ECE 1000 Final Report: Plant-Watering System

Joyhl Gwinn , GW Songer, Zachary Jones
ECE-1000
Tennessee Tech University
Cookville, United States of America
jgwinn42@tntech.edu, gwsong42@tntech.edu, zjones42@tntech.edu

The name of our project is the Plant-Watering System. We chose this type of system as our project because of how important plants are in our lives, especially when it comes to crops. Our system makes watering plants more efficient and convenient. This is possible by connecting a moisture sensor, water pump, and relay together.

I. Introduction

We chose this topic because we view this system is really practical and effective in terms of watering plants and small crops. A user could use this system to water a small tomato plant without having to watering it frequently and do other activities. Our team members include Joyhl Gwinn, GW Songer, and Zachary Jones. Joyhl's major is Computer engineering while Goerge and Zach's are Electrical engineering. The sources we used to develop the project are discussed in the next section.

II. Background

The source we used to learn how to wire the relay is an article named *5V Single-Channel Relay Module.* This article included information needed to know which wires to hook up to the relay. It taught us the function of each pin. Another source we used was code from JC Williams. We used this code as a basis for our finished code which is discussed in the Project Description.

III. Project Description and formulation

There are four main components that make the project work. The moisture sensor, which detects if there is low or high moisture. If there is low moisture, the pico board will send voltage to the relay. The relay, which acts like a button and protects the water pump from getting fried. The power was connected the common connection on the relay while the water pump was connected to commonly closed pin of the relay. The water pump, pumps water from a container into the plant. Once there is enough moisture, the sensor will detect the change and pico board will stop sending five volts to the relay. The code used was made using Thonny. In terms of code, line 4 and 5 tells the pi pico that the input from the sensor will be received through pin 28 and the relay is connected to pin 13. Lines 8 and 9 set the values received from the sensor at maximum and minimum moisture. Lines 11 through 17 change the value received from the moisture sensor into a percentage. Lines 19 through 30 display the moisture percentage value to the shell and turns the relay off and on based on the moisture level that was converted into a percentage.

IV. Discussion and results

The overall project succeeded in terms of its task. The moisture sensor and relay worked how it was supposed to. The water pump pumped water at a very slow pace but ultimately watered the plant. We later got a replacement water pump, which sped up the watering system. Joyhl enjoyed coding the pico board, GW enjoyed assembling the project, Zach enjoyed making the report and assisting with assembly.

