SMART WATER MANAGEMENT SYSTEM AT HOME

Mini-project Group No: 02

Register No:	**Name:**
1. 126003133	Kiran Soorya R S
2. 126003244	Sherwin L
3. 126003230	Sanjay M

Project Title: Smart Water Management System at home

Name of the Guide: Dr. Kuppan Chetty RM

Introduction

Water is a vital resource, and efficient use is crucial due to growing concerns about water scarcity and sustainability. Traditional home water management is often inefficient, leading to waste. Smart water management systems offer a modern solution, using technologies like sensors, controllers, IoT devices, and analytics to monitor, control, and optimise water usage. These systems provide real-time data, enabling informed decisions to promote sustainability and reduce waste.

Objectives

A Smart Water Management System at home aims to efficiently monitor, control, and optimise water usage to promote sustainability, reduce waste, and cut costs. Key objectives include:

- **Water Conservation**: Monitor and optimise consumption to minimise wastage.
- **Cost Savings**: Reduce water bills by managing usage efficiently.
- **Real-time Monitoring**: Track water usage in real-time for informed decision-making.
- **Automation and Control**: Remotely control devices and schedule water usage based on preferences or environmental conditions.
- **Data Insight**: Analyse consumption patterns for further optimization.
- **User Awareness**: Raise awareness and encourage responsible water usage through feedback and alerts.

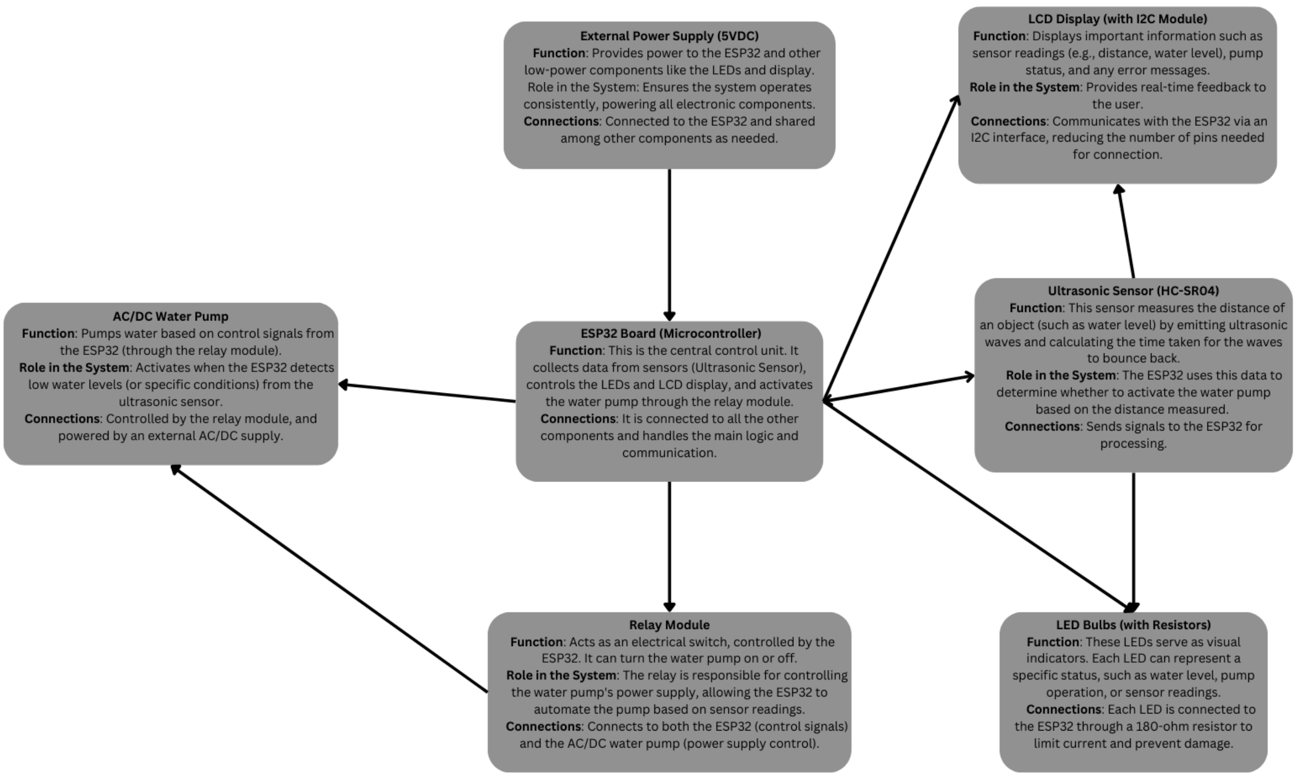
Overall, the system empowers homeowners to make informed decisions, contributing to sustainable and efficient water management.

