



*Universität der Bundeswehr München*

**Institut für**  
**Technische Informatik**

Bachelorarbeit

**Entwicklung und Umsetzung eines  
kostengünstigen Systems zur  
Erfassung, Verarbeitung und  
Bereitstellung von  
Wasserqualitätsdaten in der  
Aquaponik**

Bearbeiter: Johannes Schmelz

Abgabedatum: 30.03.2026

Aufgabenstellung: Prof. Klaus Buchenrieder, Ph.D.

Zweitprüfer: Prof. Dr. Wolfgang Hommel

Betreuung: Dr. rer. nat. Heike Rolfs



# Inhaltsverzeichnis

|                                                           |            |
|-----------------------------------------------------------|------------|
| <b>Abbildungsverzeichnis</b>                              | <b>vii</b> |
| <b>1. Einleitung</b>                                      | <b>1</b>   |
| 1.1. Problemstellung . . . . .                            | 1          |
| 1.2. Zielsetzung der Arbeit . . . . .                     | 2          |
| 1.3. Aufbau der Arbeit . . . . .                          | 2          |
| <b>2. Grundlagen</b>                                      | <b>5</b>   |
| 2.1. Aquaponik und Systemzusammenhänge . . . . .          | 5          |
| 2.1.1. Temperatur . . . . .                               | 6          |
| 2.1.2. pH-Wert . . . . .                                  | 7          |
| 2.1.3. Leitfähigkeit und Total Dissolved Solids . . . . . | 7          |
| 2.2. Messtechnische Grundlagen . . . . .                  | 8          |
| 2.2.1. Temperatursensoren . . . . .                       | 8          |
| 2.2.2. pH-Sensoren . . . . .                              | 8          |
| 2.2.3. Leitfähigkeitssensoren . . . . .                   | 9          |
| 2.3. Signalverarbeitung und Datenerfassung . . . . .      | 9          |
| 2.3.1. Analoge Signale . . . . .                          | 9          |
| 2.3.2. Digitale Übertragung . . . . .                     | 9          |
| 2.3.3. Einfluss von Störungen und Kalibrierung . . . . .  | 10         |
| 2.4. Einordnung in den Anwendungskontext . . . . .        | 10         |
| <b>3. Systemkonzeption</b>                                | <b>11</b>  |
| 3.1. Anforderungsanalyse . . . . .                        | 12         |
| 3.1.1. Messgrößen . . . . .                               | 12         |
| 3.1.2. Technische Anforderungen . . . . .                 | 12         |
| 3.1.3. Robustheit . . . . .                               | 13         |
| 3.1.4. Kosten und Umsetzbarkeit . . . . .                 | 13         |
| 3.1.5. Vorgaben aus dem Gesamtprojekt . . . . .           | 13         |

## Inhaltsverzeichnis

|                                                                   |           |
|-------------------------------------------------------------------|-----------|
| 3.2. Systemarchitektur . . . . .                                  | 14        |
| 3.2.1. Signalfluss und Verarbeitung . . . . .                     | 15        |
| 3.2.2. Komponenten des Systems . . . . .                          | 15        |
| 3.3. Hardwareplattform . . . . .                                  | 16        |
| 3.3.1. Mikrocontrollerboard . . . . .                             | 16        |
| 3.3.2. Bedeutung für das Gesamtprojekt . . . . .                  | 16        |
| 3.4. Auswahl und konzeptionelle Einbindung der Sensoren . . . . . | 17        |
| 3.4.1. Temperatursensor . . . . .                                 | 17        |
| 3.4.2. TDS-Sensor . . . . .                                       | 18        |
| 3.4.3. pH-Sensor . . . . .                                        | 19        |
| 3.5. Analog-Digital-Converter . . . . .                           | 20        |
| 3.5.1. Rolle im System . . . . .                                  | 20        |
| 3.5.2. Grundlage der Spannungsberechnung . . . . .                | 20        |
| 3.6. Zusammenfassung . . . . .                                    | 20        |
| <b>4. Hardwareimplementierung</b>                                 | <b>21</b> |
| 4.1. Spannungsversorgung . . . . .                                | 21        |
| 4.2. Physische Verschaltung der Sensoren . . . . .                | 22        |
| 4.2.1. Temperatursensor DS18B20 . . . . .                         | 23        |
| 4.2.2. TDS-Sensor . . . . .                                       | 24        |
| 4.2.3. pH-Sensor . . . . .                                        | 25        |
| 4.3. ADC-Anbindung und Signalqualität . . . . .                   | 26        |
| 4.4. Hardwareseitige Signalfilterung . . . . .                    | 26        |
| 4.4.1. Funktionsprinzip . . . . .                                 | 27        |
| 4.4.2. Einsatz im System . . . . .                                | 27        |
| 4.4.3. Einfluss auf die Messung . . . . .                         | 27        |
| 4.5. Schaltplan und Dokumentation . . . . .                       | 28        |
| 4.6. Zusammenfassung . . . . .                                    | 28        |
| <b>5. Softwareimplementierung</b>                                 | <b>31</b> |
| 5.1. Softwarearchitektur . . . . .                                | 31        |
| 5.2. Zeitsteuerung und Messintervall . . . . .                    | 31        |
| 5.3. Analog-Digital-Wandlung . . . . .                            | 32        |
| 5.3.1. Konfiguration . . . . .                                    | 32        |
| 5.4. Signalfilterung . . . . .                                    | 32        |
| 5.4.1. Mehrfachabtastung . . . . .                                | 32        |

|                                               |           |
|-----------------------------------------------|-----------|
| 5.4.2. Trimmed Mean Filter . . . . .          | 33        |
| 5.5. Temperaturmessung . . . . .              | 33        |
| 5.6. TDS-Verarbeitung . . . . .               | 34        |
| 5.7. pH-Berechnung und Kalibrierung . . . . . | 34        |
| 5.7.1. Spannungsberechnung . . . . .          | 34        |
| 5.7.2. 2-Punkt-Kalibrierung . . . . .         | 34        |
| 5.7.3. Temperaturkompensation . . . . .       | 35        |
| 5.7.4. Validierung der Messwerte . . . . .    | 35        |
| 5.8. Datenformat und Ausgabe . . . . .        | 36        |
| 5.9. Zusammenfassung . . . . .                | 36        |
| <b>6. Tests und Validierung</b>               | <b>37</b> |
| 6.1. Kalibrierung der Sensoren . . . . .      | 37        |
| 6.1.1. pH-Pufferlösungen . . . . .            | 37        |
| 6.1.2. Referenzmessungen . . . . .            | 38        |
| 6.2. Vergleich mit Referenzgeräten . . . . .  | 38        |
| 6.2.1. Versuchsaufbau . . . . .               | 38        |
| 6.2.2. Messdurchführung . . . . .             | 39        |
| 6.2.3. Ergebnisse . . . . .                   | 39        |
| 6.2.4. Bewertung . . . . .                    | 39        |
| 6.3. Zusammenfassung . . . . .                | 40        |
| <b>7. Fazit</b>                               | <b>41</b> |
| <b>A. Programmcode</b>                        | <b>43</b> |
| A.1. main.rs . . . . .                        | 43        |
| A.2. temperature.rs . . . . .                 | 46        |
| A.3. tds.rs . . . . .                         | 47        |
| A.4. ph.rs . . . . .                          | 47        |
| A.5. Cargo.toml . . . . .                     | 49        |
| <b>Literaturverzeichnis</b>                   | <b>51</b> |



# Abbildungsverzeichnis

|     |                                                           |    |
|-----|-----------------------------------------------------------|----|
| 1.  | Kreislauf eines Aquaponik Systems . . . . .               | 5  |
| 2.  | Gesamtaufbau des Systems . . . . .                        | 11 |
| 3.  | Gebauter Prototyp des Systems . . . . .                   | 14 |
| 4.  | Heltec WiFi LoRa 32 V3 . . . . .                          | 16 |
| 5.  | DS18B20 Temperatur-Sensor . . . . .                       | 17 |
| 6.  | Keystudio TDS-Sensor . . . . .                            | 18 |
| 7.  | Sportarc pH-Sensor . . . . .                              | 19 |
| 8.  | Verschaltung des DS18B20 . . . . .                        | 23 |
| 9.  | Verschaltung des TDS-Sensors . . . . .                    | 24 |
| 10. | Verschaltung des pH-Sensors mit Spannungsteiler . . . . . | 25 |
| 11. | Gesamtschaltplan des Systems . . . . .                    | 29 |





# Abkürzungsverzeichnis

**ADC** Analog-Digital-Converter

**ESP-IDF** Espressif IoT Development Framework

**GPIO** General-Purpose-Input/Output

**ppm** parts per million

**TDS** Total Dissolved Solids

**USB** Universal Serial Bus



# 1. Einleitung

Die weltweite Versorgung mit Nahrungsmitteln stellt insbesondere in Regionen mit begrenzten Ressourcen eine zentrale Herausforderung dar. Der Zugang zu proteinreichen Lebensmitteln wie Fisch sowie zu frischem Gemüse ist in vielen Teilen der Welt eingeschränkt. Aquaponik bietet hier einen Ansatz, beide Nahrungsquellen in einem gemeinsamen System zu erzeugen. Durch die Kombination von Fischzucht und Pflanzenanbau in einem geschlossenen Wasserkreislauf können Ressourcen effizient genutzt und der Bedarf an Wasser und Düngemitteln reduziert werden. [5, 9, 11]

Für den erfolgreichen Betrieb eines solchen Systems ist jedoch die Stabilität des biologischen Gleichgewichts entscheidend. Fische, Pflanzen und Mikroorganismen sind eng miteinander verbunden und reagieren empfindlich auf Veränderungen der Wasserqualität. Insbesondere Parameter wie Temperatur, pH-Wert und Nährstoffkonzentration müssen innerhalb geeigneter Grenzen gehalten werden. Werden diese nicht eingehalten, kann dies zu Wachstumsstörungen oder zum Ausfall des gesamten Systems führen. [9, 12]

In der Praxis erfolgt die Überwachung dieser Parameter häufig mithilfe technischer Messsysteme. Hohe Kosten solcher Systeme erschweren insbesondere den Einsatz von Aquaponiksystemen in weniger stark entwickelten Umgebungen, in denen kostengünstige Lösungen erforderlich sind. [9]

## 1.1. Problemstellung

Die kontinuierliche Überwachung der Wasserqualität ist eine grundlegende Voraussetzung für den stabilen Betrieb von Aquaponikanlagen. Nur wenn kritische Zustände frühzeitig erkannt werden, kann das Überleben der Fische sowie das Wachstum der Pflanzen langfristig sichergestellt werden. [9]

Das zentrale Problem besteht darin, dass geeignete Messsysteme zwar technisch verfügbar sind, jedoch häufig mit hohen Kosten verbunden sind. Dadurch sind sie für viele Anwender nicht praktikabel, insbesondere wenn diese Systeme in schlechter entwickelten Gegenden der Welt eingesetzt werden sollen [9]. Diese Systeme können auf Grund lokaler Gegebenheiten

## 1. Einleitung

oft nicht direkt mit dem Internet verbunden werden, weshalb eine Übertragung der Daten über eine Strecke von mehreren hundert Meter erforderlich ist [11].

Es ergibt sich somit die Notwendigkeit, ein System zu entwickeln, das die relevanten Wasserparameter zuverlässig erfasst, dabei jedoch deutlich kostengünstiger ist als bestehende Lösungen und die Übertragung der Daten über längere Strecken ermöglicht. Bezüglich des Problems der Übertragung wird parallel eine Arbeit entwickelt.

## 1.2. Zielsetzung der Arbeit

Ziel dieser Arbeit ist die Entwicklung und Umsetzung eines kostengünstigen Systems zur Erfassung, Verarbeitung und Bereitstellung von Wasserqualitätsdaten in der Aquaponik. Das im Rahmen dieser Arbeit entwickelte System ist Teil eines größeren Gesamtsystems zur Überwachung von Wasserqualitätsparametern. Während der Fokus dieser Arbeit auf der Erfassung und Verarbeitung der Sensordaten liegt, wird die drahtlose Übertragung der Daten über Funk in einer parallel entwickelten Arbeit umgesetzt.

Für die Erfassung und Bereitstellung der Wasserdaten wird ein mikrocontrollerbasiertes Messsystem entwickelt, das die zentralen Parameter Temperatur, pH-Wert sowie die elektrische Leitfähigkeit beziehungsweise den Total Dissolved Solids (TDS)-Wert erfasst. Die Arbeit umfasst sowohl die Auswahl geeigneter Sensoren als auch deren Integration in ein Gesamtsystem, sowie die Verarbeitung und Bereitstellung der Messdaten. [9, 12]

Darüber hinaus wird das entwickelte System hinsichtlich seiner Funktionalität, Messgenauigkeit und Wirtschaftlichkeit bewertet. Ziel ist es, eine Lösung zu finden, die mit geringem finanziellen Aufwand implementiert werden kann und dennoch eine zuverlässige Überwachung der Wasserqualität ermöglicht.

## 1.3. Aufbau der Arbeit

Die vorliegende Arbeit ist in mehrere Kapitel gegliedert, die schrittweise von den theoretischen Grundlagen bis zur praktischen Umsetzung des entwickelten Systems führen.

Zunächst werden in Abschnitt 2 die grundlegenden technischen und fachlichen Zusammenhänge erläutert. Dies umfasst die Funktionsweise von Aquaponiksystemen, die Bedeutung relevanter Wasserparameter sowie die messtechnischen Grundlagen zur Erfassung dieser Größen. Darüber hinaus werden zentrale Aspekte der mikrocontrollerbasierten Datenerfassung vorgestellt.

Aufbauend darauf wird in Abschnitt 3 die Systemkonzeption beschrieben. In diesem

Kapitel werden die Anforderungen an das System definiert und eine geeignete Systemarchitektur abgeleitet. Anschließend erfolgt die begründete Auswahl der eingesetzten Hardwarekomponenten unter Berücksichtigung technischer und wirtschaftlicher Bedingungen.

