WATER QUALITY MONITORING SYSTEM USING TDS AND TURBIDITY SENSORS WITH ARDUINO UNO

OBJECTIVE:

The objective of this project is to design and implement a real-time water quality monitoring system that utilizes TDS (Total Dissolved Solids) and turbidity sensors integrated with an Arduino UNO microcontroller. This system aims to measure and evaluate critical parameters—specifically, the turbidity level and TDS concentration—in various water sources. By providing real-time monitoring, the project addresses the need for continuous and cost-effective water quality assessment, which is crucial for applications in environmental monitoring, industrial processes, and public health. The data obtained from the sensors will offer insights into water clarity and contamination levels, which are essential for determining water suitability for consumption, agricultural use, or industrial application. This project also intends to present an accessible and scalable solution that can be deployed in both rural and urban areas, supporting better water quality management practices.

ABSTRACT:

The rapid industrialization and urbanization in recent years have led to an increasing need for real-time water quality monitoring to ensure safe and clean water for human consumption and environmental sustainability. This project presents a cost-effective water quality monitoring system utilizing Arduino UNO as the central microcontroller, combined with Total Dissolved Solids (TDS) and turbidity sensors. The system is designed to measure two key parameters: turbidity, which indicates the cloudiness or clarity of water and serves as a proxy for potential contaminants, and TDS, which quantifies the concentration of dissolved substances, a critical factor in determining water safety. By continuously capturing data on these parameters, the system provides a clear and timely picture of water quality conditions, essential for detecting contamination in drinking water sources, industrial discharges, and natural water bodies.

In this project, the TDS and turbidity sensors are interfaced with Arduino UNO to gather real-time data, which is processed and displayed on an LCD screen or transmitted for remote monitoring. The methodology involves calibration of sensors to ensure accuracy, construction of an optimized circuit, and coding for data acquisition and display. This system offers a scalable, low-cost solution suitable for deployment in both urban and rural settings, potentially benefiting municipalities, environmental monitoring agencies, and local communities. The findings from this study highlight the efficacy and reliability of using Arduino-based sensor systems for water quality monitoring, demonstrating that such solutions can be instrumental in addressing water safety challenges and promoting public health.

Additionally, this project emphasizes the adaptability of the Arduino-based water quality monitoring system for a wide range of applications by providing options for future expansion. By integrating wireless communication modules, such as Wi-Fi or GSM, the system can transmit real-time data to remote servers or cloud platforms, enabling users to monitor water quality from anywhereFurthermore, the modular design of the system allows for the integration of additional sensors, such as pH and temperature, to offer a more comprehensive analysis of water quality. The project showcases the potential of Arduino technology to create accessible, scalable, and effective water quality monitoring solutions that can be tailored to meet diverse environmental monitoring needs and support efforts toward sustainable water resource management.

INTRODUCTION:

The project addresses the application of real-time water quality monitoring, a critical process for ensuring the safety and usability of water in various sectors, including public health, agriculture, and industry. Traditional water quality assessment methods often involve costly and time-consuming laboratory analyses that do not provide immediate feedback. This project proposes a solution by implementing an Arduino-based system that continuously monitors essential water quality parameters—specifically, Total Dissolved Solids (TDS) and turbidity. These measurements offer valuable insights into water purity and contamination levels, enabling prompt detection of changes in water quality. This real-time monitoring system is particularly useful for applications where rapid response to contamination is necessary, such as in drinking water supplies, industrial processes, wastewater treatment, and environmental monitoring of rivers and lakes. By offering a portable, low-cost, and easy-to-deploy solution, this project aims to make water quality monitoring more accessible and effective, especially in areas with limited resources.

HARDWARE REQUIREMENT/DESCRIPTION:

Arduino UNO, TDS sensor, Turbidity sensor, 16x2 LCD display, breadboard, LED, Jumper wires and resistors.

Arduino UNO: The Arduino UNO is a microcontroller board based on the ATmega328P, widely used for prototyping and DIY electronics. It has 14 digital input/output pins, 6 analog inputs, and a USB interface for easy programming. In this project, it serves as the central controller, reading data from the TDS and turbidity sensors, processing it, and displaying or transmitting the results.

