Microprocessors and Microcontrollers

A MINOR PROJECT REPORT

ON

Water level Controller With Dry Run detection.

(Measure, Control, Detect)

Submitted By

•Santosh Mourya (20EEE1029)

•Sheetal Jain (20EEE1031)

•Vijay Gautam (20EEE1037)

November 2022

NATIONAL INSTITUTE OF TECHNOLOGY GOA

Farmagudi, Ponda, Goa – 403 401, India

Department of Electrical and Electronics Engineering

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

The total amount of water available on Earth has been estimated at 1.4 billion cubic kilometers, enough to cover the planet with a layer of about 3 km. About 95% of the Earth's water is in the oceans, which is unfit for human consumption. About 4% is locked in the polar ice caps, and the rest lo/o constitutes all fresh water found in rivers, streams and lakes which is suitable for our consumption. A study estimated that a person in India consumes an average of 135 liters per day' This consumption would rise by 40%by the year 2025. This signifies the need to preserve our freshwater resources.

•This project aims to help judge the water level inside the water tank and display it accordingly on the screen while also preventing any overflow of water which is achieved through the NodeMCU. To sense the dry run motor by checking the presence of water in the underground tank.

•If the water level in the overhead tank intake is full, the motor should shut off, and if it is below 25%, it should turn on.

•To prevent dry run the motor should turn off if the water level in the underground tank is low.

.

2. Block Diagram

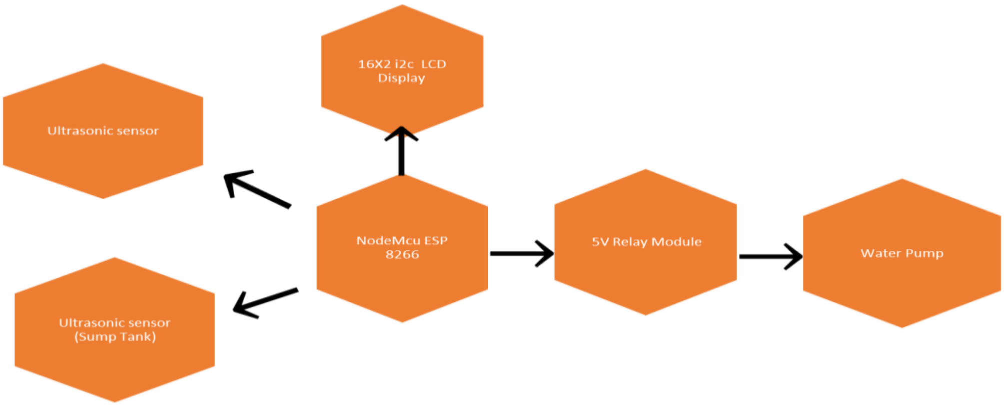


Figure 2-1 Block Diagram Of water level controller.

3. Working:

The Block Diagram consists of:

1. NodeMCU- Microcontroller with features such as GPIO, PWM, ADC, and etc,
2. 2 Ultrasonic sensors- To measure water level in both the tanks.
3. 16*2 i2c LCD Display- To display water level and Motor status
4. 5v relay module - To switch Ac motor on and off on receiving signal.
5. Ac motor (Water pump) which needs to be controlled.

This controller setup has two ultrasonic sensors, one in sump and other in overhead tank.

•Whenever the sensor is ON the controller has the power to control the pump.

•If the sensor is OFF the whole controller is turned OFF.

•Now let assume that the sump is full and the overhead tank is empty.

•Now the sensors in the overhead tank is turned ON.

•The sensor latches the AC relay so that the pump turns ON.

•The main power switch is ON, the pump is ON condition.

•As soon as the water level increases the lower sensor is turned off but the pump continues to be on.

•As water reaches the maximum level the upper sensor is also turned OFF so it unlatches the AC relay.

•So that the both relay is unlatched the pump is turned OFF.

•All informations were communicated to users through lcd and even through blynk iot applications through wifi.

