VERZEO IoT INTERNSHIP-MARCH BATCH 2021 [IoT-MAJOR-MARCH]

Project statement: Need to design a Water level IoT system which should give live feeds of the current water level in the tank to the application on a smartphone.

Team members:

- Kalepu Shree Lavalika
- Niketh bayya mahesh

Aim: To design a water level IoT system that provides the current water level to the application on a smartphone.

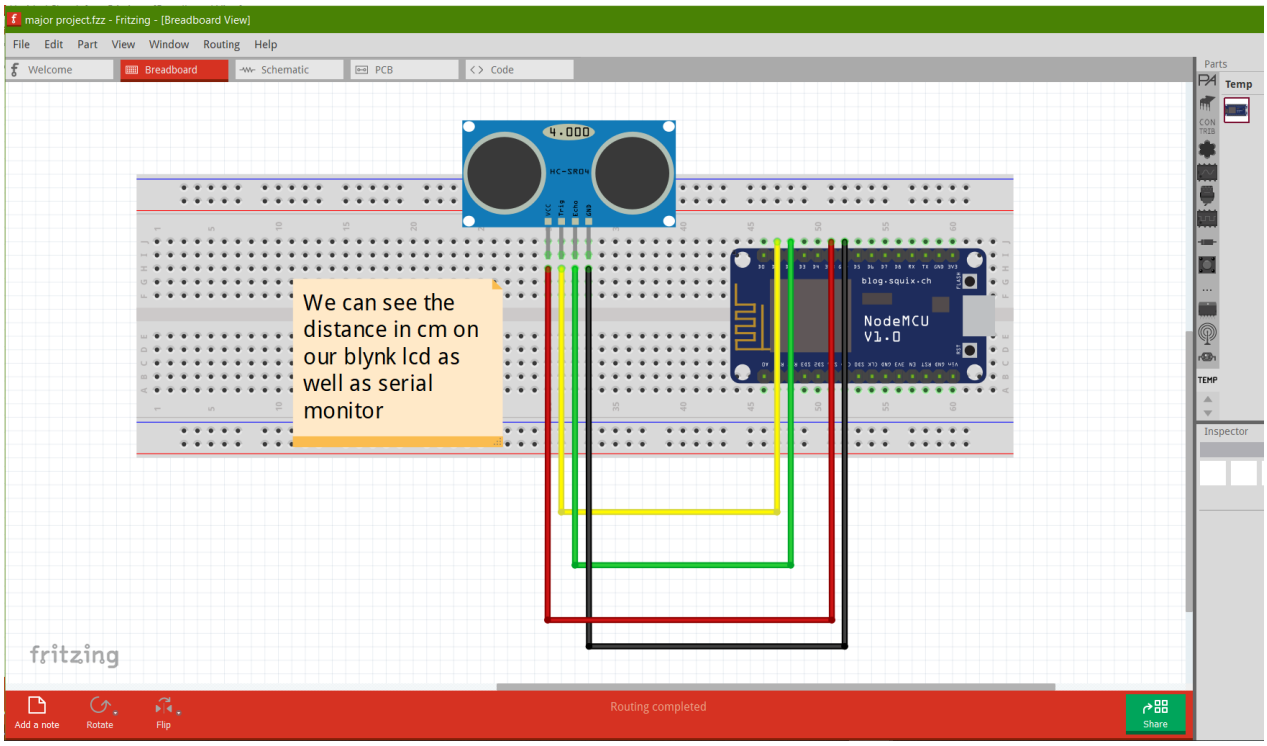
Material used:

- **ESP8266 micro-controller** -This will be the heart of our whole project.
- **Ultra-sonic sensor** -used for sensing the level by ultrasonic sound waves
- **Jumper wires** -used to hook-up wires are the way to go for connections
- **Bread board** -All the connections will be made on the breadboard itself

Application used:

- Blynk app

Circuit diagram:



Purpose and use of this project:

IOT based Water Level Monitoring system is an innovative system which will inform the users about the level of liquid and will prevent it from overflowing. Thus, this system helps to prevent the wastage of water by informing about the liquid levels of the containers.

