

**Hochschule für Technik, Wirtschaft und Kultur
Leipzig**

Fakultät Ingenieurwissenschaften

Bachelor-Studiengang Elektro- und Informationstechnik

**Sensorbasierte Messplattform zur Analyse der
Ameinsensäurebehandlung im Bienestock**

Bachelorarbeit Nr. 123/45

von

Ernst Johann Emmerich

geb. am 23.04.2001

in Leipzig

74949

Verantwortlicher Hochschullehrer: Prof. Dr.-Ing. Gerold Bausch
Leipzig, Dezember 2019 – Februar 2020

Erklärung

Ich versichere wahrheitsgemäß, die Bachelorarbeit selbständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

.....
Ernst Johann Emmerich
Leipzig, den 29. Oktober 2025

Danksagung

Zunächst möchte ich meinem Prof. Dr.-Ing. Gerold Bausch danken. Auch Dipl.-Ing. Mattias Müller gilt mein Dank.

⋮

Zuletzt danke ich meinen Freunden, meinen Eltern sowie meiner Familie für die ständige Unterstützung während meines Studiums.

Inhaltsverzeichnis

1 Einleitung	7
1.1 Varroose	7
1.2 Zielsetzung	9
2 Theoretischer Hintergrund	10
2.1 Erwartete Messumgebung	10
2.2 Sensoren	11
2.2.1 Übersicht über Temperatursensoren	11
2.2.2 Ameisensäuresensor	13
2.3 Energiebedarf	14
2.3.1 USB	15
2.4 weitere Peripherie	15
2.5 Datenspeicher	16
3 Methoden und Arbeitsweise	18
3.1 Systemaufbau	18
3.1.1 Mikroprozessor	19
3.1.2 Temperatur- und Feuchtesensoren	19
3.1.3 Ameisensäuresensor	20
3.1.4 Kommunikation mit dem Messsystem	21
3.2 Energiebedarf	22
3.2.1 weitere Peripherie	22
3.2.2 Versorgungsspannung	23
3.2.3 Abschätzen des zu erwartenden Energieverbrauchs	23
3.2.4 Optimierungsmaßnahmen	24
3.2.5 Zusammenfassung	25
3.3 Programmaufbau	25
3.3.1 Main Loop	26
3.3.2 Messablauf	27
3.3.3 Datenspeicherung	27
3.3.4 Datenausgabe vom Controller	29
3.3.5 Datenausgabe am PC	30
4 Ergebnisse und Validierung	32
4.1 Funktionalität Messsystem	32

4.2	Datenspeicherung und Ausgabe	32
4.3	Bestimmen der Validierungspunkte	32
4.4	Validierung des Ameisensäuresensor	33
4.5	Validierung der Temperatur- und Feuchtesensoren	34
4.6	Validierung der Datennverarbeitung	35
5	Diskussion und Ausblick	36
5.1	Diskussion des aktuellen Standes	36
5.2	Ausblick	36
Anhang		40

Abkürzungen

ADC Analog zu Digital Wandler (analouge to digital converter)

AL Außenluft

AS Ameisensäure

ASS Ameisensäuresensor

MS Messsystem

NTC Heißleiter (Negative Temperature Coefficient Thermistor)

PTC Kaltleiter (Positive Temperature Coefficient Thermistor)

SL Stockluftkonzentration

SL Stockluft

UART Universal Asynchronous Receiver/Transmitter

VOC Flüchtige Organische Verbindung (volatile organic compound)

1 Einleitung

1.1 Varroose

Die einheimische Honigbiene (*Apis mellifera* L.) ist verschiedenen Stressfaktoren ausgesetzt. Der mit bedeutendste Stressfaktor ist die Varroamilbe (*Varroa destructor* Anderson und Trueman). Der Befall der Milbe an Honigbienen ist eine ektoparasitäre Erkrankung dar, welche als Varroose bezeichnet wird.

Der Befall eines Bienevolkes mitnist Die Varroamilbe *Varroa destructor* ist der bedeutsamste Bienenparasit. Dieser ist in Deutschland flächendeckend verbreitet. Die Milbe parasitiert nicht nur die Bienebrut, welche sie dadurch schädigt, sondern überträgt als biologischer und mechanischer Vektor auch verschiedene Bienenviren, welche zur weiteren Schädigung der Völker beitragen können.

Wir die Varroamilbe nicht regelmäßig genug bekämpft, um den Befall unter der Schadensgrenze zu halten, sterben die Bienenvölker ab. [1]. Zur Bekämpfung der Varroamilbe stehen verschiedene Optionen zur Auswahl. Keine der Behandlungsoptionen hat dabei das Ziel die Varroose komplett zu beenden, sondern lediglich den Befall deutlich zu senken, da ein Varroamilben freies Bienenvolk in Europa praktisch nicht zu erreichen ist. Die Bekämpfungsmethoden werden in zwei kategorien unterteilt, zum einen biotechnische Methoden wie die Drohnenbrutentnahmen, das Käfigen der Königin oder das Bannwaabenverfahren. Die anderen Methoden pharmakologischer mit akarizid wirkenden Tierarzneimitteln. Bei den akariziden Wirkstoffen wird wiederum in synthetische Akarizide, organische Säuren

und Substanzen pflanzlichen Ursprungs unterschieden [1]. Eine der organischen Säuren, welche als Tierarneimittel zur Varroabekämpfung zugelassen ist, ist die Ameisensäure (AS), welche als einziger Wirkstoff die Milben in der verdeckelten Bienenbrut schädigen oder sogar abtöten kann. Die AS steht als 60%ige Lösung oder als Gelstreifen zur Verfügung. Beide Verabreichungsarten verdunsten dabbei die AS in der Stockluft (SL) innerhalb von 7 bis 10 Tagen.

Für eine wirksame Bekämpfung der Varroamilbe mit AS muss eine gewisse Konzentration in der SLim Bienenstock erreicht werden. Dabei sind die zu applizierenden Mengen im Bienenstock in den Zulassungsbedingungen genau beschrieben. Die tatsächlich im Bienenstock erreichten Konzentrationen sind aber kaum bekannt. Bei der Behandlung mit Gelstreifen wurden typischerweise Spitzenkonzentration in einem Bereich von $55 - 85\mu\text{g}/\text{cm}^3(\text{ppm})$ beschrieben. Die Mindestkonzentration über mehrere Tage blieb üblicher weise über 20ppm , wobei beide Konzentrationen abhängig von dem Aufbau der Beute und der Reaktion des Bienenvolkes auf die Wetterverhältnisse war [2].

Es gibt aktuell keine Untersuchungen der tatsächlich erreichten SLkonzentrationen in Abhängigkeit von Temperatur und Luftfeuchte. Dies ist aber nötig um ein möglichst wirksames und bienenschonendes Behandlungsregime zu Identifizieren. sebehandeltgehen an dieser in relativ kurzer Zeit zugrunde Die Behandlungsartengegen diese Krankheit können in zwei Kategorien unterteilt werden: mechanisch und chemisch. Bei der Mechanischen Behandlung wird während der Brutzeit im Frühjahr und Frühsommer, zur Schwarmzeit, gezielt verdeckelte Drohnenbrut entfernt. Durch die längere Verdeckelungszeit der Drohnenbrut priorisieren die Muttermilben gezielt den Befall dieser.



1.2 Zielsetzung

Ziel der Arbeit ist die Entwicklung und validierung eines sensorbasierten Messsystem zur Überwachung der Ameisensäurebehandlung im Bienestock ~~sein~~. Es gibt fast keine belastbaren Messdaten über den Konzentrationsverlauf der AS während der Varroabe-handlung nach der Zeit im Bienestock. Diese Information ist aber wichtig um aktuelle Behandlungskonzepte zu verbessern und zu validieren und gegebenenfalls neue Konzepte zu entwickeln.

Zu diesem Zweck soll ein kompaktes, Energieeffizientes Messsystem realisiert werden, welches Temperatur, Luftfeuchte in der SL sowie der Außenluft (AL) und die Gaskonzen-tration der AS in der SL erfasst. Die Werte werden alle fünf Minuten erfasst und dann in dem System gespeichert. Zur Analyse der Daten soll ein CSV Dokument erzeugt werden. Das System wird per USB mit Strom versorgt, da so ein stationärer Betrieb per Netzteil als auch ein Betrieb per Powerbank ohne änderung am Messsystem gewährleistet werden kann.