4. CIRCUIT SCHEMATIC DIAGRAM

Schematic Circuit is designed using proteus software and testing of program was first done using this software.

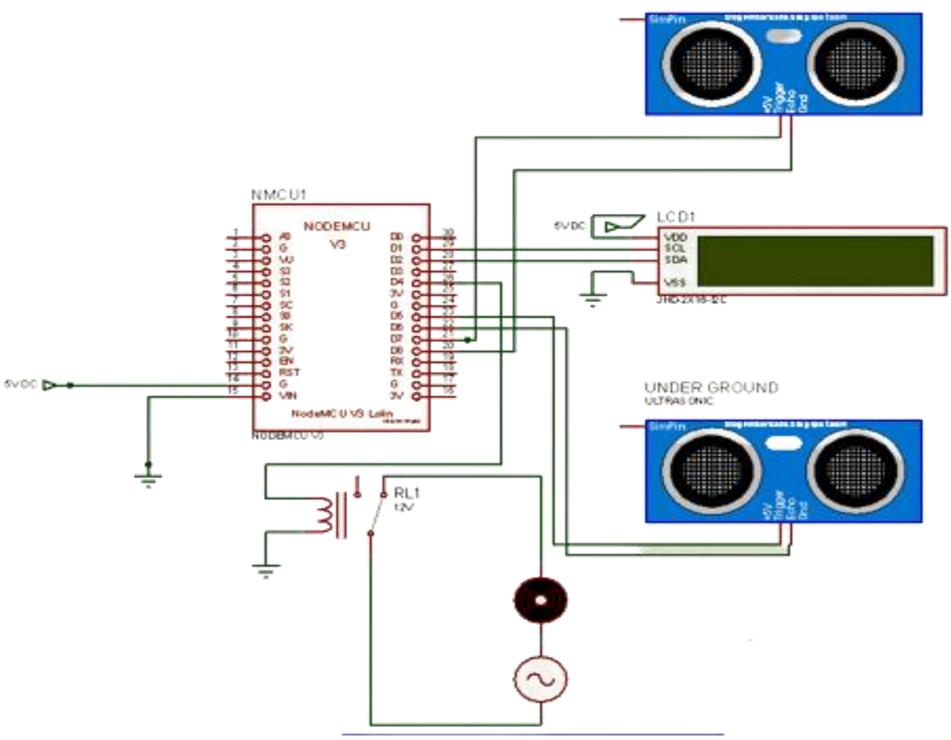


Figure 4-1: Schematic diagram for water level controller.

Calculations and Equations:

Calculation to measure Distance using Ultrasonic sensor in cm.

