Report for automatic water filling System

Assignment 02

Module: IOT

Batch: KIC-HNDCSAI-Y2-221-F

Group Members:

1. M.S.SAHEER – KIC-HNDCSAI-221-F-001
2. M.F.A.M.ANAS – KIC-HNDCSAI-221-F-012
3. M.F.M.SHIMER – KIC-HNDCSAI-221-F-010

**Contents**

|  |  |
| --- | --- |
|  | Page No |
| 1. Abstraction | 03 |
| 1. Introduction | 04 |
| 1. System Overview | 05 |
| 1. System Architecture | 06 |
| 1. System Design | 07 |
| 1. Implementation | 09 |
| 1. Monitoring and Control | 12 |
| 1. Maintenance and Support | 13 |
| 1. Benefits and Impact | 14 |
| 1. Challenges and Limitations | 15 |
| 1. Timeline | 16 |
| 1. Source Code | 17 |
| 1. Conclusion | 21 |

**Abstraction**

The efficient conservation and utilisation of water is key to sustainable living. Every individual at home has the power to contribute to the sustainability of the essential natural resource.This report show cases an analysis of an Automatic Water Filling system and with Live Time Monitoring capability.It addresses the key problems User convenience, water conservation and efficiency. The Demo system is a combination of Hardware, Software and Live Time Monitoring.

The objectives of this report are to provide an overview of the system, its design and implementation, monitoring and control features, maintenance guidelines, as well as the potential benefits and limitations. Additionally, it illustrates the practical applications of this system and outlines future enhancements for further enhancement for improved performance.

Key Findings

Real-time Monitoring - Users can monitor the water level from anywhere around the world, thereby giving them control over water usage.

Sensor Technology Integration - The system uses the Ultrasonic sensor, controllers, actuators, and software algorithms to automate the process of water tank management.

Efficiency and Conservation - The system increases the efficiency by precisely monitoring the water level saving the overall cost of the user.

Challenges and Future Enhancements - It acknowledges challenges related to cost, technical aspects, and maintenance, while also proposing future enhancements to further improve the system's capabilities and integration.

Recommendations

* We can implement the system in a real life environment and check the efficiency in water management and whether it saves the cost of the user.
* Monitor the system and Maintain it so it can be used for a long time.

In conclusion, the Automatic Water Tank System with Monitoring Capability showcased in this report represents a significant step towards responsible water management. Its real-time monitoring, and user-friendly features is valuable for effective water management with reduced costs.

**Introduction**

Water scarcity and efficient water management are global issues due to increasing demand due to population and industrialization and environmental challenges. This report portrays the development and implementation of an Automatic Water Tank System with Monitoring Capability to tackle these concerns.

Background

Traditional water tank systems often lack complete automation and real-time monitoring, leading to inefficiencies in water usage and maintenance. We aim here to introduce a solution that addresses these issues.

Objectives

The primary objectives of this report are to present an analysis of

* The Automatic Water Tank System with Monitoring Capability.
* Explain its design, implementation, and functionality.
* Explore its potential applications and benefits.
* Identify challenges and propose future enhancements.

Scope

- Providing an overview of the Automatic Water Tank System.

- Detailing its design, hardware, and software components.

- Detailing the monitoring and control capabilities.

- Highlighting challenges and proposing future improvements.

The report aims to offer a comprehensive understanding of this system's potential impact on water management and sustainability.

**System Overview**

The Automatic Water Tank System is a technological solution designed to manage water storage and consumption. It consists of a combination of hardware and software components, this system is made to enhance efficiency, convenience, and sustainability in water usage.

1. Hardware Components:

* Water Tank
* Well
* Ultrasonic Sensor
* Water Taps and Pipes
* Water pump / Motor

2. Software Components

The software components of the system encompass control algorithms and user interfaces.

Control Algorithms - The control algorithm is programmed in the Node MCU and the blynk software. This considers real-time sensor data to make decisions about when and how much water to fill or release. Also, manual overriding option is available through blynk app.

