embedded-plant-waterer

Monitor moisture levels in soil and automatically water a house plant.

Abstract

I wanted to spruce up my room and knew that I needed some sort of techno-natural themed decorations. As I brain-storming embedded systems ideas, it came to me that an automated plant watering system would be the ideal addition to my minimalist room. This readme file details the kit I used to complete my project and how I turned it from an idea into a reality.

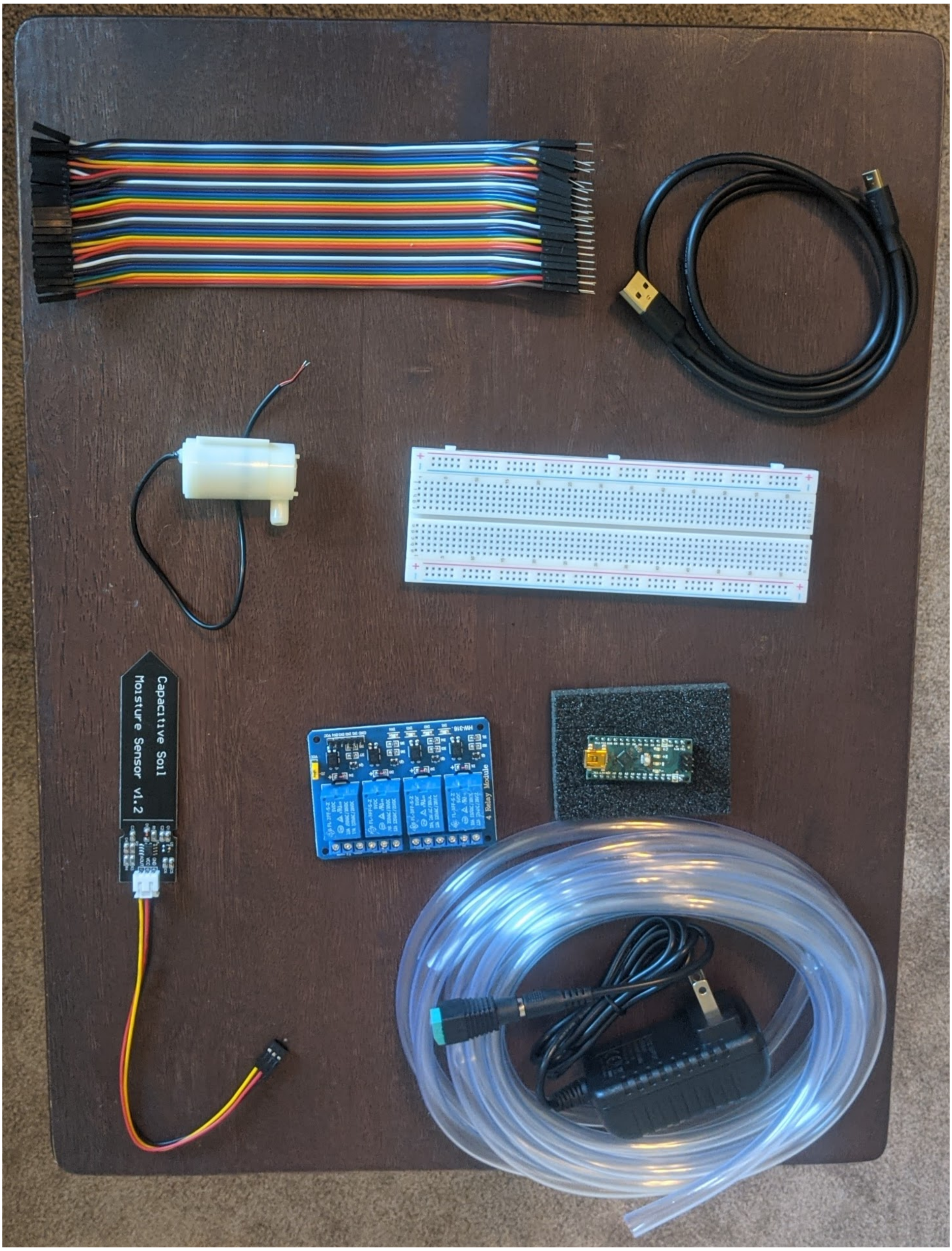
Methods

I chose the WayinTop DIY irrigation kit as my hardware package since it came with the pumps, piping, capacitive sensors, and relay board. This kit didn't include the microcontroller or power supply but did come with Arduino code that reads from the capacitive sensors and signals the relay to turn on the pump. The seller also provided me with wiring instructions detailing the connections between each submodule of the project.