In Abschnitt 4 wird die praktische Hardwareimplementierung des Systems dargestellt. Der Fokus liegt hierbei auf der konkreten Verschaltung der Sensoren, der Spannungsversorgung sowie der Integration in den Mikrocontroller.

Die Softwareimplementierung wird in Abschnitt 5 behandelt. In diesem Kapitel wird die Erfassung, Verarbeitung und Aufbereitung der Sensordaten sowie die angewendeten Verfahren zur Verbesserung der Messqualität beschrieben.

Zur Bewertung des Systems erfolgt in Abschnitt 6 eine experimentelle Validierung. Dabei werden die Sensoren kalibriert und die Messergebnisse mit Referenzwerten verglichen. Auf dieser Grundlage werden Messabweichungen analysiert und die Leistungsfähigkeit des Systems bewertet.

Abschließend werden in Abschnitt 7 die wesentlichen Ergebnisse der Arbeit zusammengefasst und im Hinblick auf die Zielsetzung eingeordnet. Zudem werden mögliche Weiterentwicklungen des Systems sowie Ansätze für zukünftige Arbeiten diskutiert.



## 2. Grundlagen

Um aquaponische Anlagen zu verstehen, werden zuerst die theoretischen und technischen Grundlagen eines Systems zur Erfassung von Wasserqualitätsparametern in der Aquaponik benötigt. Der Fokus liegt dabei sowohl auf den biologischen und physikalischen Zusammenhängen innerhalb aquaponischer Systeme als auch auf den messtechnischen Prinzipien der eingesetzten Sensorik. Ziel ist es, die Auswahl der später verwendeten Komponenten nachvollziehbar zu begründen und deren Funktionsweise einzuordnen.

### 2.1. Aquaponik und Systemzusammenhänge

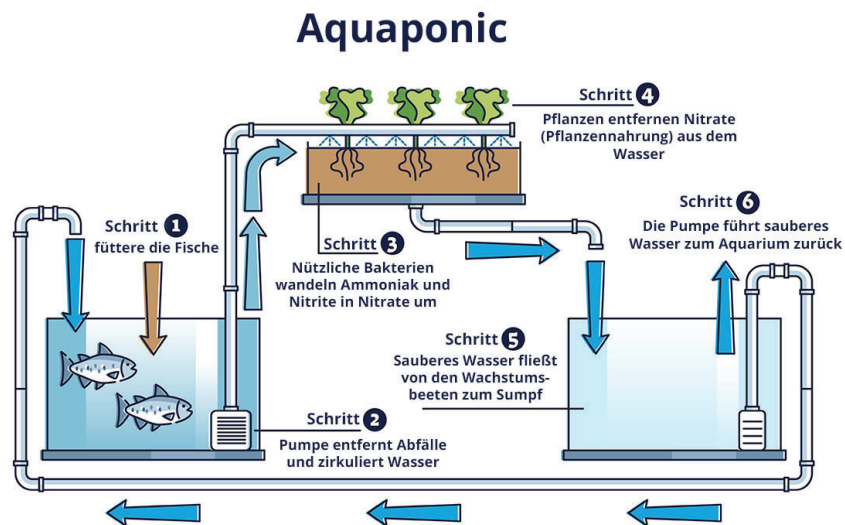


Abbildung 1.: Kreislauf eines Aquaponik Systems

Wie in Abbildung 1 [3] dargestellt, ist Aquaponik ein integriertes Produktionssystem. Hier werden Aquakultur (Fischzucht) und Hydroponik (Anbau von Pflanzen ohne Erde) kombiniert, dabei werden Fische in einem geschlossenen Wasserkreislauf gehalten, während Pflanzen ohne Erde in einem wasserbasierten Nährmedium wachsen. Die Kopplung beider

## 2. Grundlagen

Systeme ermöglicht eine effiziente Nutzung von Ressourcen, da Stoffwechselprodukte der Fische als Nährstoffe für die Pflanzen dienen [12, 14].

In einem aquaponischen System entsteht durch die Ausscheidungen der Fische unter anderem Ammoniak, welches für die meisten Organismen in höheren Konzentrationen toxisch ist. Durch die Aktivität nitrifizierender Bakterien wird Ammoniak zunächst in Nitrit und anschließend in Nitrat umgewandelt. Nitrat stellt eine wichtige Nährstoffquelle für Pflanzen dar und wird von diesen aufgenommen, wodurch das Wasser gleichzeitig gereinigt wird. [9]

Dieser biologische Kreislauf ist stark abhängig von stabilen Umweltbedingungen. Bereits geringe Veränderungen einzelner Parameter können zu einer Störung des Gleichgewichts führen. Beispielsweise kann ein ungünstiger pH-Wert die Aktivität der Bakterien reduzieren, wodurch sich toxische Zwischenprodukte ansammeln können. Ebenso beeinflusst die Temperatur sowohl die Stoffwechselrate der Fische als auch die Effizienz mikrobiologischer Prozesse. [5, 9] Solche Veränderungen der Wasserqualität werden oft durch falsche Futterzufuhr oder einer generellen Überlastung des Systems provoziert. Eine Behandlung von schlechter Wasserqualität kann über komplexe Filter oder chemische Reaktionen stattfinden. Ist dies zu aufwendig, kostspielig oder nicht umsetzbar muss ein Austausch des Wassers stattfinden. [14]

Aus diesem Grund ist die kontinuierliche Überwachung der Wasserqualität eine zentrale Voraussetzung für den stabilen Betrieb aquaponischer Systeme. Die Messung der Temperatur, des pH-Wertes und die Konzentration von gelösten Stoffen ist von zentraler Bedeutung, da diese Parameter bereits auf geringe Veränderungen im System reagieren und gleichzeitig vergleichsweise einfach erfassbar sind. Dementsprechend werden sie in der Literatur als essenzielle Größen für die Überwachung aquaponischer Systeme hervorgehoben [12].

Die Auswahl der zu messenden Parameter orientiert sich an deren Einfluss auf die Stabilität des Systems sowie an ihrer technischen Erfassbarkeit. [5]

### 2.1.1. Temperatur

Die Wassertemperatur (im Folgenden sofern nicht anders angegeben in Grad Celsius ( $^{\circ}\text{C}$ )) ist ein grundlegender Parameter in aquaponischen Systemen. Sie beeinflusst den Stoffwechsel der Fische, die Aktivität der Mikroorganismen sowie physikalische Eigenschaften des Wassers [5].

Eine Erhöhung der Temperatur führt in der Regel zu einer Beschleunigung biologischer Prozesse, wie die Arbeitsgeschwindigkeit von nitrifizierenden Bakterien. Gleichzeitig sinkt jedoch die Löslichkeit von Sauerstoff im Wasser, was insbesondere für die Fische auf



Grund von Unterversorgung bis hin zum Tod führen kann. Zu niedrige Temperaturen hingegen verlangsamen Stoffwechselprozesse und können das Wachstum von Pflanzen und Mikroorganismen beeinträchtigen. [5, 14]

Darüber hinaus hat die Temperatur einen direkten Einfluss auf andere Messgrößen. Insbesondere die elektrische Leitfähigkeit sowie pH-Messungen sind temperaturabhängig, wodurch eine isolierte Betrachtung einzelner Parameter nicht ausreichend ist. [1, 5, 9]

### 2.1.2. pH-Wert

Der pH-Wert beschreibt die Konzentration von Wasserstoffionen in einer Lösung und ist ein zentraler Parameter für chemische und biologische Prozesse im Wasser [15].

In aquaponischen Systemen beeinflusst der pH-Wert sowohl die Verfügbarkeit von Nährstoffen für Pflanzen als auch die Toxizität bestimmter Stickstoffverbindungen. Beispielsweise kann Ammoniak in Abhängigkeit vom pH-Wert in eine stärker toxische Form übergehen. [5]

Darüber hinaus ist der pH-Wert entscheidend für die Aktivität nitrifizierender Bakterien, die für die Umwandlung von Ammoniak in pflanzenverfügbare Nährstoffe verantwortlich sind. Abweichungen vom optimalen Bereich können daher zu einer Störung des gesamten Systems führen. [5]

Die Messung des pH-Wertes ist technisch anspruchsvoll, da die zugrunde liegende elektrochemische Spannung sehr klein ist und eine hohe Empfindlichkeit gegenüber Störungen aufweist [1, 9].

### 2.1.3. Leitfähigkeit und Total Dissolved Solids

Die elektrische Leitfähigkeit beschreibt die Fähigkeit des Wassers, elektrischen Strom zu leiten, und ist direkt abhängig von der Konzentration gelöster Ionen [1].

Der Total Dissolved Solids (TDS)-Wert stellt eine abgeleitete Größe dar, die als Maß für die Gesamtmenge gelöster Stoffe im Wasser verwendet wird. In vielen praktischen Anwendungen wird der TDS-Wert aus der Leitfähigkeit berechnet, wobei ein empirischer Umrechnungsfaktor verwendet wird [8, 16].

In aquaponischen Systemen liefert der TDS-Wert Hinweise auf die Nährstoffkonzentration sowie auf die Belastung des Wassers durch Stoffwechselprodukte. Eine zu hohe Konzentration kann auf eine unzureichende Filterleistung oder auf eine Überlastung des Systems hinweisen. [1, 16]

Es ist jedoch zu beachten, dass die Beziehung zwischen Leitfähigkeit und TDS von der

## 2. Grundlagen

Zusammensetzung der gelösten Stoffe sowie von der Temperatur abhängt, wodurch eine genaue Interpretation der Messwerte eine Kalibrierung erforderlich ist [1, 16].

## 2.2. Messtechnische Grundlagen

Die Erfassung der beschriebenen Parameter erfolgt über unterschiedliche Sensortypen, die jeweils spezifische physikalische Messprinzipien nutzen.

### 2.2.1. Temperatursensoren

Temperatursensoren basieren auf physikalischen Effekten, die temperaturabhängig sind, wie beispielsweise elektrische Widerstandsänderungen oder Halbleitereffekte. Die temperaturabhängige Veränderung der Leitfähigkeit des Messelementes wird intern erfasst und in ein entsprechendes elektrisches Signal umgesetzt. Digitale Temperatursensoren integrieren sowohl, das analoge Sensorelement, als auch die Signalverarbeitung in einem Bauteil. Dies ermöglicht die direkte Ausgabe digitaler Messwerte und reduziert die Anfälligkeit gegenüber Störungen, die bei analogen Signalen auftreten können. [15]

Ein weiterer Vorteil digitaler Temperatursensoren besteht darin, dass keine zusätzliche Analog-Digital-Wandlung erforderlich ist, wodurch sich die Integration in das System erleichtern lässt [1].

### 2.2.2. pH-Sensoren

pH-Sensoren basieren in der Regel auf elektrochemischen Messprinzipien. Dabei entsteht zwischen einer Referenzelektrode und einer Messelektrode eine Spannung, die proportional zur Wasserstoffionenkonzentration ist [15].

Diese Spannung liegt typischerweise im Bereich weniger hundert Millivolt und weist eine sehr hohe Ausgangsimpedanz auf. Daraus ergeben sich besondere Anforderungen an die Signalverarbeitung, da bereits geringe Störungen oder Leckströme zu erheblichen Messfehlern führen können [15].

Moderne pH-Sensoren verwenden häufig Glaselektroden, bestehend aus Referenzelektroden und Messelektroden mit Silberchloridüberzug, um eine stabile und reproduzierbare Messung zu gewährleisten [15].

### 2.2.3. Leitfähigkeitssensoren

Die Messung der Leitfähigkeit erfolgt über Elektroden, zwischen denen ein elektrischer Strom durch das Wasser geleitet wird. Der gemessene Strom ist abhängig vom elektrischen Widerstand des Wassers, welcher wiederum durch die Konzentration gelöster Ionen bestimmt wird. Eine erhöhte Konzentration von gelösten Ionen bedeutet einen Anstieg des Stroms. [15, 17]

Die gemessene Leitfähigkeit kann anschließend zur Berechnung des TDS-Wertes genutzt werden [1].

## 2.3. Signalverarbeitung und Datenerfassung

Die von den Sensoren erzeugten Signale müssen erfasst und weiterverarbeitet werden. Abhängig vom Sensortyp unterscheiden sich dabei sowohl die Signalform als auch die Art der Übertragung.

### 2.3.1. Analoge Signale

Analoge Signale liegen als kontinuierliche Spannungen vor, die proportional zur gemessenen physikalischen Größe sind. Aufgrund ihrer kontinuierlichen Natur sind sie besonders anfällig für Störungen, beispielsweise durch elektromagnetische Einflüsse oder Rauschen, und erfordern daher eine sorgfältige Signalaufbereitung. [15]

Für die weitere Verarbeitung im Mikrocontroller ist eine Analog-Digital-Wandlung notwendig, bei der die kontinuierliche Spannung in diskrete digitale Werte überführt wird. Die Genauigkeit dieser Umwandlung hängt maßgeblich von der Auflösung des Wandlers sowie von der Stabilität der Referenzspannung ab [1].

### 2.3.2. Digitale Übertragung

Im Gegensatz dazu werden digitale Signale bereits in Form diskreter Werte übertragen. Die Signalaufbereitung erfolgt dabei innerhalb des Sensors, wodurch die Übertragung weniger störanfällig ist. [6, 8]

Digitale Sensoren kommunizieren über standardisierte Schnittstellen wie beispielsweise OneWire. Dadurch können mehrere Sensoren mit geringem Verdrahtungsaufwand betrieben werden, während gleichzeitig eine robuste und zuverlässige Datenübertragung gewährleistet wird [1, 8].

### **2.3.3. Einfluss von Störungen und Kalibrierung**

In realen Anwendungen werden Messwerte durch verschiedene Faktoren beeinflusst, darunter elektrische Störungen, Temperaturschwankungen sowie Alterungseffekte der Sensoren [8].

Insbesondere bei Leitfähigkeits- und pH-Messungen ist eine Kalibrierung notwendig, um genaue Ergebnisse zu erzielen. Zudem müssen Wechselwirkungen zwischen den einzelnen Messgrößen berücksichtigt werden, wie die Temperaturabhängigkeit des pH-Wertes [1].

