LIQUID QUALITY MONITORING USING TDS SENSOR INTERFACING WITH ESP32

Submitted in partial fulfillment of the requirements for the award of a Bachelor of Technology in Biomedical Engineering.

By,

ATCHAYA E (39240009)

SABITHA S (39240034)

DEPARTMENT OF BIOMEDICAL ENGINEERING

SCHOOL OF BIO AND CHEMICAL ENGINEERING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC I 12B Status by UGC I Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY - 2022

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

DEPARTMENT OF BIOMEDICAL ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Interdisciplinary Project Report is the bonafide work of **Atchaya E (39240009), Sabitha S (39240034)** who have done the interdisciplinary project work as a team who carried out the project entitled **"Liquid quality monitoring using TDS sensor interfacing with esp32"** under our supervision from **January 2022** to **May 2022.**

Internal Guide
(**SINDU DIVAKARAN**)

Head of the Department
(**Dr. T. SUDHAKAR**)

Submitted for Viva-voce Examination held on 30.04.2022.

Internal Examiner **External Examiner**

DECLARATION

We, **Atchaya E (39240009)** and **Sabitha S (39240034)** hereby declare that the Project Report entitled **“Liquid quality monitoring using TDS sensor interfacing with esp32”** was done by us under the guidance of **Sindu Divakaran,** Department of Biomedical Engineering at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Biomedical Engineering.

DATE: 25/04/2022

PLACE: Chennai

1. **Atchaya E**

2. **Sabitha S**

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the **Board of Management of SATHYABAMA** for their kind encouragement in doing this interdisciplinary project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. T. Sudhakar M.Sc., Ph.D., Heads of the Department, Dept. of Biomedical Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our interdisciplinary Project Guide **Sindu Divakaran, B.Tech., M.E., Department of Biomedical Engineering** for her valuable guidance, suggestions, and constant encouragement paving the way for the successful completion of our Interdisciplinary project work.

We wish to express our thanks to all **Teaching and Non-teaching** staff members of the **Department of Biomedical Engineering** who were helpful in many ways for the completion of the Interdisciplinary Project.

ABSTRACT

The liquid is a vital nutrient that is essential for human life. Water is the second most crucial factor in human survival, behind oxygen. A human can go for several weeks without eating, but only a day without water. Every living thing on Earth is encroached upon by sources. To evaluate the quality of water saved, a rigorous monitoring strategy is required as it may harm aquatic life and human health by causing cholera, amoebiasis, lead poisoning, etc., hence water quality monitoring is crucial. Real-time monitoring and management trigger prompt warning, providing timely reaction to pollution in conserving, maintaining aquatic habitat, boosting agricultural productivity by managing irrigated water quality, saving human health, and so on. Laboratories and research institutions require almost clean or distilled water to conduct conclusive studies. Water with a low pH and no pollution is thus necessary for obtaining precise and trustworthy findings. Chemicals and other toxic substances are frequently discharged into rivers by production and manufacturing enterprises, endangering aquatic life. Water quality monitoring is now done in traditional labs, which is time-intensive and prone to mistakes. As a result, this work aims to look at the viability of developing an ESP32-based sensor system for monitoring the quality of water and other liquids.

KEYWORDS: liquid, quality, aquatic life, human health, laboratories, low pH, monitoring, crucial, ESP32

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER – 1

INTRODUCTION

Water is crucial for a variety of reasons, ranging from recreation to agricultural irrigation. Additionally, plants and animals rely on clean water. Water covers more than 70% of the Earth's surface and accounts for 50–80% of all living things.

Fresh, clean, drinking water accounts for barely one-half of one percent of all water on the planet. As a result, we all bear responsibility for managing and maintaining our water resources, and one method to do so is through water quality monitoring.

Human population increase has expedited contamination and exacerbated water resource issues. Water contamination status could be determined using biological, chemical, and physical indicators. Plankton, bacteria, and other biological parameters are examples of biological parameters. pH, dissolved oxygen, nitrate, nitrite, phosphate, and ammonia are all chemical parameters. Temperature, turbidity, color, and odor are examples of physical parameters. Humans want clean water for a variety of reasons, including everyday necessities, industrial purposes, farms, agriculture, and so on.

When these parameters are monitored, it is expected that a consistent set of data would be obtained. As a result, a continuous series of abnormal data would signal the possible entry of a water contaminant, and the user would be warned of this behavior using technology. The locations were limited to industrial zones, sewage waste openings, and city limits where human involvement had a significant impact. The data logging method would result in an approved format application. The third step was to choose an acceptable, competent, and accurate method of analysis. The system has been tested so that these sensor nodes can make judgments and generate warnings when anomalies are identified.