Kernaspekte der Arbeit sind die Auswahl geeigneter Sensoren, die Auswahl der weite-ren nötigen Hardware und die Entwicklung der Software für die Datenerfassung und -speicherung.

2 Theoretischer Hintergrund

2.1 Erwartete Messumgebung

Der Bau des Gesamtsystems soll in Erfahrung bringen wie hoch die Ameisensäurekonzentration in der Luft im Bienenstock während der Behandlung mit den in Europa zugelassenen Arzneimitteln ist. Dies ist die so genannte Stockluftkonzentration (SL)

Zum Einschränken des Messbereiches der Sensoren wurde das Maximum der zu erwartenden SL simuliert. Hierbei wurde eine übliche Behandlung als Vorgabe genommen. Dabei beträgt das Beutenvolumen, das Innenvolumen des Bienestocks, 77 l , das Volumen der applizierten AS 290 ml und die Luftwechselrate im Bienenstock 1-, 6- und 10-mal die Stunde.

Weitere Variablen wie die Absorption im Honig wurden nicht berücksichtigt, da diese sehr schwer zu simulieren sind.

WEITER AN SIMULATION OPTIMIEREN

Mit AS darf erst nach der letzten Honigernte im Jahr behandelt werden [3]. Damit die 60% ige Lösung ordentlich Verdampft sollte die Temperatur zwischen 10°C und 30°C liegen und die Luftfeuchte nicht zu hoch sein.

In Leipzig treten im Sommer Temperaturen von um die zehn Grad in kalten Nächten bis hin zu über 40°C an heißen Sommertagen auf [4]. Da Bienen das Stockinnere nahezu konstant bei 34°C – 36°C halten [5], hat ~~auschließlich~~ ^{wesentlich} die Außentemperatur einen Einfluss auf die Wahl der Temperaturfühler für das Messsystem.

Neben der Temperatur können Honigbienen auch die Luftfeuchte im Bienenstock regulieren

6. Durch die Anpassung sind hier ebenfalls wieder die externen Wetterfaktoren für die Sensorwahl ausschlaggebend.

2.2 Sensoren

Das Messsystem benötigt zur Erfassung der Daten mehrere Sensoren. Es werden zunächst Temperatursensoren welche jeweils in den Bienenstöcken messen. Ein dritter sitzt außerhalb der Beuten im Schatten und misst die Umgebungstemperatur.

Außerdem muss die AS Konzentration der Stockluft gemessen werden.

2.2.1 Übersicht über Temperatursensoren

Die Temperatursensoren sollen alle von der gleichen Bauart sein und einen integrierten Feuchtesensor besitzen. Der Anschluss der Sensoren soll möglichst einfach halten, um den gesamtumfang des Messsystems (MS) nicht zu groß werden zu lassen. Hier soll ein grober Abriss über die erhältlichen Temperatursensoren gegeben werden.

Temperaturabhängige Widerstände

*würde ich nicht
unterschreiben, digital ist meist
einfacher*

Temperaturabhängige Widerstände sind am einfachsten Auszutragen. Diese benötigen lediglich eine kleine Zusatzschaltung bestehend aus einem Spannungsteiler welcher einem Analog zu Digital Wandler (analog to digital converter, ADC) speist.

Temperaturabhängige Widerstände welche sich gemäß ihrer Kennlinie in ihrem Wert verändern heißen Thermistor. Diese werden in Heiß- und Kaltleiter unterteilt. Heißleiter (Negative Temperature Coefficient Thermistor, NTC) haben dabei einen negativen Temperaturkoeffizienten, sodass sie bei Temperaturerhöhung der elektrische Widerstand sinkt. NTCs haben in der Regel ein nicht lineares Verhalten, und können bis circa 250°C eingesetzt

werden.

Kalteiter (Positive Temperature Coefficient Thermistor, PTC) erhöhen ihren Widerstand wenn sie erwärmt werden, dieser Verlauf zeigt ein deutlich lineareres Verhalten auf als bei NTCs. Diese sind auch für deutlich höhere Temperaturen erhältlich. Platinwiderstände zeigen diese Charakteristik auf.

Thermoelemente

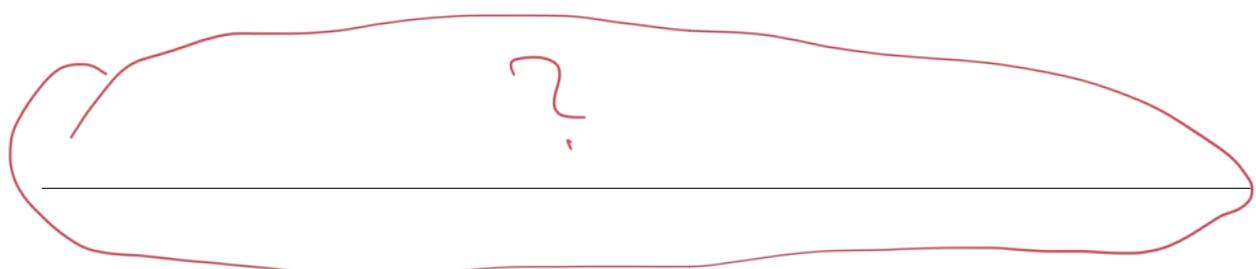
Thermoelemente bestehen aus zwei an der Messstelle verbundenen Metalldrähten unterschiedlichen Materials.

Diese generieren über die Temperaturdifferenz zwischen Messstelle und Vergleichsstelle eine Thermospannung über den Seebeck Effekt. Hierbei ist es essenziell, dass die Temperatur an der Vergleichsstelle bekannt ist, da der Senso nur Differenzen ermitteln kann. Da die resultierenden Thermospannungen sehr gering sind ($\approx 40\mu V/K$) muss das Signal sehr exakt erfasst werden, um verlässliche Ergebnisse zu erzielen.

Ein großer Vorteil von Thermoelementen ist der Goße Messbereich, welcher von fast null bis eintausend Kelvin betragen kann.

Analoge Halbleiter Temperatursensoren

Analoge Halbleiterthermometersensoren nutzen das temperaturabhängige Verhalten von p-n-Übergängen in Silizium, um eine messbare elektrische Größe zu erzeugen. Durch integrierte Schaltungen wie Konstantstromquellen und Verstärker liefern sie ein lineares Spannungs- oder Stromsignal, das direkt einer Temperatur entspricht. Diese Signale können



einfach mit einem Mikrocontroller oder Messgerät ausgewertet werden, erfordern jedoch eine stabile Versorgungsspannung und sorgfältige Leitungsführung.

Worum?

Digitale Temperatursensoren

Es gibt verschiedenste Arten an ^{von} Digitalen Temperatursensoren. Diese basieren zumeist auf Halbleitersensoren und sind häufig mit anderen Technologien wie zum Beispiel Luftfeuchte oder Luftdrucksensoren kombiniert. Zur Kommunikation mit diesen werden dann unterschiedliche Protokolle verwendet, single-wire, PWM modulierte Signale oder Bussysteme sind mögliche Varianten.

Durch die ohnehin schon verbaute Elektronik und sich dadurch ergebende Flexibilität kann bei Digitalen Sensoren weder über den Genauigkeits- noch Messbereich eine konkrete Aussage getroffen werden, da diese Modellspezifisch sind.

2.2.2 Ameisensäuresensor

Für die detektion von AS als Gas sind vor allem Sensoren zur Arbeitsplatzüberwachung erhältlich, meist als fertiges System. Diese sind aber zur automatischen Datenerfassung in einem kompakten System nicht geeignet

Gassensoren für spezifische Gase Messen die Reaktionsrate einer Redoxreaktion. Bei diesen Reaktionen entsteht ein elektrischer Strom welcher über weitere Elektronik in ein lineares Spannungspotential umgewandelt werden kann.

Damit die Reaktionsrate an den Messelektroden bekannt ist, muss das Gas erst durch eine Restriktorschicht. Dies kann zum Beispiel durch eine Kapillarbündel erzeugt werden. Bild vom Aufbau Dies hat auch noch den wünschenswerten Nebeneffekt, dass die Sensoren fast unabhängig vom anliegenden Partialdruck sind. Dadurch kann die Flussmenge über die Elektroden

per Pumpe frei gereglet werden.