User Interfaces - A user friendly Interface is made by the use of the Blynk App

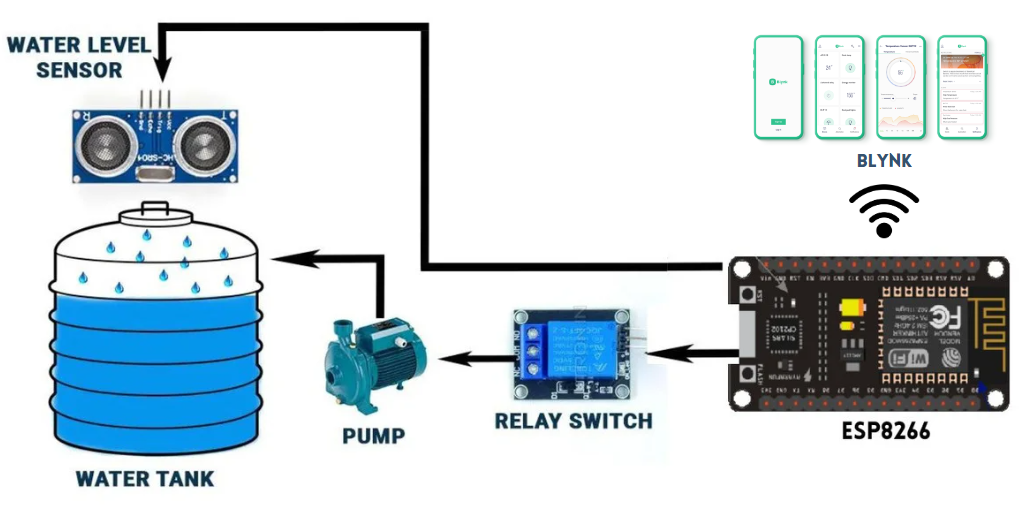
**System Architecture**

Key Features

1. Automatic Filling: The system has the capacity to automatically refill, reducing human intervention and optimising water usage.

2. Monitoring Capabilities: The real time monitoring provides continuous insights on the water level.

The integration of these hardware and software components creates a robust system that not only automates water tank operations but also offers real-time monitoring and control capabilities.

****

**System Design**

**Hardware Design**

1. Sensors



The Ultrasonic sensor is strategically positioned on the top centre of the water tank to provide accurate and continuous data regarding the water's height.

2. Microcontroller



A microcontroller is the Node MCU or ESP8266 Module.This component serves as the system's decision-making center,it analyses the sensor data and determine appropriate actions, such as regulating water flow through actuator(water pump).

3. Actuators



The Water pump operates as per the commands given by the Node MCU.

**Software Design**

1. Control Algorithms

The algorithm takes into account real-time data from the sensor(The depth of the water level). Then calculates the time it needs to fill the water based on the water flow rate and give the command to the water pump to start and stop the operation. Water filling operation starts when the water level reaches 25% of the capacity of the water tank.

2. User Interface

The Blink app acts as the User interface enabling the user to easily operate and monitor the system. Through the app user can see the current available amount of water in the tank and also manually control the water filling.

**Access Technologies & Communication Protocols**

1. Data Transmission

Data is transmitted via WiFi to the blink system. The Ultrasonic sensor uses the ESP-NOW protocol to transmit data to the Node MCU.

2. Remote Access

The User with just an internet connection can view the systems performance from anywhere around the world.

**Implementation**

In this section, we will show the detailed implementation process of the Automatic Water Tank System with Monitoring Capability, focusing on both hardware and software aspects.

