

SMART WATER LEVEL MONITORING SYSTEM USING MICROCONTROLLER

*A thesis submitted in partial fulfillment of the requirements for
the award of the degree of*

Bachelor of Technology

By

Naorem Diyabarta Singh	2101CS0216
Ricky Thoudam	2101CS0203
Naoroibam Lanngamba Luwang	2101CS0209

Under the Guidance of

Dr. Hanjabam Saratchandra Sharma



DEPT OF COMPUTER SCIENCE & ENGINEERING

MANIPUR TECHNICAL UNIVERSITY

January 21, 2025

DEDICATION

We dedicate this project report to our families, friends, and mentors who have been a constant source of inspiration and support throughout our academic journey. Their unwavering encouragement and belief in our abilities have driven us to explore new ideas and achieve our goals.

We would also like to extend our heartfelt gratitude to our esteemed institution and faculty members, whose guidance and knowledge have shaped us into better engineers. This project, Smart Water Level Monitoring System, is the culmination of our collective efforts, driven by our passion for technological innovation and problem-solving.

Lastly, we dedicate this project to all those who strive to make the world a better and more sustainable place through the application of smart technologies.

This work represents our commitment to contributing to a smarter future, and we hope it serves as a stepping stone for future innovations in the field of IoT and water resource management.

BONAFIDE CERTIFICATE

This is to certify that the project titled "**SMART WATER LEVEL MONITORING SYSTEM USING MICROCONTROLLER**" is a bonafide record of the work done by **Naorem Diyabarta Singh** (2101CS0216), **Ricky Thoudam** (2101CS0203) & **Naoroibam Lanngamba Luwang** (2101CS0209) during the academic year 2024-25, in partial fulfilment for the award of the degree of Bachelor of Technology in Computer Science and Engineering. This is a bonafide record of the 7th semester project carried out by them under my supervision. The contents of this report, in full or in parts, have not been submitted to any other institution or university of the award of any degree or diploma.

Date: January 21, 2025

Place: Imphal

Head of Department

Dr. Hanjabam Saratchandra Sharma

Assistant Professor, CSE, MTU

Imphal, Manipur – 795004

DECLARATION

I declare that the thesis entitled "**SMART WATER LEVEL MONITORING SYSTEM USING MICROCONTROLLER**" submitted in partial fulfillment for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** is a record of original work carried out by me under the guidance of **Dr. Hanjabam Saratchandra Sharma**, and has not formed the basis for the award of any other degree or diploma in this or any other institution or university. In keeping with the typical practice in reporting scientific information, due acknowledgements have been made of whatever findings of others have been cited.

Date: January 21, 2025

Place: Imphal

Naorem Diyabarta Singh

Reg. No. 2101CS0216

Dept. of Computer Science and
Engineering

DECLARATION

I declare that the thesis entitled "**SMART WATER LEVEL MONITORING SYSTEM USING MICROCONTROLLER**" submitted in partial fulfillment for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** is a record of original work carried out by me under the guidance of **Dr. Hanjabam Saratchandra Sharma**, and has not formed the basis for the award of any other degree or diploma in this or any other institution or university. In keeping with the typical practice in reporting scientific information, due acknowledgements have been made of whatever findings of others have been cited.

Date: January 21, 2025

Place: Imphal

Ricky Thoudam

Reg. No. 2101CS0203

Dept. of Computer Science and
Engineering

DECLARATION

I declare that the thesis entitled "**SMART WATER LEVEL MONITORING SYSTEM USING MICROCONTROLLER**" submitted in partial fulfillment for the award of the degree of **Bachelor of Technology in Computer Science and Engineering** is a record of original work carried out by me under the guidance of **Dr. Hanjabam Saratchandra Sharma**, and has not formed the basis for the award of any other degree or diploma in this or any other institution or university. In keeping with the typical practice in reporting scientific information, due acknowledgements have been made of whatever findings of others have been cited.

Date: January 21, 2025

Place: Imphal

Naoroibam Lanngamba Luwang

Reg. No. 2101CS0209

Dept. of Computer Science and
Engineering

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our **Head of Department and Guide, Dr. Hanjabam Saratchandra Sharma**, for his invaluable guidance and support throughout the course of this project. His expertise, encouragement, and constructive feedback were instrumental in shaping our understanding of IoT technology and its applications.

His unwavering commitment to our success and his dedication to fostering a collaborative learning environment have greatly inspired us. We are especially thankful for the time he devoted to mentoring us, answering our queries, and providing insightful suggestions that enhanced our project significantly. His ability to simplify complex concepts allowed us to grasp the intricacies of the subject matter and apply them effectively.

We would also like to acknowledge the valuable resources and facilities that enabled us to conduct our research and development efficiently. The access to advanced tools and technologies played a crucial role in the successful execution of our project, allowing us to implement our ideas with precision and confidence.

Finally, we express our gratitude to all individuals who participated in the testing and evaluation of the system. Their valuable feedback helped us refine the system and ensure its functionality and user-friendliness.

Date: January 21, 2025

Place: Imphal

Sincerly,
Naorem Diyabarta Singh
Ricky Thoudam
Naoroibam Lanngamba Luwang

ABSTRACT

This project presents the development of a Smart Water Level Monitoring System using Internet of Things (IoT) technology to address the global challenges of water scarcity and resource management. The system provides real-time water level monitoring in tanks, reservoirs, or other storage facilities, ensuring efficient water usage. It employs an ultrasonic sensor to measure water levels, which are processed by a microcontroller and displayed on a 0.96" OLED screen for visual representation. Additionally, the system integrates Blynk technology, allowing users to access real-time information on their mobile devices.

Key electronic components such as 1k resistors, BC547 NPN transistors, LEDs, push buttons, and a 5V DC buzzer enable various functionalities, including current limiting, signal amplification, visual indicators, manual control, and auditory alerts for critical water levels. An optional AC to DC converter ensures stable power supply for continuous operation. The system's push buttons offer customizable control, and the combination of visual and auditory cues helps prevent overflow and alert users when water levels fall below a critical threshold, enabling proactive water management.

Testing results demonstrate the system's effectiveness in detecting high and low water levels and providing timely alerts to users. Its real-time monitoring and alerting capabilities reduce the risk of water shortages or overflows, contributing to sustainable water management. This cost-effective solution is adaptable to a variety of applications, from small household water tanks to large industrial reservoirs, making it particularly useful in regions facing water scarcity. The system's autonomous operation, low maintenance, and scalability further enhance its utility in residential, agricultural, and industrial sectors.

