# Assignment-4

DistanceDetectionUsingUltrasonicSensor

|  |  |
| --- | --- |
| AssignmentDate | 19October2022 |
| StudentName | Mr. Gopinath R.H |
| StudentRollNumber | 111919106016 |
| MaximumMarks | 2 Marks |

# Question-1:

Write code and connections in wokwi for ultrasonic sensor. Whenever distance is less than 100 cmssend"alert" toibmcloudanddisplay indevicerecentevents.

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

|  |
| --- |
| **CODE:** |
| #include<WiFi.h>//library forwifi#include<PubSubClient.h>//libraryforMQtt  voidcallback(char\*subscribetopic,byte\*payload,unsignedintpayloadLength);  //-------credentialsofIBMAccounts------  #defineORG"f59trs"//IBMORGANITIONID  #defineDEVICE\_TYPE"ultrasonicsensor"//DevicetypementionedinibmwatsonIOTPlatform  #defineDEVICE\_ID"distancedetection"//DeviceIDmentionedinibmwatsonIOTPlatform  #defineTOKEN"AlGMGaaF01nawa1QA3" //TokenStringdata3;  floatdist;  //--------Customisetheabovevalues--------  char server[] = ORG ".messaging.internetofthings.ibmcloud.com";//ServerName  charpublishTopic[]="iot-2/evt/Data/fmt/json";//topicnameandtypeofevent perform andformatinwhich datatobesend  charsubscribetopic[]="iot-2/cmd/test/fmt/String";//  cmdREPRESENTcommandtypeANDCOMMANDIS TESTOFFORMATSTRING  charauthMethod[]="use-token-auth";//authenticationmethodchartoken[]=TOKEN;  char clientId[] = "d:" ORG ":" DEVICE\_TYPE ":" DEVICE\_ID;//clientid  // -  WiFiClientwifiClient;//creating theinstanceforwificlient |

PubSubClientclient(server,1883,callback,wifiClient);

//calling the predefined client id by passing parameter likeserverid,portandwificredential

int LED = 4;int trig = 5;int echo = 18;voidsetup()

{

**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);

float dur = pulseIn(echo,HIGH);float dist = (dur \* 0.0343)/2;**Serial**.print ("Distancein cm");**Serial**.println(dist);

PublishData(dist);delay(1000);

if (!client.loop()) {mqttconnect();

}

}

/\*.....................................retrievingto

Cloud. \*/

void PublishData(float dist) {mqttconnect();//functioncallforconnectingtoibm

/\*

creating the String in in form JSon to update the data toibmcloud

\*/

Stringobject;

if(dist<100)

{

digitalWrite(LED,HIGH);**Serial**.println("object is near");object="Near";

}

else

{

digitalWrite(LED,LOW);**Serial**.println("no object found");object="No";

}

String payload = "{\"distance\":";payload+=dist;

payload += "," "\"object\":\"";payload+=object;

payload+= "\"}";

**Serial**.print("Sendingpayload:");

**Serial**.println(payload);

if(client.publish(publishTopic,(char\*)payload.c\_str())){

**Serial**.println("Publish ok");// if it sucessfully upload dataon the cloud then it will print publish ok in Serial monitor orelseitwill printpublishfailed

}else{

**Serial**.println("Publishfailed");

}

}

voidmqttconnect(){

if (!client.connected()) {**Serial**.print("Reconnecting client to ");**Serial**.println(server);

while(!!!client.connect(clientId,authMethod,token)){

**Serial**.print(".");delay(500);

}

initManagedDevice();

**Serial**.println();

}

}

voidwificonnect()//functiondefinationforwificonnect

{

**Serial**.println();**Serial**.print("Connectingto");

WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentialstoestablish theconnection

while (WiFi.status() != WL\_CONNECTED) {delay(500);

**Serial**.print(".");

}

**Serial**.println("");**Serial**.println("WiFi connected");**Serial**.println("IP address: ");**Serial**.println(WiFi.localIP());

}

voidinitManagedDevice(){

if (client.subscribe(subscribetopic)) {**Serial**.println((subscribetopic));**Serial**.println("subscribetocmdOK");