### **2.4. Einordnung in den Anwendungskontext**

Die Entwicklung eines kostengünstigen und robusten Messsystems ist insbesondere im Kontext praktischer Anwendungen von Bedeutung. Projekte wie INCiTIS-FOOD [11] zeigen, dass skalierbare und einfach einsetzbare Systeme benötigt werden, die unter unterschiedlichen Bedingungen zuverlässig arbeiten [2, 7].

Dabei steht nicht die maximale Messgenauigkeit im Vordergrund, sondern ein sinnvoller Kompromiss zwischen Genauigkeit, Kosten und Robustheit. Ziel ist es, Veränderungen der Wasserqualität frühzeitig zu erkennen und somit einen stabilen Betrieb des Systems zu garantieren.

### 3. Systemkonzeption

Abbildung 2 zeigt den grundlegenden Aufbau des Gesamtsystems. Dargestellt ist das Projekt in seiner vollständigen Struktur, wie es in dieser und der weiteren Arbeit behandelt wird. Die vorliegende Arbeit befasst sich dabei ausschließlich mit dem rot umrandeten Teilsystem. Dieses beinhaltet die Erfassung, Verarbeitung und Bereitstellung der Sensordaten, während die drahtlose Übertragung der Daten über Funk in einer separat entwickelten Arbeit umgesetzt wird. Durch diese Aufteilung ergibt sich eine modulare Systemarchitektur, bei der Sensordatenerfassung und Kommunikation voneinander getrennt betrachtet werden können.

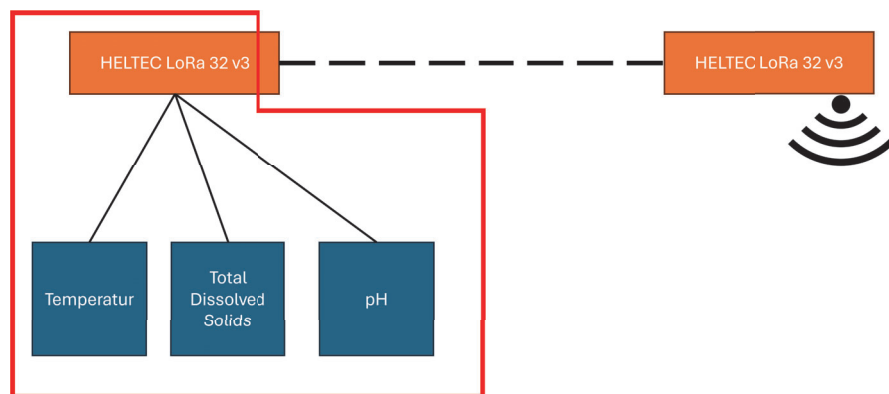


Abbildung 2.: Gesamtaufbau des Systems

Aufbauend auf den in Abschnitt 2 dargestellten Grundlagen wird eine konkrete Systemarchitektur abgeleitet und die Auswahl der eingesetzten Hardwarekomponenten begründet.

Ziel ist die Entwicklung eines kompakten, kostengünstigen und gleichzeitig ausreichend genauen Systems zur kontinuierlichen Messung zentraler Wasserparameter in aquaponischen Anlagen.

## 3.1. Anforderungsanalyse

Die Anforderungen an das System ergeben sich sowohl aus den physikalischen Gegebenheiten aquaponischer Systeme als auch aus praktischen und wirtschaftlichen Randbedingungen.

### 3.1.1. Messgrößen

Zentrale Aufgabe des Systems ist die Erfassung der Wasserparameter Temperatur, pH-Wert sowie der Konzentration gelöster Stoffe, welche in dieser Arbeit über den Total Dissolved Solids (TDS)-Wert abgebildet wird [8, 12]. Diese Größen sind maßgeblich für die Stabilität aquaponischer Systeme, reagieren schnell auf Veränderungen der Wasserqualität und sind leicht zu erheben, weshalb sie in der Literatur als essenzielle Parameter für die Überwachung und Steuerung hervorgehoben werden. [5, 9, 14]

### 3.1.2. Technische Anforderungen

Neben der Auswahl der Messgrößen ergibt sich eine zentrale technische Anforderung aus der Heterogenität der Sensorsignale. Während der Temperatursensor ein digitales Signal liefert, arbeiten sowohl der pH-Sensor als auch der TDS-Sensor mit analogen Ausgangsspannungen [1].

Das System muss daher in der Lage sein, beide Signalarten zu verarbeiten. Insbesondere für die analogen Sensoren ist die Integration eines Analog-Digital-Converter (ADC) erforderlich. Dieser muss eine ausreichende Auflösung bieten, um auch kleine Spannungsänderungen zuverlässig erfassen zu können. [8]

Ein weiterer wesentlicher Aspekt ist die Empfindlichkeit der analogen Signale gegenüber Störungen. Gerade bei der pH-Messung handelt es sich um ein vergleichsweise kleines und störanfälliges Signal, welches durch äußere Einflüsse wie elektromagnetische Störungen oder Kontaktprobleme beeinflusst werden kann. Daraus ergibt sich die Anforderung sowohl auf Hardware- als auch auf Softwareebene Maßnahmen zur Stabilisierung der Messwerte vorzusehen. [1]

Zusätzlich muss eine stabile Spannungsversorgung gewährleistet sein. Das System arbeitet mit einer internen 3,3 V-Versorgung für die Sensorik, während der Mikrocontroller über USB mit 5 V versorgt wird. Diese Spannungsstruktur erfordert eine saubere Referenzierung, insbesondere im Hinblick auf die analogen Messungen. [8]

### 3.1.3. Robustheit

Das System muss unter realen Einsatzbedingungen zuverlässig nutzbar sein. Dazu gehört insbesondere die Stabilität gegenüber elektrischen Störungen sowie mechanischen Einflüssen. [8, 11]

Da analoge Sensorsignale, insbesondere beim pH-Sensor, empfindlich gegenüber Rauschen und Störungen sind, müssen geeignete Maßnahmen zur Signalstabilisierung getroffen werden [1]. Diese umfassen sowohl eine saubere Verdrahtung mit kurzen Leitungswegen als auch hardwareseitige Filter in Form von RC-Tiefpässen.

Zusätzlich muss das System in der Lage sein, fehlerhafte oder unplausible Messwerte zu erkennen. Dies wird durch softwareseitige Plausibilitätsprüfungen (siehe Abschnitt 5.7.4) sowie durch Mittelwertbildung (siehe Abschnitt 5.4) erreicht.

Ein weiterer Aspekt der Robustheit ist die stabile Spannungsversorgung, da Schwankungen der Versorgungsspannung direkte Auswirkungen auf die Messgenauigkeit haben können.

### 3.1.4. Kosten und Umsetzbarkeit

Ein zentrales Ziel der Arbeit ist die Entwicklung eines kostengünstigen Systems, das mit leicht verfügbaren Komponenten realisiert werden kann. Die Auswahl der Sensoren erfolgte daher gezielt auf Basis preisgünstiger, am Markt weit verbreiteter Module. Maßgeblich war hierbei, dass die in den Datenblättern angegebenen Genauigkeiten für die vorgesehene Anwendung als ausreichend bewertet wurden. Unter diesen Bedingungen wurde jeweils die kostengünstigste verfügbare Variante gewählt.

Zusätzlich soll das System mit minimalem Hardwareaufwand realisiert werden. Dies bedeutet insbesondere den Verzicht auf externe Analog-Digital-Wandler oder komplexe Signalaufbereitungsschaltungen. Die Umsetzung erfolgt stattdessen mit den im Mikrocontroller integrierten Funktionen, wodurch sowohl Kosten als auch Komplexität reduziert werden. [1]

### 3.1.5. Vorgaben aus dem Gesamtprojekt

Die Auswahl des Mikrocontrollers erfolgte nicht frei, sondern war durch die parallel entwickelte Arbeit innerhalb des Gesamtprojekts vorgegeben. Dadurch ergibt sich eine zusätzliche Bedingung, da die Systemarchitektur an die vorhandene Plattform angepasst werden musste. Gleichzeitig bietet diese Vorgabe den Vorteil, dass eine direkte Integration

### 3. Systemkonzeption

in das Gesamtsystem möglich ist, insbesondere im Hinblick auf die spätere drahtlose Datenübertragung.

Zusammenfassend ergibt sich, dass das System die relevanten Wasserparameter zuverlässig erfasst, unterschiedliche Signalarten verarbeitet, mit kostengünstiger Hardware realisierbar sein und in eine modulare Gesamtarchitektur integrierbar sein muss.

## 3.2. Systemarchitektur

Das System basiert auf einer mikrocontrollergestützten Architektur, bei der alle Sensoren direkt an eine zentrale Verarbeitungseinheit angeschlossen sind. Die Verarbeitung der Messwerte erfolgt lokal im Mikrocontroller, wodurch eine kompakte und integrierte Systemstruktur entsteht.

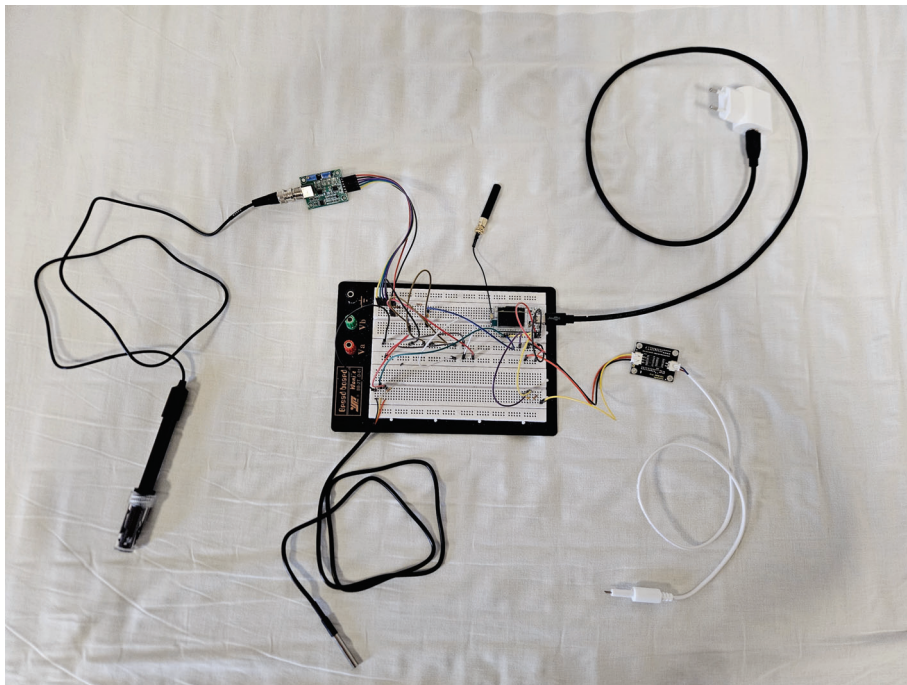


Abbildung 3.: Gebauter Prototyp des Systems

Abbildung 3 zeigt den realisierten Versuchsaufbau des Systems.

Im linken oberen Bereich befindet sich der pH-Sensor mit angeschlossener Elektrode, der über ein Sensormodul mit dem Mikrocontroller verbunden ist. Im linken unteren Bereich ist der Temperatursensor (DS18B20) zu sehen, der über eine einzelne Datenleitung angebunden ist.



Im rechten unteren Bereich ist der TDS-Sensor positioniert, welcher über ein analoges Ausgangssignal mit dem Mikrocontroller verbunden ist. Die Stromversorgung erfolgt im rechten oberen Bereich über ein Universal Serial Bus (USB)-Netzteil.

Im Zentrum des Aufbaus befindet sich der Mikrocontroller (Heltec WiFi LoRa 32 V3), der auf einem Steckbrett montiert ist. Die Verkabelung erfolgt über Jumper-Leitungen, wobei sowohl die analogen als auch die digitalen Sensorsignale direkt mit den entsprechenden General-Purpose-Input/Output (GPIO)-Pins verbunden sind.

#### 3.2.1. Signalfluss und Verarbeitung

Die Sensoren liefern dabei unterschiedliche Signaltypen, die jeweils spezifisch verarbeitet werden müssen. Der Temperatursensor überträgt seine Messwerte digital über das OneWire-Protokoll, wodurch eine robuste und vergleichsweise störungsarme Kommunikation möglich ist [6]. Im Gegensatz dazu liefern der pH-Sensor und der TDS-Sensor analoge Spannungen, die über den internen ADC des Mikrocontrollers digitalisiert werden [1].

Beim pH-Sensor ist zusätzlich eine Anpassung des Signals erforderlich, da die Ausgangsspannung außerhalb des zulässigen Eingangsbereichs des Mikrocontrollers liegen kann. Diese Anpassung erfolgt über einen Spannungsteiler, wodurch das Signal skaliert und gleichzeitig der ADC geschützt wird.

Die Kombination aus digitaler und analoger Signalverarbeitung stellt unterschiedliche Anforderungen an das System, ermöglicht jedoch gleichzeitig eine flexible und kostengünstige Umsetzung der Messaufgabe.

#### 3.2.2. Komponenten des Systems

Das System besteht aus folgenden Komponenten:

- Heltec WiFi LoRa 32 V3 <sup>1</sup>
- Temperatursensor AZDelivery DS18B20 <sup>2</sup>
- TDS-Sensor Keystudio KS0429 <sup>3</sup>
- pH-Sensor Sportarc <sup>4</sup>

Die genaue Auswahl wird im Folgenden erläutert.

