

(Ich habe meinen Text aus Jux von ChatGPT glätten lassen.)

Hallo zusammen,

wie besprochen, erhaltet ihr hier eine detaillierte Anleitung zur Verwendung und Programmierung eurer Sensoren für die Wassergüteuntersuchung. Die Projektziele umfassen die Messung verschiedener Wasserparameter sowie deren Speicherung, Anzeige und Übertragung.

Die Wikis zu den 5 Sensoren findet ihr hier:

1. **DFRobot Gravity - Analoger pH-Sensor/Meter Kit V2**
[Wiki](#)
2. **DFRobot Dissolved Oxygen Sensor Kit (DO)**
[Wiki](#)
3. **Gravity: Analog TDS Sensor (Total Dissolved Solids)**
[Wiki](#)
4. **Gravity: Analog Turbidity Sensor (Trübung)**
[Wiki](#)
5. **Gravity: Waterproof DS18B20 Temperatur-Sensor**
[Wiki](#)

Bitte prüft zuerst, ob die bestellten Sensoren den obigen Beschreibungen entsprechen. Solltet ihr Sensoren anderer Hersteller verwenden, beachtet, dass diese oft ähnliche Funktionsweisen besitzen. Ein Vergleich der technischen Daten ist in solchen Fällen empfehlenswert.

Grundlagen zu analogen und digitalen Sensoren

Digitale Sensoren

- **Digitale Sensoren** unterscheiden sich von analogen Sensoren darin, dass sie ihre Messwerte bereits als digitale Signale ausgeben. Das bedeutet, dass der Sensor die analoge Messung (z.B. Temperatur, Druck, etc.) selbstständig in einen digitalen Wert umwandelt und an den Mikrocontroller sendet.

Analoge Sensoren

- **Analoge Sensoren** liefern eine Spannung, die proportional oder umgekehrt proportional zum Messwert ist.
 - **Proportional:** Höherer Messwert → höhere Spannung
 - **Umgekehrt proportional:** Höherer Messwert → niedrigere Spannung
- Beispiel: Ein pH-Sensor könnte eine Spannung von 0–3,3 V ausgeben, wobei 0 V einem pH-Wert von 0 und 3,3 V einem pH-Wert von 14 entspricht.

Digitalisierung des Signals

Da Mikrocontroller wie der ESP32 oder Raspberry Pi Pico nur digitale Signale verarbeiten können, wird ein **Analog-Digital-Wandler (ADC)** benötigt. Dieser wandelt die analoge Spannung in einen digitalen Wert um.

Beispiel:

- Eine Spannung von 0–3,3 V wird durch einen 12-Bit-ADC in Werte von 0–4095 umgerechnet.

Berechnung des Messwerts

Nach der Digitalisierung wird der eigentliche Messwert wie folgt berechnet:

$$\text{Messwert} = \text{Digitalwert} \times \text{Proportionalitätsfaktor}$$

Der **Proportionalitätsfaktor** muss durch Kalibrierung des jeweiligen Sensors bestimmt werden.

Hardware-Überblick

Geeignete Mikrocontroller

1. ESP32

- 15 nutzbare ADC-Kanäle → ideal für Projekte mit mehreren analogen Sensoren
- Unterstützt Wi-Fi für Serverkommunikation

2. Raspberry Pi Pico

- 3 eingebaute ADC-Kanäle → externer ADC nötig für mehr als 3 analoge Sensoren

Warum ESP32?

- Für das Projekt sind mindestens 4 ADC-Kanäle erforderlich. Der ESP32 ist deshalb die bessere Wahl, da er mehr integrierte ADC-Kanäle bietet.

Programmiersprachen und Tools

- **MicroPython:** Anfängerfreundlich, ideal für schnelle Entwicklung. Empfehlung: **Thonny IDE**
- **C++ (Arduino):** Breite Unterstützung und zahlreiche Beispielprogramme von DFRobot. Empfehlung: **Arduino IDE**

Hinweis: Die Beispielprogramme von DFRobot sind in C++ geschrieben. Bei Bedarf können diese mit **ChatGPT** oder anderen Tools in MicroPython übersetzt werden. Bitte testet und debuggt diese Übersetzungen sorgfältig.