Scope

The Smart Water Management System is designed to monitor, control, and optimize household water usage. It integrates hardware and software components for real-time data collection, automation, and remote control. The system tracks water usage, schedules and manages flow to reduce wastage, and lowers water bills, contributing to environmental sustainability. Data analytics provide insights into consumption patterns, with alerts and feedback to encourage responsible water use. Potential challenges and limitations are considered, highlighting areas for future improvement.

Method

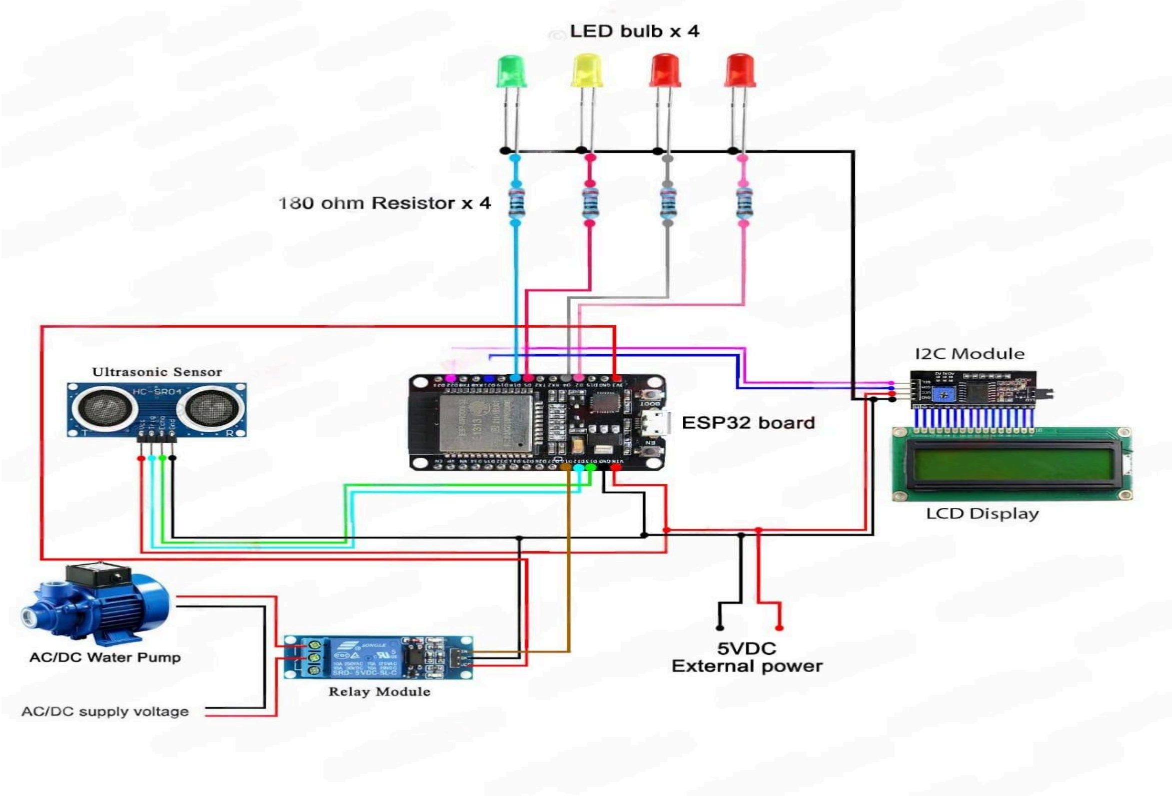
BLOCK DIAGRAM



COMPONENTS SELECTION

- ESP32 Board (Microcontroller)
- Ultrasonic Sensor (HC-SR04)
- LCD Display (with I2C Module)
- AC/DC Water Pump
- LED Bulbs (with Resistors)
- External Power Supply (5V DC)
- Relay Module

CIRCUIT DIAGRAM



ALGORITHM

1. System Initialization:
    1. Start the serial monitor for debugging.
    2. Connect to WiFi and Blynk using provided credentials.
    3. Initialise the LCD display and configure the pin modes for LEDs, relay, and ultrasonic sensor.
