

Assignment -4
WOWKI SIMULATION

Assignment Date	4th NOVEMBER 2022
Student Name	Aiswarya.S
Student Roll Number	960219106011
Maximum Marks	2 Marks

Question-1:

Write a code and make a connection in WOKWI for ultrasonic sensor. Whenever distance is less than 100 , send “alert” to IBM cloud and display in device recent events.

PROGRAM

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient;
String data3;
#define ORG "2t8zs3"
#define DEVICE_TYPE "b11m3e-device" #define
DEVICE_ID "11111111deviceid"
#define TOKEN "2*kzBuumHxd+BeL*H)"
#define speed 0.034 #define led 14 char server[] = ORG
".messaging.internetofthings.ibmcloud.com"; char publishTopic[] =
"iot-2/evt/Amrin/fmt/json"; char topic[] = "iot-
2/cmd/led/fmt/String"; char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);
```

```
const int trigpin=5; const
int echopin=18;
String command;
String data="";
```

```
long duration; float
dist;
```

```
void setup()
```

```

{
  Serial.begin(115200);
  pinMode(led, OUTPUT);
  pinMode(trigpin, OUTPUT);
  pinMode(echopin, INPUT);
  wifiConnect(); mqttConnect();
}

void loop() {  bool isNearby
= dist < 100;
digitalWrite(led, isNearby);

  publishData();
  delay(500);

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

void wifiConnect() {
  Serial.print("Connecting to "); Serial.print("Wifi");
  WiFi.begin("Wokwi-GUEST", "", 6); while
(WiFi.status() != WL_CONNECTED) { delay(500);
  Serial.print(".");
}
  Serial.print("WiFi connected, IP address: "); Serial.println(WiFi.localIP());
}

void mqttConnect() { if
(!client.connected()) {
  Serial.print("Reconnecting MQTT client to "); Serial.println(server);
  while (!client.connect(clientId, authMethod, token)) {
    Serial.print("."); delay(500);
  }
  initManagedDevice();
  Serial.println();
}
}

void initManagedDevice() {
if (client.subscribe(topic)) {
//
Serial.println(client.subscribe
(topic));
  Serial.println("IBM subscribe to cmd OK");
}
}

```

```

    } else {
        Serial.println("subscribe to cmd FAILED");
    }
}

void publishData()
{
    digitalWrite(trigpin,LOW);
    digitalWrite(trigpin,HIGH);
    delayMicroseconds(10);
    digitalWrite(trigpin,LOW);
    duration=pulseIn(echopin,HIGH);
    dist=duration*speed/2;  if(dist<100){
        String payload = "{\"Alert Distance\":\"";
        payload += dist;    payload += "\"}";

        Serial.print("\n");
        Serial.print("Sending payload: ");
        Serial.println(payload);
        if (client.publish(publishTopic, (char*) payload.c_str())) {
            Serial.println("Publish OK");
        }

    }

    if(dist>100){
        String payload = "{\"Distance\":\"";
        payload += dist;    payload += "\"}";

        Serial.print("\n");
        Serial.print("Sending payload: ");
        Serial.println(payload);
        if(client.publish(publishTopic, (char*) payload.c_str())) {
            Serial.println("Publish OK");
        }else {
            Serial.println("Publish FAILED");
        }

    }

}

}

```

OUTPUT:

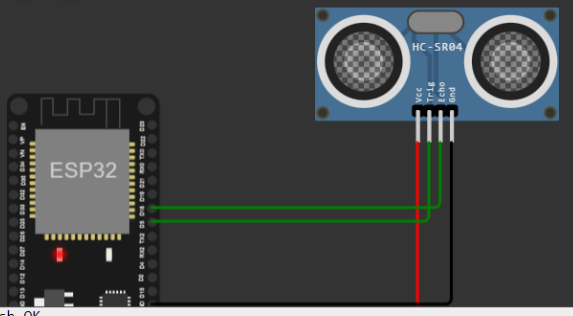
WOKWI SIMULATION

esp32-blink.ino • diagram.json • libraries.txt • Library Manager

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 WiFiClient wifiClient;
4 String data3;
5 #define ORG "dgohie"
6 #define DEVICE_TYPE "PNT2022TMD34083"
7 #define DEVICE_ID "PNTID34083"
8 #define TOKEN "X5ZfHwFVJ35rwDmqS3v"
9 #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/Aiswarya/fmt/json";
13 char topic[] = "iot-2/cmd/led/fmt/String";
14 char authMethod[] = "use-token-auth";
15 char token[] = TOKEN;
16 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server, 1883, wifiClient);
18
19
20 const int trigpin=5;
21 const int echopin=18;
22 String command;
23 String data="";
24
25 long duration;
26 float dist;
27
28 void setup()
29 {
30   Serial.begin(115200);
```