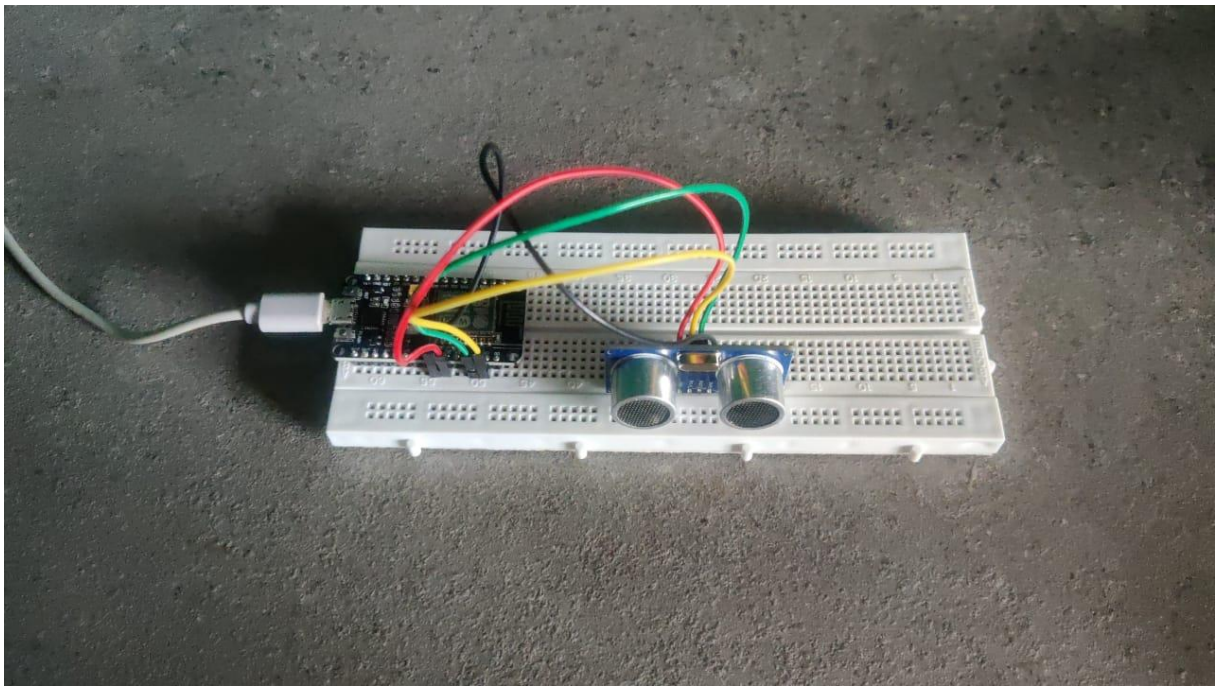
The Ultra sonic sensor is placed above the tank which continuously monitors the water level in real time. This information will be updated in the cloud and user can analyse the amount of water.