}else{

**Serial**.println("subscribetocmdFAILED");

}

}

void callback(char\* subscribetopic, byte\* payload, unsigned intpayloadLength)

{

**Serial**.print("callbackinvokedfortopic:");

**Serial**.println(subscribetopic);

for(inti=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);

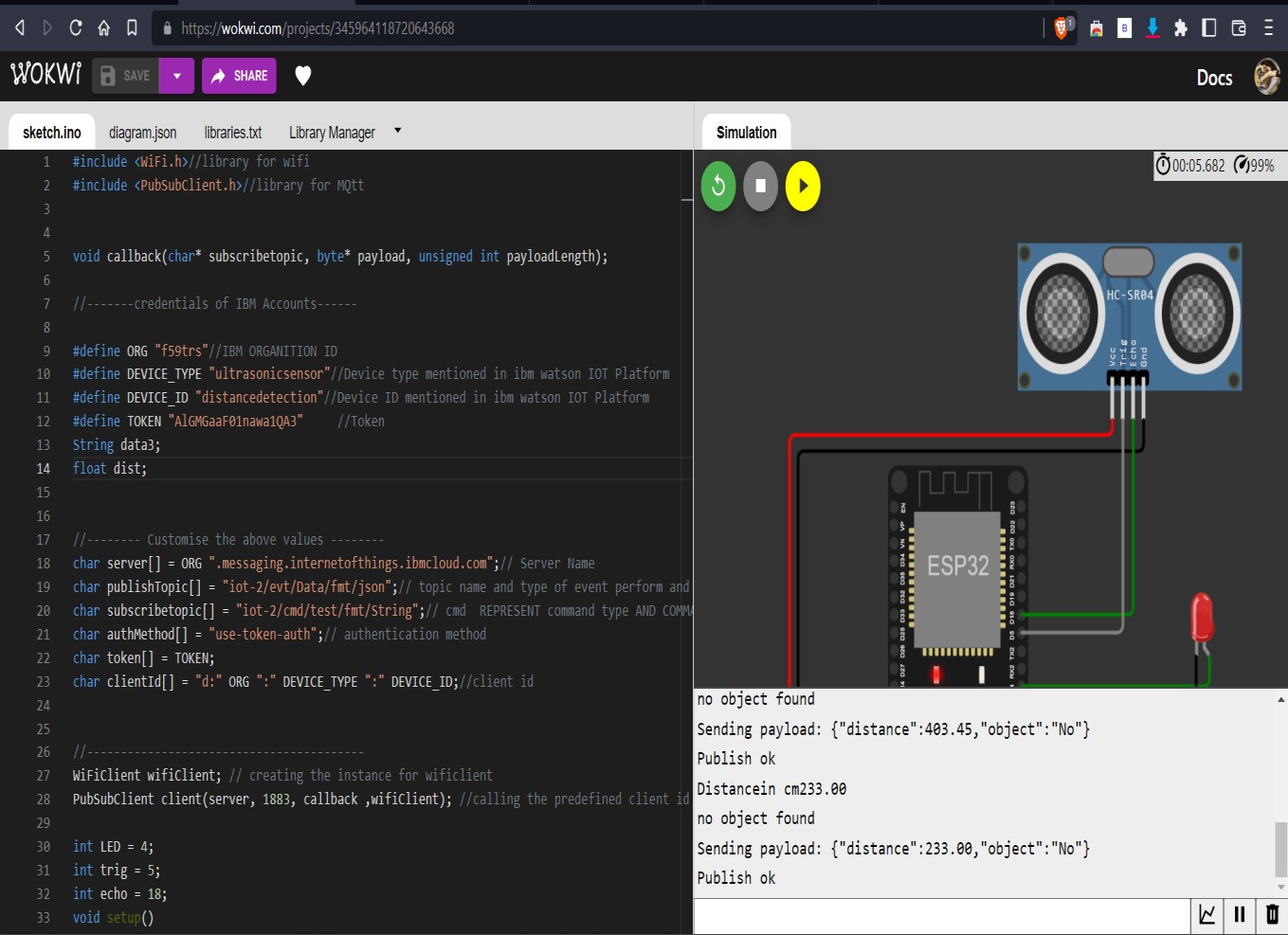
//digitalWrite(LED,LOW);

