AUTOMATIC WATER LEVEL SENSOR

Project report submitted in partial fulfilment of the requirements
for the Second year of Computer Science and Engineering(Iot and CS BC)

by

Abhijeet Singh	**Roll No. - 55**
Sujal Raut	**Roll No. - 68**
Apurva Waghmare	**Roll No. - 63**
Swarali Patil	**Roll No, - 74**

Under the guidance of
Prof. Amol H Patil

Department of Computer Science and Engineering(Iot and CS BC)
A. C. Patil College of Engineering, Kharghar, Navi Mumbai
University of Mumbai

2023-2024

Institute Vision and Mission

VISION

To create skilled professionals and engineers for catering the needs of industries and society.

MISSION

1. To provide qualified faculty and required infrastructure to impart quality education inculcating continuous learning attitude
2. To provide platform for the interaction between academia and industry.
3. To inculcate social values and responsible attitude amongst students through co-curricular and extracurricular activities.

Department Vision and Mission

VISION

To be a pioneering center of excellence in computer science and engineering, developing skilled professionals and engineers, aspire to meet the needs of industries and society.

MISSION

1. To provide competent faculty and necessary infrastructure to deliver high-quality education while fostering a culture of ongoing learning.
2. To Promote industrial partnerships to drive both technical and socioeconomic advancement toward a sustainable future.
3. To inculcate ethical values and responsibility among students through extracurricular and co-curricular participation.

CERTIFICATE

This is to certify that the project entitled**"Automatic Water Level Sensor"** is a bonafide work of **Abhijeet Singh Roll No.- 55**, **Sujal Raut Roll No.- 68**, **Apurva Waghmare Roll No. - 63**, **Swarali Patil Roll No. - 74**, of **A. C. Patil College of Engineering, Kharghar** have submitted to University of Mumbai in partial fulfilment of the requirement for award of second year of Engineering in Computer science and Engineering(Iot and CS BC) from University of Mumbai.

Internal Examiner

External Examiner

Dr. Ulka Shirole

Head Of Department

Dr. V. N. Pawar

Principal

Project Report Approval for S. E.

This project report entitled ”Automatic Water Level Sensor” is approved for the degree of **Computer Science and Engineering(Iot and CS BC)**.

Examiners

1. ____________________

2. ____________________

Date: ____________________

Place: ____________________

DECLARATION

I declare that this written submission represents my ideas and does not invovle plagiarism. I have adequately cited and referenced the original sources wherever others' ideas or words have been included. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action against me by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Abhijeet Singh	Roll No. - 55
Sujal Raut	Roll No. - 68
Apurva Waghmare	Roll No. - 63
Swarali Patil	Roll No, - 74

Date:

Acknowledgement

It is with immense pleasure and satisfaction that we express our gratitude and thanks to the people who have contributed in making our project work and enriching experience.

I express our gratitude towards my guide Prof.Amol H Patil for supporting throught the year for successfully completing our project.

Abhijeet Singh	Roll No. - 55
Sujal Raut	Roll No. - 68
Apurva Waghmare	Roll No. - 63
Swarali Patil	Roll No, - 74

Date:

ABSTRACT

Water level monitoring systems are crucial for various sectors, including agriculture, environmental conservation, and disaster prevention, due to their impact on human life, ecosystems, and infrastructure. Traditional methods often face challenges such as limited accuracy, high cost, and manual data collection processes. To address these issues, this paper explores the design, implementation, and evaluation of an IoT-based water level monitoring system using the ESP8266 microcontroller , ultrasonic sensors, and the Blynk platform. The system offers real-time monitoring, remote accessibility, and automated alerts, making it suitable for applications in villages where water scarcity is a pressing issue. By leveraging IoT technologies, this system provides a cost-effective and reliable solution for managing water resources and mitigating the risks of water-related disasters in rural areas. Through the implementation of such systems, it is envisioned that villages lacking adequate water management infrastructure can improve water availability, optimize agricultural practices, and enhance environmental sustainability.

List of Figures

Chapter 1

Introduction

Water level monitoring plays a crucial role in various sectors, including agriculture, environmental conservation, and disaster prevention, due to its profound impact on human life, ecosystems, and infrastructure. In agriculture, accurate monitoring of water levels in fields, reservoirs, and irrigation systems is essential for optimizing water usage, ensuring crop health, and maximizing agricultural productivity . Additionally, water level data is critical for environmental conservation efforts as it helps in monitoring water bodies, wetlands, and aquatic ecosystems, thereby enabling effective management and protection of natural resources . Furthermore, in disaster prevention, timely and accurate monitoring of water levels in rivers, lakes, and coastal areas is vital for predicting and mitigating the impact of floods, tsunamis, and other water-related disasters . Despite the significance of water level monitoring, traditional methods often face challenges such as limited accuracy, high cost, and difficulty in real-time data acquisition and analysis. Conventional monitoring techniques, including manual gauging and fixed-point sensors, are labor-intensive, time-consuming, and prone to errors . Additionally, these methods may not provide sufficient coverage or granularity, especially in large-scale or remote areas, leading to gaps in monitoring and potential risks of mismanagement or inadequate response to water-related events .Therefore, there is a growing need for innovative and IoT-based solutions that can overcome these challenges and offer efficient, reliable, and cost-effective water level monitoring capabilities.