Keywords: IoT, Smart Water Monitoring, Ultrasonic Sensor, Blynk Technology, Water Resource Management.

Contents

BONAFIDE CERTIFICATE	ii
DECLARATION	iii
ACKNOWLEDGEMENT	vi
ABSTRACT	vii
List of Figures	x
1 INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Objectives of the study	3
2 LITERATURE REVIEW	5
2.1 Overview of Water Level Monitoring Systems	5
2.1.1 Traditional Water Level Monitoring Methods	5
2.1.2 Modern Water Level Monitoring Systems	5
2.1.3 Key Components of a Modern Water Level Monitoring System	6
2.1.4 Applications of Water Level Monitoring Systems	6
2.1.5 Challenges and Future Trends	7
2.2 Existing Technologies and Methods	7
3 PROPOSED SYSTEM	9
3.1 System Design Overview	9
3.2 Hardware Components Used	11
4 SYSTEM IMPLEMENTATION	16
4.1 Hardware Assembly and Circuit Diagram	16
4.2 Software Development and Integration with Arduino IDE and Blynk	18
4.2.1 Arduino IDE Setup and Programming	19
4.2.2 Blynk Cloud Integration	20

5	CONCLUSION	22
5.1	Summary of the Project	22
5.2	Recommendation for Future Research	22
	REFERENCES	25
	APPENDICES	26

List of Figures

3.1	Ultrasonic Sensor (HC-SR04)	11
3.2	NodeMCU/ ESP8266	11
3.3	0.96"OLED Display	12
3.4	Buzzer	12
3.5	Push Button	12
3.6	Jumper Wires	13
3.7	Single channel relay	13
3.8	Breadboard	14
3.9	Resistors	14
3.10	NPN-Transistor	14
4.1	Water Level Monitoring and Control Circuit Diagram	16

Chapter 1

INTRODUCTION

1.1 Background

Water is one of our planet's most vital resources, essential for both ecosystems and human survival. As global demand for clean and safe water rises, managing this precious resource effectively has become a major concern. Many regions face challenges like unmonitored water wastage, inefficient management, and limited access to clean water. With urbanization and population growth, ensuring sustainable water resources is increasingly difficult.

Traditional methods of monitoring water levels, such as manual measurements, are often inefficient and prone to human error. These methods typically fail to provide real-time data, making effective water management challenging. This is particularly problematic in areas where water scarcity is common or where water bodies are vulnerable to sudden changes due to weather or other environmental factors.

The rise of the Internet of Things (IoT) and smart technology offers new solutions to these challenges. A Smart Water Level Monitoring System uses advanced sensors and communication technologies to monitor and manage water levels in real-time. These systems automate data collection, reduce the need for human intervention, and enable remote monitoring, leading to more efficient water management.

By integrating sensors with wireless communication technologies and cloud-based platforms, smart water level monitoring systems provide real-time data on water levels. They can trigger alerts for abnormal water levels, predict future trends based on historical data, and even integrate with mobile or web-based applications for easy user access. This level of automation and intelligence greatly enhances

the ability of individuals, communities, and industries to conserve water, prevent overflow or underflow, and optimize water use.

In this context, the Smart Water Level Monitoring System aims to offer a cost-effective, reliable, and scalable solution for monitoring water levels in various applications, including residential water tanks, irrigation systems, flood monitoring, and water reservoirs. With real-time data analytics and control mechanisms, this system can significantly contribute to water conservation efforts, ensuring efficient management of the world's water resources.

This thesis explores the development, implementation, and testing of a Smart Water Level Monitoring System, designed to provide accurate, real-time data for effective water resource management.

1.2 Problem Statement

In many regions, water management is a critical issue due to the increasing demand and limited availability of water resources. Inefficient water usage, wastage, and overflow of storage tanks are common problems that affect residential, industrial, and agricultural sectors. Traditional methods of monitoring water levels often rely on manual checks, which are time-consuming, prone to human error, and unable to provide real-time data. This can lead to either excessive water consumption or water shortages, resulting in wastage or inadequate supply when most needed.

A Smart Water Level Monitoring System aims to address these challenges by providing an automated, efficient solution. Using a microcontroller-based system, sensors are employed to measure water levels in tanks or reservoirs. The system continuously monitors the water level and can trigger alerts when predefined thresholds, such as low or high levels, are reached. This real-time data can be displayed on digital dashboards or sent remotely to users via smartphones or web platforms, enabling timely action.

The system can also be integrated with water pumps for automatic control. When the water level drops below a certain limit, the pump can be activated to refill the tank, and when the tank reaches a specified upper limit, the pump is turned off, preventing overflow. This automation reduces the need for manual intervention, minimizes water wastage, and ensures a steady water supply.

Such systems are particularly useful in places where water is a scarce resource or where large volumes need to be managed efficiently, such as in agricultural irri-

gation, industrial processes, or household water storage. Additionally, the system can help conserve energy by optimizing pump usage, making it both an environmentally friendly and cost-effective solution.

1.3 Objectives of the study

The Smart Water Level Monitoring System is a highly innovative project designed to use Internet of Things (IoT) technology to better manage water quality. Its main objective is to address the increasing challenges related to water management and natural resource management. They provide solutions that are not only practical. But it's also scalable and cost-effective. The project focuses on developing a real-time water level monitoring system that can provide live information on water levels. Automate the water inspection process and reduce human intervention. The main objective of this project is real-time monitoring, automatic system, water conservation, remote access and reducing costs.

One of the main objectives of the system is real-time monitoring. Traditional water monitoring methods usually involve manual sampling or basic mechanical drainage systems. This is time consuming and prone to errors. The smart water level monitoring system comes with IoT-based sensors that provide continuous updates on the water level in a reservoir such as a reservoir or reservoir. It gives operators access to accurate and up-to-date information about water storage conditions. This allows them to respond quickly to changes. For example, operators will be alerted when the water level drops significantly or the reservoir is close to overflowing. Makes it possible to take action in a timely manner The rapid availability of this information greatly improves efficiency in water management systems. This is especially true in situations where timely action is important, such as in agricultural or municipal water systems.