Durch die Reaktion an den Elektroden verlieren die Sensoren mit der Zeit ihre Empfindlichkeit.

2.3 Energiebedarf

Für den Betrieb des Systems wird eine stabile und effiziente Stromversorgung benötigt, um eine zuverlässige Funktion aller Komponenten sicherzustellen. Die Spannungsversorgung erfolgt über die standardisierte USB-Schnittstelle, welche typischerweise eine Betriebsspannung von 5 V bereitstellt. Diese Versorgungsspannung ist sowohl für den eingesetzten Mikrocontroller als auch für die angeschlossenen Sensoren geeignet, sofern eine entsprechende Spannungsregelung auf 3,3 V erfolgt. Durch die Nutzung der USB-Versorgung kann das System flexibel an gängige Energiequellen angeschlossen werden, wie etwa Computer, Netzteile oder mobile Akkusysteme.

Bei der Verwendung von Powerbanks ist zu beachten, dass ein signifikanter Unterschied zwischen der nominellen und der tatsächlich nutzbaren Kapazität besteht. Die in den Powerbanks integrierten DC/DC-Wandler arbeiten verlustbehaftet, da die internen Lithiumzellen typischerweise eine Nennspannung von 3,7 V besitzen und auf 5 V gewandelt werden müssen. Aufgrund dieser Umwandlungsverluste steht nur ein Teil der angegebenen Energie für den Verbraucher zur Verfügung. Eine Effizienz von etwa 80 % gilt hierbei als realistische Annahme [7].

Unter dieser Annahme können aus einer Powerbank mit einer nominalen Kapazität von 10 Ah bei 14 Tagen Laufzeit etwa 90 mWh nutzbare Energie kontinuierlich bereitgestellt werden. Das Gesamtsystem muss daher stromsparend konzipiert sein, um einen dauerhaften

Batteriebetrieb zu ermöglichen. Maßnahmen wie das Abschalten nicht benötigter Peripherie, die Nutzung von Energiesparmodi des Mikrocontrollers und ein sparsames Messintervall tragen wesentlich zur Verlängerung der Betriebsdauer bei.

2.3.1 USB

Bei der Einführung des USB Standards waren nur sehr geringe Datenraten und nur eine Leistungsversorgung von 0,5W möglich. Dies hat sich inzwischen aber geändert.

Für einfache Verbraucher wie Mikrocontroller oder Sensorik genügt in der Regel die

Spezifikation	Spannung [V]	Stromstärke (max, [A])	Kommunikation nötig
USB 1.0/1.1	5	0,1 <i>nach Anmeldung und 0,5</i>	nein
USB 2.0	5	0,5	ja
USB 3.0	5	0,9	ja
USB BC1.2	5	1,5(min)	ja
USB 3.2	5	3	ja
USB-PD SPR	5,9,12,20	5	ja
USB-PD EPR	5,9,12,20,28,26,48	5	ja

Tabelle 2.1: Übersicht der USB Standards bezogen auf Spannung und Stromstärke

Grundversorgung von 5 V, die ohne zusätzliche Kommunikation bereitgestellt [8]. Höhere Ströme oder alternative Spannungen – etwa 9 V oder 12 V – setzen hingegen eine digitale Aushandlung zwischen Quelle und Verbraucher voraus. Diese erfolgt zwischen den USB Controllern der Geräte gemäß des USB-PD standards.

2.4 weitere Peripherie

Der Ameisensäuresensor muss laut der Beschreibung des Transmitter Boards bei der Kalibrierung mit 0,3-11 Luft die Minute überströmt werden [9]. ~~Um zuverlässige Werte zu generieren wird dieser Volumenstrom im Messsystem ebenfalls angepeilt.~~ Dafür muss eine Pumpe mit dieser Förderleistung eingebaut werden. Diese kann dann per Relais

angesteuert werden.

UART

Universal Asynchronous Receiver/Transmitter (UART) ist ein Protokoll zum Austausch von seriellen Daten zwischen Geräten [10]. Dabei benötigt das Protokoll nur eine Sende- und eine Empfangsleitung. Eine Takteleitung ist nicht nötig, Sender und Empfänger müssen auf eine gemeinsame Baudrate und den Aufbau der Datenframes eingestellt werden.

Start und Stoppbits markieren dabei den Beginn oder das Ende der Nutzerdaten. Zwischen diesen Bits werden 5 bis 9 Datenbits und ein Paritätsbit immer abwechselnd gesendet. Durch ihre einfache Implementierung und die geringen Hardwareanforderungen wird die UART-Schnittstelle häufig für die Kommunikation zwischen Mikrocontrollern, Sensoren und Computern eingesetzt.

2.5 Datenspeicher

Bei der digitalen Erfassung und Speicherung von Sensordaten spielt die effiziente Nutzung des verfügbaren Speichers eine zentrale Rolle. Der Speicherbedarf hängt ~~direkt~~ von der Anzahl der zu erfassenden Messwerte sowie der Bitbreite der jeweiligen Variablen ab. Messwerte, welche beispielsweise keine vollständigen Bytes benötigen, können kompakt gepackt werden. So ist es möglich, vier 6-Bit-Werte in nur drei Bytes zu speichern und damit den effektiven Speicherverbrauch um 25 % zu reduzieren.

Bei einfacher Speicherung der Daten ergibt sich die benötigte Speichergröße zu:

$$D_{ges} = T_{Messzeitraum} * f_{Messungen} * D_{Messwert}$$

Bei zwölf Messungen pro Stunde ergeben sich 288 Messungen pro Tag, was über einen Gesamtzeitraum von zwei Wochen 4032 Messungen entspricht. Durch die Aufnahme von sieben Messgrößen und der Notwendigkeit eines zusätzlichen Zeitstempels ist ein entsprechend großer Speicher erforderlich. Bei einer angenommenen Datenbreite von 16 Byte pro Datensatz ergibt sich ein theoretischer Speicherbedarf von rund 65 kByte.

Zur weiteren Optimierung kann der Speicherbedarf durch verschiedene Verfahren reduziert werden. Eine Möglichkeit besteht in der differenziellen Speicherung, bei der nur Änderungen gegenüber dem vorherigen Messwert abgelegt werden. Dadurch lassen sich bei langsam variierenden Messgrößen deutliche Einsparungen erzielen. Auch der Einsatz zyklischer Ringspeicher ist vorteilhaft, wenn nur aktuelle Messdaten benötigt werden, während ältere Daten überschrieben werden können. Alternativ kann eine komprimierte Ablage in Flash- oder SD-Speicher vorgenommen werden, wobei verlustfreie Algorithmen wie Run-Length Encoding oder Huffman-Codierung zur Anwendung kommen können. Diese Maßnahmen tragen dazu bei, den verfügbaren Speicherplatz effizient zu nutzen, ohne die Datenintegrität zu beeinträchtigen.

Darüber hinaus ist die Speicherung von Statusinformationen und Prüfdaten, zum Beispiel Prüf- per Checksummen, für die Verlässlichkeit des Gesamtsystems von Bedeutung. Statusdaten können beispielsweise den Betriebszustand des Sensors, die Gültigkeit einer Messung oder Kommunikationsfehler kennzeichnen. Eine Prüfsumme ermöglicht die Erkennung von Übertragungs- oder Speicherfehlern und trägt somit zur Datensicherheit bei. Der zusätzliche Speicheraufwand bleibt dabei vernachlässigbar, da Statusinformationen typischerweise nur wenige Bits erfordern und Prüfsummen effizient berechnet werden können.

3 Methoden und Arbeitsweise

3.1 Systemaufbau



Abbildung 3.1: Messsystem aufgestellt zwischen zwei Bienenvölkern ©I.Emmerich

Das Messsystem soll Wetterfest zwischen den Beuten aufgebaut werden.

3.1.1 Mikroprozessor

A Die Vorgabe des Mikroprozessors ist einer auf STM32 Basis. Für das bloße Ansteuern von Sensoren, dem Speichern der Daten und der Ausgabe dieser ist nicht viel Rechenleistung nötig.

Es soll ein fertiges Devboard verwendet werden, um die Entwicklung zu vereinfachen. Dieses benötigt genug Verbindungsmöglichkeiten zum Anschluss von drei Temperatursensoren, dem Ameisensäuresensor sowie die Ansteuerung der restlichen Peripherie. Ein vergleichsweise großer integrierter Speicher ist wünschenswert.