CHAPTER – 2

LITERATURE SURVEY

2.1 INTRODUCTION

The following shows the survey done for Liquid quality monitoring using TDS sensor interfacing with esp32. The most popular of the existing research has been discussed as follows.

2.2 LITERATURE REVIEW

Nitin Dhawas(Jan 2020) made a publication where the proposed block diagram consists of several sensors which are connected to the core controller. The main controller is accessing the sensor values and processes them to transfer the data through the internet. ESP8266 is used as a core controller. The sensor data can be viewed on the internet using the built-in Wi-Fi module on ESP8266. The architecture consists of a pH Sensor, Flow Sensor, Ultrasonic Sensor, ESP8266, and Solenoid Valve. The goal of this project was to create a portable system that could control the number of water-borne infections while also solving the problem of water waste. It falls under the umbrella of the Internet of Things (IoT). The goal was to create a smart system for monitoring water quality, approximating tank water levels, and preventing overflow. The sensors are utilized for water monitoring and control. The information received from all of the sensors is used for research purposes to better manage water challenges. A Wi-Fi module transmits the data to the cloud server and can monitor sensor values and assess water usage using the smart system.

Said Sulaiman Ambu Saidi(2021) described a thorough investigation into how to create the Arduino UNO, a smartphone-based liquid-level monitoring device. ATMEL has been able to develop Arduino in recent years by creating the Arduino UNO. In our project, we utilize the Arduino UNO R3 to measure the liquid level and monitor it using a smartphone equipped with a Bluetooth module, and the output level is displayed on the I2C LCD screen. Because we employ an ultrasonic sensor instead of a float sensor, it can measure any sort of liquid. Based on Arduino, they created a wireless water level monitoring device. An Arduino, an ultrasonic sensor, and a Bluetooth module make up the system. The Arduino gets level information from the sensors and uses preconfigured level indicators to track the liquid level. The Bluetooth module takes the command from Arduino and sends it through Bluetooth to the registered mobile phone. They also installed a buzzer as an additional indicator.

S.K. Kamble(2019) in his paper presentation used the sensor modules, specifically the three sensors - pH, turbidity, and temperature, are used to collect input parameters. The Node MCU board receives analog output signals from sensors after they have been converted to digital values. With the help of the GPS module, information about these parameters is transmitted over the internet from the project module to the web page or desired PC location. Because the system is real-time and operates with the help of IoT aid technology, we can obtain the current and exact parameter values at any moment. Simultaneously, we are verifying the parameter levels so that if the values given via the web page do not fall within the required range, an SMS will be sent to the source industry.

M Ramprasath(Mar 2020) in his paper described the functioning model by combining an Arduino UNO (R3) with an ESP8266 module for wireless connectivity. The embedded device is connected to the internet via the built-in ADC and ESP8266 module. Sensors are attached to an Arduino UNO board for monitoring. An ADC will convert the sensor reading to a digital value, and the corresponding water parameters (pH, NTU) and other parameters (level, flow, and object detection) will be confirmed based on that value. When a good connection is established with the server node, the sensed data will be automatically supplied to the user through MQTT after sensing data from several sensors positioned in a specific region of interest.

Yashwanth Gowda K.N (2020) in his work proposed that according to the code uploaded to the microcontroller, the system operates automatically and autonomously. Three sensors are utilized in this system to measure the critical water parameters. The circuits of the sensors are linked to the microcontroller, and the probes of the turbidity, pH and temperature sensors are submerged in water. The ultrasonic sensor is linked to the system and is positioned on top of the water container. After touching the water's surface, the ultrasonic sensor sends electromagnetic waves to it and gets them back. The distance that shows the water level in the container is estimated by the ultrasonic sensor based on the time it takes to send and receive the wave and the velocity of the electromagnetic waves. All sensors read the water quality parameters and relay the information to the microcontroller via electrical signals. The microcontroller is configured to examine the outcome and compare it to the specified standard ranges in the code. The alarm system will activate if any water parameter exceeds the standard limit.