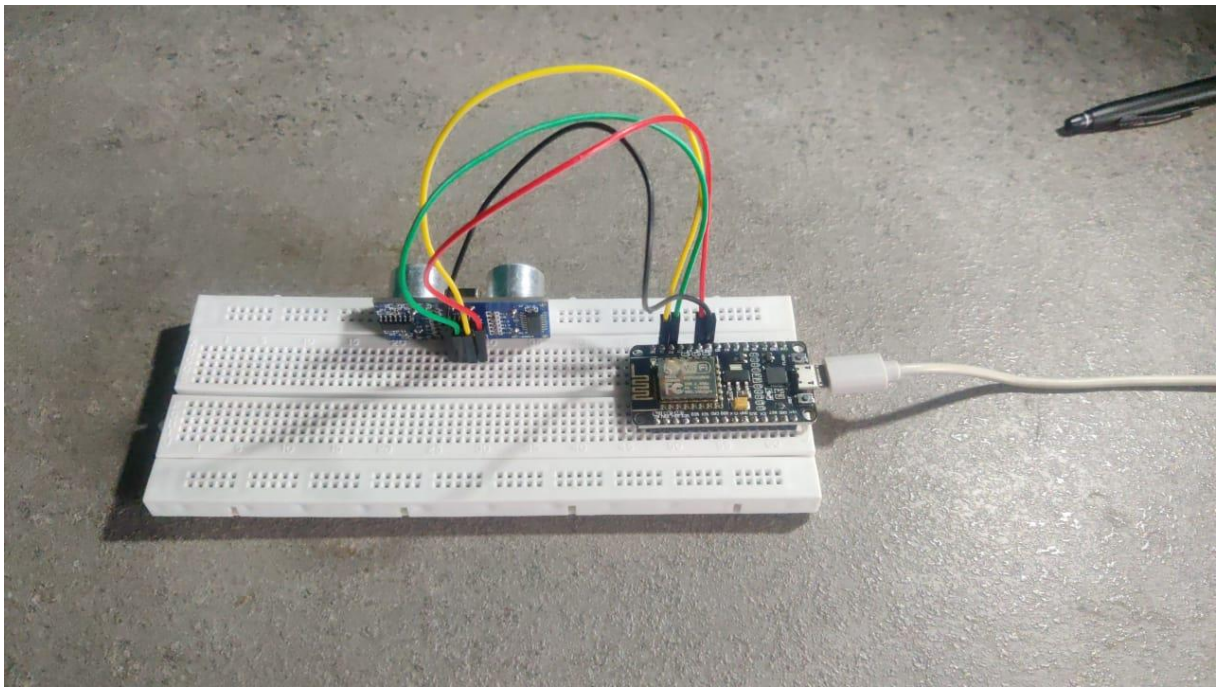
To determine the distance to the water, sensor transmits a sound pulse that reflects from the surface of the water and measures the time it takes for the echo to return.

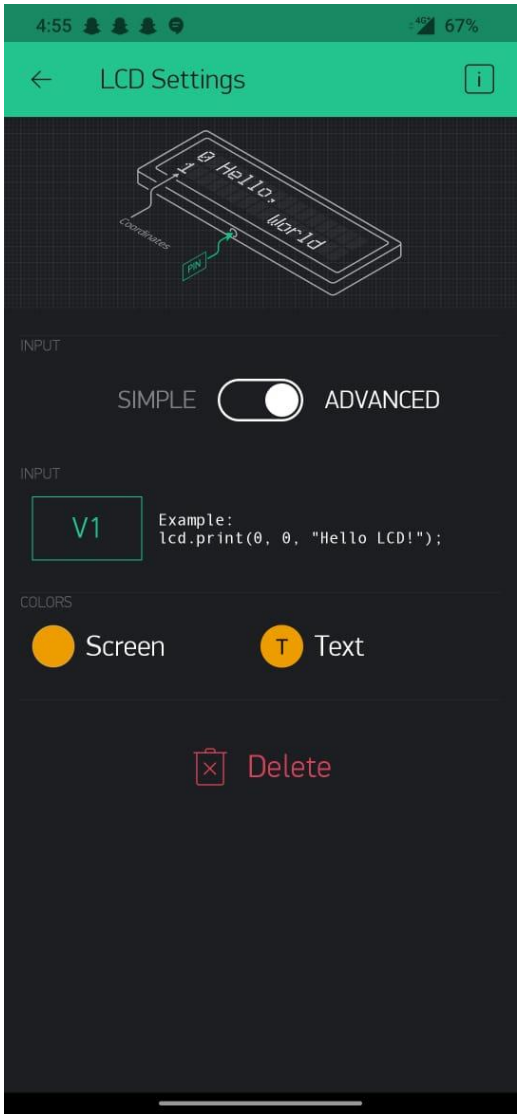
Procedure:

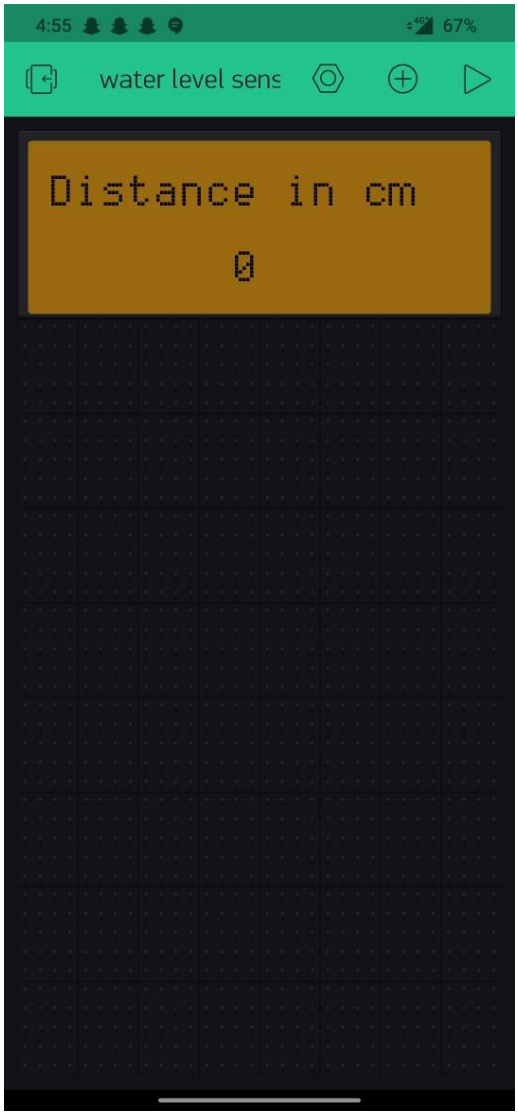
1. In this project for receiving the values from ultra-sonic sensor we will be using Blynk application in mobile.
2. After installing the Blynk app, click create new project. You will receive an API key through your registered mail-id which is very essential for transferring data to your Blynk app
3. Write the code in Arduino IDE (software) for connecting the esp8266 with Wi-Fi, enter your API key, Wi-Fi ssid, password and integrate it with Blynk app
4. In Blynk, create a new project, choose device name as esp8266. Assign the respective pin numbers in the Blynk, which you have used in the code for the connection
5. After creating a new project, in the widget box add led's (select different colours) and a gauge (provides the real-time reading of the distance)
6. After connecting all the components as shown in the circuit diagram upload the code in Arduino IDE and run the project in Blynk, you will see the distance from the sensor to the water level of the container
7. As you vary the water level, it calculates and provides the real time values in the Blynk
8. Finally, you will receive the live feeds of water level from the sensor to the application on a software.

Pictures of this project:









Code for this project:

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#define TRIGGERPIN D1
#define ECHOPIN    D2

char auth[] = " YOUR API KEY";

char ssid[] = "WIFI ID";
char pass[] = "WIFI  PASSWORD";

WidgetLCD lcd(V1);

void setup()
{
```

```
  Serial.begin(9600);
  pinMode(TRIGGERPIN, OUTPUT);
  pinMode(ECHOPIN, INPUT);
  Blynk.begin(auth, ssid, pass);
  lcd.clear();
  lcd.print(0, 0, "Distance in cm");
}

void loop()
{
  lcd.clear();
  lcd.print(0, 0, "Distance in cm");
  long duration, distance;
  digitalWrite(TRIGGERPIN, LOW);
  delayMicroseconds(3);

  digitalWrite(TRIGGERPIN, HIGH);
  delayMicroseconds(12);

  digitalWrite(TRIGGERPIN, LOW);
  duration = pulseIn(ECHOPIN, HIGH);
  distance = (duration/2) / 29.1;
  Serial.print(distance);
  Serial.println("Cm");
  lcd.print(7, 1, distance);
  Blynk.run();

  delay(3500);
}
```

Result:

Hence, Water level IoT system for sending the real time water level to an application on a software is successfully implemented.