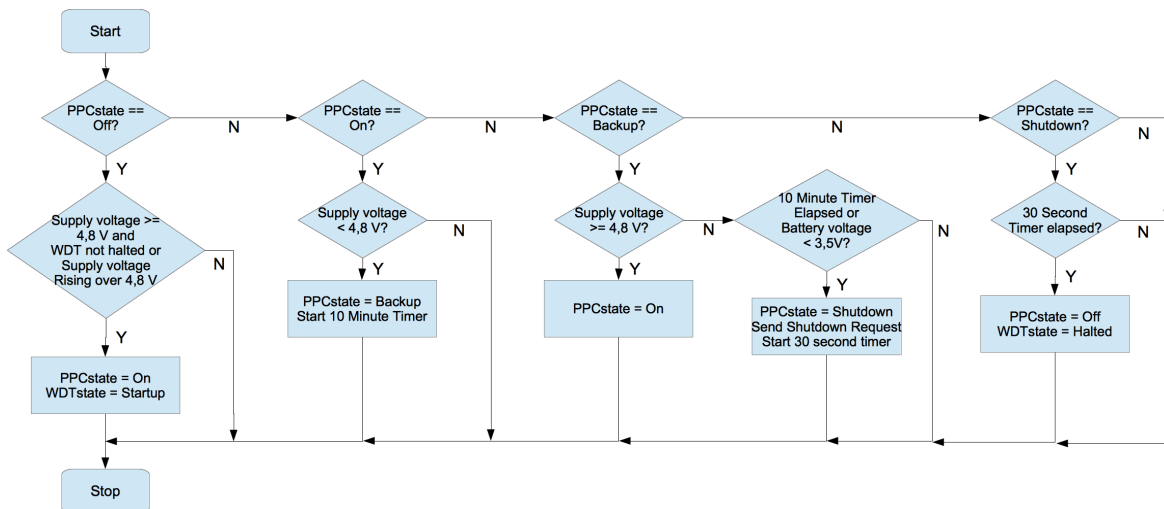


Abbildung 5.11.: Programmablaufplan Koprozessor Steuerung der Stromversorgung des Raspberry Pi

5.2.4. Status-LED

Auf der Platinenoberseite ist eine Status-LED eingebracht, diese ermöglicht die Signalisierung von verschiedenen Betriebszuständen. Der AVR ist für die Ansteuerung dieser LED zuständig und kann folgende Zustände erkennen und ausgeben:

LED aus: System abgeschaltet

LED konstant an: Normalbetrieb

LED blinkt (1 Hz): Kein Akku angeschlossen

LED blinkt (4 Hz): Raspberry Pi reagiert nicht / ist nicht angeschlossen

LED blinkt (kurz): System in Standby-Modus, wartet auf Wiederherstellung der Stromversorgung

Die Behandlung der LED erfolgt ähnlich wie bei der Zustandsanzeige der Abzweige (Abbildung 5.8) durch eine Tabelle mit den Mustern, deren Elemente über den kodierten Zustand adressiert werden. Im 100 Millisekunden Raster wird ein wechselnder Teil des adressierten Elements ausgegeben. Die Erkennung des fehlerhaften Raspberry Pi erfolgt durch die Zustände des Watchdog Timers und die des fehlenden Akkus über die anliegende Spannung.

5.2.5. Programmablauf Watchdog Timer

Damit die Folgen eines Absturzes des Raspberry Pi minimiert werden, ist auf dem Koprozessor ebenfalls ein Watchdog Timer implementiert. Durch Verwendung des AVR statt eines fertigen IC ist es möglich die Überwachung flexibler zu machen und damit die Startzeit sowie die Bedürfnisse des Computers zum schonenden Herunterfahren zu berücksichtigen. Der Ablauf ist ähnlich wie der des auf dem Raspberry Pi implementierten Watchdog Timers (beschrieben in Abschnitt 5.1.4). Der 'Startup'-Zustand wurde hinzugefügt, um die Zeit zwischen Einschalten der Spannungsversorgung und vollständiger Betriebsbereitschaft des Raspberry Pi mit einzubeziehen (etwa 30 Sekunden). Wenn sich der Watchdog Timer in diesem Zustand befindet, wird von dem Raspberry Pi innerhalb von 35 Sekunden eine Meldung über die Betriebsbereitschaft erwartet, bleibt diese aus, wechselt der WDT in den 'Prewarn'-Zustand und sendet eine Aktivitätsanfrage an den Raspberry Pi. Wird diese Anfrage nicht beantwortet wird durch unterbrechen der Spannungsversorgung ein Neustart erzwungen. Nach zwei erfolglosen Startversuchen wird der Raspberry Pi komplett abgeschaltet, der Watchdog Timer angehalten und der Koprozessor geht von einem Fehlen des Raspberry Pi aus. Sollte der Startvorgang erfolgreich sein, wechselt der AVR in den 'Running'-Zustand. In diesem wird im 10 Sekunden Takt eine valide Nachricht erwartet, bei Ausbleiben dieser Nachrichten wird ebenfalls in den 'Prewarn'-Zustand gewechselt und eine Aktivitätsanfrage ausgesendet. Auch hier wird der Raspberry Pi zurückgesetzt, wenn diese Anfrage unbeantwortet bleibt. Der Code-Verlauf ist auch Abbildung 5.12 genauer dargestellt:

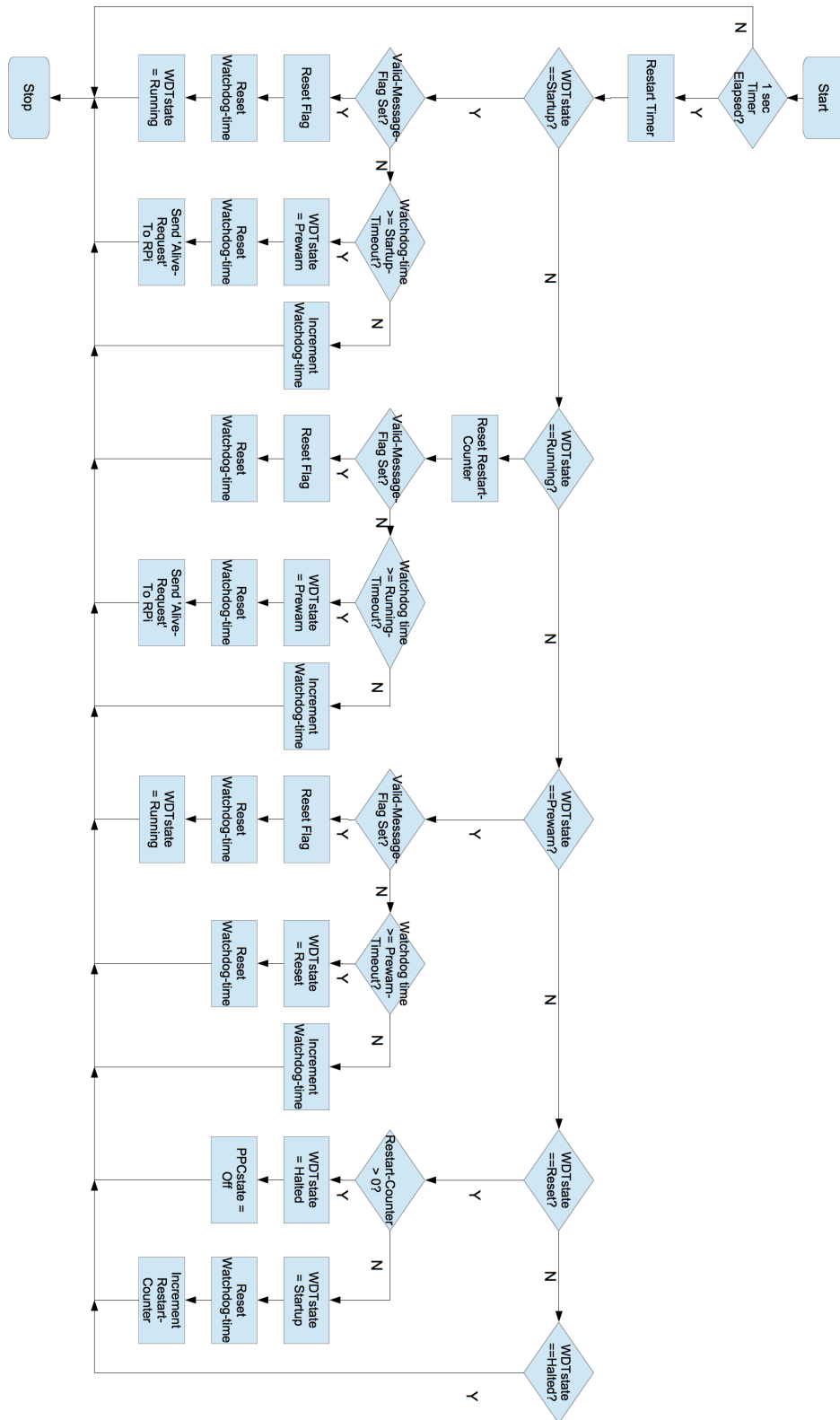


Abbildung 5.12.: Programmablaufplan Koprozessor Watchdog Timer Überwachung des Raspberry Pi

