A Simple Feedback And Control Loop

Deepanshi.

456807

d.807@mybvc.ca

Bow Valley College

TECH1102: Internet of Things

Instructor: Dima Marachi

February 28, 2024.

Assignment Name:

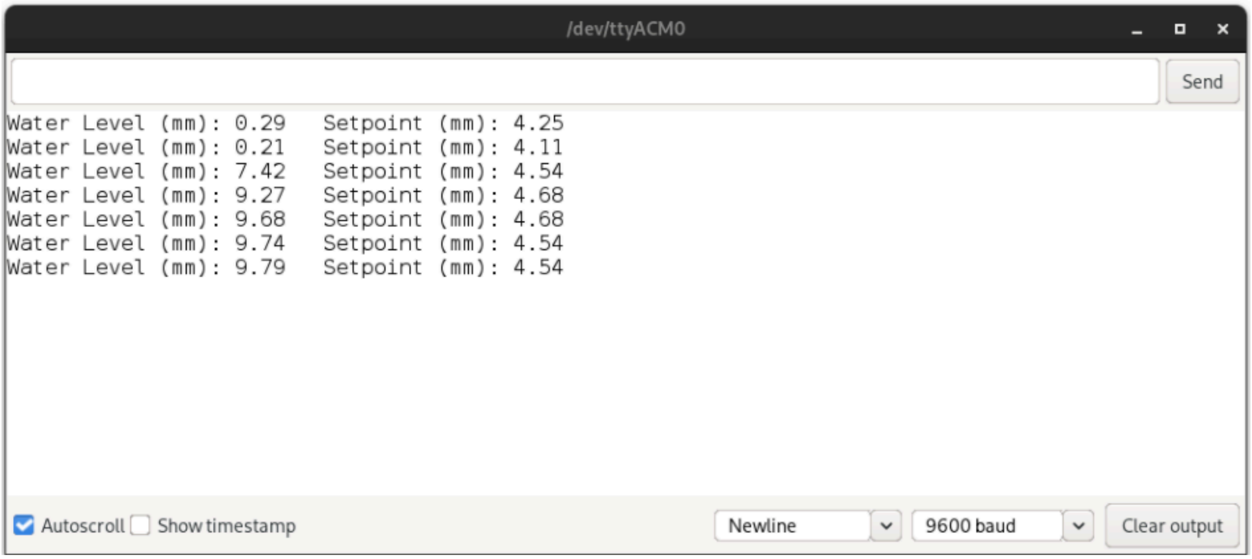
A Simple Feedback And Control Loop

Q1. Please paste in the Arduino code for your very smart water level sensor.