Durch die kompakte Größe und gute Verfügbarkeit viel die Wahl auf den STM32L432KC als Nucleo Devboard. Dieser besitzt einen 12 bit ADC, mehr als genug GPIO Pins, welche dazu noch kompatibel mit dem Arduino nano Pinout sind, und 256kB Flash Speicher.

Die Ein- und Ausgänge des Microcontrollers sind nur 3,3V statt wie beim Arduino 5V tolerant, was bei der Auswahl der weiteren Hardware beachtet werden muss.

3.1.2 Temperatur- und Feuchtesensoren

Die meisten kommerziell erhältlichen integrierten Temperatur- und Feuchtesensoren decken den zu erwarteten Messbereich ab. Durch die Platzierung der Sensoren in bis zu einem Meter Entfernung in verschiedene Richtungen ist eine Singlewire Lösung die beste Variante. Wären die Sensoren dichter zusammen hätte eine Bustopologie, zum Beispiel I²C, den Vorteil.

Die Wahl der Sensoren fiel am Ende auf den DHT11. Dieser kann nur im Bereich von 0-50°C und 20-80% Luftfeuchte mit maximal 1Hz Abtastrate messen. Dies sind aber alles keine Einschränkungen, da die Temperatur und Feuchtigkeitswerte, welche deutlich seltener als 1 Hz verwendet werden, nicht zu schnell schwanken. Diese Sensoren sind außerdem weit verbreitet und können für die Prototypen einfach per Breadboard angeschlossen werden.

Die Genauigkeit der Sensoren reicht ~~auch~~ aus, da vor allem die Korrelation des Temperaturverlaufs zur Stockluftkonzentration aufgezeichnet werden soll.

3.1.3 Ameisensäuresensor

Allgemeine Informationen

Die Wahl des Ameisensäuresensor (ASS) gestaltete sich als schwierig. AS ist ein Flüchtige Organische Verbindung (volatile organic compound, VOC), aber nicht ~~der Einzige~~ welcher ~~in~~ in Bienenstöcken auftritt. Die VOC kommen aus vielen Quellen. Dadurch kann nicht gesagt werden welche Bestandteile alle in der SL vorkommen. Dadurch ist es wichtig dass der Sensor spezifisch auf Organische Säuren getrimmt ist.

Der gewählte Sensor ist der Acetic Acid Gas Sensor Acid von Membrapor. Die einzigen Gase welche einen starken Einfluss auf das Messergebnis haben sind Chlor und Stickstoffdioxid [11]. Da diese Gase aber nicht in außreichender Konzentration in Luft vorkommen ist dies kein Grund den Sensor nicht zu nutzen.

Anschluss und Funktionsweise

Das Potential des Sensors baut sich erst eine Weile nach dem Anschluss der Energieversorgung und damit dem Beginn der Messung auf. Bei den Tests ohne System zum Verstehen der Funktionsweise ergab sich, dass nach circa 90 Sekunden das vollständige Potential erreicht ist. Nach einer Minute sind über 90% des Vollausschlages erreicht. Wenn man die Messung noch weiter verzögert sinkt das Potential kurz danach wieder. Der Sensor sitzt bei allen Messungen auf einer Transmitter Board, welche den Stromausgang des Sensors in eine Spannung von 40-200mV Umwandelt. Wichtig dabei ist, dass der Nullpunkt

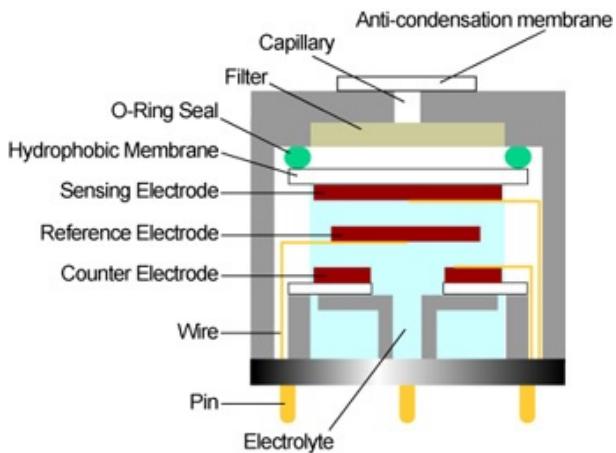


Abbildung 3.2: Aufbau eines Membrapor Gassensors ©Membrapor AG

und der Maximalausschlag des Sensors mithilfe der Potentiometer auf Platine korrekt eingestellt wird. Über an dem Sensor befestigte Schläuche kann dieser Geziehlt mit dem zu untersuchenden Gas versorgt werden. *Bei L1 Sensor + Schläuche*

Nach jeder Messung muss der Pegel des Sensors erst einmal wieder auf den Nullpunkt absinken. Dies kann durch das Überströmen mit Frischluft beschleunigt werden. Damit wird auch die Lebensdauer des Sensors erhöht. Der Pegel sinkt dabei erst einmal unter den Nullpunkt und nähert sich diesem dann über mehrere Minuten wieder an. Dabei ist es egal ob dieser mit Energie versorgt wird oder nicht. Die Gesamtdauer von Start der Messung bis zum Wiedererreichen des Nullpunktes legt direkt die Maximale Messfrequenz des Systemteils fest. Eine Zeit von unter 25 Minuten lies sich nicht erreichen. Deswegen wurde die Messung der Ameisensäurekonzentration auf einmal pro halbe Stunde festgelegt. Die Messfrequenz der anderen Systemteile bleiben davon unberührt.

3.1.4 Kommunikation mit dem Messsystem

Das Messsystem kann per USB Port über UART2 überwacht werden. Über diesen werden zu jeder Messung die aktuellen Messwerte geschickt. Ein Empfangen von Daten ist hier

nicht berücksichtigt. Der Port ist zum Debuggen von Fehlern im Programm gedacht. Die Kommunikation erfolgt UART1 per Extrplatine. Diese enthält auch einen USB Isolator, welche den USB-C Port galvanisch vom Messsystem trennt. Beim Verbinden mit dieser Platine wird kein Reset an den STM32 gesendet, was bei dem anderen USB Port passiert, weswegen sich auch für diese Variante entschieden wurde. So kann das Messsystem weiterhin per USB mit Strom versorgt werden, ohne dass das Programm bei jedem Verbindungsaufbau abgebrochen wird. Um die Daten zu Speichern muss an UART1 das Kommando `ßave` geschickt werden. Zurück kommt dann die bestätigung, welches kommando erkannt wurde und danach der dump der Daten in Chronologischer Reihenfolge in folgendem Format:

Darstellung als Tabelle mit Größen
timestamp, t in s, t1, t2, t3, h1, h2, h3, adc, sb0, sb1, sb2, sb3

Hardware Hintergrund
erklärt (welches Time)

Dabei ist **timestamp**, **t in s** die Zeit in Millisekunden per HAL_GetTick(), beziehungsweise in Sekunden nach Messbeginn, **t**, **h** die Temperatur und Luftfeuchtewerte, **adc** der ADC Ausgang auf 12 Bit genau, gemittelt über 32 Samples und **sb** die Statusbits zum Beschreiben des Systemzustands.

Extra Abschiff über die ADC
Extra für wenn die Peripherie
Durch diese Formatierung und das Senden des Headers kann die Ausgabe per Shell Skript direkt in einer CSV gespeichert werden. Per Tabellenkalkulationsprogramm ist dann eine leichte Weiterverarbeitung der Werte möglich.

3.2 Energiebedarf

3.2.1 weitere Peripherie

Zum Betrieb des gesamten Systems sind noch ein paar weitere Komponenten nötig. Der ASS muss mit ungefähr einem Liter Luft die Minute gespült werden, damit er die Elektroden

richtig funktionieren. Dafür kann eine einfache Membranpumpe verwendet werden.

Zum zu und abschalten der SL zum Sensor wird ein Ventil verwendet welches per Servo angesteuert wird. Da die Luft zum Sensor mit AL im Verhältnis 1 zu 1 gemischt wird, kann durch blocken der SL der Sensor mit normaler Atmosphäre gespült werden, da bei permanentem Aussetzen mit AS die Elektroden deutlich schneller degrieren.

hoch abhängig!