**Hardware Implementation**

1. Sensor Installation

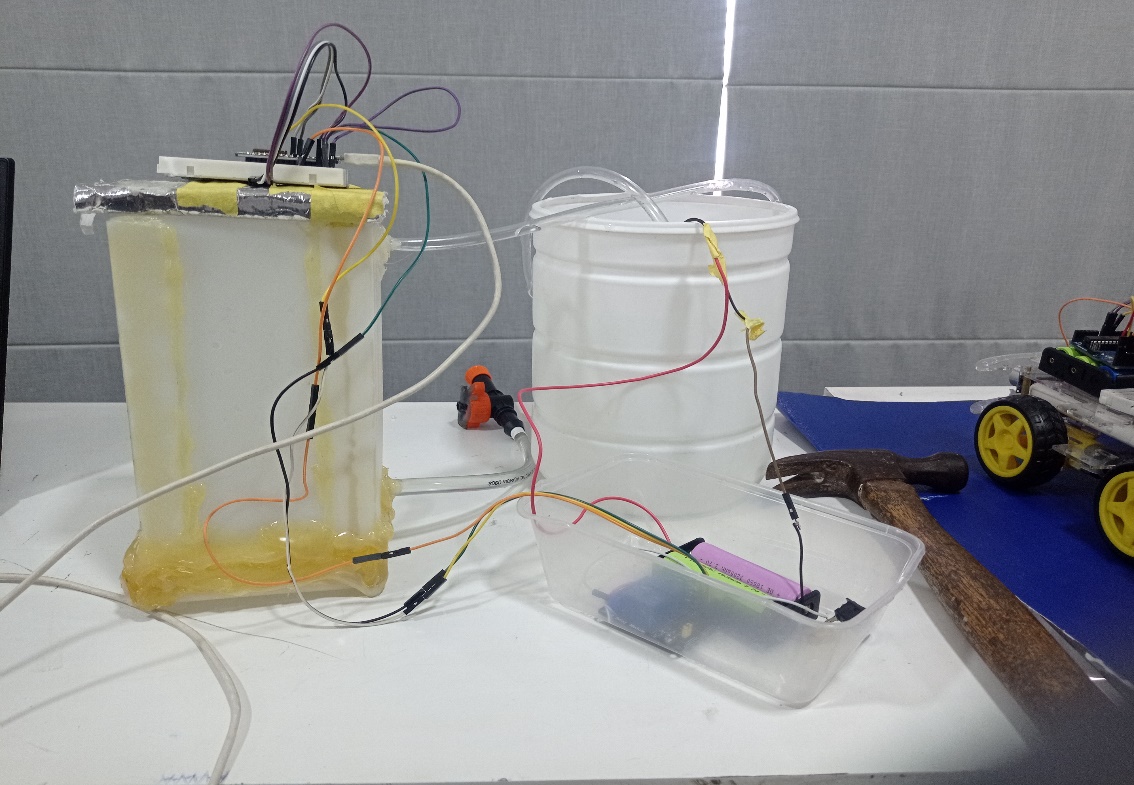
The first step in implementing our automatic water tank system was the installation of the ultrasonic sensor for water level measurement. The ultrasonic sensor was securely mounted at the top of the water tank, ensuring a stable and reliable measurement point.

2. Controller Setup

For the core control unit, we used aNode MCU, which was programmed to manage the system's operations. It was selected due to its reliability and compatibility with the sensors and actuators and its ability to transfer data via the internet. It is powered using a stable DC power source to ensure uninterrupted operation.

3. Actuator Integration

To automate the water filling process, a waterpump was integrated into the system. The water pump was connected to the Node MCU and controlled based on the water level measurements received from the ultrasonic sensor. The integration was done carefully to ensure proper functionality and prevent water wastage or overflow.



**Software Implementation**

1. Algorithm Development

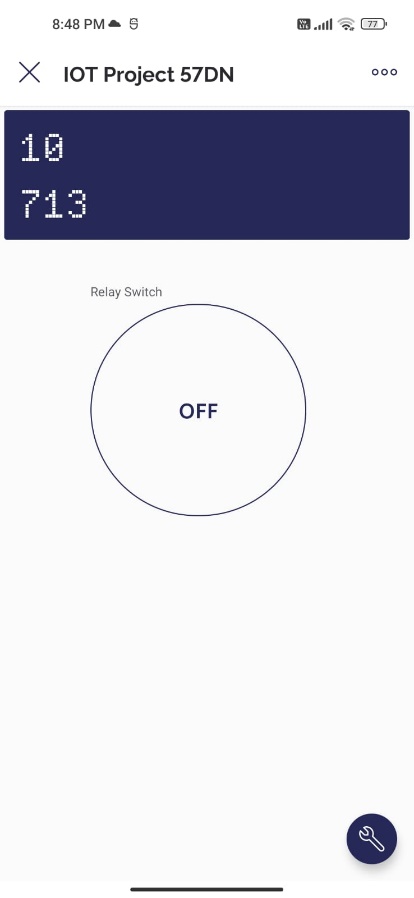
The heart of the software implementation lies in the algorithms that govern the system's behavior. We developed a set of algorithms to control the following key functions:

- Water Level Monitoring - The ultrasonic sensor continuously measures the water level, and the Node MCU processes this data to determine the current water level in the tank.

- Control Logic - The control logic determines when to activate or deactivate the waterpump based on predefined water level thresholds. If the water level falls below a certain threshold, the pump in turn on to fill the tank; otherwise, it remains closed.

