



Finalisierung der Inbetriebnahme eines Mess-Systems für die Aufnahme von Parametern an Siebträger Espressomaschinen mit und ohne E61 Brühgruppe

Patricia Viebke

Finalization of commissioning a measurement system for capturing parameters on portafilter espresso machines with and without E61 group head

Projektarbeit II
Master of Applied Research (MAPR)

unter Betreuung von: LbA Armin Rohnen

Abgabe: 20. April 2024

ABSTRACT

Es wird ein Mess-System entwickelt, welches für die Aufzeichnung von Parametern an E61 und nicht E61 Siebträger Espressomaschinen verwendet wird. Das Mess-System verfügt über ausgewählte Sensorik, welche Temperatur, Druck, Durchflussrate und die Kaffeeausflussmenge erfasst. Das Mess-System basiert auf einem Raspberry Pi, welches mit zwei Modulen von Measuring Computing ausgestattet ist. Über die Module werden bis zu sieben Messparameter gleichzeitig erfasst. Für die Inbetriebnahme benötigt es Python Codes, über welche die Konfiguration der Messprozesse stattfindet. Es werden die zwei Messprozesse Aufheizverhalten und Kaffeebezug definiert. Über eine MATLAB GUI werden die Umgebungsparameter wie Brühgruppenart, Messart, Objektname und zu messende Sensoren ausgewählt und anschließend die Messung gestartet. Das Aufheizverhalten muss bei Erreichen eines Temperaturplateaus manuell beendet werden, während die Messung für Kaffeebezug eine definierte Zeit von 60 s hat. Die erzeugten Messwerte werden an MATLAB übergeben und dort findet die Aufbereitung und Umwandlung der Daten statt. Die Messwerte werden in physikalischen Größen angezeigt und sind interpretierbar. Neben E61 Siebträger Espressomaschinen ist die Vermessung von Maschinen mit undefinierter Brühgruppe möglich. Durch eine erweiterte Brühgruppe bietet das Mess-System die Möglichkeit parallel den Druck und die Temperatur vor dem Kaffeepuck zu erfassen. Für die Brühgruppen Erweiterung muss die Halteplatte und der Brühgruppeneinsatz optimiert werden. Zuletzt wird das Mess-System an einer E61 Espressomaschine abgesichert und die ersten Verläufe stehen zur Verfügung. Die kenntnisreiche Inbetriebnahme ist erfolgreich abgeschlossen und die zuverlässige Vermessung von Siebträger Espressomaschinen wird gewährleistet.

A measurement system is being developed for recording parameters on E61 and non-E61 portafilter espresso machines. The measurement system incorporates selected sensors capable of capturing temperature, pressure, flow rate, and coffee output volume. It is based on a Raspberry Pi equipped with two modules from Measuring Computing, allowing for the simultaneous capture of up to seven measurement parameters. Python code is required for the setup process, enabling the configuration of measurement processes. Two measurement processes are defined: heating behavior and coffee extraction. An MATLAB GUI is utilized for setting environmental parameters such as brew group type, measurement type, object name, and sensors to be used, followed by initiating the measurement. The heating behavior measurement needs to be manually terminated upon reaching a temperature plateau, whereas the coffee extraction measurement has a defined duration of 60 seconds. The generated measurement data is transferred to MATLAB, where data processing and conversion take place. The measurement values are displayed in physical units and are interpretable.

In addition to E61 portafilter espresso machines, the measurement system allows for the assessment of machines with undefined group heads. Through an extended group head, the system provides the capability to simultaneously capture pressure and temperature before the coffee puck. The group head extension requires optimization of the holding plate and the group head insert. Finally, the measurement system is validated on an E61 espresso machine, and initial results are available. The knowledgeable setup process is successfully completed, ensuring reliable measurement of portafilter espresso machines.

INHALTSVERZEICHNIS

| | | |
|----------|---|----|
| 1 | Einleitung | 3 |
| 2 | Definition des Messsystems | 3 |
| 2-A | Aufbau des Messsystems | 3 |
| 2-B | Lokale Verbindung mit dem Mess-System | 3 |
| 2-C | Verwendete Sensorik | 4 |
| 3 | Software der Messprozesse | 5 |
| 3-A | Kaffeebezug | 5 |
| 3.A.1 | Skript für MCCA | 5 |
| 3.A.2 | Skript für die Waage | 7 |
| 3-B | Aufheizverhalten | 7 |
| 4 | Kalibrierung der Sensoren | 8 |
| 4-A | Temperatursensoren | 8 |
| 4.A.1 | NTCs | 8 |
| 4.A.2 | Infrarot | 10 |
| 4-B | Drucksensor | 10 |
| 4-C | Waage | 10 |
| 5 | MATLAB GUI | 12 |
| 5-A | Datenverarbeitung der Messprozesse | 13 |
| 5.A.1 | Kaffeebezug | 13 |
| 5.A.2 | Aufheizverhalten | 14 |
| 6 | Adapter für nicht E61 Kaffeemaschinen | 15 |
| 6-A | Auslegung des Halblechs | 15 |
| 6-B | Anpassung des Brühgruppeneinsatzes | 16 |
| 7 | Absicherung des Mess-Systems | 17 |
| 8 | Zusammenfassung und Ausblick | 18 |
| | Literatur | 19 |
| | Anhang | 20 |

1. EINLEITUNG

Im Bereich der Kaffeemaschinen befindet sich eine Siebträger Espressomaschine in Entwicklung. Diese Siebträger Espressomaschine hat die Besonderheit, dass alle auf dem Markt befindlichen Siebträger Maschinen abgebildet werden. Das Ziel der zu entwickelten Maschine ist es, den Kaffee wie an der Original Maschine zu beziehen. Hierzu werden die Parameter benötigt, über welche der Kaffeebezug definiert wird. Die Erfassung dieser Parameter benötigt ein Mess-System, welches die Temperaturen, den Druck, die Durchflussrate und die Kaffeausflussmenge erfasst. Über die Parameter ist der Ablauf des Kaffeebezugs nachvollziehbar. Weiter werden durch die Analyse der Parameter Schlussfolgerungen über den Aufbau unbekannter Brühgruppen gezogen. Die Parameter geben Informationen über den Brühprozess wieder, welche in die Entwicklung der Glasboiler Siebträger Espressomaschine fließen.

Durch die Informationen gelingt es die Elektronik und Programmierung so auszulegen, dass die Brühprozesse identisch ablaufen und zu folglich gleichem Kaffeegeschmack führen.

2. DEFINITION DES MESSSYSTEMS

Das Mess-System hat zum Ziel unterschiedliche Siebträger Espressomaschinen zu vermessen. Nach der Vermessung müssen die Messdaten analysiert und interpretiert werden. Für einen gleichwertigen Vergleich der Vermessungen müssen definierte Umgebungsbedingungen geschaffen werden. Hierfür muss das Mess-System in den Einzelheiten vollständig definiert vorliegen. Der Aufbau des Mess-Systems, die verwendete Elektronik, die Konfiguration der Elektronik und die Sensorik müssen identisch bleiben.

Während eines Vergleichs von Messreihen ist das Ändern des Aufbaus bzw. der Konfiguration des Mess-Systems nicht zu empfehlen, da ein aussagekräftiger Vergleich somit nicht mehr gewährleistet wird.

A. Aufbau des Messsystems

Das Mess-System basiert auf dem Raspberry Pi, welches mit Modulen von Measuring Computing (MCC) erweitert wird. Diese werden übereinander gestapelt und über den 40 Pin GPIO des Raspberry Pis miteinander verbunden. Durch Probemessungen wird herausgefunden, dass eine Elektronikplatine mit Signalkonditionierungen benötigt wird, welche die Rohdaten in Form von Spannungswerten mit Tiefpassfiltern aufbereiten. Neben der Aufbereitung bietet die Platine Federklemmleisten, worüber die Sensoren und die Stromversorgung verbunden werden (Appendix A).

Ursprünglich ist das Mess-System mit drei Modulen (MC-Cs) von Measuring Computing ausgelegt [Vie22a]. Da die Module für die Differential Messung konfiguriert werden, sind maximal 4 Messkanäle pro Modul parallel benutzbar. Pro Modul liegt die maximale Abtastrate bei 100 kS/s . Bei vier Messkanälen bedeutet dies pro Kanal maximal eine Abtastrate von 25 kS/s , da die festgelegte Abtastrate auf allen Kanälen des Moduls gleich ist. Für den Durchflusssensor ist die Annahme getroffen, dass es eine Abtastrate von 40 kS/s benötigt, um das Rechtecksignal fehlerfrei zu erfassen. Im

Laufe der Entwicklung zeigt das Modul für das Flowmeter vermehrt Probleme auf und eine zuverlässige Erfassung der Durchflussrate wird nicht mehr gewährleistet. Aus diesem Grund wird die benötigte Abtastrate weiter untersucht. Hierfür wird eine Messreihe mit unterschiedlichen Abtastraten durchgeführt. Jede Messung hat eine Dauer von 30 s und die Durchflussrate wird durch den Einsatz des Pumpenprüfstands konstant gehalten. Das erfasste Signal muss ausgewertet und verglichen werden. Es wird eine Abtastrate von 30 kS/s angestrebt, da somit die Konfiguration des Messkanals für das Flowmeter auf dem zweiten Modul realisierbar ist. Für

Tabelle I: Statistischer Vergleich der Periodenstreuung

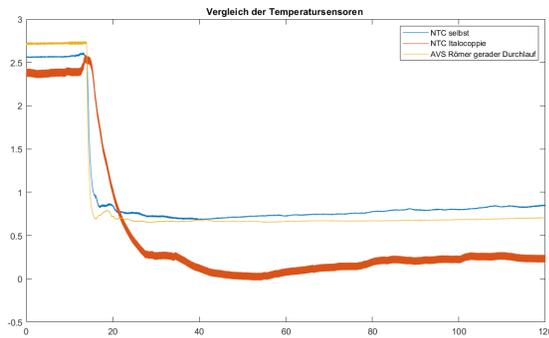
| Abtastrate | 40 kS/s | 33 kS/s | 30 kS/s |
|--|------------------|------------------|------------------|
| Streuung in $\times 10^{-4} \text{ s}$ | 1.5498 | 1.5520 | 1.5916 |

die Auswertungen der Abtastraten werden mit dem MATLAB Befehl `pulseperiod` die Perioden ermittelt und ein Mittelwert gebildet. Im nächsten Schritt werden die Abweichungen berechnet, indem der Mittelwert von den einzelnen Perioden subtrahiert wird. Von den Abweichungen wird anschließend über `fitdist` die Normalverteilung durchgeführt. Die in Tabelle I dargestellte Streuung bezieht sich auf die Abweichungen pro Messung mit einer Abtastrate. [Vie23b] Mit einer ursprünglich definierten Abtastrate von 40 kS/s ist eine korrekte Erfassung der Durchflussrate gegeben und wird somit als Referenzwert verwendet. Bei Verringerung der Abtastrate ist eine leichte Erhöhung der Streuung zu erkennen. Eine graphische Gegenüberstellung des Signals mit 30 kS/s zur Ursprungsrate zeigt, dass die Flanken des Rechtecksignals noch korrekt erfasst werden. Eine weitere Verringerung der Rate ist nicht zu empfehlen, da die Rechtecke des Signals aufgrund verringerter Datenpunkten die Form zu Dreiecken annehmen. Durch den Vergleich und die Analyse der Abtastraten am Rechtecksignal wird bestätigt, dass die Integration des Flowmeters am zweiten Modul mit einer Abtastrate von 30 kS/s realisierbar ist. Für die Weiterentwicklung des Messsystems wird der Durchflusssensor auf dem dritten Kanal des zweiten Moduls verwendet.

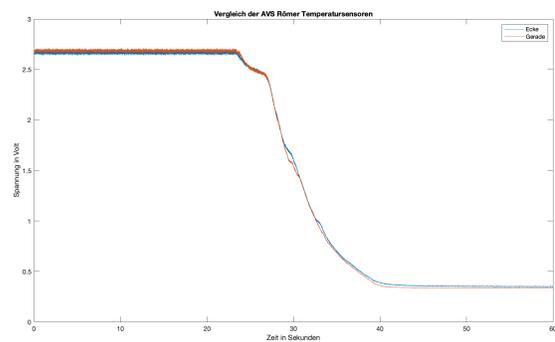
B. Lokale Verbindung mit dem Mess-System

Das Mess-System ist über ein Ethernet Kabel mit dem PC verbunden. Aktuell wird eine SSH Verbindung zum Mess-System über das Terminal und dem Hostnamen hergestellt. Dies funktioniert, da sowohl der PC als auch das Mess-System im gleichen Netzwerk liegen.

Für die Vermessung von Siebträger Espressomaschinen ausserhalb des Labors muss die Herstellung einer Verbindung weiterhin ermöglicht werden. Hierfür bleibt die Ethernet Verbindung zwischen den beiden Komponenten vorhanden. Lediglich der Ethernet Port am PC muss konfiguriert werden. Das Raspberry Pi wird von der dynamischen IP Adresse auf die statische IP Adresse $192.168.10.1$ umgestellt. Damit die Verbindung über den PC hergestellt wird, muss der Ethernet Port, über welchen das Raspberry mit dem PC verbunden ist umkonfiguriert werden. Dieser wird ebenfalls auf eine lokale Umgebung umgestellt mit $192.168.10.2$ mit der Subnetzmaske $255.255.255.0$, wobei bei der IP-Adresse die letzte Zahl



(a) NTC 10 kΩ verschiedener Hersteller



(b) AVS Römer NTC Vergleich

Abbildung 1: Vorgehen für die Auswahl von NTC Sensoren

beliebig gewählt werden kann. So wird der Zugriff auf das Raspberry Pi ohne Router und Netzwerk ermöglicht. Das LAN-Kabel muss jedoch immer am selben Ethernetport des PCs angeschlossen werden.

Wird der Anschluss am PC für Internet verwendet, muss der Port von der statischen IP Adresse auf die dynamische bzw. automatisch beziehende IP-Adresse zurückgestellt werden. Die Einstellungen sind am Windows PC über *Einstellungen > Netzwerk und Internet > Status > Adapteroptionen ändern > Rechtsklick auf Ethernet > Eigenschaften > IPv4* durchzuführen.

C. Verwendete Sensorik

Neben der Anzahl der verwendeten Module, hat sich die Auswahl der Sensoren geändert. Das erste Modul (MCC0) ist für die Temperaturmessung über NTC definiert, während auf dem zweiten Modul (MCC1) die Werte für IR Sensor, Druck und Durchflussrate erfasst werden. An der Nummerierung der Messkanäle ist die Position auf dem Modul zu erkennen. Die erste Zahl hinter der Abkürzung für Channel (CH) steht für das Modul. Die Zuweisung der Adressen der Module ergibt sich durch das Setzen eines Jumpers. Die Jumper adressieren die Module, so dass eine eindeutige Identifizierung dieser gegeben ist. Auf jedem Modul befinden sich vier Messkanäle, angefangen mit dem Kanal CH00 bis CH03. Der Messkanal CH11 befindet sich somit auf dem zweiten Modul und ist der Messkanal 1. Die Auswahl und die Zuordnung von

dem Messkanal und der Elektronikplatine muss berücksichtigt werden, was die Funktion der Kalibrierung beschreibt.

Ausschließlich bei Kanal CH03 und CH11, ist für jeden Messkanal ein Sensor geplant. Diesen Kanälen sind zwei Sensoren zugeordnet. Optimalerweise wird an CH03 die Temperatur vor der Brühgruppe gemessen. Hier ist die Integration des Sensors an der Messstelle von der vorliegenden Siebträger-Espressomaschine abhängig. Um die Möglichkeiten des Einbaus zu erweitern, wird hier ein NTC von AVS als Eckausführung und als Geradausführung zugewiesen. Da beim Vermessen komplizierte Eingriffe in die Maschine vermieden werden müssen, wird die Messstelle in diesem Fall vernachlässigt. Die zugeordneten Sensoren können jedoch an einer anderen relevanten Messstelle eingesetzt werden, welche über die GUI benannt werden muss.

Für die beiden Drucksensoren wird beschlossen den gleichen Messkanal CH11 zu verwenden, da auf diesem Modul mit einer hohen Abtastrate von 30 kS/s gemessen wird. Da bei einer Vermessung entweder eine E61 oder nE61 Siebträger-Espressomaschine vorliegt und für beide Fälle der gleiche Sensor verwendet wird, ist die gemeinsame Nutzung möglich. Durch das Flowmeter ist die Benutzung eines vierten Kanals ausgeschlossen, da sonst die Abtastrate auf 25 kS/s reduziert werden muss.