3.2.2 Versorgungsspannung

Der STM32 kann entweder per 3,3V, 5V oder 7-12V über die Pins versorgt werden. Per USB nutzt dieser nur 5V. Bei Versorgungsspannungen unter 5V ist dabei der ST-Link debugger nicht versorgt.

Der ASS benötigen eine Spannung von mindestens 12V. Diese Spannung muss möglichst konstant gehalten werden, da die Ausgangsspannung des Sensors direkt von der Eingangsspannung abhängig ist. Es wurde sich deswegen entschieden diese über ein Separates Netzteil zu erzeugen. Dies kann auch über einen USB PD-Adapter geschehen.

Der Energiebedarf des Gesamtsystems ergibt sich aus der Summe der Leistungsaufnahmen aller Komponenten im aktiven und im Ruhezustand. Für den stationären Betrieb über eine Powerbank ist es notwendig, die durchschnittliche Leistungsaufnahme möglichst gering zu halten, um eine mehrtägige Betriebsdauer zu erreichen.

3.2.3 Abschätzen des zu erwartenden Energieverbrauchs

Um eine Abschätzung der benötigten Elektrischen Energie im Akku zu erreichen wurden zunächst die Typischen Leistungsaufnahmen der Komponenten gemessen.

Versorgung	A (mA)	P(mW)
3,3V (Pin)	2,39	7,9
5V (Pin)	4,3	21,5
5V (USB)	20	100

in welchen
Betriebszustand?

Tabelle 3.1: STM32 Energiebedarf

V	A [V]	I [mA]	P[mW]	Dutycycle [%]	P_{cont} [mW]
Pumpe	5	100	500	100	500
ASS bei 0ppm	12	4	48	5	2,4
ASS bei 100ppm	12	20	240	5	12

Tabelle 3.2: Energiebedarf ASS und Pumpe

Der ASS besitzt einen Energieverbrauch welcher linear zum Messwert ist. In den weiteren Berechnungen wurde deswegen mit dem Worst-Case-Szenario gerechnet, der Energiebedarf bei 100ppm AS konzentration. Dabei fällt auf, dass die Pumpe mit Abstand der größte Verbraucher ist. Der Rest der Hardware benötigt nicht einmal 10% der Energie.

Bei einem durchschnittlichen Energiebedarf von rund 550 mW, einer Spannungswandlereffizienz von 80% ergibt sich ein Gesamtbbedarf von 225,33 Wh für eine Laufzeit von 14 Tagen. Dies ist nicht mit kleinen tragbaren Powerbanks zu erfüllen. Eine 20000mAh Powerbank hätte nur eine Laufzeit von etwas unter 100 Stunden. Über eine Autobatterie ist eine Laufzeit von 2 Wochen aber realisierbar.

3.2.4 Optimierungsmaßnahmen

Zur weiteren Reduzierung des Energieverbrauchs können verschiedene Strategien umgesetzt werden. Der Mikrocontroller unterstützt mehrere Energiesparmodi, in denen Taktfrequenz und Peripherie abgeschaltet oder in einen Low-Power-Zustand versetzt werden. Auch die

Sensoren können während der inaktiven Phasen stromlos geschaltet werden. Die Software übernimmt dabei das sequentielle Aktivieren der Komponenten und gewährleistet, dass nur die für die jeweilige Messung erforderlichen Module eingeschaltet sind.

Eine zusätzliche Energieeinsparung kann durch den Verzicht auf dauerhaft aktive Anzeigen oder Kommunikationsschnittstellen erzielt werden. Weiterhin sollte die Stromversorgung der Pumpe und des Ventils über Treiberschaltungen mit möglichst geringem Ruhestrom realisiert werden. Diese Maßnahmen führen in Summe zu einer deutlichen Verlängerung der Betriebsdauer und erhöhen gleichzeitig die Zuverlässigkeit des Systems im Langzeiteinsatz.

Zählerweise?

3.2.5 Zusammenfassung

Die durchgeführte Analyse des Energiebedarfs zeigt, dass die Pumpe und das Ventil die mit Abstand größten Verbraucher darstellen, während Mikrocontroller und Sensorik nur einen geringen Beitrag zur Gesamtleistung leisten. Durch den intermittierenden Betrieb und den gezielten Einsatz von Energiesparmodi ist der Dauerbetrieb des Systems über mehrere Wochen mit einer handelsüblichen Powerbank realisierbar. Die Ergebnisse verdeutlichen die Bedeutung einer energieeffizienten Systemarchitektur insbesondere bei mobilen Messsystemen, die autark und wartungsarm betrieben werden sollen.

noch Ablauf?

3.3 Programmaufbau

Das Messprogramm ist strategisch in mehrere Dateien aufgeteilt. Diese sind per Header-Dateien verlinkt. Die wichtigsten Features werden kurz beschrieben.

3.3.1 Main Loop

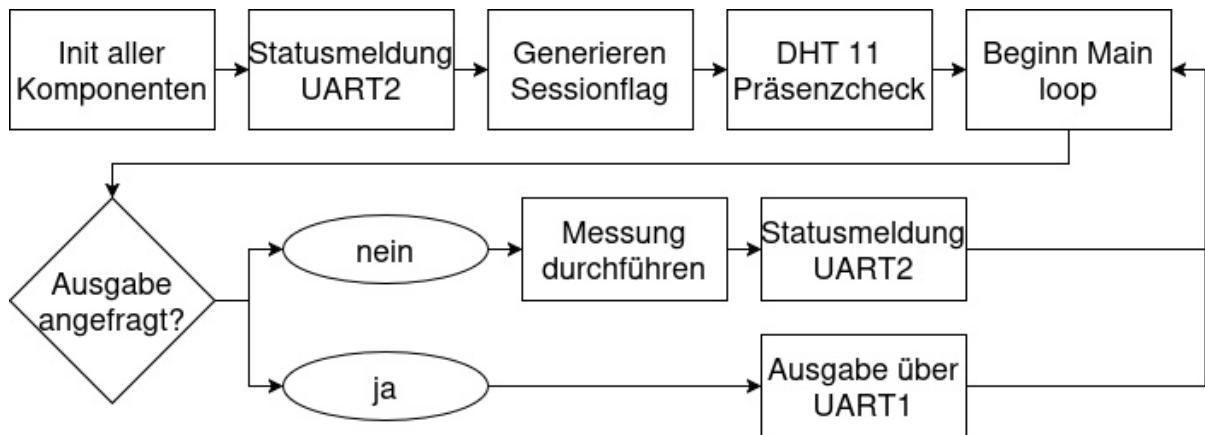


Abbildung 3.3: Main Loop

Die Maindatei startet mit der Initialisierung der Peripherie. Testweise wird per `printf()`, welches an UART2 umgeleitet wird, eine Statusmeldung des Erfolgreichen Programmstarts gesendet. An UART1 geschieht das gleiche, allerdings per `HAL_UART_Transmit`. Nachdem evtl. die Sessionflag generiert wurde und ein Präsenzcheck der DHT Sensoren durchgeführt wurde beginnt der Main Loop. Dieser kontrolliert zunächst, ob die Ausgabe der Messwerte angefragt wurde. Falls dies der Fall ist gibt er diese komplett aus. Falls nicht startet der denkt Messloop.

3.3.2 Messablauf

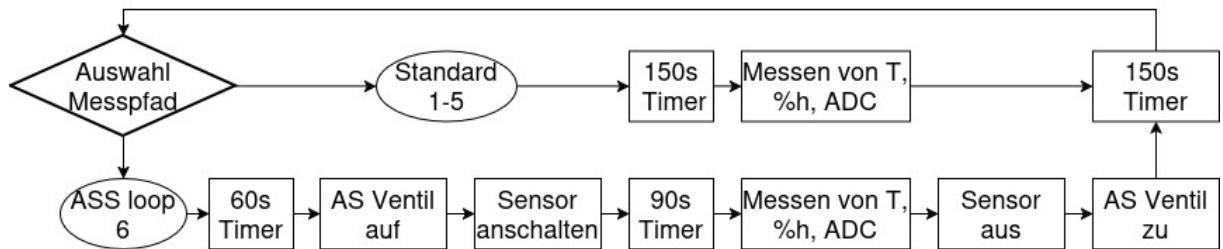


Abbildung 3.4: Messablauf

Die Messreihenfolge startet wenn aktuell keine Ausgabe über UART erfolgt. Dabei werden fünf mal hintereinander nur die Werte der Sensoren abgerufen. Bei jedem sechsten Durchlauf wird zunächst die Stockluftzufuhr geöffnet, damit der Sensor mit Ameisensäuregas durchströmt und nach 90 Sekunden dann die Messwerte aufgenommen. Der sechste Loop wird durch das flippen von zwei der Statusbits kenntlich gemacht. Danach wird das Ventil wieder geschlossen, der Sensor Stromlos geschalten und der gesamte Ablauf von vorne gestartet. Jeder Umlauf soll circa fünf Minuten in Anspruch nehmen, sodass die Gaskonzentration jede halbe Stunde gemessen werden kann.