$$distance\_cm = \frac{(\frac{duration}{29})}{2;}$$

$$distance\_in = distance\_cm * 0{,}393701;$$

$$1\,m/s = \frac{100}{1000000} = 0.0001\,cm/\mu s$$

$$343\,m/s = 0.0001 * 343 = 0.0343\,cm/\mu s$$

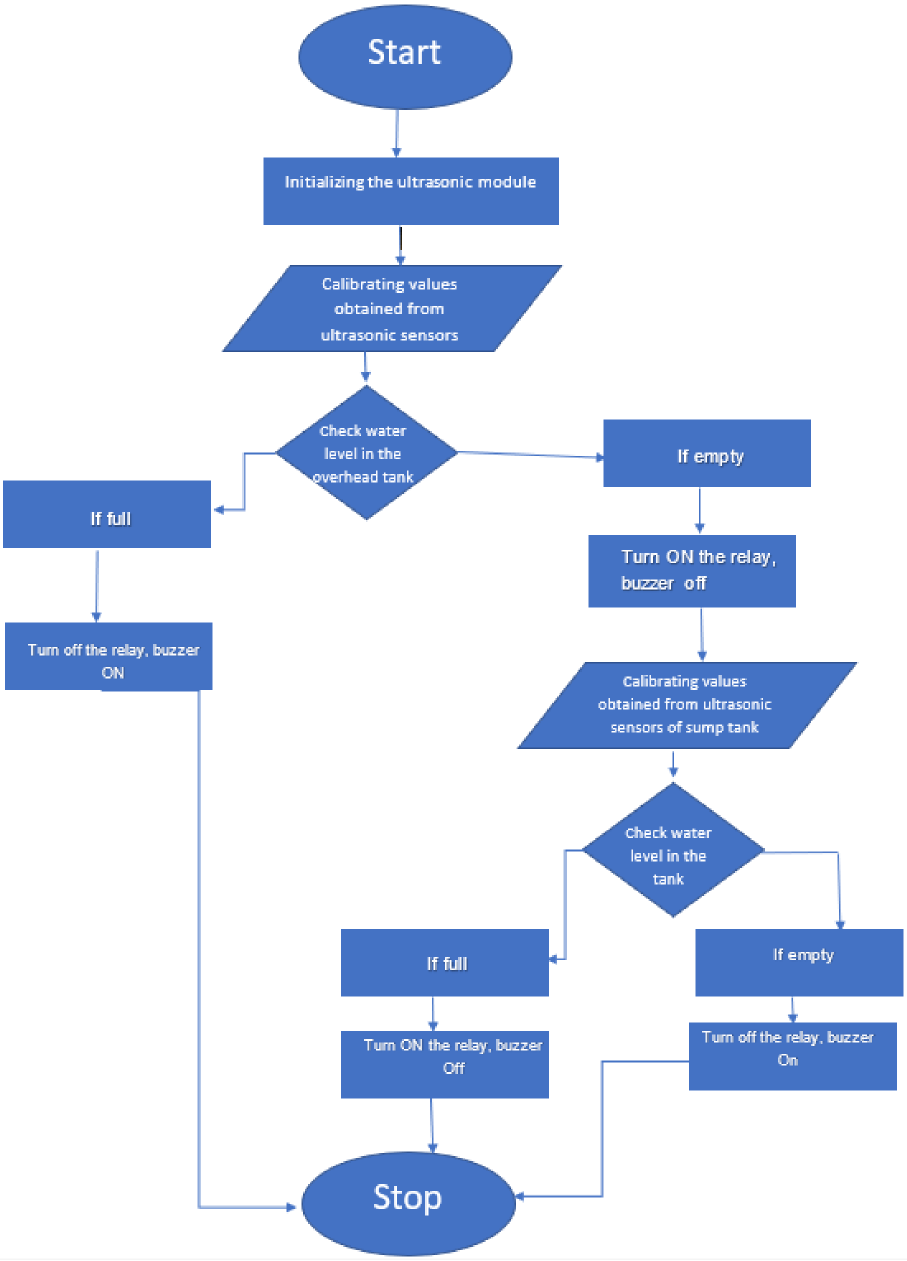
$$0.0343\,cm/\mu s = 1\,/\,0.0343 = 29.155\,\mu s/cm$$

Sound travels at 343 meters per second, which means it needs 29.155 microseconds per centimeter. So, we have to **divide the duration by 29 and then by 2**, because the sound has to travel the distance twice. It travels to the object and then back to the sensor.

Table 4-1 List of components

SL. No	Component	Quantity
1	Ultrasonic Sensor	2
2	NodeMCU	1
3	16*2 I2C LCD Display	1
4	Relay	1
5	Ac Motor (Water pump) 230Vll8V,50Hz	1

5. Flow chart


Start
Initializing the ultrasonic module
Calibrating values obtained from ultrasonic sensors
Check water level in the overhead tank
If empty
If full
Turn ON the relay, buzzer off
Turn off the relay, buzzer ON
Calibrating values obtained from ultrasonic sensors of sump tank
Check water level in the tank
If full
If empty
Turn ON the relay, buzzer Off
Turn off the relay, buzzer On
Stop

6 SOFTWARE DESCRIPTION

Arduino IDE (integrated development environment):- Arduino IDE is an open-source software program that allows users to write and upload code within a real-time work environment.

