

ASSIGNMENT-4

Student Name : R.HEMALATHA

BATCH NO : B1-1M3E

ASSIGNMENT 4:

Write code and connections in wokwi for ultrasonic sensors.

Whenever distance is less than 100cms send "alert" to IBM cloud and display in device recent events.

Upload document with wokwi share link and images of IBM cloud

CODE:

```
#include<WiFi.h>#include
<PubSubClient.h>WiFiClien
twifiClient;

#defineORG "o1z9pr"
#define DEVICE_TYPE
"raspberrypi"#define DEVICE_ID
"USE YOUR ID"#defineTOKEN"USE
YOURTOKEN"
#define speed0.034

char server[] =
ORG".messaging.internetofthings.ibmcloud.com";char publi
shTopic[] = "iot-
2/evt/raspberrypi_1/fmt/json";chartopic[]="iot-
2/cmd/home/fmt/String"; char
authMethod[] = "use-
tokenauth";chartoken[] =
TOKEN;
charclientId[]="d:"ORG":"DEVIC
E_TYPE": "DEVICE_ID;
PubSubClient client(server,
1883,
wifiClient);voidpublishData();
```

```

const int
  trigpin=5;constintechop
  in=18;
  Stringcommand;
  Stringdata="";
long
  duration;float
dist;
voidsetup()
{
  Serial.begin(115200);pinMode(tr
igpin, OUTPUT);pinMode(echopin,
INPUT);wifiConnect();
mqttConnect();
}
voidloop() {

  publishData();
  delay(500);
  if (!client.loop())
  {mqttConnect();
  }
}
voidwifiConnect(){
  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());}

voidmqttConnect(){ 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();
  }
}
voidinitManagedDevice(){
  if(client.subscribe(topic)){
    //Serial.println(client.subscribe(topic));
    Serial.println("subscribetocmdOK");
  }
}

```

```

}
else
{
    Serial.println("subscribetocmdFAILED");
}
}
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 = "{\"Alertdistance\": ";
    payload += dist; payload +=
    "\"}";

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

```

CONNECTIONS:

The screenshot shows the IBM Watson IoT Platform interface. The top navigation bar includes tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains icons for various platform features. The main content area displays a list of devices, with one device selected and its details expanded.

Device List:

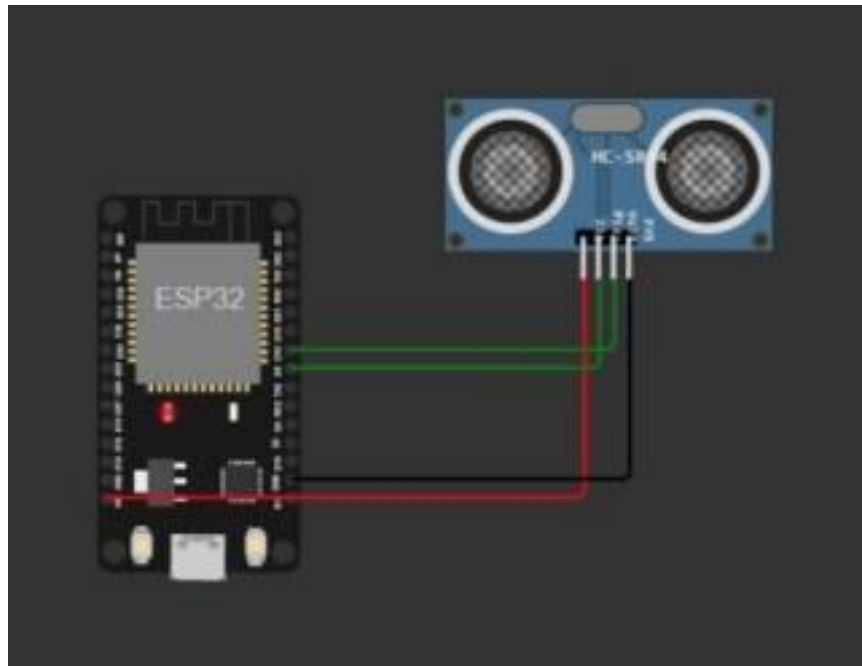
ID	Name	Status	Type	Location	Last Received
12345	raspberrypi	Disconnected	Device		Oct 6, 2022 8:02 PM
raspberrypi_1	raspberrypi	Connected	Device		Oct 24, 2022 3:57 PM

Device Details (raspberrypi_1):

The 'Recent Events' tab is active, showing a live stream of data from the device. The events are listed in a table:

Event	Value	Format	Last Received
event_1	["Alert distance":10]	json	a few seconds ago
event_1	["Alert distance":27]	json	a few seconds ago
event_1	["Alert distance":49]	json	a few seconds ago
event_1	["Alert distance":57]	json	a few seconds ago
event_1	["Alert distance":87]	json	a few seconds ago

At the bottom of the interface, a status bar indicates '9 Simulations running'.



OUTPUT :

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include <ArduinoJson.h>
4
5 WiFiClient wifiClient;
6
7 #define ORG "ol29pr"
8 #define DEVICE_TYPE "raspberrypi"
9 #define DEVICE_ID "12345"
10 #define TOKEN "12345678"
11 #define speed 0.034
12
13 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
14 char publishTopic[] = "iot-2/evt/raspberrypi1/evt/json";
15 char topic[] = "iot-2/cmd/home/evt/String";
16 char authMethod[] = "use-token-auth";
17 char token[] = TOKEN;
18 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
19 PubSubClient client(server, 8883, wifiClient);
20 void publishData();
21
22 const int trigPin=5;
23 const int echoPin=18;
24 String command;
25 String data="";
26 String lat="14.167589";
27 String lon="98.248518";
28 String name="point2";
29 String icon="";
```

Connecting to Wifi...Wifi connected, IP address: 10.10.0.2
Reconnecting MQTT client to
ol29pr.messaging.internetofthings.ibmcloud.com
1
subscribe to cmd OK