Aus der Tabelle II wird erkenntlich, dass bei den NTC Sensoren fast ausschließlich die NTCs von AVS Römer verwendet werden. Die Auswahl für die NTCs von AVS Römer resultiert aus einem Vergleich mit zwei NTC Sensoren anderer Herstellung. Die NTCs von AVS werden mit einem eigenkonstruierten NTC und dem NTC von Italcoppie gegenübergestellt.

Die Abbildung 1a zeigt den Spannungsverlauf von drei NTC Sensoren mit einem Innenwiderstand von $10 \text{ k}\Omega$. Zu Beginn der Messung befinden sich die Sensoren bei Raumtemperatur und werden anschließend gleichzeitig in heißes Wasser gelassen. NTC (=negative temperature coefficient) charakterisiert sich durch den Spannungsabfall bei Temperaturerhöhung, was in Abbildung 1a deutlich zu erkennen ist. Anhand von zwei Kriterien werden die drei Sensoren auf Eignung für das Messsystem bewertet. Das erste Kriterium dT/dt ist die errechnete Steigung beim Sprung von Raumtemperatur in heißes Wasser. Der Tabelle III sind bei der Steigung negative Werte zu

Tabelle II: Deklaration Sensoren zu festen Messkanälen

| Kanal | Messstelle | Sensor | fs |
|-------|---------------------|--|---------|
| CH00 | Siebträgerauslauf | NTC AVS (gerade) | 1 kS/s |
| CH01 | vor Kaffeepuck E61 | NTC Eigenfertigung | |
| CH02 | vor Kaffeepuck nE61 | NTC AVS (Eckstück) | |
| CH03 | vor Brühgruppe | NTC AVS (gerade) NTC AVS (Eckstück) | 30 kS/s |
| CH10 | undefiniert | IR-Sensor | |
| CH11 | Druck E61 (vor KP) | SEED Studio | |
| | Druck nE61 (vor KP) | | |
| CH12 | Durchflussrate | Digmesa Nano Brass | |

Sensoren zu den Messkanälen ist relevant für den nachfolgenden Kalibrierprozess. Das Zusammenwirken aus dem Sensor,

Tabelle III: Nachweis für NTC Auswahl

| | Eigenherstellung | Italcoppie | AVS Römer |
|---------------|------------------|------------|-----------|
| dT/dt in V/s | -0.2362 | -0.1437 | -0.9383 |
| T65-Wert in s | 1.470 | 7.433 | 0.459 |

entnehmen. Dies resultiert aus dem umgekehrt proportionalen Verhalten der Temperatur zu den Spannungswerten. Da keine Umrechnungskurve für die Spannungswerte zum physikalischen Wert $^{\circ}\text{C}$ vorliegt, muss das Kriterium der Steigung anhand der Spannungswerte erfolgen und ist somit negativ bei Erhöhung der Temperatur.

Das zweite Kriterium ist der T65-Wert. Hierbei wird die Zeit betrachtet bis 65 % vom Endwert erreicht werden. Anhand dieses Kriteriums ist deutlich zu erkennen, dass der NTC von AVS Römer am schnellsten reagiert. Der Wert der steilsten Steigung bestätigt die Aussage zusätzlich.

Es gibt zwei Ausführungen von den AVS Römer NTCs. Einmal die Geradausführung und als Eckausführung. Beide Sensoren bieten die Möglichkeit über ELSA Anschlüsse in ein System integriert zu werden. Für den Vergleich dieser beider Sensoren werden diese an der Teelanze der labortechnischen Siebträger Maschine angeschlossen. Die Sensoren werden über 6 mm Rohrleitungen nacheinander angeschlossen. Die Messung beginnt bei Raumtemperatur und anschließend wird die Teelanze voll betätigt, so dass ein plötzlicher Spannungsabfall wahrzunehmen ist. Der Spannungsabfall ist durch das ca. 110 $^{\circ}\text{C}$ heiße Wasser zu erklären. Die Unebenheiten während dem Spannungsabfall sind mit dem Sieden zu begründen.

Die graphische Darstellung der Vergleichsmessung in Abbildung 1b zeigt, dass kein Unterschied in der Reaktionszeit festzustellen ist. Mit dem Vorgehen wird die Einsetzbarkeit von AVS Römer Sensoren in Gerad- und Eckausführung bestätigt und bevorzugt.

Für die Drucksensoren wurden ursprünglich Realtivdruck Sensoren von B+B verwendet. Durch Probemessungen wird festgestellt, dass die Sensoren nur bis zu einer Umgebungstemperatur von 80 $^{\circ}\text{C}$ zuverlässig einsatzbereit sind. Bei Überschreiten der Obergrenze schaltet die Elektronik des Sensors ab und es werden keine weiteren Messwerte erfasst. Daher muss ein Sensor verwendet werden, welcher die Brühgruppentemperatur aushält. Es wird ein Drucksensor von SEED Studio verwendet mit einem Messbereich bis 12 bar und Einsatzbereich bis zu 105 $^{\circ}\text{C}$.

3. SOFTWARE DER MESSPROZESSE

Sowohl für die Kalibrierung als auch für den späteren Messablauf wird Software auf dem Raspberry Pi benötigt. Über Python Skripte müssen die Module konfiguriert werden. Hierzu steht von `daqhats` eine Python Bibliothek zur Verfügung. Bei dem Mess-System wird unterschieden zwischen einer Erfassung des Kaffeebezugs und des Aufheizverhaltens. Die Messungen haben einen unterschiedlichen Fokus und müssen entsprechend definiert werden. Es werden zwei Python Skripte programmiert, welche die unterschiedlichen Messprozesse ermöglichen.

A. Kaffeebezug

Für den Kaffeebezug wird das gesamte Mess-System benötigt. Es werden optimalerweise alle Messkanäle und zusätzlich die Waage verwendet, welche die Kaffeeausfluss Menge misst. Die Waage wird im Vergleich zu den anderen Sensoren nicht über die Module erfasst. Hierfür ist auf der Elektronikplatine mit den Signalaufbereitungen ein HX711 eingebaut. Dieser wandelt die Messspannung der Brückenschaltung auf der Waage in 2^{23} Bit Werte um. Um die Ausflussmessung mit den Module zeitlich zu synchronisieren, müssen die beiden Prozesse gleichzeitig anfangen und parallel laufen.

Für den Kaffeebezug wird eine Messdauer von 60 s angesetzt. So ist genügend Zeit den Zustand vor und nach dem Kaffeebezug mit aufzunehmen. Die Messdauer wird dem Skript `coffee_shot.py` mitgegeben und wird an die im Skript enthaltenen Subprozesse weitergegeben. Die Subprozesse starten gleichzeitig jeweils ein Python Skript. Das erste Skript `mccs_shot.py` beinhaltet die Konfiguration und Messprozess für die Erfassung über die Module. Das zweite Skript `hx_shot.py` ist analog für die Waage definiert. Der Messprozess für den Kaffeebezug endet, wenn beide Subprozesse beendet sind.

1) Skript für MCCS

Zu Beginn dieses Prozesses für die Module wird geprüft, ob CSV-Files bereits vorliegen. Falls welche vorliegen, werden diese aus dem Python Pfad gelöscht. Der Schritt ist wichtig, damit jede Messung in einer neuen separaten Datei festgehalten wird. Pro Modul wird ein CSV-File erstellt, für welches geprüft muss, ob es bereits vorliegt. Im nächsten Schritt müssen die Module konfiguriert werden. Hierzu gehört die Anzahl der verwendeten Kanäle, Inputmodus, Messbereich, Optionen, Buffergröße und Abtastrate. Nachdem die

Tabelle IV: Konfiguration der MCC Module

| Einstellung | MCC_0 | MCC_1 |
|-------------|-------------------------|--------|
| Channels | 0,1,2,3 | 0,1,2 |
| Input Mode | AnalogInputMode.DIFF | |
| Input Range | AnalogInputRange.BIP_5V | |
| Optionen | OptionFlags.CONTINUOUS | |
| Buffer size | 10000 | 100000 |
| Scan rates | 1000 | 30000 |

Konfiguration der Module abgeschlossen ist, muss die Anzahl der verwendeten Module angegeben werden. Dieser Schritt ist für die Zuordnung wichtig. Mit dem Befehl `hat_list` werden alle erkannten Module aufgelistet, die eine eindeutige Adresse durch den Jumper aufweisen. In der Variable `hats` werden die Adressen der zwei Module gespeichert. Bevor die Erfassung über die MCCs gestartet wird, müssen die Variablen `input_mode`, `input_range` und `channels` initialisiert werden.

Kurz vor Messbeginn, wird ein Zeitstempel erstellt, welcher für das Berechnen der Dauer benötigt wird. Der Messprozess der Module wird nach der Erstellung des Zeitstempels im Hintergrund gestartet, jedoch noch nicht die Erfassung und

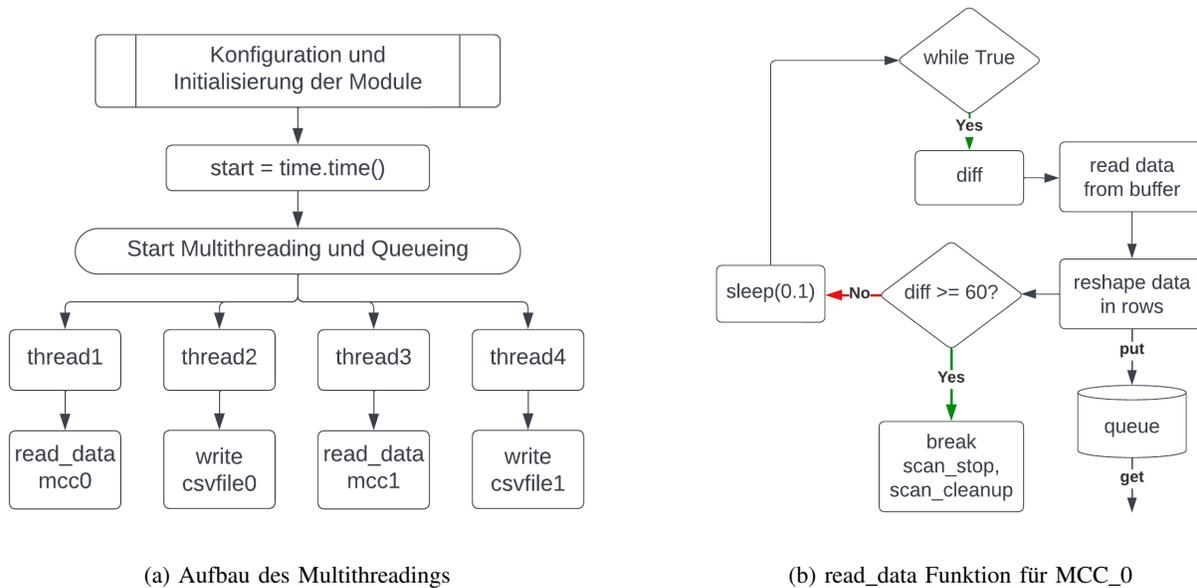


Abbildung 2: Übersicht und Ablauf der Datenerfassung über MCCs

Speicherung der Daten. Die Messdaten werden wie bei einem Multiplexer üblich, der Reihe nach aufgenommen und in einen Buffer abgelegt. Für die Erfassung der Daten müssen diese schrittweise aus dem Buffer geholt werden und in eine Variable gespeichert werden. Über Multithreading [Pyt22] gelingt es dem Programm, die Messdaten aus dem Buffer zu nehmen, einer Variable zuzuweisen und in ein CSV-File zu dokumentieren. In Abbildung 2a sind die Threads des Multithreadings dargestellt. Innerhalb eines Threads muss eine Zielfunktion ausgewählt werden, welche bei Starten des Threads ausgeführt wird. Pro Modul sind zwei Threads definiert. Eins für das Holen der Daten aus dem Buffer und ein Zweites für das Schreiben der Daten in ein CSV-File.

In Abbildung 2b ist der Ablauf der Funktion für die Datenerfassung des MCC0 dargestellt. Die Erfassung wird mittels einer while-Schleife realisiert. Innerhalb der Schleife wird die Variable `diff` erstellt, welche die Messdauer errechnet. Es wird bei jedem Schleifendurchlauf ein Zeitstempel mit `time.time()` erstellt, welcher von dem Startzeitstempel subtrahiert wird. Während die while-Schleife aktiv ist, wird nach Erstellen des Zeitstempels einer Variable `raw` die Messdaten zugeordnet.

```
raw = hat.a_in_scan_read_numpy(-1, 2).data
```

Die Daten werden als Numpy Vektor aus dem Buffer geholt, da dies die Weiterverarbeitung und Aufbereitung erleichtert. Die aus dem Buffer gehaltenen Daten liegen in der gleichen Reihenfolge vor, wie sie gemessen wurden. Der Befehl für das Auslesen hat zwei Inputvariablen, wobei die erste für die Anzahl der Variablen pro Messkanal steht, die aus dem Buffer gelesen werden. Die zweite Zahl ist die Zeit in Sekunden bis ein Timeout erkannt wird. Für die Werte pro Kanal wird der Wert -1 verwendet, da die Anforderung besteht alle aus dem Buffer vorhandenen Messdaten auszulesen. Die Daten liegen in einem NumPy Vektor vor, welcher in einer Zeile

alle Messwerte enthält. Für das Schreiben eines CSV-Files müssen die Daten in die richtige Form gebracht werden. Ziel ist es die Daten nach Kanal zu sortieren. Die angewendete Datenaufbereitung wird mit einem Beispiel von 3 Messwerten pro Kanal veranschaulicht. Das Vorgehen ist für eine beliebige Anzahl an Messwerten pro Kanal anzuwenden.

Die Rohdaten liegen als NumPy Vektor vor mit einer Länge von 12 Messwerten bei 3 Werten pro Kanal. Die Reihenfolge vom Multiplexer spiegelt sich wieder. Es besteht

```
raw = [ch00, ch01, ch02, ch03, ch00, ch01, ch02, ch03,
      ch00, ch01, ch02, ch03]
```

die Möglichkeit den Vektor als ganzes in ein CSV-File zu schreiben, woraus ein erhöhter Aufwand nach Importieren in MATLAB folgt. Die Eigenschaften der NumPy Bibliothek werden genutzt für das Sortieren des Vektors nach Kanälen.

Mit dem Befehl `reshape(-1, 4)` werden die Messdaten in die Form von 4 Spalten bzw. 4 Kanäle gebracht. Das erste Argument steht für die Anzahl der Zeilen und das zweite für die Anzahl der Spalten bei der Umskalierung des Vektors. Für eine bestimmte Anzahl an Messwerten pro Kanal kann die -1 mit der errechneten Anzahl von Zeilen ersetzt werden, in diesem Fall 3. Da für die Zeilen -1 angegeben ist, bleibt die Zeilenanzahl undefiniert.

```
rows = [[ch00, ch01, ch02, ch03],
       [ch00, ch01, ch02, ch03],
       [ch00, ch01, ch02, ch03]]
```

Anhand der inneren Klammern ist in NumPy eine Zeile zu erkennen. Die Messwerte weisen nun die Struktur der Kanäle auf und jede Zeile stellt ein Durchlauf des Multiplexers dar. Die Spannungswerte werden auf einen Wert von 5 V beschränkt. Ist der Wert höher, so wird der Wert 5 angenommen. Die Variable `rows` wird mit den geordneten Messdaten in eine Warteschlange `queue` gesetzt. Die Queue arbeitet nach dem

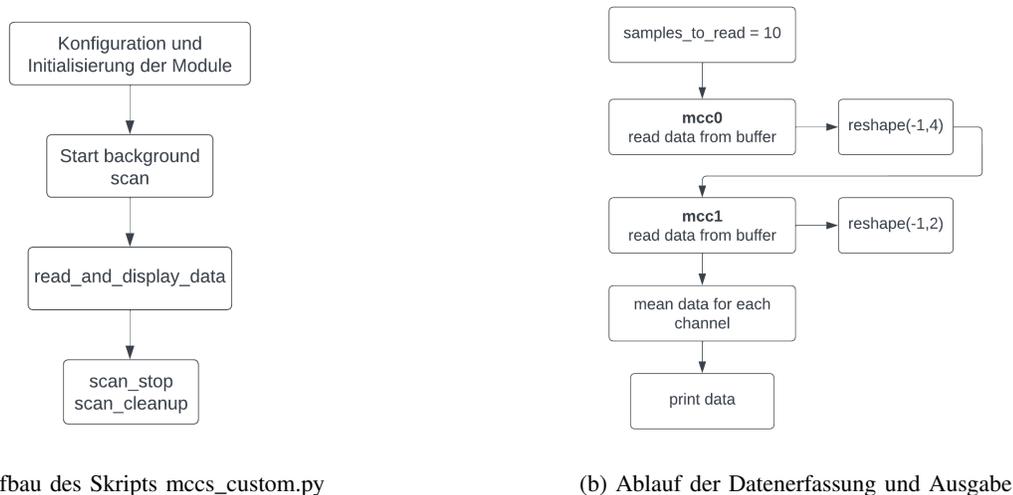


Abbildung 3: Systematische Darstellung für das Skript des Aufheizverhaltens

FIFO (First In First Out) Prinzip [pyt24]. Die Warteschlange bietet sich an, wenn mehr Zeit als verfügbar benötigt wird um Prozesse auszuführen. In diesem Fall bietet sich die Anwendung einer Queue an, da das Schreiben eines CSV-Files der letzte Prozessschritt ist und somit im Hintergrund ausgeführt werden kann ohne die Messgeschwindigkeit zu beeinflussen.