Karthik Maheshwari develops an Internet of Things (IoT)-based system for monitoring several factors relevant to water quality and water resource management. It comprises several sensors for detecting water flow and level and TDS and turbidity to determine the water's quality. As a microcontroller, the Arduino Uno is connected to an Ethernet Shield, which includes a water flow sensor, a water level sensor, a turbidity sensor, and a TDS sensor. For the analysis, water samples with various amounts of contaminants are used. The data collected by the turbidity, TDS, and flow sensors is processed by the microcontroller before being sent to the ThingSpeak platform.

Dr.G.Fathima proposed to detect the number of various factors present in water, the suggested system initially measures the quantity of several characteristics present in water using sensors such as temperature, pH, turbidity, and conductivity/TDS. The Arduino Mega – 2560 board, which has an AtMega Microprocessor, is linked to these sensors. The Arduino Mega board is linked to the Wi-Fi and GSM modules. The Microcontroller's acquired data is sent to the Firebase cloud. The Machine Learning model collects and processes data from the Firebase cloud to verify the quality of water by building a web page. The assessed result is shown alongside graphical data on measurement variances over time.

I M Hakim presented the Kolora meter as a replacement for commercially available monitoring systems. The microcontroller and Wi-Fi connection was constructed using the open-source platform Arduino UNO model and NodeMCU board, respectively. In the early stages of Kolora meter development, two sensors were chosen to be installed: temperature and turbidity. The physical characteristics of water (temperature and turbidity) were measured, and the obtained data may be accessed and monitored on a mobile phone over a Wi-Fi connection using the Kolora Mobile Application. As a result, because of the restricted mobility of COVID-19, this surface water quality device has the potential to be used in real-time monitoring for early pollution identification and during pandemic spread.

The sensor is read by the Arduino Intel Galileo Gen 2 board, which is based on a 32-bit MCU, as Farmanullah Jan showed. The data is then sent to an ESP826 Wi-Fi transceiver module inserted into a Galileo board's mini-PCI slot, which then delivers it to the ThingSpeak IoT Platform through a local Wi-Fi router. The inbuilt TCP/IP protocols of the ESP826 Wi-Fi transceiver can be modified. The sensor is read by the Arduino Intel Galileo Gen 2 board, which is based on a 32-bit microcontroller, according to Farmanullah Jan. The data is then sent to an ESP826 Wi-Fi transceiver module inserted into a Galileo board's mini-PCI slot, which then delivers it to the ThingSpeak IoT-Platform through a local Wi-Fi router. TCP/IP protocols are built into the ESP826 Wi-Fi transceiver and are modified for it.

Vaishnavi V. Daigavane demonstrated the hardware, which includes sensors for measuring real-time data, an Arduino atmega328 for converting analog to digital values, and an LCD for displaying sensor output. A Wi-Fi module connects the hardware and software. In terms of software, we created a program using the embedded C programming language. In the first phase of construction, the PCB is designed and components and sensors are put on it. To view the output, the BLYNK app is installed on the Android version. When the system is turned on, DC is sent to the kit and the Arduino, and the WIFI is turned on. One by one, the parameters of water are tested, and the results are shown on the LCD. The software came with a hotspot, which allows you to connect to the internet.

CHAPTER – 3

AIM AND SCOPE

3.1 AIM

The main idea of this research work is to provide an efficient liquid quality monitoring system. In this project, we propose to determine the quality of the liquid in its natural state and assess the nature and extent of pollution control needed in different water bodies and the impact of human activities on quality and suitability.

This project has numerous sections where this can be employed, such as laboratories, aquaculture, cooling towers, sewage treatment plants, and so on. As a result, we created a low-cost yet quantitative and qualitative monitoring system that is combined with TDS and temperature sensors to measure its nature.

3.2 EXISTING SYSTEM

It is a semi-automated or manually managed method that requires human intervention. There is a requirement for human interaction in taking various water parameter values. These samples are tested at state-of-the-art laboratories.

Samples are taken to these laboratories for analysis, which can be done by evaporating the liquid and collecting the remnants. The amount of unknown content in the known volume of liquid is then determined using an analytical balance and weighing the residual.

The problem with this approach is that water is not continuously monitored and constantly requires human intervention.

3.3 PROPOSED SYSTEM

As the system's fundamental controller, we propose using a constructed Esp32 microcontroller. According to the coding, the system operates automatically and autonomously.

Temperature and TDS are two important water factors that may be detected by this system, as well as the probes of the sensors placed inside the water.

All sensors read the water quality parameters and relay the information to the microcontroller via electrical signals. The microcontroller is configured to examine the outcome which is converted into digital values and compare it to the specified standard ranges in the code.