The project also aims to achieve automation. This is the key to reducing manual labor and increasing efficiency. The system can automatically turn the water pump on or off based on the water level readings. If the water level drops below a certain point, the system will command the pump to fill the tank. Likewise, when the tank is full, the system stops the pump from overflowing. This level of automation not only makes the operation of the hydraulic system easier. But it also ensures that the system can function without constant human supervision. Automation helps prevent human errors, such as forgetting to turn off the pump. This may lead to overflow, wasting water and unnecessary use of electricity.

Another big target is water. The importance of water conservation cannot be

overstated. Especially in water shortage areas or drought-prone areas. Smart water level monitoring systems help in conserving water by ensuring the water is functioning properly. Minimize waste by preventing overflow. Reduce unnecessary work with an automatic pump and allows operators to track water use over time. This is especially useful in urban areas where water quality must be strictly controlled or in industries that store and use large amounts of water. By providing information on consumption and usage the system can help identify inefficiencies and identify improvements and leads to better resource management.

Remote access is another important objective of this project. One of the key benefits of using IoT technology is that it allows users to remotely monitor and control the system. With an intelligent water level monitoring system, user can check water level, receive notifications and even monitor the pump from the mobile app or online. This technology is especially useful for those who manage water systems in multiple locations, such as farmers with remote water systems or homeowners who want to monitor their water source remotely. Remote access allows users to stay connected to their system at any time regardless of physical location, provide comfort and peace of mind.

Ultimately, cost-effectiveness is the main objective of the project. Smart water level monitoring systems are designed to be practical using cost-effective sensors and energy-efficient components. This makes it a good choice for both residential and commercial applications. Automated irrigation systems reduce labor costs and by conserving water Water costs can be significantly reduced. Additionally, reduced energy consumption due to automatic pump control results in long-term cost savings.

In summary, smart water level control technology offers a comprehensive solution to modern water management challenges. through real-time monitoring, automation, water conservation, remote access and cost efficiency, the project aims to transform how water resources are managed, ensuring sustainability and convenience for users across different sectors.

Chapter 2

LITERATURE REVIEW

2.1 Overview of Water Level Monitoring Systems

Overview of Water Level Monitoring Systems Water Level Monitoring Systems (WLMS) are essential technologies designed to monitor, manage, and control the water levels in various storage units such as tanks, reservoirs, lakes, and other water bodies. These systems play a crucial role in optimizing water usage, preventing overflows, and ensuring sustainable water management in households, industries, and agricultural settings.

2.1.1 Traditional Water Level Monitoring Methods

Historically, water levels were monitored using mechanical devices like float switches and manual inspection techniques. Float-based systems rely on physical movement to indicate water levels, triggering pumps when necessary. However, these methods often require maintenance and are prone to inaccuracies due to wear and tear, as well as environmental conditions like rust or clogging.

2.1.2 Modern Water Level Monitoring Systems

Modern water level monitoring systems use advanced sensors, microcontrollers, and communication technologies to offer automated and precise monitoring. Key advancements in these systems include:

- **Sensor Technology:** Ultrasonic, capacitive, and pressure sensors are commonly used. These sensors provide non-contact, high-precision measurements, which are more reliable than traditional float systems.
- **Microcontrollers and IoT Integration:** Microcontrollers like Arduino, ESP8266, and ESP32 are used to process sensor data and communicate

with remote systems. These systems often utilize Wi-Fi or GSM modules to connect to the internet, enabling real-time data transfer to cloud platforms or mobile applications.

- **Remote Monitoring:** The integration of IoT (Internet of Things) technologies has revolutionized water level monitoring. Users can remotely monitor water levels using web dashboards or mobile apps, receive real-time alerts, and control water pumps or valves from anywhere in the world.

2.1.3 Key Components of a Modern Water Level Monitoring System

- **Sensors:** Sensors measure the water level and provide data to the microcontroller. Ultrasonic sensors are particularly popular due to their accuracy and non-contact measurement capabilities.
- **Microcontroller:** Microcontrollers process the sensor data and communicate with output devices such as displays or cloud platforms. ESP8266 and ESP32 are widely used due to their built-in Wi-Fi capabilities.
- **Display and Communication Interface:** Local displays such as OLED or LCD screens can provide real-time data on-site, while remote communication interfaces like web apps or mobile apps (e.g., Blynk) allow users to access the data remotely.
- **Power Supply:** These systems are powered either by direct AC or batteries, and some designs incorporate solar power for off-grid applications.

2.1.4 Applications of Water Level Monitoring Systems

Water level monitoring systems find applications across multiple sectors:

- **Household Water Tanks:** Automated systems ensure that water tanks are efficiently filled, preventing overflows and shortages, and automating water pump control.
- **Industrial Water Reservoirs:** In industries where water is a critical resource, these systems optimize water usage, prevent downtime due to water scarcity, and ensure operational efficiency.
- **Agricultural Irrigation Systems:** In agriculture, monitoring water levels in reservoirs or irrigation channels helps optimize water usage, preventing both over-irrigation and water shortages, contributing to sustainable farming practices.

- **Flood Monitoring:** Advanced WLMS are used in flood-prone areas to monitor river or dam levels, providing early warnings to reduce flood risks.

2.1.5 Challenges and Future Trends

While modern systems offer high precision and automation, challenges such as sensor calibration, power reliability in remote areas, and connectivity issues still exist. Future trends in water level monitoring systems focus on integrating artificial intelligence and machine learning to predict water usage patterns, automate pump control, and provide more efficient water management solutions.

Overall, Water Level Monitoring Systems have evolved from simple mechanical devices to highly automated, IoT-enabled solutions that play a critical role in water conservation, operational efficiency, and environmental sustainability.

2.2 Existing Technologies and Methods

The need for water quality has seen many technological advances in the past decade. Many available methods and techniques for water level monitoring are explored. This includes traditional methods and techniques such as mechanical cranes, ultrasonic sensors, and more complex IoT-based solutions. Each technology has its own strengths and limitations. This lays the foundation for the development of more complex technology, such as smart water level monitoring systems.

To deal with mechanical limitations, the use of ultrasonic sensors has become popular. These sensors use ultrasonic waves that bounce off the water surface and return to the sensor. It uses a time delay to measure the water level. Ultrasonic sensors are more accurate than floating mechanisms and can be integrated with a microcontroller to automate water level readings. They are widely used in industry and large water storage facilities due to their accuracy and durability. However, ultrasonic systems can be expensive and the accuracy of the system may be affected by environmental factors such as temperature fluctuations and physical obstacles.