Programs:

```
//Include the library files
#include <LiquidCrystal_I2C.h>
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
//Initialize the LCD display
LiquidCrystal_I2C lcd(0x27, 16, 2);
char auth[] = "SOSGXNn5CxsaVFQmUudeYUzzaqZse9Am";  //Enter your Auth token
char ssid[] = "OPPO A53s 5G";                    //Enter your WIFI name
char pass[] = "1234567890";                      //Enter your WIFI password
BlynkTimer timer;
// Define the component pins
#define trig1 D7
#define echo1 D8
#define trig2 D5
#define echo2 D6
#define relay D4
//#define buzzPin D4
int distance_1 = 0;
int distance_2 = 0;
int MaxLevel = 15;
int Level1 = (MaxLevel * 75) / 100;
int Level2 = (MaxLevel * 55) / 100;
int Level3 = (MaxLevel * 45) / 100;
int Level4 = (MaxLevel * 35) / 100;
int Level5 = (MaxLevel * 25) / 100;
bool Relay =1;
int distance1 = 0, distance2 = 0;
void setup() {
  Serial.begin(9600);
  lcd.init();
  lcd.backlight();
  pinMode(trig1, OUTPUT);
  pinMode(echo1, INPUT);
  pinMode(trig2, OUTPUT);
  pinMode(echo2, INPUT);
  pinMode(relay, OUTPUT);
  Blynk.begin(auth, ssid, pass, "blynk.cloud", 80);
  lcd.setCursor(0, 0);
  lcd.print("Water level");
  lcd.setCursor(4, 1);
  lcd.print("Controller");
  delay(4000);
  lcd.clear();
  timer.setInterval(100L, ultrasonic);
}
void ultrasonic() {
  digitalWrite(trig2, LOW);
  delayMicroseconds(4);
  digitalWrite(trig2, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig2, LOW);
  long t2 = pulseIn(echo2, HIGH);
  distance2 = t2 / 29 / 2;
  distance_2 = distance2;
  int blynkDistance2 = (distance2 - MaxLevel) * -1;
  if (distance2 <= MaxLevel) {
    Blynk.virtualWrite(V2, blynkDistance2);
  } else {
    Blynk.virtualWrite(V2, 0);
  }
  digitalWrite(trig1, LOW);
  delayMicroseconds(4);
  digitalWrite(trig1, HIGH);
```

```
delayMicroseconds(10);
digitalWrite(trig1, LOW);
long t1 = pulseIn(echo1, HIGH);
distance1 = t1 / 29 / 2;
distance_1 = distance1;
int blynkDistance1 = (distance1 - MaxLevel) * -1;
if (distance1 <= MaxLevel) {
  Blynk.virtualWrite(V0, blynkDistance1);
} else {
  Blynk.virtualWrite(V0, 0);
}
lcd.setCursor(0, 0);
lcd.print("WLevel:");
if (distance1 > Level1) {
  lcd.setCursor(8, 0);
  lcd.print("Empty");
  lcd.print("      ");
  //digitalWrite(relay, LOW);  //only wnat motor run when overhead level decrease below 25%
} else if (distance1 <= Level1 && distance1 > Level2) {
  lcd.setCursor(8, 0);
  lcd.print("very Low");
  lcd.print("      ");
  //digital.Write(relay, LOW);
} else if (distance1 <= Level2 && distance1 > Level3) {
  lcd.setCursor(8, 0);
  lcd.print("Low");
  lcd.print("      ");
  //digital.Write(relay, LOW);
} else if (distance1 <= Level3 && distance1 > Level4) {
  lcd.setCursor(8, 0);
  lcd.print("Medium");
  lcd.print("      ");
  //digitalWrite(relay, LOW);
} else if (distance1 <= Level4 && distance1 > Level5) {
  lcd.setCursor(8, 0);
```

```
    lcd.print("High");
    lcd.print("      ");
    //digitalWrite(relay, LOW);
  } else if ( distance1 <= Level5 && distance1 > MaxLevel){// water level is greater than 85%
    lcd.setCursor(8, 0);
    lcd.print("Full");
    lcd.print("      ");
    //digitalWrite(relay, HIGH);  // turn the motor off when overhead water level is full
  }
}//Get the button value
BLYNK_WRITE(V1) {
  Relay = param.asInt();
}
// void motor(){
//   digitalWrite(relay, HIGH);
// }
void loop() {  //
                if (Relay == 1) {
   // digitalWrite(relay, LOW);
    lcd.setCursor(1, 1);
    lcd.print("Motor is ON   ");
    //lcd.setCursor(0, 0);
    //lcd.print("WLevel:");
    if (distance2 >= Level1) {
      digitalWrite(relay, HIGH);  //only want motor run when underground tank has water more than 25% overrides overhead tank
    } else {
      if (distance1 > Level1) {
        //lcd.setCursor(8, 0);
        //lcd.print("Empty");
        //lcd.print("      ");
        digitalWrite(relay, LOW);  //only wnat motor run when overhead level decrease below 25%
      } else if (distance1 <= Level1 && distance1 > Level2) {
        //lcd.setCursor(8, 0);
        //lcd.print("very Low");
        //lcd.print("      ");
        //digital.Write(relay, LOW);
      } else if (distance1 <= Level2 && distance1 > Level3) {
        //lcd.setCursor(8, 0);
        //lcd.print("Low");
        //lcd.print("      ");
        //digital.Write(relay, LOW);
      } else if (distance1 <= Level3 && distance1 > Level4) {
        // lcd.setCursor(8, 0);
        // lcd.print("Medium");
        // lcd.print("      ");
        //digitalWrite(relay, LOW);
      } else if (distance1 <= Level4 && distance1 > Level5) {
        // lcd.setCursor(8, 0);
        // lcd.print("High");
        // lcd.print("      ");
        //digitalWrite(relay, LOW);
      } else {  // water level is greater than 85%
        // lcd.setCursor(8, 0);
        // lcd.print("Full");
        // lcd.print("      ");
        digitalWrite(relay, HIGH);  // turn the motor off when overhead water level is full
      }
    }
  }
  else {
    digitalWrite(relay, HIGH);
    lcd.setCursor(1, 1);
    lcd.print("Motor is OFF   ");
  }

  Blynk.run();  //Run the Blynk library
  timer.run();  //Run the Blynk timer
}
```