Schritt-für-Schritt-Plan für die Programmentwicklung

Planung

Legt fest, welche Funktionen das Programm erfüllen soll. Folgende Punkte sind sinnvoll:

1. **Messwerterfassung:** Alle 5 Sensoren auslesen.
2. **Anzeige:** Darstellung der Werte auf einem Display (z. B. OLED, LCD, oder serieller Monitor).
3. **Datenaufzeichnung:** Speicherung der Messwerte auf einer SD-Karte.
4. **Datenübertragung:** Senden der Messwerte an einen Server (z. B. über MQTT oder HTTP).
5. **Zeitstempel:** Ergänzung der Daten mit einer Echtzeituhr (RTC-Modul).

Empfohlene Vorgehensweise

1. Einzelsensor-Programme erstellen

- Testet jeden Sensor separat. Nutzt die bereitgestellten Beispielcodes als Grundlage.
- Stellt sicher, dass jeder Sensor korrekte Daten liefert.

2. Sensoren kombinieren

- Integriert die Programme der einzelnen Sensoren in ein Hauptprogramm.
- Achtet darauf, die richtigen ADC-Kanäle zuzuweisen.

3. Messwerte anzeigen

- Nutzt ein Display (z. B. SSD1306 OLED). Entscheidet, ob eine **grafische Anzeige** oder nur Zahlen notwendig sind.
- Beispiel: Zeigt die Messwerte als Tabelle oder Diagramm an.

4. Daten speichern

- Verwendet ein SD-Kartenmodul, um die Messwerte mit Zeitstempeln zu speichern.
- Format: CSV-Dateien eignen sich gut für eine spätere Analyse.

5. Zeitstempel integrieren

- Schließt ein RTC-Modul (z. B. DS3231) an und synchronisiert die Zeit mit den Messwerten.

6. Daten an einen Server senden

- Nutzt die Wi-Fi-Funktionen des ESP32, um die Daten z. B. über MQTT oder HTTP zu übertragen.
- Richtet einen Server (z. B. einen Raspberry Pi oder eine Cloud-Lösung) ein.

Zusätzliche Hinweise zur Kalibrierung der Sensoren**1. pH-Sensor**

- Kalibriert den Sensor mit Standardlösungen (pH 4, 7, 10).
- Stellt sicher, dass die gemessenen Spannungen korrekt in pH-Werte umgerechnet werden.

2. Dissolved Oxygen (DO)-Sensor

- Führt die Kalibrierung in Wasser mit bekanntem Sauerstoffgehalt durch.
- Beachtet die Temperaturabhängigkeit bei der Kalibrierung.

3. TDS-Sensor

- Kalibriert mit Lösungen bekannter Leitfähigkeit.

4. Trübungs-Sensor

- Kalibriert mit klarem und stark getrübbtem Wasser.

5. Temperatursensor (DS18B20)

- Dieser Sensor benötigt keine Kalibrierung, liefert aber genaue Werte.

Zusammenfassung

Mit dieser Anleitung habt ihr einen klaren Fahrplan zur Entwicklung eures Programms. Folgt dem schrittweisen Ansatz, testet gründlich und achtet auf die Kalibrierung der Sensoren. Bei Problemen könnt ihr ChatGPT für Unterstützung bei der Fehlersuche oder beim Übersetzen von Codes verwenden.

Viel Erfolg bei eurem Projekt! 😊

Für den Fall, das die obigen Links nicht funktionieren:

1. DFRobot Gravity - Analog pH-Sensor/Meter Kit V2
https://wiki.dfrobot.com/Gravity_Analog_pH_Sensor_Meter_Kit_V2_SKU_SEN0161-V2
2. DFRobot Dissolved Oxygen Sensor Kit, galvanische Sonde für Wasserqualitäts-Messung
https://wiki.dfrobot.com/Gravity_Analog_Dissolved_Oxygen_Sensor_SKU_SEN0237
3. Gravity: Analog TDS Sensor/ Meter for Arduino
https://wiki.dfrobot.com/Gravity_Analog_TDS_Sensor_Meter_For_Arduino_SKU_SEN0244
4. Gravity: Analog Turbidity Sensor for Arduino
https://wiki.dfrobot.com/Turbidity_sensor_SKU_SEN0189
5. Gravity: Waterproof DS18B20 Temperature Sensor Kit
https://wiki.dfrobot.com/Waterproof_DS18B20_Digital_Temperature_Sensor_SKU_DFR0198