

ASSIGNMENT 4

Date	30 October 2022
Register number	110319106002
Student Name	ARAVINDHAN A

QUESTION:

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 "oa9ck1"//IBM ORGANITION ID
#define DEVICE_TYPE "ardino"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "a02"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "110319106002" //Token
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());
}

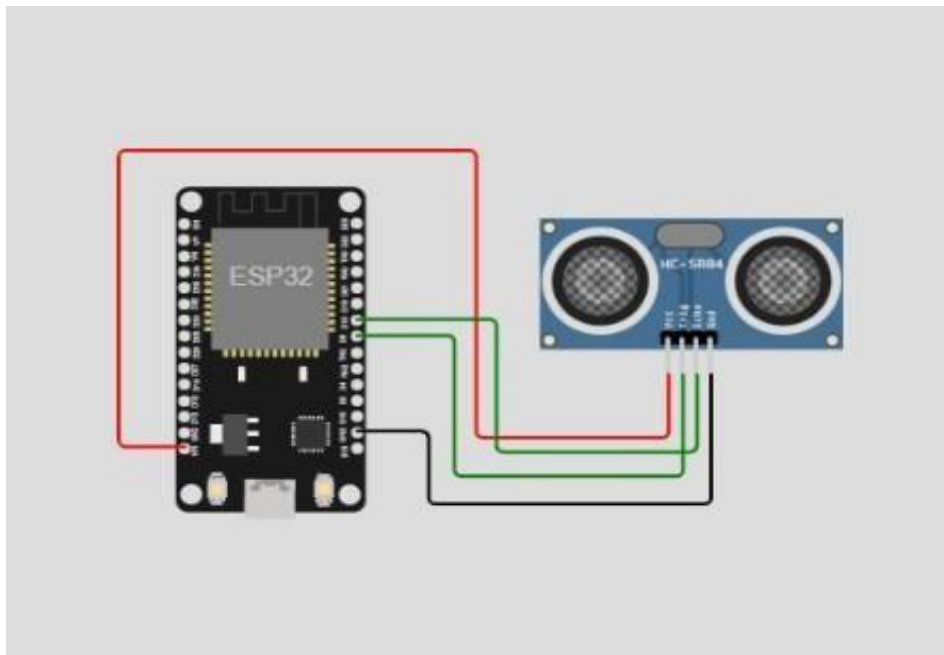
```

```

}
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/CIRCUIT DIAGRAM:



OUTPUT DIAGRAM:

The image shows a Wokwi simulation environment. At the top, there's a 'Simulation' tab and a toolbar with a green refresh button, a grey stop button, and a yellow play button. In the top right corner, a timer shows '00:04.864' and a refresh icon with '101%'. The main area displays an ESP32 microcontroller board connected to an HC-SR04 ultrasonic sensor. The wiring is as follows: a red wire connects the sensor's VCC pin to the ESP32's VCC pin; a black wire connects the sensor's GND pin to the ESP32's GND pin; a green wire connects the sensor's Trig pin to the ESP32's GPIO4 pin; and another green wire connects the sensor's Echo pin to the ESP32's GPIO5 pin. Below the simulation area is a terminal window with the following text:

```
Reconnecting client to qkg8nh.messaging.internetofthings.ibmcloud.com
iot-2/cmd/test/fmt/String
subscribe to cmd OK

Distance (cm): 399.96
Distance (cm): 399.94
Distance (cm): 399.94
```

At the bottom right of the terminal, there are icons for a graph, a pause button, and a trash can.

WOKWI LINK:

<https://wokwi.com/projects/348561023932826195>

IBM CLOUD OUTPUT:

Identity

Device Information

Recent Events

State

Logs

X

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
Data	{"Distance":96.99,"ALERT!!":"Distance less than ...	json	a few seconds ago

Items per page 50 | 1-1 of 1 item

1 of 1 page

<

1

>