Another commonly used method is using capacitive or conductive sensors. These sensors measure changes in capacitance or electrical conductivity as a function of the water content. For example, capacitive sensors detect water level based on the dielectric constant of the water and the proximity of the water with sensor electrode. Although these systems are highly accurate and relatively cheap, but

these systems are limited by environmental factors such as water quality, as contaminants or changes in water levels can affect readings.

The development of IoT-based water quality monitoring systems is an important step forward in this field. IoT enables the integration of sensors (such as ultrasonic, capacitive or pressure sensors) with connectivity. Internet connection Allowing for real-time monitoring and remote access to water level data, IoT technology is especially useful. Because it not only automates the water level checking. But it can also monitor the water pump remotely. Provides detailed data analysis and notify when a critical threshold is reached. IoT-based water quality monitoring systems show great potential for water management in both urban and rural areas. In general, these systems involve with a cloud-based platform that can store, analyze, and access data from anywhere through a mobile device or web-based gateway. However, these systems often require a stable internet connection and may face problems related to privacy and data security.

In summary, while traditional technologies such as mechanical cranes and ultrasonic sensors have laid the foundation for water level monitoring, the evolution of IoT-based systems is an important step forward. These advanced technologies provide real-time monitoring, remote access and data-driven insights. This makes it possible to apply it to modern water management. However, there are limitations related to cost, network connection and environmental problems still exist. This drives research and innovation in this field. Smart water quality monitoring systems are built on these advanced technologies. Its aim is to provide an efficient, scalable and cost-effective solution for water management.

Chapter 3

PROPOSED SYSTEM

3.1 System Design Overview

The smart water level monitoring system consists of sensors, microcontrollers, and IoT technology for efficient and automated water management. Below is an overview of the system design:

1. Components and Hardware Design

- **Ultrasonic Sensor (HC-SR04):** Measures the distance to the water surface and determines the water level.
- **Microcontroller (NodeMCU or Raspberry Pi):** Processes sensor data, controls pump operations, and connects to the Blynk cloud.
- **Wi-Fi Module (Built-in NodeMCU):** Allows for remote monitoring and real-time feedback from the Blynk application.
- **OLED Display (0.96-inch):** Visualizes water levels locally.
- **Buzzer:** Signals over/underflow or low level.
- **Water Pump:** Controlled automatically with preset water level thresholds.
- **Power Supply:** Stabilizes all elements to ensure continuous running of the system.

2. Functional Modules

- **Data Acquisition Module:** The ultrasonic sensor captures water level information based on the distance from the surface.
- **Processing Module:** Microcontroller processes sensor inputs, determining whether alerts are required or the pump should be switched on.

- **Communication Module:** Wi-Fi ensures that all data is uploaded to the Blynk cloud, thereby providing remote access and control.
- **Display Module:** OLED Screen shows real-time information for water level information in local access.
- **Control Module:**
 - The automated pumping is controlled based on preset thresholds to maintain a target level of water.
 - Override via the mobile application to make custom operations.

3. Software Design

- **Blynk Integration:** Real-time monitoring and control through the Blynk application, including notified alerts and notifications to the mobile.
- **Cloud Connectivity:** Data can be saved in the Blynk cloud, visualized, and insights can be derived over history. The system automatically switches the pump ON or OFF based on the sensor readings to minimize manual intervention.

4. Workflow

- The ultrasonic sensor takes real-time measurements of the water level.
- Sensor data is transmitted to the microcontroller for processing.
- The microcontroller updates the OLED display and sends data to the Blynk cloud.
- Depending on the water level:
 - If it is low, the pump is turned ON.
 - If it is full, the pump is turned OFF.
- Users are notified on their mobile devices through the Blynk app.
- Alerts are thrown if critical thresholds (overflow/low water) are exceeded.

5. Scalability and Future Enhancements

- Support for multiple tanks or reservoirs.
- Additional sensors that could be integrated for quality water monitoring, such as pH and turbidity.
- Further advanced data analysis and insights from cloud services.
- This design of the system will come in as cost-effective and scalable with reliability in providing real-time water level monitoring and resource management.

3.2 Hardware Components Used

1. Ultrasonic Sensor (HC-SR04)

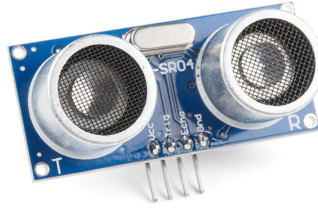


Figure 3.1: Ultrasonic Sensor (HC-SR04)

The HC-SR04 is a widely used ultrasonic sensor designed for distance measurement. It operates by emitting ultrasonic waves at a frequency of 40 kHz through its transmitter and receiving the echo reflected from an object via its receiver. The time taken for the echo to return is measured and used to calculate the distance to the object based on the speed of sound.

2. Microcontroller (NodeMCU/ESP8266)

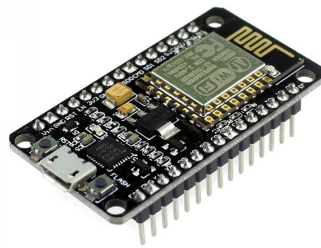


Figure 3.2: NodeMCU/ ESP8266

The ESP8266 is a low-cost Wi-Fi-enabled microcontroller that serves as the system's core processing unit. It processes data from sensors, executes control logic, and facilitates wireless communication with cloud platforms and mobile devices. Its built-in Wi-Fi module makes it ideal for IoT applications.

3. 0.96" OLED Display



Figure 3.3: 0.96"OLED Display

A 0.96-inch OLED display is a compact, high-contrast, and energy-efficient screen used in IoT projects to visually present data such as sensor readings, system status, or notifications.

4. **Buzzer**



Figure 3.4: Buzzer

An electronic signaling device that produces sound. Commonly used in alarms, timers, and notifications in electronic circuits.