TDS Sensor: The TDS (Total Dissolved Solids) sensor measures the concentration of dissolved solids (like salts, minerals, and metals) in water, which can indicate water quality. It works by measuring the electrical conductivity of the water, with higher conductivity reflecting higher levels of dissolved solids. This sensor provides an analog output, which is read by the Arduino to determine the TDS value of the water.\

Turbidity Sensor: The turbidity sensor measures the cloudiness or haziness of water, which is caused by suspended particles. It works by emitting infrared light through the water and detecting the amount of light scattered by the particles. The greater the turbidity, the higher the scattering, and the sensor generates an output that corresponds to the water's turbidity level. This helps assess the water's cleanliness.

16x2 LCD Display: A 16x2 LCD display shows real-time readings of the water quality parameters (TDS and turbidity). It has two rows of 16 characters, making it easy to display the data. The LCD interface with Arduino is often simplified using an I2C module, allowing for easy wiring and communication. It provides a visual output for users to monitor water quality on-site without needing additional devices.

Breadboard: A breadboard is a tool used for prototyping electronic circuits without soldering. It allows you to quickly build and modify the connections between components like sensors, the Arduino, and the LCD. It has rows and columns of holes to insert wires and components, enabling easy experimentation and testing during the development phase of the project

.

LED (Light Emitting Diode): An LED is a semiconductor device that emits light when current flows through it. It is commonly used for visual indicators in electronic circuits. In this project, LEDs can be used to signal the water quality status, such as turning green for safe water or red for contaminated water.

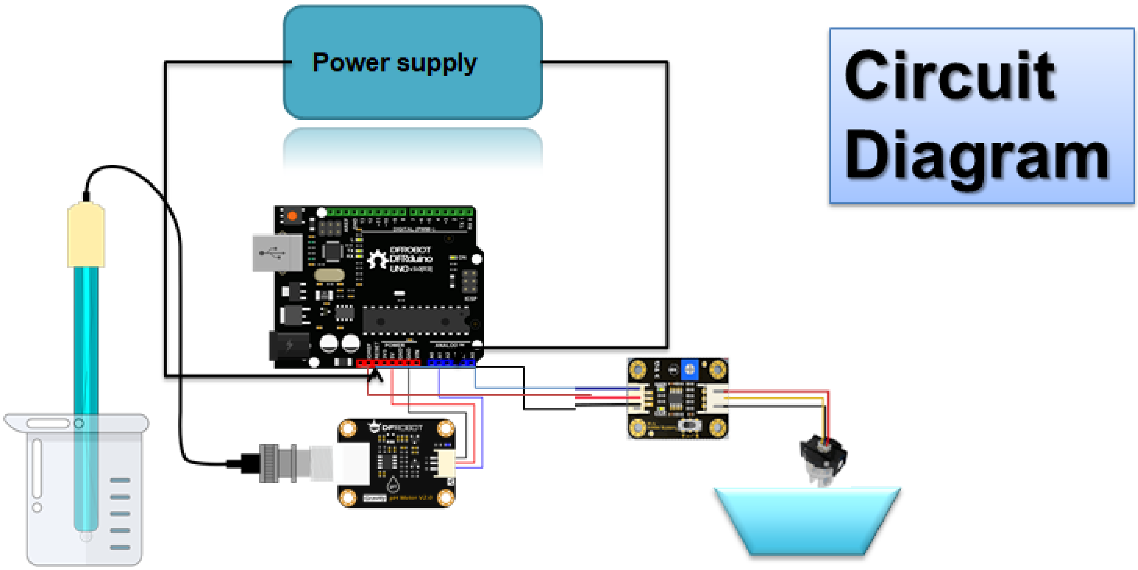
Jumper Wires: Jumper wires are used to make connections between different components on the breadboard and Arduino. They come in different lengths and types (male-to-male, female-to-male, etc.), providing flexibility to connect sensors, display, and other modules.

Resistors: Resistors are passive components used to limit the current flow in circuits. They are used to protect sensitive components from excessive current and help with signal conditioning, such as adjusting the sensitivity of sensors or controlling the brightness of the LCD display.