Innerhalb der while-Schleife wird im letzten Schritt ein `sleep` hinzugefügt. Die Verzögerung der Schleifenauführung ist notwendig, da sonst nicht genügend Messwerte im Buffer gesammelt vorliegen. Mit dem Wert von 0.1 s wird der Buffer ausreichend mit Daten gefüllt, so dass bei jedem Schleifendurchlauf die gleiche Anzahl an Messdaten gelesen wird. Bei Erreichen der Messdauer von 60 s wird die Aufnahme der Messdaten des Moduls gestoppt und die Konfiguration wird zurückgesetzt. An die Warteschlange wird ein STOP geschickt, welches als Terminator fungiert.

Parallel zur Datenerfassung aus dem Buffer läuft das Schreiben in ein CSV-File. Hier werden die Daten zeilenweise aus der Warteschlange geholt und mit `writer.writerows` aus der CSV Bibliothek dokumentiert. Der Prozess findet so lange statt, bis der Terminator erkannt wird. Bei einer 60 s entsteht eine zusätzliche Zeit von 7 Sekunden , bis alle Messdaten aus der Warteschlange geholt wurden und in das CSV-File geschrieben werden.

Für das zweite Modul MCC1 läuft der Prozess analog ab. Hier werden die Daten in drei Spalten geordnet, da nur 3 Messkanäle belegt sind. Eine weitere Abweichung stellt die Verzögerung des Schleifendurchlaufs dar. Da hier mit einer Abtastrate von 30 kS/s gemessen wird, muss die Verzögerung geringer gehalten werden, um den Messdatenverlust zu minimieren. Hier wird der Wert von 0.1 s auf 0.01 s gesetzt.

Die Messung über die MCCs ist vollständig abgeschlossen, wenn die Threads fertig durchlaufen sind. Mit dem Befehl `thread.join()` wird gewartet, bis alle Threads fertig sind.

2) Skript für die Waage

Die Messdaten der Waage werden nicht über die MCCs erfasst. Hierfür wird auf der Elektronikplatine die Spannungswerte der Brückenschaltung in 2^{23} Bit Werte umgewandelt. Für die Erfassung wird der 40-Pin GPIO des Raspberry Pi verwendet. Hier müssen vier Pins definiert werden für CLK, DAT, VCC und GND [Vie22b].

Über die kleine Platine mit dem HX711 ist eine Abtastrate von 10 Hz festgelegt. Durch Blockieren einer Leitung auf der Platine kann eine Abtastrate bis 80 Hz erreicht werden. Die Realität zeigt eher eine Rate von 70 Hz . Die Abtastrate hat bei wiederholter Messung nicht den gleichen Wert, dadurch ist die Anzahl der erfassten Messwerte ebenfalls nicht konstant. Durch Erzeugung eines Zeitstempels bei jedem Schleifendurchlauf wird die Variabilität der Rate berücksichtigt. Das Skript zeigt im Aufbau eine vereinfachte Analogie zu dem Skript für die MCCs.

Im ersten Schritt wird geprüft, ob das CSV-File `csvhx711.csv` bereits existiert. Falls ja, wird dies aus dem Pfad entfernt. Anschließend wird ein `hx` Objekt erstellt über die Bibliothek von HX711.

Mit `hx = HX711(dout_pin=5, pd_sck_pin=6)` werden die benötigten Pins auf dem Raspberry Pi definiert. Es wird ein Zeitstempel vom Startzeitpunkt erstellt. Es folgt eine while-Schleife, welche die aktuelle Messdauer errechnet aus dem aktuellen Zeitstempel und der Startzeit. Die Daten werden erfasst mit `hx.get_raw_data_mean(1)`. Der Befehl ist von der HX711 Bibliothek, welcher die Rohdaten erfasst und gemittelt ausgibt. Die Zahl gibt die Anzahl an zu mittelnden Werten an. In diesem Fall werden die Rohdaten ohne Mittelwert ausgegeben.

B. Aufheizverhalten

Das Aufheizverhalten wird über einen zeitlich nicht definierbaren Zeitraum aufgezeichnet. Dieser Zeitraum ist abhängig von dem Verhalten der Siebträger Espressomaschine. Da die Zeit bis zum vollständigen Aufheizen unbekannt ist, muss die Messung manuell gestartet und gestoppt werden. Für die

Mitverfolgbarkeit müssen die Messwerte in Echtzeit für den Benutzer zur Verfügung stehen. Dies wird erreicht indem das Python Skript in einer bestimmten Frequenz ausgeführt wird. In jedem Durchlauf des Python Skripts werden die Messwerte an MATLAB weitergegeben und dort aufbereitet. Die aufbereiteten Messwerte werden dort in Diagrammen grafisch dargestellt. Die graphische Darstellung ermöglicht das Erkennen der Stagnation während des Aufheizprozesses. Ist eine Station an den Temperaturwerten zu erkennen, so ist die Maschine vollständig aufgeheizt und der Messprozess wird beendet.

Abbildung 3a zeigt den allgemeinen Ablauf des Python Skriptes. Wie bei der SHOT Messung, müssen die Module ebenfalls zu Beginn konfiguriert und initialisiert werden. Hier werden alle Messkanäle konfiguriert ausschließlich dem CH02, welcher für das Flowmeter ist. Während des Aufheizprozesses wird die Überwachung des Durchflusses nicht benötigt, da das Wasser zu dem Zeitpunkt nicht fließt. Nach Messbeginn werden mit der Funktion `read_and_display_data` 10 Messwerte pro Kanal als NumPy Vektor erfasst. Die Rohdaten werden in zwei separaten Variablen gespeichert. Die Variable

```
data0 = [ch001, ch011, ch021, ch031, ...
         ch0010, ch0110, ch0210, ch0310]

data1 = [ch101, ch111, ..., ch1010, ch1110]
```

`data0` nimmt die Messwerte des Moduls MCC_0 auf und hat somit 40 Messwerte gespeichert, während die `data1` die Spannungswerte von dem MCC_1 beinhaltet. Die Messwerte müssen für den Schritt der Mittelwertbildung nach Kanal sortiert werden. Hierfür werden die Vektoren umgeformt. Die Messwerte des ersten Moduls werden mit `reshape(10, 4)` umgeformt. Bei dem zweiten Modul wird die 4 durch eine 2 ersetzt. Die Spalten der beiden Vektoren werden jeweils gemittelt, so dass die beiden Vektoren nur noch als eine Zeile vorliegen mit 4 und 2 Spalten. Im letzten Schritt werden die Vektoren mit `np.hstack` aneinander gefügt. Wird der fertige Vektor direkt ausgegeben, so werden die eckigen Klammern mitgegeben. Um die Datenaufbereitung für MATLAB zu erleichtern, werden diese mit dem Befehl `strip` entfernt.

Zuletzt wird die Datenerfassung der Module gestoppt und die Konfiguration wird zurückgesetzt. Mit einer Optimierung wird versucht die Konfiguration einmalig zu Beginn der Messung durchzuführen. Hier wird die Datenerfassung der Module nur gestoppt, jedoch nicht zurückgesetzt. Dies ist nicht umsetzbar, da das Skript bei jedem Durchlauf neu aufgerufen werden muss. Die Variablen sind somit nicht mehr definiert und die Konfiguration wird nicht übernommen. Mit den aktuellen Einstellungen auf den Modulen muss bei jedem Aufruf des Python Skriptes die Konfiguration und Initialisierung durchgeführt werden. Jedoch wird dadurch keine Zeitverzögerung festgestellt und ist somit unproblematisch.

4. KALIBRIERUNG DER SENSOREN

Die Sensoren müssen aufgrund der Zuweisung immer am selben Messkanal angeschlossen werden, um die Reproduzierbarkeit der Messwerte zu gewährleisten. Die Messkette pro Kanal muss kalibriert werden. Ziel der Kalibrierung ist es,

die vom Sensor erfassten Spannungswerte in physikalische Werte umzuwandeln. Aus dem Kalibrierprozess wird eine Kalibrierkurve bzw. Kennlinie ermittelt, welche die Umwandlung ermöglicht. Die Kalibrierkurve muss für jede Sensor-Kanal Kombination erstellt werden. Dies gilt auch bei gleichen Sensoren an einem anderen Messkanal bzw. bei zwei Sensoren am gleichen Messkanal. Durch das Zusammenspiel der Elektronik, des Sensors und des Kabels entstehen Messabweichungen, welche durch die Kalibrierung berücksichtigt werden. Es müssen alle Sensoren ausschließlich dem Flowmeter einer Kalibrierung unterzogen werden. Die Erstellung der Kalibrierkurven wird am Pumpenprüfstand angestrebt. Für die Waage wird ein manueller Kalibrierprozess mit Eichgewichten durchgeführt.

A. Temperatursensoren

Im Mess-System werden zwei Arten von Temperatursensoren verwendet. Neben den NTC Sensoren wird auch ein Infrarot Temperatursensor eingesetzt. Die Vorgehensweise für die beiden Sensorarten ist identisch. Lediglich der Messbereich und die Form der Kennlinie unterscheiden sich.

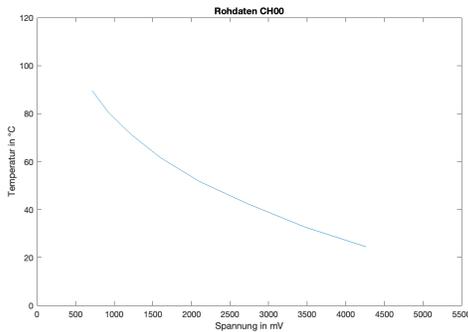
1) NTCs

Das Mess-System beinhaltet 4 NTC Sensoren. Drei der vier Sensoren sind von AVS Römer und bieten die Möglichkeit über ELSA Verbindungen mit einem 6 mm Rohr in Systeme integriert zu werden. Der NTC für die Erfassung der Temperatur an der E61 Brühgruppe ist eine Eigenkonstruktion und muss mittels eines Adapters in den Prüfstand integriert werden. Dieser hat ein M6 Gewinde und wird mit Dichtmasse in einen Adapter geschraubt. An dem Adapter werden über zwei G 1/4" Gewinde die ELSA Anschlüsse verschraubt. An das dritte G 1/4" Gewinde wird der Referenzsensor des Prüfstands mit Dichtmasse verschraubt. Mit einem Rohrstück wird der Adapter und ein weiterer NTC (4a) verbunden. So wird in einem Kalibriervorgang die Kurve für zwei Sensoren parallel aufgezeichnet. Aufgrund von Platzmangel werden keine weiteren Sensoren für diesen Durchlauf verwendet.

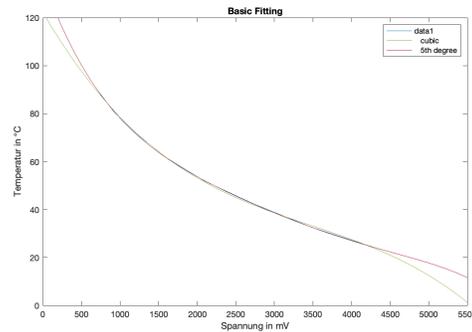
Die zu kalibrierenden Sensoren müssen korrekt mit der Elektronikplatine verbunden werden. Bei der Federklemme der NTCs wurden bei der Inbetriebnahme Probleme in Form von Signalstreuungen festgestellt. Daher wird der zweite und dritte Pin eines NTC Anschlusses miteinander verlötet. Der dritte Pin wird für einen Zweileiter NTC ignoriert und ist im Schaltplan mit 0 gekennzeichnet.

Die Anschlüsse für den Kalibrierprozess von CH00 und CH03 4a müssen dem Schaltplan entnommen werden. Nachdem die Sensoren am Mess-System angeschlossen sind, muss eine Verbindung zwischen MATLAB und dem Mess-System hergestellt werden.

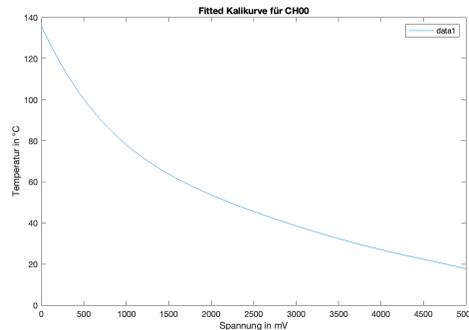
Für den Kalibriervorgang muss die MATLAB GUI für den Pumpenprüfstand geöffnet werden. Anschließend muss die Verbindung zur NI-Messkarte und zum Raspberry Pico hergestellt werden. Bei grünem Füllstand wird der Kalibrierprozess direkt gestartet. Hier wird eine Endtemperatur von 95 °C ausgewählt und 7 Kontrollpunkte werden errechnet. Bei Starten des Prozesses wird die Starttemperatur ebenfalls



(a) Rohdaten Kalibrierung



(b) Auswahl des Polynoms beim Fitting



(c) Angepasste Kalibrierkurve

Abbildung 4: Systematische Darstellung für das Skript des Aufheizverhaltens

als Kontrollpunkt definiert. Befindet sich die Temperatur in der definierten Toleranz des Kontrollpunktes, wird ein Counter gestartet. Erreicht der Counter den Wert 10 erreicht, wird in blauer Schrift die Stabilität des Messpunktes kommuniziert. Mit einem MATLAB Skript wird eine 30 Sekunden Messung über das Raspberry Pi ausgeführt. Das Skript `import_shot.mat` ist unter den Zusatzskripten zu finden. Die Messwerte des Raspberry Pis werden nach 30 Sekunden in MATLAB importiert und in einem Struct gespeichert. Am Ende des Skriptes wird die Zählvariable `xi` um 1 hochgezählt, so dass neue Messwerte in eine neue Zeile des Strukt geschrieben werden.

Nachdem die Werte eines Kontrollpunktes erfolgreich in dem Strukt gespeichert wurden, wird in der GUI zum nächsten Kontrollpunkt geregelt. Der Prozess muss wiederholt werden, bis der letzte Kontrollpunkt erreicht wird. Bei Abschluss der letzten Messung müssen die Messwerte auf der GUI über den entsprechenden Button gespeichert werden. Es wird eine Datei erstellt in dem aktiven MATLAB Ordner mit den Referenztemperaturen. Die Referenztemperaturen werden in einem Vektor gespeichert. Hier müssen die Messwerte den Kontrollpunkten zugeordnet werden und Zeilenweise abgespeichert werden. Die Werte sind eindeutig den Kontrollpunkten zuzuordnen anhand der vorhandenen Temperatursprünge in den Werten. Bei einer 30 Sekunden Messung sind etwa 11 Messpunkte pro Kontrollwert zu erwarten.

Für das Erstellen der Kalibrierkurve werden die Messdaten in einem weiteren Struct gegenübergestellt. Die erste Spalte

beinhaltet die Spannungswerte pro Kontrollwert, während die zweite Spalte die Referenzwerte in $^{\circ}\text{C}$ enthält. Es bildet sich ein Struct mit zwei Spalten und 8 Zeilen, resultierend aus den 8 Kontrollwerten. Die Erstellung der Kalibrierkurve erfordert das Gegenüberstellen einzelner Messwerte. Hierfür müssen die Werte einer Messung gemittelt werden. Es resultieren 8 Spannungswerte, welche 8 Temperaturwerten gegenübergestellt werden. Die Gegenüberstellung der Rohdaten ist in Abbildung 4a graphisch dargestellt. Die Kurve zeigt das typische NTC Verhalten. Das Verhalten charakterisiert sich durch die abnehmende Spannung bei zunehmender Temperatur. Es ist zu erkennen, dass die Kurve nicht den gesamten elektrischen Messbereich von 0 V bis 5 V abdeckt. Dies folgt aus dem eingeschränkten Temperaturbereich, der bei der Kalibrierung anfängt bzw. erreicht wird. Für die Umwandlung der Spannungswerte über den gesamten Messbereich muss die Kurve erweitert werden. Die Erweiterung wird durch das Basic Fitting durchgeführt. In Abbildung 4b werden unterschiedliche Polynome verglichen. Der zweiten Grafik ist der Vergleich eines Polynoms 3. Grades und 5. Grades zu entnehmen. Optimalerweise sind die 0°C bei 5000 mV zu erreichen. Dies ist mit dem Basic Fitting nicht möglich, daher wird das nächst Ähnlichere verwendet. Für die Erstellung der Kalibrierkurve wird das Polynom 3. Grades (grün) ausgewählt. Das Basic Fitting definiert die interpolierte Kurve mit einer Funktion. Ziel ist es die Umwandlung anhand der Vektorindizes aufzubauen. Über einen Messbereich von 0 bis 5000 mV wird eine Auflösung von $0,1\text{ mV}$ angestrebt. Es wird ein Vektor

erstellt, welcher die Werte 0 bis 50000 enthält. Jedes Vektorelement wird in die Funktion des Polynoms aus dem Basic Fitting eingesetzt. Ein Vektor y enthält die Werte der Kalibrierkurve über den gesamten Messbereich mit der gewünschten Auflösung. Dieser Vorgang wird für alle NTC Sensoren wiederholt. Die finalen Kalibrierkurven, die unter y gespeichert sind, werden der Datei `C:/0_Messsystem/Kali/kalistruct.mat` abgelegt. Das Struct ist in der Reihenfolge der Messkanäle aufgebaut. Das erste Element `kali(1).y` ist somit dem NTC für Siebträgerauslauf zugeordnet. Das Struct beinhaltet eine Spalte `line`, welche die `cfit` Elemente speichert. Das Verwenden der `cfit` Elemente zeigt einen vereinfachten Umgang mit den Kalibrierkurven.