5. **Push Buttons**



Figure 3.5: Push Button

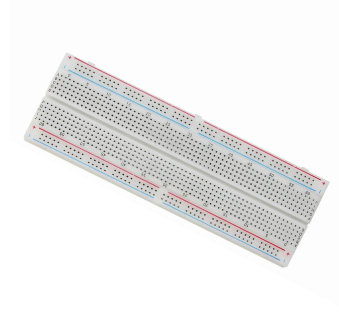


Figure 3.8: Breadboard

Prototyping tool for electronics projects, allowing components to be easily connected and tested without soldering. Ideal for rapid prototyping and experimenting with circuit designs.

9. Resistors

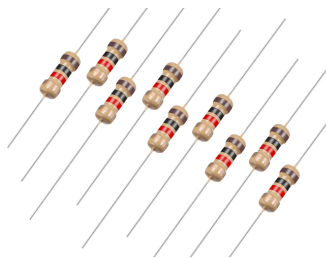


Figure 3.9: Resistors

A resistor is a passive electronic component used in IoT devices to limit current, divide voltages, or protect other components from excessive current flow. Resistors are critical for ensuring the safe and efficient operation of IoT circuits.

10. NPN Transistors



Figure 3.10: NPN-Transistor

An NPN transistor is commonly used as a switch or amplifier for controlling electronic signals in circuits. The NPN transistor consists of three layers of semiconductor material: n-type (negative), p-type (positive), and another n-type (negative), which are arranged as Emitter (E), Base (B), and Collector (C).

Chapter 4

SYSTEM IMPLEMENTATION

4.1 Hardware Assembly and Circuit Diagram

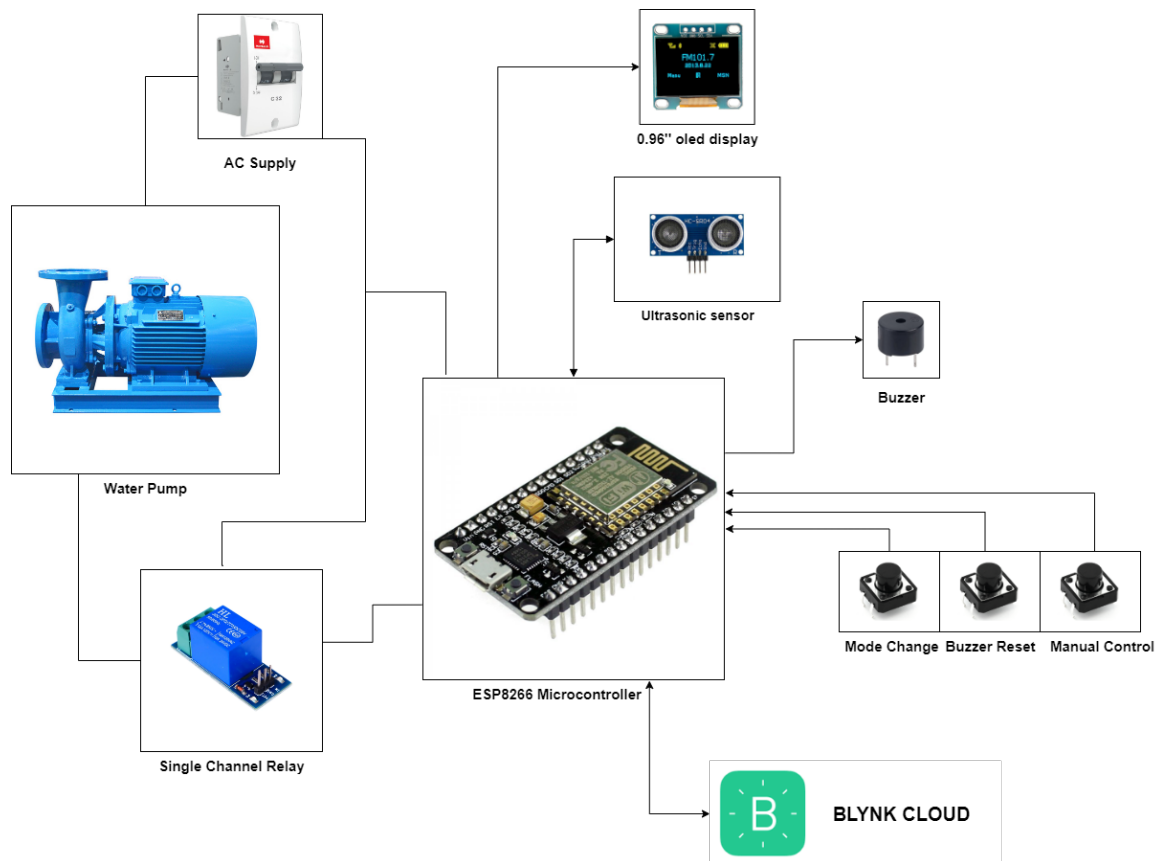


Figure 4.1: Water Level Monitoring and Control Circuit Diagram

The hardware setup of this IoT-based water level monitoring and pump control system integrates multiple components to ensure real-time data acquisition, remote monitoring, and automated control of the water pump.

The following components are used in the system, with each playing a crucial role:

1. **ESP8266 Microcontroller**

The ESP8266 is the core of the system, acting as the main controller for processing data from sensors, controlling output devices, and managing communication with the Blynk cloud platform. Its built-in Wi-Fi module allows seamless remote monitoring and control via the Blynk mobile application.

2. **Ultrasonic Sensor (HC-SR04)**

The HC-SR04 ultrasonic sensor measures the distance between itself and the water surface, determining the water level in the tank. It sends high-frequency sound waves and calculates the time taken for the echo to return, which the ESP8266 processes to derive the water level.

3. **0.96" OLED Display**

The OLED screen provides a local, real-time display of the water level. It serves as an immediate visual representation of the tank's status for users near the system.

4. **Single Channel Relay Module**

The relay acts as a switch to control the water pump. Based on the water level thresholds processed by the ESP8266, the relay automatically activates or deactivates the water pump.

5. **Water Pump**

The water pump fills the tank when the water level is low. Its operation is automated by the ESP8266 via the relay module.

6. **Push Buttons**

Three push buttons provide manual control and additional functionality:

- **Mode Change Button:** Toggles between different operational modes, such as automatic and manual control.
- **Buzzer Reset Button:** Silences the buzzer when a critical water level alert is triggered.
- **Manual Control Button:** Allows the user to turn the pump on or off manually.

7. **Buzzer**

The buzzer provides an audible alert for critical conditions, such as the tank being nearly empty or full. It ensures users are immediately aware of the system's status.