2. User Interface Development

To provide a user-friendly interface, we designed a simple yet intuitive user interface. The interface displays real-time information on the current water level. The interface is accessible through a smartphone, making it convenient for users to monitor and control the system remotely.



**Testing and Calibration**

1. Sensor Calibration

Sensor calibration is a critical step to ensure accurate water level measurements. We performed a calibration process to account for factors like temperature and atmospheric conditions that might affect the sensor's accuracy. Calibration data was then used to convert raw sensor readings into meaningful water level measurements.

2. System Testing

After the hardware and software integration, extensive testing was conducted. The system was tested under various conditions, including different water levels, filling rates. The goal was to ensure that the system operated reliably and safely in all situations.

3. Fine-tuning

Based on the results of the testing phase, we fine-tuned the control algorithms and sensor calibration parameters to optimize system performance. Fine-tuning included adjustments to the water level thresholds for activating and deactivating the water pump to prevent overflows and ensure efficient water utilization.

The successful implementation of the Automatic Water Tank System with Monitoring Capability demonstrates its potential to provide a reliable and efficient solution for water management.

**Monitoring and Control**

**Real-time Monitoring**

Water Level - The system provides real-time monitoring of the water level within the tank using the ultrasonic sensor. Users can access this information through the Blink user interface on their devices.

**Automatic Control**

1. Filling Process - The system offers automatic control over the water filling process. It uses the water level measurements to determine when to initiate and terminate the filling process, ensuring the tank is neither underfilled nor overfilled.

2. Water Pump Control - The Water Pump allows for precise control over the water flow. The pump is automatically activated and deactivated as needed, preventing unnecessary water wastage.

**Maintenance and Support**

**Routine Maintenance**

1. Sensor Maintenance - Regular sensor maintenance is essential to ensure accurate water level measurements. This includes cleaning the sensor and checking for any physical damage or corrosion.

2. Software Updates - To keep the system up to date and secure, regular software updates are necessary. These updates can also include enhancements and new features based on user feedback and evolving technology.

**Troubleshooting**

1. Common Issues - The system may encounter common issues such as sensor malfunctions, pump failure, or connectivity problems.

2. Solutions - Solutions to common issues may include sensor recalibration, maintenance of the water pump, or checking and improving network connectivity.

**Benefits and Impact**

**Water Conservation**

The system promotes water conservation by ensuring efficient water usage. It prevents overfilling, reduces water wastage, and optimizes the filling process, ultimately conserving water.

**Cost Savings**

Users benefit from reduced water bills due to improved water management. The system's automation minimizes the risk of overflows and associated repair costs, leading to financial savings.

**Environmental Benefits**

By conserving water and reducing unnecessary water pumping, the system contributes to a lower carbon footprint and reduced energy consumption, thereby benefiting the environment.

**User Experience**

Users enjoy a seamless and user-friendly experience with remote monitoring and control capabilities. This enhances their control over water resources and overall satisfaction with the system.

**Challenges and Limitations**

**Technical Challenges**

Implementing and maintaining the sensor can pose technical challenges, including sensor accuracy, connectivity issues, and power management.

**Cost Considerations**

The cost of sensors, controllers, and data connectivity can be a limitation for some users. Balancing cost-effectiveness with system performance is crucial.

**Accessibility and Connectivity**

The system's functionality relies on network connectivity. Remote areas with poor connectivity may face limitations in real-time monitoring and control.

**Maintenance Challenges**

Regular maintenance and software updates are essential but can be overlooked or postponed, affecting system performance and reliability.

**Future Enhancements**

**Integration with Smart Home Systems**

Future enhancements may include integrating the system with smart home platforms, enabling seamless integration with other home automation devices and voice assistants.

**Enhanced Monitoring and Analytics**

Advanced analytics capabilities could provide users with insights into long-term water usage trends, enabling better water conservation strategies.