    4. Display a "System Loading" message on the LCD for 4 seconds.
2. Ultrasonic Sensor:
    1. Send a trigger pulse and calculate the distance based on the echo pulse.
    2. If the distance is less than or equal to the maximum tank level:
    3. Display the water level in Blynk (virtual pin V2).
    4. Compare the distance to predefined water levels:
    5. If the water level is very low, display the corresponding message and turn on LED1.
    6. If low, turn on LED1 and LED2.
    7. If medium, turn on LED1, LED2, and LED3.

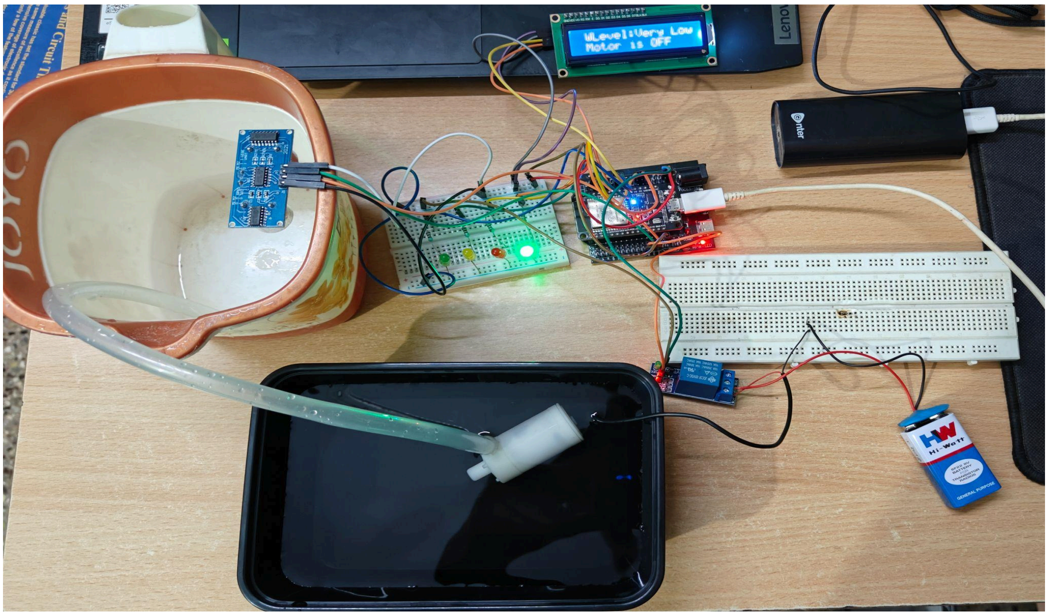
    8. If full, turn on all LEDs and ensure the motor is OFF by controlling the relay.
3. Blynk Button Control:
    1. Check the status of the motor (relay) using a button from Blynk (virtual pin V1).
    2. If the button is pressed, turn the motor ON and update the LCD.
    3. If the button is released, turn the motor OFF and update the LCD.
4. Main Loop:
    1. Continuously monitor the water level and update the motor status based on the Blynk input.