---

<sup>1</sup><https://heltec.org/project/wifi-lora-32-v3>, 22.03.2026

<sup>2</sup><https://www.amazon.de/dp/B01MZG48OE>, 12.02.2026

<sup>3</sup><https://de.aliexpress.com/item/1005007694990805.html>, 10.02.2026

<sup>4</sup><https://de.aliexpress.com/item/1005009432150755.html>, 12.02.2026

## 3.3. Hardwareplattform

### 3.3.1. Mikrocontrollerboard

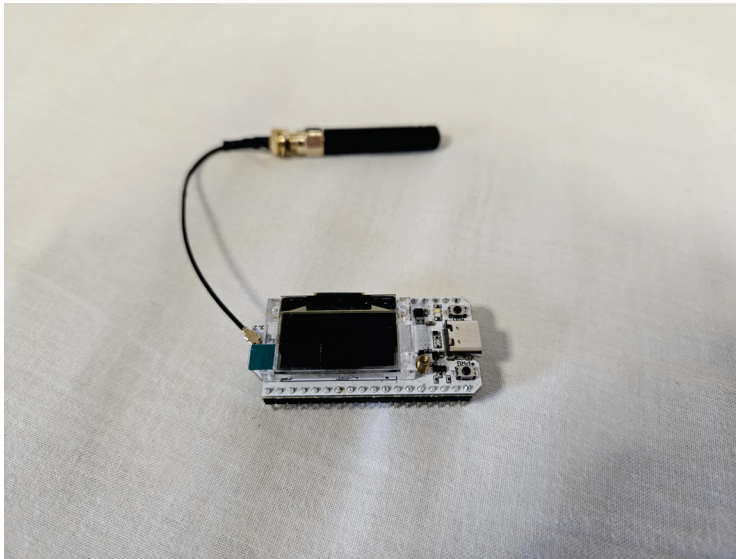


Abbildung 4.: Heltec WiFi LoRa 32 V3

Als zentrale Verarbeitungseinheit wird ein ESP32-S3 basiertes Mikrocontrollerboard (Heltec WiFi LoRa 32 V3), wie in Abbildung 4 gezeigt, eingesetzt. Dieses integriert einen Analog-Digital-Wandler mit 12 Bit Auflösung, digitale Schnittstellen sowie ein LoRa-Modul SX1262 [10].

### 3.3.2. Bedeutung für das Gesamtprojekt

Die Verwendung eines integrierten Systems reduziert die Komplexität und ermöglicht eine kompakte Bauweise [8]. Gleichzeitig ist die Wahl dieses Boards für das Gesamtprojekt von Bedeutung, da hierdurch die spätere Funkübertragung in der parallel entwickelten Arbeit direkt realisiert werden kann.

Die ADC-Auflösung von 12 Bit erlaubt eine Diskretisierung des Eingangssignals in 4096 Stufen, was insbesondere für die pH-Messung relevant ist [1, 8, 10].

## **3.4. Auswahl und konzeptionelle Einbindung der Sensoren**

### **3.4.1. Temperatursensor**



Abbildung 5.: DS18B20 Temperatur-Sensor

Der Temperatursensor DS18B20 (Abbildung 5) ist ein digitaler Temperatursensor, der über ein OneWire-Protokoll kommuniziert. Der Einsatz eines digitalen Sensors reduziert die Störanfälligkeit und vereinfacht die Integration, da keine analoge Signalverarbeitung erforderlich ist. Die Anbindung erfolgt digital, wodurch sich der Sensor gut für die vorgesehene Systemarchitektur eignet. [1, 6, 8]

### *3. Systemkonzeption*

#### **3.4.2. TDS-Sensor**

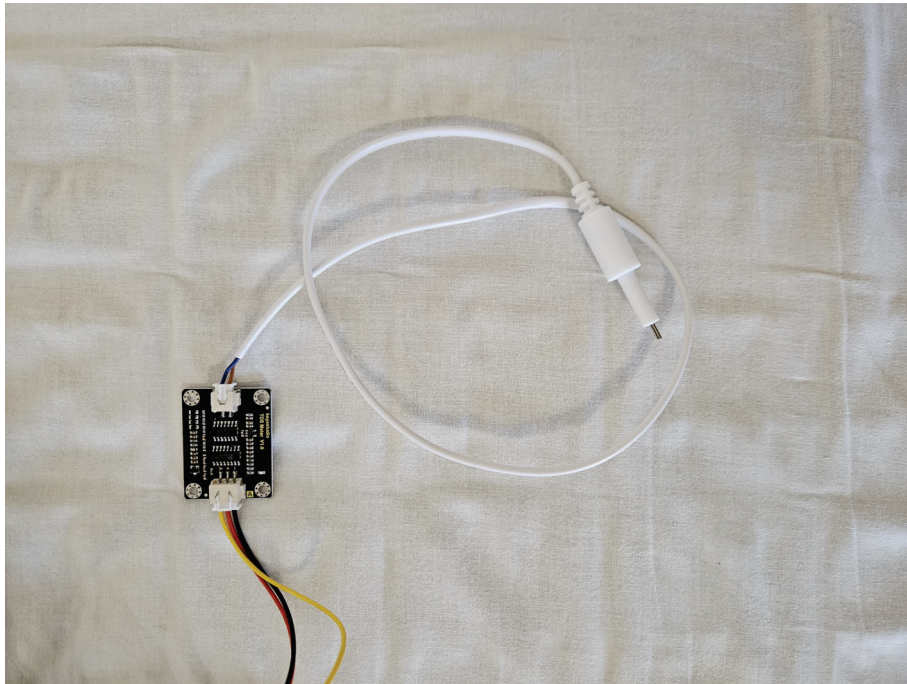


Abbildung 6.: Keystudio TDS-Sensor

Der verwendete TDS-Sensor (Abbildung 6) liefert ein analoges Ausgangssignal proportional zur Konzentration gelöster Stoffe [1]. Das analoge Signal wird über den internen ADC des Mikrocontrollers erfasst. Eine genaue Umrechnung in parts per million (ppm) erfolgt in dieser Implementierung nicht, da hierfür eine Kalibrierung erforderlich wäre. Näheres ist in Abschnitt 6 beschrieben. Damit wird der Sensor in dieser Arbeit primär zur relativen Bewertung der Konzentration gelöster Stoffe eingesetzt.



### 3.4.3. pH-Sensor

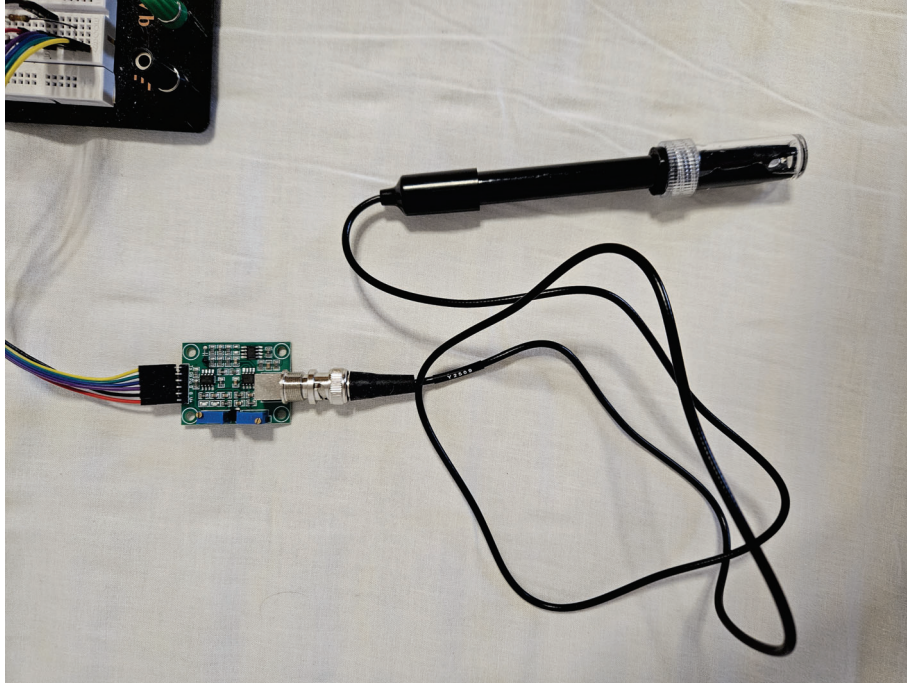


Abbildung 7.: Sportarc pH-Sensor

Der verwendete pH-Sensor (Abbildung 7) liefert ebenfalls ein analoges Spannungssignal. Da das Ausgangssignal über dem zulässigen ADC-Bereich liegen kann, ist eine Signalanpassung erforderlich. Diese erfolgt über einen Spannungsteiler. Nur der analoge Ausgang (PO) wird verwendet. Digitale Ausgänge bleiben ungenutzt, da diese in den durchgeführten Tests keine sinnvollen Werte lieferten und darüber hinaus kein Datenblatt des Sensors vorliegt.

Die Bestimmung des pH-Werts erfolgt nicht über feste Kennlinienparameter, sondern über eine Zweipunktkalibrierung mit Pufferlösungen, beispielsweise bei pH 7,00 und pH 4,00. Dabei werden die gemessenen Rohwerte des ADC den bekannten pH-Werten zugeordnet und anschließend linear interpoliert. Zusätzlich wird die Temperatur der Messlösung berücksichtigt, da die Elektrodenkennlinie temperaturabhängig ist [8, 15].

Allgemein ergibt sich für die lineare Zweipunktkalibrierung:

$$pH = pH_1 + \frac{x - x_1}{x_2 - x_1}(pH_2 - pH_1)$$

wobei  $x$  der aktuell gemessene Wert,  $x_1$  und  $x_2$  die durch das System gemessenen Kalibrierwerte und  $pH_1$ ,  $pH_2$  die zugehörigen Pufferwerte sind.

## 3.5. Analog-Digital-Converter

### 3.5.1. Rolle im System

Die analogen Sensoren werden konzeptionell über den internen Analog-Digital-Converter (ADC) des ESP32-S3 erfasst. Dieser stellt die zentrale Schnittstelle zwischen analoger Sensorik und digitaler Weiterverarbeitung dar.

### 3.5.2. Grundlage der Spannungsberechnung

Die Eingangsspannung wird folgendermaßen berechnet:

$$U = \frac{\text{ADC}}{4095} \cdot 3,3 \text{ V}$$

Die konkrete Konfiguration sowie die praktische Einbindung des ADC werden in Kapitel 4 beschrieben.

## 3.6. Zusammenfassung

Das entwickelte System kombiniert digitale und analoge Sensorik in einer gemeinsamen Architektur [1, 8]. Durch die direkte Anbindung der Sensoren an den Mikrocontroller kann auf zusätzliche externe ADCs oder komplexe Signalaufbereitung verzichtet werden, wodurch eine einfache und kostengünstige Systemstruktur realisiert wird.

Gleichzeitig ergeben sich insbesondere bei analogen Sensoren Einschränkungen hinsichtlich Messgenauigkeit und Kalibrierbarkeit [1]. Im Vergleich zu komplexeren Systemen aus der Literatur, die zusätzliche Hardware zur Signalaufbereitung einsetzen, stellt dieser Ansatz daher einen bewusst gewählten Kompromiss zwischen Aufwand, Kosten und Genauigkeit dar [8].

Insgesamt ermöglicht die gewählte Architektur die zuverlässige Erfassung zentraler Wasserparameter bei geringem Hardwareaufwand und bildet eine geeignete Grundlage für eine kompakte und erweiterbare Systemlösung. Die konkrete Umsetzung sowie praktische Herausforderungen werden im folgenden Kapitel beschrieben.

## 4. Hardwareimplementierung

Hier wird nun die physische Umsetzung des entwickelten Systems beschrieben. Der Fokus liegt auf der konkreten Verschaltung der Sensoren, der Spannungsversorgung sowie der Integration in den Mikrocontroller.

Ziel der Hardwareimplementierung ist eine möglichst einfache und gleichzeitig stabile Lösung zur Erfassung, Verarbeitung und Bereitstellung von Wasserqualitätsdaten.

Dieses System besteht aus:

- Heltec WiFi LoRa 32 V3
- Temperatursensor AZDelivery DS18B20 (digital)
- TDS-Sensor Keyestudio KS0429 (analog)
- pH-Sensor Sportarc (analog)

Die Kombination aus digitaler und analoger Sensorik stellt unterschiedliche Anforderungen an die Verschaltung und Signalverarbeitung.

### 4.1. Spannungsversorgung

Die Spannungsversorgung des Systems erfolgt über die Universal Serial Bus (USB)-Schnittstelle des Mikrocontrollers. Über USB wird eine Eingangsspannung von  $5\text{ V}$  bereitgestellt [10].

Die Sensoren müssen mit  $5\text{ V}$  sowie  $3,3\text{ V}$  versorgt werden, diese werden auf dem Mikrocontroller entsprechend umgewandelt [10]. Die internen Spannungsschienen des Mikrocontrollers dienen als Versorgung für alle angeschlossenen Sensoren. Dies reduziert die Komplexität der Schaltung und minimiert potenzielle Fehlerquellen.

#### *4. Hardwareimplementierung*

Alle Komponenten teilen sich eine gemeinsame Masseverbindung, wodurch stabile Referenzbedingungen für analoge Messungen geschaffen werden. Eine saubere Masseführung ist insbesondere bei empfindlichen Signalen, wie sie beim pH-Sensor auftreten, von großer Bedeutung.

### **4.2. Physische Verschaltung der Sensoren**

Die Sensoren sind an General-Purpose-Input/Output (GPIO)-Pins am ESP32-S3 angebunden. Die Pinbelegung ist wie folgt:

- GPIO2 → TDS-Sensor
- GPIO3 → pH-Sensor
- GPIO4 → DS18B20

Die Auswahl der verwendeten GPIO-Pins erfolgte unter Berücksichtigung der verfügbaren Hardwarefunktionen des ESP32-S3. Für die analogen Sensoren wurden GPIO2 und GPIO3 gewählt, da diese über den internen Analog-Digital-Converter (ADC) angebunden sind.

Der digitale Temperatursensor wurde an GPIO4 angeschlossen, da dieser Pin frei verfügbar ist und sich für die Implementierung des OneWire-Protokolls eignet. Da OneWire keine spezielle Hardwareunterstützung erfordert, kann die Kommunikation flexibel über geeignete GPIO-Pins realisiert werden.



### 4.2.1. Temperatursensor DS18B20

Der Temperatursensor DS18B20 wird über das OneWire-Protokoll angebunden [6]. Die Datenleitung ist mit GPIO4 verbunden und wird über einen Pull-Up-Widerstand von  $4,7\text{ k}\Omega$  auf  $3,3\text{ V}$  gezogen. Zusätzlich werden Versorgungsspannung und Masse angeschlossen.

