ASSIGNMENT-4

ULTRASONIC SENSOR SIMULATION IN WOKWI

Assignment Date	21 October 2022
Student Name	ILAMATHI M
Student Roll Number	312319106052
Maximum marks	2 Marks

TASK:

Write code and connections in Wokwi for the ultrasonic sensor.
Whenever the distance is less than 100cms send an “Alert” to the IBM cloud and display in the device recent events.
Upload document with Wokwi share link and images of IBM cloud

CODE:

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//IBM credentials
#define ORG "kapgrv"//IBM ORGANITION ID
#define DEVICE_TYPE "ibm-iot"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "10"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678" //Token

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char subscribetopic[] = "iot-2/cmd/test/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback ,wifiClient);
const int trigPin = 4;
const int echoPin = 2;


void setup() {

  Serial.begin(115200);
  pinMode(echoPin, INPUT);
  pinMode(trigPin, OUTPUT);

  wificonnect();
  mqttconnect();
```

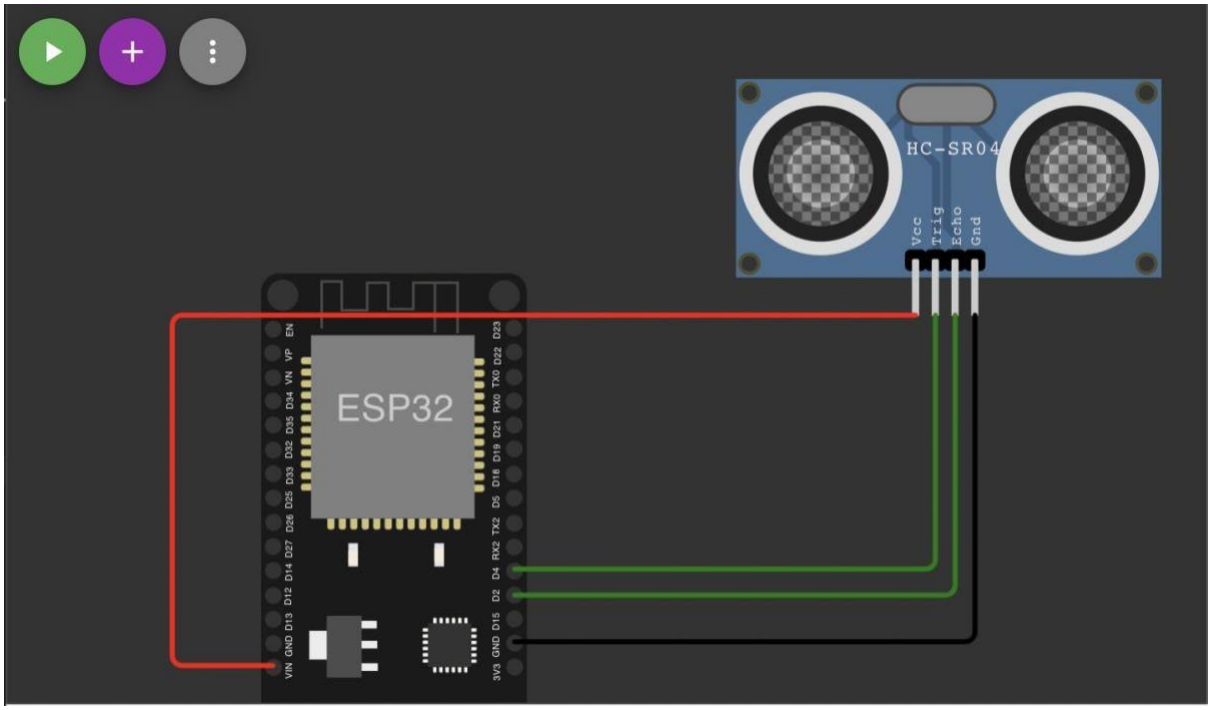
```
30    mqttconnect();
31  }
32
33  void loop() {
34
35    digitalWrite(trigPin, LOW);
36    delayMicroseconds(10);
37    digitalWrite(trigPin, HIGH);
38    delayMicroseconds(10);
39    digitalWrite(trigPin, LOW);
40    float dur = pulseIn(echoPin, HIGH);
41    float dis = (dur*0.0343)/2;
42    Serial.print("Distance from obstacle: ");
43    Serial.println(dis);
44    if(dis < 100) {
45      Serial.println("ALERT!!! Distance less than 100cm...");
46      delay(1000);
47      publishData(dis);
48      delay(1000);
49      if(!client.loop()) {
50        mqttconnect();
51      }
52    }
53
54    delay(1000);
55  }
56
57  void mqttconnect() {
58
59    if (!client.connected()) {
```

