# Projektarbeit Getränkemischanlage

## Bauteilauswahl

**Wasserpumpen**

* <https://www.conrad.de/de/p/offgridtec-tauchpumpe-12v-dc-10l-0-5-bar-max-5m-pumphoehe-884909107.html>
* Offgridtec Tauchpumpe 12V DC 10l 0,5 bar max. 5m Pumphöhe
* Technische Daten:
  + 10l/min Fördermenge
  + 12V Betriebsspannung
  + 3V Anlaufspannung
  + 0,5 bar Pumpendruck
  + 14,5W Nennleistung
  + 1,2 A DC Strom
  + 1,5m Kabellänge
  + Gewicht 0,2kg
  + Schutzart IP 68
  + Produktart Tauchpumpe
  + 6,53 € / Stück
  + nur für Geschäftskunden erhältlich
* Schutzart IP 68 Vorraussetzung, da die Pumpe im Flüssigkeitsbehälter verbaut ist
* Gewicht 0,2kg zum Vorteil damit Anlage leicht transportiert werden kann und Behälter schnell gewechselt werden können
* 1,5m Kabellänge notwendig aber ausreichend da Pumpen ind en Behältern verteilt sind und Kabel bis zur Steuerung reichen muss
* Eine relativ geringe Leistung von 14,5W ist vorteilhaft, da Steckboard des Raspberry und restliche Elektronik (z.B Transistor) nicht auf große Leistung ausgelegt werden muss, was Kosten spart und sicherer ist
* Eine Fördermenge von 10l/min ist völlig ausreichend. Bsp.-Rechnung: Gemischtes Getränk soll einen Inhalt von 400ml haben. Dafür bräuchte eine Pumpe 2,4s. Wenn dazu noch mehrere Pumpen gleichzeitig laufen (falls so viel Leistung zur Verfügung steht) ist das schnell genug.
* Es bietet sich die Ausführung als Tauchpumpe an, da dabei die Pumpe dauerhaft unterhalb des Wasserpegels steht und dadurch immer mit Wasser gefüllt ist. Dadurch läuft die Pumpe nie „leer“ (saugt Luft an) was dazu führen könnte, dass keine Flüssigkeit angesaugt wird da diese durch die Schwerkraft wieder zurück in den Behälter fliesen würde. Dadurch wird ein Betrieb bei zu hoher Leistung vermieden, der zu Überlast führen könnte und die Elektronik beschädigen könnte. Ein Sensor soll den Pegel des Flüssigkeitbehälters überwachen und somit sicherstellen, dass die Pumpe dauerhaft im Kontakt mit Wasser ist.

**Anschluss Pumpen**

* Der Raspberry Pi selbst kann keine 12V-Pumpen direkt betreiben, da:
  + seine GPIOs nur 3,3V Signale liefern,
  + der Strom pro GPIO auf max. ~16 mA begrenzt ist,
  + und die Pi-eigene Stromversorgung nur 5V mit begrenzter Leistung bereitstellt.
* um die Wasserpumpen ansteuern zu können wird entweder ein Relais-Modul oder ein Transistor verwendet werden

|  |  |
| --- | --- |
| Komponente | Funktion |
| Externe 12V-Stromquelle | Versorgt die Pumpe mit ausreichend Leistung |
| MOSFET-Modul oder Relais-Modul | Ermöglicht das Schalten der Pumpe mit GPIO |
| Freilaufdiode (z. B. 1N4007) | Schützt vor Spannungsspitzen beim Ausschalten (besonders bei Relais) |