Chapter 2

SYSTEM ARCHITECTURE AND COMPONENTS

Water level monitoring is a critical aspect of various sectors, including agriculture, environmental conservation, and disaster prevention. The accurate measurement of water levels in rivers, reservoirs, and other bodies of water is essential for managing water resources effectively, protecting ecosystems, and mitigating the risks of floods and droughts. Traditional methods of water level monitoring often face challenges such as limited accuracy, high cost, and manual data collection processes.

In recent years, the advent of Internet of Things (IoT) technologies has opened up new possibilities for improving water level monitoring systems by enabling real-time data collection, remote accessibility, and automated alerts. This paper explores the design, implementation, and evaluation of a water level monitoring system utilizing IoT technologies, specifically focusing on the integration of the ESP8266 microcontroller, ultrasonic sensors, and the Blynk platform.

(1)ESP8266 Microcontroller:

The choice of ESP8266 pins, namely D0, D1, VIN, and GND, for inter-

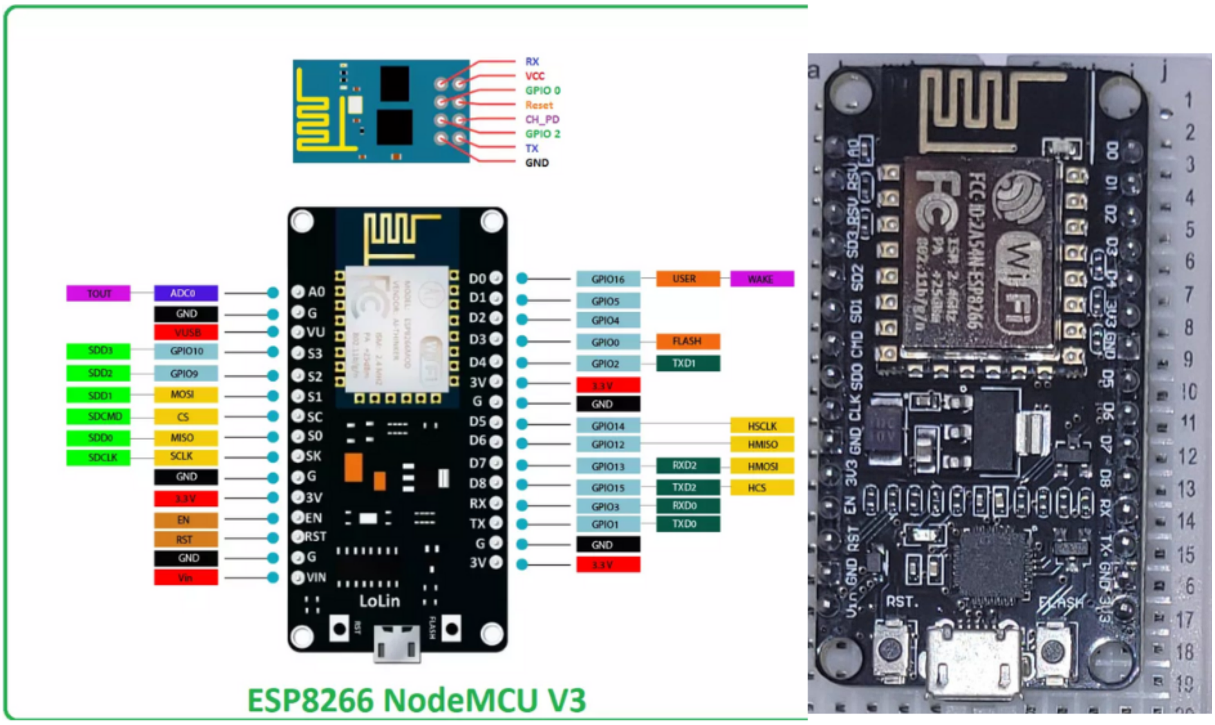

Figure 2.1: ESP 8266 with pin configuration

facing with the JSN-SR04T ultrasonic sensor is based on several factors crucial for the successful operation of the sensor within the project. Beginning with D0 and D1, these pins are strategically selected due to their versatility as general-purpose input/output (GPIO) pins on the ESP8266 board. D0 and D1 are capable of both digital input and output operations, making them ideal candidates for facilitating communication between the ESP8266 and the ultrasonic sensor. D0 serves as the trigger pin, responsible for initiating the ultrasonic pulse transmission, while D1 functions as the echo pin, capturing the reflected signal from the sensor. By utilizing these GPIO pins, the ESP8266 board can effectively control the timing and measurement processes required for accurate distance sensing.