Ans 1:-

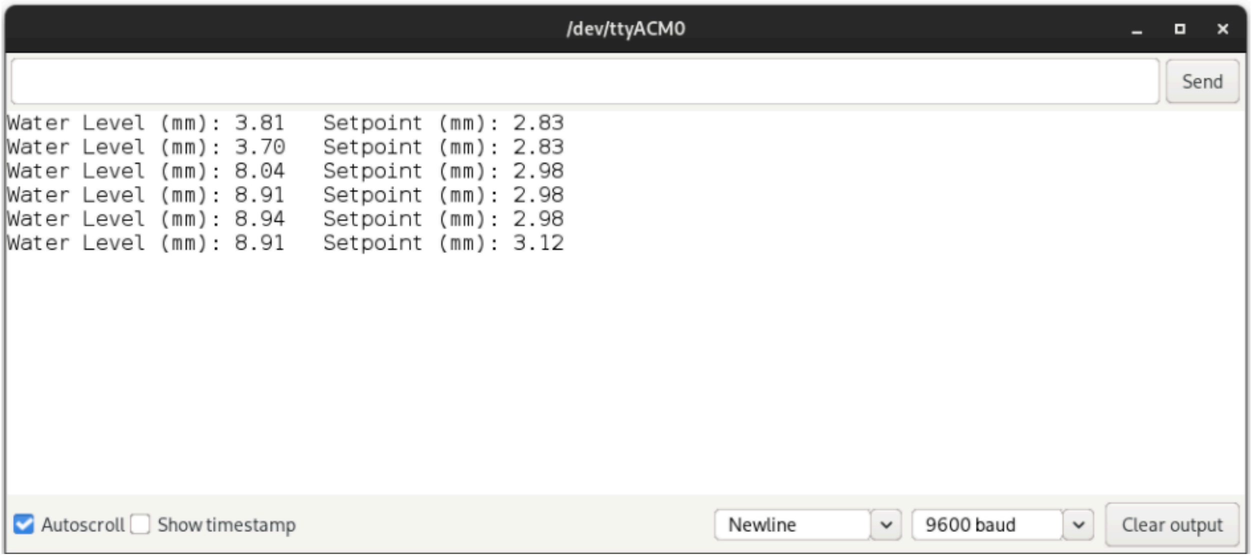
```
// Define pins for water sensor, red LED (pump), blue LED (fan), and potentiometer
const int waterSensorPin = A0;
const int pumpPin = 3; // Red LED
const int fanPin = 5;  // Blue LED
const int potentiometerPin = A1;


// Define constants for maximum sensor reading and maximum water level
const int maxSensorReading = 1023;
const int maxWaterLevel = 30; // Maximum water level in millimeters
void setup() {
  Serial.begin(9600);
 // Set pin modes
  pinMode(waterSensorPin, INPUT);
  pinMode(pumpPin, OUTPUT);
  pinMode(fanPin, OUTPUT);
```

```
}
void loop() {
  // Read sensor value
  int sensorReading = analogRead(waterSensorPin);
// Convert sensor reading to millimeters
  float waterLevel = (float(sensorReading) / maxSensorReading) * maxWaterLevel;
  // Read potentiometer value to adjust setpoint
  int potentiometerReading = analogRead(potentiometerPin);
  float setpoint = (float(potentiometerReading) / maxSensorReading) * 145;
// Adjusted to sensor's range
 // Print the current water level and setpoint
  Serial.print("Water Level (mm): ");
  Serial.print(waterLevel);
  Serial.print("\t Setpoint (mm): ");
  Serial.println(setpoint);
// Control pump and fan based on water level compared to setpoint
  if (waterLevel > setpoint + 5) {
    digitalWrite(fanPin, HIGH);   // Turn on fan
    digitalWrite(pumpPin, LOW);    // Turn off pump
  } else if (waterLevel < setpoint + 5) {
    digitalWrite(fanPin, LOW);      // Turn off fan

    digitalWrite(pumpPin, HIGH);   // Turn on pump

  } else {  // Water level is within 5, turn off both pump and fan

    digitalWrite(fanPin, LOW);      // Turn off fan

    digitalWrite(pumpPin, LOW);    // Turn off pump

  }

delay(5000);  // Wait for 5 seconds before the next reading

}
```



```
Water Level (mm): 0.29   Setpoint (mm): 4.25
Water Level (mm): 0.21   Setpoint (mm): 4.11
Water Level (mm): 7.42   Setpoint (mm): 4.54
Water Level (mm): 9.27   Setpoint (mm): 4.68
Water Level (mm): 9.68   Setpoint (mm): 4.68
Water Level (mm): 9.74   Setpoint (mm): 4.54
Water Level (mm): 9.79   Setpoint (mm): 4.54
```

Initially, I received the signal from the sensor that my tank was empty please turn on the pumpPin (red LED) up to a water level of 9.27 mm and when the water level reached to point 9.74 mm the the fanPin would turn on as the water level is greater than (setpoint + 5 and indicate to us that our tank is about to overflow please turn off your motor.



```
Water Level (mm): 3.81   Setpoint (mm): 2.83
Water Level (mm): 3.70   Setpoint (mm): 2.83
Water Level (mm): 8.04   Setpoint (mm): 2.98
Water Level (mm): 8.91   Setpoint (mm): 2.98
Water Level (mm): 8.94   Setpoint (mm): 2.98
Water Level (mm): 8.91   Setpoint (mm): 3.12
```

The same thing will happen with this reading. Initially, I received the signal from the sensor that my tank was empty please turn on the pumpPin (red LED) up to a water level of 3.81 mm and when the water level reached to point 8.04 mm then the fanPin would turn on as the water level is greater than (setpoint + 5) and indicate to us that our tank is about to overflow please turn off your motor.

Q2. If the water level sensor was installed in your fish tank, but unfortunately was installed very close to a source of water turbulence (bubbles or a tank filter), what problem would you encounter and how could you mitigate this problem using the software?