5. PROJECT OUTCOMES

The water level controller designed in this project can be used to control water flow. There is another sensor in the underground tank to know the presence of source water, i.e. the UGT, actually has water or not. If no water source is present, then the submersible pump would start running unnecessarily and overheat itself. This is taken care of by implementing another sensor in UGT.

1. The time taken by the control circuit to stop and start the motor when water reaches its predetermined level is about 0.5-1 sec.
2. Mainly, we have faced the problem using ultrasonic sensors with **limited testing distance, inaccurate readings, and inflexible scanning methods**.
3. Limited testing distance: Ultrasonic sensors can measure distance a maximum of about 30 cm.
4. Inaccurate reading: original distance and measured distance using blynk application showed an errors of +-3%
5. Inflexible scanning method: An ultrasonic sensor **uses a transducer to send and receive ultrasonic pulses that relay back information about an object's proximity**.
6. Since this is an electrical device there is a need for continuous supply for its operation.

ADVANTAGES & DISADVANTAGES

Advantages –

- Automatic water level controller is used to automatically fill the overhead tank as and when it gets empty and control the water level in it as well as it ensures presence of water in underground or source and prevents dry run.
- Automatic water level controller is simple and easy to install.
- Automatic water level controller has low maintenance.
- Automatic water level controller has compact and elegant design.
- Automatic water level controller is fully automatic.
- Automatic water level controller avoids the seepage of walls and roofs when the tank overflows.
- Automatic water level controller has safe operation of motor/pump within permissible voltage limits.

