

Vorbemerkung

Über Unsere Firma

WayinTop, Your Top Way to Inspiration, ist ein professioneller Hersteller von über 2.000 Open Source-Motherboards, -Modulen und -Komponenten. Vom Entwerfen von Leiterplatten, Drucken, Löten, Testen, Debuggen und Anbieten von Online-Tutorials, WayinTop hat es sich zur Aufgabe gemacht, die wunderbare Welt der eingebetteten Elektronik zu erkunden und zu entmystifizieren. Einschließlich, aber nicht beschränkt auf Arduino und Raspberry Pi. Wir sind bestrebt, die am besten gestalteten Produkte für Hersteller aller Altersgruppen und Könnensstufen herzustellen. Unabhängig von Ihrer Vision oder Ihrem Fähigkeits Level sind unsere Produkte und Ressourcen darauf ausgelegt, die Elektronik besser zugänglich zu machen. WayinTop wurde 2013 gegründet und ist mittlerweile auf über 100 Mitarbeiter und eine über 50.000 Quadratmeter große Fabrik in China angewachsen. Mit unseren unermüdlichen Bemühungen, wir haben das Angebot auch um Werkzeuge, Ausrüstungen, Verbindungssätze und verschiedene Heimwerkerprodukte erweitert, dass wir sorgfältig ausgewählt und getestet haben.

US Amazon Store Homepage:

<https://www.amazon.com/shops/A22PZZC3JNHS9L>

CA Amazon Store Homepage:

<https://www.amazon.ca/shops/A22PZZC3JNHS9L>

UK Amazon Store Homepage:

<https://www.amazon.co.uk/shops/A3F8F97TMOROP>

DE Amazon Store Homepage:

<https://www.amazon.de/shops/A3F8F97TMOROP>

FR Amazon Store Homepage:

<https://www.amazon.fr/shops/A3F8F97TMOROP>

IT Amazon Store Homepage:

<https://www.amazon.it/shops/A3F8F97TMOROP>

ES Amazon Store Homepage:

<https://www.amazon.es/shops/A3F8F97TMOROP>

JP Amazon Store Homepage:

<https://www.amazon.co.jp/shops/A1F5OUAXY2TP0K>

1 Kanal Pflanzenbewässerungssystem mit Raspberry Pi



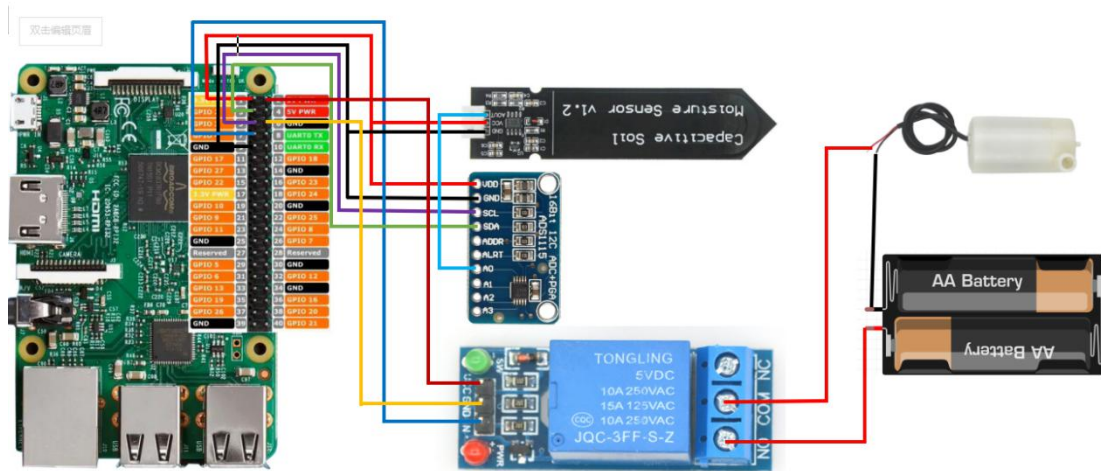
Überblick

Die folgende Anleitung zeigt Ihnen, wie Sie ein Bewässerungssystem einrichten, das die Pflanze oder Blume intelligent überwacht und automatisch bewässert.