The alarm system will activate if any water parameter exceeds the standard limit. If the microcontroller detects an irregularity in a water parameter, the buzzer will sound to signal that the water is unsafe to consume.

3.4 SCOPE

To examine water quality data, a variety of statistical approaches or methodologies are available. Because water shortage is developing due to a lack of rainfall in many locations, future study scope includes determining the water quality for recycled water, research linked to changing hard water into soft water, and so on.

Creating a single prototype that is integrated with sensors for detecting organic and inorganic matter, ORP, BOD, dissolved oxygen, algae growth, bacterial growth, and other parameters. As a result, there will be no need to use various devices to measure different parameters.

CHAPTER – 4

MATERIALS AND METHOD

4.1 ESP 32

ESP32 is a family of low-cost, low-power systems-on-chip microcontrollers with integrated Wi-Fi and dual-mode Bluetooth. The ESP32 series is powered by a Tensilica Xtensa LX6 dual-core or single-core CPU, a Tensilica Xtensa LX7 dual-core or single-core RISC-V microprocessor, and contains built-in antenna switches, RF balun, filters, power amplifier, power management modules and, low-noise receive amplifier. It can operate reliably in industrial conditions with operational temperatures ranging from –40°C to +125°C. ESP32 can dynamically remove exterior circuit defects and respond to changes in external conditions thanks to improved calibration circuitries.

ESP 32 can function as a stand-alone system or as a slave device to a host MCU, eliminating communication stack overhead on the primary application CPU. Through its SPI / SDIO or I2C / UART interfaces, the ESP32 may connect to other systems to provide Wi-Fi and Bluetooth capability.

4.1.1 *Technical specification*

- Tensilica Xtensa 32-bit LX6 microprocessor with 2 cores
- External flash memory: up to 16 MB is supported
- Current consumption: 20µA – 240mA. In DeepSleep-Mode only 5µA
- Power supply: 2.3V – 3.6V
- Operating temperature range: -40°C – 125°C
- Interfaces

  UART/SDIO/SPI/I2C/I2S/IR Remote Control

  36 programmable I/O pins max 20mA

  2 analog input 0V to 1V with 12-bit resolution

  all inputs tolerate a maximum of 3.6V
- Network

  WiFi, Bluetooth: v4.2 BR/EDR and Bluetooth Low Energy (BLE)

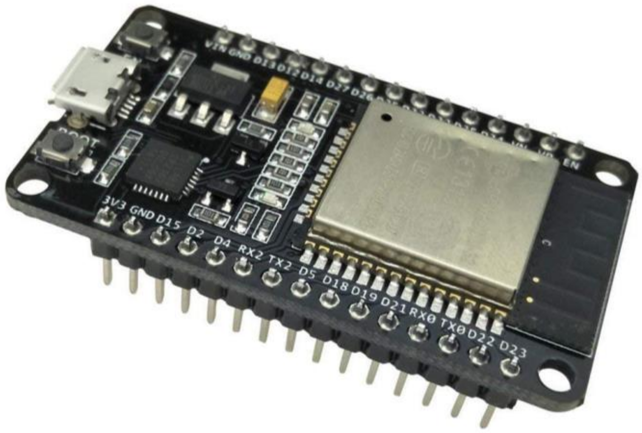

Fig: 4.1 - ESP 32 microcontroller

4.2 BREADBOARD

A breadboard is a popular tool for designing and testing circuits. When utilizing a breadboard, you do not need to solder wires or components to create a circuit. It is easier to mount and reuse components. Because the components are not soldered, you can change your circuit design at any time. The plastic box has several holes that are organized in a specific pattern. A typical breadboard layout is made up of two sorts of regions known as strips. Strips for buses and sockets Bus strips are commonly used to give electricity to a circuit. It is divided into two columns, one for power voltage and the other for ground. The majority of the components in a circuit are held in place by socket strips. It is divided into two portions, each with 5 rows and 64 columns. From the inside, each column is electrically connected.

