

(Progress)

Automatisches Bewässerungssystem mit dem Raspberry Pi Pico

Dieses Projekt soll Teilnehmende dazu motivieren über ökologisch-nachhaltige Verbindungen zwischen Natur und Technologie ins Gespräch zu kommen. Dabei können Diskussionen zu digitaler Nachhaltigkeit sowie die Möglichkeit Ressourcen mit der Hilfe von Technologien zu sparen.

Dieses Webressource steht unter der Creative Commons Lizenz:
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

- [Einleitung](#)
- [Kosten](#)
- [Installation](#)
- [Bauteile](#)
 - [Das Raspberry Pi Pico 'RP2040'](#)
 - [Kapazitiver Bodenfeuchtesensor v1.2](#)
 - [Der Bewegungssensor 'HC-SR501'](#)
 - [DFplayer Mini Lautsprecher](#)

- DFPlayer MP3 Player
- Relais 5V KY-019
- 3V-5V DC Wasserpumpe

- Zusammenbau
- Der Code
- Reflexion
- Zitation
- Downloads

Einleitung

Kosten

Bauteil	Shoplink	~Stückpreis	URL-Datum
1x Raspberry Pi Pico Set	Shop Berrybase	~6,40 EUR	27.06.2023
1x Breadboard	Shop Berrybase	~1,50 EUR	27.06.2023
1x Jumper, Dupont Kabel Set	Shop Berrybase	~4,40 EUR	27.06.2023
2x	Shop AZ Delivery	~0,00 EUR	27.06.2023
1x	Shop Berrybase	~0,00 EUR	27.06.2023
1x	Shop AZ Delivery	~0,00 EUR	27.06.2023
1x	Shop Berrybase	~0,00 EUR	27.06.2023
1x	Shop AZ Delivery	~0,00 EUR	27.06.2023
		~00,00 EUR	

“ Info

Als Zusatz können extra Jumper nur für das Breadboard verbaut werden, die das Kabelmanagement vereinfachen

- Jumper Set 'nur' für das Breadboard | [Shop Barrybase](#) | ~3,70 EUR

Installation

Wir programmieren das Raspberry Pi Pico mit der Script-Sprache Micropython in der freien Entwicklungsumgebung (IDE) Thonny. Dazu nutzen wir einen Computer oder Laptop.

Was ist eine IDE?

Eine IDE (Integrated Development Environment) ist ein Programm, das beim Schreiben, Testen und Ausführen von Code hilft. Sie bietet viele nützliche Werkzeuge an einem Ort, wie z.B.:

- Texteditor zum Schreiben von Code
- Debugger zum Finden von Fehlern
- Terminal zum Ausführen des Codes

Was ist Thonny?

Thonny ist eine einfache und benutzerfreundliche IDE, die speziell für Python entwickelt wurde. Sie ist gut geeignet für alle Menschen, welche das erste Mal anfangen zu Programmieren, so wie ich zum Beispiel, weil sie eine einfache Benutzeroberfläche hat und viele hilfreiche Funktionen bietet.

“ Übung

Versuche nach der Installation, siehe unten, über die IDE Thonny eine Bibliothek zu installieren, zum Beispiel die Bibliothek 'NeoPixel'.

- Thonny öffnen ->
- Menü -> Tools -> Manage Plug-ins
- Suche nach 'neopixel' -> installieren

drawing drawing

Installation von Thonny auf verschiedenen Betriebssystemen

Installation auf Windows

- Gehe zur offiziellen [Thonny-Website](#).
- Klicke auf den Download für Windows Button.
- Lade die Installationsdatei herunter und öffne sie, wenn der Download abgeschlossen ist.
- Folge den Anweisungen des Installationsassistenten, um Thonny zu installieren.
- Sobald die Installation abgeschlossen ist, kannst du Thonny über das Startmenü öffnen.

Installation auf Linux (Ubuntu)

- Öffne das Terminal auf deinem Ubuntu-System.
- Gib den folgenden Befehl ein, um das Paket zu aktualisieren:

```
sudo apt update
```

- Installiere Thonny, indem du den folgenden Befehl eingibst:

```
sudo apt install thonny
```

- Warte, bis die Installation abgeschlossen ist. Danach kannst du Thonny im Anwendungsmenü finden und starten.