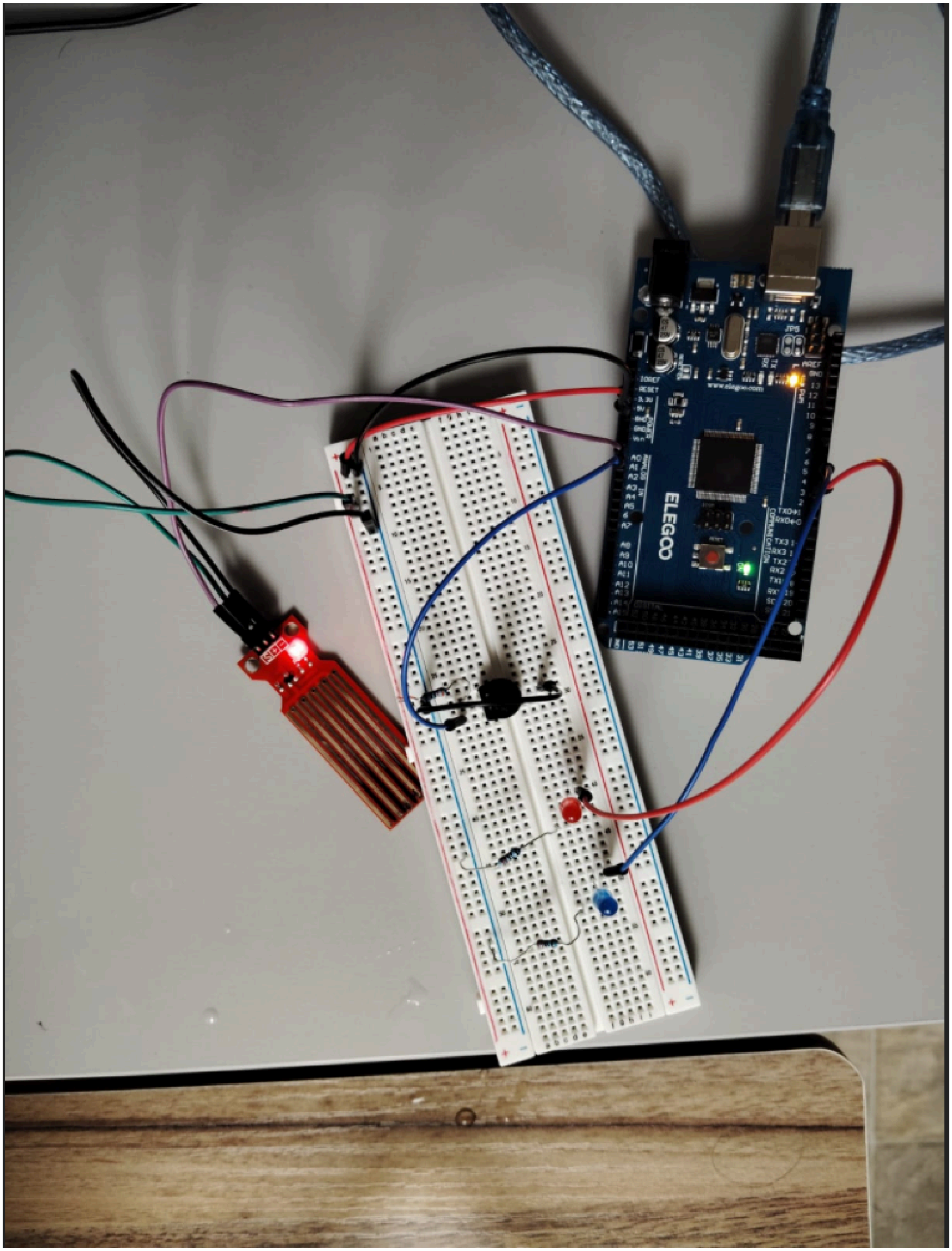
Ans 2:- If the water level sensor is installed very close to a source of water turbulence, such as bubbles or a tank filter, it may encounter issues with inaccurate readings or fluctuations in the readings due to the disturbances in the water.

