

#### IBM ASSIGNMENT -04

Assignment Date	06 November 2022
Student Name	SURENTHIRAN V
Student Roll Number	710019106046
Team ID	PNT2022TMID42278

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

#### CODE:

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);
//-----credentials of IBM Accounts-----
#define ORG "n89sjt"//IBM ORGANITION ID
#define DEVICE_TYPE "Suren"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "046"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "surenthiran06" //Token
String data3;
float dist;
String data3;
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 = 5;
const int echoPin = 18;
#define SOUND_SPEED 0.034
long duration;
float distance;
void setup() {
  Serial.begin(115200);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  wificonnect();
  mqttconnect();
}
void loop()
```

```

{
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = duration * SOUND_SPEED/2;
Serial.print("Distance (cm): ");
Serial.println(distance);
if(distance<100)
{
Serial.println("ALERT!!");
delay(1000);
PublishData(distance);
delay(1000);
if (!client.loop()) {
mqttconnect();
}
}
delay(1000);
}
void PublishData(float dist) {
mqttconnect();
String payload = "{\"Distance\":\"";
payload += dist;
payload += "\",\"ALERT!!\":\"\"Distance less than 100cms\"";
payload += "\"}";
Serial.print("Sending payload: ");
Serial.println(payload);

if (client.publish(publishTopic, (char*) payload.c_str())) {
Serial.println("Publish ok");
} 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);
}
}
initManagedDevice();
Serial.println();
}