Erforderliche Teile

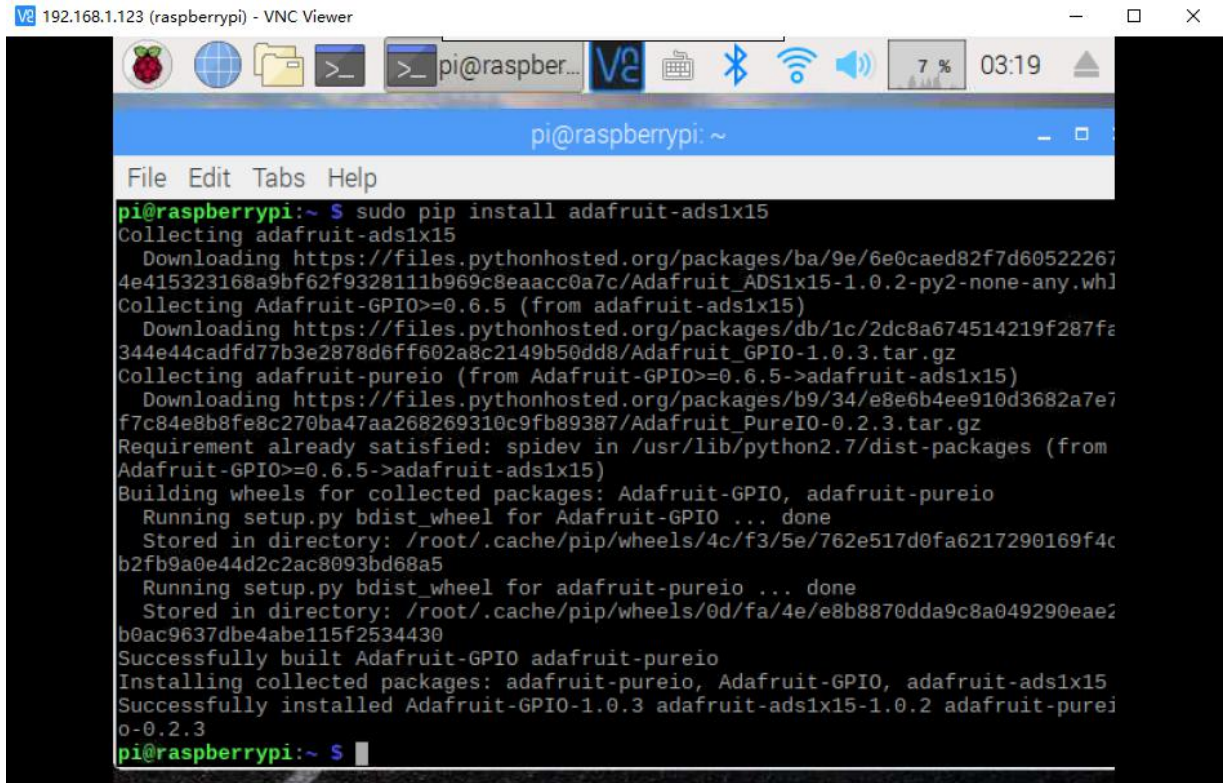
- 1 x Raspberry Pi Board
- 1 x ADS1115 Analog Digital Converter
- 1 x Kapazitiver Bodenfeuchtesensor
- 1 x Einkanal-Relais
- 1 x Wasserpumpe
- 15 x Jumper Drähte

Verbindung Diagramm



Wie man mit Raspberry Pi einrichtet?