Moreover, the VIN pin is employed to supply the requisite voltage for powering the JSN-SR04T sensor. This pin serves as the primary source of electrical power, ensuring that the sensor receives the necessary voltage level for its operation. By connecting the VIN pin to an appropriate power source on the ESP8266 board, such as the regulated 5V output, the sensor receives stable and consistent power, thereby enhancing its performance and reliability. Additionally, the GND pin, serving as the ground reference, establishes a common electrical reference point between the ESP8266 board and the ultrasonic sensor. This connection is essential for maintaining signal integrity, minimizing electrical noise, and ensuring reliable communication between the two devices.

In summary, the strategic selection and utilization of the ESP8266 pins D0, D1, VIN, and GND facilitate seamless integration of the JSN-SR04T ultrasonic sensor with the ESP8266 board. By leveraging the GPIO pins for triggering and capturing ultrasonic pulses, supplying adequate power via VIN, and establishing a stable ground connection through GND, the project achieves accurate and reliable water level measurements. This meticulous approach to pin selection and wiring ensures optimal performance of the ultrasonic sensor within the project's framework, thereby contributing to the overall success and functionality of the water level measuring system.

(2)ULTRASONIC SENSORS:

Transducers: A "transducer" is a fancy name for a device that can either convert mechanical vibrations into electrical energy or convert electrical energy into mechanical vibrations.Two common examples of transducers are microphones and speakers.Ultrasonic distance sensors employ transducers that can work at ultrasonic frequencies, usually 40KHz. Some, like the HC-SR04 or

