PHASE 5-FLOOD MONITORING AND EARLY WARNING

PROJECT OBJECTIVE:

The project involves deploying IoT sensors near water bodies and flood-prone areas to monitor water levels and provide early flood warnings through a public platform. The objective is to enhance flood preparedness and response by issuing timely warnings to both the public and emergency response teams. This project includes defining objectives, designing the IoT sensor network, developing the warning platform, and integrating them using IOT technology ESP32 board.

Define objectives such as real-time flood monitoring, early warning issuance, public safety, and emergency response coordination

NOW:

IN THIS PART WE WILL FIRST LIKELY TAKE STEPS TO IDENTIFY THE PREINFORMATION ABOUT THE ARRIVAL OF FLOOD PARTICULARY IN WHICH AREA IT IS GOING TO OCCUR OR ARISE .

AND ALSO WE WILL DETERMINE THE STRENGTH OF THE FLOOD AND MONITOR ITS STATUS USING THE ACCURATE SENSORS

IOT SENSOR DEPLOYMENT:

AFTER THE BASIC KNOWLEDGE THE PLACEMENT OF SENSORS MUST BE DECIDED BY US.

ULTRASONIC SENSOR IS USED IN THIS PROJECT TO DETERMINE THE LEVEL OF WATER AND INCATION IS DONE USING A SIMPLE REPLICATOR AS A BUZZER , AND AN ONLINE PLATFORM BLYNK

IS USED INSTEAD OF MOBILE NETWORKING WHICH CONNECTS ALL THE PEOPLE OF THE SOCIETY.

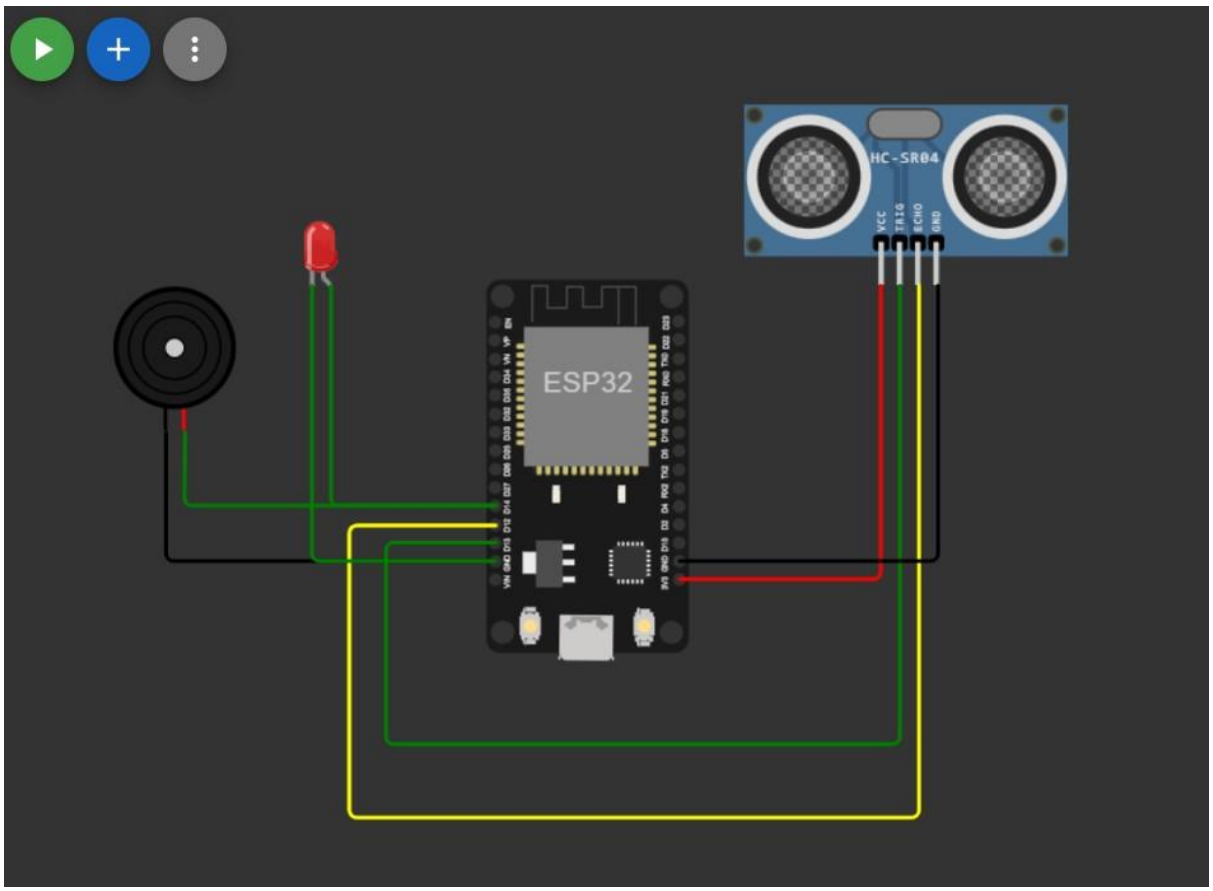
THE SENSOR MUST BE DEPLOYED IN THE DAM TO MEASURE THE LEVEL OF WATER .

