ULTRASONIC SENSOR CODE

Date	03 November
Team ID	PNT2022TMID52636
Project Name	Signs with Smart Connectivity for Better Road Safety
Maximum Marks	4 Marks

```
#include <WiFi.h>
#include <PubSubClient.h>


#define ORG "4i2rfo"
#define DEVICE_TYPE "Traffic_Analyser"
#define DEVICE_ID "Ultrasonic_Sensor"
#define TOKEN "12345678"

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/UltraSonic_Sensor/fmt/json";
char subscribetopic[] = "iot-2/cmd/command/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;

void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);

WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, wifiClient); //calling the predefined client
id by passing parameter like server id,portand wificredential

const int trigPin = 18;
const int echoPin = 5;
long duration;
float distanceCm;
String data3;
int count = 0;
int sec = 0;

void setup()// configureing the ESP32
{
  Serial.begin(115200);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  wificonnect();
```

```
  mqttconnect();
}

void loop()// Recursive Function
{
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);

  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);

  // Calculate the distance
  distanceCm = duration * 0.034/2;

  // Prints the distance in the Serial Monitor
  Serial.print("Distance (cm): ");
  Serial.println(distanceCm);

  if (distanceCm < 210)
  {
    count++;
  }

  delay(2000);
  sec += 2;

  if (sec > 60)
  {
    PublishData(distanceCm, count);
    count = 0;
    sec = 0;
  }

  if (!client.loop()) {
    mqttconnect();
  }
}



/*.....................................retrieving to
Cloud.............................*/

void PublishData(float dist, int count) {
  mqttconnect();//function call for connecting to ibm
  /*
```

```
    creating the String in in form JSon to update the data to ibm cloud
  */
  String payload = "{\"Distance\":";
  payload += dist;
  payload += ",\n";
  payload += "\"Count\":";
  payload += count;
  payload += "}";


  Serial.print("Sending payload: ");
  Serial.println(payload);


  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud
then it will print publish ok in Serial monitor or else it will print publish
failed
  } else {
    Serial.println("Publish failed");
  }

}


void mqttconnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!!!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }
     Serial.println();
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish
the connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
```

```
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
```