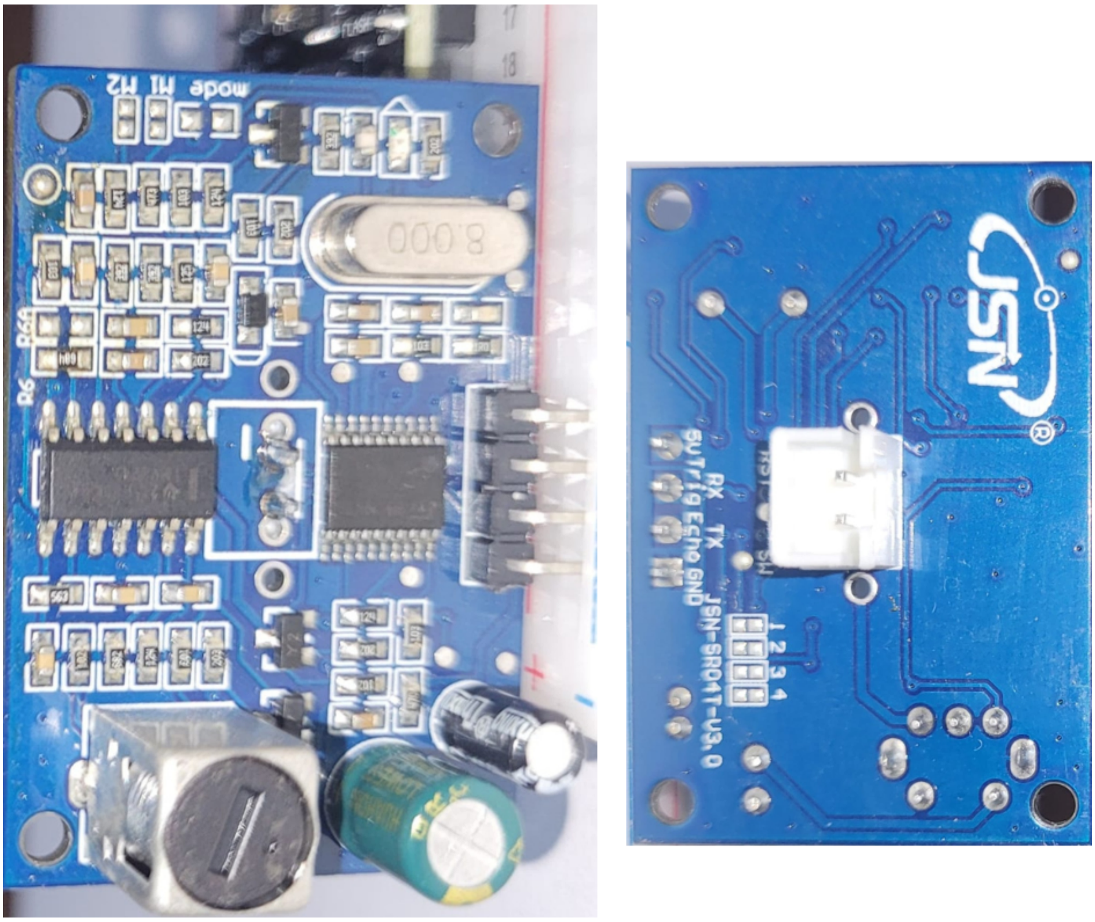

Figure 2.2: Breakout board

A02YYUW, use a separate transducer to send and receive pulses of sound. Others, like the JSN-SRT04T, employ only one transducer which can serve as either a transmitter or receiver.

Ultrasonic Pulses: In operation, the sensor transmits pulses of ultrasonic sound and then listens to see if they get reflected back. If they do, then the time delay between transmission and reception is measured. This time delay can be used to compute the distance to the object that reflected the sound.

Ultrasonic Distance Sensors Operation Speed of Sound To calculate the distance, you first need to divide the time delay in half, as it represents the back-and-forth travel of the ultrasonic pulses. You then multiply the speed of sound, 343 meters per second, by the time delay to see how far the object that

Figure 2.3: 2 meter extendable sensor cable

reflected the sound is from the transducer.

HOW WE ARE USING THE COMPONENT:

The JSN-SR04T ultrasonic sensor is interfaced with the ESP8266 microcontroller by establishing connections between specific pins on both devices. The trig pin of the JSN-SR04T sensor, responsible for triggering ultrasonic pulses, is connected to a GPIO pin on the ESP8266 board, typically denoted as D0. This GPIO pin allows the ESP8266 microcontroller to initiate the transmission of ultrasonic pulses from the sensor. Conversely, the echo pin of the JSN-SR04T sensor, used to receive the reflected ultrasonic signal, is connected to another GPIO pin on the ESP8266 board, commonly referred to as D1. This

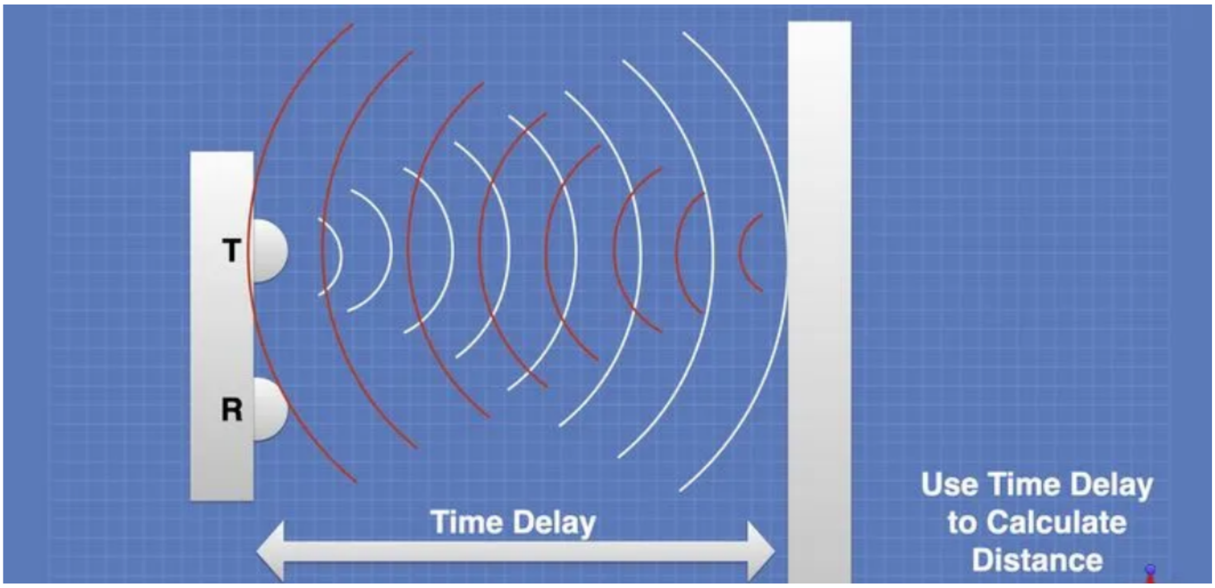

Figure 2.4: scientific representation

GPIO pin enables the ESP8266 microcontroller to detect and measure the time it takes for the ultrasonic signal to return after being emitted by the sensor.