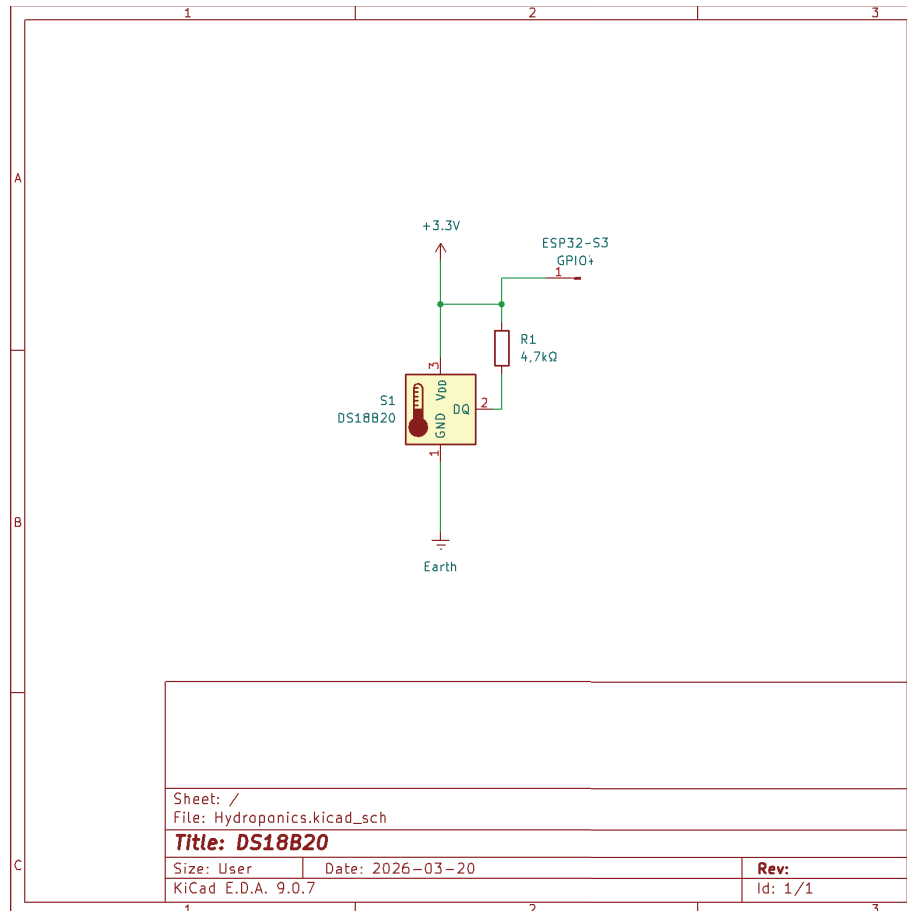


Abbildung 8.: Verschaltung des DS18B20

Durch die digitale Signalübertragung ist der Sensor relativ unempfindlich gegenüber Störungen. Dennoch wurde die Leitung möglichst kurz gehalten, um Signalprobleme zu vermeiden.

#### 4. Hardwareimplementierung

##### 4.2.2. TDS-Sensor

Der TDS-Sensor (Keyestudio KS0429) wird ebenfalls mit  $3,3\text{ V}$  betrieben und liefert ein analoges Ausgangssignal im Bereich von  $0$  bis etwa  $2,3\text{ V}$ . Dieses Signal kann direkt vom ADC des Mikrocontrollers erfasst werden. Der analoge Ausgang (AO) ist mit GPIO2 verbunden.

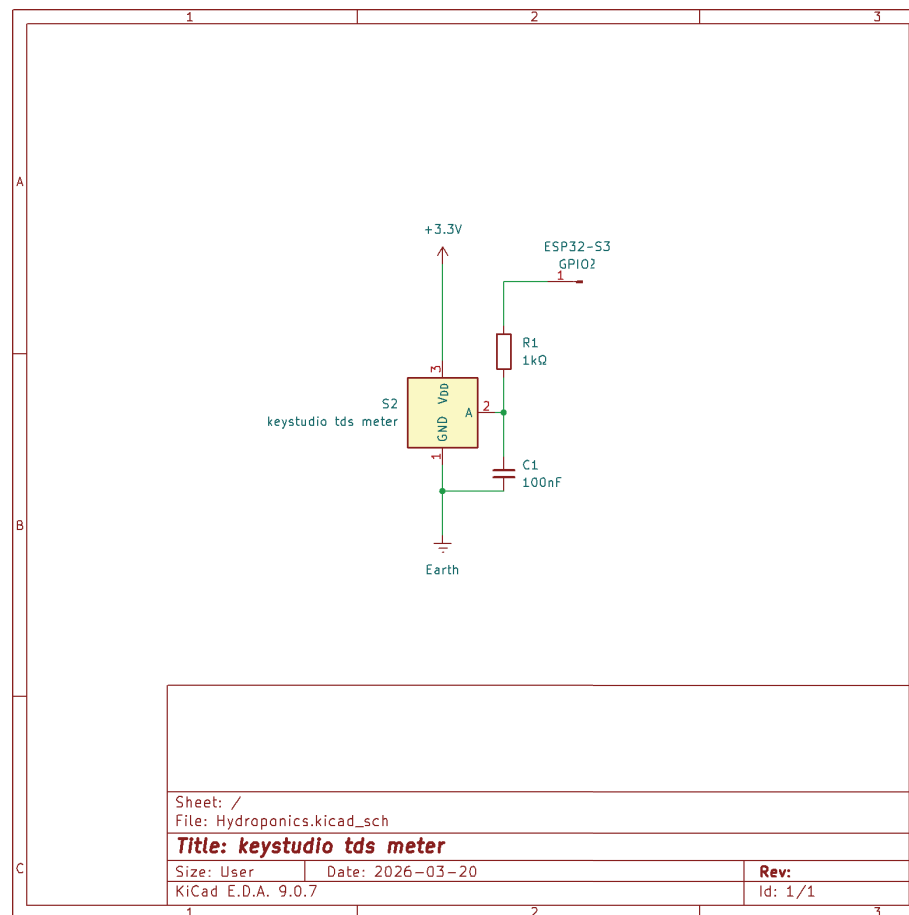


Abbildung 9.: Verschaltung des TDS-Sensors

In der praktischen Umsetzung zeigte sich, dass das Ausgangssignal des Sensors Schwankungen unterliegt. Diese sind sowohl auf das Messprinzip als auch auf elektrische Störungen zurückzuführen. Gegenmaßnahmen werden in Abschnitt 4.3 erläutert.

### 4.2.3. pH-Sensor

Der pH-Sensor wird mit 5 V versorgt und stellt ein analoges Signal über den Ausgang PO bereit. Dieses Signal ist aufgrund der hohen Impedanz besonders empfindlich gegenüber Störungen.

Zur Anpassung des Signals an den ADC sowie zum Schutz des Eingangs wird ein Spannungsteiler verwendet. Dieser besteht aus zwei Widerständen:

- $R2 = 12\text{ k}\Omega$
- $R3 = 20\text{ k}\Omega$

Die Wahl der Widerstandswerte stellt einen Kompromiss zwischen ausreichender Signalabsenkung und möglichst geringer Belastung des Sensorsignals dar. Gleichzeitig wird der Strom durch den Spannungsteiler gering gehalten.

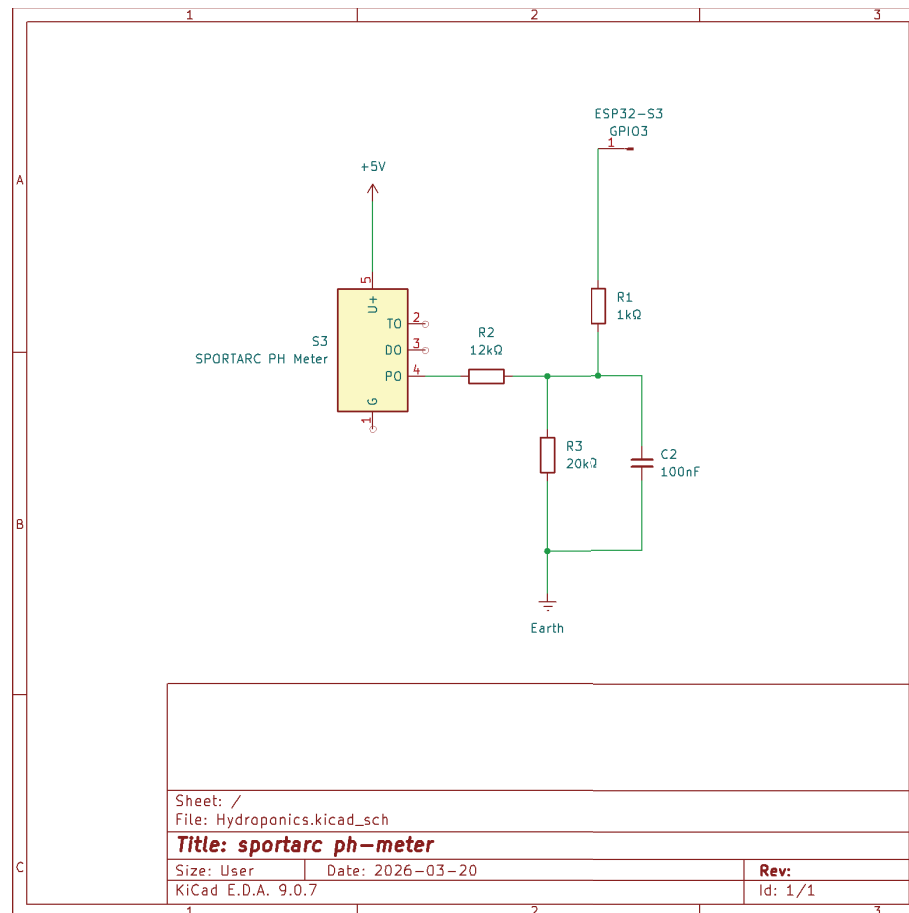


Abbildung 10.: Verschaltung des pH-Sensors mit Spannungsteiler

#### 4. Hardwareimplementierung

Die Spannung am ADC ergibt sich mit:

$$V_{ADC} = V_{PO} \cdot \frac{R3}{R2 + R3}$$

Zur Rückrechnung gilt:

$$V_{PO} = V_{ADC} \cdot \frac{R2 + R3}{R3}$$

Der Spannungsteiler dient dazu, mögliche Spannungsspitzen zu begrenzen und den ADC-Eingang zu schützen. Gleichzeitig trägt er zur Stabilisierung des Signals bei.

Aufgrund der hohen Empfindlichkeit des pH-Signals wurde besonderer Wert auf eine saubere Verdrahtung gelegt. Insbesondere lose Verbindungen oder lange Leitungen können zu erheblichen Messfehlern führen. [1]

### 4.3. ADC-Anbindung und Signalqualität

Die analogen Sensoren sind direkt an die ADC-Eingänge des ESP32-S3 angeschlossen.

Der ADC arbeitet mit einer Auflösung von 12 Bit und ermöglicht die Digitalisierung von Spannungen im Bereich von etwa 0 V bis 3,3 V. Die konkrete Konfiguration des ADC erfolgt mit einer Auflösung von 12 Bit, einem Wertebereich von 0 bis 4095, einer Referenzspannung von etwa 3,3 V und einer Attenuation von 12 dB. Die Auflösung von 12 Bit ermöglicht eine Spannungsauflösung im Millivoltbereich, wodurch auch kleine Änderungen des Sensorsignals, insbesondere beim pH-Sensor, erfasst werden können. [10]

In der praktischen Anwendung zeigte sich, dass die ADC-Werte durch Rauschen beeinflusst werden. Dies betrifft insbesondere analoge Sensoren. Um diese Effekte zu minimieren, wurden sowohl hardware- als auch softwareseitige Maßnahmen umgesetzt. Die hardwareseitigen Systeme werden im Folgenden beschrieben, während die softwareseitige Umsetzung in Abschnitt 5.4 beschrieben ist.

### 4.4. Hardwareseitige Signalfilterung

Neben der softwareseitigen Filterung wurde auch eine hardwareseitige Glättung der analogen Signale implementiert. Ziel ist die Reduktion von hochfrequentem Rauschen sowie die Stabilisierung der ADC-Eingangssignale.

Hierzu wurden einfache RC-Tiefpassfilter an den analogen Sensoreingängen eingesetzt. Diese bestehen aus einem Serienwiderstand und einem parallel zum Eingang geschalteten Kondensator gegen Masse. [15]

##### 4.4.1. Funktionsprinzip

Der RC-Tiefpass wirkt als frequenzabhängiger Filter, der schnelle Signaländerungen dämpft, während langsame Änderungen nahezu unverändert passieren.

Die Grenzfrequenz ergibt sich mit:

$$f_c = \frac{1}{2\pi RC}$$

Dabei gilt:

- $R$ : Serienwiderstand
- $C$ : Kondensator gegen Masse

Signale oberhalb dieser Frequenz werden abgeschwächt, wodurch hochfrequentes Rauschen reduziert wird [15].

##### 4.4.2. Einsatz im System

Die Filter wurden an den analogen Signalen des TDS-Sensors (GPIO2) sowie des pH-Sensors (GPIO3) eingesetzt. Dabei wurde der Tiefpass jeweils direkt vor den ADC-Eingang geschaltet, wodurch eine Glättung des Eingangssignals direkt vor der Digitalisierung erreicht wird. [15]

##### 4.4.3. Einfluss auf die Messung

Durch den Einsatz der Filter konnte eine deutliche Reduktion der Messwertschwankungen beobachtet werden. Insbesondere beim pH-Sensor führte dies zu stabileren und besser reproduzierbaren Werten.

Ein Nachteil besteht in einer leichten Verzögerung des Signals, da schnelle Änderungen gedämpft werden. Die Wasserparameter jedoch nur langsam ändern, daher ist dieser Effekt für die Anwendung nicht von Bedeutung. [15]

## 4.5. Schaltplan und Dokumentation

Der vollständige Schaltplan des Systems ist in Abbildung 11 dargestellt. Dieser zeigt die direkte Anbindung aller Sensoren an den Mikrocontroller, die Umsetzung des Spannungsteilers für den pH-Sensor und die RC-Tiefpassfilter.