```
59    if (!client.connected()) {
60      Serial.print("Reconnecting client to ");
61      Serial.println(server);
62      while (!!!client.connect(clientId, authMethod, token)) {
63        Serial.print(".");
64        delay(500);
65      }
66      initManagedDevice();
67      Serial.println();
68    }
69  }
70
71  void wificonnect()
72  {
73
74    Serial.println();
75    Serial.print("Connecting to ");
76    WiFi.begin("Wokwi-GUEST", "", 6);
77    while (WiFi.status() != WL_CONNECTED) {
78      delay(500);
79      Serial.print(".");
80    }
81    Serial.println("");
82    Serial.println("WiFi connected");
83    Serial.println("IP address: ");
84    Serial.println(WiFi.localIP());
85  }
86
87  void initManagedDevice() {
88
```

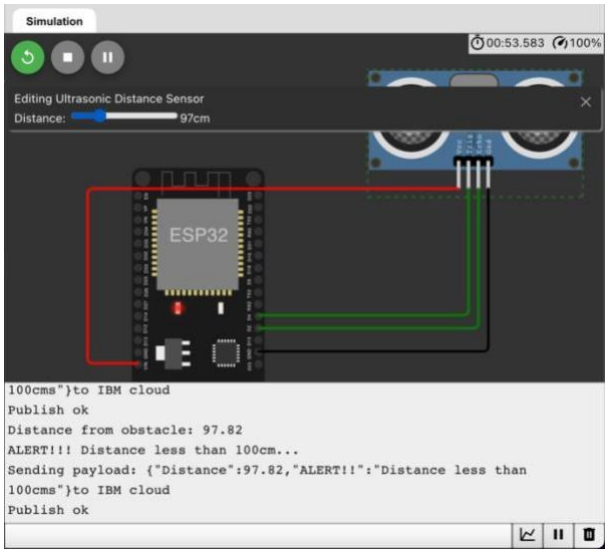
```
88
89    if (client.subscribe(subscribetopic)) {
90      Serial.println((subscribetopic));
91      Serial.println("subscribe to cmd OK");
92    } else {
93      Serial.println("subscribe to cmd FAILED");
94    }
95  }
96
97  void publishData(float distance) {
98
99    mqttconnect();
100   String payload = "{\"Distance\":";
101   payload += distance;
102   payload += ",\"ALERT!!\":""\"Distance less than 100cms\"";
103   payload += "}";
104
105   Serial.print("Sending payload: ");
106   Serial.print(payload);
107   Serial.println("to IBM cloud");
108
109   if (client.publish(publishTopic, (char*) payload.c_str())) {
110     Serial.println("Publish ok");
111   } else {
112     Serial.println("Publish failed");
113   }
114 }
115
116 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
117 {
```

```
117 {
118   String data3;
119   Serial.print("callback invoked for topic: ");
120   Serial.println(subscribetopic);
121   for (int i = 0; i < payloadLength; i++) {
122     data3 += (char)payload[i];
123   }
124   Serial.println("data: "+ data3);
125   data3="";
126 }
```

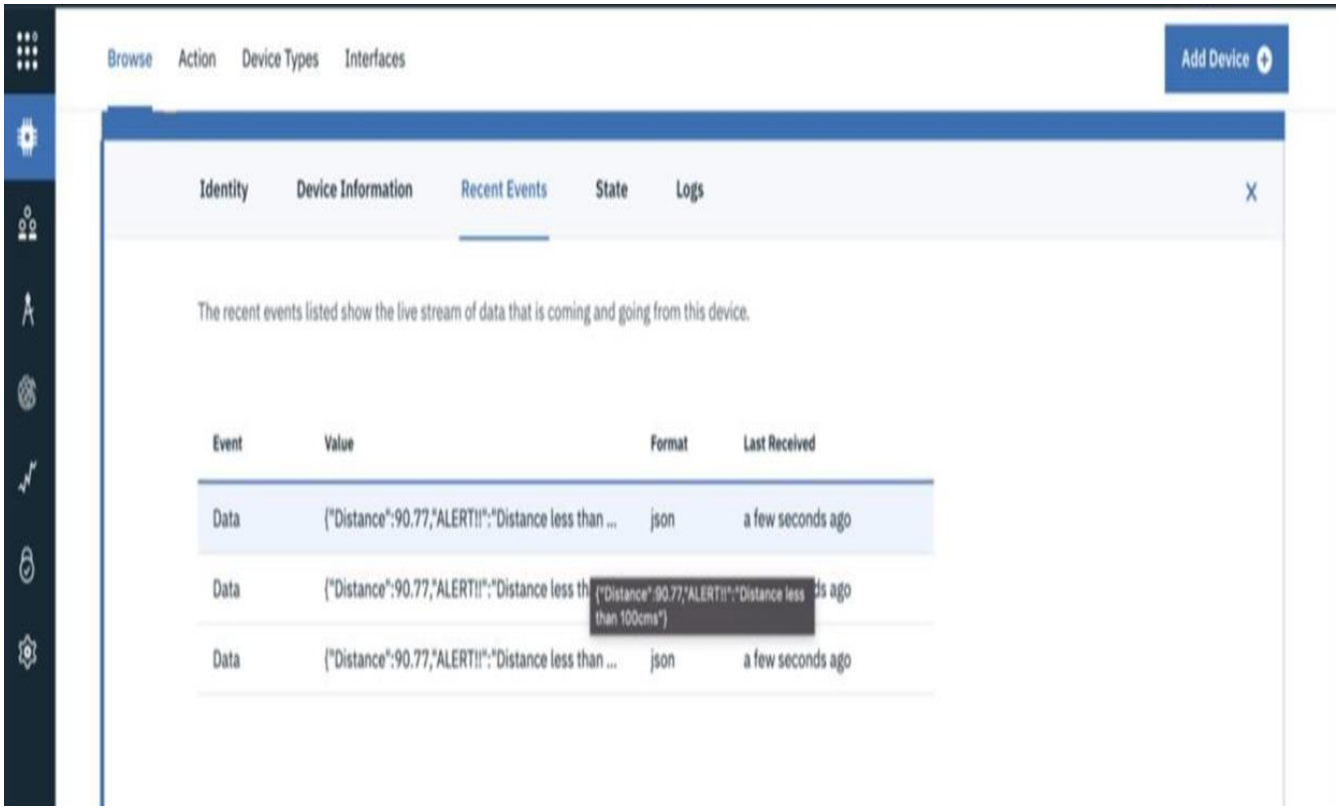
Connections:



WOKWI OUTPUT:



IBM Cloud Output:



WOKWI SIMULATION LINK:
https://wokwi.com/projects/346853412238787156