CIRCUIT/COMPONENT SPECIFICATIONS:

Arduino UNO	ATmega328P microcontroller, 14 digital I/O pins, 6 analog inputs, 16 MHz clock, USB powered or 9V battery.
TDS Sensor	Output: Analog, Voltage: 3.3V to 5V, Range: 0-1000 ppm (typical), Accuracy: ±2%.
Turbidity Sensor	Output: Analog, Voltage: 5V, Range: 0-1000 NTU (Nephelometric Turbidity Units), Accuracy: ±5%.
16x2 LCD Display	16 characters per line, 2-line display, I2C interface for easy connection with Arduino, Voltage: 5V.
Breadboard	Standard full-size breadboard, 830 tie-points, non-soldering connections, fits standard jumper wires.
LED	Standard 5mm LED, forward voltage: 2V, current: 20mA, used for visual indication of water quality status.
Jumper Wires	Male-to-male, female-to-male, and female-to-female, various lengths (10 cm to 20 cm).
Resistors	Typically 220Ω, 10kΩ (depends on sensor and LED requirements), used for current limiting and voltage division.

CIRCUIT DIAGRAM:



TDS Sensor Turbidity Sensor

DESIGN FORMULA:

1. TDS Sensor Calculation

The TDS (Total Dissolved Solids) value is calculated based on the electrical conductivity (EC) measured by the sensor:

$$TDS = K \times EC$$

Where:

- **TDS** = Total Dissolved Solids in ppm (parts per million)
- **K** = Calibration constant (typically between 0.5 and 1.0)
- **EC** = Electrical Conductivity in μS/cm (microsiemens per centimeter)

2. Turbidity Sensor Calculation

The turbidity of the water is calculated based on the sensor's output voltage:

$$\text{Turbidity (NTU)} = (Vout - Vmin \setminus Vmax - Vmin) \times 1000$$

Where:

- **Turbidity (NTU)** = Turbidity value in Nephelometric Turbidity Units (NTU)
- **V_out** = Output voltage from the turbidity sensor
- **V_min** = Minimum voltage corresponding to 0 NTU (clear water)
- **V_max** = Maximum voltage corresponding to the highest turbidity level

DESIGN ISSUES:

- **Sensor Calibration**: Achieving accurate and consistent calibration for both the TDS and turbidity sensors is essential for reliable water quality measurements. Inaccurate calibration can lead to significant errors in the readings.
- **Power Consumption**: Managing power usage efficiently is critical, especially with multiple sensors and components. The system needs to be optimized to run on limited battery power for extended periods without frequent recharging.
- **Sensor Sensitivity**: Both the TDS and turbidity sensors must be sensitive enough to detect small variations in water quality. Balancing sensitivity without overwhelming the system with noise or false readings is a key challenge.
- **Data Processing and Display**: Real-time data processing and the display of accurate water quality information on the LCD can put a strain on the Arduino's processing capabilities. Optimizing the code and ensuring smooth performance is necessary to handle continuous sensor data input and updates.

APPROACH/METHODOLOGY:

1. **System Design and Hardware Setup**:

- Connect the **TDS and turbidity sensors** to the Arduino UNO along with a **16x2 LCD display** for real-time data output.
- Utilize a breadboard and jumper wires to construct the circuit, ensuring stable connections between sensors, display, and microcontroller.
- Calibrate sensors by testing with known water samples to establish baseline readings and adjust for environmental factors.

2. **Sensor Calibration and Data Acquisition**:

- **TDS Sensor Calibration**: Place the TDS sensor in distilled water (0 ppm) and in a standard solution (e.g., 500 ppm) to set calibration points. Adjust the calibration constant (K) as required for accurate results.
- **Turbidity Sensor Calibration**: Test the turbidity sensor in clean (low NTU) and cloudy (high NTU) water samples to set minimum and maximum voltage values. These values will be used to map voltage outputs to NTU levels.
- Code the Arduino to read analog sensor inputs, convert the readings to digital values, and apply calibration formulas to calculate TDS (ppm) and turbidity (NTU).

3. **Data Processing and Display**:

- Implement the conversion formulas within the Arduino code to process raw sensor data and calculate real-time TDS and turbidity values.
- Display these values on the **16x2 LCD screen** with a refresh rate that balances responsiveness with stability.
- Include conditional statements in the code to trigger an alert (e.g., LED light) if TDS or turbidity exceeds safe thresholds.