2) Infrarot

Die Vorgehensweise für die Kalibrierung des Infrarot Sensors ist analog zum NTC Sensor durchzuführen. Für die Erfassung von Spannungswerten während dem Kalibrierprozess am Pumpenprüfstand muss eine geeignete Messstelle festgelegt werden. Auf dem Messstrang für die Temperaturen befindet sich ein Adapter, in welchem unter anderem der Referenzsensor verschraubt wird. Auf den Adapter wird mit Isolierband eine spiegelfreie Fläche geschaffen, welche als Messstelle für den IR-Sensor dient.

Analog zum NTC Sensor, wird pro Kontrollpunkt eine 30 Sekunden Messung gestartet. Die Messdaten werden gemittelt, so dass eine Gegenüberstellung möglich ist. Der IR-Sensor ist ein digitaler Sensor und verläuft linear über den Messbereich. Da der Sensor auf dem zweiten Modul definiert ist, erstreckt sich der Messbereich von 0 bis 10 V. Der physikalische Messbereich liegt bei $-50\text{ }^{\circ}\text{C}$ bis $600\text{ }^{\circ}\text{C}$. Die Kalibrierkurve wird über diesen Bereich analog zum NTC erstellt und abgespeichert. Die Umwandlung der Spannungswerte in Temperaturwerte findet hier ebenfalls über den Index des Vektors statt. Der Messbereich 0 mV bis 10000 mV wird mit einer Auflösung von 0,1 mV umgesetzt. Daraus errechnet sich die Anzahl von 100000 Werten im Vektor.

B. Drucksensor

Für die Kalibrierung der Drucksensoren wird in beiden Fällen der Sensor an den Kanal CH11 angeschlossen. Bevor die Kalibrierung gestartet wird, wird der Prüfstand bis 12 bar schrittweise hochgeregelt, um die Stabilität und Dichtheit zu testen. Es fällt auf, dass der Druck ab 6 bar nicht weiter zunimmt. Durch Analyse des Problems wird die Pumpe als Ursache erkannt. Die Kalibrierung am Pumpenprüfstand ist somit nicht möglich. Als Notfall Lösung steht die manuelle Kalibrierung mittels der Handpumpe zur Verfügung.

Ein Alublock bietet die Anschlussmöglichkeiten für Sensoren durch Gewindebohrungen. Für die Kalibrierung wird ein EL-SA Anschluss an den Alublock druckdicht verschraubt. Über ein 6 mm Schlauch wird der zu kalibrierende Drucksensor und der Referenzdrucksensor aus dem Pumpenprüfstand verbunden. Bevor die manuelle Kalibrierung startet, muss die Stabilität eines Messpunktes sichergestellt werden. Hierfür wird auf 15 bar aufgepumpt und der Druck über einen längeren

Zeitraum beobachtet. Für die Eignung muss der Druck für etwa eine Minute stabil bleiben. Der Druck ist über einige Minuten stabil, was die Dichtheit des Kalibrieraufbaus bestätigt.

Für den Kalibrierprozess wird zuerst der E61 Sensor verbaut. Es werden 7 Messpunkte im Bereich von 0 bis 15 bar ausgewählt. Bei jedem Messpunkt wird nacheinander eine 30 Sekunden Messung über die NI-Messkarte und über die MCCs gestartet. Nach Abschluss der Messungen von einem Messpunkt wird die Luft aus der Pumpe entlassen und auf einen neuen Druck wird aufgepumpt. Das Vorgehen wird so lange wiederholt bis der Messbereich von Kontrollpunkten abgedeckt ist. Die Messwerte werden nacheinander in einem Struct gespeichert. Es entsteht eine Struktur von 2 Spalten mit 7 Zeilen.

Anschließend müssen die Werte aufbereitet werden. Durch die Erkenntnis eines stabilen Messpunktes, werden die Messwerte pro Kontrollpunkt gemittelt. Durch die Gegenüberstellung wird eine Kalibrierlinie ermittelt, welche einen linearen Verlauf aufweist. Die Kalibrierkurve wird gefittet und ein `cfit` Objekt wird erstellt. Der Vorteil hierbei ist, dass Spannungswerte interpoliert werden und präzise ausgegeben werden. Der Nachteil ist, dass es keine Beschränkungen nach oben und unten hat. Da der Spannungswert des SEED Studio Sensor von 4.6 Volt nicht überschritten wird, folglich kein höherer Druck als 12 bar angezeigt.

Die gleiche Vorgehensweise wird für den Drucksensor für nE61 durchgeführt.

C. Waage

Die Kalibrierung der Waage erfolgt mit Eichgewichten bis 800 g. Das Auftreten eines Messwertdrifts wurde bereits ausgeschlossen [Vie22b] und die Kalibrierung kann direkt durchgeführt werden. Die Waage wird mit der Elektronikplatine laut Schaltplan (Appendix A.1) verbunden. Über MATLAB wird eine 10 Sekunden Messung gestartet, über welche die Bitwerte der Waage erfasst werden. Bei einem ersten Kalibrierversuch wird das Python Skript `hx_shot.py` verwendet. Hierzu wird zu Beginn jeder Messung mit dem Befehl `hx.zero()` die Waage genullt. Die Bit Werte streuen um den Wert Null. Nach der Erfassung der Bit Werte ohne Zusatzgewicht, werden die Daten importiert und in MATLAB in einem Struct gespeichert. Das Struct beinhaltet in der ersten Spalte die Messwerte des HX711 und in der zweiten Spalte das Zusatzgewicht in Gramm. Für die nächste Messung wird ein 100 g Eichgewicht auf die Waage gesetzt und der Messprozess wird wiederholt. Bei Vergleichen der Messwerte des Eichgewichts mit dem ohne Eichgewicht fällt auf, dass die Werte identisch sind. Dies ist auf das Nullen zu Messbeginn zurückzuführen. Das Nullen zu Messbeginn hat den Vorteil, dass bei einer späteren Vermessung der Kaffeeauslaufmenge nur der Kaffee gemessen wird, ohne des Gewichts der Tasse. Hier handelt es sich um eine relative Gewichtsmessung. Um hierfür eine Kalibrierkurve zu erstellen, darf das Eichgewicht zu Beginn der Messung nicht auf der Waage stehen. Erst nach einigen Sekunden muss das Gewicht aufgesetzt werden, so dass eine Differenz zu messen ist. Dieses Vorgehen wird für die Gewichte bis 800 Gramm wiederholt. Die Daten

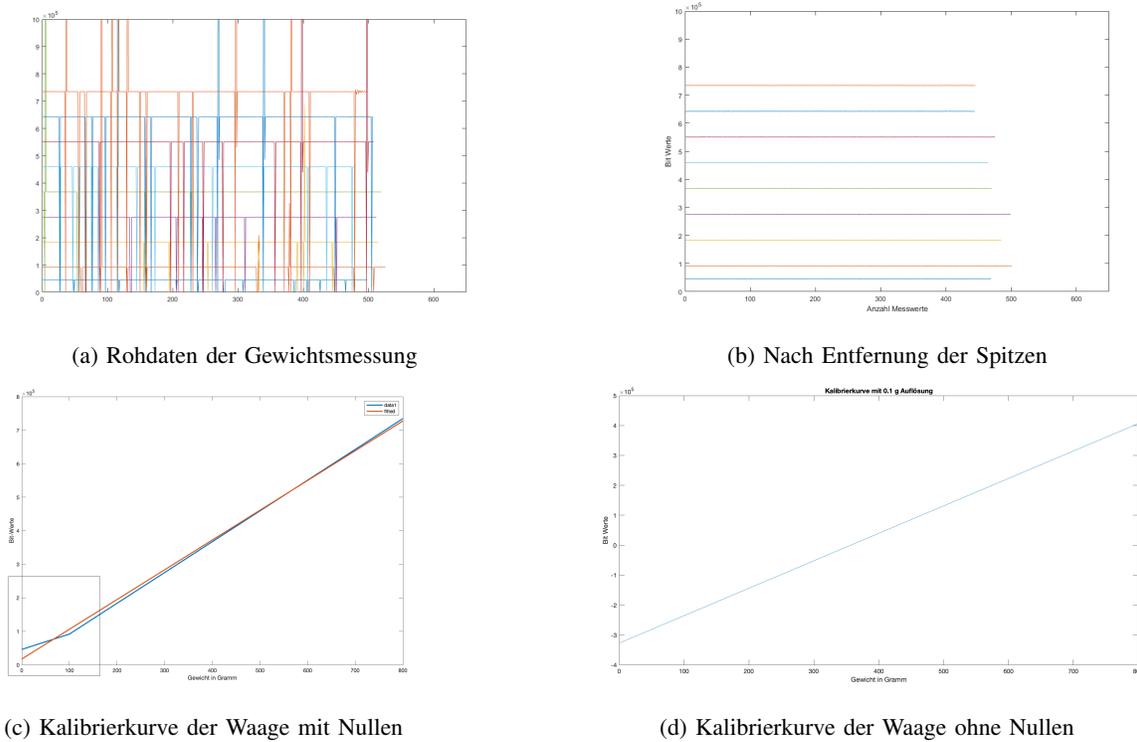


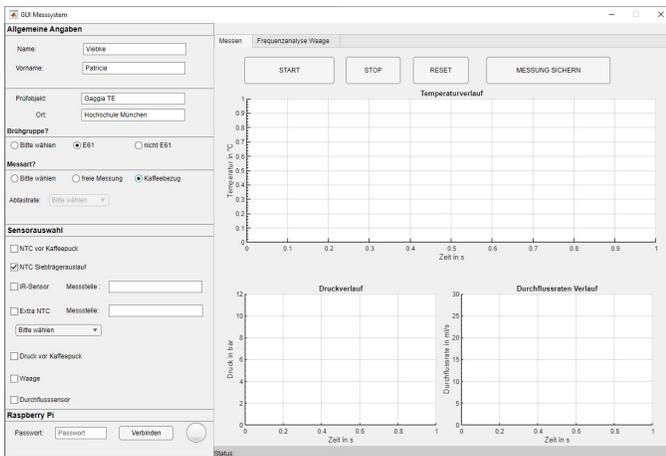
Abbildung 5: Aufbereitung der Messwerte für die Kennlinie der Waage

weisen durch das Aufsetzen der Gewichte einen Sprung auf. Dieser Sprung muss aus den Vektoren entfernt werden, um einen Kalibrierpunkt pro Gewicht zu erstellen. Die graphische Darstellung 5a zeigt die Linien der Eichgewichte, die jedoch Ausreißer enthalten. Dies werden mit der mittleren absoluten Abweichung vom Median entfernt. Als Ausreißer wird hier ein Messpunkt definiert, welcher den dreifachen Wert des Medians überschreitet. Die Linien in Abbildung 5b stellen die gefilterten Messwerte ohne Ausreißer dar. Diese Werte werden pro Kalibrierpunkt gemittelt und mit dem jeweiligen Gewicht gegenübergestellt. Mit dem Basic Fitting wird eine Gerade ermittelt. Abbildung 5c zeigt die Gegenüberstellung der Kalibrierwerte in blau und die gefittete Gerade in orange. Der für die Messung relevante Bereich bis 30 g zeigt jedoch eine deutliche Abweichung der Kalibrierkurve zur gefitteten Gerade. Bei Verwendung dieser Umwandlung, entspricht die gemessene Menge nicht der Realität. Es wird beschlossen das Nullen zu Beginn des Skriptes wegzulassen, so dass die Messwerterfassung im linearen Bereich liegt.

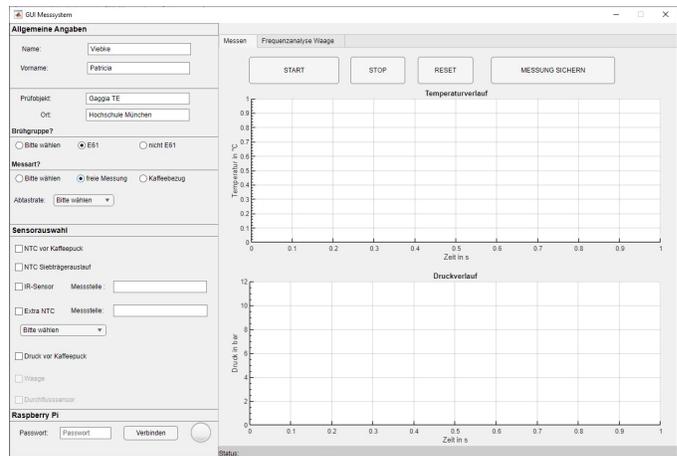
Mit dieser Erkenntnis wird eine neue Kalibrierkurve ohne Nullen erstellt. Für die Erstellung wird analog vorgegangen wie bei der ersten Kalibrierkurve. Es wird erneut in dem Bereich von 0 bis 800 g gemessen. In 100 g Schritten werden Messungen aufgezeichnet. Das Gewicht wird diesmal zu Beginn der Messung bereits auf die Waage gestellt. Da hier die Nullung nicht stattfindet, wird hier der absolute Bit Wert der Waage ausgegeben. Anschließend werden die Messungen der Eichgewichte gemittelt und mit den Gewichten in Gramm gegenübergestellt. In Abbildung 5d ist die Gerade der Gegenüberstellung graphisch dargestellt.

Es entsteht eine stetig steigende Gerade mit einem Nullpunktversatz von etwa -320000 Bit hat. Über die Funktion `fit(gramm, bit, 'linearinterp')` wird ein cfit Element erstellt. Dieses beinhaltet alle interpolierten Werte der Geraden. Die Umwandlung wird getestet durch die Erfassung eines Gewichtverlauf durch Auflegen unterschiedlicher Eichgewichte. Da dies mit Eichgewichten durchgeführt wird, ist die Umwandlung in Gramm nachvollziehbar. Die Umwandlung ist korrekt verlaufen. Somit wird die Kalibrierkurve bzw. das cfit Element in die GUI für die Umwandlung der Bit Werte implementiert.