Furthermore, the JSN-SR04T sensor requires a stable power supply to operate effectively. To meet this requirement, the 5V pin of the JSN-SR04T sensor is connected to the VIN pin of the ESP8266 board. The VIN pin of the ESP8266 microcontroller provides a regulated 5V power output, ensuring that the JSN-SR04T sensor receives the necessary voltage for its operation. Additionally, the ground (GND) pin of the JSN-SR04T sensor is connected to the ground (GND) pin of the ESP8266 board, establishing a common reference point for electrical signals and completing the circuit.

Chapter 3

Software

3.1 Software Setup:

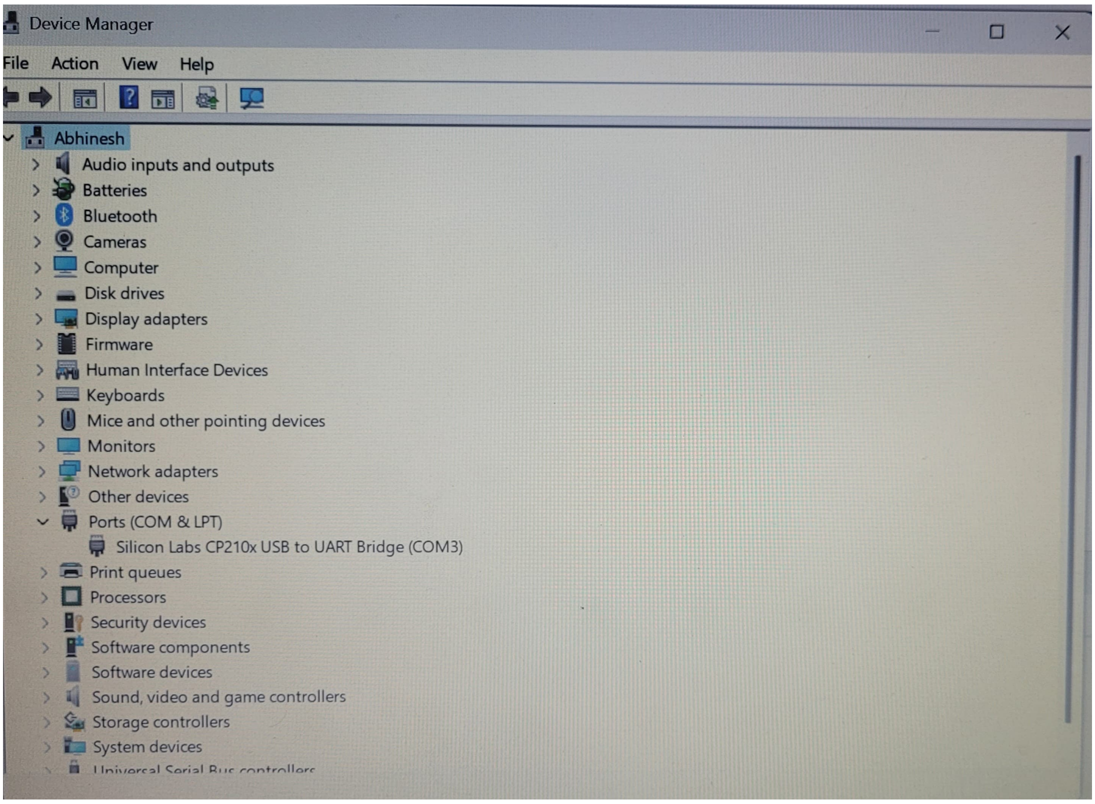

Figure 3.1: Downloaded Com3 UART bridge driver from siliconlabs for esp8266

Downloading Drivers for UART Bridge (COM3)