5.2.6. Raspberry Pi Power State

Der Programmabschnitt 'Raspberry Pi Power State' behandelt die Steuerung des Feldeffekttransistors in der Versorgungsleitung des Raspberry Pi (siehe Abbildung 4.15). Die Software führt hier die Zustände des Watchdog Timers und des 'Pi Power Control'-Automaten zusammen, um ein gegenseitiges Überschreiben und unbestimmtes Verhalten des Systems zu verhindern. Für die Zusammenführung der Zustände wurde folgende Tabelle 5.1 angelegt, diese zeigt in welchen Zustandskombinationen der Raspberry Pi eingeschaltet ist:

Tabelle 5.1.: Zustände der Raspberry Pi Stromversorgung nach Systemzuständen von 'Pi Power Control' (PPC) und Watchdog Timer

PPC/WDT	Startup	Running	Prewarn	Reset	Halted
Off	Aus	Aus	Aus	Aus	Aus
On	Ein	Ein	Ein	Aus	Aus
Backup	Ein	Ein	Ein	Aus	Aus
Shutdown	Ein	Ein	Ein	Aus	Aus

Mithilfe der Systemzustände können dann Werte in der Tabelle adressiert und der Einschaltzustand des Raspberry Pi definiert werden.

6. Zusammenfassung und Ausblick

Das entwickelte System bietet (verglichen mit der bisherigen Struktur) die Möglichkeit Kosten einzusparen. Durch die Zentralisierung des Systems können für die Betreiber der Beamline sowie deren Gäste sichere und besser kontrollierbare Vorvakua bereitgestellt werden, ohne die finanzielle und akustische Belastung zu erhöhen. Außerdem bietet das entwickelte System prinzipiell eine höhere Sicherheit, als die bisher eingesetzten einzelnen Pumpen, da diese ohne Steuerung und Absicherung betrieben wurden.

Die Ausfallsicherheit wird durch die Redundanz der Netzteile und Transistoren sowie den Koprozessor erreicht. Die mögliche Überbrückungszeit durch die USV übersteigt die geforderte Dauer von 10 Minuten deutlich. Die in den Anforderungen geforderten Ein- und Ausgänge werden durch das System bereitgestellt. Über den Anschluss des Displays kann im Bedarfsfall auch ein über SPI angesteuertes Daughterboard angeschlossen werden, um die Funktionalität des Systems zu erweitern. Des Weiteren sind viele Anschlüsse des Raspberry Pi ungenutzt, auch dies bietet Optionen für Erweiterungen.

In der weiteren Entwicklung kann das System über die Ethernet Schnittstelle des Raspberry Pi in das 'Tango-Controls'-Steuersystem eingebunden werden, um es damit vollständig in die Umgebung der Beamline zu integrieren. Auch kann das System aufgrund der relativ geringen Abmessungen in ein standard 19"-Gehäuse integriert werden. In diesem kann auch das User Interface (Taster, LEDs und das optionale Display) montiert werden. In zukünftigen Revisionen könnten bei Bedarf noch digitale Schnittstellen für den Betrieb bereitgestellt werden (um beispielsweise Freigabesignale zu versenden oder zu empfangen).

7. Literatur

DESY Historie:

DESY: 50 Jahre DESY (Broschüre), 09/2009,
[pr.desy.de/sites2009/site_pr/content/e113/e143399/
e143401/DESY50_Jubilaeumsbroschuere_D_ger.pdf](http://pr.desy.de/sites2009/site_pr/content/e113/e143399/e143401/DESY50_Jubilaeumsbroschuere_D_ger.pdf) (Zugriff: 02.08.2016)

Grundlagen Druckmessung:

VACOM Vakuum Komponenten & Messtechnik GmbH: Die kleine Fibel der Vakuum Druckmessung, 2011,
www.vacom.de/downloads/sonstige-dokumente?download=2:die-kleine-fibel-der-vakuum-druckmessung (Zugriff: 03.08.2016)

Raspberry Pi 2:

Raspberry Pi Foundation: RASPBERRY PI 2 MODEL B,
<https://www.raspberrypi.org/products/raspberry-pi-2-model-b/> (Zugriff: 05.08.2016)

Atmel AVR Atmega328:

Atmel: ATmega48A/PA/88A/PA/168A/PA/328/P Datasheet, 11/2015,
http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf (Zugriff: 15.08.2016)

Fehlermechanismen von Halbleiterbauteilen:

Panasonic: Failure Mechanism of Semiconductor Devices, T04007BE-3 2009.4,
www.semicon.panasonic.co.jp/en/aboutus/pdf/t04007be-3.pdf (Zugriff: 12.08.2016)

A. Anhang

Der Inhalt des Anhangs ist auf der Anhangs-DVD bei Prof. Dr. Jünemann einzusehen.

Versicherung über die Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit im Sinne der Prüfungsordnung nach §16(5) APSO-TI-BM ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen habe ich unter Angabe der Quellen kenntlich gemacht.

Hamburg, 29. August 2016

Ort, Datum

Unterschrift