V. Conclusion

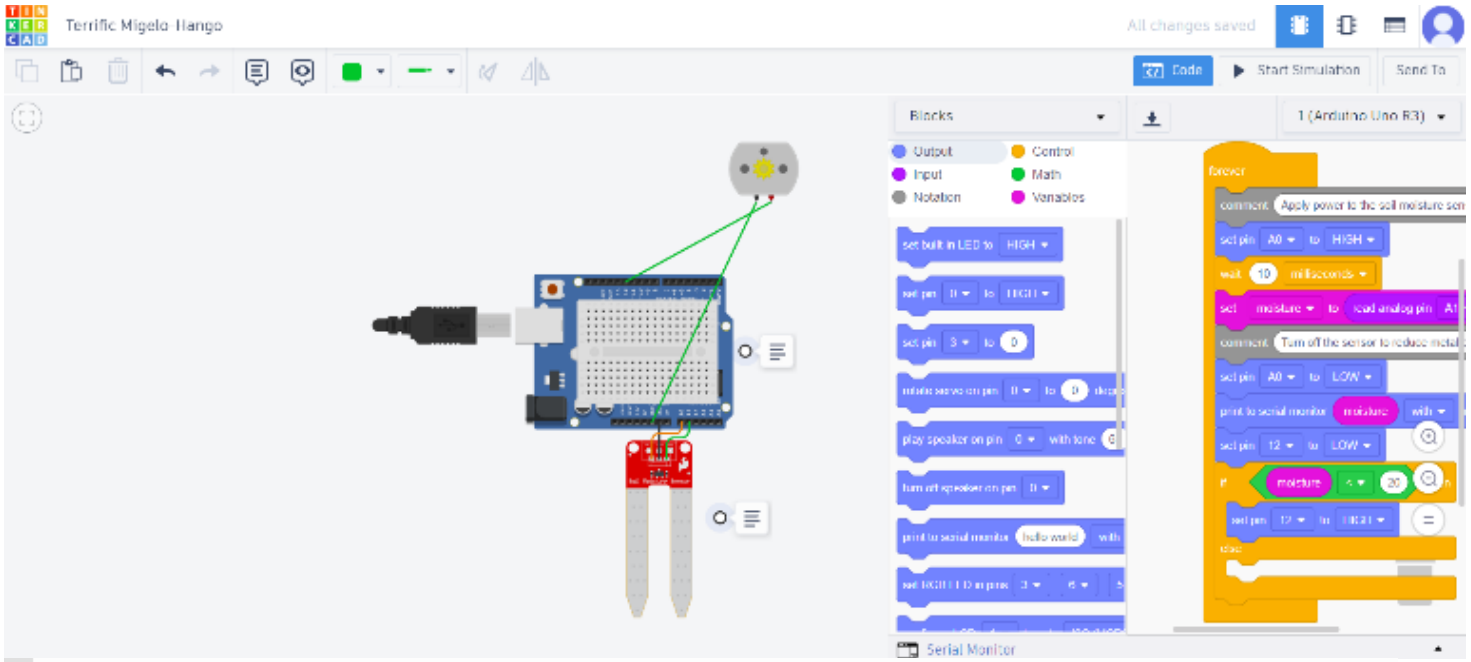
In conclusion, the Plant-Watering System designed to make the watering process more self-proficient and effective was success in terms of functionality. The sensor, relay, water pump, and code for the pi pico worked well together to water the plant. The skills acquired by the group were learning how to code in Python, assembling a device with a Raspberry pi pico board and other electrical components, and learning how to collaborate well with each other to finish the project.

References

[1] "5V Single-Channel Relay Module." Components 101, 21 December 2020, 5V Single-Channel Relay Module - Pin Diagram, Specifications, Applications, Working (components101.com)

[2] Williams, JC. "ECE_1000_Soil_Moisture_Sensor_Example.py." GETHUB, 15 April 2024, ECE-1000-Spring-2024-Final-Project-Insert-Project-Name/Example Micropython Codes/ECE_1000_Soil_Moisture_Sensor_Example.py at main · JCWilliams1003/ECE-1000-Spring-2024-Final-Project-Insert-Project-Name · GitHub

Pictures of TinkerCad and Python Code:




```python
from machine import ADC, Pin,
import utime
#tells the pi pico that the input from the moisture sensor will be recieved through pin 28 and that the relay will be connected to pin
soil_probe = ADC(Pin(28))
relay = Pin(13, Pin.OUT)

#sets the values recieved from the sensor at maximum and minimum moisture
max_moisture = 27574
min_moisture = 57100
#changes the value recieved from the moisture sensor into a percentage
def get_moisture_percentage(moisture_level):
    point_1_x = min_moisture
    point_2_x = max_moisture
    point_1_y = 0
    point_2_y = 100
    m = ((point_2_y - point_1_y) / (point_2_x - point_1_x))
    return int((m*moisture_level) - (m*min_moisture) + point_1_y)
#displays the moisture percentage value to the shell and turns the relay off and on based on that percentage
while True:
    moisture_level = soil_probe.read_u16()

    moisture_level_percentage = get_moisture_percentage(moisture_level)

    print(moisture_level_percentage)

    if moisture_level_percentage <= 20:
        relay.value(1)
    if moisture_level_percentage > 20:
        relay.value(0)
    utime.sleep(0.8)
```