WATER TURBIDITY SENSOR

TEAM MEMBERS

Shrijay Naik	23BCE8159
Aniruddh Dwivedi	23BCE8304
Yash Chaudhary	23BCE8384
Supratim Ghosh	23BCE8428
Janga Thulasi	23BEC7104

MATERIALS REQUIRED

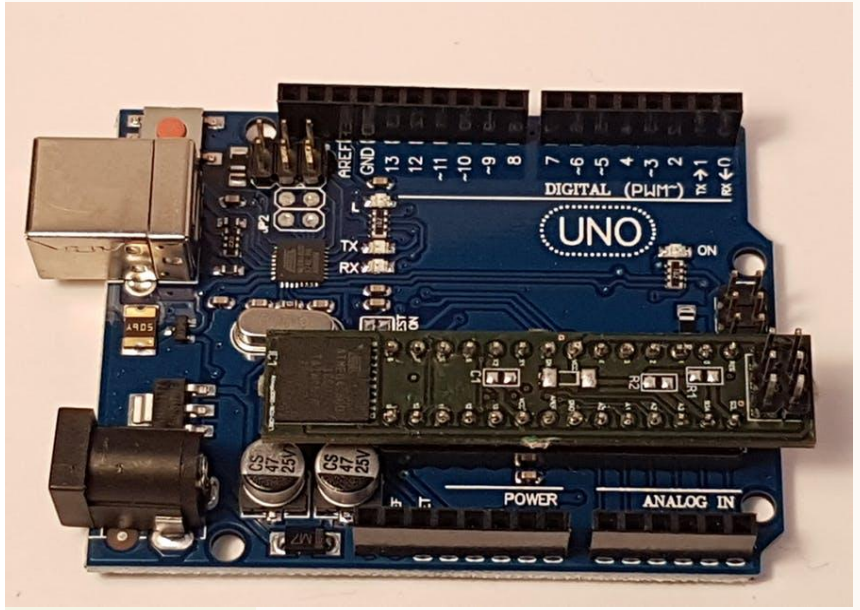
Arduino Uno
Water Turbidity Probe
16 x 2 LCD connection (with i2c)
Jumper Wires
Wire Strippers
5V battery clip

ARDUINO UNO

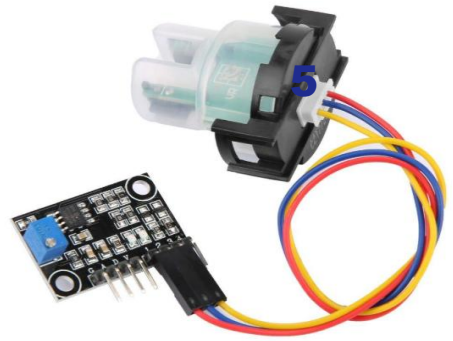
Arduino Uno is a microcontroller board based on the **ATmega328P** chip, widely used for electronics projects and prototyping. It features:

- **Digital & Analog Pins**: 14 digital I/O pins (6 PWM), 6 analog input pins.
- **Power Supply**: Operates at **5V**, with input voltage range **7-12V**.
- **Connectivity**: USB connection for programming, power jack, and ICSP header.
- **Memory**: 32 KB Flash, 2 KB SRAM, 1 KB EEPROM.
- **Programming**: Uses Arduino IDE and C/C++ language.

It's ideal for beginners and advanced users in **robotics, IoT, automation, and sensor-based projects**.



WATER TURBIDITY PROBE



A **water turbidity probe** measures the clarity of water by detecting the presence of suspended particles. It works by emitting a light beam (often infrared or laser) into the water and analyzing how much light is scattered or absorbed. The more particles present, the higher the turbidity, meaning the water is less clear.

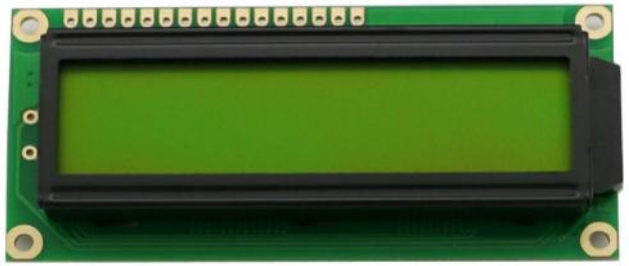
Key Features:

- **Measurement Unit**: NTU (Nephelometric Turbidity Units)
- **Types**: Optical sensors (nephelometric, absorption-based)
- **Applications**:
    - Drinking water quality monitoring
    - Wastewater treatment
    - Environmental research
    - Industrial process control
- **Connectivity**: Can be integrated with **Arduino, Raspberry Pi**, or other controllers for automation.