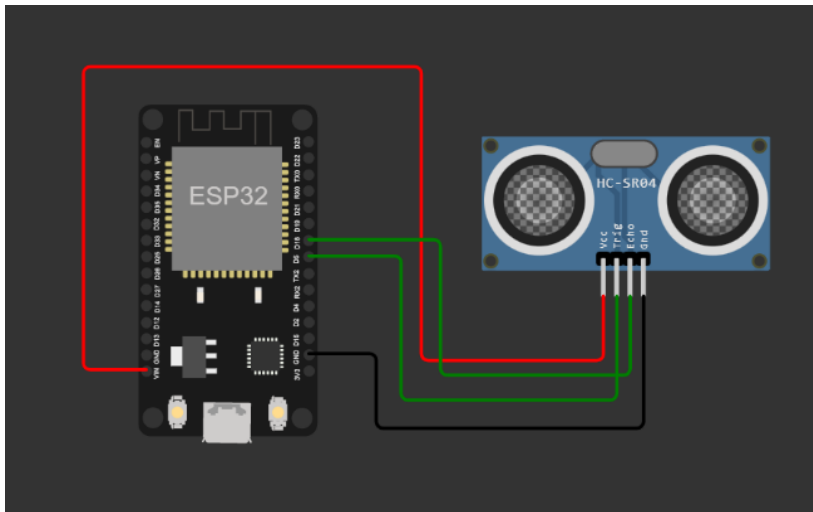
```

```

}
void wificonnect()
{
  Serial.println();
  Serial.print("Connecting to ");
  WiFi.begin("Wokwi-GUEST", "", 6);
  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");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}
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.print((char)payload[i]);
    data3 += (char)payload[i];
  }
  Serial.println("data: "+ data3);
  data3="";
}

```

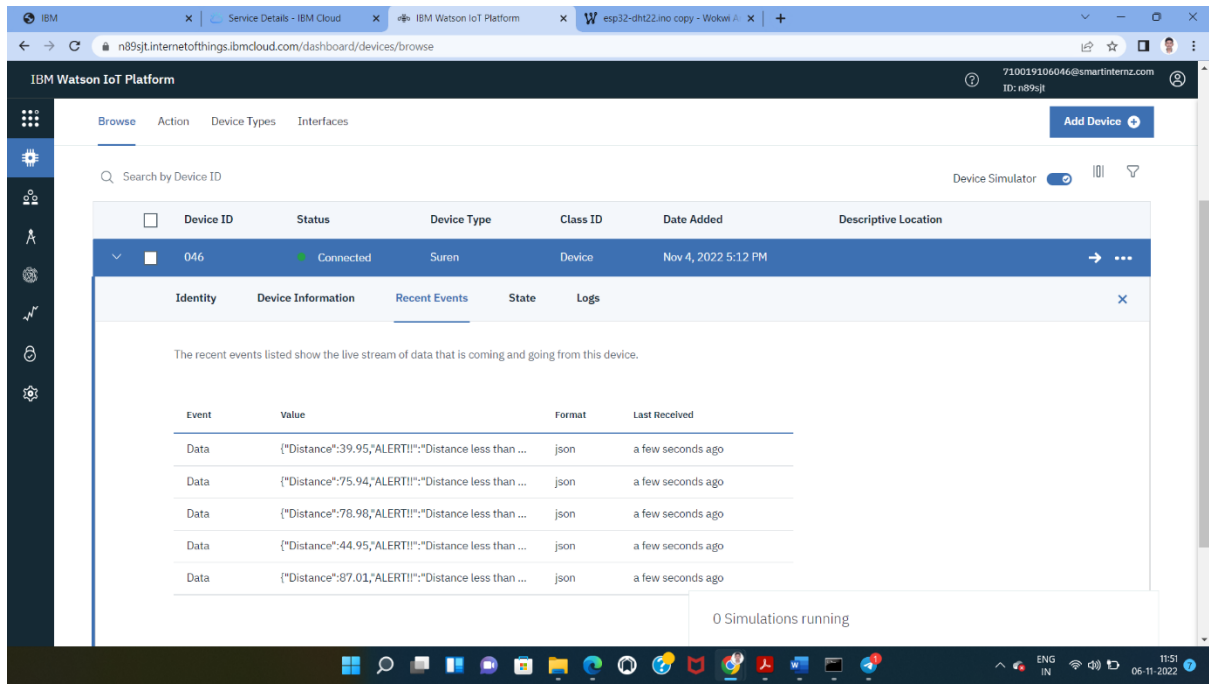
## Schematic Diagram:



## WOKWI OUTPUT:

```
IBM | Service Details - IBM Cloud | IBM Watson IoT Platform | esp32-dht22.ino copy - Wokwi | +
wokwi.com/projects/347557575181468242
WOKWI | SAVE | SHARE | esp32-dht22.ino copy | Docs
esp32-dht22.ino | diagram.json | libraries.txt | Library Manager | Simulation
5 //-----credentials of IBM Accounts-----
6 #define ORG "n89s1t"//IBM ORGANIZATION ID
7 #define DEVICE_TYPE "Suren"//Device type mentioned in ibm watson IoT Platform
8 #define DEVICE_ID "040"//Device ID mentioned in ibm watson IoT Platform
9 #define TOKEN "suren1ran00" //Token
10 String data;
11 float dist;
12 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
13 char publishTopic[] = "iot-2/evt/data/fmt/json";
14 char subscribTopic[] = "iot-2/cmd/test/fmt/String";
15 char authMethod[] = "use-token-auth";
16 char token[] = TOKEN;
17 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
18 WiFiClient wifiClient;
19 PubSubClient client(server, 1883, callback, wifiClient);
20 const int trigPin = 5;
21 const int echoPin = 18;
22 #define SOUND_SPEED 0.034
23 long duration;
24 float distance;
25 void setup() {
26   Serial.begin(115200);
27   pinMode(trigPin, OUTPUT);
28   pinMode(echoPin, INPUT);
29   wifiConnect();
30   mqttConnect();
31 }
32 void loop() {
33   digitalWrite(trigPin, LOW);
34   delayMicroseconds(2);
35   digitalWrite(trigPin, HIGH);
36   delayMicroseconds(10);
37   digitalWrite(trigPin, LOW);
38   duration = pulseIn(echoPin, HIGH);
39   dist = (duration * 0.034) / 2;
40   if (dist < 100) {
41     Serial.println("Distance: " + String(dist, 2) + "cm");
42     client.publish(publishTopic, String(dist, 2) + "cm");
43     client.publish(subscribTopic, "Distance less than 100cms");
44     Serial.println("Distance (cm): " + String(dist, 2));
45     Serial.println("ALERT!!");
46     client.publish(subscribTopic, "Distance less than 100cms");
47     Serial.println("Sending payload: {"Distance":44.95,\"ALERT!!\":\"Distance less than 100cms\"}");
48     Serial.println("Publish ok");
49     dist = 78.98;
50     client.publish(subscribTopic, "Distance: 78.98cm");
51     Serial.println("Distance (cm): 78.98");
52     Serial.println("ALERT!!");
53     client.publish(subscribTopic, "Distance: 78.98cm");
54     Serial.println("Sending payload: {"Distance":78.98,\"ALERT!!\":\"Distance less than 100cms\"}");
55     Serial.println("Publish ok");
56     dist = 109.99;
57     client.publish(subscribTopic, "Distance: 109.99cm");
58     Serial.println("Distance (cm): 109.99");
59     Serial.println("ALERT!!");
60     client.publish(subscribTopic, "Distance: 109.99cm");
61     Serial.println("Sending payload: {"Distance":109.99,\"ALERT!!\":\"Distance less than 100cms\"}");
62     Serial.println("Publish ok");
63     dist = 109.99;
64     client.publish(subscribTopic, "Distance: 109.99cm");
65     Serial.println("Distance (cm): 109.99");
66     Serial.println("ALERT!!");
67     client.publish(subscribTopic, "Distance: 109.99cm");
68     Serial.println("Sending payload: {"Distance":109.99,\"ALERT!!\":\"Distance less than 100cms\"}");
69     Serial.println("Publish ok");
70     dist = 75.94;
71     client.publish(subscribTopic, "Distance: 75.94cm");
72     Serial.println("Distance (cm): 75.94");
73     Serial.println("ALERT!!");
74     client.publish(subscribTopic, "Distance: 75.94cm");
75     Serial.println("Sending payload: {"Distance":75.94,\"ALERT!!\":\"Distance less than 100cms\"}");
76     Serial.println("Publish ok");
77     dist = 39.95;
78     client.publish(subscribTopic, "Distance: 39.95cm");
79     Serial.println("Distance (cm): 39.95");
80     Serial.println("ALERT!!");
81     client.publish(subscribTopic, "Distance: 39.95cm");
82     Serial.println("Sending payload: {"Distance":39.95,\"ALERT!!\":\"Distance less than 100cms\"}");
83     Serial.println("Publish ok");
84   }
85 }
```

## IBM CLOUD OUTPUT:



The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes tabs for 'Service Details - IBM Cloud', 'IBM Watson IoT Platform', and 'Wokwi'. The main dashboard shows a list of devices, with one device selected: 'Suren' (Device ID: 046). The device status is 'Connected'. Below the device list, the 'Recent Events' tab is active, showing a stream of data events. The events are listed in a table with columns: Event, Value, Format, and Last Received. The events are JSON payloads containing distance and alert information. The bottom of the screen shows a Windows taskbar with various application icons and a system tray indicating the time as 11:51 on 06-11-2022.

Event	Value	Format	Last Received
Data	{"Distance":39.95,"ALERT!":"Distance less than ...	json	a few seconds ago
Data	{"Distance":75.94,"ALERT!":"Distance less than ...	json	a few seconds ago
Data	{"Distance":78.98,"ALERT!":"Distance less than ...	json	a few seconds ago
Data	{"Distance":44.95,"ALERT!":"Distance less than ...	json	a few seconds ago
Data	{"Distance":87.01,"ALERT!":"Distance less than ...	json	a few seconds ago

0 Simulations running

## WOKWI LINK:

<https://wokwi.com/projects/347557575181468242>