3.3.3 Datenspeicherung

Die Daten werden in 12 Byte Blöcken gespeichert. Dabei ist jeder der Einträge folgendermaßen Aufgebaut:

Byte	Bit	Wert
0-3	0-7	Timestamp ms
4-6	0-7	Temperaturwerte (je 8 Bit)
7-9	0-7	Feuchtigkeitswerte (je 8 Bit)
10,11	0-3	ADC Wert (12 Bit)
11	4-7	4 Statusbits

Tabelle 3.3: Aufbau der Speicherblöcke

Da der verfügbare Speicher deutlich größer ist als die zu erwartende Datenmenge, wurde nur ein Teil, ~~der 128kB~~, des Speichers für die Ablage der Daten freigegeben. Der mehr als ausreichende Speicher war auch die Entscheidung die Daten nicht zu komprimieren ~~Auf Lö~~ A sonder Roh in den Speicher geschrieben. Der ADC hat eine Native ~~Bandbreite~~ von 12 Bit, welche ebenfalls unverarbeitet verwendet werden. Die 4 verbleibenden Bit sind als Statusbits reserviert.

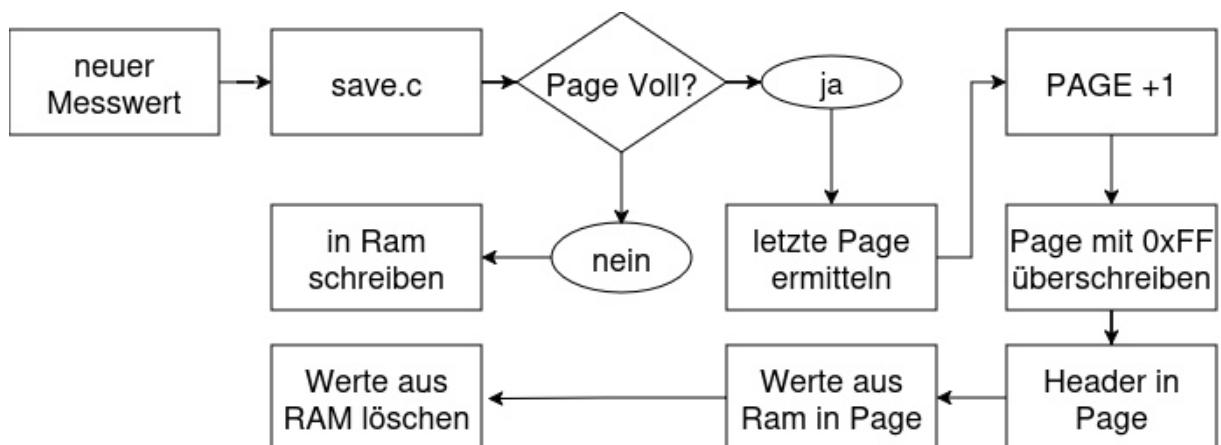


Abbildung 3.5: Speicherlogik

Die Messwerte werden zunächst im RAM abgelegt. Wenn 2kB an Daten voll sind werden diese dann als komplette Seite in den Speicher abgelegt. Nach 170 Messungen ist dies erreicht. Die letzten 8 Bit werden bei schreiben für die Sessionflag, Page Magic (0xA55A) und vier Leerbytes beschrieben. Mit ihnen wird eine Gültige Page beschrieben. Bei 64 zur Verfügung Stehenden Seiten ergibt dies bis zu 10880 Messungen welche am Stück gespeichert werden können. Da das System für eine Messung aller 5 Minuten ausgelegt ist kann so fast 38 Tage am Stück gemessen werden.

3.3.4 Datenausgabe vom Controller

Wenn über den USB-Isolator ein Gerät an die UART1-Schnittstelle angeschlossen wird, werden alle empfangenen Zeichen unmittelbar zurückgesendet, sodass die Eingabe im Terminal sichtbar bleibt. Wird das Kommando „**save**“ empfangen, startet die Datenausgabe. Bei einer fehlerhaften Eingabe wird die Meldung „**unbekanntes Kommando: <gesendetes Kommando>**“ zurückgegeben.

Nach erfolgreicher Anforderung der Ausgabe wird zunächst der aktuelle RAM-Puffer in den Flash-Speicher geschrieben. Anschließend werden der Header und die erste gültige Page ermittelt. Aus diesen Informationen wird die Gesamtanzahl der gespeicherten Datensätze bestimmt und als Meldung über UART1 ausgegeben.

Danach startet eine Ausgabeschleife, welche seitenweise alle Datensätze überträgt. Dabei wird fortlaufend geprüft, ob

- die aktuelle Page zur aktiven Messreihe gehört
- noch nicht vollständig ausgegeben wurde und
- gültige Daten enthält.

Fließ text

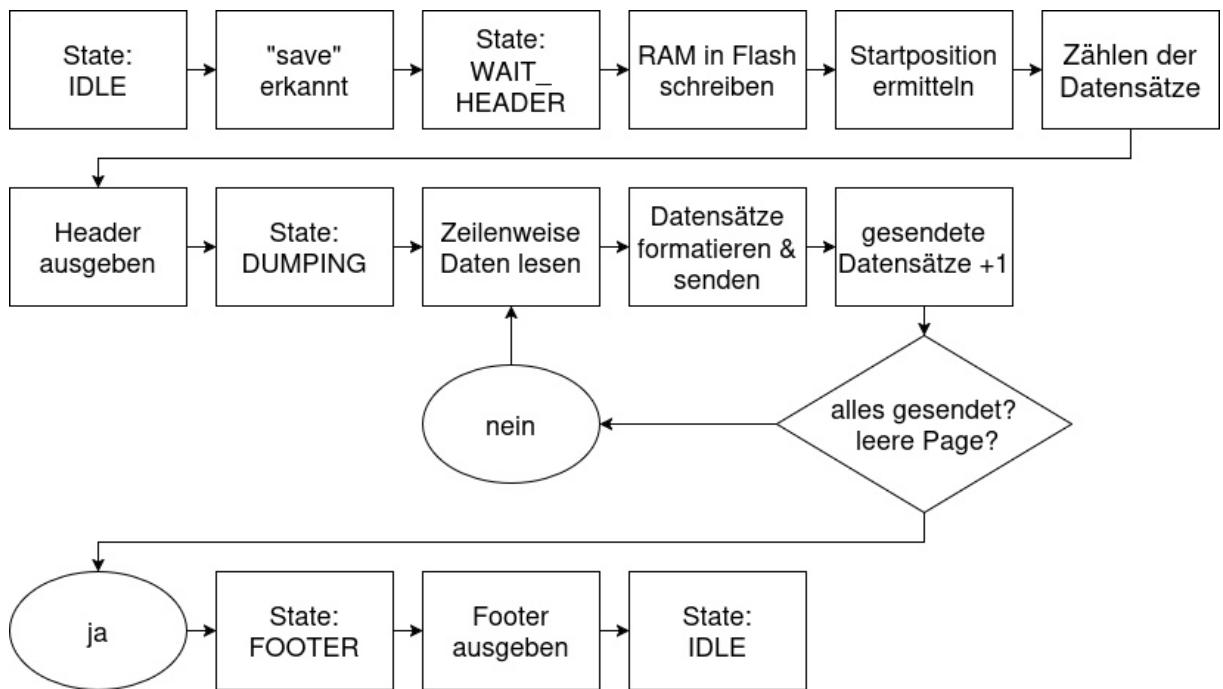


Abbildung 3.6: State machine Aufbau
als State Diagram darstellen

Für jeden Datensatz wird der Zeitstempel zusätzlich in Sekunden umgerechnet und alle Messwerte als dezimale Zahlen (ohne Einheit), zeilenweise über UART1 ausgegeben. Sind alle Datensätze übertragen oder tritt eine der Abbruchbedingungen auf, wird abschließend ein Footer gesendet und die Übertragung beendet.