Schritt 1: Fügen Sie die ADS1115-Modulbibliothek nach Abschluss der Verbindung zum Raspberry Pi-System hinzu, geben Sie im Raspberry Pi-Terminalfenster `'sudo pip install adafruit-ads1x15'` ein und drücken Sie die Eingabetaste, wie unten gezeigt..



```

192.168.1.123 (raspberrypi) - VNC Viewer
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ sudo pip install adafruit-ads1x15
Collecting adafruit-ads1x15
  Downloading https://files.pythonhosted.org/packages/ba/9e/6e0caed82f7d605222674e415323168a9bf62f932811b969c8eaacc0a7c/Adafruit_ADS1x15-1.0.2-py2-none-any.whl
Collecting Adafruit-GPIO>=0.6.5 (from adafruit-ads1x15)
  Downloading https://files.pythonhosted.org/packages/db/1c/2dc8a674514219f287fa344e44cadfd77b3e2878d6ff602a8c2149b50dd8/Adafruit_GPIO-1.0.3.tar.gz
Collecting adafruit-pureio (from Adafruit-GPIO>=0.6.5->adafruit-ads1x15)
  Downloading https://files.pythonhosted.org/packages/b9/34/e8e6b4ee910d3682a7e7f7c84e8b8fe8c270ba47aa268269310c9fb89387/Adafruit_PureIO-0.2.3.tar.gz
Requirement already satisfied: spidev in /usr/lib/python2.7/dist-packages (from Adafruit-GPIO>=0.6.5->adafruit-ads1x15)
Building wheels for collected packages: Adafruit-GPIO, adafruit-pureio
  Running setup.py bdist_wheel for Adafruit-GPIO ... done
  Stored in directory: /root/.cache/pip/wheels/4c/f3/5e/762e517d0fa6217290169f4cb2fb9a0e44d2c2ac8093bd68a5
  Running setup.py bdist_wheel for adafruit-pureio ... done
  Stored in directory: /root/.cache/pip/wheels/0d/fa/4e/e8b8870dda9c8a049290eae2b0ac9637dbe4abe115f2534430
Successfully built Adafruit-GPIO adafruit-pureio
Installing collected packages: adafruit-pureio, Adafruit-GPIO, adafruit-ads1x15
Successfully installed Adafruit-GPIO-1.0.3 adafruit-ads1x15-1.0.2 adafruit-pureio-0.2.3
pi@raspberrypi:~ $
  
```

Schritt 2: Geben Sie nach Abschluss des Downloads den folgenden

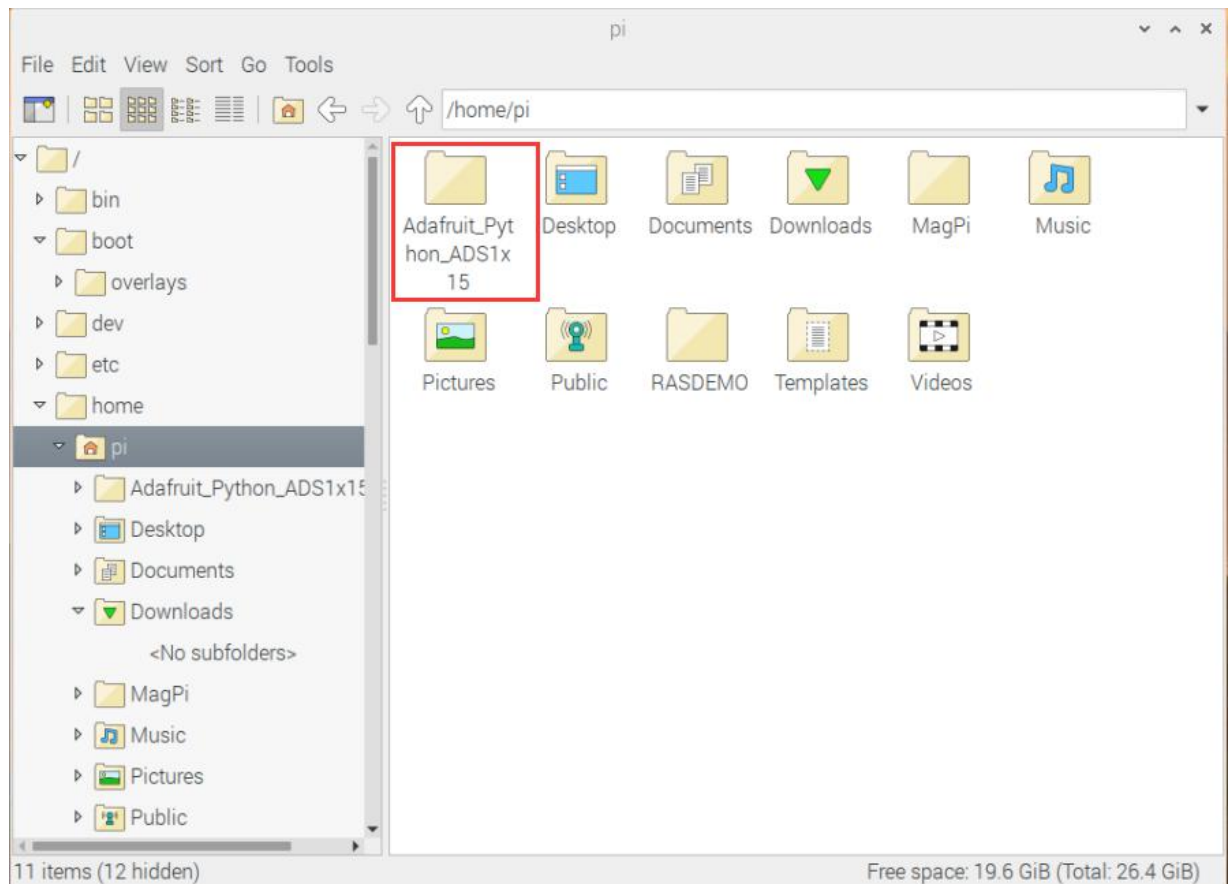
Befehl in die Befehlszeile ein:

Git-Klon https://github.com/adafruit/Adafruit_Python_ADS1x15.git

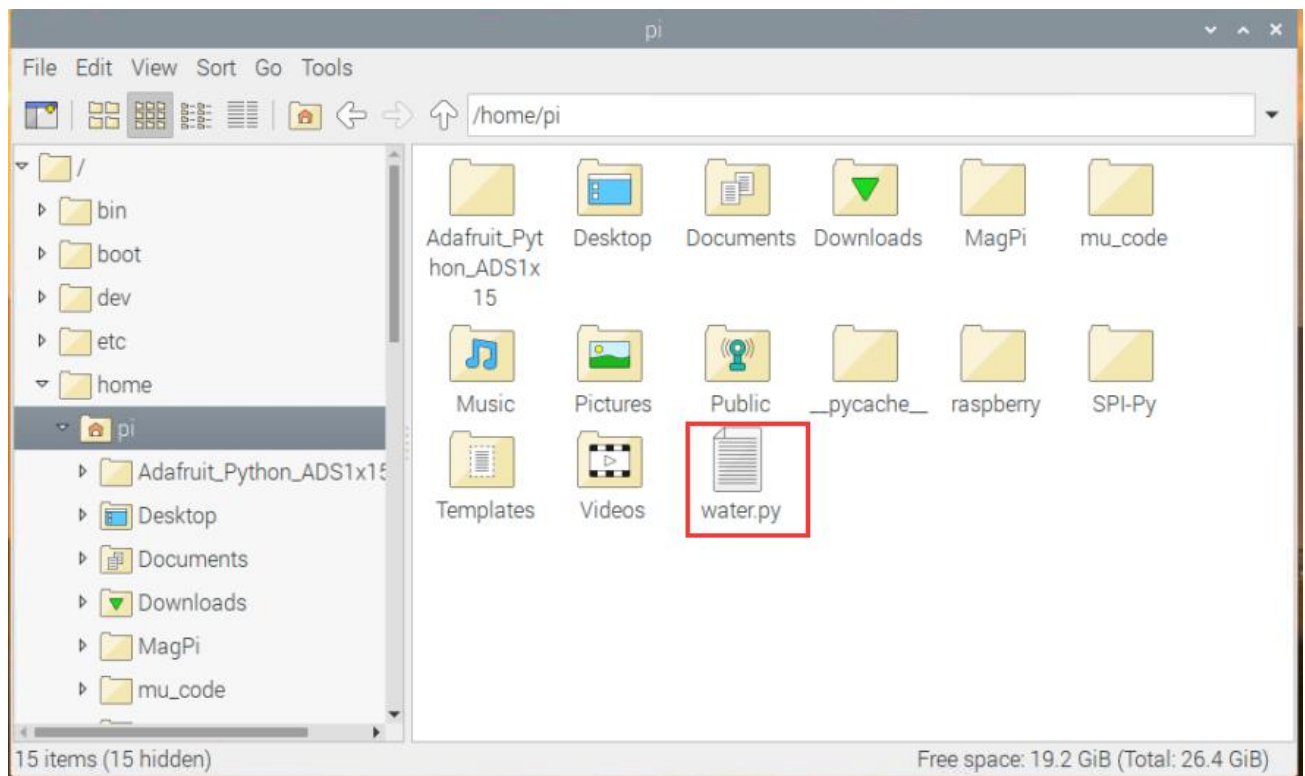
Herunterladen einer weiteren Datei von ADS1115. Nach Abschluss

des Downloads wird die unten gezeigte Datei in Ihrem

pi-Dateiverzeichnis angezeigt.