8. AC Power Supply

The water pump is powered by an AC power supply, with the system ensuring safe and efficient operation.

Circuit Diagram Explanation

The circuit diagram visually represents the interconnections between the various hardware components. Each component is linked to the ESP8266, forming a centralized control system. The HC-SR04 ultrasonic sensor is connected to the ESP8266 via its trigger and echo pins. These signals are processed to determine the water level in the tank. The OLED display communicates with the ESP8266 through an I2C interface to show real-time water level data. The relay module connects to the ESP8266's GPIO pins to control the water pump based on the processed data. The push buttons are connected to the GPIO pins of the ESP8266, enabling manual user input for operational adjustments. The buzzer is connected to another GPIO pin and is activated for critical alerts. The ESP8266 sends data to the Blynk cloud platform via its Wi-Fi module, enabling remote monitoring and control through a smartphone app. **System Functionality** The ultrasonic sensor continuously monitors the water level and sends data to the ESP8266 for processing. Based on predefined thresholds, the ESP8266 decides whether to turn the water pump on or off. Data is displayed locally on the OLED screen and transmitted to the Blynk cloud for remote monitoring. The push buttons allow users to override the system for manual control, reset alarms, or switch between modes. Alerts are triggered via the buzzer and mobile notifications for low or critical water levels. This circuit diagram reflects the modular and scalable nature of the system, making it suitable for future enhancements, such as additional sensors or water quality monitoring modules.

4.2 Software Development and Integration with Arduino IDE and Blynk

The software development process forms the backbone of the water monitoring and control system, enabling seamless communication between hardware components, real-time monitoring, and remote control functionality. The system leverages the Arduino IDE for microcontroller programming and the Blynk platform for IoT integration, creating a robust and user-friendly solution. This section provides a detailed explanation of the software design and implementation, highlighting its key components and functionalities.

4.2.1 Arduino IDE Setup and Programming

The Arduino IDE serves as the primary environment for programming the ESP8266 microcontroller. It provides tools for writing, debugging, and uploading code, ensuring efficient interaction with the system's hardware components.

1. Environment Setup:

- The Arduino IDE is configured to support the ESP8266 microcontroller by adding the ESP8266 board manager through the IDE preferences. This allows seamless compatibility between the IDE and the hardware.
- Relevant drivers are installed to ensure the microcontroller is recognized by the computer.

2. Library Installation:

- Essential libraries are integrated into the IDE to enable specific functionalities:
 - Blynk Library: Facilitates communication between the ESP8266 and the Blynk cloud platform.
 - Adafruit GFX and SSD1306 Libraries: Support the OLED display module for real-time data visualization.
 - Ultrasonic Sensor Library: Simplifies the interaction with the HC-SR04 ultrasonic sensor.
- These libraries are added through the library manager or manually imported into the project.

3. Code Development: The software is designed to handle multiple tasks simultaneously, ensuring the system operates efficiently. Key code components include:

• Sensor Data Acquisition:

The HC-SR04 ultrasonic sensor measures the water level by calculating the distance between the sensor and the water surface. The sensor's echo and trigger pins are defined, and the data is read using precise timing functions.

• Data Processing:

The raw sensor data is processed into meaningful water level values, which are then used to determine system actions. For instance, threshold levels are set to activate or deactivate the water pump.

- **OLED Display Control:**

The processed water level data is sent to the OLED display for local visualization. The display updates in real-time to provide an intuitive interface for users.

- **Buzzer Alerts:**

The buzzer is activated when critical water levels are reached, providing an audible alert to ensure timely action.

- **Automation Logic:**

Control logic is implemented to automatically switch the water pump on or off based on water level thresholds. This reduces the need for manual intervention.

4. **Testing and Debugging:** The Arduino IDE's serial monitor is utilized extensively during development to monitor real-time data from the microcontroller. Debugging tools within the IDE help identify and resolve any issues in the code, ensuring smooth operation.

4.2.2 Blynk Cloud Integration

Blynk is employed as the IoT platform to enable real-time monitoring and remote control through a mobile application. It acts as a bridge between the ESP8266 microcontroller and the user, providing an intuitive interface for system management.

1. **Cloud Platform Configuration:**

A Blynk account is created, and a new project is set up within the mobile application. An authentication token generated by Blynk is included in the Arduino code, linking the microcontroller to the cloud platform.

2. **Wi-Fi Connectivity:**

The ESP8266 is programmed to connect to a Wi-Fi network using the SSID and password provided in the code. This connection enables the microcontroller to send and receive data via the Blynk cloud.

3. **Mobile Application Widgets:**

The Blynk mobile application is configured with various widgets to enhance user interaction:

- **Value Display:** Displays the current water level in real-time.
- **Button Widget:** Allows manual control of the water pump.
- **Gauge Widget:** Provides a graphical representation of the water level.

- Notification Widget: Sends alerts to the user's mobile device for critical events, such as low or high water levels.

4. **Data Transmission:**

- The processed water level data from the ultrasonic sensor is transmitted to the Blynk cloud via virtual pins.
- The Blynk platform relays this data to the mobile application, where users can monitor it in real-time.

5. **Control Mechanisms:**

- Users can control the water pump remotely through the Blynk app.
- Manual overrides are supported, allowing users to activate or deactivate the pump as needed, irrespective of the automation logic.

6. **Automation Features:**

The Blynk platform integrates seamlessly with the automation logic in the Arduino code. For example, when a critical water level is reached, the system sends a notification to the user while simultaneously triggering the appropriate system response.

By combining Arduino IDE and Blynk, the system benefits from a simple yet powerful development and integration process. The Arduino IDE provides flexibility and control over the microcontroller, while Blynk offers user-friendly tools for IoT integration. Together, they create a system that is reliable, responsive, and accessible.

This software setup allows for real-time monitoring, efficient automation, and intuitive remote control, making it a vital component of the water monitoring and control system.

Chapter 5

CONCLUSION

5.1 Summary of the Project

The Smart Water Level Monitoring System using a microcontroller successfully achieved its objectives of monitoring and managing water levels in real-time. The system efficiently utilizes ultrasonic sensors to measure the water level, calculates the percentage of tank capacity, and provides clear visual feedback via an OLED display and the Blynk platform. Additionally, the system includes a buzzer alert to notify users when the tank is almost full, preventing water wastage due to overflow. This solution demonstrates the potential of integrating IoT and embedded systems to automate and improve water management in households, industries, and agriculture. The project is a cost-effective, user-friendly, and reliable solution for addressing water monitoring challenges.