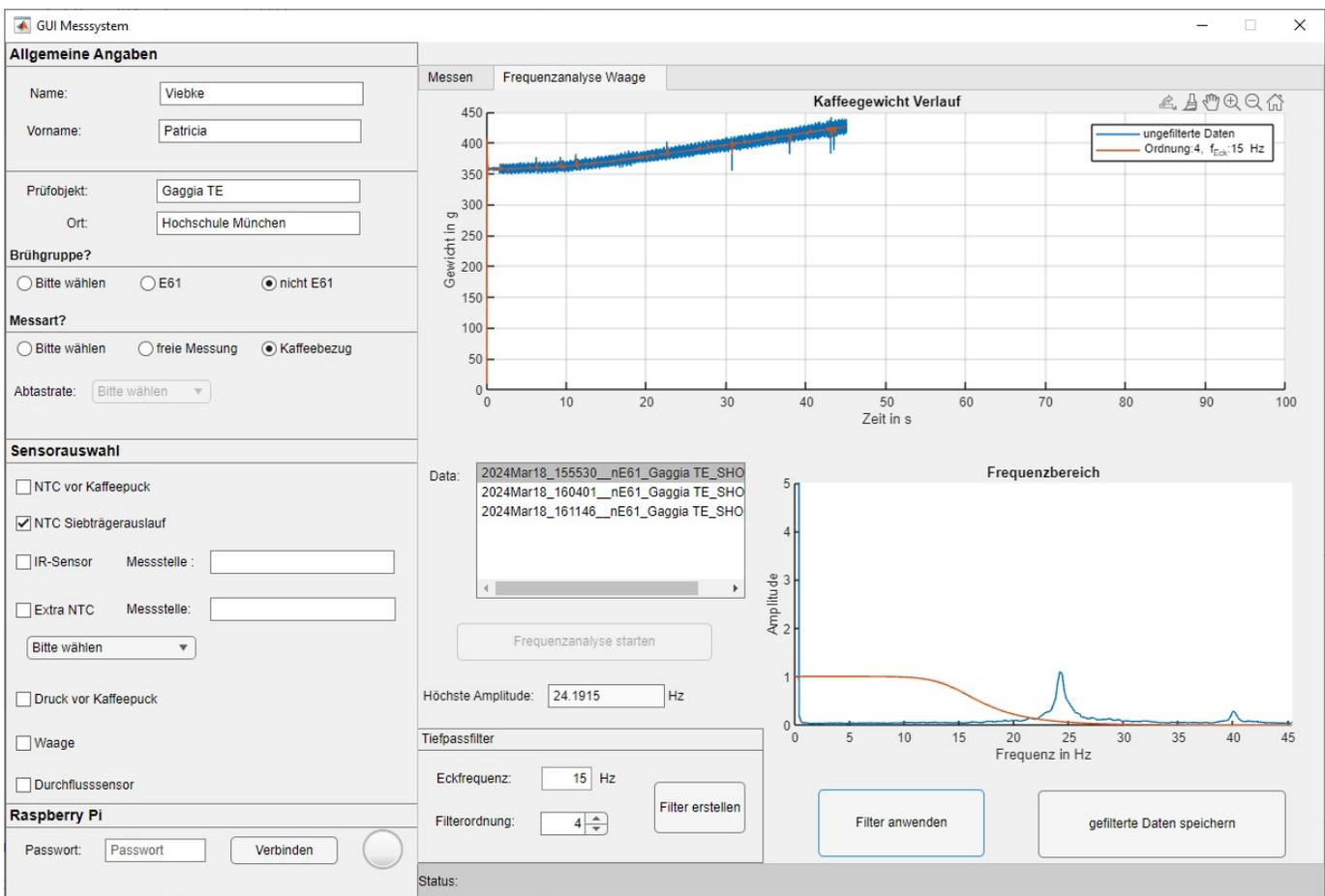
Für die Absicherung werden Probemessungen an der Gaggia TE durchgeführt. Hier besteht die Aufgabe die Durchflussrate über die Kaffeeausflussmenge mit der Waage zu ermitteln. Hierfür werden 20 g Kaffeemehl in einen Siebträger gefüllt und mit 1,2 N getampert. Das Sieb wird in die vorliegende Maschine eingesetzt. Anschließend wird die Waage mit einer daraufliegenden Tasse vorsichtig auf dem Über die GUI wird ein Kaffeebezug gestartet, welcher 60 Sekunden dauert. Nach einigen Sekunden wird der Wasserzulauf für den Kaffeebezug über einen Kippschalter gestartet. Das Auswerten der Auslussmenge zeigt zu Beginn ein sauberes Signal mit einem stabilen Gewicht von 350 g. Nach Einschalten des Wasserzulaufs ist ein Anstieg des Gewichts zu erkennen. Während dem Kaffeebezug ist eine Schwingung im Signal zu erkennen. Eine Signalanalyse zeigt eine Frequenz von 25 Hz. Die Frequenz ist auf das Vibrieren der Pumpe während des Kaffeebezugs zurückzuführen. Für eine saubere Auswertung der Kaffeeausflussmenge muss ein Tiefpassfilter entwickelt werden, welcher die Schwingung aus dem Signal eliminiert.



(a) GUI für Kaffeebezug



(b) GUI für Aufheizverhalten



(c) Tab für die Frequenzanalyse und Filterung

Abbildung 6: GUI mit ihren Funktionalitäten

5. MATLAB GUI

Die MATLAB GUI ist die Benutzeroberfläche, über welches das Mess-System bedient wird und ist in drei Teilen aufgebaut. Der erste Teil befindet sich auf der linken Seite. Dieser ist für das Angeben der Umgebungsparameter einer Messung. Hierunter fallen neben allgemeinen Angaben wie Prüferrname, Bezeichnung des Objekts und Standort auch

messspezifische Angaben.

Für eine Messung muss die Brühgruppenart angegeben werden. Hierfür wird zwischen E61 und nE61 (nicht E61) unterschieden. Weiter muss die Messart ausgewählt werden. Die freie Messung umfasst eine beliebig definierbare Messung, ist jedoch für die Erfassung des Aufheizverhaltens geplant. Für eine freie Messung muss zusätzlich eine Abtastrate ausgewählt werden. Hier stehen 4 Hz, 2 Hz und

1 Hz zur Verfügung. Da es sich bei der Erfassung des Aufheizverhaltens um einen trägen Prozess handelt, welcher über einen längeren Zeitraum verläuft, ist eine maximale Abtastrate von 1 Hz ausreichend.

Für die Darstellung und Abspeicherung der relevanten Messkanäle, müssen die verwendeten Sensoren ausgewählt werden über die Checkboxen. Bei dem IR-Sensor und dem Messkanal CH03, hier Extra NTC, muss die Bezeichnung der Messstelle angegeben werden.

Der zweite Teil der GUI ist der erste Tab der TabGroup auf der rechten Seite. Dieser beinhaltet das Starten, Stoppen, Zurücksetzen und Abspeichern der Messung. Unter den Bedienelementen ist die graphische Darstellung zu entnehmen. Je nachdem, welche Messart ausgewählt wurde, ändert sich die Anzahl der graphischen Darstellungen. In Abbildung 6b ist die freie Messung ausgewählt. Hier sind vor allem die Temperaturverläufe von hoher Relevanz. Der Druck im Boiler während des Aufheizens ist ebenfalls als relevante Messstelle festzuhalten und wird daher mit angezeigt. Bei Umschalten auf den Kaffeebezug wird die graphische Darstellung für die Durchflussrate hinzugefügt, wie in Abbildung 6a zu entnehmen. Bei dem Kaffeebezug wird außerdem die Kaffeeausflussmenge durch die Waage erfasst. Der Verlauf der Messgröße ist auf dem zweiten Tab mit der Frequenzanalyse der Waage vorzufinden. Durch Probemessungen bei der Waage wird festgestellt, dass durch Aufliegen der Waage auf dem Abtropfblech Schwingungen während des Kaffeebezugs erfasst werden. Die Schwingungen sind auf die Pumpe der Kaffeemaschine zurückzuführen. Für eine genaue Auswertung der Daten muss die Störfrequenz ermittelt und eliminiert werden. In Abbildung 6c ist der entsprechende Bereich der GUI dargestellt. Durch eine ListBox besteht die Auswahl der zu analysierenden Daten. Es werden die Datensätze angezeigt, welche unter `C:/0_Messsystem/Data` gespeichert sind. Für die Frequenzanalyse muss ein Datensatz ausgewählt werden, welcher die Messdaten der Waage beinhaltet. Durch Doppelklicken auf die Datei, wird diese geladen und entsprechend graphisch dargestellt. Anschließend wird die Frequenzanalyse durch den Button darunter ausgeführt.

Hier werden die im Signal vorkommenden Frequenzen im Frequenzbereich analysiert und graphisch dargestellt. Für präzise Bestimmung der Frequenzspitzen wird eine Frequenzauflösung von 0.2 Hz verwendet. Durch ein Ausgabefenster wird die Frequenz mit der höchsten Amplitude ausgegeben, um die spätere Auslegung der Eckfrequenz des Tiefpassfilters zu erleichtern.

Für den Tiefpassfilter muss die Eckfrequenz und die Filterordnung angegeben werden. Mit dem Button Filter erstellen wird der Verlauf des Tiefpassfilters im Frequenzbereich angezeigt. In diesem Fall liegt eine Störfrequenz von 24,19 Hz vor und ein Tiefpassfilter 4. Ordnung mit einer Eckfrequenz von 15 Hz wird verwendet, um die die Störfrequenz zu eliminieren. Die Eliminierung ist noch nicht auf das Ursprungssignal angewendet, jedoch ist die Amplitude 0 ab einem Wert von 23 Hz zu erkennen. Mit Filter anwenden wird das Signal ohne der Störfrequenz dargestellt. Die gefilterten Daten werden über den entsprechenden Button im vorhandenen Struct mit den

gefilterten Daten bei `messung.hx.filtered` erweitert und ist unter dem gleichen Dateinamen vorzufinden.

A. Datenverarbeitung der Messprozesse

Die zwei Messarten sind unterschiedlich aufgebaut und benötigen somit jeweils einen eigenen Verarbeitungsprozess in MATLAB. In beiden Fällen werden die über die Messkarten erfassten Spannungswerte an MATLAB übergeben, jedoch zu unterschiedlichen Zeitpunkten. Bei dem Aufheizverhalten werden die Messdaten in der ausgewählten Abtastrate an MATLAB übermittelt, während bei dem Kaffeebezug erst nach 60 s Messdauer. Da die Prozesse unterschiedliche Vorgehen für die Aufbereitung benötigen, müssen hierfür zwei MATLAB Funktionen entwickelt werden, welche die jeweilige Umwandlung gewährleisten. Die Funktion `coffeeshot(app)` ist für das Starten und Aufbereiten der Messwerte eines Kaffeebezugs definiert. Für das Aufheizverhalten wird eine Timer Funktion erstellt, welche Messwerte im ausgewählten Zeitabstand importiert, verarbeitet und graphisch darstellt. Beim Aufheizverhalten ist die Beobachtung der Messwerte eine Anforderung, welche umgesetzt wird durch den Einsatz der `TimerFcn(app)`.

1) Kaffeebezug

Die Funktion für den Kaffeebezug startet mit der `system` Anweisung, um das Skript `coffee_shot.py` auf dem Raspberry Pi zu starten. Das Messung ist aktiv, wenn auf dem Raspberry Pi die LEDs der Module blinken. In den 60 Sekunden Messdauer, ist die GUI weiterhin nicht bedienbar. Nach Ablauf der 60 Sekunden müssen die Daten importiert werden. Durch die individuelle Konfiguration der zwei MCCs, wird pro MCC ein CSV-File erstellt. Ein weiteres CSV-File wird beim Verwenden der Waage mit Messdaten gefüllt. Die drei CSV-Files liegen in unterschiedlicher Form vor und benötigen angepasste Importoptionen. Die Optionen werden durch das einmalige manuelle Importieren in MATLAB ermittelt. Für das Importieren in MATLAB werden zuerst die Optionen `opts` für das CSV-File von MCC0 erstellt. Die Datei wird importiert und unter der Variable `mcc0` gespeichert. Die `opts` werden gelöscht und der Prozess wird für das CSV-File von MCC1 und für das HX711 wiederholt. Die Rohdaten sind in den Variablen `mcc0`, `mcc1` und `waage` gespeichert.

Die in Abbildung 7 grün markierten Schritte, stellen Funktionen dar, die innerhalb der Hauptfunktion aufgerufen werden. Die Daten der beiden Module liegen als Spannungswerte vor und werden mit der Funktion `physWerteSHOT` in physikalische Werte umgewandelt. Bevor die Umwandlung stattfindet, werden die gemessenen Spannungswerte mit 10000 multipliziert und der nächste ganzzahlige Wert wird gebildet. Durch den Faktor 1000 werden die Spannungswerte von Volt in Millivolt umgerechnet. Die zusätzliche Multiplikation mit 10 erlaubt eine höhere Auflösung der Messung auf 0,1 mV.

Für die Umwandlung gilt der Index der Kalibrierkurve als Spannungswert in Millivolt. Diese Werte sind ganzzahlig und somit müssen die gemessenen Spannungswerte auf ganzzahlige Werte gerundet werden.

Die Umwandlung wird mit jeweils einem switch-case pro

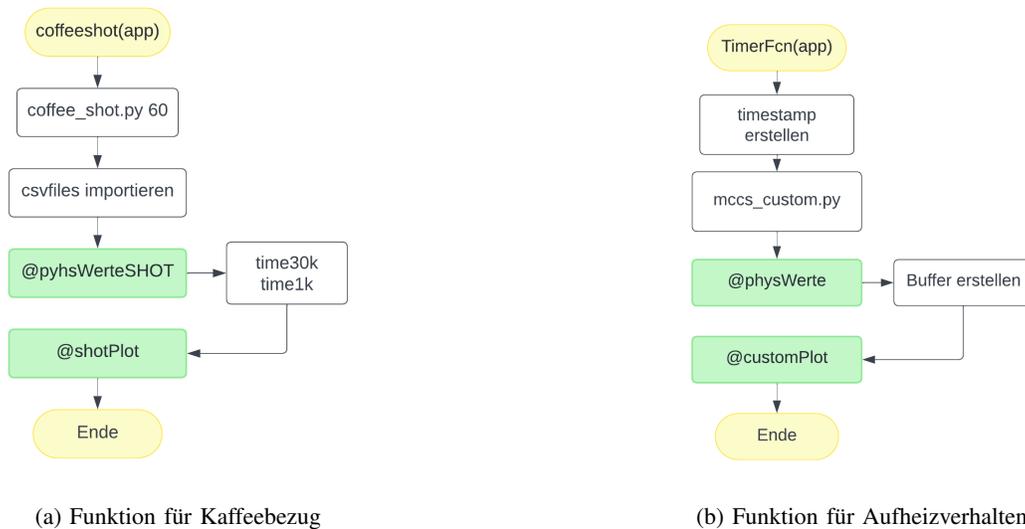


Abbildung 7: Flowcharts für die Funktionen der Messarten

Modul durchgeführt. Als Input müssen die Variablen mit den Rohdaten der Module übergeben werden. Die Messwerte jedes Messkanals werden zeilenweise mit den importierten Kalibrierkurven umgewandelt. Mit der Zeile

```
out1(1,xi) = app.kali00(input1(:,xi))
```

werden die Spannungswerte des Kanals CH00 in Temperaturen in °C umgewandelt anhand der Kalibrierkurve für den entsprechenden Kanal. Die Kanäle CH00 bis CH03 werden in `out1` ausgegeben. Die Kanäle CH10 und CH11 werden in `out2` ausgegeben und für das Flowmeter gibt es eine separate Ausgabe mit `out3`. Die physikalischen Werte der Waage liegen unter der Variable `out4` vor. Das Prinzip der Umwandlung läuft analog zum NTC Sensor ab mit der Kalibrierkurve der Waage.

Für alle Kanäle außer dem Flowmeter liegt eine Kalibrierkurve vor. Die Berechnung der Durchflussrate erfolgt über die Verarbeitung der Periodenlängen und der Schwellwertüberschreitung. Mit dem Befehl `pulseperiod` werden die Periodenlängen und die Zeitpunkte der Schwellwertüberschreitungen des Rechtecksignals ermittelt. Die Periodenlängen werden in dem der Variable `pulse` gespeichert. Eine Überschreitung wird ausgelöst, wenn $\frac{1}{39.9}$ ml Wasser befördert werden. Der Vektor der Periodenlängen wird mit $\frac{1}{39.9}$ multipliziert, um den Durchfluss pro Periode zu errechnen. Durch die Multiplizierung der einzelnen Elemente des Vektors, kommt es zu einer Streuung innerhalb der Durchflussrate. Für die graphische Darstellung und die Sicherung der Messdaten wird eine Interpolation der Durchflussrate durchgeführt. Die Interpolation der Messwerte wird für eine vorgegebene Anzahl an Messwerten durchgeführt. Bei einer niedrigeren Anzahl an Werten pro Sekunde ist ein stärkerer Glättungseffekt wahrzunehmen. Der Nachteil hierbei ist, dass eine Abweichung nicht deutlich zu erkennen ist. Für die Interpolation wird ein neuer Zeitvektor erstellt, welcher eine reduzierte Anzahl an Werten beinhaltet im Vergleich zum Ursprungsvektor `time30k`. Über `polyfit`

```
coeffs = polyfit(time30k, pulse, 5)
```

werden die Koeffizienten einer interpolierten Funktion 5. Grades ausgegeben. Durch die Koeffizienten wird mit dem Befehl `polyval`

```
flow_int = polyval(coeffs, time_int)
```

der reduzierte Vektor mit den interpolierten Durchflussraten berechnet. Bei einer Messdauer von 60 Sekunden und der neuen Abtastrate `fs_int = 25`, entsteht ein interpolierter Zeit- und Datenvektor von 1500 Werten. Der interpolierte Zeit- und Durchflussvektor werden in zwei Spalten der Variable `out3` ausgegeben.