## 4.6. Zusammenfassung

Die Hardwareimplementierung basiert auf einer direkten Anbindung aller Sensoren an den ESP32-S3 ohne zusätzliche externe Signalverarbeitungseinheiten. Die Spannungsversorgung erfolgt zentral über USB, wobei die interne Spannungsregelung des Mikrocontrollers zur Versorgung aller Sensoren genutzt wird. Dadurch konnte die Schaltung einfach gehalten und auf zusätzliche Pegelwandler verzichtet werden.

Die Sensorintegration umfasst sowohl digitale als auch analoge Signalquellen. Der Temperatursensor DS18B20 wird über das OneWire-Protokoll angebunden und liefert störungsarme digitale Messwerte. Im Gegensatz dazu werden die analogen Signale des TDS- und des pH-Sensors direkt über den internen ADC des Mikrocontrollers erfasst.

Die Digitalisierung erfolgt mit einer Auflösung von 12 Bit, wodurch eine ausreichende Diskretisierung der Eingangssignale erreicht wird. In der praktischen Anwendung zeigte sich jedoch, dass insbesondere analoge Signale durch Rauschen beeinflusst werden. Zur Verbesserung der Signalqualität wurden daher hardwareseitig RC-Tiefpassfilter implementiert, die hochfrequente Störungen reduzieren und zu stabileren Messwerten führen.

## 4.6. Zusammenfassung

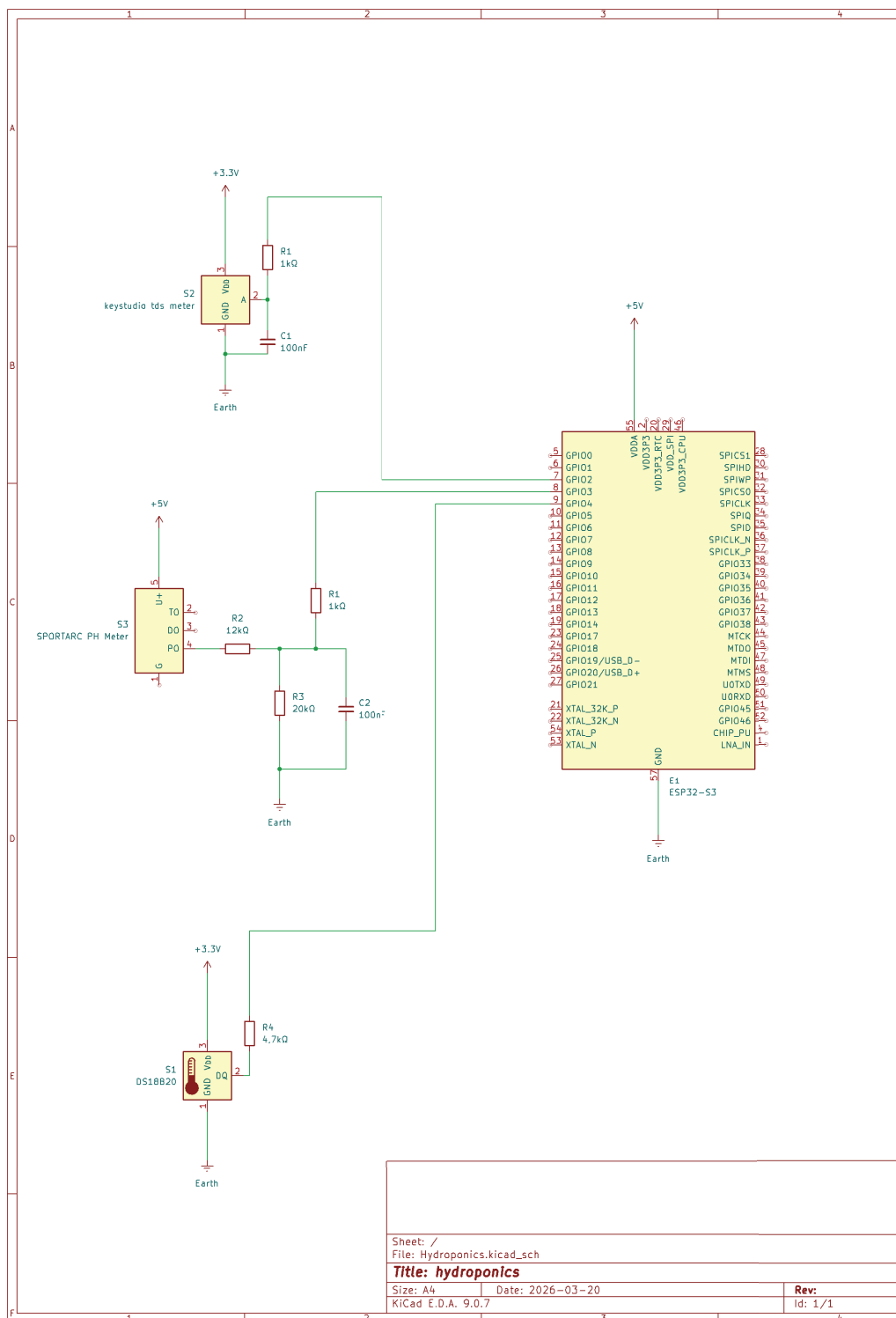


Abbildung 11.: Gesamtschaltplan des Systems





## 5. Softwareimplementierung

Dieses Kapitel beschreibt die Implementierung der Software zur Erfassung und Verarbeitung und Bereitstellung der Sensordaten. Die Software wurde in der Programmiersprache Rust unter Verwendung der *esp-idf-hal*-Bibliothek entwickelt und läuft auf dem ESP32-S3 Mikrocontroller.

Im Vergleich zu einer einfachen Messwerterfassung beinhaltet die Implementierung zusätzlich Mechanismen zur Signalfilterung, Kalibrierung sowie Temperaturkompensation, um die Qualität der Messdaten zu verbessern.

### 5.1. Softwarearchitektur

Die Software ist modular aufgebaut und in mehrere Komponenten unterteilt:

- `main.rs`: zentrale Steuerung, Zeitlogik und Datenerfassung
- `tds.rs`: Verarbeitung der Leitfähigkeitsdaten
- `ph.rs`: Kalibrierung und Berechnung des pH-Wertes
- `temperature.rs`: Ansteuerung des DS18B20 Sensors

Diese Struktur ermöglicht eine klare Trennung zwischen Hardwarezugriff und Datenverarbeitung. Der vollständige Rust-Code, sowie Konfigurationsdateien sind in Anhang A enthalten.

### 5.2. Zeitsteuerung und Messintervall

Die Messung erfolgt nicht kontinuierlich, sondern in definierten Zeitintervallen. Das Intervall ist auf  $t = 10 \text{ min}$  festgelegt und wird durch eine zeitbasierte Überprüfung realisiert. Hierzu wird der Zeitpunkt der letzten Messung gespeichert und fortlaufend mit der aktuellen Systemzeit verglichen. Sobald das definierte Intervall überschritten wird, wird

## 5. Softwareimplementierung

eine neue Messung ausgelöst. Zwischen den einzelnen Messungen läuft eine Schleife mit einer Verzögerung von einer Sekunde, um die Systemlast zu reduzieren und eine einfache zeitliche Steuerung zu ermöglichen.

## 5.3. Analog-Digital-Wandlung

Die analogen Sensoren werden über den internen Analog-Digital-Converter (ADC) des ESP32-S3 erfasst.

### 5.3.1. Konfiguration

Der ADC wird wie folgt konfiguriert:

- Auflösung: 12 *Bit* (0 – 4095)
- Attenuation: 12 *dB*
- Referenzspannung: 3,3 *V*

Diese Konfiguration ermöglicht die Messung von Spannungen im gesamten relevanten Bereich der Sensorsignale.

## 5.4. Signalfilterung

Eine zentrale Erweiterung gegenüber einfachen Implementierungen ist die Filterung der ADC-Werte, wie nachfolgend erläutert.

### 5.4.1. Mehrfachabtastung

Für jede Messung werden mehrere ADC-Werte aufgenommen:

- Anzahl der Samples: 32
- Abstand zwischen Messungen: 1 *s*

### 5.4.2. Trimmed Mean Filter

Zur Reduktion von Ausreißern wird ein sogenannter *Trimmed Mean* verwendet [4]:

- Sortieren der Messwerte
- Entfernen der kleinsten und größten Werte (jeweils 4 Werte)
- Mittelwertbildung der verbleibenden Werte

Mathematisch ergibt sich:

$$\bar{x} = \frac{1}{N} \sum_{i=k}^{n-k} x_i$$

mit:

- $n = 32$
- $k = 4$

Dieser Ansatz reduziert effektiv Störungen und verbessert die Stabilität der Messwerte.

## 5.5. Temperaturmessung

Die Temperaturmessung erfolgt über den digitalen Sensor DS18B20.

Die Implementierung verwendet das OneWire-Protokoll, das eine präzise zeitliche Steuerung erfordert [6]. Hierzu wird eine eigene Delay-Struktur implementiert, die auf Hardware-Timern basiert.

Der Messablauf umfasst:

1. Start der Temperaturmessung
2. Warten auf Abschluss (abhängig von Auflösung)
3. Suche nach Geräten auf dem Bus
4. Auslesen der Temperaturdaten

## 5.6. TDS-Verarbeitung

Die Verarbeitung der Total Dissolved Solids (TDS)-Daten erfolgt in `tds.rs`.

Der ADC-Rohwert wird in einen normierten Prozentwert umgerechnet:

$$\text{TDS}_{pct} = \frac{\text{ADC}}{4095} \cdot 100$$

Diese Implementierung stellt eine vereinfachte Darstellung dar und dient primär der relativen Bewertung der Wasserqualität.

Eine direkte Umrechnung in parts per million (ppm) erfolgt nicht, da hierfür eine Kalibrierung erforderlich wäre.

## 5.7. pH-Berechnung und Kalibrierung

Die pH-Verarbeitung stellt den komplexesten Teil der Software dar.

### 5.7.1. Spannungsberechnung

Zunächst wird der ADC-Wert in eine Spannung umgerechnet:

$$V_{ADC} = \frac{\text{ADC}}{4095} \cdot 3,3 \text{ V}$$

Anschließend wird die ursprüngliche Sensorspannung rekonstruiert [13]:

$$V_{PO} = V_{ADC} \cdot \frac{R2 + R3}{R3}$$

mit:

- $R2 = 12 \text{ k}\Omega$
- $R3 = 20 \text{ k}\Omega$

### 5.7.2. 2-Punkt-Kalibrierung

Im Gegensatz zu einer einfachen linearen Näherung wird eine Kalibrierung mit zwei Referenzpunkten verwendet:

- pH 7
- pH 4

Die Kalibrierung erfolgt direkt auf Basis der ADC-Rohwerte:

$$pH = pH_1 + \frac{(raw - raw_1)(pH_2 - pH_1)}{raw_2 - raw_1}$$

Dabei wird überprüft, ob die Kalibrierwerte plausibel sind, um fehlerhafte Berechnungen zu vermeiden.

### 5.7.3. Temperaturkompensation

Zusätzlich wird eine Temperaturkompensation basierend auf der Nernst-Gleichung implementiert [17].

Die Steigung der pH-Kennlinie ist temperaturabhängig:

$$S(T) = S_{25} \cdot \frac{T}{298.15}$$

mit:

- $S_{25} = 59.16 \frac{mV}{pH}$
- $T$  über DS18B20 gemessene Temperatur (hier in Kelvin ( $K$ ))

Die Kompensation erfolgt relativ zum Neutralpunkt (pH 7), wodurch systematische Fehler reduziert werden.

### 5.7.4. Validierung der Messwerte

Zusätzlich wird überprüft, ob die verwendeten Kalibrierwerte gültig sind und ob der ADC-Wert innerhalb eines plausiblen Bereichs liegt. Das Ergebnis dieser Prüfung wird als boolescher Wert `valid` ausgegeben.

Die Verifizierung erfolgt in mehreren aufeinanderfolgenden Schritten:

Zunächst wird geprüft, ob die verwendeten Kalibrierwerte gültig sind, indem sichergestellt wird, dass sich die beiden Referenzpunkte (z. B. pH 4 und pH 7) ausreichend unterscheiden und keine Division durch nahezu gleiche Werte entsteht. Anschließend wird kontrolliert, ob der aktuelle ADC-Rohwert innerhalb eines plausiblen Bereichs liegt und nicht auf einen Messfehler wie Kurzschluss oder offene Leitung hindeutet (z. B. Werte nahe 0 oder 4095).

Daraufhin wird aus dem Messwert der pH-Wert berechnet und auf den physikalisch sinnvollen Bereich von 0 bis 14 begrenzt.

## 5.8. Datenformat und Ausgabe

Die Messwerte werden als strukturierter String ausgegeben:

```
T=26.25 , TDS=22.9 , PH=6.81
```

Zusätzlich erfolgt eine detaillierte Ausgabe auf der Konsole zur Analyse:

```
Temp: 26.25 °C | TDS raw ADC(filtered): 936 / 4095 (22.9%) | pH raw  
      ADC(filtered): 1968 / 4095 | pH=6.81 | valid=true
```

Die Ausgabe erfolgt für Demonstrationszwecke dieser Arbeit über das Logging-System des Espressif IoT Development Framework (ESP-IDF). Im realen System wird dieses dann mittels Funk weiter geleitet.

## 5.9. Zusammenfassung

Die implementierte Software kombiniert mehrere Techniken zur Verbesserung der Messqualität, darunter eine Mittelwertbildung mit Ausreißerunterdrückung, eine 2-Punkt-Kalibrierung, eine Temperaturkompensation sowie eine Plausibilitätsprüfung der Messwerte. Im Vergleich zu einfachen Messsystemen wird dadurch eine deutlich stabilere und realistischere Erfassung der Wasserparameter erreicht, wobei gleichzeitig die Systemkomplexität gering gehalten wird.