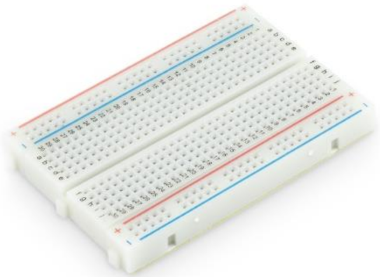

Fig: 4.2 - Breadboard

4.3 LCD DISPLAY

LCD 16×2 is a type of electronic gadget that displays data and messages. As the name implies, it has 16 Columns and 2 Rows, allowing it to display 32 characters (16×2=32) in total, with each character made up of 5×8 (40) Pixel Dots. The basic premise of LCD is the passage of light from one layer (sheet) to another using module. The modules vibrate and align their positions at 90 degrees, allowing light to pass through the polarized sheet. The molecules are in charge of displaying the data on each pixel. To display the numeral, each pixel uses the light-absorbing technique. To demonstrate the value, molecules must shift their position to adjust the angle of light. As a result of this light deflection, the human eye will see the light of the remaining part, which will turn the black section into a value and digits on the grid pixels. We can see that the data will be the part where the light is absorbed. The data will be transmitted to the molecules and will remain there until they are altered.

Fig: 4.3 - LCD 16×2 display

4.4 TDS SENSOR

The TDS value of water is measured with the DFROBOT Gravity Analog TDS Sensor. It can be used to examine the quality of home water, hydroponics, and other liquids. This product accepts 3.3 - 5.5V broad voltage input and outputs 0 - 2.3V analog voltage, making it suitable with 5V or 3.3V control systems or boards. The excitation source is an AC signal, which may effectively prevent the probe from polarization and extend the probe's life while also helping to strengthen the output signal's stability. The TDS probe is waterproof and can be immersed in water for an extended period.

Fig: 4.4 – DFROBOT TDS sensor

4.5 TEMPERATURE SENSOR

Maxim Integrated's DS18B20 is a 1-wire programmable Temperature sensor. It is commonly used to detect the temperature in harsh settings such as chemical solutions, mines, or soil. The sensor's constrictions are tough, and it can also be purchased with a waterproof variant, making attachment simple. It has a good accuracy of 5°C and can measure temperatures ranging from -55°C to +125°. Each sensor has a unique address and uses only one MCU pin to transport data, making it an excellent choice for sensing temperature at several sites without compromising many of the microcontroller's digital pins.

Fig: 4.5 – Temperature sensor probe

4.6 BUZZER

The buzzer is a speaking device that translates sound impulses from an audio model. It is mostly used to alert or prompt. It may produce a variety of sounds, including music, flute, buzzer, alarm, electric bell, and other noises, depending on the design and application.

Fig: 4.6 – Piezoelectric buzzer

4.7 ARDUINO IDE

The Arduino IDE is free and open-source software for writing and uploading code to Arduino boards. The IDE generates a Hex file when a user writes code and compiles it. The data is subsequently transferred to the board through a USB wire.

4.7.1 *FEATURES*

- Modern, fully-featured development environment
- Dual Mode, Classic Mode (identical to the Classic Arduino IDE), and Pro Mode (File System view)
- New Board Manager
- New Library Manager
- Board List
- Basic Auto-Completion (Arm targets only)
- Git Integration

- Serial Monitor
- Dark Mode
- Sketch synchronization with Arduino Create Editor
- Debugger
- Fully open to third party plug-ins
- Support for additional languages other than C++