4. **Testing and Validation**:

- Test the system with different water samples (e.g., tap water, distilled water, and polluted water) to validate the accuracy and consistency of the readings.
- Refine the calibration constants and code as needed to improve measurement precision.
- Monitor the power consumption and overall stability of the system, making adjustments to ensure optimal performance.

5. **Deployment and Analysis**:

- Deploy the system in the intended environment (e.g., water source or tank) for real-time monitoring.
- Record data over a specified period and analyze it for trends, anomalies, or consistency.
- Evaluate the system's effectiveness and reliability, and make any necessary adjustments for future use or scaling.

RESULTS:

Sample No.	Water Source	TDS (ppm)	Turbidity(NTU)	Status
1	Distilled Water	10	0.1	Safe
2	Tap Water	350	0.5	Safe
3	Ground Water	600	1.5	Caution Required

CONCLUSIONS:

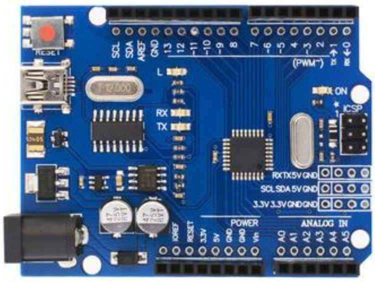
In conclusion, the Arduino-based water quality monitoring system utilizing TDS and turbidity sensors has proven to be a cost-effective and practical solution for real-time water quality assessment. The system effectively identifies water quality variations across different sources, allowing for prompt detection of unsafe conditions, which is crucial for both public health and environmental protection. Its modular design ensures scalability, making it suitable for diverse applications, from small-scale rural use to larger municipal deployments. Furthermore, this project demonstrates the significant potential of accessible technology in addressing water safety challenges, with future enhancements like wireless data transmission and additional sensors promising even greater utility. This low-cost monitoring approach holds substantial promise for communities in need of reliable water quality data, ultimately supporting safer and more sustainable water management practices.

REFERENCES:

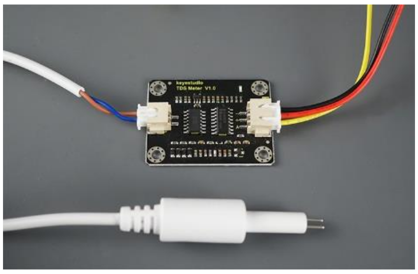
1. https://www.tomsonelectronics.com/blogs/news/arduino-uno-specification
2. https://how2electronics.com/diy-turbidity-meter-using-turbidity-sensor-arduino/
3. https://github.com/duyhuynh/Turbidity_Sensor
4. https://forum.arduino.cc/t/ph-turbidity-temperature-tds-sensor-monitoring-wirelessly-using-arduino-uno-and-esp-015/1086514

APPENDIX:

1. Arduino UNO: The Arduino UNO is a microcontroller board that acts as the central control unit for the system, processing data from the sensors and displaying the results. It interfaces with the sensors and LCD to monitor water quality in real-time.



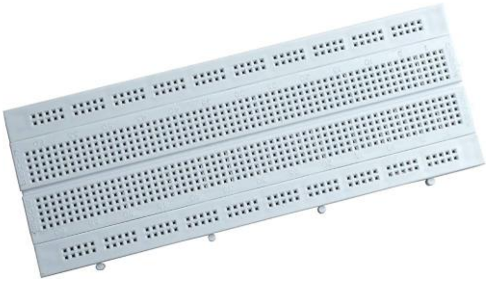
2. TDS Sensor: The TDS sensor measures the concentration of Total Dissolved Solids in the water, which helps in assessing water purity and contamination. It provides real-time readings of dissolved salts, minerals, and other substances in the water.



3. Turbidity Sensor: The turbidity sensor detects the cloudiness or turbidity of water, which can indicate the presence of suspended particles or contaminants. It measures water clarity to help identify potential contamination.

4. 16x2 LCD Display: The 16x2 LCD display shows the processed data from the sensors, such as TDS levels and turbidity values, in a readable format. It provides real-time feedback about the water quality to the user.