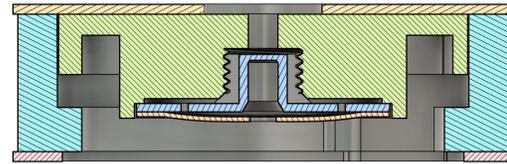
Nach der Umwandlung in physikalische Werte, wird ein Zeitvektor für die Temperaturen und Drücke für die graphische Darstellung benötigt. Der Zeitvektor wird mit der Abtastrate und der Messdauer ermittelt. Es wird ein Zeitvektor `time1k` für die Daten des MCC0 erstellt und ein weiterer Vektor `time30k` für den IR- und Drucksensor. Die Vektoren haben eine Länge 6000 bzw. 1.8 Mio Werten. Für die Waage werden während der Messung bereits Zeitstempel erstellt. Die umgewandelten Werte werden mit dem bereits vorliegenden Zeitvektor der Waage zusammengefügt. Abschließend werden die Verläufe der Sensoren graphisch dargestellt, die in der GUI ausgewählt wurden. Die Verläufe erscheinen in den entsprechenden Diagrammen.

2) Aufheizverhalten

Die Verarbeitung der Daten beim Aufheizverhalten fängt mit der Erstellung des Zeitstempels an. Der iterative Prozess zeigt, dass eine MATLAB-seitige Erzeugung des Zeitstempels sich einfacher gestaltet. Zu Beginn der Timer Funktion wird ein Zeitstempel generiert. Anschließend wird das Python Skript `mccs_custom.py` ausgeführt. Da am Ende des Python Skripts eine `print` Anweisung steht, wird diese auch in MATLAB ausgegeben. Die Ausgabe wird einer Variable zugeordnet und liegt in Form eines Strings vor. Hier sind die Kanäle



(a) zusammengebauter Brühgruppenadapter



(b) Adapter als E61 Brühgruppe

Abbildung 8: Erweiterung für nE61 Brühgruppen

CH00, CH01, CH02, CH03, CH10 und CH11 aktiv. Das Flowmeter und die Waage werden beim Aufheizverhalten nicht benötigt und somit nicht weiter berücksichtigt. Für die Weiterverarbeitung der Messwerte, müssen die sechs Werte des Strings separiert und in numerische Werte umgewandelt werden. Nach Umwandlung liegen die Spannungswerte in einer 1x6 Matrix vor. Die Werte werden an die Funktion `physWerte` übergeben, was die Umwandlung in physikalische Werte durchführt. Das Prinzip und der Aufbau der Funktion läuft wie bei dem Kaffeebezug ab. Der Unterschied ist die Anzahl der umzuwandelnden Werte. Nach Umandlung liegen die Werte in einem Vektor vor, welcher mit dem Zeitstempel an erste Stelle erweitert wird. Der Vektor `zeile` liegt als 1x7 Matrix vor und wird in einen Buffer abgelegt. Bei dem ersten Durchlauf der Timer Funktion entspricht der Buffer gleich der Zeile, bei weiteren Durchläufen wird die Zeile vertikal angehängt. Der Buffer ist für die spätere Datenspeicherung und die graphische Darstellung notwendig. Nach einem Durchlauf der Timer Funktion wird der Buffer mit der Funktion `customPlot` graphisch dargestellt. Es werden jedoch nur die Messwerte der Sensoren angezeigt, die vor Beginn der Messung ausgewählt wurden. Dieser Vorgang findet so lange statt bis die Messung angehalten wird. Nach manuellem Beenden der Messung bleiben die Verläufe graphisch dargestellt, bis diese nicht zurückgesetzt werden.

6. ADAPTER FÜR NICHT E61 KAFFEEMASCHINEN

Das Mess-System hat zum Ziel E61 und nicht E61 Siebträger Espressomaschinen zu vermessen. Für die Vermessung von eigenkonstruierten Brühgruppen, muss eine Erweiterung entwickelt werden [Vie23a]. Für das Messen muss zwischen manuellem und automatisiertem Kaffeebezug unterschieden werden. Bei einem automatisierten Vorgang des Kaffeebezugs ist der Vorgang über die Zeit bzw. über den Durchfluss festgelegt. Hierfür wird im nE61 Siebträger ein Kunststoffeinsatz benötigt, welcher die Totzeit des Wasserzulaufs minimiert. Bei manuellem Bezug wird dieser Einsatz nicht benötigt, da der Kaffeebezug nach beliebiger Zeit gestoppt werden kann. Der Einsatz wird in ein Sieb eingesetzt. In der unteren Bohrung wird ein ELSA Anschluss eingeschraubt. Über diesen wird über 6 mm Rohre die erweiterte Brühgruppe befestigt. Als Erweiterung wird ein Adapter entwickelt, welche als E61 Brühgruppe fungiert

[Kis23]. Der zusammengebaute Brühgruppenadapter ist in Abbildung 8a dargestellt. Dieser wird mit Rohrverbindungen an den Siebträger einer nE61 Espressomaschine adaptiert. In Abbildung 8b ist der Konstruktionsquerschnitt der Adapterbrühgruppe zu sehen. Das Halteblech (gelb) wird über 12 M4x25 Schrauben mit der Siebträgeraufnahme (türkis) verschraubt. Weiter wird das Halteblech über vier M4x12 Schrauben mit dem Brühgruppeneinsatz verschraubt, um eine Versteifung zwischen den Bauteilen herzustellen. Der Wasserverteiler (blau) wird über das bereits vor geschnittene Gewinde im Brühgruppeneinsatz eingedreht. Im Wasserverteiler befindet sich eine M4 Bohrung, über welche die Dusche angebracht wird.

Der Adapter bietet den Vorteil, dass Temperatur und Druck vor dem Kaffee puck gleichzeitig erfasst werden. Eine Bohrung im Brühgruppeneinsatz stellt die Messstelle für den Druck dar. Die Temperatur vor dem Kaffee puck wird über Rohrverbindungen und ELSA Anschluss zwischen nE61 Siebträger und erweiterte Brühgruppe eingesetzt.

Es ist ein Unterschied in der Dicke des Halteblechs in Abbildung 8a und 8b zu erkennen. Dies liegt an der Abwandlung der Konstruktion für das Messsystem. Da das die Erweiterung lose ist und die gleichen Kräfte wie eine Blechversteifung aushalten muss, muss hierfür das Halteblech neu ausgelegt werden.

A. Auslegung des Halteblechs

Die Konstruktion wird ursprünglich von der labortechnischen Espressomaschine übernommen. Der zu beachtende Unterschied hierbei ist, dass das Halteblech an der Labormaschine ein Blechteil ist, welches so gebogen ist, dass die Kräfte aufgenommen werden. Die Erweiterung ist eine lose Brühgruppe, welche eine Platte für Versteifung benötigt. Die Dicke der losen Platte muss neu ausgelegt werden, damit die Kräfte während des Kaffeebezugs kompensiert werden. Die Festigkeit wird anhand der auftretende Scherspannung zwischen Brühgruppeneinsatz (grün) und Siebträgeraufnahme (türkis) berechnet. Die auftretende Kraft verteilt sich auf die Mantelfläche der Kontaktoberfläche der beiden Teile. Es wird angenommen, dass eine Kraft von 5 kN auftritt und die dadurch auftretende maximale Scherspannung kleiner ist als

die zulässige Spannung.

$$\begin{aligned}\tau_{\max} &\leq \tau_{\text{zul}} & (1) \\ \tau_{\text{zul}} &= 0.6 \cdot 190 \text{ N/mm}^2 = 114 \text{ N/mm}^2 \\ \tau_{\max} &= \frac{F}{A} = 1.54 \text{ N/mm}^2 \\ F &= 5 \text{ kN} \\ A &= 2\pi \cdot 12.3 \text{ mm} \cdot 42 \text{ mm}\end{aligned}$$

Durch die Mantelfläche und der auftretenden Kraft wird die maximal auftretende Scherspannung τ_{\max} berechnet. Als zulässige Spannung wird der Wert für Edelstahl verwendet. Der Festigkeitsnachweis zeigt, dass Edelstahl als Material geeignet ist. Neben der Festigkeit, muss anschließend der Steifigkeitsnachweis durchgeführt werden. Für die Steifigkeit wird der Querschnitt nach der Balkentheorie betrachtet. Für die Auslegung der Steifigkeit wird die zulässige Spannung für Edelstahl X5CrNi18-10 (1.4301) verwendet (Roloff Matek TB 1-1). Die Länge der Platte ist mit 120 mm fest definiert. Bei einer Länge von 60 mm tritt somit das höchste Biegemoment auf. Aus der Kraft von 300 kN und der zulässigen Biegespannung, ergibt sich die Dicke der Platte von 8,8 mm.

$$\begin{aligned}W_{\text{erf}} &= \frac{M}{\sigma_{\text{zul}}} = \frac{I}{z_{\max}} & (2) \\ \sigma_{\text{zul}} &= 190 \text{ N/mm}^2 \\ M &= F \cdot l/2 \\ \Rightarrow h &= \sqrt{\frac{6 \cdot W_{\text{erf}}}{b}} = 8,8 \text{ mm} & (3)\end{aligned}$$

Für die Fertigung und die spätere Auswahl der Schrauben wird die Dicke auf 10 mm aufgerundet.

B. Anpassung des Brühgruppeneinsatzes

Der in Abbildung 8b in grün dargestellte Brühgruppeneinsatz muss angepasst werden. Für die Optimierung der Konstruktion müssen zwei Analysen durchgeführt werden.

Die erste Optimierung umfasst die Höhe des Brühgruppeneinsatzes. Beim Einsetzen des E61 Siebträgers in die Brühgruppe, ragt der Brühgruppeneinsatz in das Sieb ein. Die Höhe von der Dichtung bis zur Unterkante des Einsatzes muss anhand der Kaffeemenge im Sieb bestimmt werden.

Hierfür muss zuerst der Mahlgrad angepasst werden von 22 g mit einem Tamperdruck von 1,2 N. An der Gaggia TE wird anschließend ein Kaffee bezogen mit paralleler Erfassung der Durchflussrate über die Waage. Die Waage wird auf dem Abtropfblech der Gaggia TE platziert und eine Tasse wird auf die Waage gestellt. Anschließend wird über die GUI die Aufzeichnung des Kaffeebezugs gestartet. Um den angestrebten Wert von 2 ml/s zu messen, werden die Daten des Kaffeebezugs in MATLAB ausgewertet. Der Kaffeebezug wird über einen Kippschalter manuell gestartet. So lange dieser aktiviert ist, läuft die Pumpe, die das Wasser befördert.

Das Signal wird in MATLAB betrachtet und es ist eine Schwingung im Signal zu erkennen. Durch die Darstellung der auftretenden Frequenzen im Frequenzbereich ist ein deutlicher Peak bei etwa 22 Hz zu erkennen. Es muss ein Tiefpassfilter erstellt und angewendet werden, um die Schwingung zu

Tabelle V: Kaffeemengen mit Abstand zur Siebkante

| Kaffeemenge | 18 | 20 | 22 |
|-------------|------|----|-----|
| Abstand | 10.9 | 10 | 8.5 |

eliminieren. Die Eckfrequenz wird bei 15 Hz definiert. Der Tiefpass bewirkt die Eliminierung der Störfrequenz im Signal. Die Ausreißer sind weiterhin vorhanden, die bei einer späteren Signalanalyse zu berücksichtigen sind.

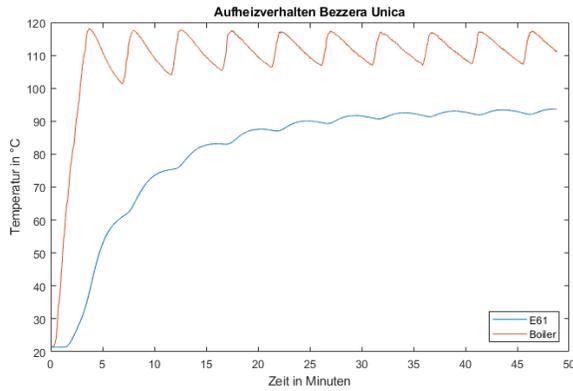
Die Durchflussrate über die Waage wird anhand einer Rechnung der Brüh-Ratio durchgeführt. Es wird eine Durchflussrate von 2 ml/s angestrebt mit einer Menge von 22 g Kaffeemehl. Durch die Brüh-Ratio von 2.5 errechnet sich eine Kaffeeauslaufmenge von 55 g nach 27.5 s Bezugszeit.

Aus den Messwerten der Kaffeemenge wird der das Gewicht zu Beginn der auftretenden Schwingung mit dem Wert nach 27.5 s verglichen. Durch Subtraktion wird der Wert 55 g angestrebt. Bei einer höheren Kaffeemenge muss das Mehl feiner gemahlen werden während bei einer zu geringen Menge das Mehl eine etwas gröbere Textur benötigt. Durch den iterativen Vorgehen in kleinen Schritten wird ein Mahlgrad festgelegt, welcher nach 27.5 s eine Menge von 53 g liefert. Dieser Wert liegt innerhalb der akzeptierten Toleranz und der Mahlgrad gilt somit als definiert.

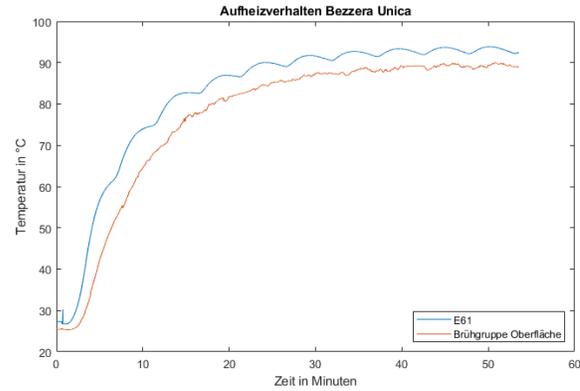
Nachdem der Mahlgrad und der Tamperdruck definiert sind, muss die Höhe von Sieboberkante bis zur Kaffeeoberfläche ermittelt werden. Hierzu wird das Vorgehen auf drei Kaffeemengen angewendet. Es werden 18 g, 20 g und 22 g Vermessen. Mit dem Messschieber werden die Abstände am Rand des Siebes gemessen und anschließend gemittelt.

Es wird beschlossen die Konstruktion für einen Wert von 20 g Kaffeemehl auszulegen. Hierfür wird der Abstand des Brühgruppeneinsatzes verringert, so dass die Dichtung bei getampertem trockenem Kaffee auf dem Sieb aufliegt, ohne dass die Dusche den Kaffee berührt. Es ist ein Abstand bei trockenem Kaffee zur Dusche notwendig, da der Kaffee während des Kaffeebezugs aufquillt.

Weiter wird beschlossen, dass der Brühgruppeneinsatz und der Wasserverteiler aus lebensmittelzertifiziertem Kunststoff 3D-gedruckt werden.



(a) Aufheizverhalten mit IR an Boiler



(b) Aufheizverhalten mit IR an Brühgruppe

Abbildung 9: Aufheizverhalten der Bezzera Unica

7. ABSICHERUNG DES MESS-SYSTEMS

Bei der Absicherung werden alle Funktionen des entwickelten Mess-Systems in Kombination getestet. Hierzu zählen die Python Codes, die MATLAB GUI und die Funktion der verwendeten Sensorik. Da beim Vermessen von Siebträger Espressomaschinen die Komponenten in Kombination verwendet werden, muss das Zusammenspiel zuverlässig funktionieren und somit auch in Kombination abgesichert werden.

Die Funktionalität der Sensoren wird durch einfaches erstes Überprüfen bestätigt. Durch das Ändern des Drucks bzw. der Temperatur am Sensorelement, wird eine Spannungsänderung in dem Terminal ausgegeben. Für die Auswertung des Messprozesses müssen die Messdaten in physikalische Werte umgerechnet werden, um diese interpretieren zu können. Die Kalibrierkurven werden in die GUI eingepflegt und durch Eintauchen der Sensoren in heißes Wasser wird der erwartete Verlauf bzw. Temperaturen in der GUI angezeigt. Mit einer Handpumpe wird analog die Kalibrierkurve für die Drucksensoren überprüft. Durch Abgleich des Drucks auf der Pumpe mit den Werten in der GUI wird die Richtigkeit der Umwandlung gewährleistet. Mit Eichgewichten wird die Kalibrierkurve der Waage überprüft. Die Werte werden korrekt angezeigt, jedoch sind Ausreißer im Signal wahrzunehmen, welche eliminiert werden müssen.

Für die Überprüfung der kombinierten Funktionalität werden Aufheizverhalten und Kaffeebezüge an der Bezzera Unica durchgeführt.