Next we open up our arduinoIDE download the listed libraries: ”ESPDou-

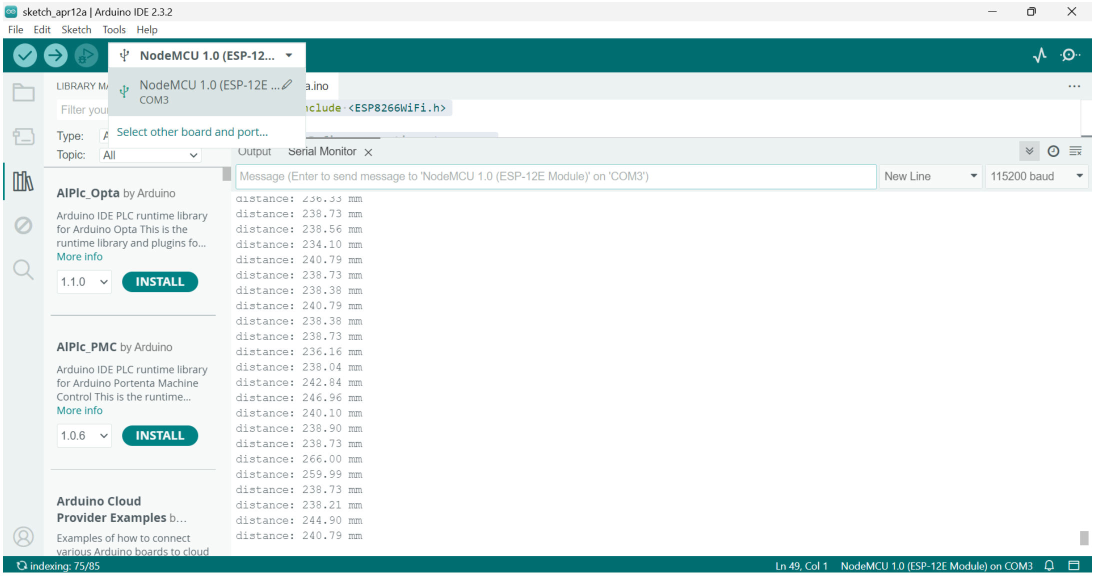

Figure 3.2: Arduino IDE

bleResetDetector" for our esp control as there is no direct port for the unit and then after we select the board and port which is com3 connected through an usb cable we get something like this

3.2 The code base we are using and its implications

Code:

This Arduino code is designed to interface an ESP8266 microcontroller with an ultrasonic sensor (such as the JSN-SR04T) for measuring distances. Let's break down the code:

1. include ESP8266WiFi.h: This line includes the necessary library for working with the ESP8266 WiFi module.

2. 'define TRIGPIN D0' and 'define ECHOPIN D1': These lines define the GPIO pins connected to the TRIG (trigger) and ECHO pins of the ultrasonic sensor, respectively. You should replace 'D0' and 'D1' with the actual GPIO pins connected to your sensor.

3. ‘float duration, distance;‘: These lines declare two floating-point variables, ‘duration‘ and ‘distance‘, which will be used to store the duration of the ultrasonic pulse and the calculated distance, respectively.

4. ‘void setup()‘: This function runs once when the microcontroller starts up. It initializes the serial communication at a baud rate of 115200 and sets the pin modes for the TRIGPIN and ECHOPIN.

5. ‘pinMode(ECHOPIN, INPUT);‘ and ‘pinMode(TRIGPIN, OUTPUT);‘: These lines set the TRIGPIN as an output pin and the ECHOPIN as an input pin, preparing them for communication with the ultrasonic sensor.

6. ‘void loop()‘: This function continuously runs after the ‘setup()‘ function. It contains the main logic for measuring distances using the ultrasonic sensor.

7. ‘digitalWrite(TRIGPIN, LOW);‘: This line sets the TRIGPIN LOW for 2 microseconds to ensure a clean pulse.

8. ‘delayMicroseconds(2);‘: This line introduces a short delay of 2 microseconds.

9. ‘digitalWrite(TRIGPIN, HIGH);‘: This line sets the TRIGPIN HIGH for 20 microseconds to trigger the ultrasonic pulse.

10. ‘delayMicroseconds(20);‘: This line introduces a delay of 20 microseconds.

11. ‘digitalWrite(TRIGPIN, LOW);‘: This line brings the TRIGPIN back to LOW state.

12. ‘duration = pulseIn(ECHOPIN, HIGH);‘: This line measures the duration of the pulse received on the ECHOPIN, which corresponds to the time taken for the ultrasonic pulse to travel to the target and back.

13. ‘distance = (duration / 2) * 0.343;‘: This line calculates the distance

based on the duration of the pulse. The speed of sound is assumed to be 343 meters per second, and the duration is divided by 2 since the pulse travels to the target and back.

14. ‘Serial.print(”distance: ”);‘, ‘Serial.print(distance);‘, ‘Serial.println(” mm”);‘: These lines print the measured distance in millimeters to the serial monitor for debugging purposes.

15. ‘delay(100);‘: This line introduces a delay of 100 milliseconds before the next measurement to prevent rapid successive readings and to allow the sensor to stabilize.