PSEUDO CODE

```
START

 // Initialize system
 Start serial for debugging
 Connect to WiFi and Blynk using auth, ssid, and pass
 Initialize LCD and set pin modes (LEDs, relay, ultrasonic sensor)
 Display "System Loading" message on LCD

 LOOP
  // Read ultrasonic sensor
  Trigger ultrasonic sensor, measure distance to water level

  IF distance <= MaxLevel THEN
   Write water level to Blynk (V2)

   IF distance >= Level1 THEN
    Display "Very Low", turn ON LED1, turn OFF others
   ELSE IF distance >= Level2 THEN
    Display "Low", turn ON LED1, LED2, turn OFF others
   ELSE IF distance >= Level3 THEN
    Display "Medium", turn ON LED1, LED2, LED3, turn OFF LED4
   ELSE IF distance >= Level4 THEN
    Display "Full", turn ON all LEDs
    Turn OFF motor (relay HIGH), display "Motor is OFF"
   ENDIF
  ELSE
   Write 0 to Blynk (V2)
  ENDIF

  // Check motor control from Blynk
  IF Blynk button (V1) pressed THEN
   Turn ON relay (LOW), display "Motor ON"
  ELSE
   Turn OFF relay (HIGH), display "Motor OFF"
  ENDIF
 END LOOP

END
```

RESULTS

OVERALL DIAGRAM



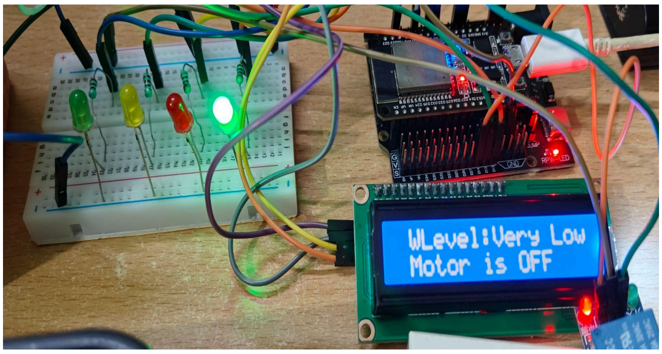
CODE

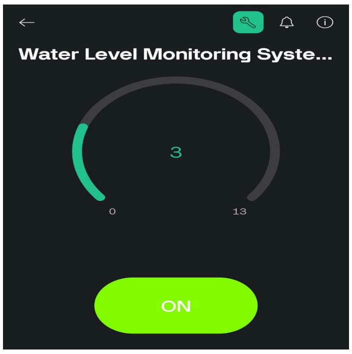
```
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#define BLYNK_PRINT Serial
#define LED1 2
#define LED2 4
#define LED3 5
#define LED4 18
#define trig 12
#define echo 13
#define relay 14
#define BLYNK_TEMPLATE_ID "TMPL3ZbWLLD1s"
#define BLYNK_TEMPLATE_NAME "Water Level Monitoring System"
int MaxLevel = 13;
int Level1 = (MaxLevel * 75) / 100;
int Level2 = (MaxLevel * 65) / 100;
int Level3 = (MaxLevel * 55) / 100;
int Level4 = (MaxLevel * 35) / 100;
LiquidCrystal_I2C lcd(0x27,16,2);BlynkTimer timer;
char auth[] = "WMojxJ-b6VzB0b4G4Ovi3dep6BPTbwJc";
```