CODE IMPLEMENTATION:

For this project we used buzzer led light , esp32 board and ultra sonic sensor .

- Buzzer is used as an indicator to indicate when the water level is increased.
- Led light is to display the visual change of it so that we can see and confirm
- Esp32 this board is used to connect all the components and can able to work using WiFi , it acts as a microcontroller.
- Ultra sonic sensor is used to measure the distance from our project to the water to measure the level of water.

CIRCUIT DIAGAM:



CODE:

```
#define BLYNK_TEMPLATE_ID "TMPL3w-dCObVZ"

#define BLYNK_TEMPLATE_NAME "IOTPhase4"

#define BLYNK_AUTH_TOKEN "i4xhUCb3epG8weF80tEfBppGaOts0SMv"

#include  <WiFi.h>

#include <WiFiClient.h>

#include <BlynkSimpleEsp32.h>

#define trigger_wave 13

#define distance_echo 12

#define buzzer 14
```

```
unsigned int condition=0;
BlynkTimer timer;
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "Wokwi-GUEST";
char pass[] = "";
#define BLYNK_PRINT Serial
void setup()
 {
  Serial.begin(115200);
  pinMode(trigger_wave, OUTPUT);
  pinMode(distance_echo, INPUT);
  pinMode(buzzer,OUTPUT);
  Blynk.begin(auth,ssid,pass,"blynk.cloud",8080);
}
void loop()
{
  //send the wave to measure the length and the capacity of the water
  digitalWrite(trigger_wave, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigger_wave, LOW);
  // receiving wave to judge the water level and hit the buzzer
  int duration = pulseIn(distance_echo, HIGH);
  Serial.print("Distance in feet: ");
  Serial.println((duration / 58)*0.3);
//now I have a value of 400 and the average dam will be  120 feet
//I need to convert my values to 120 so
```

```
condition = (duration / 58)*0.3;
Blynk.virtualWrite(V0,condition);
if(condition >= 90)
{
  digitalWrite(buzzer,HIGH);
  Serial.print("DAM WILL BE OPENED FLOW WARNING!!!!!!!!! ");
  Blynk.virtualWrite(V0,condition);
}
else if (condition<90)
{
  digitalWrite(buzzer,LOW);
}
  delay(1000);
  Blynk.run();
}
```

CODE EXPLANATION:

- THE BLYNK TEMPLATE ID AND ITS NAME IS USED AS PROVIDED BY THE SOURCE

- TO WORK MY PROGRAM THE NECESSARY LIBRARIES ARE AADED IN THE WOKWI SOFTWARE.
    - EG:WIFICLIENT , WIFI,BLYNK SOFTWARES ETC.

- WE ARE DEFINING THE TEMPLATE ID , TEMPLATE NAME , BLYNK AUTHENTICATION TOKEN ETC AS REQUIRED ANS FOR AUTHENTICATION TOKEN .

- **WE USE #INCLUDE KEYWORD TO INCLUDE THE LIBRARIY FILES SO THAT OUR CODE CAN ACTIVAE AND RUN TO PRODUCE OUR REQUIRED FLOOD MONITORING OUTPUT.**

- **WE DEFINE THE PIN NNUMBER FROM ESP32 BOARD WITH THE VARIABLE NAME TRIGGER_WAVE AS PIN 13 AND DISTANCE_ECHO AS PIN 12.**