Code for arduino

```
#include <ESP8266WiFi.h>

// Define connections to sensor
#define TRIGPIN D0 // Replace D0 with the GPIO pin connected to the TRIG pin of your sensor
#define ECHOPIN D1 // Replace D1 with the GPIO pin connected to the ECHO pin of your sensor

// Floats to calculate distance
float duration, distance;

void setup() {
 // Set up serial monitor
 Serial.begin(115200);

 // Set pinmodes for sensor connections
 pinMode(ECHOPIN, INPUT);
 pinMode(TRIGPIN, OUTPUT);
}

void loop() {

 // Set the trigger pin LOW for 2uS
 digitalWrite(TRIGPIN, LOW);
 delayMicroseconds(2);

 // Set the trigger pin HIGH for 20us to send pulse
 digitalWrite(TRIGPIN, HIGH);
 delayMicroseconds(20);

 // Return the trigger pin to LOW
 digitalWrite(TRIGPIN, LOW);

 // Measure the width of the incoming pulse
 duration = pulseIn(ECHOPIN, HIGH);

 // Determine distance from duration
 // Use 343 metres per second as speed of sound
 // Divide by 1000 as we want millimeters

 distance = (duration / 2) * 0.343;

 // Print result to serial monitor
 Serial.print("distance: ");
 Serial.print(distance);
 Serial.println(" mm");

 // Delay before repeating measurement
 delay(100);
}
```

Figure 3.3: code for processing the data from sensor and through esp Arduino IDE

Chapter 4

HardwareSetup

1. **Components**: - ESP8266 microcontroller (NodeMCU or similar) - JSN-SR04T ultrasonic sensor - Breadboard - Jumper wires

2. **Connections**: - **ESP8266 to JSN-SR04T**: - Connect the TRIG pin of the JSN-SR04T to GPIO pin D0 (or any other GPIO pin you prefer) on the ESP8266. - Connect the ECHO pin of the JSN-SR04T to GPIO pin D1 (or another GPIO pin) on the ESP8266. - Connect the VCC pin of the JSN-SR04T to the 5V pin or VIN pin on the ESP8266 for power. - Connect the GND pin of the JSN-SR04T to any GND pin on the ESP8266 to complete the circuit.

3. **Power Supply**: - The ESP8266 can be powered either through a USB connection (e.g., from a computer or USB power adapter) or through an external power source (e.g., a battery or a power supply module). - Ensure that the power supply provides sufficient voltage and current to power both the ESP8266 and the JSN-SR04T sensor.

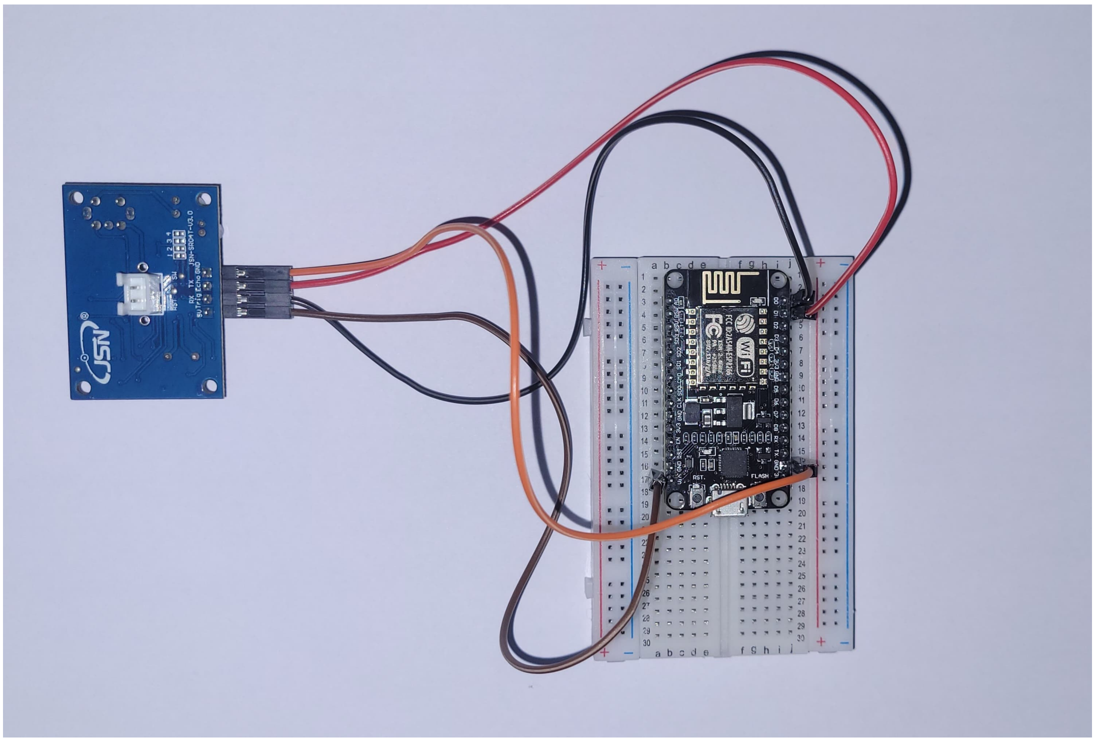

Figure 4.1: breadboardconnection

Chapter 5

Advantages and Disadvantages

Advantages:

A water level sensor project can offer several advantages, depending on its application and implementation:

1. **Water Conservation**: By monitoring water levels in tanks or reservoirs, the project can help in efficient water management, reducing wastage and promoting conservation.

