

## Assignment-4

Date	24 October 2022
Name	Vasanthakumar S
Roll Number	620119106102
Team ID	PNT2022TMID30936
Project Name	IoT Based Safety Gadget for Child Safety Monitoring and Notification.

### Question :

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

Upload document with wokwi share link and images.

### Wokwi:

<https://wokwi.com/projects/348105244660466260>

### Code:

```
#include <WiFi.h>
#include <PubSubClient.h>

WiFiClient wifiClient;

#define ORG "jk50ow"
#define DEVICE_TYPE "vasanth"
#define DEVICE_ID "kumar"
#define TOKEN "lLtNyXXt7Tb4wV?Bg?"
#define speed 0.034

char server[] = ORG".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/status1/fmt/json"; char
topic[] = "iot-2/cmd/home/fmt/String"; char authMethod[] =
"use-token-auth"; char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient); void
publishData(); const int trigpin=5; const int echopin=18;
String command;
String data=""; long
duration; float
dist;
void
setup()
{
```

```

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

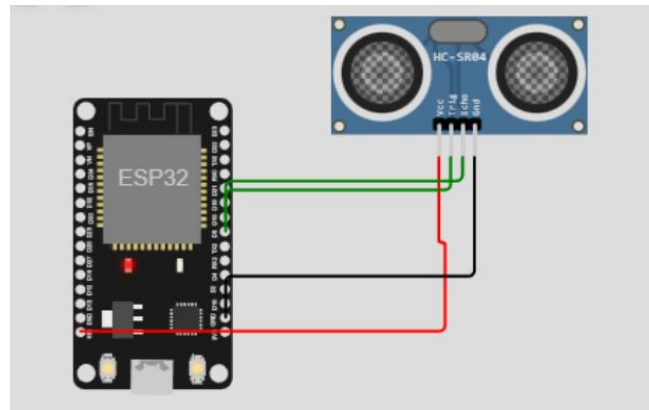
```

```

    } else
    {
        Serial.println("Publish FAILED");
    }
}
}

```

Diagram:



Wokwi Output:

Wokwi simulation interface showing the sketch code and the simulation output.

**Sketch Code:**

```

1 #include <WiFi.h>
2 #include <PubSubClient.h>
3
4 WiFiClient wifiClient;
5
6 #define ORG "jk50ow"
7 #define DEVICE_TYPE "vasanth"
8 #define DEVICE_ID "kumar"
9 #define TOKEN "lthlyXXt7Tb4wV?Bg?"
10 #define speed 0.034
11
12
13 char server[] = ORG".messaging.internetofthings.ibmcloud.com";
14 char publishTopic[] = "iot-2/evt/event_1/fmt/json";
15 char topic[] = "iot-2/cmd/home/fmt/String";
16 char authMethod[] = "use-token-auth";
17 char token[] = TOKEN;
18 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
19 PubSubClient client(server, 1883, wifiClient);
20 void publishData();
21 const int trigpin=5;
22 const int echopin=18;
23 String command;
24 String data="";
25 long duration;
26 float dist;
27 void setup()
28 {
29   Serial.begin(115200);
30   pinMode(trigpin, OUTPUT);

```

**Simulation Output:**

```

Sending payload: {"Alert distance":99.99}
Publish OK

Sending payload: {"Alert distance":99.98}
Publish OK
Reconnecting MQTT client to
jk50ow.messaging.internetofthings.ibmcloud.com

```

IBM Watson IoT Platform

jk50ow.internetofthings.ibmcloud.com/dashboard/devices/browse

vasanthbe473@gmail.com  
ID: jk50ow

Browse Action Device Types Interfaces

All Devices Diagnose

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Search by Device ID

Device Simulator ☒

<input type="checkbox"/>	Device ID	Status	Device Type	Class ID
> <input type="checkbox"/>	kumar	<span style="color: green;">●</span> Connected	vasanth	Device

Items per page 50 | 1-1 of 1 item

1 of 1 page

## IBM cloud output:

IBM Watson IoT Platform

jk50ow.internetofthings.ibmcloud.com/dashboard/devices/browse

vasanthbe473@gmail.com  
ID: jk50ow

Browse Action Device Types Interfaces

Add Device

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
event_1	{"Alert distance":99.98}	json	a few seconds ago
event_1	{"Alert distance":99.98}	json	a few seconds ago
event_1	{"Alert distance":99.98}	json	a few seconds ago
event_1	{"Alert distance":99.98}	json	a few seconds ago
event_1	{"Alert distance":99.99}	json	a few seconds ago

0 Simulations running

Items per page 50 | 1-1 of 1 item