Disadvantages - It is a passive electrical system and hence it requires continuous power supply.

6. Conclusion:

- These days, when Earth's reserve of consumable water is decreasing every moment, every drop has its value. Water level controller is a simple yet effective way to prevent wastage of water. Its simplicity in design and low cost components make it an ideal piece of technology for the common man.
- Thus the automatic water level controller is a big boon as concerned with the house hold applications as well as other water saving purposes including agricultural sector and industries.
- Based on the survey result, it is found that the automatic water level controller has a rising demand and it is a good asset from the electronics perspective.
- Hence we conclude that This system is very beneficial in rural as well as urban areas.
- It helps in the efficient utilization of available water sources.
- If used on a large scale, it can provide a major contribution in the conservation of water for us and the future generations.

8. Circuit Layout:

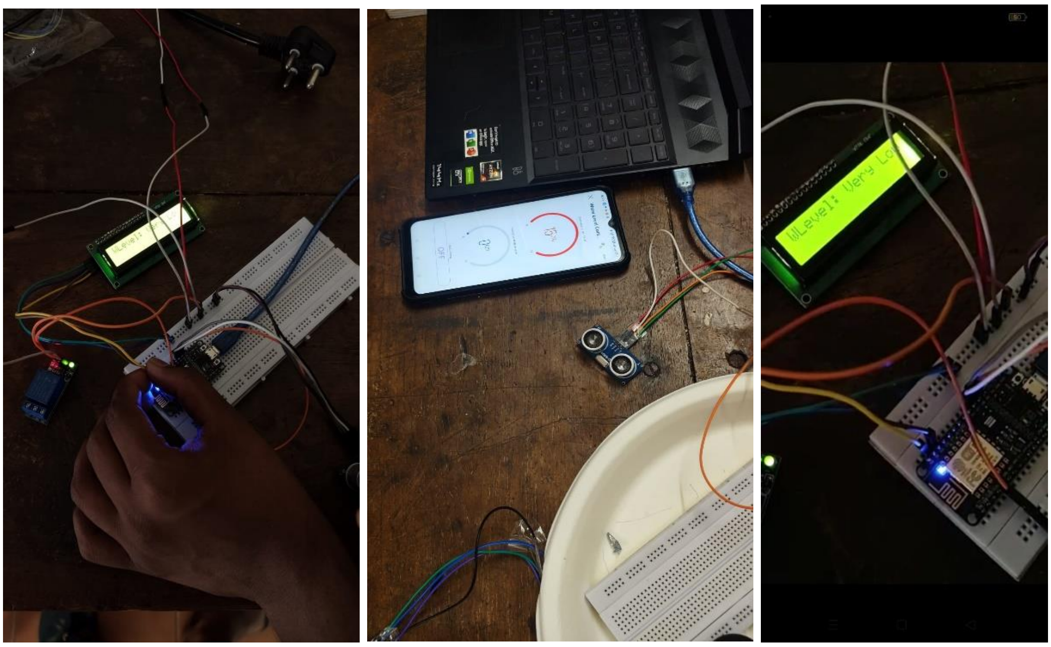


Figure 8-1: Circuit Layout.

9. REFERENCES

- https://docs.blynk.io/en/

- https://www.youtube.com/watch?v=6R4YGrXtQ8c

- https://github.com/gamegine/HCSR04-ultrasonic-sensor-lib

- https://nodemcu.readthedocs.io/en/release/

- https://create.arduino.cc/projecthub/akshayjoseph666/interface-i2c-16x2-lcd-with-arduino-uno-just-4-wires-273b24

- https://create.arduino.cc/projecthub/akshayjoseph666/interface-i2c-16x2-lcd-with-arduino-uno-just-4-wires-273b24

	Evaluator	**Signature**
1		
2		