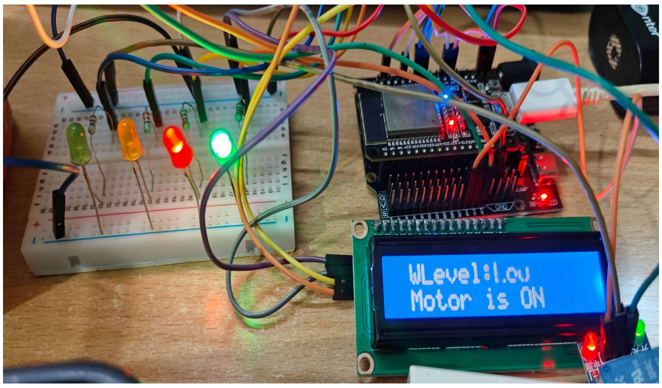
```
char ssid[] = "OnePlus 12R Sher";char pass[] = "f5g2gkg7";

void setup() {
  Serial.begin(115200); Blynk.begin(auth, ssid, pass);
  lcd.init(); lcd.backlight();
  pinMode(LED1, OUTPUT); pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT); pinMode(LED4, OUTPUT);
  pinMode(trig, OUTPUT); pinMode(echo, INPUT);
  pinMode(relay, OUTPUT); digitalWrite(relay, HIGH);
  lcd.setCursor(3,0); lcd.print("System");
  lcd.setCursor(3,1); lcd.print("Loading..");
  delay(4000); lcd.clear();
}

void ultrasonic() {
  digitalWrite(trig, LOW); delayMicroseconds(4);
  digitalWrite(trig, HIGH); delayMicroseconds(10);
  digitalWrite(trig, LOW); long t = pulseIn(echo, HIGH);
  int distance = t / 29 / 2; Serial.println(distance);
  int blynkDistance = (distance - MaxLevel) * -1;
  if (distance <= MaxLevel) {
    Blynk.virtualWrite(V2, blynkDistance);
  } else {
    Blynk.virtualWrite(V2, 0);
  }
  lcd.setCursor(1, 0); lcd.print("WLevel:");

  if (Level1 <= distance) {
    lcd.setCursor(8, 0); lcd.print("Very Low");
    digitalWrite(LED1, HIGH); digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW); digitalWrite(LED4, LOW);
  } else if (Level2 <= distance && Level1 > distance) {
    lcd.setCursor(8, 0); lcd.print("Low");
    lcd.print("      "); digitalWrite(LED1, HIGH);
    digitalWrite(LED2, HIGH); digitalWrite(LED3, LOW);
    digitalWrite(LED4, LOW);
  } else if (Level3 <= distance && Level2 > distance) {
    lcd.setCursor(8, 0); lcd.print("Medium");
    lcd.print("      "); digitalWrite(LED1, HIGH);
    digitalWrite(LED2, HIGH); digitalWrite(LED3, HIGH);
    digitalWrite(LED4, LOW);
  } else if (Level4 <= distance && Level3 > distance) {
    lcd.setCursor(8, 0); lcd.print("Full");
    lcd.print("      "); digitalWrite(relay, HIGH);
```

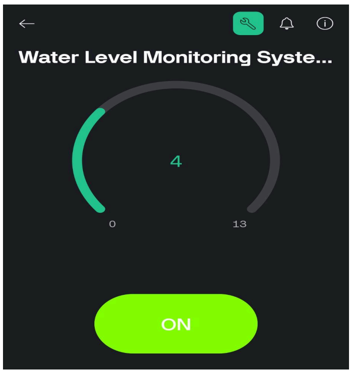
```
    lcd.setCursor(1, 1); lcd.print("Motor is OFF");
    Blynk.virtualWrite(V1, 0); digitalWrite(LED1, HIGH);
    digitalWrite(LED2, HIGH); digitalWrite(LED3, HIGH);
    digitalWrite(LED4, HIGH);
  }
}
BLYNK_WRITE(V1) {
  int Relay = param.asInt();
  if (Relay == 1) {
    digitalWrite(relay, LOW); lcd.setCursor(1, 1);
    lcd.print("Motor is ON "); Blynk.virtualWrite(V1,0);
  } else {
    digitalWrite(relay, HIGH); lcd.setCursor(1, 1);
    lcd.print("Motor is OFF");
  }
}

void loop() {
  ultrasonic(); Blynk.run();
}
```

