

**Assignment-4**  
DistanceDetectionUsingUltrasonicSensor

AssignmentDate	19October2022
Student Name	Mr.Surya V
StudentRollNumber	MECR19IT028
MaximumMarks	4 Marks

**Question-1:**

Writecodeandconnections inwokwiforultrasonicsensor.Wheneverdistanceislessthan100cms send "alert" to ibm cloud and display in device recent events.

WOKWILINK: <https://wokwi.com/projects/345964118720643668>

**CODE:**

```
#include <WiFi.h> //library for
wifi#include<PubSubClient.h> //libraryforMQT
t

voidcallback(char*subscribetopic,byte*payload,unsignedint
payloadLength);

//-----credentialsofIBMAccounts----- #define

ORG "f59trs" //IBM ORGANITION ID
#defineDEVICE_TYPE"ultrasonicsensor" //Devicetypementionedin
ibmwatsonIOTPlatform
#defineDEVICE_ID"distancedetection" //DeviceIDmentionedinibmwatson
IOT Platform
#defineTOKEN "AlGMGaaF01nawa1QA3" //Token
String data3;
floatdist;

//-----Customisetheabovevalues-----
charserver[]=ORG".messaging.internetofthings.ibmcloud.com";//
Server Name
charpublishTopic[]="iot-2/evt/Data/fmt/json";//topicnameand type
of event perform and format in which data to be send
charsubscribetopic[]="iot-2/cmd/test/fmt/String";//
cmdREPRESENTcommandtypeANDCOMMANDISTESTOFFORMATSTRING
charauthMethod[]="use-token-auth";//authenticationmethod char
token[] = TOKEN;
charclientId[]="d:ORG": "DEVICE_TYPE": "DEVICE_ID";//client id

//
WiFiClientwifiClient;//creatingtheinstanceforwificlient
```

```

PubSubClientclient(server,1883,callback,wifiClient);
//callingthepredefinedclientidbypassingparameterlike server
id,portandwificredential

int LED = 4;
int trig = 5;
intecho=18;
void setup()
{
  Serial.begin(115200);
  pinMode(trig,OUTPUT);
  pinMode(echo,INPUT);
  pinMode(LED,OUTPUT);
  delay(10);
  wificonnect();
  mqttconnect();
}
voidloop()//RecursiveFunction
{

  digitalWrite(trig,LOW);
  digitalWrite(trig,HIGH);
  delayMicroseconds(10);
  digitalWrite(trig,LOW);
  floatdur=pulseIn(echo,HIGH); float
  dist= (dur * 0.0343)/2;
  Serial.print("Distanceincm");
  Serial.println(dist);

  PublishData(dist);
  delay(1000);
  if(!client.loop()){
    mqttconnect();
  }
}

/*.....retrievingto
Cloud. .... */

void PublishData(float dist) {
  mqttconnect();//functioncallforconnectingtoibm
  /*
    creatingtheStringininformJSontoupdatethedatatato ibm cloud
  */
  Stringobject;

```

```

if(dist<100)
{
    digitalWrite(LED,HIGH);
    Serial.println("objectisnear");
    object = "Near";
}
else
{
    digitalWrite(LED,LOW);
    Serial.println("noobjectfound"); object
    = "No";
}

Stringpayload="{\"distance\":";
payload += dist;
payload+=",\"\"object\\\":\\\"";
payload += object;
payload+="\"}";

Serial.print("Sendingpayload:");
Serial.println(payload);

if(client.publish(publishTopic,(char*)payload.c_str())){
    Serial.println("Publishok");//ifitsuccessfullyuploaddata on the
cloud then it will print publish ok in Serial monitor or else it
will print publish failed
}else{
    Serial.println("Publishfailed");
}
}

voidmqttconnect(){
    if (!client.connected()) {
        Serial.print("Reconnectingclientto");
        Serial.println(server);
        while(!!!client.connect(clientId,authMethod,token)){
            Serial.print(".");
            delay(500);
        }

        initManagedDevice();
        Serial.println();
    }
}
}