Die gesamte Datenausgabe erfolgt über eine Zustandsmaschine nicht-blockierend per DMA, sodass der Controller währenddessen auch weiterhin Messen kann.

3.3.5 Datenausgabe am PC

Zur Verbindung mit einem MessPC steht ein USB-C Port zur Verfügung, über welchen die Daten per UART übertragen werden können. Die Datenübertragung funktioniert wie in Abschnitt 3.3.4 beschrieben. Durch ein Bash Skript lässt sich dies komplett automatisieren.

```
savve
Unbekanntes Kommando: "savve"
save
Kommando erkannt: save
Datensaetze im Speicher: 6
--- CSV Dump ---
timestamp,t in s,t1,t2,t3,h1,h2,h3,adc,sb0,sb1,sb2,sb3
8681,9,21,21,21,51,51,52,576,0,0,0,1
18761,19,21,21,21,51,50,52,581,0,0,0,1
28841,29,21,21,21,50,50,52,573,0,0,0,1
38921,39,21,21,21,50,50,52,579,0,0,0,1
49001,49,21,22,21,50,49,52,579,0,0,0,1
55761,56,21,22,21,50,49,51,579,0,0,1,0
--- Ende ---
```

Abbildung 3.7: Beispieldausgabe

Dieses Verbindet sich mit dem STM32 über UART, sendet den Speicherkommand und wartet die Zeile `-- CSV Dump --` ab. Wenn diese Empfangen wurde, beginnt es in eine neue CSV Datei die Werte Zeilenweise zu schreiben und die Daten im Terminal anzuzeigen, bis der Endstring erkannt wurde. Nach der Meldung der erfolgreichen Datenübertragung wartet das Terminal darauf geschlossen zu werden, damit der Benutzer erkennen kann, das das Skript fertig ist. Die geschriebene Datei kann dann direkt geöffnet und weiterverarbeitet werden.

4 Ergebnisse und Validierung

4.1 funktionalität Messystem

Das Messsystem ist in der Lage die Messdaten im Gewählten Zeitraum zu erfassen, auszugeben und zu speichern. Dabei kann in der aktuellen Version die Konzentration der Ameisensäure nur jede halbe Stunde festgestellt werden. Dies ist eine Sensorbeschränkung.

4.2 Datenspeicherung und Ausgabe

Die Datenspeicherung und Ausgabe wurde ohne den Ameisensäuresensor Validiert. Gesichert wurde an dieser Stelle das Rauschen des ADC. Dabei werden auch die Statusbits so gesetzt, dass dies in den Werten ersichtlich ist.

4.3 Bestimmen der Validierungspunkte

Es ist außerordentlich wichtig, die Funktionalität des Gesamtsystems zu validieren. Dabei soll beurteilt werden, ob die Funktionsweise den Anforderungen an das System entspricht.

Die Überprüfung der erfassten Mweeswerte mit den Realen Werten ist auch zu überprüfen. Dabei kann Hard- und Software teilweise einzeln betrachtet werden. Wenn nicht anders angegeben wurden folgende Messmittel zur Bestimmung der Werte genutzt:

Multimeter	UNI-T UT139B
Ameisensäure	Ameisensäure 60 % ad us. vet. Serumwerk Bernburg
Pipette	Gilson Pipetman P20
Versuchskiste	9,3l Lock&Lock box

Tabelle 4.1: Verwendete Messmittel

4.4 Validierung des Ameisensäuresensor

Um den Sensor korrekt einstellen zu können wurde folgender Messaufbau realisiert:

Für die Nullpunkteinstellung des Sensors wurde dieser mit der Membranpumpe mit normaler Frischluft Überströmt. Für die Validierung der Ameisensäurekonzentrationen muss zunächst das Testvolumen ermittelt werden. Dies wurde mit einem Messzylinder durch Abmessen mit Wasser bestimmt und ergab 9,3 l. Zum Einstellen der Sensoren kann nun eine bekannte Konzentration AS in eine Urschale pipettiert werden. Für eine Zielkonzentration nach der Verdampfung von 100 ppm werden 5,6 μ L Ameisensäure benötigt.

Nullpunkt eisheit ist nicht Teil
der Validierung sondern des Aufbaus

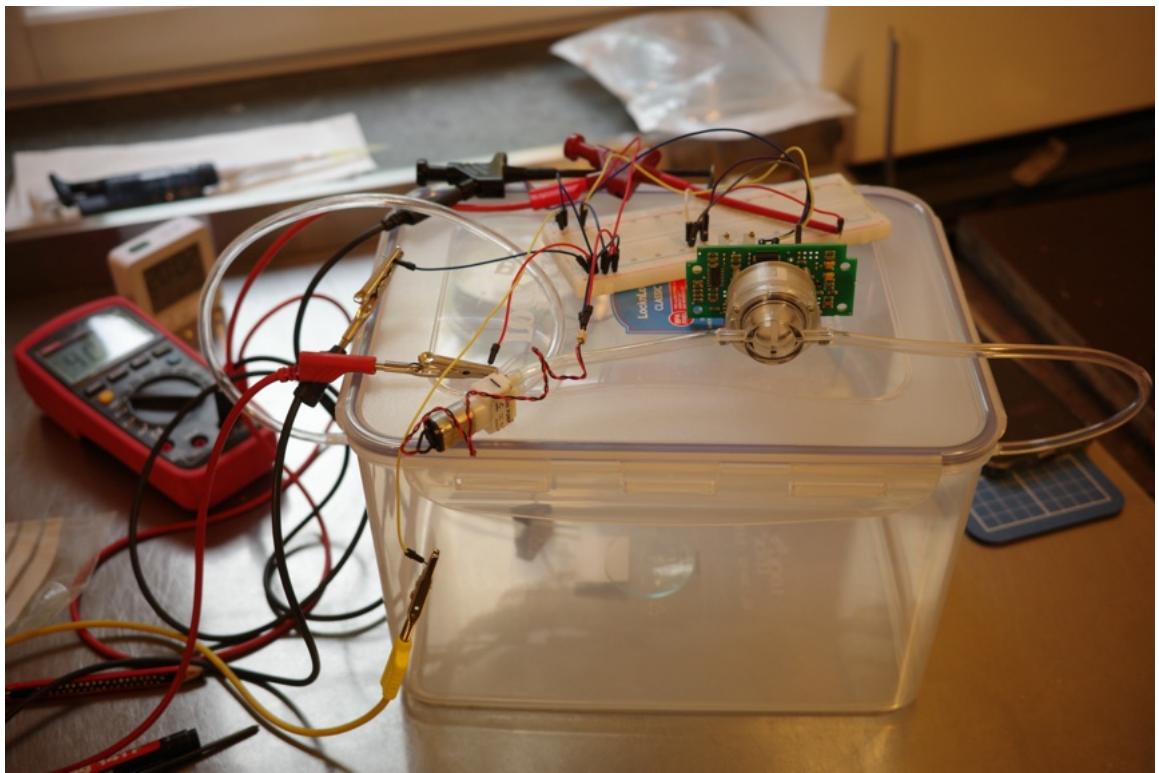


Abbildung 4.1: Validierungsaufbau Ameisensäuresensor ©J. Emmerich

→ schematisches Aufbau als Blockschaltbild

Zum Einstellen der Sensoren wurde dabei die Spannung am Ausgang der Platine per Multimeter gemessen. Zur Abstimmung der Sensoren untereinander wurden diese per Schlauch in Reihe geschalten, sodass das Gas beide Sensoren gleich durchströmt. Da ein Teil der AS bei der Messung verloren geht wurde die Flussrichtung auch in der anderen Reihenfolge gemessen.

4.5 Validierung der Temperatur- und Feuchtesensoren

Die verwendeten Temperatur und Feuchtesensoren sind nicht sehr exakt. Durch die Auswahl von drei der Sensoren mit der geringsten Abweichung untereinander kann die Abweichung zwischen den Messstellen verringert werden.