4.8 BLOCK DIAGRAM

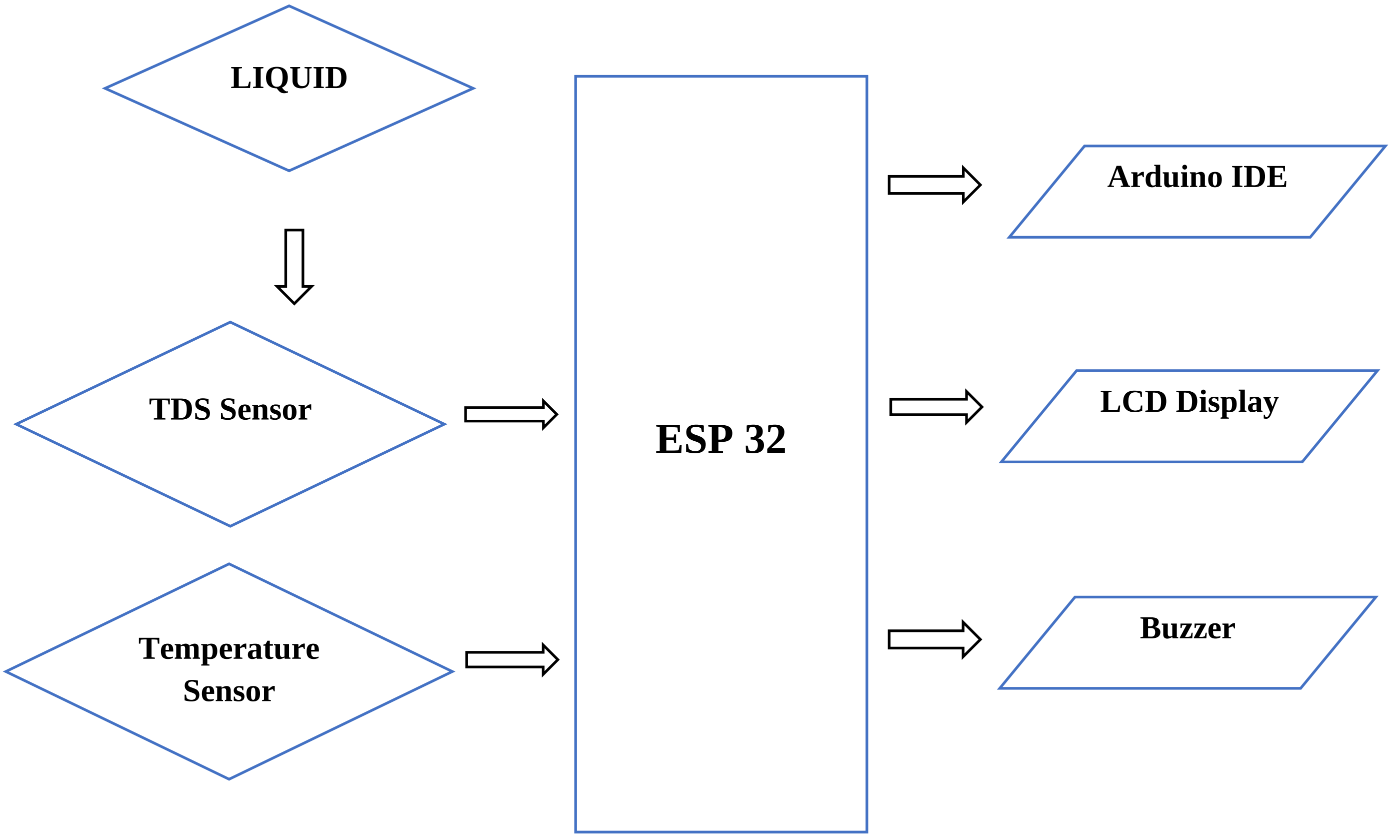

Fig: 4.7 - Block diagram

4.9 WORKING

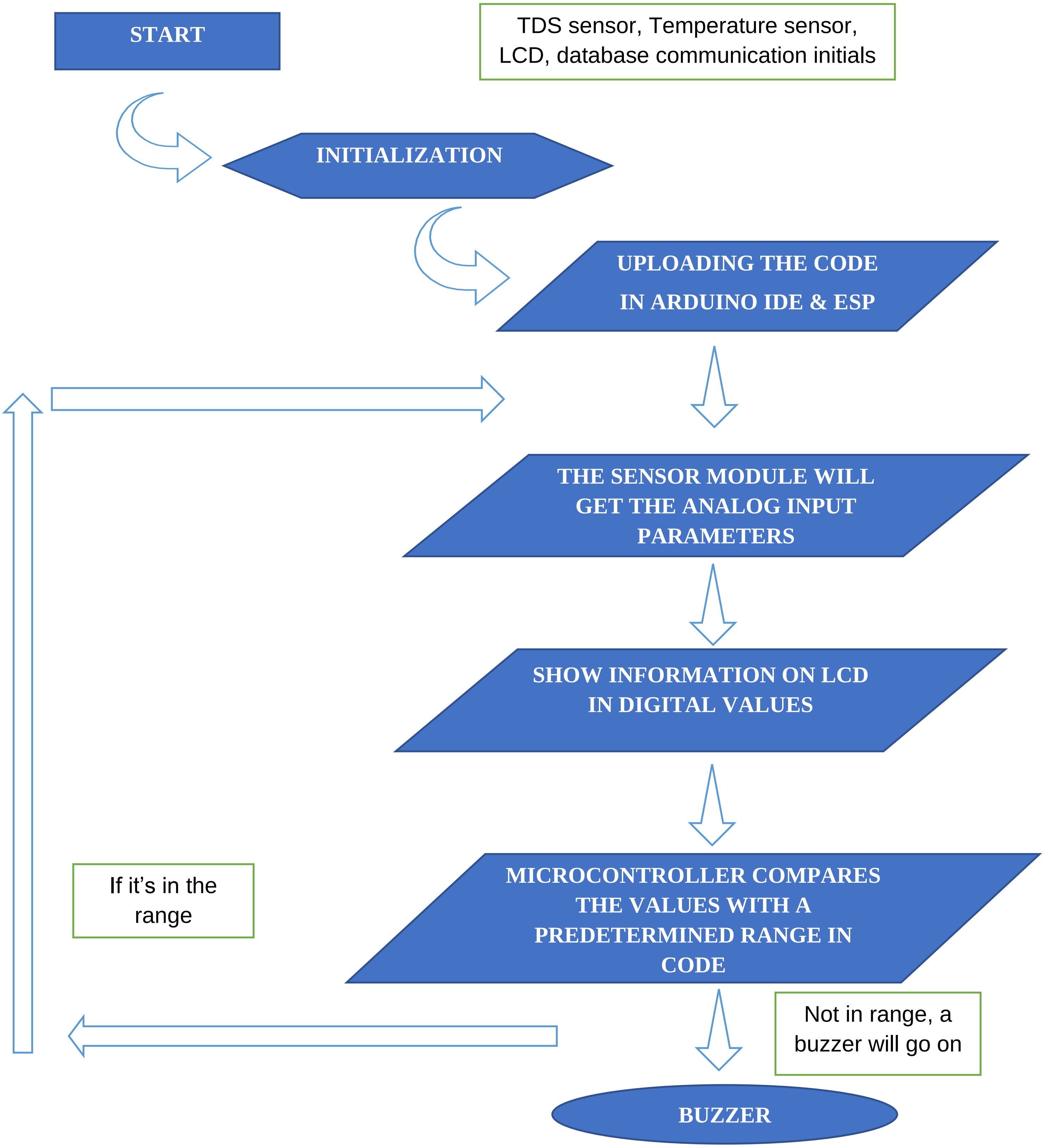

Fig: 4.8 – Workflow of the device

CHAPTER – 5 RESULTS & DISCUSSION

Fig: 5.1 – Hardware and result

Table 5.1 – TDS sensor and temperature Test Data

NO.	Sample	TDS Value	DS18B20 Sensor	Thermometer	Temperature Difference
1.	Distilled water	0.2	25.1	25	0.1
2.	Water in aquarium	214	23.16	23	0.16
3.	Mineral water	766	26.65	27	0.35
4.	Drinking water	480	28.32	28	0.32
5.	River water	337	15.03	15	0.03
6.	Coffee	301	75.60	76	0.4
7.	Mirinda	294	26.25	26	0.25
8.	Pepsi	588	27.08	27	0.08

CHAPTER – 6

CONCLUSION

6.1 CONCLUSION

This work aimed to develop a low-cost, efficient, real-time, adaptable, easily configurable, and, most significantly, a portable system capable of controlling the amount of water-borne diseases caused by dirty water and providing accurate results in laboratories. Some sensors are used for this. The data obtained from all of the sensors are analyzed to provide better solutions. This method is useful for the Pollution Control Department since it allows them to monitor the various industrial water outlets to their neighboring water resources, which reduces human work and improves health. The quality of any sort of liquid can be determined with this approach. As a result, it is a straightforward technique for monitoring the water level system.

6.2 SUMMARY

The current study leads to the following points:

a) The test results from the design of a microcontroller-based liquid quality monitoring system reveal good performance, even though the measurement findings were not constant or variable.

b) More study is needed to evaluate measurements with chemical parameters to generate an accurate value.

REFERENCE

1) Chetna Gawade et al., Advanced Water Quality Monitoring with IoT - IRJET - Vol.07, Issue 1 – 2020 - Page no. [1783-1785]

2) Said Sulaiman Ambu Saidi et al., Arduino Based Smart Phone Liquid Level Monitoring System Using Ultrasonic Sensor - IJERA - Vol. 11, Issue 8 – 2021 - Page no.[41-45]

3) S.K. Kamble et al., IoT-based Industrial Water Quality Monitoring System - IRJET - Vol.06, Issue 5 – 2019 - Page no.[7952-7955]

4) M Ramprasath et al., IoT-based Water Quality Monitoring and Flow Control of Tank Water - IRJET - Vol.07, Issue 3 – 2020 - Page no.[43-47]

5) Yashwanth Gowda K.N et al., Real-Time Water Quality Monitoring System - IJERT - Special Issue – 2020 - Page no.[142-145]