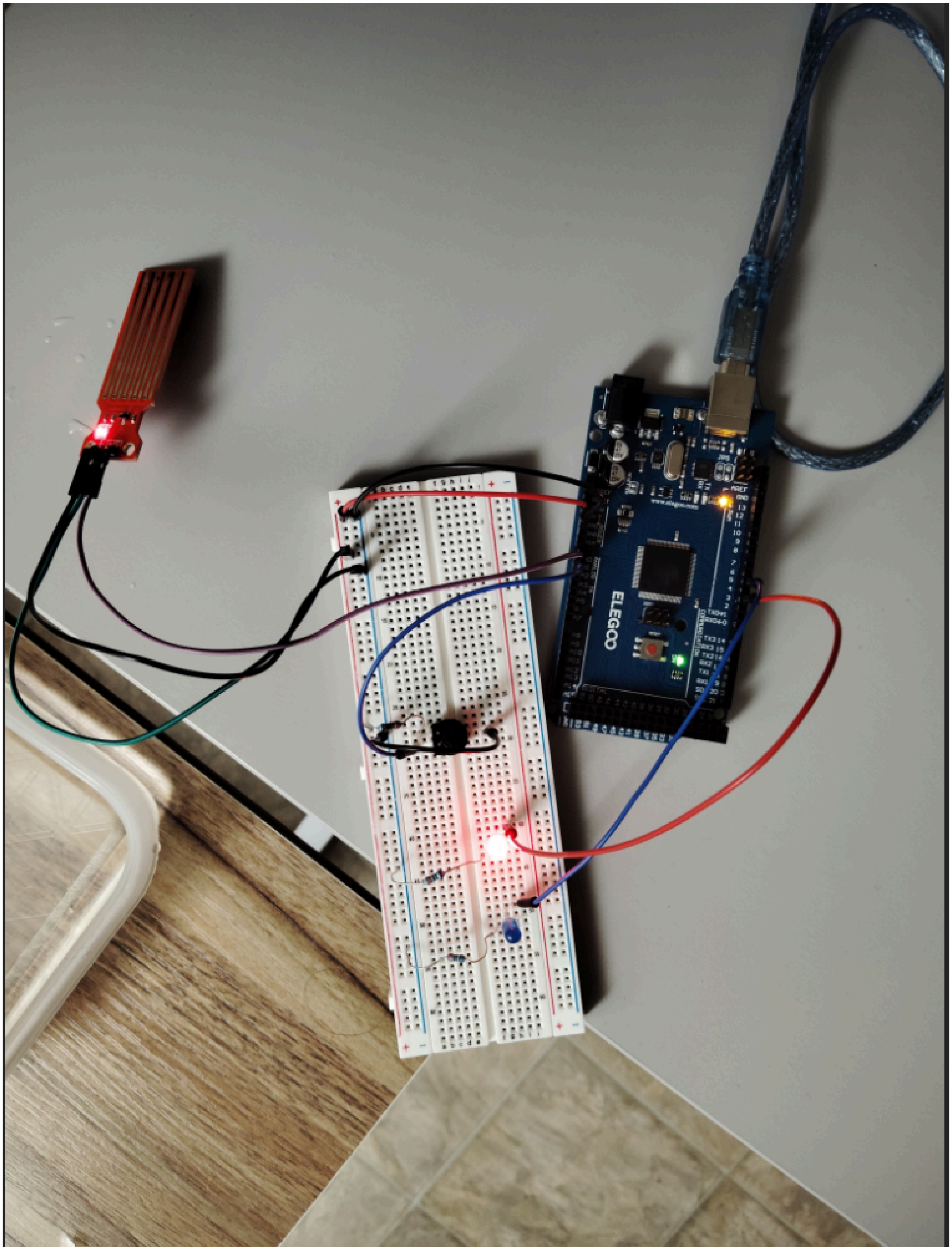
The problem with turbulence near the sensor arises because the sensor may detect bubbles or disturbances as changes in the water level, leading to false readings. This could result in unreliable data and improper control of the pump and the fan based on the water level.

We can use various techniques to migrate this issue:-

- We can install a physical barrier around the water level sensor to shield it from direct exposure to turbulence. This can help minimize the impact of turbulent water flow on the sensor reading.
- We can introduce a threshold around the setpoint. This prevents rapid switching of the pump and fan in response to minor disturbances.

Q3. Take a picture of your circuit and paste it here.




ELEGOO

Q4. Take a short video of your working circuit while showing all the results and upload it on D2L when submitting your work.

Ans 4:- The video will be uploaded on D2L under the folder Assignment Simple_feedbackandControl

REFERENCES

1. The Robotics Back-end. (n.d.). Arduino Potentiometer - Complete Tutorial. https://roboticsbackend.com/arduino-potentiometer-complete-tutorial/
2. Ardumotive.(n.d.). How to Use Potentiometer. Design by DivTag Templates. from, https://www.ardumotive.com/how-to-use-a-potentiometer-en.html
3. AutoDesk Instructables. (n.d.). Read a Potentiometer with Arduino's Analog Input. By circuits in Circuits. from, https://www.instructables.com/Arduino-Potentiometer-Analog-Input-Tinkercad/
4. Forum Arduino. (n.d). Water Sensor. Post on September 2009. By system. from, https://forum.arduino.cc/t/water-sensor/4001