## 6. Tests und Validierung

Zur Bewertung der Funktionalität und Genauigkeit des entwickelten Systems wurden Kalibrierungen sowie Vergleichsmessungen mit Referenzgeräten durchgeführt. Ziel ist es, die Aussagekraft der Messwerte zu überprüfen und systematische Abweichungen zu identifizieren.

Der Schwerpunkt liegt auf der Validierung der pH- und Temperaturmessung, da für diese Parameter geeignete Referenzmethoden verfügbar sind.

Für die Total Dissolved Solids (TDS)-Messung wurde keine Validierung gegenüber einem Referenzgerät durchgeführt, da kein geeignetes Vergleichssystem zur Verfügung stand. Die TDS-Werte dienen in der vorliegenden Arbeit daher primär der relativen Bewertung sowie der Beobachtung von Veränderungen im System. Eine absolute Bewertung der Messgenauigkeit ist ohne Kalibrierung und Referenzmessung nicht möglich. Die TDS-Messung zeigte jedoch reproduzierbare Messergebnisse bei unterschiedlichen Wasserproben.

### 6.1. Kalibrierung der Sensoren

Die Kalibrierung stellt einen wesentlichen Schritt dar, um die Rohmesswerte der Sensoren in physikalisch sinnvolle Größen zu überführen. Insbesondere bei der pH-Messung ist eine Kalibrierung erforderlich, da das Sensorsignal von verschiedenen Einflussgrößen abhängig ist.

#### 6.1.1. pH-Pufferlösungen

Die Kalibrierung des pH-Sensors erfolgt mithilfe standardisierter Pufferlösungen mit bekannten pH-Werten. In der vorliegenden Arbeit wurde eine 2-Punkt-Kalibrierung durchgeführt.

Hierzu wurden zwei Referenzlösungen verwendet:

- Pufferlösung 1:  $\text{pH} = 7$
- Pufferlösung 2:  $\text{pH} = 4$

## 6. Tests und Validierung

Für beide Lösungen wurden die entsprechenden Analog-Digital-Converter (ADC)-Rohwerte aufgenommen:

- $raw_1 = 1948$
- $raw_2 = 2270$

Die Messungen erfolgten nach ausreichender Stabilisierung des Sensors in der jeweiligen Lösung, um Übergangseffekte zu minimieren.

Auf Basis dieser Werte wird eine lineare Kalibrierfunktion bestimmt:

$$pH = pH_1 + \frac{(raw - raw_1)(pH_2 - pH_1)}{raw_2 - raw_1}$$

Diese Funktion wird in der Software verwendet, um aus den ADC-Werten pH-Werte zu berechnen.

### 6.1.2. Referenzmessungen

Zur Überprüfung der Kalibrierung wurden zusätzliche Messungen in weiteren Lösungen durchgeführt. Dabei wurden die berechneten pH-Werte mit bekannten Referenzwerten verglichen.

Die Ergebnisse zeigen, dass die 2-Punkt-Kalibrierung die Genauigkeit der Messung deutlich verbessert. Besonders im Bereich zwischen den Kalibrierpunkten ergibt sich eine gute Übereinstimmung zwischen gemessenen und erwarteten Werten.

Abweichungen treten vor allem außerhalb dieses Bereichs auf, was auf die lineare Näherung der Sensorkennlinie zurückzuführen ist.

## 6.2. Vergleich mit Referenzgeräten

Zur Validierung der Messgenauigkeit wurde das entwickelte System mit einem Referenzmessgerät verglichen.

### 6.2.1. Versuchsaufbau

Die Messungen wurden parallel mit dem entwickelten System und einem Referenzgerät durchgeführt. Beide Systeme wurden in derselben Wasserprobe betrieben, um identische Messbedingungen sicherzustellen.

Das verwendete Referenzgerät ist:



- Gerät: Analytical Instruments ph-W218
- Messparameter: pH, Temperatur

Vor Beginn der Messungen wurde eine ausreichende Stabilisierung der Systeme abgewartet, um temperaturabhängige Einflüsse zu minimieren.

### 6.2.2. Messdurchführung

Für die Validierung wurden mehrere Messungen durchgeführt. Dabei wurden jeweils die Messwerte beider Systeme erfasst und miteinander verglichen.

### 6.2.3. Ergebnisse

Eine Messung der pH-Werte und der Temperatur wurde unabhängig von einander durchgeführt, jedoch in Tabelle 1 gemeinsam dargestellt.

| Messung | pH (System) | pH (Referenz) | Temp (System) | Temp (Referenz) |
|---------|-------------|---------------|---------------|-----------------|
| 1       | 4,01        | 4,00          | 5,2           | 5,0             |
| 2       | 5,58        | 5,50          | 20,1          | 20,0            |
| 3       | 10,27       | 10,00         | 35,3          | 35,0            |

Tabelle 1.: Vergleich der Messwerte zwischen System und Referenzgerät

Zur Bewertung der Messgenauigkeit wird die absolute Abweichung berechnet:

$$\Delta pH = |pH_{System} - pH_{Referenz}|$$

$$\Delta T = |T_{System} - T_{Referenz}|$$

### 6.2.4. Bewertung

Die Ergebnisse zeigen, dass das entwickelte System sowohl für die Temperatur- als auch für die pH-Messung grundsätzlich zuverlässige Werte liefert.

Die Temperaturmessung weist aufgrund des digitalen Sensors eine geringe Abweichung auf und zeigt eine gute Übereinstimmung mit dem Referenzgerät.

## 6. Tests und Validierung

Die pH-Messung profitiert deutlich von der durchgeführten Kalibrierung. Die verbleibenden Abweichungen liegen im Bereich von:

- pH: 0,27
- Temperatur: 0,3

### 6.3. Zusammenfassung

Die durchgeführten Tests zeigen, dass das entwickelte System für die Erfassung von Temperatur und pH-Wert geeignet ist. Durch die 2-Punkt-Kalibrierung konnte die Genauigkeit der pH-Messung signifikant verbessert werden.

Die Validierung bestätigt, dass das System insbesondere zur Überwachung von Trends und Veränderungen eingesetzt werden kann. Einschränkungen bestehen bei der TDS-Messung, da hierfür keine Validierung durchgeführt wurde.

## 7. Fazit

Im Rahmen dieser Arbeit wurde ein kostengünstiges System zur Erfassung und Verarbeitung von Wasserqualitätsdaten in aquaponischen Anwendungen entwickelt. Zentrale Messgrößen sind Temperatur, pH-Wert und Total Dissolved Solids (TDS), die über eine Kombination aus digitaler und analoger Sensorik erfasst werden. Die entwickelte Lösung bildet die Grundlage für ein modulares Gesamtsystem, bei dem die drahtlose Datenübertragung über LoRa in einer separaten Arbeit behandelt wird.

Die Umsetzung auf Basis eines ESP32-S3 ermöglicht eine kompakte und integrierte Systemarchitektur ohne zusätzliche externe Komponenten. Während die digitale Temperaturmessung robust realisiert werden konnte, zeigte sich bei den analogen Sensoren eine erhöhte Anfälligkeit gegenüber Störungen.

Zur Verbesserung der Messqualität wurden sowohl hardware- als auch softwareseitige Maßnahmen umgesetzt. Dazu zählen RC-Tiefpassfilter zur Reduktion von Rauschen sowie eine softwareseitige Mittelwertbildung mit Ausreißerunterdrückung. Zusätzlich wurde für den pH-Sensor eine 2-Punkt-Kalibrierung mit Temperaturkompensation implementiert.

Die Ergebnisse zeigen, dass mit kostengünstiger Hardware ein funktionsfähiges System zur kontinuierlichen Überwachung von Wasserparametern realisiert werden kann. Einschränkungen ergeben sich, da die TDS-Messung nicht kalibriert wurde.

Insgesamt stellt das entwickelte System einen praktikablen Kompromiss zwischen Kosten, Komplexität und Funktionalität dar und eignet sich insbesondere zur Erkennung von Veränderungen und Trends innerhalb aquaponischer Systeme.

Trotz der genannten technischen und wirtschaftlichen Einschränkungen zeigt sich, dass Aquaponik insbesondere für landwirtschaftliche Betriebe in Afrika ein vielversprechendes System darstellt. Die Möglichkeit, Wasser effizient zu recyceln und Nährstoffe im Kreislauf zu nutzen, macht diese Technologie besonders geeignet für Regionen mit Wasserknappheit und begrenzten landwirtschaftlichen Flächen [2, 11]. Selbst wenn Herausforderungen wie Systemkomplexität, Wartungsaufwand oder Anfangsinvestitionen bestehen, überwiegen die Vorteile in Bezug auf Ernährungssicherheit, Ressourcenschonung und lokale Wertschöpfung. Damit kann Aquaponik einen wichtigen Beitrag zur nachhaltigen Landwirtschaft und zur

## 7. Fazit

Verbesserung der Lebensbedingungen leisten.

Für zukünftige Arbeiten ergeben sich mehrere Ansatzpunkte zur Weiterentwicklung des Systems. Eine wesentliche Verbesserung besteht in der Kalibrierung der TDS-Messung, um eine direkte Umrechnung in ppm zu ermöglichen. Darüber hinaus könnten externe, hochauflösende Analog-Digital-Converter (ADC) eingesetzt werden, um die Messgenauigkeit insbesondere beim pH-Sensor zu erhöhen, dies sind jedoch Maßnahmen welche mit erhöhten Kosten verbunden sind.

Weiterhin bietet sich die Integration zusätzlicher Sensoren, wie beispielsweise für gelösten Sauerstoff oder Redoxpotenzial, an. Auch die Erweiterung der Software um eine automatisierte Kalibrierung würde die praktische Anwendung des Systems deutlich vereinfachen.

Im Kontext größerer Projekte, wie beispielsweise im Bereich der nachhaltigen Lebensmittelproduktion, kann das entwickelte System als Grundlage für skalierbare Monitoringlösungen dienen. Insbesondere in ressourcenlimitierten Umgebungen bietet der vorgestellte Ansatz eine kostengünstige Möglichkeit zur Überwachung der Wasserqualität.

# A. Programmcode

## A.1. main.rs

```
1 use std::thread;
2 use std::time::{Duration, Instant};
3
4 use esp_idf_hal::adc::oneshot::config::AdcChannelConfig;
5 use esp_idf_hal::adc::oneshot::{AdcChannelDriver, AdcDriver};
6 use esp_idf_hal::adc::{attenuation, Resolution};
7 use esp_idf_hal::gpio::{AnyIOPin, PinDriver, Pull};
8 use esp_idf_hal::peripherals::Peripherals;
9
10 use esp_idf_svc::log::EspLogger;
11 use esp_idf_svc::sys::link_patches;
12
13 use log::info;
14 use one_wire_bus::OneWire;
15
16 mod ph;
17 mod tds;
18 mod temperature;
19
20 const SEND_INTERVAL: Duration = Duration::from_secs(10 * 60);
21 const ADC_FILTER_SAMPLES: usize = 32;
22 const ADC_SAMPLE_DELAY_MS: u64 = 5;
23 const ADC_TRIM_COUNT: usize = 4;
24
25 fn main() -> anyhow::Result<()> {
26     link_patches();
27     EspLogger::initialize_default();
28
29     let peripherals = Peripherals::take().unwrap();
30
31     let adc = AdcDriver::new(peripherals.adc1)?;
32
33     let ch_cfg = AdcChannelConfig {
34         attenuation: attenuation::DB_12,
35         resolution: Resolution::Resolution12Bit,
36         calibration: Default::default(),
37     };
```

## A. Programcode

```
38
39 // TDS on GPIO2
40 let mut tds_ch = AdcChannelDriver::new(&adc,
    peripherals.pins.gpio2, &ch_cfg)?;
41
42 // pH PO on GPIO3
43 let mut ph_po_ch = AdcChannelDriver::new(&adc,
    peripherals.pins.gpio3, &ch_cfg)?;
44
45 // DS18B20 on GPIO4
46 let ds_pin: AnyIOPin = peripherals.pins.gpio4.into();
47 let ds_io = PinDriver::input_output_od(ds_pin, Pull::Floating)?;
48
49 let mut delay = temperature::OwDelay;
50 let mut one_wire_bus =
51     OneWire::new(ds_io).map_err(|e| anyhow::anyhow!("OneWire
        init failed: {:?}", e))?;
52
53 let mut last_send = Instant::now() - SEND_INTERVAL;
54
55 loop {
56     if last_send.elapsed() >= SEND_INTERVAL {
57         let temp_c = temperature::get_first_temperature(&mut
            delay, &mut one_wire_bus)
58             .map_err(|e| anyhow::anyhow!("get_temperature
                failed: {:?}", e))?;
59
60         let tds_raw = read_adc_trimmed_mean(&adc, &mut tds_ch)?;
61         let tds_percent = tds::percent_from_raw(tds_raw);
62
63         let ph_po_raw = read_adc_trimmed_mean(&adc, &mut
            ph_po_ch)?;
64         let ph_reading = ph::reading_from_raw(ph_po_raw, temp_c);
65
66         let payload = format!(
67             "T={:.2},TDS={:.1},PH={:.2}",
68             temp_c,
69             tds_percent,
70             ph_reading.ph,
71         );
72
73         info!(
74             "Temp: {:.2} °C | TDS raw ADC(filtered): {} / {}
                ({:.1}%) | pH raw ADC(filtered): {} / {} |
                pH{:.2} | valid={}",
75             temp_c,
76             tds_raw,
77             tds::ADC_MAX_12BIT,
78             tds_percent,
79             ph_reading.raw,
```

```

80             ph::ADC_MAX_12BIT,
81             ph_reading.ph,
82             ph_reading.valid
83         );
84
85         info!("TX: {}", payload);
86
87         last_send = Instant::now();
88     }
89
90     thread::sleep(Duration::from_secs(1));
91 }
92 }
93
94 fn read_adc_trimmed_mean<'d, C, U>(
95     adc: &AdcDriver<'d, U>,
96     ch: &mut AdcChannelDriver<'d, C, &'d AdcDriver<'d, U>>,
97 ) -> anyhow::Result<u16>
98 where
99     U: esp_idf_hal::adc::AdcUnit,
100     C: esp_idf_hal::adc::AdcChannel<AdcUnit = U>,
101 {
102     let mut values = [0u16; ADC_FILTER_SAMPLES];
103
104     for value in &mut values {
105         *value = adc.read(ch)?;
106         thread::sleep(Duration::from_millis(ADC_SAMPLE_DELAY_MS));
107     }
108
109     values.sort_unstable();
110
111     let start = ADC_TRIM_COUNT;
112     let end = ADC_FILTER_SAMPLES - ADC_TRIM_COUNT;
113
114     let mut sum: u32 = 0;
115     for value in &values[start..end] {
116         sum += *value as u32;
117     }
118
119     let count = (end - start) as u32;
120     Ok((sum / count) as u16)
121 }