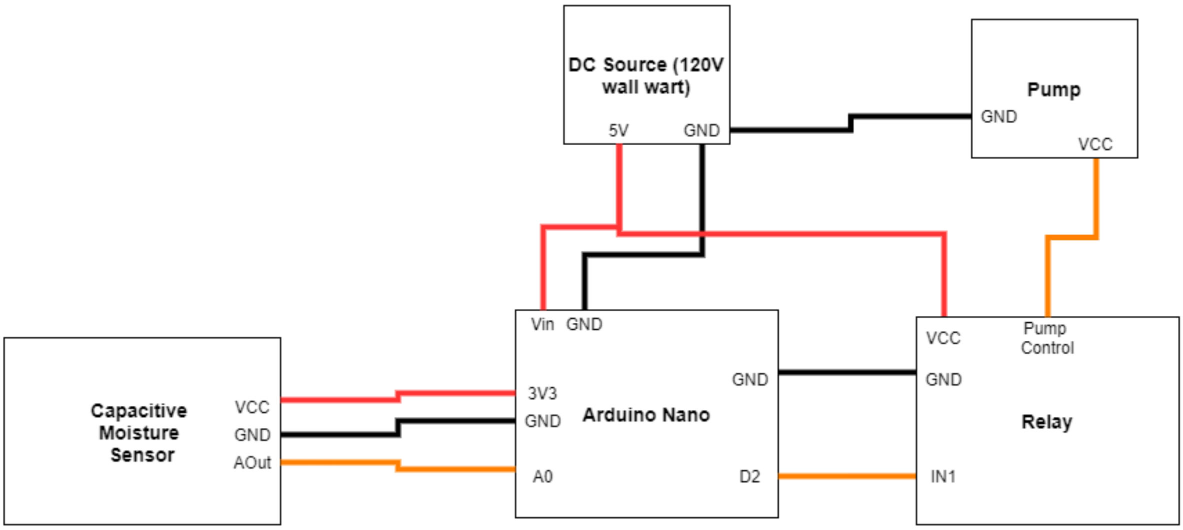
After a bit of digging I decided to use the Arduino Nano because of it's simplicity as well as its ability to fit snuggly onto my breadboard.

My full parts list can be found on this repository on my costs spreadsheet. The entire project cost only $81.61. I did also need to use two plastic Tupperware containers to demo the pump's action as well as a few towels to clean up some spilled water. A knife was used to cut the pump tubing and it also doubled as a screwdriver for securing the relay terminals and 5V output terminals.

Although the DC 5V source was not included in the irrigation kit, I thankfully came across an AC -> DC 5V converter which includes a +/- screw-down terminator allowing me to use my breadboard wires to power the project with ease.



The following wiring diagram shows how the different parts of the project interconnected. Note that this wiring diagram was inspired by the guide provided by the irrigation kit seller but is my own original work.



The original code for the project was also provided by the seller of the irrigation kit. However, I have reworked the code and calibrated it to nurture an *Invicible Plaintain Lily* and have added more features such as lighting control and changed the variable names to be more easily read and debugged.

`

```
int pumpControl = 2;
int CapacitiveMoistureSensor = A0;
float myDryness = 0;

void setup() {
Serial.begin(9600);
pinMode(pumpControl, OUTPUT);
pinMode(CapacitiveMoistureSensor, INPUT);
digitalWrite(pumpControl, HIGH);
delay(500);
}

void loop() {
  // put your main code here, to run repeatedly:

Serial.print("Capacitive sensor reading:");
myDryness = analogRead(CapacitiveMoistureSensor);
Serial.println(myDryness);

  if(myDryness>550) {
    Serial.print("Your plant is dry, activating pump.");
    digitalWrite(pumpControl, LOW);}
  else {
    Serial.print("Your plant is watered, pump will stay inactivated.");
    digitalWrite(pumpControl, HIGH);}

Serial.println();
delay(1000);
}
```

`