5. Breadboard: The breadboard is used to build and test the electronic circuit without soldering, allowing easy connections between the Arduino, sensors, and other components. It facilitates rapid prototyping of the monitoring system.



6. LED: The LED serves as a visual indicator, alerting users to unsafe water quality based on preset thresholds for TDS and turbidity. It provides immediate feedback when water quality falls below acceptable standards.

7. Jumper Wires: Jumper wires are used to establish connections between components, such as sensors, the Arduino board, and the LCD display. They provide flexible, quick connections for prototyping and circuit assembly.

8. Resistors: Resistors are used to limit the current flowing through components, protecting them from potential damage. They ensure the proper functioning of the circuit by maintaining safe current levels.

Arduino Code:

```
#include <LiquidCrystal_I2C.h>

#include <EEPROM.h>

#include "GravityTDS.h"


#define turbidityPin A0

#define TdsSensorPin A1

#define greenLEDPin 8

#define yellowLEDPin 9

#define redLEDPin 10


GravityTDS gravityTds;


// Initialize the LCD with the I2C address (usually 0x27 or 0x3F for 16x2 LCDs)

LiquidCrystal_I2C lcd(0x27, 16, 2);


float temperature = 25, tdsValue = 0;


void setup() {

  Serial.begin(9600);

  pinMode(turbidityPin, INPUT);


  // Set up LCD

  lcd.begin();

  lcd.backlight();

  lcd.setCursor(0, 0);

  lcd.print("Water Quality");


  // Initialize TDS sensor

  gravityTds.setPin(TdsSensorPin);

  gravityTds.setAref(5.0);  // Reference voltage on ADC, default 5.0V on Arduino UNO
```

```
  gravityTds.setAdcRange(1024);  // 1024 for 10bit ADC; 4096 for 12bit ADC
  gravityTds.begin();  // Initialization

  // Initialize LED pins
  pinMode(greenLEDPin, OUTPUT);
  pinMode(yellowLEDPin, OUTPUT);
  pinMode(redLEDPin, OUTPUT);

  delay(2000);  // Allow time to read "Water Quality" message on LCD
}

void loop() {
  // Read turbidity sensor
  int turbiditysensorValue = analogRead(turbidityPin);
  int turbidity = map(turbiditysensorValue, 0, 750, 100, 0);

  // Temperature setting and TDS value calculation
  gravityTds.setTemperature(temperature);  // Set temperature for compensation
  gravityTds.update();  // Sample and calculate
  tdsValue = gravityTds.getTdsValue();  // Get the TDS value

  // Display turbidity and TDS on Serial Monitor
  Serial.print("Turbidity: ");
  Serial.print(turbidity);
  Serial.println(" NTU");

  Serial.print("TDS: ");
  Serial.print(tdsValue);
  Serial.println(" ppm");

  // Determine water quality and update LEDs and LCD display
```

```
String quality;
if (turbidity <= 5 && tdsValue < 300) {
  // Good water quality
  digitalWrite(greenLEDPin, HIGH);
  digitalWrite(yellowLEDPin, LOW);
  digitalWrite(redLEDPin, LOW);
  quality = "Good";
} else if ((turbidity > 5 && turbidity <= 50) || (tdsValue >= 300 && tdsValue < 600)) {
  // Moderate water quality
  digitalWrite(greenLEDPin, LOW);
  digitalWrite(yellowLEDPin, HIGH);
  digitalWrite(redLEDPin, LOW);
  quality = "Moderate";
} else {
  // Poor water quality
  digitalWrite(greenLEDPin, LOW);
  digitalWrite(yellowLEDPin, LOW);
  digitalWrite(redLEDPin, HIGH);
  quality = "Poor";
}

// Update LCD with quality, turbidity, and TDS values
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Quality: ");
lcd.print(quality);  // Display quality on the first line

lcd.setCursor(0, 1);
lcd.print("T:");
lcd.print(turbidity);
lcd.print("NTU ");
```

```
    lcd.print("D:");
    lcd.print(tdsValue);
    lcd.print("ppm");

    delay(1000);  // Delay for 1 second
}
```