Installation auf macOS

- Besuche die Thonny-Website.
- Klicke auf den Download für macOS Button.
- Lade die .dmg-Datei herunter und öffne sie, wenn der Download abgeschlossen ist.
- Ziehe das Thonny-Symbol in den Programme-Ordner, um die Installation abzuschließen.
- Öffne Thonny, indem du es im Programme-Ordner findest oder über Spotlight suchst.

Bauteile

Das Raspberry Pi Pico

'RP2040'

drawing

- Der RP2040 Microcontroller ('*Miniprozessor*') hat 2 MByte Flash Speicher
- Das Raspberry Pi Pico kann mit 1,8V bis 5,5V Spannung betrieben werden
- Mit den 'Pinouts' (*Stiftleisten*) kann das Pico direkt programmiert werden
- Diese Pinouts werden auch als GPIOs bezeichnet (von GP0 bis GP28)
- Bei der Programmierung gibst du dabei die GP-Nummer an, also bei GP28 wäre dies die Zahl 28

Hier ein Beispiel bei der Programmierung in Micropython:

Pin = 28

“ Note

Manchmal kann es verwirrend sein, welche Nr. genau für die Programmierung verwendet wird, da es die PIN- und GP-Nummer gibt. Für uns als angehende Entwickler*innen sind aber nur die GP-Nummern wichtig, welche in der oberen Abbildung in grünen Kästchen dargestellt sind!

Bauteile

Kapazitiver Bodenfeuchtesensor v1.2

PXL_20230627_101549644.jpg

Der Bewegungssensor 'HC-SR501'

drawing drawing

- Der Sensor hat einen Stromverbrauch von rund 65 mA (mA = Milliampere)
- Außerdem benötigt der Sensor 5V Spannung (V = Volt)
- Der Bewegungssensor wird auch PRI-Sensor genannt
- PRI ist dabei eine Abkürzung und bedeutet Pulse Repetition Interval (Deut.: Intervall der Impulsfolge)
- Der Bewegungssensor besitzt 2 orangene Drehschalter, diese werden auch Potenziometer genannt

“ Info

Die Potenziometer an dem Bewegungsmelder 'HC-SR501' ermöglichen es der benutzenden Person, die Empfindlichkeit (*Eng. Sensitivity Adjust*) des Sensors anzupassen. Der zweite Potenziometer bestimmt die zeitliche Verzögerung (*Eng. Time Delay Adjust*) bis eine neue Bewegung erfasst wird.

Bauteile

DFplayer Mini Lautsprecher

PXL_20230627_101444060.jpg

Bauteile

DFPlayer MP3 Player

PXL_20230627_104540295.jpg

Bauteile

Relais 5V KY-019

PXL_20230627_101339919.jpg

Bauteile

3V-5V DC Wasserpumpe

PXL_20230627_101413218.jpg

Zusammenbau

PXL_20230627_110646945.jpg

anteiltung-pico-pump-soilmoisture-prisensor-dfplayer_04_bb.png

Der Code

Hier ist ein Beispielcode für eure Programmierung in Micropython in der IDE Thonny auf dem Raspberry Pi Pico.

```
from machine import Pin
import time

# Initialisiere den GPIO-Pin für das Relais (hier GPIO 15)
relay = Pin(15, Pin.OUT)

# Funktion zum Einschalten der Pumpe
def turn_on_pump():
    relay.value(1)
    print("Pumpe eingeschaltet")

# Funktion zum Ausschalten der Pumpe
def turn_off_pump():
    relay.value(0)
    print("Pumpe ausgeschaltet")

# Hauptprogramm
try:
    while True:
        # Schalte die Pumpe für 5 Sekunden ein
        turn_on_pump()
        time.sleep(5)

        # Schalte die Pumpe für 10 Sekunden aus
        turn_off_pump()
        time.sleep(10)

except KeyboardInterrupt:
    # Schalte die Pumpe aus, wenn das Programm beendet wird
    turn_off_pump()
    print("Programm beendet")
```


Reflexion

Einleitung

Mit diesem Making und Coding Projekt zu einem automatischen Bewässerungssystem, können junge Menschen durch das zusammenstecken der verschiedenen Bauteile ein kleines automatisches Bewässerungssystem bauen. Nachdem alles Zusammengebaut wurde, kann an einem Computer mit der Programmierumgebung (IDE) Thonny der Code, das Programm korrigiert werden. Beispielsweise wie häufig und wie lange die Pumpe pumpe soll. Mit diesem Projekt könnt ihr zum Beispiel ein Beet, Kreuterpflanzen usw. bewässern und zu ökologischer Nachhaltigkeit ins Gespräch kommen.