```

Listing A.1: Quellcode des Hauptprogramms

## A.2. temperature.rs

```
1 use esp_idf_hal::delay::Ets;
2
3 use embedded_hal::blocking::delay::{DelayMs, DelayUs};
4 use embedded_hal::digital::v2::{InputPin, OutputPin};
5
6 use ds18b20::Resolution;
7 use one_wire_bus::{OneWire, OneWireResult};
8
9 pub struct OwDelay;
10
11 impl DelayUs<u16> for OwDelay {
12     fn delay_us(&mut self, us: u16) {
13         Ets::delay_us(us as u32);
14     }
15 }
16
17 impl DelayMs<u16> for OwDelay {
18     fn delay_ms(&mut self, ms: u16) {
19         Ets::delay_ms(ms as u32);
20     }
21 }
22
23 pub fn get_first_temperature<P, E>(
24     delay: &mut (impl DelayUs<u16> + DelayMs<u16>),
25     one_wire_bus: &mut OneWire<P>,
26 ) -> OneWireResult<f32, E>
27 where
28     P: OutputPin<Error = E> + InputPin<Error = E>,
29     E: core::fmt::Debug,
30 {
31     ds18b20::start_simultaneous_temp_measurement(one_wire_bus,
32         delay)?;
33     Resolution::Bits12.delay_for_measurement_time(delay);
34
35     let mut search_state = None;
36     loop {
37         if let Some((device_address, state)) =
38             one_wire_bus.device_search(search_state.as_ref(), false,
39                 delay)?
40         {
41             search_state = Some(state);
42
43             if device_address.family_code() != ds18b20::FAMILY_CODE {
44                 continue;
45             }
46
47             let sensor = ds18b20::Ds18b20::new(device_address)?;
```



```

46         let data = sensor.read_data(one_wire_bus, delay)?;
47         return Ok(data.temperature);
48     } else {
49         break;
50     }
51 }
52
53 panic!("No DS18B20 found");
54 }

```

Listing A.2: Quellcode des Temperatur-Sensors

## A.3. tds.rs

```

1 pub const ADC_MAX_12BIT: u16 = 4095;
2
3 pub fn percent_from_raw(raw: u16) -> f32 {
4     (raw as f32 / ADC_MAX_12BIT as f32) * 100.0
5 }

```

Listing A.3: Quellcode des TDS-Sensors

## A.4. ph.rs

```

1 pub const ADC_MAX_12BIT: u16 = 4095;
2
3 /// ----- Hardware -----
4 /// Spannungsteiler:
5 /// PO --- 12k ----+---- ADC
6 ///                |
7 ///                20k
8 ///                |
9 ///                GND
10 const R_TOP_OHMS: f32 = 12_000.0;
11 const R_BOT_OHMS: f32 = 20_000.0;
12 const VDIV_FACTOR: f32 = (R_TOP_OHMS + R_BOT_OHMS) / R_BOT_OHMS;
13
14 /// Nur für Logging/Debug, nicht für die eigentliche Kalibrierung
15 const ADC_FULL_SCALE_VOLTS_APPROX: f32 = 3.3;
16
17 /// ----- 2-Punkt-Kalibrierung -----
18 /// Diese Rohwerte sind ADC-Werte NACH dem Spannungsteiler.
19 const CAL_PH7_RAW: u16 = 1948;
20 const CAL_PH4_RAW: u16 = 2270;
21
22 /// pH-Werte der verwendeten Puffer
23 const CAL_PH7_VALUE: f32 = 7.0;
24 const CAL_PH4_VALUE: f32 = 4.0;
25

```

## A. Programmcode

```
26 /// Mindestabstand zwischen den Kalibrierpunkten
27 const MIN_CAL_RAW_DELTA: u16 = 20;
28
29 /// Referenztemperatur der Kalibrierung
30 const CAL_TEMP_C: f32 = 25.0;
31
32 /// Nernst-Steigung in mV/pH bei 25°C
33 const NERNST_MV_PER_PH_AT_25C: f32 = 59.16;
34
35 pub struct PhReading {
36     pub raw: u16,
37     pub v_adc: f32,
38     pub v_po: f32,
39     pub ph: f32,
40     pub valid: bool,
41 }
42
43 pub fn reading_from_raw(po_raw: u16, temp_c: f32) -> PhReading {
44     let v_adc = raw_to_vadc_approx(po_raw);
45     let v_po = v_adc * VDIV_FACTOR;
46
47     let ph = match calibrated_ph_from_raw(po_raw, temp_c) {
48         Some(value) => value,
49         None => 7.0,
50     };
51
52     let valid = calibration_is_valid() && raw_is_plausible(po_raw);
53
54     PhReading {
55         raw: po_raw,
56         v_adc,
57         v_po,
58         ph,
59         valid,
60     }
61 }
62
63 fn calibrated_ph_from_raw(raw_meas: u16, temp_c: f32) -> Option<f32>
64 {
65     if !calibration_is_valid() || !raw_is_plausible(raw_meas) {
66         return None;
67     }
68
69     let raw1 = CAL_PH7_RAW as f32;
70     let raw2 = CAL_PH4_RAW as f32;
71
72     let ph1 = CAL_PH7_VALUE;
73     let ph2 = CAL_PH4_VALUE;
74
75     let denom = raw2 - raw1;
```

```

75     if denom.abs() < 1.0 {
76         return None;
77     }
78
79     // Kalibrierung zuerst bei Referenztemperatur
80     let ph_at_cal_temp = ph1 + (raw_meas as f32 - raw1) * (ph2 -
        ph1) / denom;
81
82     // Temperaturkompensation über Verhältnis der Nernst-Steigung
83     let slope_ratio = nernst_slope_mv_per_ph(temp_c) /
        nernst_slope_mv_per_ph(CAL_TEMP_C);
84
85     // Um pH 7 herum kompensieren
86     let compensated = 7.0 + (ph_at_cal_temp - 7.0) / slope_ratio;
87
88     Some(compensated.clamp(0.0, 14.0))
89 }
90
91 fn calibration_is_valid() -> bool {
92     let delta = CAL_PH7_RAW.abs_diff(CAL_PH4_RAW);
93     delta >= MIN_CAL_RAW_DELTA
94 }
95
96 fn raw_is_plausible(raw: u16) -> bool {
97     raw > 0 && raw < ADC_MAX_12BIT
98 }
99
100 fn nernst_slope_mv_per_ph(temp_c: f32) -> f32 {
101     // Näherung:
102     //  $S(T) = 59.16 \text{ mV/pH} * (T_K / 298.15 \text{ K})$ 
103     let temp_k = temp_c + 273.15;
104     NERNST_MV_PER_PH_AT_25C * (temp_k / 298.15)
105 }
106
107 fn raw_to_vadc_approx(raw: u16) -> f32 {
108     (raw as f32 / ADC_MAX_12BIT as f32) * ADC_FULL_SCALE_VOLTS_APPROX
109 }

```

Listing A.4: Quellcode des pH-Sensors

## A.5. Cargo.toml

```

1 [package]
2 name = "hydroponics"
3 version = "0.1.0"
4 authors = ["peet"]
5 edition = "2024"
6 resolver = "2"
7

```

### A. Programmcode

```
8  [[bin]]
9  name = "hydroponics"
10 harness = false
11
12 [profile.release]
13 opt-level = "s"
14
15 [profile.dev]
16 debug = true
17 opt-level = "z"
18
19 [features]
20 default = []
21
22 experimental = ["esp-idf-svc/experimental"]
23
24 [dependencies]
25 log = "0.4"
26 esp-idf-svc = "0.52"
27 esp-idf-hal = "0.46"
28 anyhow = "1.0.102"
29 one-wire-bus = "0.1.1"
30 ds18b20 = "0.1.1"
31 embedded-hal = "0.2.7"
32
33
34 [build-dependencies]
35 embuild = "0.33"
36
37 [patch.crates-io]
38 esp-idf-svc = { git = "https://github.com/esp-rs/esp-idf-svc" }
39 esp-idf-hal = { git = "https://github.com/esp-rs/esp-idf-hal" }
40 esp-idf-sys = { git = "https://github.com/esp-rs/esp-idf-sys" }
```

Listing A.5: Inhalt der Cargo.toml

# Literaturverzeichnis

- [1] Hooman Abolfathi, Alireza Nikfarjam und Bahareh Abbaspour. „An Integrated Sensor Array for Water Quality Monitoring“. In: *IET Science, Measurement & Technology* 19.1 (Januar 2025), e70013. URL: <https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/smt2.70013> (Stand: 10.03.2026).
- [2] *Aquaponik in Afrika – Halbzeitbilanz Des EU-Projekts INCiTiS-FOOD*. 15. August 2024. URL: <https://www.aquaponik-manufaktur.de/en/incitis-food-die-ersten-18-monate/> (Stand: 10.03.2026).
- [3] *Aquaponik: Nachhaltige Fisch-Pflanzen-Kultur*. REKUBIK GmbH. URL: <https://www.rekubik.de/landingPage/0194f3fffd9027f0a94055d991e5b5b1c> (Stand: 24.03.2026).
- [4] J. Bednar und T. Watt. „Alpha-Trimmed Means and Their Relationship to Median Filters“. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.1 (Februar 1984), Seiten 145–153. URL: <http://ieeexplore.ieee.org/document/1164279/> (Stand: 21.03.2026).
- [5] John Colt. „Water Quality Requirements for Reuse Systems“. In: *Aquacultural Engineering* 34.3 (Mai 2006), Seiten 143–156. URL: <https://linkinghub.elsevier.com/retrieve/pii/S014486090500124X> (Stand: 10.03.2026).
- [6] *DS18B20 - Programmable Resolution 1-Wire Digital Thermometer*. URL: <https://www.analog.com/media/en/technical-documentation/data-sheets/ds18b20.pdf> (Stand: 22.03.2026).
- [7] *FrontAg Nexus – Pushing the Frontiers of Climate-Smart Agriculture in the Mediterranean*. URL: <https://frontagnexus.eu/> (Stand: 10.03.2026).
- [8] Abdelmalek Ghobrini, Abdessamed Belhakimi und Youcef Remram. „Design and Implementation of a Cost-Effective IoT-Enabled Multi-Parameter System for Water Quality Monitoring in Aquaculture and Aquaponics“. In: *2023 2nd International Conference on Electronics, Energy and Measurement (IC2EM)*. 2023 2nd International Conference on Electronics, Energy and Measurement (IC2EM). Medea, Algeria: IEEE,

28. November 2023, Seiten 1–6. URL: <https://ieeexplore.ieee.org/document/10419750/> (Stand: 10.03.2026).
- [9] Simon Goddek, Boris Delaide, Utra Mankasingh, Kristin Ragnarsdottir, Haissam Jijakli und Ragnheidur Thorarinsdottir. „Challenges of Sustainable and Commercial Aquaponics“. In: *Sustainability* 7 (10. April 2015), Seiten 4199–4224. URL: <https://research.ebsco.com/linkprocessor/plink?id=3b07b99f-2433-3582-ab01-459f835538d6> (Stand: 19.03.2026).
- [10] *Heltec WiFi LoRa 32 V3.2*. URL: [https://resource.heltec.cn/download/WiFi\\_LoRa\\_32\\_V3/HTIT-WB32LA\\_V3.2.pdf](https://resource.heltec.cn/download/WiFi_LoRa_32_V3/HTIT-WB32LA_V3.2.pdf) (Stand: 22.03.2026).
- [11] *Home - Incitis Food*. 25. Januar 2024. URL: <https://incitis-food.eu/> (Stand: 10.03.2026).
- [12] Tadaaki Ikehara und Kazuyuki Kojima. „The Development of a Water Quality Monitoring System for Aquaponics“. In: *2025 IEEE 14th Global Conference on Consumer Electronics (GCCE)*. 2025 IEEE 14th Global Conference on Consumer Electronics (GCCE). Osaka, Japan: IEEE, 23. September 2025, Seiten 473–474. URL: <https://ieeexplore.ieee.org/document/11274696/> (Stand: 10.03.2026).
- [13] K Landes. *Skriptum Zur Vorlesung Grundlagen Der Elektrotechnik I*.
- [14] Wilson Lennard und Simon Goddek. „Aquaponics: The Basics“. In: *Aquaponics Food Production Systems: Combined Aquaculture and Hydroponic Production Technologies for the Future*. Herausgegeben von Simon Goddek, Alyssa Joyce, Benz Kotzen und Gavin M. Burnell. Cham: Springer International Publishing, 2019, Seiten 113–143. URL: [https://doi.org/10.1007/978-3-030-15943-6\\_5](https://doi.org/10.1007/978-3-030-15943-6_5) (Stand: 24.03.2026).
- [15] Johannes Niebuhr. *Physikalische Messtechnik Mit Sensoren*. 6., aktualisierte Aufl. München: Oldenbourg, 2011. XIV, 497 S.
- [16] Anna F Rusydi. „Correlation between Conductivity and Total Dissolved Solid in Various Type of Water: A Review“. In: *IOP Conference Series: Earth and Environmental Science* 118 (Februar 2018), Seite 012019. URL: <https://iopscience.iop.org/article/10.1088/1755-1315/118/1/012019> (Stand: 10.03.2026).
- [17] Gerd Wedler und Hans-Joachim Freund. *Lehr- und Arbeitsbuch physikalische Chemie*. Siebte, wesentlich überarbeitete und erweiterte Auflage. Weinheim, Germany: Wiley-VCH Verlag GmbH & Co. KGaA, 2018. 1 Seite.