Es werden zwei Aufheizverhalten aufgezeichnet. Der Sensor an der E61 Messstelle (Temperatur vor Kaffeeputz E61) bleibt bei beiden Messungen gleich, wie in Abbildung 9 zu erkennen. Lediglich die Position des Infrarot Sensors wird verändert. Im ersten Aufheizverhalten wird schwarzes Kabelisolierband auf die Boileroberfläche geklebt, um anschließend den IR-Sensor darauf auszurichten. In Abbildung 9a wird die Messung nach etwa 48 Minuten Messzeit beendet, da sich die Temperatur der Brühgruppe bei etwa 90 °C eingependelt hat. Der Boiler zeigt ein leicht abnehmendes Zickzag Verhalten, für welches ein Zweipunktregler des Heizelements die mögliche Ursache ist.

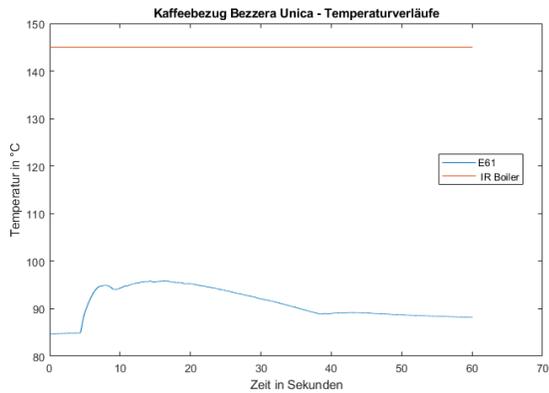
Das Verhalten des Heizelements lässt sich im E61 Temperaturverlauf widerspiegeln. Die Wellen zeigen die abgeschächte Version des Zickzags des Heizelements.

In Abbildung 9b zeigt das Aufheizverhalten mit dem IR-Sensor auf der Brühgruppen Oberfläche. Der IR-Sensor ist nah am E61 Sensor platziert. Es zeigt sich eine Temperaturdifferenz, welches als Gefälle zu interpretieren ist. Durch Raumtemperatur, ist die Temperatur an der Brühgruppen Oberfläche um etwa 7 °C geringer. Weiter sind die durch das Heizelement verursachten Wellen an der E61 Bohrung an der Oberfläche kaum wahrzunehmen.

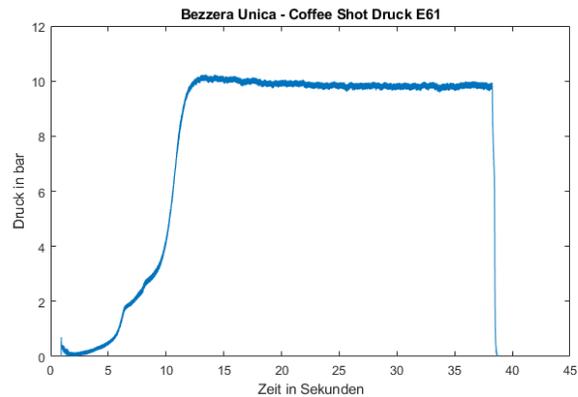
Für die Absicherung werden beide Fälle durchgeführt, da davon ausgegangen wird, dass der Zugang zum Boiler nicht möglich ist. Das Mess-System funktioniert während des Aufheizverhaltens zuverlässig und die Temperaturen sind korrekt. Zwischen den in Abbildung 9 dargestellten Messungen wurde mit der gleichen Funktion der Abkühlprozess aufgezeichnet. Dieser zeigt, dass Messungen mit einer Dauer von 2,5 Stunden zuverlässig aufgezeichnet werden. In allen Fällen wird mit der maximalen Abtastrate von 1 Hz gemessen.

Nachdem die Maschine vollständig aufgeheizt ist, werden Kaffeebezüge aufgezeichnet. Die Bezzera Unica ist eine E61 Siebträger Espressomaschine. Hier ist daher entweder eine Temperatur oder eine Druckmessung möglich. Für den Kaffeebezug werden 20 g des gemahlten Versuchskaffees in den Siebträger der Bezzera Unica gefüllt und mit 1.2 N getampert. Anschließend muss der Messingadapter mit dem Sensor eingeschraubt werden. Der Adapter für die E61 Messstelle müssen mit der Dichtung und Teflonband eingesetzt werden, da die Dichtheit ansonsten nicht gegeben ist. Die Messingadapter müssen mit maximal Handkraft eingeschraubt werden.

Im ersten Kaffeebezug wird die Temperatur vor dem Kaffeeputz und die Temperatur an der Boileroberfläche aufgezeichnet. In Abbildung 10a ist über den Kaffeebezug eine stabile Boilertemperatur zu erkennen. Die Temperatur vor dem Kaffeeputz steigt bis 95 °C und fällt anschließend ab. Während der Messung wird festgestellt, dass die Messdauer



(a) Temperaturverlauf bei Kaffeebezug E61



(b) Druckverlauf bei Kaffeebezug E61

Abbildung 10: Kaffeebezug Parameter Verläufe

von 60 Sekunden zu lang ist und wird daher auf 45 Sekunden reduziert.

In einem zweiten Kaffeebezug wird der Drucksensor in die E61 Messstelle eingesetzt. Eine weitere Messung wird gestartet und ist in Abbildung 10b dargestellt. Sowohl der Temperatur- als auch der Druckverlauf zeigen einen erwarteten Verlauf mit sinnvollen Werten.

Durch das Aufzeichnen von Aufheizverhalten und von Kaffeebezügen wird die Richtigkeit und Zuverlässigkeit des Mess-Systems für E61 bestätigt.

Der gleiche Prozess wird für das Testen des nE61 Adapters verwendet. Hierfür wird ein E61 Siebträger in die Bezzera Unica eingesetzt. Dieser Siebträger entspricht nicht zu 100 % dem Bezzera Siebträger und ist somit nicht vollständig dicht. Der eingesetzte E61 Siebträger hat am Ausfluss ein ELSA Anschluss mit Rohrstück, worüber die Brühgruppenerweiterung adaptiert wird.

In der Brühgruppenerweiterung ist der NTC für vor dem Kaffeepuck nE61 und der dazugehörige Drucksensor von SEED Studio verbaut. Der Adapter für den Drucksensor muss mit Teflonband und Dichtung in den Brühgruppeneinsatz eingeschraubt werden. Es wird ein E61 Siebträger mit 20 g Kaffee gefüllt und getampert und in die Brühgruppenerweiterung eingesetzt.

Anschließend wird ein Kaffeebezug gestartet, welcher die Werte parallel aufnimmt. Die Druck- und Temperaturwerte werden korrekt umgewandelt. Es ist jedoch ein reduzierter Druck von etwa 7 bar erfasst, da der falsche Siebträger in der Bezzera eingesetzt ist.

Die Dichtheit der erweiterten Brühgruppe und Adaption an die Maschine werden bestätigt und die Erfassung und Umwandlung der Messdaten während des Kaffeebezugs läuft korrekt ab.

8. ZUSAMMENFASSUNG UND AUSBLICK

In der Projektarbeit ist es gelungen ein Mess-System für die Vermessung von Siebträger Espressomaschinen erfolgreich in Betrieb zu nehmen. Das Mess-System basierend auf dem Raspberry Pi, bietet über 7 definierte Messkanäle die Erfassung von Parametern an. Es werden Temperaturen, Druck, Durchflussrate und die Kaffeeausflussmenge erfasst. Die Messprozesse sind durch ausgewählte Sensorik als Vermessungsobjekte sind sowohl E61 als auch Maschinen mit anderer Brühgruppe geeignet

Das Mess-System ist vollständig einsatzbereit bietet jedoch noch Punkte zur Optimierung an, um das System stabiler und langlebiger zu gestalten.

Die Messing Adapter für den Temperatur- und Drucksensor an der E61 Messstelle zeigen aufgrund ihren Aufbaus ein erhöhtes Risiko zu reißen, bei zu viel Kraftanwendung. Da hier druckdicht verschraubt werden muss, bietet es sich an die Adapter aus Edelstahl zu fertigen, um eine erhöhte Festigkeit zu erzeugen.

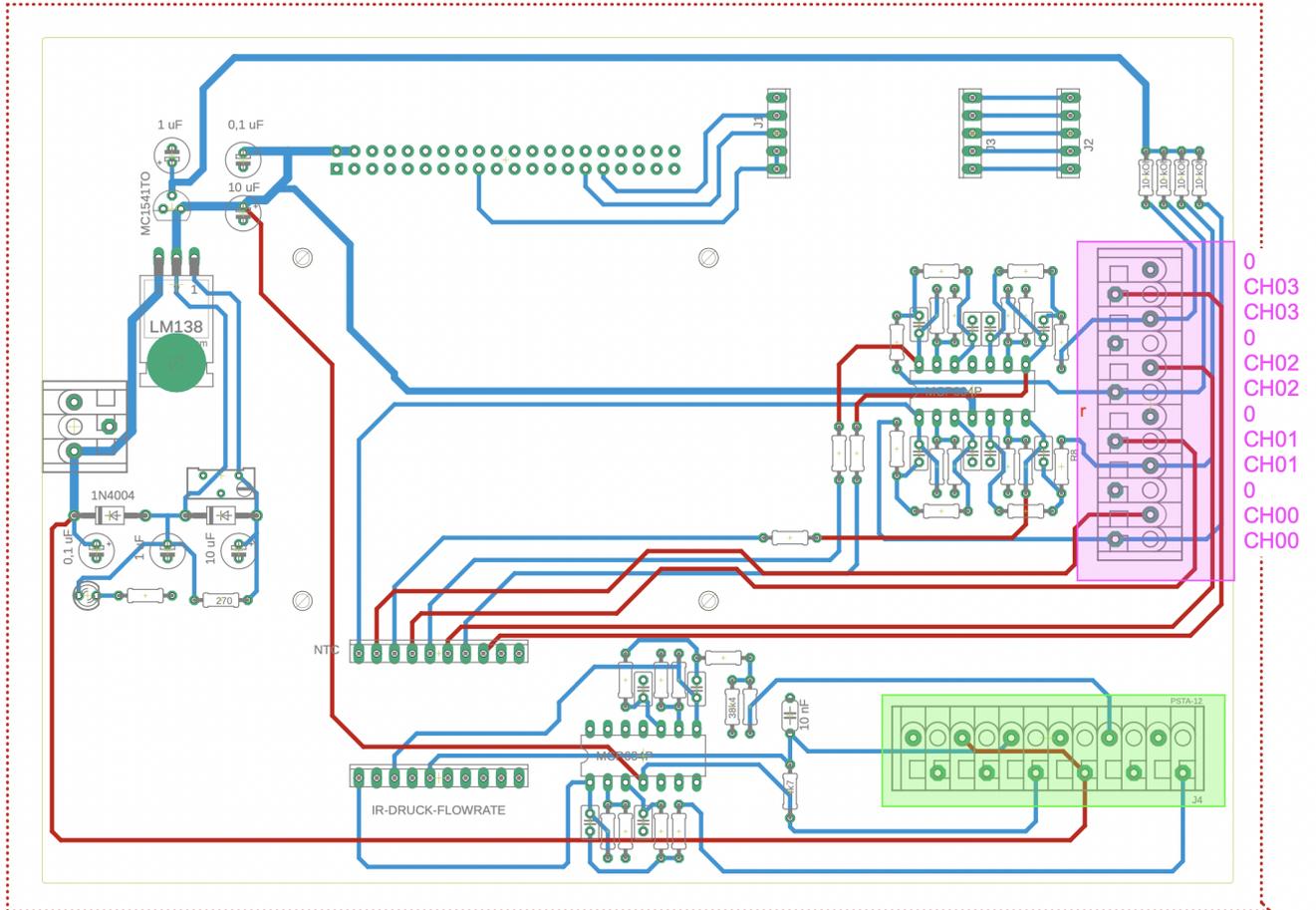
Weiter muss das Gehäuse um das Mess-System aufgebaut werden. Hier müssen Stecker von den Federklemmen nach außerhalb angebracht werden. Über Stecker werden die Sensoren mit dem Mess-System verbunden, was ein komfortables Ein- und Abstecken ermöglicht, ohne die Kabel zu beanspruchen. Im Laufe der Entwicklung sind Erkenntnisse aufgekommen, welche ein optimiertes Layout der Platine benötigen. Der IR-Sensor wird nicht mehr über eine separate Spannungsquelle gespeist, sondern über die 12 V der Platine. Weiter ist die ursprünglich ausgelegte Signalaufbereitung für Druck auf der Platine zu analysieren. Da diese nicht funktioniert, wird der Drucksensor direkt mit dem Messkanal des MCCs verbunden.

LITERATUR

- [Kis23] Felix Kistler. Konstruktionsanpassungen der brühgruppe, 2023.
- [Pyt22] Python Software Foundation. threading — Thread-based parallelism, 2022. Python 3.10.0 documentation.
- [pyt24] queue — a synchronized queue class. <https://docs.python.org/3/library/queue.html>, 2024. Accessed: April 2, 2024.
- [Vie22a] Patricia Viebke. Inbetriebnahme eines Messsystems zur Aufnahme von Betriebsparametern bei Siebträger Espressomaschinen mit und ohne E61 Brühgruppe, 2022.
- [Vie22b] Patricia Viebke. Modulararbeit zum HÜ-Kurs „Messen und Signalanalyse mit MATLAB, 2022.
- [Vie23a] Patricia Viebke. Datenerfassung bei nicht E61 Brühgruppen. Website, May 2023. Accessed on 27th May 2023.
- [Vie23b] Patricia Viebke. Datenerfassung der Durchflussmessung, 2023.