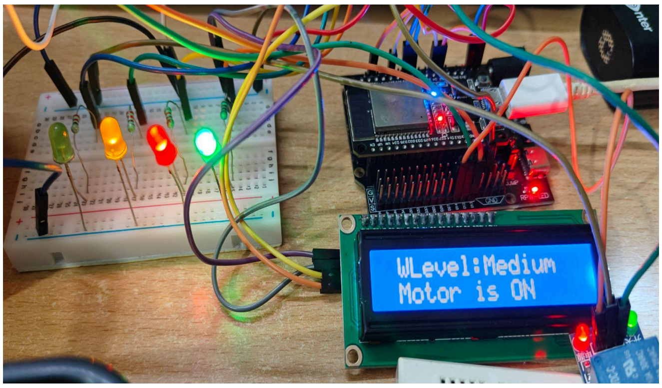
PICTURES DEPICTING THE WORKING

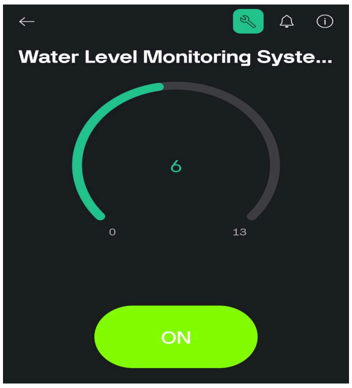


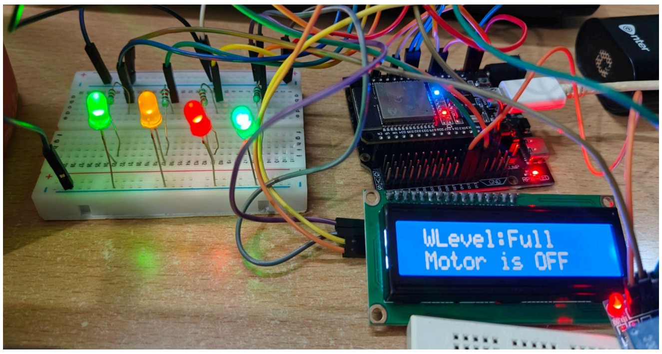


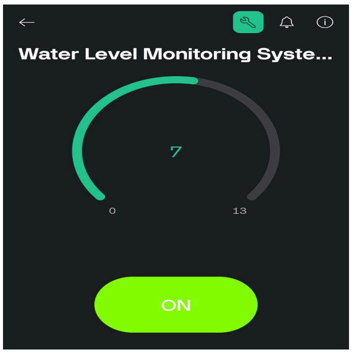












INFERENCE

A Smart Water Management System (SWMS) utilizes Internet of Things (IoT) technology to enhance the efficiency, sustainability, and management of water resources. It leverages various sensors, devices, and data analytics to monitor, control, and optimize water usage.

Automated Water Management

Smart water management system that adjust based on water level in the tank.

Automated valves that manage flow based on real-time data.

CONCLUSION & FUTURE SCOPE

In conclusion, implementing a Smart Water Management System (SWMS) at home is a vital step toward sustainability and resource conservation. By leveraging IoT technologies and real-time data analytics, homeowners can effectively monitor and control their water usage, reducing wastage and fostering responsible consumption habits. This not only leads to significant cost savings but also enhances awareness of water consumption. As water scarcity and environmental concerns intensify, adopting smart water management solutions is essential for creating resource-efficient homes of the future.

The future of SWMS is promising, with potential developments in artificial intelligence and machine learning to predict usage trends and optimise water supply automatically. Integrating mobile applications can further enhance user engagement by providing personalised insights. Additionally, collaboration with municipal water authorities could enable real-time monitoring of water distribution systems, reducing losses and improving efficiency. Decentralised water treatment and recycling systems connected to SWMS can promote community-wide sustainability. Raising awareness about the benefits of these systems will be crucial for widespread adoption, ensuring that smart water management plays a pivotal role in fostering sustainable living in our homes and communities.

REFERENCES

https://blynk.io/

https://github.com/blynkkk/blynk-library

https://www.youtube.com/@adtechknow2761

https://www.youtube.com/@TechStudyCell