- **WE ALSO DEFINE A VARIABLE BUZZER FOR PIN 14**

- **WE INITIALIZE INTEGER CODITION AS 0 .**

- **NOW WITH THE CHARATER TYPE WE SET THE AUTH,SSID, PASSWORD FORM OUR PROGRAM TO CONNNCT WITH .**

- **NOW WITHIN THE VOID SETUP SEGMENT.WE PROVIDE THE BAUD RATE FOR OUR PROGRAM WITH THE ESP32 VIRTUAL BOARD.**

- **NOW WE SET THE PIN MODE FOR OUR DECLARED VARIABLES EITHER THE PIN MODE WILL BE INPUT OR OUTPUT. FOR OUR PROGRAM , TRIGGER_WAVE IS SET AS OUTPUT , DISTANCE_ECHO IS SET AS INPUT,BUZZER IS SET TO OUTPUT SINCE WE NEED TO HEAR THE SOUND AS ITS OUTPUT.**

- **LETS START OUR LOOPING PART WHICH RUNS THE PROGRAM CONTINUOUSLY**

- **DIGITALWRITE IN THE CODE SAYS THAT OUR TRIGGER_WAVE IS SET TO HIGH TO SWITH THE PART ON IN ULTRASONIC SENSOR.**

- **DIGITALWRITE IN THE CODE FOR TRIGGER_WAVE IS SET TO LOWTO SWITH THE PART OFF IN ULTRASONIC SENSOR.**

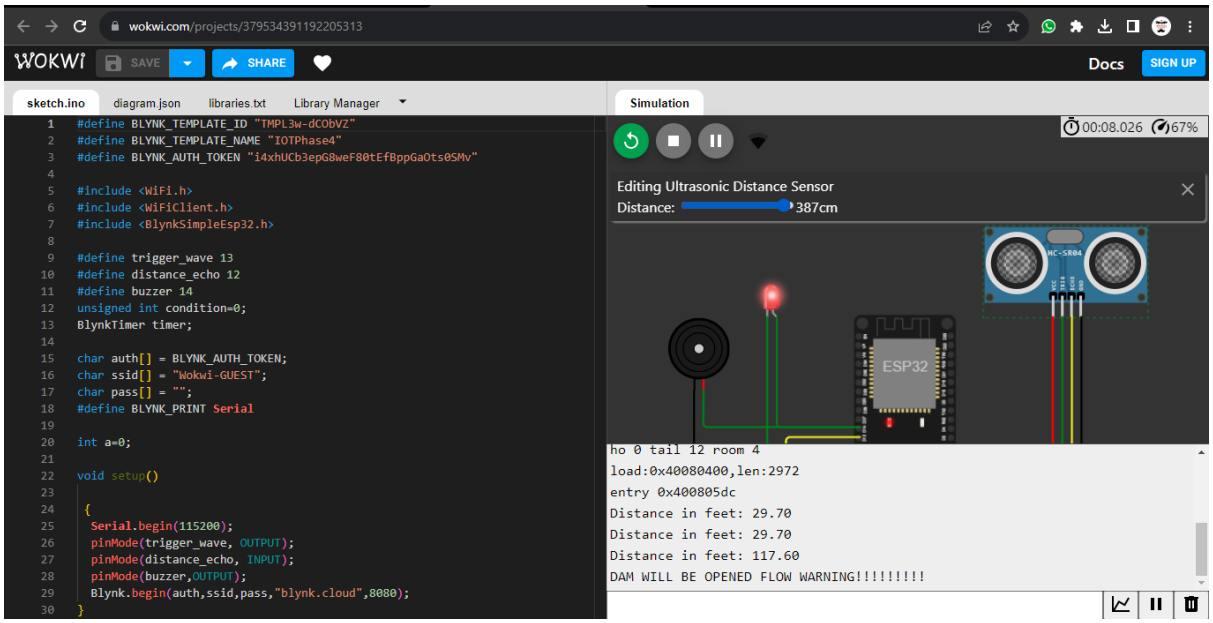
- **NOW THE DISTANCE _ECHO PART IS SET TO HIGH SO THAT IT CAN RECEIVE THE TRIGGER_WAVE SIGNAL WITH THE MEASUREMENT.**