|  |  |  |
| --- | --- | --- |
| Variante | Vorteile | Nachteile |
| 1. N-Kanal MOSFET-Schaltung | - Sehr effizient (geringer Spannungsabfall)  - Schnell schaltbar (PWM möglich)- Kein mechanischer Verschleiß  - Leise im Betrieb  - Kompakt, günstig | - Erfordert korrektes Verständnis von Gate-Spannung (Logic-Level nötig)  - MOSFET muss geeignet gewählt sein (z. B. IRLZ44N statt IRF540N) |
| 2. Relaismodul (mechanisch) | - Einfach anzuschließen (auch für Anfänger)  - Galvanische Trennung (je nach Modul)  - Schaltet auch größere Lasten sicher | - Mechanischer Verschleiß (Begrenzte Lebensdauer)  - Klick-Geräusch beim Schalten- Langsamer als MOSFET (nicht für PWM geeignet)  - Höherer Stromverbrauch beim Anziehen |

wir wählen Variante 1: N-Kanal-MOSFET, z. B. IRLZ44N oder IRF3708 (logiklevel-tauglich).

* Vorteile:
  + Schneller, leiser, effizienter als Relais, Strom für Relais evtl zu gering. günstiger
  + Gut für PWM-Steuerung (z. B. Geschwindigkeit)

**Schaltung:**

1. **MOSFET-Modul anschließen:**
   * **Gate (Steuereingang):** Verbinde den Gate-Pin des MOSFET-Moduls über einen 220 Ω Widerstand mit einem GPIO-Pin des Raspberry Pi.
   * **Drain (Ausgang):** Verbinde den Drain-Pin mit dem Minuspol der Pumpe.
   * **Source (Eingang):** Verbinde den Source-Pin mit dem gemeinsamen Massepunkt (GND).
2. **Pumpe anschließen:**
   * **Pluspol:** Verbinde den Pluspol der Pumpe mit dem Pluspol der 12V-Stromversorgung.
   * **Minuspol:** Verbinde den Minuspol der Pumpe mit dem Drain-Pin des MOSFET-Moduls.
3. **Freilaufdiode installieren:**
   * Schalte die Freilaufdiode parallel zur Pumpe, wobei die Kathode (Markierung) mit dem Pluspol der Pumpe verbunden wird.
4. **Gemeinsame Masse:**
   * Verbinde den Masseanschluss der 12V-Stromversorgung mit dem GND des Raspberry Pi, um einen gemeinsamen Massepunkt sicherzustellen

**Testen der Schaltung**

1. **GPIO-Pin konfigurieren:**
   * Setze den verwendeten GPIO-Pin des Raspberry Pi als Ausgang.
2. **Pumpe ein- und ausschalten:**
   * Setze den GPIO-Pin auf HIGH, um die Pumpe einzuschalten.
   * Setze den GPIO-Pin auf LOW, um die Pumpe auszuschalten.
3. **PWM-Steuerung (optional):**
   * Für eine Geschwindigkeitsregelung der Pumpe kannst du PWM-Signale verwenden. Stelle sicher, dass das MOSFET-Modul PWM-kompatibel ist.

**Raspberry Pi Pico**

Als Steuereinheit wird der Raspberry Pi Pico gewählt, da er ein kostengünstiges Mikrocontroller-Board ist das sich besonders für einfache bis fortgeschrittene Elektronik- und Steuerungsprojekte eignet. Er basiert auf dem selbst entwickelten RP2040-Chip und ist für seine Flexibilität, Energieeffizienz und Benutzerfreundlichkeit bekannt.

Vorteile des Raspberry Pi Pico

* Günstiger Preis
* Eigener RP2040-Mikrocontroller ausreichend für viele Embedded-Anwendungen.
* Vielfältige Schnittstellen
* USB-Programmierung (Drag & Drop)
* Niedriger Energieverbrauch
* Große Community & gute Dokumentation
* Kompakte Bauform