```

```

void wificonnect() //function definition for wificonnect
{
    Serial.println();
    Serial.print("Connecting to");

    WiFi.begin("Wokwi-GUEST","",6); //passing the wifi credentials to
    establish the connection
    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("subscribetocmdOK");
    } else {
        Serial.println("subscribetocmdFAILED");
    }
}

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);
    // if(data3=="Near")
    // {
    // Serial.println(data3);
    // digitalWrite(LED,HIGH);

    // }

    // else
    // {
    // Serial.println(data3);

```

```

//digitalWrite(LED,LOW);

//    }
data3="";

}

```

**OUTPUT:**

**When object is not near to the ultrasonic sensor**

The screenshot shows the Wokwi IoT simulator interface. On the left is the code editor for a sketch.ino file, and on the right is the simulation window showing a circuit diagram and a console log.

**Code (sketch.ino):**

```

1  #include <WiFi.h> //library for wifi
2  #include <PubSubClient.h> //library for MQTT
3
4
5  void callback(char* subscribtopic, byte* payload, unsigned int payloadLength);
6
7  //-----credentials of IBM Accounts-----
8
9  #define ORG "f59trs" //IBM ORGANIZATION ID
10 #define DEVICE_TYPE "ultrasonicsensor" //Device type mentioned in ibm watson IOT Platform
11 #define DEVICE_ID "distancedetection" //Device ID mentioned in ibm watson IOT Platform
12 #define TOKEN "ALGMGaaF01nawa1QA3" //Token
13 String data3;
14 float dist;
15
16
17 //----- Customise the above values -----
18 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // Server Name
19 char publishTopic[] = "iot-2/evt/Data/fmt/json"; // topic name and type of event perform and
20 char subscribtopic[] = "iot-2/cmd/test/fmt/String"; // cmd REPRESENT command type AND COMM
21 char authMethod[] = "use-token-auth"; // authentication method
22 char token[] = TOKEN;
23 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
24
25
26 //-----
27 WiFiClient wificlient; // creating the instance for wificlient
28 PubSubClient client(server, 1883, callback, wificlient); //calling the predefined client id
29
30 int LED = 4;
31 int trig = 5;
32 int echo = 18;
33 void setup()

```

**Simulation Window:**

The simulation window shows a circuit diagram with an ESP32 microcontroller, an HC-SR04 ultrasonic sensor, and an LED. The sensor is connected to the ESP32 via I2C (VCC to 5V, GND to GND, Echo to D18, Trig to D5). The LED is connected to the ESP32 via digital pins (LED to D4, Trig to D5). The simulation is running, and the console log shows the following output:

```

no object found
Sending payload: {"distance":403.45,"object":"No"}
Publish ok
Distance in cm 233.00
no object found
Sending payload: {"distance":233.00,"object":"No"}
Publish ok

```

# Data sent to the IBM cloud device when the object is far

The screenshot shows the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains various icons for navigation. The main content area displays details for a device named 'distancedetection', which is 'Connected'. The 'Recent Events' tab is active, showing a table of data events. The table has columns for 'Event', 'Value', 'Format', and 'Last Received'. All events are 'Data' type with a JSON value containing distance and object information. The 'Last Received' column indicates 'a few seconds ago' for each entry. At the bottom, it shows '0 Simulations running' and 'Items per page 50 | 1-1 of 1 item'.