Beim Testen des Gesamtsystems viel auf, dass die Feuchtigkeitssensoren nach dem Beginn der Verdunstung der Ameisensäure sehr stark ausschlagen. Rein rechnerisch darf in der Testbox bei Applizierung der AS die Luftfeuchte durch den Wasseranteil nur um rund 1,4% steigen, stieg aber um circa 30%. Da aber die Sensoren in den Bienenstöcken immer der Ameisensäure exponiert sind können die Feuchtwerte dieser während der Behandlung nicht verwendet werden.

4.6 Validierung der Datenverarbeitung

Um die Datenspeicherung zu Überprüfen wurde das Messintervall auf eine Sekunde gesetzt. Damit wurde der Speicher mehrfach beschrieben und die Daten danach ausgegeben. Die Werte

Was muss validiert werden? Messwerte stimmen richtige Reihenfolge

Wie erfasse ich das? unterpunkte, was ist interessant zu messen, veränderung? Ist das Messbar? Luftfeuchte -> Fazit? nicht lösbar -> Andere Sensoren? TEXT + MESSAUFBAU
Bis Multimetertyp Lüftertyp soviel infos wie möglich

Referenzvolumen Nach Abschluss der Messkette Konzentration stimmt überein Zeitlicher Zusammenhang korrekt

5 Diskussion und Ausblick

5.1 Diskussion des aktuellen Standes

Es konnten beim Bau des Messsystems nicht alle anforderungen erfüllt werden. Einer der Hauptgründe dafür ist, dass der Ameisensäuresensor die Messfrequenz von einem Datenpunkt alle 5 Minuten nicht erreichen kann. Die anderen Sensoren und die Datenspeicherung erfüllen die Ansprüche aber komplett.

Da der Ameisensäuresensor noch zusätzlich eine Pumpe und ein Ventil benötigt und er selber ~~reicht~~ viel elektrische Energie Benötig ist aktuell kein Batteriebetrieb über die vollen zwei Wochen möglich.

5.2 Ausblick

Die weiter Suche nach anderen ASS sollte weitergeführt werden. Hier sollten Vergleichsmessungen unter realen Bedingungen mit dem Aktuellen Membrapor Sensor durchgeführt werden, um eventuell energiesparendere und oder schneller Messende Sensoren zu finden, um einen Batteriebetrieb realistischer zu gestalten. Allgemein soll je geschaut werden, inwiefern ~~die~~ alle Hardware im Bienenstock mit der Säurehaltigen Atmosphäre zurecht kommt, hier sind langzeittests vonnöten.

Die Datenkommunikation kann auch noch ausgebaut werden. Erst einmal ist es denkbar mehr Daten über den UART zu übertragen, unter anderem eine Erweiterung des Funktionsraumens. Eine Ausgabe des Zeitstempels, des belegten Speichers oder auch der die option alle bisher gemessenen Werte welche noch im Ram sind in den Speicher zu schreiben

währen einige Möglichkeiten die Funktion zu erweitern.

In einer komplett überarbeiteten Version wäre auch die Verwendung eines anderen Mikroprozessors, zum Beispiel mit Wlan unterstützung denkbar, sodass dieser die Daten direkt auf einem Server ablegt oder direkt einen Webserver betreibt sodass die Interaktion mit dem Messsystem noch weiter begrenzt werden kann.

Abbildungsverzeichnis

Abbildung 3.1: Messsystem aufgestellt zwischen zwei Bienenvölkern ©I.Emmerich	18
Abbildung 3.2: Aufbau eines Membrapor Gassensors ©Membrapor AG	21
Abbildung 3.3: Main Loop	26
Abbildung 3.4: Messablauf	27
Abbildung 3.5: Speicherlogik	28
Abbildung 3.6: State machine Aufbau	30
Abbildung 3.7: Beispieldaten	31
Abbildung 4.1: Validierungsaufbau Ameisensäuresensor ©J. Emmerich	34

Tabellenverzeichnis

Tabelle 2.1: Übersicht der USB Standards bezogen auf Spannung und Sturomstärke	15
Tabelle 3.1: STM32 Energiebedarf	24
Tabelle 3.2: Energiebedarf ASS und Pumpe	24
Tabelle 3.3: Aufbau der Speicherblöcke	28
Tabelle 4.1: Verwendete Messmittel	33

Literatur

- [1] Peter Rosenkranz, Pia Aumeier und Bettina Ziegelmann. „Biology and control of Varroa destructor“. In: *Journal of invertebrate pathology* 103 Suppl 1 (Sep. 2009), S96–119. DOI: [10.1016/j.jip.2009.07.016](https://doi.org/10.1016/j.jip.2009.07.016).
- [2] European Medicines Agency / Veterinary Medicines. *Formicpro 68,2 g imprägnierte Streifen für den Bienenstock für Honigbienen*. Zugriff am 29. September 2025. 2025. URL: <https://medicines.health.europa.eu/veterinary/de/600000046666>.
- [3] Bundesinstitut für Arzneimittel und Medizinprodukte (BfArM). *Monographien zu Bienentherapeutika*. Arzneimittelzulassung / Monografie. Zugriff am 6. Oktober 2025. Bundesinstitut für Arzneimittel und Medizinprodukte, 2021. URL: https://www.bfarm.de/SharedDocs/Downloads/DE/Arzneimittel/Zulassung/ZulRelThemen/stdZul_und_Reg/sz_monografien_bienentherapeutika_2021.pdf.
- [4] Leipziger Institut für Meteorologie - LIM/ Falk Kaiser. *LIM - Klimastatistik - Monatsstatistik*. Zugriff am 03. Oktober 2025. 2025. URL: https://wetterdaten.meteo.uni-leipzig.de/stat_verg_mon.php.
- [5] Anton Stabentheiner u. a. „Coping with the cold and fighting the heat: thermal homeostasis of a superorganism, the honeybee colony“. In: *Journal of Comparative Physiology A* 207 (Feb. 2021), S. 1–15. DOI: [10.1007/s00359-021-01464-8](https://doi.org/10.1007/s00359-021-01464-8).
- [6] Iris Eouzan u. a. „Hygroregulation, a key ability for eusocial insects: Native Western European honeybees as a case study“. In: *PLoS ONE* 14.2 (2019), e0200048. DOI: [10.1371/journal.pone.0200048](https://doi.org/10.1371/journal.pone.0200048), URL: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0200048>.
- [7] Anker Innovations Limited. *Why Is the Rated Capacity Marked on a Power Bank Lower Than Expected?* abgerufen am 08 October 2025. 2023. URL: <https://service.anker.com/article-description/Why-Is-the-Rated-Capacity-Marked-on-a-Power-Bank-Lower-Than-Expected> (besucht am 08. 10. 2025).
- [8] Elektronik-Kompendium. *USB Power Delivery (USB PD)*. abgerufen am 14 October 2025. 2023. URL: <https://www.elektronik-kompendium.de/sites/com/1809251.htm> (besucht am 08. 10. 2025).
- [9] Membrapor. *4-20 mA Transmitter Board Electrochemical Gas Sensor*. abgerufen am 25 Oktober 2025. 2021. URL: <https://www.membrapor.ch/sheet/Gas-Sensor-Transmitter-Compact.pdf>.
- [10] Rohde & Schwarz. *UART verstehen*. Online verfügbar bei Rohde & Schwarz. 2023. URL: https://www.rohde-schwarz.com/de/produkte/messtechnik/essentials-test-equipment/digital-oscilloscopes/uart-verstehen_254524.html (besucht am 26. 10. 2025).

- [11] Membrapor AG. *Organic Acids Gas Sensor Acid/C-100: Performance Data Sheet*. Rev.: Aug-24, 6 Seiten. Aug. 2024. URL: <https://www.membrapor.ch/sheet/Organic-Acids-Gas-Sensor-Acid-C-100.pdf>
-

Anhang

Anhangsverzeichnis

Abbildungs- und Tabellenverzeichnis im Anhang	42
A Anhangskapitel	43

Abbildungs- und Tabellenverzeichnis im Anhang

Abbildungen im Anhang

Abbildung A.1: Bildunterschrift im Anhang	43
---	----

Tabellen im Anhang

Tabelle A.1: Tabellenüberschrift im Anhang	43
--	----

A Anhangskapitel

Abbildung A.1: Bildunterschrift im Anhang

Tabelle A.1: Tabellenüberschrift im Anhang