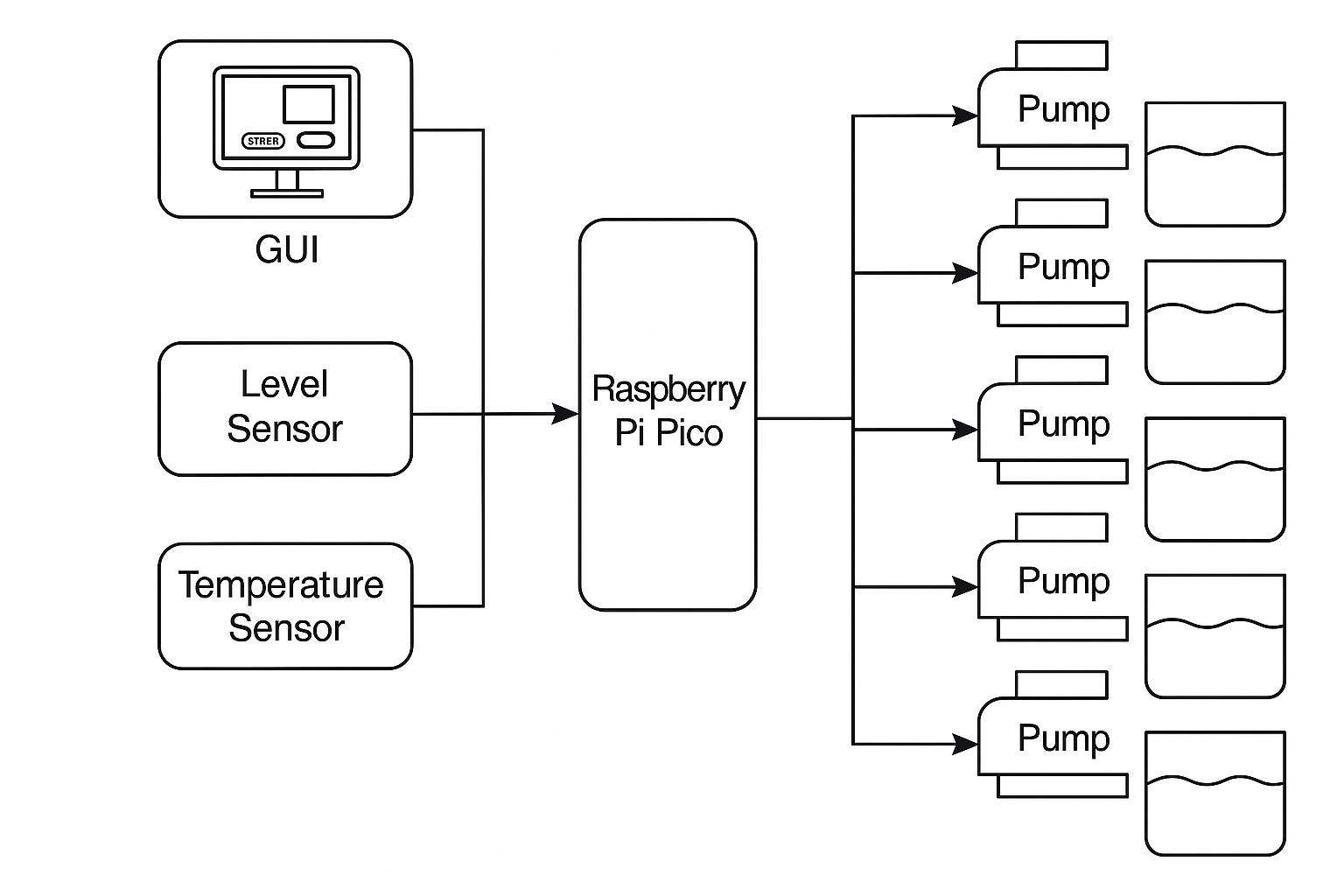
**Freilaufdiode**

**Sensoren**

**Schnittstelle**

## mechanischer Aufbau

## Schaltplan



## Arbeitsweise

gemeinsame ToDo Liste

wöchenteliche Meetings

* Analyse Projektziele und -ergebnisse
* Fortshrittsüberwachung
* Besprechung von Fehlern/Problemen
* Meilensteine/Deadlines setzen

**Zeitplan**

Woche 1:

* mechanischen Aufbau planen
* welche Bauteile werden benötigt?
* elektrischen Aufbau (Blockschaltbild erstellen)
* Bauteile bestellen

GitHub

## Klassen

## Algorithmen