6) Karthik Maheshwari and Adrija Chakraborty., Water Quality Monitoring System Implemented With IoT – IJETER – Vol.09, Issue 7 – 2021 - Page no. [1020 – 1024]

7) Dr. G. Fathima et al., An Automated System For Effective Monitoring of Water Quality – IJRCS – Vol. 04, Issue 3 – 2020 - Page no. [152 – 158]

8) I M Hakim et al., Development of Water Quality Monitoring Device Using Arduino UNO – IOP – 2021 – Page no.[31 – 39]

9) Farmanullah Jan - IoT Based Smart Water Quality Monitoring: Recent Techniques, Trends, and Challenges for Domestic Applications – MDPI – Vol. 03 – 2021 – Page no.[61 – 98]

10) Vaishnavi V. Daigavane and Dr. M.A Gaikwad - Water Quality Monitoring System Based on IoT – ISSN – Vol.03 – Page no.[1107 – 1116]

APPENDIX

```
#define TdsSensorPin 27

#define VREF 3.3                    // analog reference voltage (Volt) of the ADC

#define SCOUNT  30                  // sum of sample point

#define LiquidCrystal_h

#include <LiquidCrystal.h>

int analogBuffer[SCOUNT];          // store the analog value in the array, read from ADC

int analogBufferTemp[SCOUNT];

int analogBufferIndex = 0;

int copyIndex = 0;

const int TdsSensorPin = 27;

int liqval;


float averageVoltage = 0;

float tdsValue = 0;

float temperature = 25;            // current temperature for compensation


int getMedianNum(int bArray[], int iFilterLen){        // median filtering algorithm

  int bTab[iFilterLen];

  for (byte i = 0; i<iFilterLen; i++)

  bTab[i] = bArray[i];

  int i, j, bTemp;

  for (j = 0; j < iFilterLen - 1; j++) {
```

```
    for (i = 0; i < iFilterLen - j - 1; i++) {
      if (bTab[i] > bTab[i + 1]) {
        bTemp = bTab[i];
        bTab[i] = bTab[i + 1];
        bTab[i + 1] = bTemp;
      }
    }
  }
  if ((iFilterLen & 1) > 0){
    bTemp = bTab[(iFilterLen - 1) / 2];
  }
  else {
    bTemp = (bTab[iFilterLen / 2] + bTab[iFilterLen / 2 - 1]) / 2;
  }
  return bTemp;
}
void setup(){
  Serial.begin(115200);
  lcd.begin(16,2);
  sensors.begin();
  pinMode(TdsSensorPin,INPUT);
}
void loop(){
```

```
    static unsigned long analogSampleTimepoint = millis();

  if(millis()-analogSampleTimepoint > 40U)

{

    analogSampleTimepoint = millis();

    analogBuffer[analogBufferIndex] = analogRead(TdsSensorPin

    analogBufferIndex++;

    if(analogBufferIndex == SCOUNT){

      analogBufferIndex = 0;

    }

  }

static unsigned long printTimepoint = millis();

if(millis()-printTimepoint > 800U){

  printTimepoint = millis();

  for(copyIndex=0; copyIndex<SCOUNT; copyIndex++){

     analogBufferTemp[copyIndex] = analogBuffer[copyIndex];

     averageVoltage = getMedianNum(analogBufferTemp,SCOUNT) * (float)VREF /
4096.0;

     float compensationCoefficient = 1.0+0.02*(temperature-25.0);

     float compensationVoltage=averageVoltage/compensationCoefficient;
//temperature compensation ，    //convert voltage value to tds value

tdsValue=(133.42*compensationVoltage*compensationVoltage*compensationVoltag
e- 255.86*compensationVoltage*compensationVoltage +
857.39*compensationVoltage)*0.5;

  liqval = analogRead(TdsSensorPin);
```

```
    Serial.print("TDS Value:");
    Serial.print(tdsValue,0);
    Serial.println("ppm");
    lcd.noCursor();
    lcd.setCursor(0,0);
    lcd.print("TDS: ");
    lcd.print(tdsValue,0);
    lcd.print(" PPM");
    lcd.setCursor(0,1);
    lcd.print("Temp: ");
    lcd.print(sensors.getTempCByIndex(0));
    lcd.print(" C");
    delay(1500);
    lcd.clear();
  if (liqval <= 800){
   noTone (27);
  }
  else{
   tone(27,900);
  }
   }
  }
}
```