

**Department of Computer Science and Engineering**

**Course Title:** Internet Of Things

**Code:** CSE406

**Section:** 1

**LAB-02**

**Submitted To:**

Dr. Raihan Ul Islam

Associate Professor

Department of Computer Science & Engineering

**Submitted by**

**Name:** Shairin Akter Hashi

**ID:** 2022-2-60-102

**TASK:** How Water Level Sensor Works and Interface it with Arduino

**Introduction**

In IoT applications, real-time monitoring of environmental parameters is critical. This lab explores water level detection using an analogue sensor interfaced with an Arduino UNO, providing both serial feedback and LED-based status indicators. The objective is to design and implement a system that accurately measures water level and categorizes it as Empty, Low, Medium, or High.

**Tools and Devices Used**

* Arduino UNO.
* Analog Water Level Sensor.
* LEDs (Red, Yellow, Green).
* Resistors.
* Breadboard and Jumper Wires.
* Arduino IDE (v1.8+).
* USB Cable for Arduino.
* Serial Monitor (via Arduino IDE).

**Process**

**1. Sensor Setup:**

The water level sensor was powered through pin 7 (controlled digitally), and its analogue output was read from pin A0. A short delay was used to stabilize readings.

**2. Threshold Definition:**

Two thresholds were set:

* Lower threshold: 200
* Upper threshold: 400

**3. LED Logic Control:**

Based on the sensor value:

* Red LED indicates high water level
* Yellow LED indicates medium level
* Green LED indicates low level
* All LEDs are off when the sensor reads empty

**4. Data Feedback:**  Sensor readings and water level status were printed to the serial monitor every second for live tracking.

**Arduino UNO Code:**

*// Sensor pins*

#define sensorPower 7

#define sensorPin A0

*// Value for storing water level*

int val = 0;

int lowerThreshold = 200;

int upperThreshold = 400;

*// Declaring pins to which LEDs are connected*

int redLED = 2;

int yellowLED = 3;

int greenLED = 4;

void setup() {

Serial.begin(9600);

pinMode(sensorPower, OUTPUT);

digitalWrite(sensorPower, LOW);

*// Set LED pins as an OUTPUT*

pinMode(redLED, OUTPUT);

pinMode(yellowLED, OUTPUT);

pinMode(greenLED, OUTPUT);

*// Initially turn off all LEDs*

digitalWrite(redLED, LOW);

digitalWrite(yellowLED, LOW);

digitalWrite(greenLED, LOW);

}

void loop() {

int level = readSensor();

**if** (level == 0) {

Serial.println("Water Level: Empty");

digitalWrite(redLED, LOW);

digitalWrite(yellowLED, LOW);

digitalWrite(greenLED, LOW);

} **else** **if** (level > 0 && level <= lowerThreshold) {

Serial.println("Water Level: Low");

digitalWrite(redLED, HIGH);

digitalWrite(yellowLED, LOW);

digitalWrite(greenLED, LOW);

} **else** **if** (level > lowerThreshold && level <= upperThreshold) {

Serial.println("Water Level: Medium");

digitalWrite(redLED, LOW);

digitalWrite(yellowLED, HIGH);

digitalWrite(greenLED, LOW);

} **else** **if** (level > upperThreshold) {

Serial.println("Water Level: High");

digitalWrite(redLED, LOW);

digitalWrite(yellowLED, LOW);

digitalWrite(greenLED, HIGH);

}

delay(1000);

}

*//This is a function used to get the reading*

int readSensor() {

digitalWrite(sensorPower, HIGH);

delay(10);

val = analogRead(sensorPin);

digitalWrite(sensorPower, LOW);

**return** val;

}

**Output Image:**

A circuit board with wires

AI-generated content may be incorrect.

Image:1

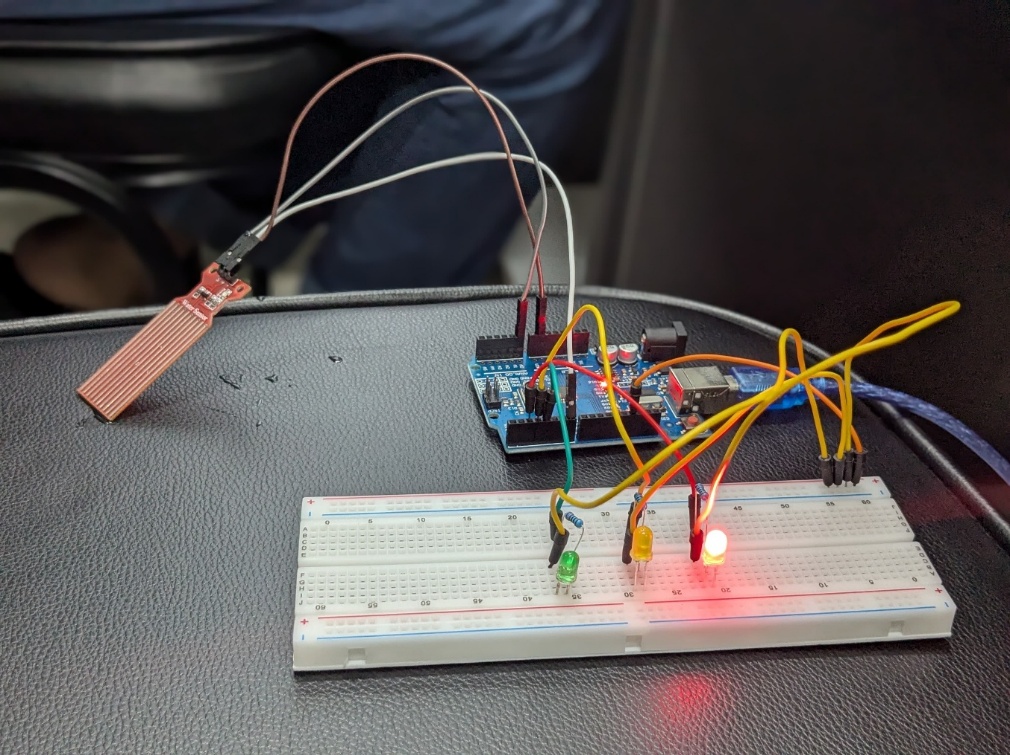


Image:2

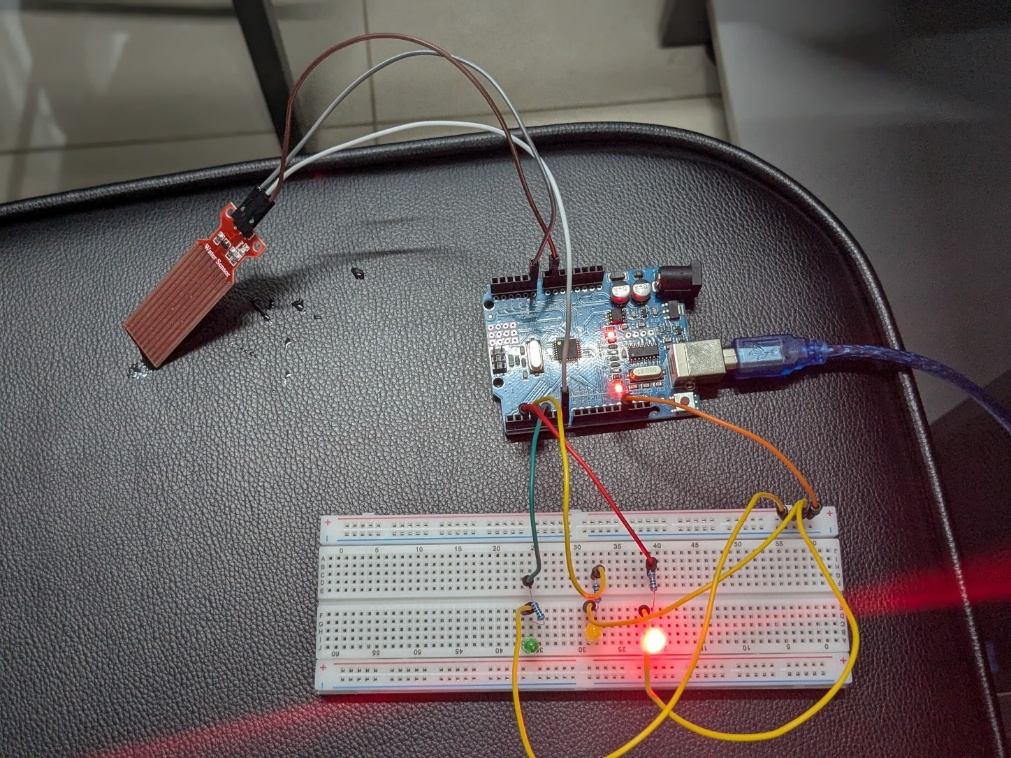


Image:3

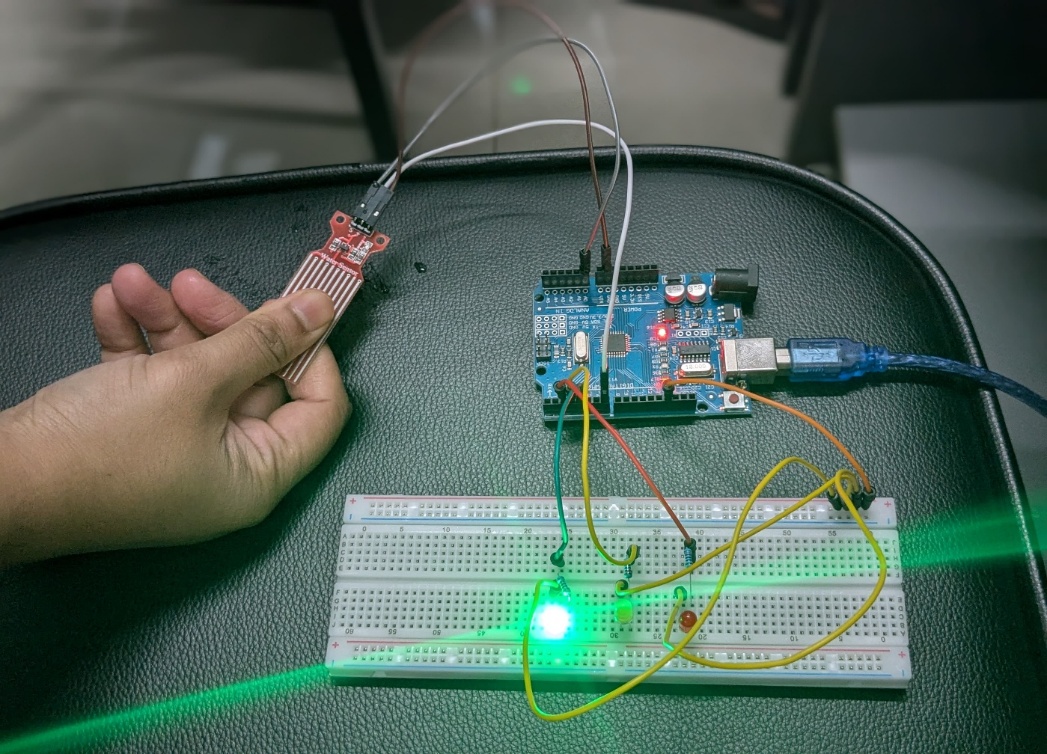


Image:4

**Result and Discussion:**

The circuit successfully categorized water levels using the predefined thresholds. LED indicators responded accurately:

* No sensor detection resulted in all LEDs off and an “Empty” status.
* Readings above 400 activated the red LED, confirming high water presence.
* Values between 200 and 400 lit the yellow LED, indicating a medium level.
* Values below 200 triggered only the green LED, signalling low water.

**Conclusion:**

In this experiment, we successfully developed and tested an Arduino-based water level monitoring system using an analogue sensor and a simple LED-based visual feedback mechanism. The system effectively measured water levels and triggered corresponding LED indicators based on predefined thresholds, providing real-time status updates via serial communication.

**GitHub Link:** <https://github.com/Shairin207/IoT-Lab-Assignments>