5.2 Recommendation for Future Research

1. Integration with Automated Pump Systems

Future iterations of the Smart Water Level Monitoring System could incorporate an automated pump control mechanism to enhance its functionality and user convenience. By integrating a relay module or a similar device, the system could automatically start or stop the water pump based on predefined water levels, eliminating the need for manual intervention. For instance, the pump could automatically turn on when the water level falls below a certain threshold and turn off when the tank is almost full. This automation not only reduces the risk of overflow or dry runs but also improves energy efficiency and water resource management. Furthermore, such an enhancement would be especially beneficial for households, industries, and agricultural setups where timely water replenishment is critical.

2. Ultrasonic Waterproof Sensors

The integration of ultrasonic waterproof sensors, such as the widely used JSN-SR04T, can significantly enhance the accuracy and reliability of the water level monitoring system. These sensors operate by emitting ultrasonic sound waves and measuring the time taken for the waves to return after bouncing off the water's surface. This non-contact method ensures precise measurements without any wear and tear caused by direct exposure to water. Additionally, their waterproof design allows them to function effectively in submerged or high-humidity environments, making them ideal for water tanks, reservoirs, and other applications. The JSN-SR04T, in particular, is known for its durability, affordability, and resistance to harsh environmental conditions, ensuring long-term reliability and low maintenance. Adopting such sensors ensures that the system remains robust and accurate, even under challenging operational conditions.

3. Solar Power Integration

Integrating solar power into the water level monitoring system offers a sustainable and energy-efficient solution, especially for remote areas with limited or unreliable electricity access. Solar panels can provide the necessary energy to power the sensors, microcontroller, and additional components, making the system self-sufficient and reducing dependency on external power sources. This approach not only lowers operational costs but also aligns with environmental sustainability goals by utilizing renewable energy. In addition, a battery backup can be added to store excess solar energy, ensuring uninterrupted operation during nighttime or cloudy weather. The adoption of solar power not only broadens the applicability of the system but also enhances its resilience in diverse environments, including rural and off-grid areas.





4. Support for Multiple Tanks

To increase the versatility and scalability of the water level monitoring system, future upgrades could include the ability to monitor multiple tanks simultaneously. This feature would be particularly useful for large-scale applications, such as in residential complexes, industrial plants, or agricultural irrigation systems, where multiple water storage units are used. By incorporating multiple sensor inputs and designing a centralized control interface, users could monitor and manage the water levels of all tanks from a single platform, such as the Blynk app or a web dashboard. Additionally, the system could incorporate separate alerts and automated controls for each tank, further streamlining water management. Such a multi-tank monitoring system would provide a comprehensive solution for efficient water usage and conservation across different sectors.

5. Leakage Detection

Adding a leakage detection feature to the water level monitoring system would provide an additional layer of functionality and ensure better water conservation. This could be achieved by integrating flow sensors or pressure transducers capable of detecting abnormal flow rates or pressure drops, which are indicative of leaks. When a potential leakage is identified, the system could send real-time alerts to the user via the Blynk platform or trigger an automatic shutoff mechanism to prevent water wastage. Such a feature would be invaluable in both residential and industrial setups, where undetected leaks could lead to significant water loss and increased utility costs. By implementing this upgrade, the system would not only monitor water levels but also actively contribute to minimizing water wastage, aligning with global efforts toward sustainable water resource management.

REFERENCES

1. Arduino Documentation,
 <https://docs.arduino.cc/>
2. Blynk Documentation
 <https://docs.blynk.io/en>
3. Wireless Communications With RF-Based Energy Harvesting: From Information Theory to Green Systems, *IEEE Xplore*,
 <https://ieeexplore.ieee.org/document/>
4. Smart Water Level Monitoring System Using Internet of Things (IoT), *ResearchGate*,
 <https://www.researchgate.net/publication/>

APPENDICES

APPENDIX A : Source Code

```
1 #define BLYNK_TEMPLATE_ID "TMPL3IB6tWBiU"
2 #define BLYNK_TEMPLATE_NAME "Esp8266 Water Level"
3 #define BLYNK_AUTH_TOKEN "4fqrL-V4lYbfVKfVsgYarGmM0JtSCkmy"
4
5 // Your WiFi credentials
6 char ssid[] = "Your_WiFi_Network"; // WiFi Name
7 char pass[] = "Your_password"; // WiFi Password
8
9 #include <BlynkSimpleEsp8266.h>
10 #include <SPI.h>
11 #include <Wire.h>
12 #include <Adafruit_GFX.h>
13 #include <Adafruit_SH110X.h>
14 #define BLYNK_PRINT Serial
15 #include <ESP8266WiFi.h>
16
17 // Define pins for ESP8266
18 #define D4 2 // GPIO2
19 #define D5 14 // GPIO14
20 #define D6 12 // GPIO12
21
22 #define i2c_Address 0x3c // Initialize with the I2C addr 0x3C
23 #define SCREEN_WIDTH 128 // OLED display width, in pixels
24 #define SCREEN_HEIGHT 64 // OLED display height, in pixels
25 #define OLED_RESET -1 // QT-PY / XIAO
26
27 Adafruit_SH1106G display = Adafruit_SH1106G(SCREEN_WIDTH,
28     SCREEN_HEIGHT, &Wire, OLED_RESET);
29
30 const int trigPin = D5;
31 const int echoPin = D6;
32 const int buzzer = D4;
33
34 const float tankHeight = 15; // Height of the water tank in cm
```

```

34
35 float duration, distance, waterLevel;
36
37 bool buzzerEnabled = true; // Flag to enable/disable the buzzer
38
39 // Timer for sending data to Blynk
40 BlynkTimer timer;
41
42 void setup() {
43     // Debug console
44     Serial.begin(9600);
45
46     // Blynk setup
47     Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
48
49     // Ultrasonic sensor
50     pinMode(trigPin, OUTPUT);
51     pinMode(echoPin, INPUT);
52     pinMode(buzzer, OUTPUT);
53
54     // OLED setup
55     delay(250); // Wait for the OLED to power up
56     display.begin(i2c_Address, true);
57     display.display(); // Show Adafruit splash screen
58     display.clearDisplay();
59
60     // Timer to send data to Blynk
61     timer.setInterval(1000L, sendToBlynk); // Send data every second
62 }
63
64 void loop() {
65     Blynk.run(); // Run Blynk
66     timer.run(); // Run the timer
67
68     // Measure distance
69     digitalWrite(trigPin, LOW);
70     delayMicroseconds(2);
71     digitalWrite(trigPin, HIGH);
72     delayMicroseconds(10);
73     digitalWrite(trigPin, LOW);
74
75     duration = pulseIn(echoPin, HIGH);
76     distance = (duration * 0.0343) / 2;
77
78     // Calculate water level
79     waterLevel = ((tankHeight - distance) / tankHeight) * 100;
80     waterLevel = constrain(waterLevel, 0, 100); // Constrain value to