IBM Watson IoT Platform

mdabdullahmpt@gmail.com  
ID: f59trs

Browse Action Device Types Interfaces

distancedetection Connected ultrasonicsensor Device Oct 19, 2022 11:56 AM

Identity Device Information Recent Events State Logs

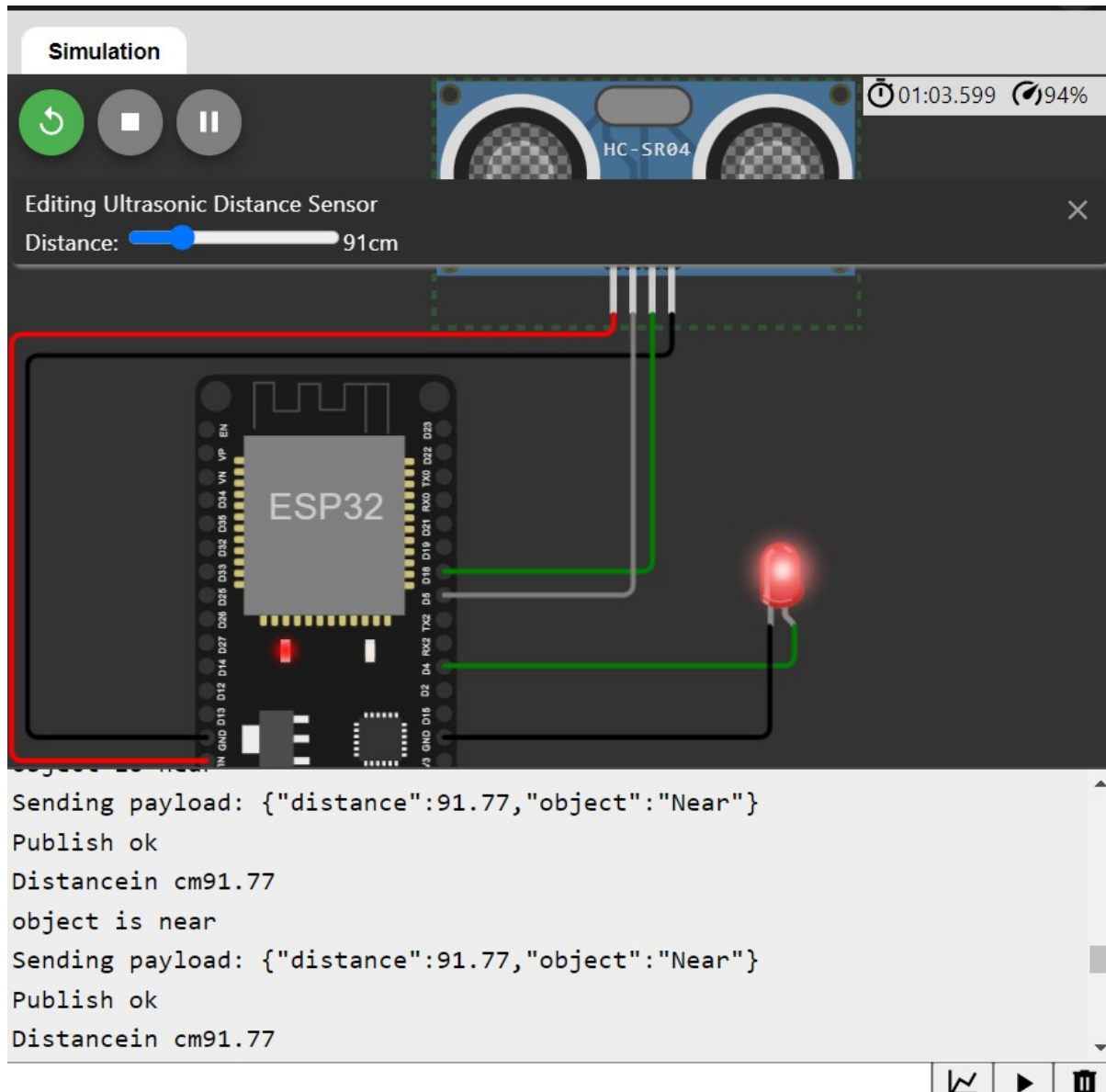
The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
Data	{"distance":235.02,"object":"No"}	json	a few seconds ago
Data	{"distance":235.02,"object":"No"}	json	a few seconds ago
Data	{"distance":235.02,"object":"No"}	json	a few seconds ago
Data	{"distance":235.02,"object":"No"}	json	a few seconds ago
Data	{"distance":235.02,"object":"No"}	json	a few seconds ago