ANALOG WATER INFRARED SENSING OUTPUT MODULE

- An **analog water infrared sensing output module** is used to measure the turbidity (clarity) of water by detecting suspended particles using **infrared (IR) light**.
- **How It Works:**
- The sensor emits **infrared light** into the water.
- Suspended particles scatter the light.
- A **photodetector** measures the scattered light intensity.
- The sensor outputs an **analog signal** proportional to turbidity.
- **Key Features:**
- **Analog Output**: Provides real-time voltage signals for turbidity levels.
- **Infrared-Based Measurement**: Works in both clear and slightly murky water.
- **Microcontroller Compatibility**: Works with **Arduino, Raspberry Pi, and ESP32**.
- **Low Power Consumption**: Ideal for continuous monitoring.

16X2 LCD DISPLAY



A **16×2 LCD (Liquid Crystal Display)** is a common alphanumeric display that shows **16 characters per line on 2 rows.** It is widely used in embedded systems, IoT projects, and microcontroller-based applications.

Key Features:

1. **16 Characters × 2 Rows** – Can display letters, numbers, and symbols.
2. **HD44780 Controller** – Standard driver for easy interfacing.
3. **Backlight & Contrast Control** – Improves visibility in different lighting conditions.
4. **Parallel Interface (4-bit/8-bit Mode)** – Connects easily to **Arduino, Raspberry Pi, ESP32, etc.**
5. **Low Power Consumption** – Ideal for battery-operated project.

Applications:

IoT Devices
Robotics
Home Automation
Weather Stations
Industrial Control Systems

PROCEDURE

Step 1: Prepare the Components

- Ensure all components are available and in working condition.
- Use wire strippers to prepare jumper wires as needed for connections.

Step 2: Connect the Turbidity Sensor to the Arduino Uno

1. **VCC (Power):**
    - Connect the VCC pin of the turbidity sensor to the 5V pin on the Arduino Uno.
2. **GND (Ground):**
    - Connect the GND pin of the turbidity sensor to the GND pin on the Arduino Uno.
3. **Signal Output:**
    - Connect the analog output pin (often labeled as A0) of the turbidity sensor to the A0 analog input pin on the Arduino Uno.

Step 3: Connect the 16×2 I2C LCD Display to the Arduino Uno

1. **VCC (Power):**
    - Connect the VCC pin of the I2C LCD display to the 5V pin on the Arduino Uno.
2. **GND (Ground):**
    - Connect the GND pin of the I2C LCD display to the GND pin on the Arduino Uno.
3. **SDA (Data Line):**
    - Connect the SDA pin of the I2C LCD display to the A4 pin on the Arduino Uno.
4. **SCL (Clock Line):**
    - Connect the SCL pin of the I2C LCD display to the A5 pin on the Arduino Uno.

PROCEDURE

Step 4: Power the System

- Attach the 9V battery to the battery clip.
- Connect the battery clip to the Arduino Uno's power jack to provide power to the system.

Step 5: Upload the Arduino Code

1. **Install Necessary Libraries:**
    - In the Arduino IDE, install the LiquidCrystal_I2C library to facilitate communication with the I2C LCD display.
2. **Write the Code:**
    - Initialize the turbidity sensor input and the I2C LCD display.
    - In the loop() function, read the analog value from the turbidity sensor.
    - Convert the analog reading to a turbidity value (e.g., in NTU - Nephelometric Turbidity Units).
    - Display the turbidity value on the 16×2 LCD screen.
3. **Upload the Code:**
    - Connect the Arduino Uno to your computer via USB.
    - Upload the code to the Arduino using the Arduino IDE.

Step 6: Calibrate the Sensor

- Use water samples with known turbidity levels to calibrate the sensor readings.
- Adjust the code as necessary to ensure accurate turbidity measurements.

Step 7: Test the System

- Submerge the turbidity sensor in various water samples.
- Observe the turbidity readings displayed on the LCD screen to assess water quality.