```

```

0-100%
81
82 // Display water level on OLED
83 String levelStr = "Water Level: " + String(waterLevel, 1) + "%";
84 displayText(levelStr.c_str(), 0, 2, 2);
85
86 // Buzzer logic
87 if (buzzerEnabled && distance <= 5) {
88     tone(buzzer, 1000); // Sound buzzer at 1kHz
89 } else {
90     noTone(buzzer);      // Turn off the buzzer
91 }
92
93 delay(500); // Update every half second
94 }
95
96 /**
97  * Function to send data to Blynk
98  */
99 void sendToBlynk() {
100     Blynk.virtualWrite(V1, waterLevel); // Send water level percentage
        to Virtual Pin V1
101     Blynk.virtualWrite(V2, distance); // Send distance to Virtual Pin
        V2 (optional)
102 }
103
104 /**
105  * Function to display text on the OLED screen
106  * @param text The string to display
107  * @param x The x-coordinate for the starting position
108  * @param y The y-coordinate for the starting position
109  * @param textSize The size of the text (1 = small, 2 = medium, etc.)
110  */
111 void displayText(const char *text, int x, int y, int textSize) {
112     display.clearDisplay(); // Clear the display buffer
113     display.setTextSize(textSize); // Set the text size
114     display.setTextColor(SH110X_WHITE); // Set text color
115     display.setCursor(x, y); // Set cursor position
116     display.println(text); // Print the text
117     display.display(); // Display the updated buffer
118 }
119
120 /**
121  * Function to handle Blynk virtual pin V5 for enabling/disabling the
        buzzer
122  */
123 BLYNK_WRITE(V5) {

```



```
124  int value = param.asInt(); // Get the value from the Blynk app (0
    or 1)
125  buzzerEnabled = (value == 1); // Enable buzzer if 1, disable if 0
126  Serial.print("Buzzer Enabled: ");
127  Serial.println(buzzerEnabled ? "Yes" : "No");
128 }
```

APPENDIX B : Calibration and Setup Guide

1. Hardware Setup:

- **Components Required:**

- ESP8266 (e.g., NodeMCU)
- Ultrasonic sensor (HC-SR04)
- OLED display (SH1106)
- Buzzer
- Jumper wires
- Power supply (USB or battery)
- Water tank for testing

- **Connections:**

- **Ultrasonic Sensor (HC-SR04):**

- * Trig pin → GPIO14 (D5)
- * Echo pin → GPIO12 (D6)
- * VCC → 5V
- * GND → GND

- **OLED Display (SH1106):**

- * SDA → D2 (I2C Data)
- * SCL → D1 (I2C Clock)
- * VCC → 3.3V
- * GND → GND

- **Buzzer:**

- * Positive terminal → GPIO2 (D4)
- * Negative terminal → GND

- **Powering the Circuit:**

- Ensure the ESP8266 is powered via USB or a 3.3V battery source.
- Verify all connections before powering on.

2. Software Configuration:

- **Install Arduino IDE:**

- Download and install the Arduino IDE from <https://www.arduino.cc/>.

- **Install Necessary Libraries:**

- (a) Open the Arduino IDE.
- (b) Navigate to **Sketch > Include Library > Manage Libraries**.
- (c) Search for and install:
 - Blynk
 - Adafruit GFX Library
 - Adafruit SH110X
- Set Up Blynk:
 - (a) Create a new project in the Blynk app.
 - (b) Select the device as **ESP8266** and connection type as **WiFi**.
 - (c) Note the Auth Token provided and replace it in the code.
- Configure WiFi Credentials:
 - Replace the placeholders for SSID and password in the code:


```
char ssid[] = "YourWiFiName";
char pass[] = "YourWiFiPassword";
```
- Upload the Code:
 - (a) Select the board as **NodeMCU 1.0** in **Tools > Board**.
 - (b) Select the correct COM port.
 - (c) Click on the upload button to flash the code onto the ESP8266.

3. Calibration:

- Tank Height:
 - Measure the height of the water tank in centimeters.
 - Replace the value of `tankHeight` in the code with the measured height.
- Testing the Ultrasonic Sensor:
 - Position the sensor at the top of the water tank.
 - Measure the distance manually using a ruler and compare it with the sensor's output.
 - Adjust placement for accurate readings.

4. Testing and Validation:

- OLED Display:
 - Ensure the display shows the water level in percentage.
 - Adjust the screen connections if it remains blank.

- Buzzer:
 - Test the buzzer functionality by lowering the water level to trigger the threshold (`distance <= 5 cm`).
 - Enable/disable the buzzer via the Blynk app by toggling Virtual Pin V5.
- Blynk Dashboard:
 - Add widgets in the Blynk app:
 - * Label for Virtual Pin V1 (Water Level Percentage).
 - * Label for Virtual Pin V2 (Distance in cm).
 - * Button for Virtual Pin V5 (Buzzer control).
 - Validate data updates in real-time.

5. Troubleshooting:

- No Display on OLED:
 - Verify the I2C address (`0x3c`) and connections.
 - Ensure the `Adafruit_SH1106` library is properly installed.
- Buzzer Not Working:
 - Check GPIO2 (D4) connections.
 - Ensure `buzzerEnabled` is set to `true`.
- Blynk Not Connecting:
 - Confirm WiFi credentials and internet connectivity.
 - Check the Auth Token in the code.
- Ultrasonic Sensor Issues:
 - Test the sensor separately using a simple distance measurement script.
 - Ensure clean and unobstructed sensor placement.