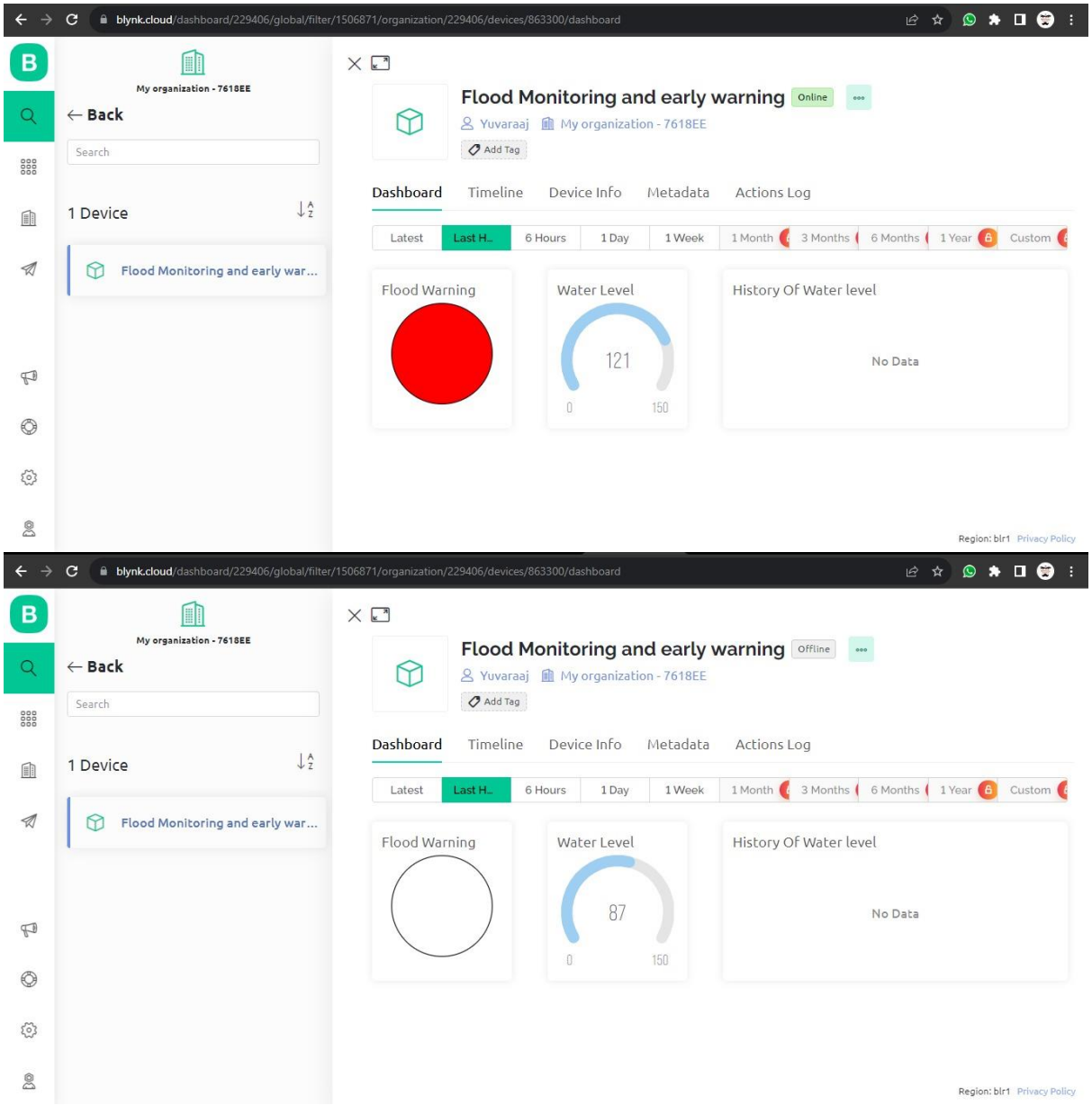
- **NOW WE SET OUR CALCULATION (DURATION/58)*0.3 IN THE VARIABLE CONDITION TO DISPLAY THE OUTPUT AS A RESULT IN PROGRAM AS WELL AS BLYNK PLATFORM.**

- **NOW TO CHECK THE CONDIION THAT IF THE LEVEL IS GREATER THAN 90 THEN WARNING MUST BE GIVEN BY MAKING THE BUZZER ON AND INDICATING THROUGH TEXT.**
- **OR ELSE SIMPLY THE LEVEL OF WATER MUST BE DISPLAYED WITHOUT ANY SIDE OPERATION.**
- **ID THE LEVEL OF WTER IS LESS THEN 90 THEN THE BUZER WILL BE LOW.**

- **BLYNK.VIRTUALWRITE KEYWORD IS USED TO ASSIGN THE PIN IN BLYNK SO THAT IT MAY CALL THE BLYNK PIN TO DISPLAY THE OUTPUT.**

OUTPUT IN BLYNK:





Wokwi Code:

https://wokwi.com/projects/379534391192205313

CONCLUSION:

In conclusion, the project on flood monitoring and early warning systems has been a significant endeavor focused on improving the efficiency, effectiveness, and safeness of the public during floods and heavy rains .Throughout this project, we have analyzed various aspects of the existing data and proposed innovative solutions to improve the prevention of life during floods.