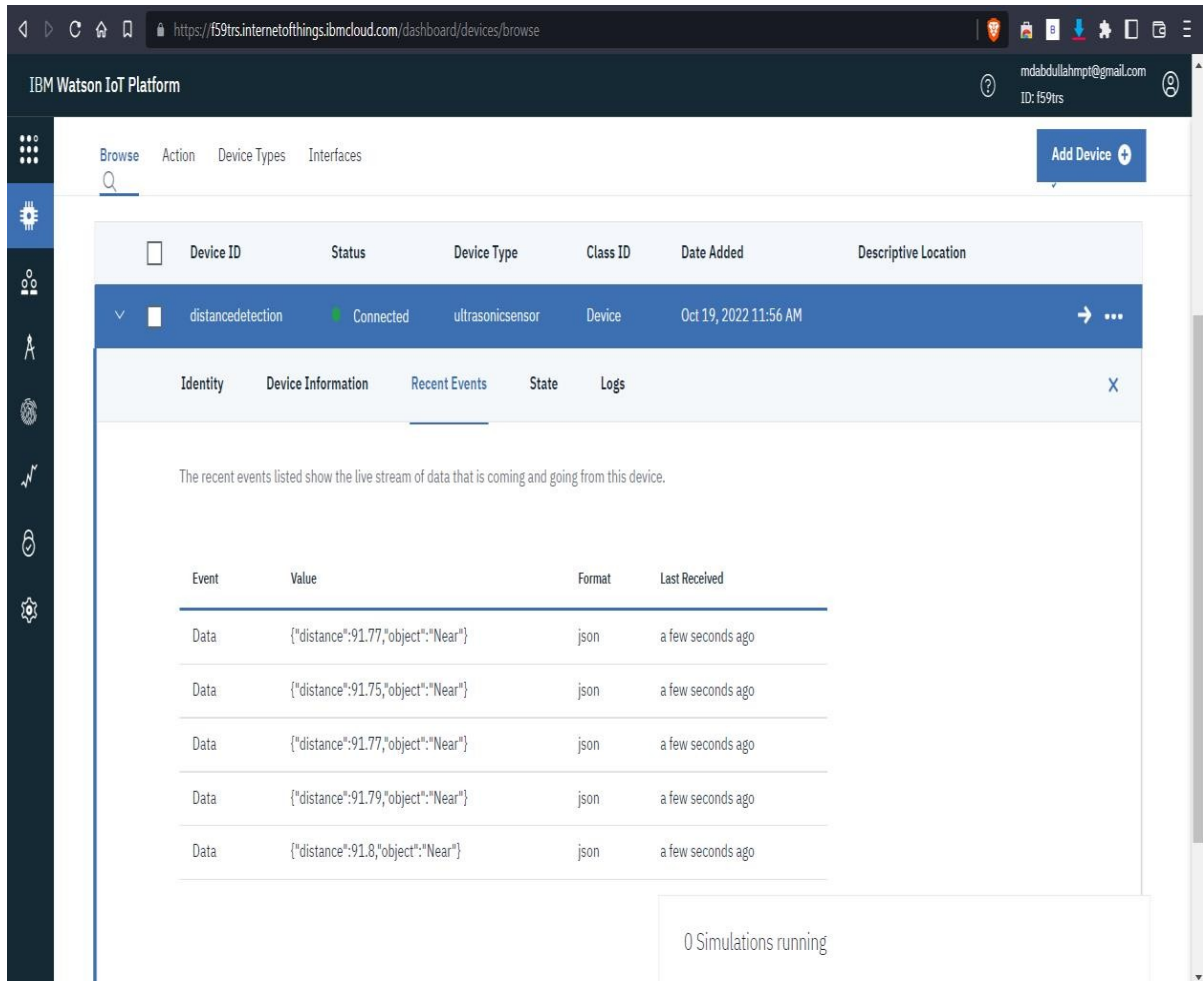
0 Simulations running

Items per page 50 | 1-1 of 1 item

## When object is near to the ultrasonic sensor



# Data sent to the IBM cloud device when the object is near



IBM Watson IoT Platform

mdabdullahmp@gmail.com  
ID: f59trs

Browse Action Device Types Interfaces

Add Device

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
distancedetection	Connected	ultrasonicsensor	Device	Oct 19, 2022 11:56 AM	

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
Data	{"distance":91.77,"object":"Near"}	json	a few seconds ago
Data	{"distance":91.75,"object":"Near"}	json	a few seconds ago
Data	{"distance":91.77,"object":"Near"}	json	a few seconds ago
Data	{"distance":91.79,"object":"Near"}	json	a few seconds ago
Data	{"distance":91.8,"object":"Near"}	json	a few seconds ago

0 Simulations running

<https://wokwi.com/projects/345964118720643668>