FUNCTIONING OF THE DEVICE

A water turbidity measurement device utilizing an Arduino Uno operates by assessing the cloudiness or haziness of a water sample, which indicates the presence of suspended particles. The system's functionality involves several key components working together:

1. **Turbidity Sensor Operation:**
    - The turbidity sensor employs an optical method to evaluate water clarity. It consists of an infrared light-emitting diode (LED) and a photodetector positioned to measure the light that passes through the water sample.
    - In clearer water, more light reaches the photodetector, resulting in a higher output voltage. Conversely, in turbid water with more suspended particles, less light reaches the detector, leading to a lower output voltage.
2. **Arduino Uno Processing:**
    - The Arduino Uno reads the analog voltage output from the turbidity sensor through one of its analog input pins.
    - It then converts this analog signal into a digital value, which can be processed to determine the turbidity level.
3. **Data Display Using 16×2 I2C LCD:**
    - The processed turbidity value is displayed on a 16×2 LCD screen connected to the Arduino via the I2C interface. This setup provides a clear and immediate visual representation of the water's turbidity level.

CODE USED

turbidity_sensor | Arduino 1.8.19 (Windows Store 1.8.57.0)

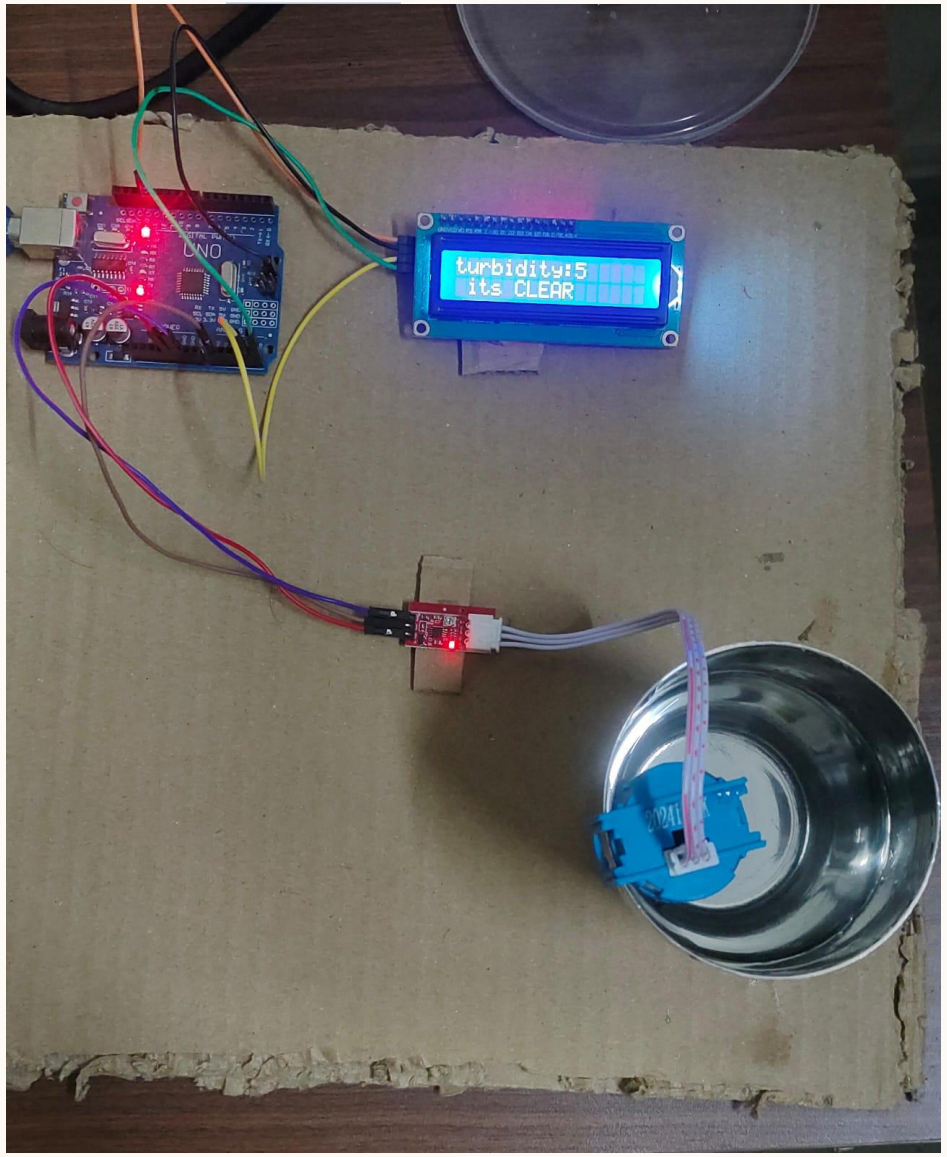
File Edit Sketch Tools Help

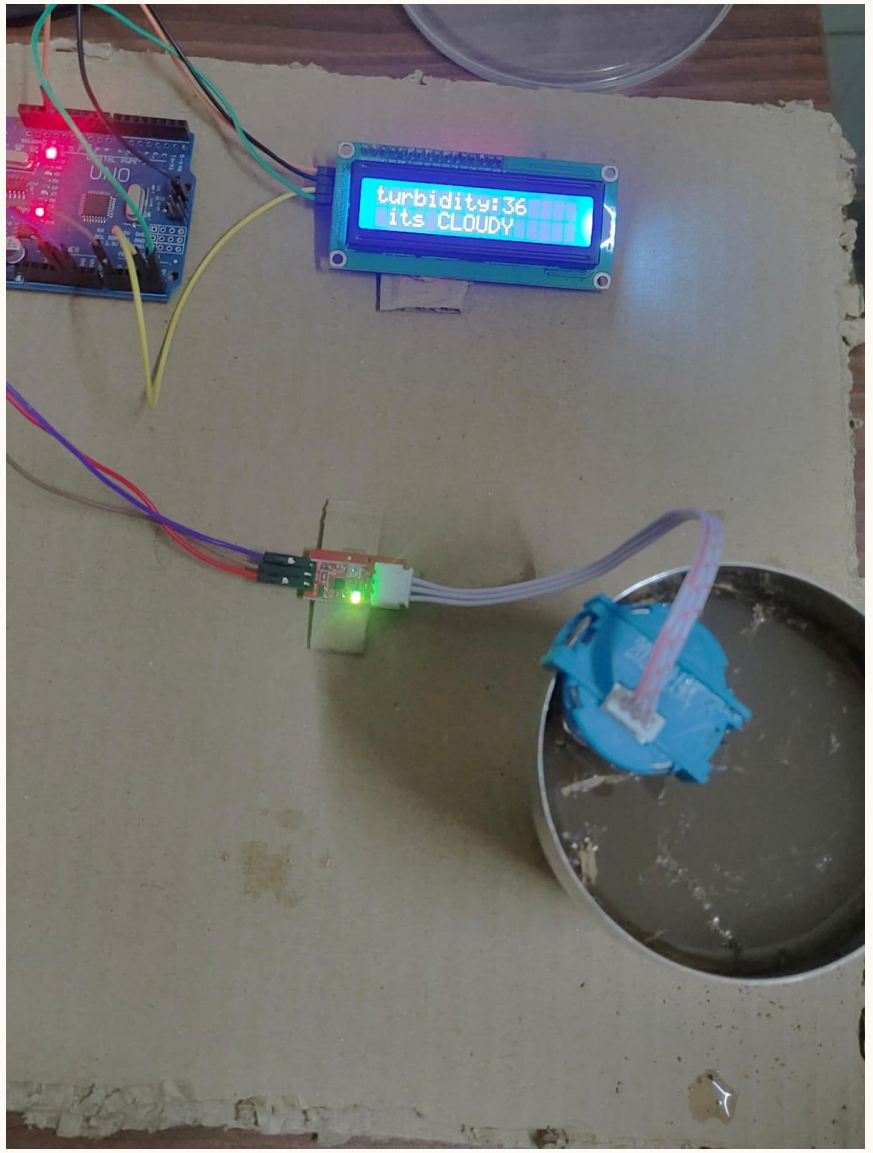
turbidity_sensor

```
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE );

int sensorPin = A0;
void setup()
{
  lcd.begin(16,2);
  pinMode(3,OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);

}
```

```
void loop() {
  int sensorValue = analogRead(sensorPin);
  int turbidity = map(sensorValue, 0,640, 100, 0);
  delay(100);
  lcd.setCursor(0, 0);
  lcd.print("turbidity:");
  lcd.print("   ");
  lcd.setCursor(10, 0);
  lcd.print(turbidity);
  delay(100);
  if (turbidity < 20) {
    digitalWrite(7, HIGH);
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);
    lcd.setCursor(0, 1);
    lcd.print(" its CLEAR ");
  }
  if ((turbidity > 10) && (turbidity < 50)) {
    digitalWrite(7, LOW);
    digitalWrite(8, HIGH);
    digitalWrite(9, LOW);
    lcd.setCursor(0, 1);
    lcd.print(" its CLOUDY ");
  }
  if (turbidity > 50) {
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
    digitalWrite(9, HIGH);
    lcd.setCursor(0, 1);
    lcd.print(" its DIRTY ");
  }

}
```

Product Testing Images





THANK YOU

Thank you for your attention