Project Report: Plant Watering System | Colby Alley

Project Description:

Build an automatic plant watering system that will dispense water when the Raspberry Pi Pico detects no moisture.

Project Citations:

ChatGPT - Assisted in the code development during this project.

Instructables Automatic Raspberry Pico W Watering System – Demonstrated the schematic to build the circuit and gave helpful tips when performing this project.

Project Thoughts and Struggles:

During this project, I would like to say I did not struggle as much as I expected to. The building of the circuit is straight-forward. However, the fundamentals of this project were very important to my understanding of how code works when using actual components within a breadboard. Until this point, I was not sure how code manipulated parts on a circuit. After completing this project, I can understand more real-life scenarios and how software engineering plays a crucial role in daily tasks. When it comes to this project, I believe it was simple but very informative. The main struggle was the code involved in this project. I am more familiar with C++ than Python (which was used during this project), however, getting used to the syntax was not the difficult part. The difficult part was understanding how the Raspberry Pi Pico board reads pins and understands different components. I would say this was the most fascinating part of this project. This has sparked my interest in doing my own personal projects to fully understand how components are read through code. It also has influenced me to begin to learn Python more in depth as well as other programming languages early to gain better skills for future projects.

Project Conclusion:

I would like to say this was my favorite project of my semester. It broadened my understanding of circuit development and how code impacts components. This has influenced me to dip my feet deeper into personal projects and understand more about how components can be manipulated. Thank you for taking the time to read this report.

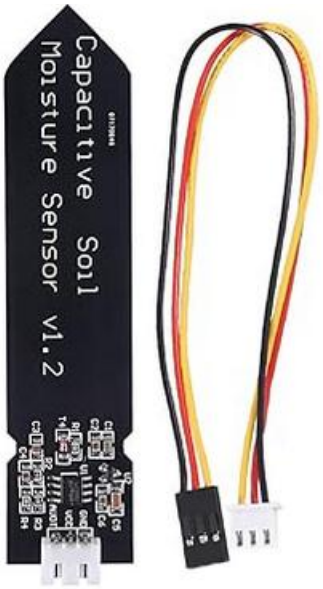
Main Components:

Images of the main components that were important for the function of this project.