Simulation

00:11.649 100%

A simulation diagram showing an ESP32 microcontroller connected to an HC-SR04 ultrasonic sensor. The ESP32 is a black component with gold pins. The HC-SR04 is a blue component with two circular sensors. Wires connect the ESP32's pins to the sensor's pins: a red wire from the sensor's VCC pin to the ESP32's VCC pin, a black wire from the sensor's GND pin to the ESP32's GND pin, a green wire from the sensor's Trig pin to the ESP32's pin 5, and a blue wire from the sensor's Echo pin to the ESP32's pin 18.

Publish OK

Sending payload: {"Alert Distance":47.96}

Publish OK

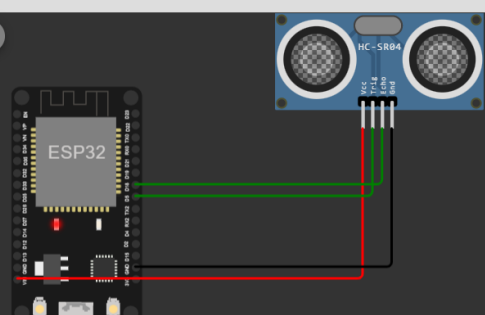
Sending payload: {"Alert Distance":47.96}

Publish OK

When distance<100:

Simulation

04:59.400 97%

A simulation diagram showing an ESP32 microcontroller connected to an HC-SR04 ultrasonic sensor. The ESP32 is a black component with gold pins. The HC-SR04 is a blue component with two circular sensors. Wires connect the ESP32's pins to the sensor's pins: a red wire from the sensor's VCC pin to the ESP32's VCC pin, a black wire from the sensor's GND pin to the ESP32's GND pin, a green wire from the sensor's Trig pin to the ESP32's pin 5, and a blue wire from the sensor's Echo pin to the ESP32's pin 18.

Publish OK

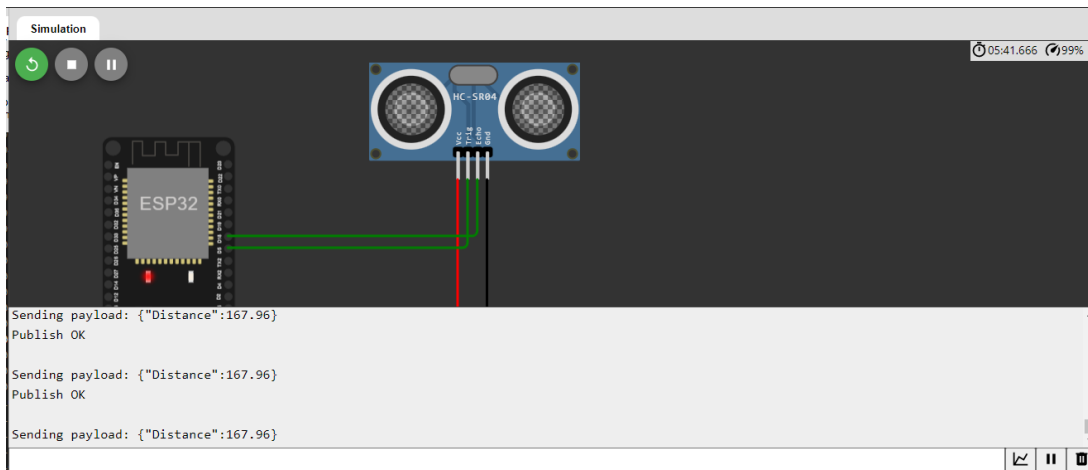
Sending payload: {"Alert Distance":47.96}

Publish OK

Sending payload: {"Alert Distance":47.96}

Publish OK

When distance>100:



IBM CLOUD OUTPUT

IBM Watson IoT Platform

Browse Action Device Types Interfaces

Device ID	Status	Device Type	Class ID	Date Added
PNTID34083	Connected	PNT2022TMID34083	Device	30 Oct 2022 08:24

Identity Device Information **Recent Events** State Logs

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

Event	Value	Format	Last Received
Aiswarya	{"Alert Distance":47.97}	json	a few seconds ago
Aiswarya	{"Alert Distance":47.96}	json	a few seconds ago
Aiswarya	{"Alert Distance":47.97}	json	a few seconds ago
Aiswarya	{"Alert Distance":47.96}	json	a few seconds ago
Aiswarya	{"Alert Distance":47.96}	json	a few seconds ago

WOKWI LINK:

<https://wokwi.com/projects/347414869145813586>