Schritt 3: Nachdem der Download abgeschlossen ist, erstellen Sie eine neue **water.py** Datei im pi-Verzeichnis wie unten gezeigt.



Schritt 4: Kopieren Sie den folgenden Code in **water.py** file.

```
import RPi.GPIO as GPIO
import time
import Adafruit_ADS1x15
import math
adc = Adafruit_ADS1x15.ADS1115()
GAIN = 1
PIN=7
def setup():
    GPIO.setmode(GPIO.BOARD)
    GPIO.setup(PIN, GPIO.OUT)
    GPIO.output(PIN, GPIO.HIGH)
    time.sleep(0.1)

values = [0]*100
def loop():
    while True:
        for i in range(100):
```

```
        values[i] = adc.read_adc(0, gain=GAIN)
    print(max(values))
    if (max(values))>20000:
        GPIO.output(PIN, GPIO.LOW)
        print("ON")
        print(PIN)
        time.sleep(0.1)
    else:
        GPIO.output(PIN, GPIO.HIGH)
        print("OFF")
        print(PIN)
        time.sleep(0.1)

def destroy():
    GPIO.output(PIN, GPIO.HIGH)
    GPIO.cleanup()

if __name__ == '__main__':
    setup()
    try:
        loop()
    except KeyboardInterrupt:
        destroy()
```

Schritt 5: Führen Sie das Programm aus, indem Sie den Befehl **python water.py** im Terminal eingeben.

Code-Interpretation:

1. Wir müssen die vier Module importieren, dies sind: das

Raspberry Pi GPIO-Modul, das Verzögerungsfunktionsmodul, das ADS1115-Modul und das Mathematikmodul.

```
import RPi.GPIO as GPIO
```

```
import time
```

```
import Adafruit_ADS1x15
```

```
import math
```

2. Instantieren Sie ein adc-Objekt mit dem ADS1115-Modul, um die Sensordaten später auszulesen. Definieren Sie dann eine Positionsvariable GAIN und den physischen Pin 7 des Raspberry Pi.

```
adc = Adafruit_ADS1x15.ADS1115()
```

```
GAIN = 1
```

```
PIN=7
```

3. Stellen Sie den 7-Pin-Pin-Modus, den Pin-Eingangs- / Ausgangsmodus und den Initialisierungs-Pin 7 in der Funktion `setup()` ein und verzögern Sie dann die Entprellung um 0,1 Sekunden.

```
def setup():
```

```
    GPIO.setmode(GPIO.BOARD)
```

```
GPIO.setup(PIN, GPIO.OUT)
```

```
GPIO.output(PIN, GPIO.HIGH)
```

```
time.sleep(0.1)
```

4. Schließlich wird die spezifische Funktion in die `loop()` Funktion implementiert, und der Wert des Sensors wird durch die if-Anweisung als größer als 20000 beurteilt. Wenn es größer als 20000 ist, wird das Relais von der Pumpe betätigt. Wenn es kleiner als ist, wird das Relais ausgeschaltet und die Pumpe funktioniert nicht mehr.

```
def loop():
```

```
    while True:
```

```
        for i in range(100):
```

```
            values[i] = adc.read_adc(0, gain=GAIN)
```

```
        print(max(values))
```

```
        if (max(values))>20000:
```

```
            GPIO.output(PIN, GPIO.LOW)
```

```
            print("ON")
```

```
            print(PIN)
```

```
            time.sleep(0.1)
```

else:

```
GPIO.output(PIN, GPIO.HIGH)
```

```
print("OFF")
```

```
print(PIN)
```

```
time.sleep(0.1)
```

Achtung: Das hier abgebildete 20.000er-Set muss auf die Feuchtigkeit eingestellt werden, die Ihre eigenen Blumen benötigen. Dies wird nur anhand eines Beispiels von 20000 erläutert.

5. Das letzte Stück Code ist, wo die Funktion beginnt und endet.

Wenn das Programm ausgeführt wird, können Sie 'Ctrl+c' drücken, um die Codeausführung zu beenden.

```
if __name__ == '__main__':
```

```
    setup()
```

```
    try:
```

```
        loop()
```

```
    except KeyboardInterrupt:
```

```
        destroy()
```