Relay Module

link

Soil Sensor



link

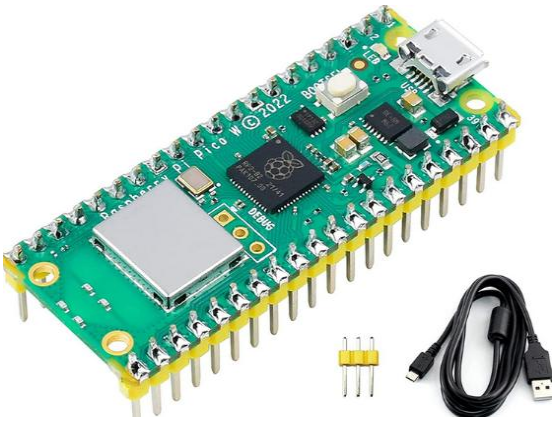
Water Pump

http

Tubing

link

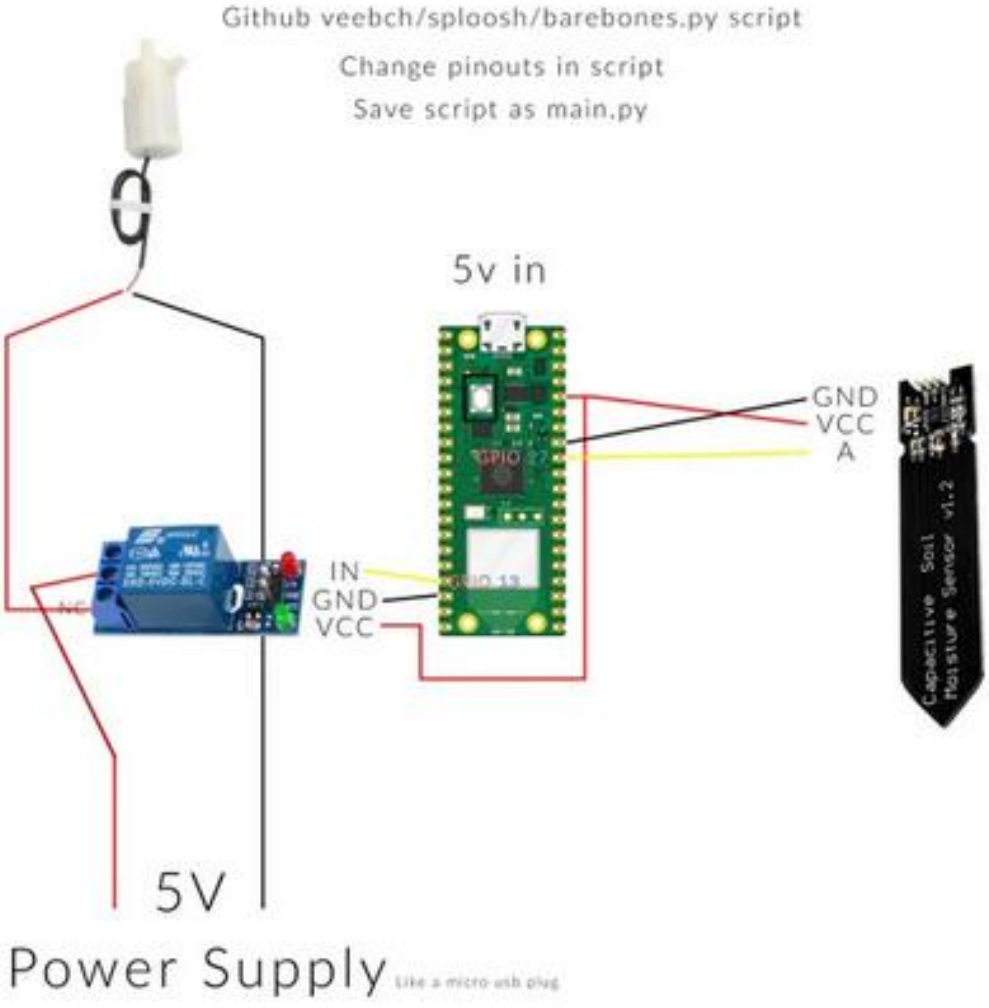
Raspberry Pi Pico H



link

Assembly:

Schematic to show how the main parts of this project are connected.



link

Code Involvement:

The code to make this project work as well as some notes to help explain the purposes.

```
1 from machine import ADC, Pin
2 import utime
```

Importing libraries and classes that are already built onto our Raspberry Pico board.

```
6 relay = Pin(13,Pin.OUT)
7 moisture_sensor = ADC(27)
8 relay.value(0)
9 thresh = 30000
```

Identifies three variables: relay, moisture sensor, and thresh. The relay variable points to pin 13 on our Raspberry Pico board. The moisture sensor variable configures pin 27 to read values from our soil sensor. The *relay.value(0)* function is simply setting the relay to 0 initially, meaning the relay is off. Finally, the thresh variable defines a specific threshold value for our moisture sensor reading, which is 30,000.

```
11 while True:
12     reading = moisture_sensor.read_u16()
13 
14     voltage = 3.3*reading/65535
15 
16     print("Voltage:",voltage)
17     print("Moisture:", reading)
```

This section of code begins the infinite loop to monitor the soil moisture. I also defined a reading variable and voltage variable. The reading variable reads a 16-bit value from our soil sensor. The voltage variable converts that value to voltage, assuming we are connected to our 3.3 voltage rail. Lines 16 and 17 print our voltage and soil sensor readings.

```
19      if reading > thresh:
20          print ("Water detected. Pump off.")
21          relay.value(0)
22
23      if reading < thresh:
24          relay.value(1)
25          print ("No water. Turning on pump!")
26      print ("----")
27
28      utime.sleep(1)
```

This is our logic to check the soil sensor moisture levels. If the reading is above our threshold, the pump will remain off. If our reading is less than the threshold, the pump will activate, pumping water. The *utime.sleep(1)* function tells us this happens in 1 second intervals.

Image of Circuit: