

# Micro Project Proposal

## Water Level Monitoring using Arduino Uno

Name of Student	Roll No.
Milan Manoj	40

**Semester & Branch:** 3rd Sem, ER

**Department:** Electronics and Communication

**College Name:** Saintgits College of Engineering

---

### 1 Introduction

The **Water Level Monitoring using Arduino Uno** system is a microcontroller-based project that measures and displays the water level in a tank using an **HC-SR04 ultrasonic sensor**. The sensor detects the distance to the water surface, and the Arduino Uno processes this data to determine the fill level. The measured level is displayed in real time on an **LCD with I2C interface**. This system provides a simple and low-cost solution for monitoring water levels accurately without direct contact with water.

### 2 Objectives

- To design a system that can accurately measure the water level using an ultrasonic sensor.
- To display the real-time water level on an LCD screen.
- To create a compact, contactless, and cost-effective measurement system.
- To demonstrate basic microcontroller interfacing and data visualization.

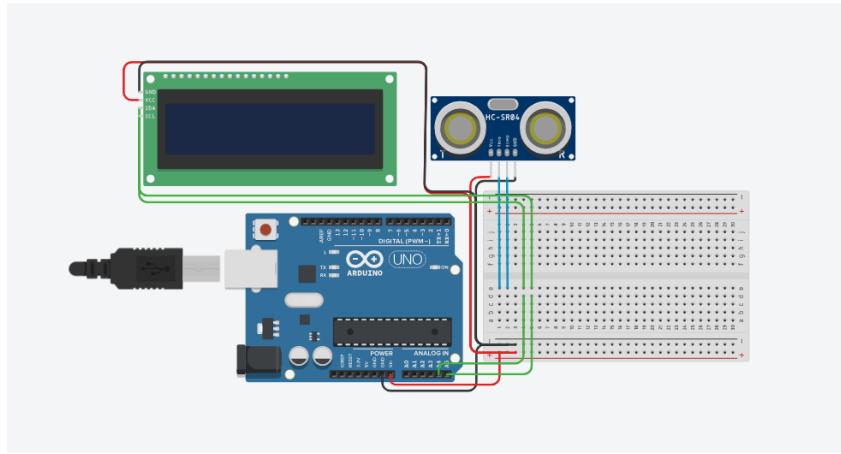
### 3 Block Diagram and Explanation



#### Explanation:

- **HC-SR04 Ultrasonic Sensor:** Measures distance from the sensor to the water surface.
- **Arduino Uno:** Processes sensor data and converts distance to water level.
- **LCD with I2C Module:** Displays the water level percentage or distance in real time.

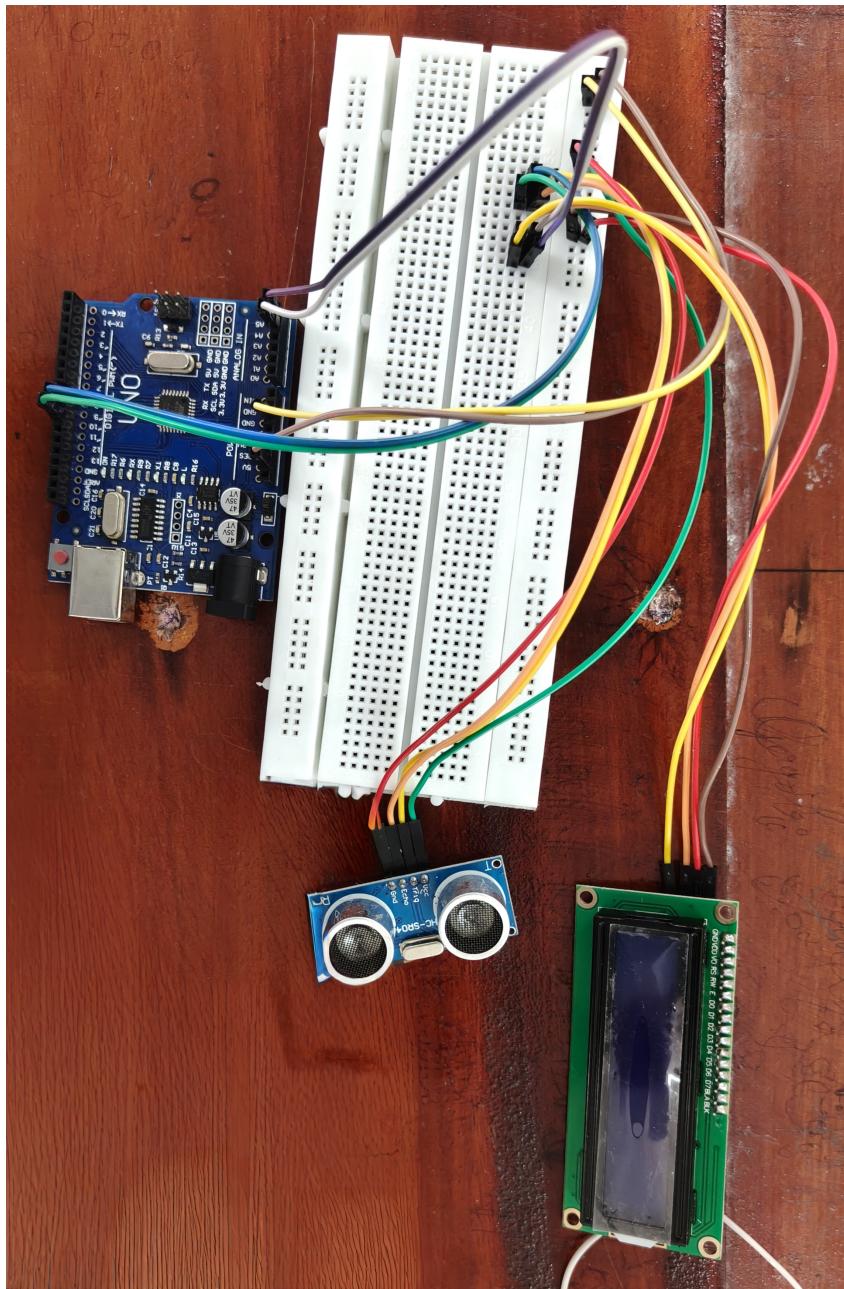
## 4 Circuit Diagram and Explanation



### Explanation of Components:

- **Arduino Uno:** Main controller that reads sensor data and controls the display.
- **HC-SR04 Ultrasonic Sensor:** Measures the distance to the water surface using ultrasonic waves.
- **LCD (16x2) with I2C:** Displays the measured water level or distance.
- **Connecting Wires:** Used for all interconnections between components.

## 5 Working Model



**Figure:** Working model of the Arduino-based Water Level Monitoring System showing Arduino Uno, HC-SR04 sensor, and LCD display.

## 6 Program Code

```
1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3
4 LiquidCrystal_I2C lcd(0x27, 16, 2);
5
6 const int trigPin = 9;
7 const int echoPin = 10;
```

```

8  bool running = true;
9  float duration, distance;
10
11 void setup() {
12   pinMode(trigPin, OUTPUT);
13   pinMode(echoPin, INPUT);
14   lcd.init();
15   lcd.backlight();
16   lcd.setCursor(1, 0);
17   lcd.print("Hello!");
18   delay(1500);
19   lcd.setCursor(1, 1);
20   lcd.print("Goodbye");
21   delay(1500);
22   lcd.clear();
23   Serial.begin(9600);
24 }
25
26 void loop() {
27   while (running == true) {
28     digitalWrite(trigPin, LOW);
29     delayMicroseconds(2);
30     digitalWrite(trigPin, HIGH);
31     delayMicroseconds(10);
32     digitalWrite(trigPin, LOW);

33     duration = pulseIn(echoPin, HIGH);
34     distance = (duration * 0.0343) / 2;

35     delay(100);
36     if (distance > 40) {
37       lcd.print("More!");
38       delay(300);
39       lcd.clear();
40       Serial.print("Distance: ");
41       Serial.println(distance);
42     }
43     if (distance < 40) {
44       lcd.print("Stop!");
45       delay(300);
46       Serial.print("Distance: ");
47       Serial.println(distance);
48       running = false;
49     }
50   }
51 }
52 }
53 }
```

## 7 Description of Code and Interfacing Logic

The project uses an **Arduino Uno** as the central controller, an **HC-SR04 ultrasonic sensor** to measure water distance, and a **16x2 LCD with I2C interface** to display the water level.

## Interfacing Logic

- **HC-SR04 Sensor:**

- Trig pin is connected to Arduino digital pin 9.
- Echo pin is connected to Arduino digital pin 10.
- The Arduino sends a short pulse on the Trig pin to start the measurement.
- The sensor returns a pulse on the Echo pin, whose duration represents the distance to the water surface.

- **LCD I2C Display:**

- Connected to Arduino SDA (A4) and SCL (A5) pins via I2C.
- Displays messages based on the distance measured: "More!" if water level is low, and "Stop!" if water level is high.

## Code Description

- The `setup()` function initializes the LCD, the sensor pins, and the serial monitor. A greeting message ("Hello!" and "Goodbye") is displayed briefly.
- In the `loop()`, the Arduino continuously measures the distance using the HC-SR04 sensor:
  - A pulse is sent to the Trig pin.
  - The duration of the received pulse on the Echo pin is measured.
  - The distance to the water surface is calculated as `distance = (duration * 0.0343) / 2.`
- Based on the measured distance:
  - If `distance > 40 cm`, the LCD displays "More!" indicating the water level is low.
  - If `distance < 40 cm`, the LCD displays "Stop!" indicating the tank is full or water is high.
- The serial monitor simultaneously prints the measured distance for debugging or monitoring purposes.

This logic ensures real-time monitoring of water level without any physical contact, and provides a simple visual feedback using the LCD.

## 8 Expected Outputs/Outcomes

- Real-time display on the LCD showing the distance measured by the HC-SR04 sensor.
- Displays "**More!**" on the LCD when the water level is low (distance  $> 40$  cm).
- Displays "**Stop!**" on the LCD when the water level is high (distance  $< 40$  cm).

- Provides a simple visual 0 of water level without any contact with the liquid.
- Demonstrates basic microcontroller interfacing and programming with Arduino and I2C LCD.