ASSIGNMENT-4 DISTANCE DETECTION USING ULTRASONIC SENSOR

Date	08 November 2022
Team ID	PNT2022TMID19722
Maximum Marks	2 Marks

Question1:

Write code and connections in wok wi for ultrasonic sensor. Whenever distance is less than 100 cms send "alert" to ibm cloud and display in device recent events.

CODE:

```
esp32-blink.ino
                 diagram.json •
                                   libraries.txt •
                                                  Library Manager
       pinMode(trig,OUTPUT);
       pinMode(echo,INPUT);
       pinMode(LED, OUTPUT);
       delay(10);
       wificonnect();
       mqttconnect();
       void loop()// Recursive Function
        digitalWrite(trig,LOW);
         digitalWrite(trig,HIGH);
         delayMicroseconds(10);
         digitalWrite(trig,LOW);
         float dur = pulseIn(echo,HIGH);
         float dist = (dur * 0.0343)/2;
         Serial.print ("Distancein cm");
         Serial.println(dist);
         PublishData(dist);
         delay(1000);
         if (!client.loop()) {
           mqttconnect();
       void PublishData(float dist) {
         mqttconnect();//function call for connecting to ibm
```

```
creating the String in in form JSon to update the data to ibm cloud

// */

// String object;
// if (dist <100)

{
    digitalWrite(LED,HJGH);
    Sertal.println("object is near");
    object = "Near";
}

else
{
    digitalWrite(LED,LOW);
    Sertal.println("no object found");
    object = "No";
}

String payload = "(\"distance\":";
    payload += dist;
    payload += dist;
    payload += "," "\"object\":\"";
    payload += "\")";

Sertal.println(payload);

**/
*/**

Sertal.println(payload);

*/**

*/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/**

**/*

**/**

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*

**/*
```

```
esp32-blink.ino
                    diagram.json •
                                    libraries.txt ●
                                                    Library Manager *
          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());
        void initManagedDevice() {
         if (client.subscribe(subscribetopic)) {
            Serial.println((subscribetopic));
            Serial.println("subscribe to cmd OK");
            Serial.println("subscribe to cmd FAILED");
        void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
          Serial.print("callback invoked for topic: ");
          Serial.println(subscribetopic);
 148
          for (int i = 0; i < payloadLength; i++) {</pre>
           data3 += (char)payload[i];
```

```
esp32-blink.ino  diagram.json  libraries.btt  Library Manager  void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)

serial.print("callback invoked for topic: ");

serial.println(subscribetopic);

for (int i = 0; i < payloadLength; i++) {

//Serial.println(char)payload[i];

data3 += (char)payload[i];

}

// Serial.println("data: "+ data3);

// if(data3=="Near")

// // Serial.println(data3);

// digitalwrite(LED,HIGH);

// else

// else

// digitalwrite(LED,LOW);

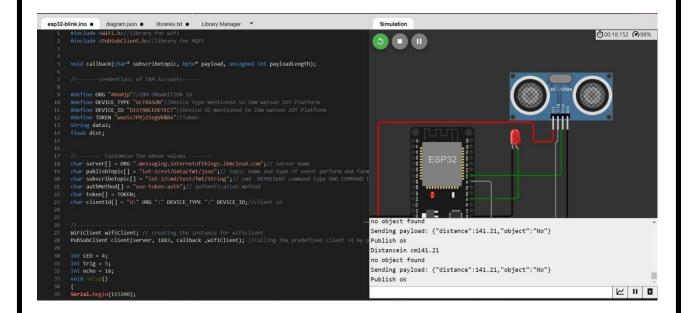
// digitalwrite(LED,LOW);

// digitalwrite(LED,LOW);

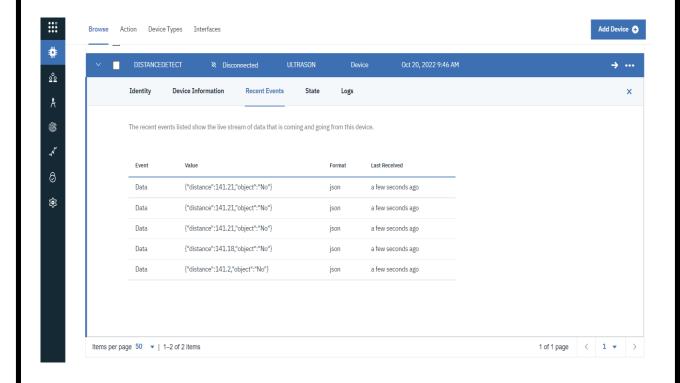
// digitalwrite(LED,LOW);

// digitalwrite(LED,LOW);
```

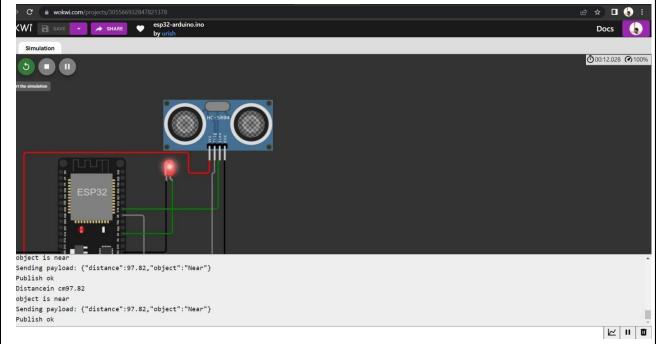
OUTPUT:



Data send to the IBM cloud device when the object is far



when object is near to the ultrasonic sensor



Data sent to the IBM Cloud Device when the object is near