**Time Line**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tasks | Week 1 | Week 2 | Week 3 | Week 4 |
| 1. Solution identification and research |  |  |  |  |
| 1. Finalizing design and components |  |  |  |  |
| 1. Designing and connecting hardware components |  |  |  |  |
| 1. Coding the program |  |  |  |  |
| 1. Testing |  |  |  |  |
| 1. Fixing bugs |  |  |  |  |
| 1. Finishing with report and presentation |  |  |  |  |

**Source Code**

/\*

 \* this program is under editing for push notification

 \*/

// Fill-in information from your Blynk Template here

#include <Arduino.h>

#define BLYNK\_TEMPLATE\_ID "TMPL686qJHKjb"

#define BLYNK\_TEMPLATE\_NAME "IOT Project"

#define BLYNK\_FIRMWARE\_VERSION        "0.1.0"

#define BLYNK\_PRINT Serial

#define USE\_NODE\_MCU\_BOARD

#include "BlynkEdgent.h"

#define echoPin D5

#define trigPin D6

#define SOUND\_VELOCITY 0.034

int x = 0;

long duration;

float distance;

long max\_depth, current\_depth, rec\_max\_height, rec\_max\_depth, max\_water;

long height, current\_water, rec\_min\_height, rec\_min\_depth, min\_water, area;

bool state = LOW;

BLYNK\_WRITE(V0)

{

  area = 69.97;

  height = 22.17;

  max\_depth = 5;

  rec\_max\_height = (height - max\_depth);

  current\_depth = distance;

  current\_water = ((height - current\_depth)\*area)-115;

  if(param.asInt()==1){

    digitalWrite(4, HIGH);

    Serial.println("Relay ON");

    state = HIGH;

  }

  else{

    digitalWrite(4, LOW);

    Serial.println("Relay OFF");

    state = LOW;

  }

}

BLYNK\_CONNECTED()

{

  Blynk.syncVirtual(V0);

}

void water\_level(float distance)

{

  area = 69.97;

  height = 22.17;

  max\_depth = 5;

  rec\_max\_height = (height - max\_depth);current\_depth = distance;

  current\_water = ((height - current\_depth)\*area)-115;

  Serial.print("Water Level: ");

  Serial.println(current\_water);

  Blynk.virtualWrite(V3, current\_water);

  delay(500);

}

/\*if(distance > (max\_depth + 3)){

  }

  else{

    Serial.print("Water Level: ");

    Serial.println(1000);

    Blynk.virtualWrite(V3, 1000);

    delay(500);

  }

}

###\*/

/\*void automation()

{

  if(current\_water < 10){

    digitalWrite(4, LOW);

    Serial.println("Relay ON");

    delay;

  }

} \*/

void ultrasonic()

{

    digitalWrite(trigPin, LOW);

    delayMicroseconds(2);

    // Sets the trigPin on HIGH state for 10 micro seconds

    digitalWrite(trigPin, HIGH);

    delayMicroseconds(10);

    digitalWrite(trigPin, LOW);

    // Reads the echoPin, returns the sound wave travel time in microseconds

    duration = pulseIn(echoPin, HIGH);

    // Calculate the distance

    distance = duration \* SOUND\_VELOCITY/2;

    Serial.print("Distance: ");

    Serial.println(distance);

    Blynk.virtualWrite(V2, distance);

    delay(100);

}

void setup()

{

  Serial.begin(9600);

  pinMode(34, INPUT);

  pinMode(trigPin, OUTPUT);

  pinMode(echoPin, INPUT);

  pinMode(4, OUTPUT);

  BlynkEdgent.begin();

  delay(10);

}

void loop()

{

  BlynkEdgent.run();

  ultrasonic();

  water\_level(distance);

  if(state != HIGH){

    if(distance > 15 && distance < 17){

      digitalWrite(4, HIGH);

      delay(35000);

      digitalWrite(4, LOW);

      delay(5000);

      if(distance > 5 && distance < 6){

        digitalWrite(4, HIGH);

        delay(600);

      }

      else if(distance > 4){

        while(distance > 5){

          digitalWrite(4, HIGH);

        }

      }

      else{

        digitalWrite(4, LOW);

      }

    }

    else if(distance > 17){

      digitalWrite(4, HIGH);

    }

    else{

      digitalWrite(4, state);

    }

  }

}

**Conclusion**

In conclusion, the Automatic Water Tank System with Monitoring Capability represents a significant step towards efficient water management. It offers real-time monitoring and control, and the potential for future enhancements. While challenges and limitations exist, the benefits, including water conservation and cost savings, make it a valuable solution for users. Continuous maintenance ensures its long-term success and effectiveness in addressing water resource management challenges.