1 Bauteile vorbereiten

Bereitet am besten vor dem Projekt die Bauteile etwas vor. Wenn euer Projekt besonders niedrigschwellig sein soll, wäre es gut, wenn ihr schon die Pins an das Raspberry Pi Pico anlötet. Außerdem könnt ihr schon einmal Micropython (<https://www.raspberrypi.com/documentation/microcontrollers/micropython.html>) auf das Raspberry Pi Pico installieren. Hilfreich wäre auch etwas Isolierklebeband (Z. B. <https://www.berrybase.de/isolierband-10er-pack-15mm-10m-farblich-sortiert>) mit dem ihr die Steckverbindungen der Jumper 'sichern' könnt. Die jungen Menschen oder Teilnehmenden können dann anhand der Vorlage, siehe Zusammenbau, die Bauteile zusammensetzen. Stellt zum Testen zum Beispiel ein Glas zur Verfügung und ein weiteres leeres Glas.

2 Nachdem Zusammenbau

Nachdem alle Menschen ihren Bausatz zusammen gebaut haben, können diese mit einem Micro-USB-Kabel und einem USB-Netzteil mit (1A und 5V) das Bewässerungssystem testen. Die Pumpe kann hier einfach das Glas mit was gelegt/Stellt werden. Den Schlauch führt ihr in das leere Glas. Nun könnt ihr das Raspberry Pi Pico an das USB-Netzteil anschließen.

3 Verbessern und anpassen des Codes

Nun, nachdem ihr das Bewässerungssystem etwas getestet habt, könnt ihr versuchen, den Code/das Programm etwas anzupassen. Zum Beispiel die Dauer des Pumpens von Wasser, oder wie häufig die Pumpe pumpen soll (Z. B. alle 7 Tage).

4 Beim 'Machen' über Nachhaltigkeit sprechen

Während des Projektes könnt ihr mit den Menschen in Gespräch kommen. Sprecht vielleicht über Wasserverbrauch und wie Technologie helfen kann, Wasser zu sparen oder wo die Teilnehmenden vielleicht ein solches System einsetzen würden. Ggf. ergeben sich neue gestalterische Projekte in

eurer Einrichtung/Institution. Das Projekt ist extra so konzeptioniert, dass es auch mit einem relativ geringen ökonomischen Aufwand umgesetzt werden kann. Dies ist nur dank Open Source, also offen einsehbaren Programmiercode möglich und Menschen, die diesen Code meist ehrenamtlich erstellen und bereitstellen. Hier kann sogar das Thema Gemeinwohlökonomie thematisiert mit den Teilnehmenden thematisiert werden

Zitation

Autor*in

- Michelle Pröhl (keine/weibl.)
- Kontakt-Blatt Fit for Gestalt WAS
- Kontakt-Blatt JFF
- E-Mail: michelle.proehl@jff.de

Lizenz

- Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

Veröffentlichung

- 12.07.2023
- JFF – Jugend Film Fernsehen e. V. im Rahmen der Fit for GestaltWas Initiative der Telekomstiftung

APA-Zitierweise

Pröhl, M. (2023, July 12). Automatisches Bewässerungssystem mit dem Raspberry Pi Pico. JFF – Jugend Film Fernsehen e. V. (Berlin). Fit for Gestalt Was Initiative. Abgerufen von <https://wiki.fitforgestaltwas.org/books/bewegungsmelder-mit-led-streifen>

Harvard-Zitierweise

Pröhl, M. (2023) Automatisches Bewässerungssystem mit dem Raspberry Pi Pico. [Online] Berlin: JFF – Jugend Film Fernsehen e. V. for Fit for Gestalt Was Initiative. Verfügbar unter: <https://wiki.fitforgestaltwas.org/books/bewegungsmelder-mit-led-streifen> (Zugriff am: 12 July 2023).

LaTeX bibtex-Zitationscode

Hier ist ein Beispielcode für die Zitation für LaTeX in der bibtex-Schreibweise

```
@misc{proehl2023,  
  author = {Pröhl, Michelle},  
  title = {Automatisches Bewässerungssystem mit dem Raspberry Pi Pico},  
  year = {2023},  
  address = {Berlin},
```

```
publisher = {JFF - Jugend Film Fernsehen e. V.},  
note = {Für: Fit for Gestalt Was Initiative},  
url = {https://wiki.fitforgestaltwas.org/books/bewegungsmelder-mit-led-streifen},  
urldate = {2023-07-12}  
}
```

Downloads