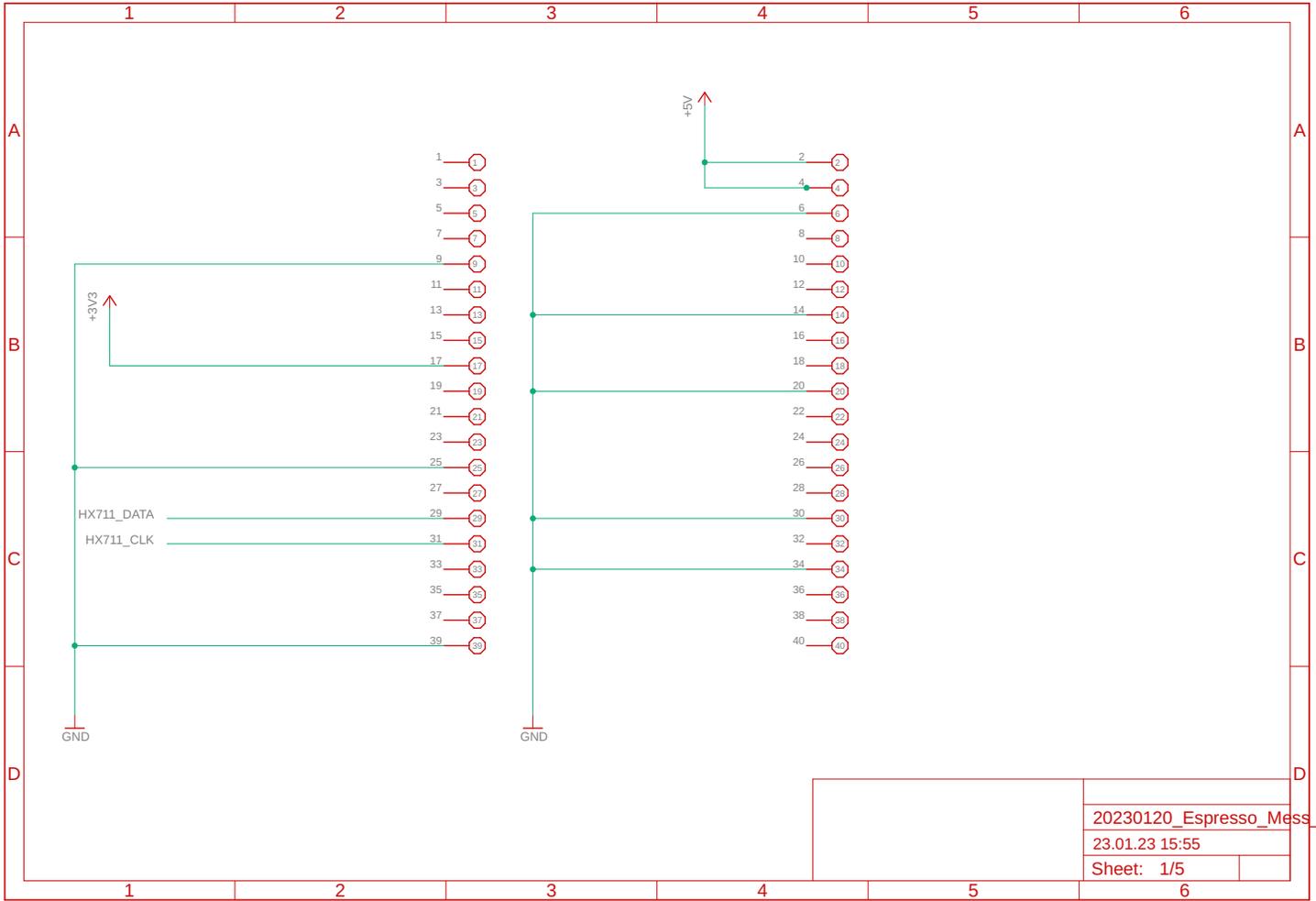
ANHANG

Elektronikplatine

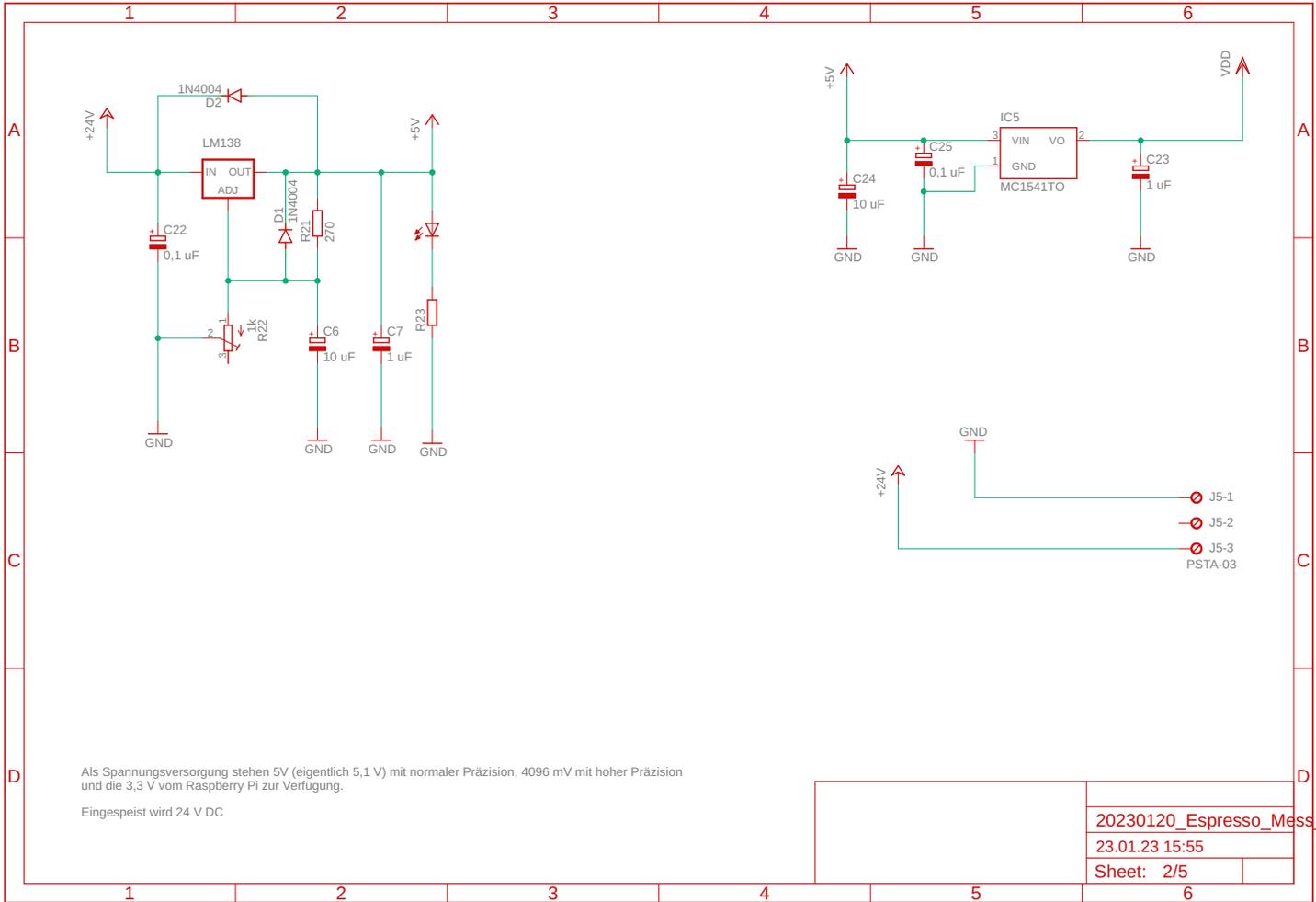


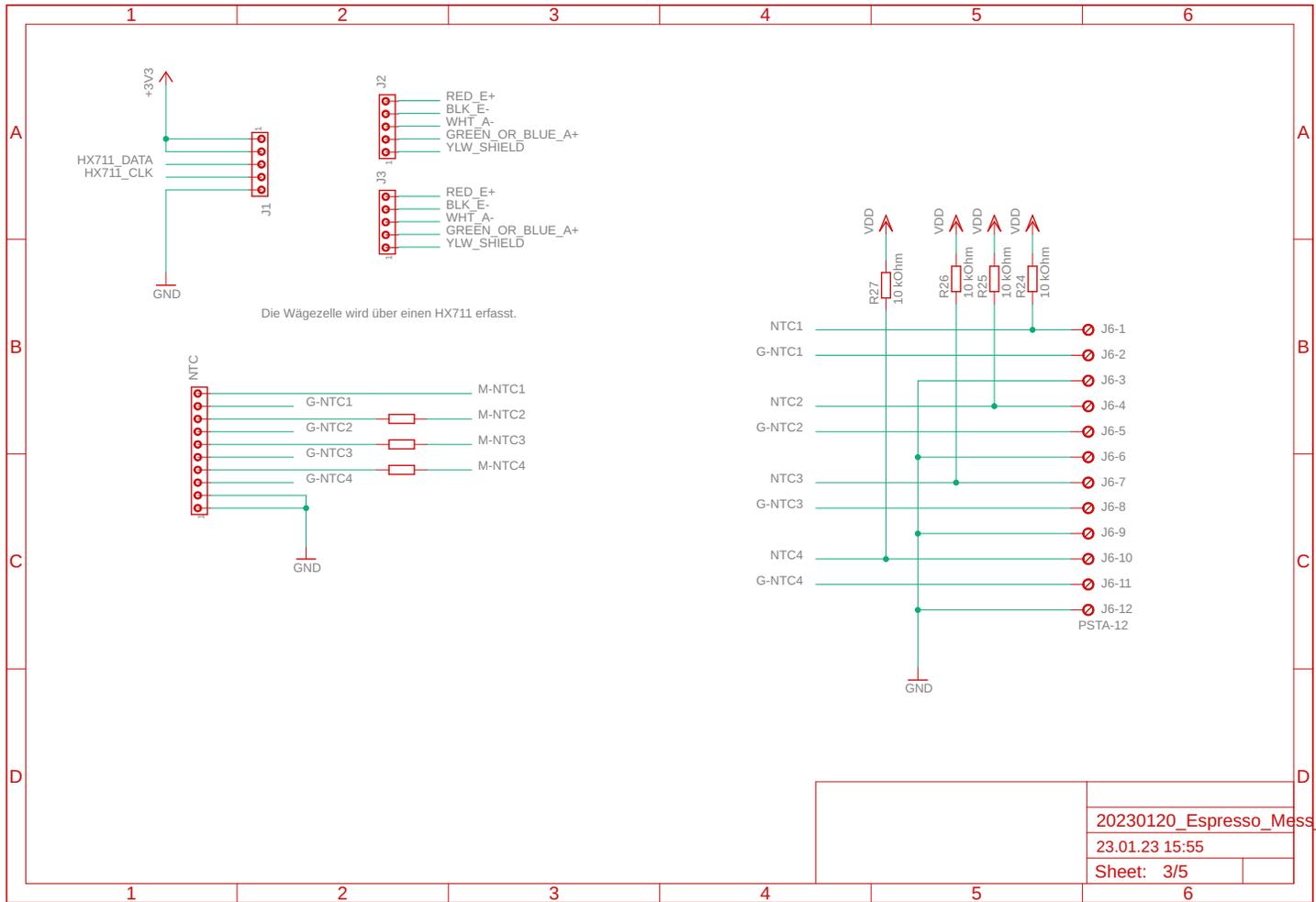
17.12.23 09:49 f=1.50 /Volumes/HOME_ARMIN/20_Auftraege_Projekte/04_Espressomaschine/07_Elektronik_u_Signalaufbereitung/10_Mess_System/02_Messplatine/20232001_Fina

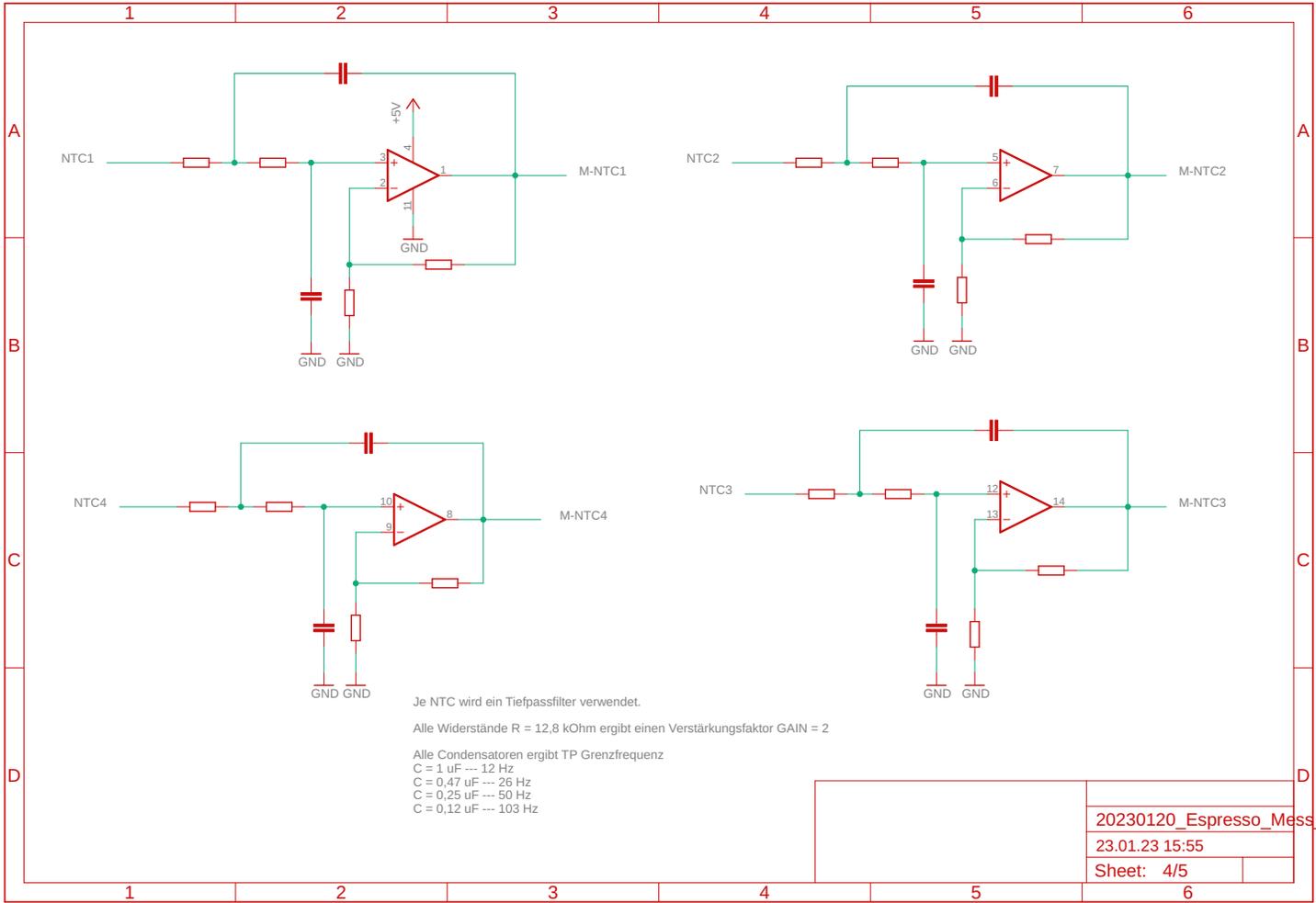
Abbildung A.1: Schaltplan der Platine

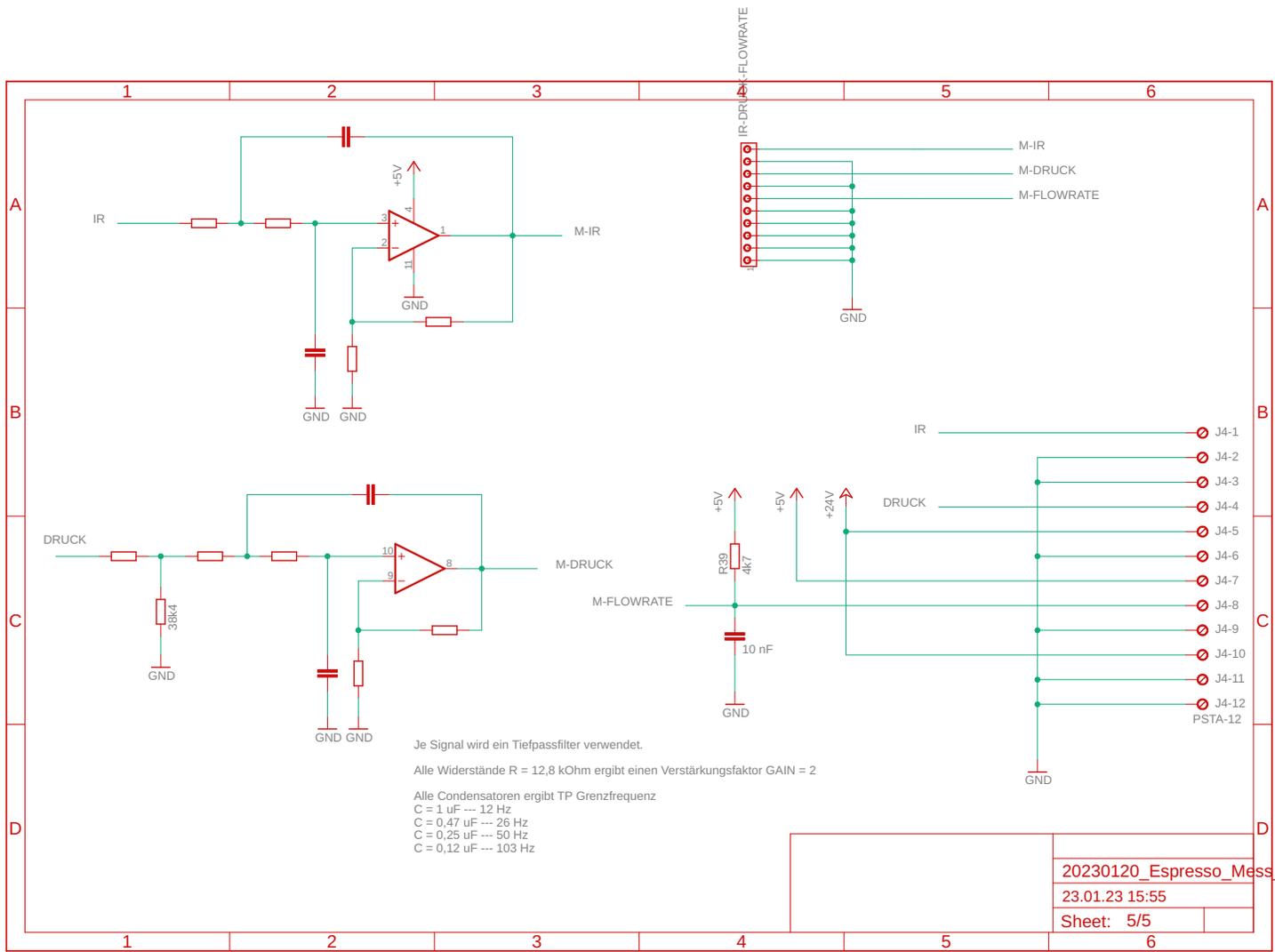


20230120_Espresso_Mess_System
 23.01.23 15:55
 Sheet: 1/5









weitere Abbildungen

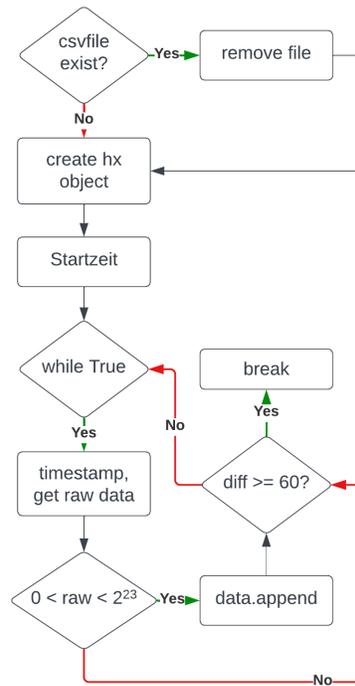


Abbildung A.2: Ablauf des Skriptes für HX711 Messung