2. **Prevent Overflows**: Water level sensors can prevent overflow situations by alerting users when the water level reaches a certain threshold .

3. **Automation**: Integrating water level sensors with automated systems allows for the automatic control of pumps or valves, ensuring optimal water levels without manual intervention.

4. **Cost Savings**: Efficient water management facilitated by water level sensors can lead to cost savings in terms of reduced water bills, lower maintenance costs, and minimized damage from overflow incidents.

5. **Remote Monitoring**: With the addition of IoT (Internet of Things) water level sensor projects can be monitored remotely, allowing users to keep track of water levels from anywhere.

Disadvantages :

While water level sensor projects offer numerous advantages, they also come with some potential disadvantages:

1. **Initial Cost**: The cost of purchasing and installing water level sensors and associated equipment can be a barrier, especially for individuals or organizations with limited budgets.

2. **Maintenance Requirements**: Water level sensors may require regular maintenance to ensure accurate readings and proper functioning, which can add to the overall cost and effort involved in implementing the project.

3. **Calibration**: Calibration of water level sensors is essential to maintain accuracy, and this process can be time-consuming and may require specialized knowledge or equipment.

4. **Environmental Factors**: Environmental conditions such as temperature fluctuations, humidity, and the presence of debris or contaminants in the water can affect the performance of water level sensors, leading to inaccurate readings or malfunctions.

5. **Power Source**: Depending on the type of water level sensor used, a reliable power source may be required, which could be challenging to provide in remote or off-grid locations.

Chapter 6

Future Scope

The future scope of water level sensor projects is promising, with several potential advancements and applications on the horizon:

1. **IoT Integration**: Further integration with Internet of Things (IoT) technologies will enable real-time monitoring and remote control of water levels from anywhere with an internet connection. This could involve the development of more advanced sensor nodes, cloud-based data analytics platforms, and mobile applications for user interaction.

2. **Energy Harvesting**: Research into energy harvesting techniques, such as solar or kinetic energy harvesting, could enable the development of self-powered water level sensors. This would reduce dependency on external power sources and increase the feasibility of deploying sensors in remote or off-grid locations.

3. **Miniaturization**: Advances in microelectronics and sensor technology will continue to drive miniaturization, making water level sensors smaller, more energy-efficient, and cheaper to produce. This will expand the range of applications for water level sensing in areas such as wearable devices, smart infrastructure, and environmental monitoring.

4. **Smart Cities and Infrastructure**: Water level sensor projects will play a crucial role in the development of smart cities and infrastructure. By integrating with smart grid systems, urban drainage networks, and flood warning systems, these projects can contribute to more efficient water management, disaster resilience, and sustainable urban development.

5. **Environmental Monitoring**: Water level sensors will be increasingly used for environmental monitoring purposes, such as tracking sea level rise, monitoring groundwater levels, and assessing the impact of climate change on water resources. This will provide valuable data for scientific research, policy-making, and adaptation strategies.

6. **Blockchain Technology**: Integration with blockchain technology can enhance the security, transparency, and traceability of water level data, particularly in applications involving water trading, supply chain management, and compliance reporting. Blockchain-based solutions can help prevent tampering or unauthorized access to sensitive data.

7. **Customization and Interoperability**: Future water level sensor projects will need to focus on customization and interoperability, allowing users to adapt the technology to their specific needs and integrate it with existing infrastructure seamlessly. Open-source hardware and software platforms will facilitate collaboration and innovation in this space.

Chapter 7

Conclusion

In conclusion, the proposed IoT-based water management system represents a significant advancement in addressing the challenges associated with water resource management, particularly in rural areas. By integrating components such as the ESP8266 microcontroller, ultrasonic sensors, and the Ardunio IDE, the system offers scalable, reliable, and cost-effective solutions for monitoring water levels, ensuring water quality, and mitigating the risks of water-related disasters .

The system's real-time monitoring capabilities, remote accessibility, and automated alerts provide crucial information for decision-makers, enabling proactive measures to optimize water usage, protect ecosystems, and enhance disaster resilience. Moreover, the scalability of the system ensures its adaptability to varying infrastructure and resource constraints, making it suitable for deployment in diverse environments, including remote villages with limited access to water management infrastructure .

Through the implementation of such systems, communities can improve water availability, optimize agricultural practices, and promote environmental sustainability.