// }data3="";

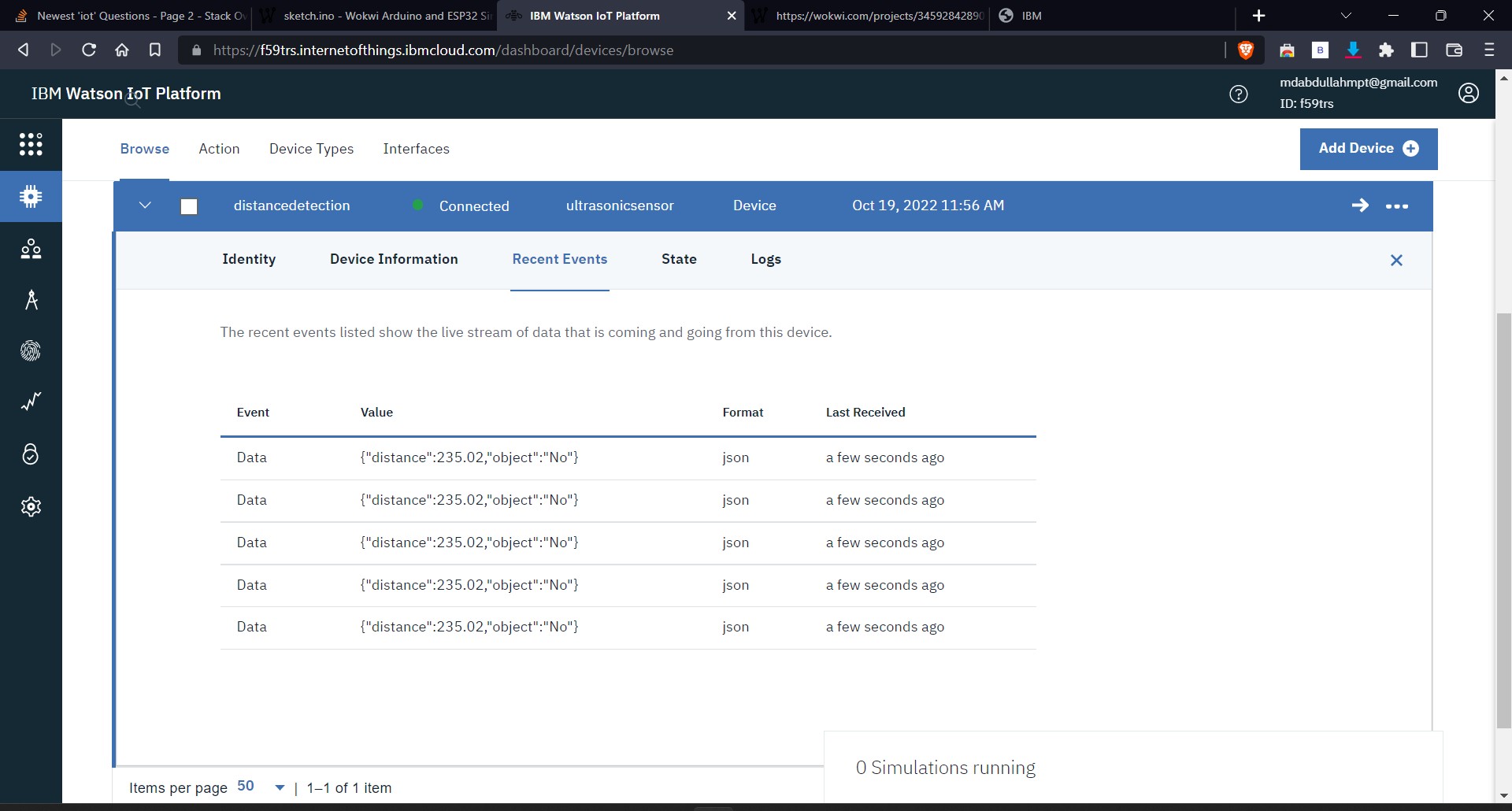
}

**OUTPUT:**

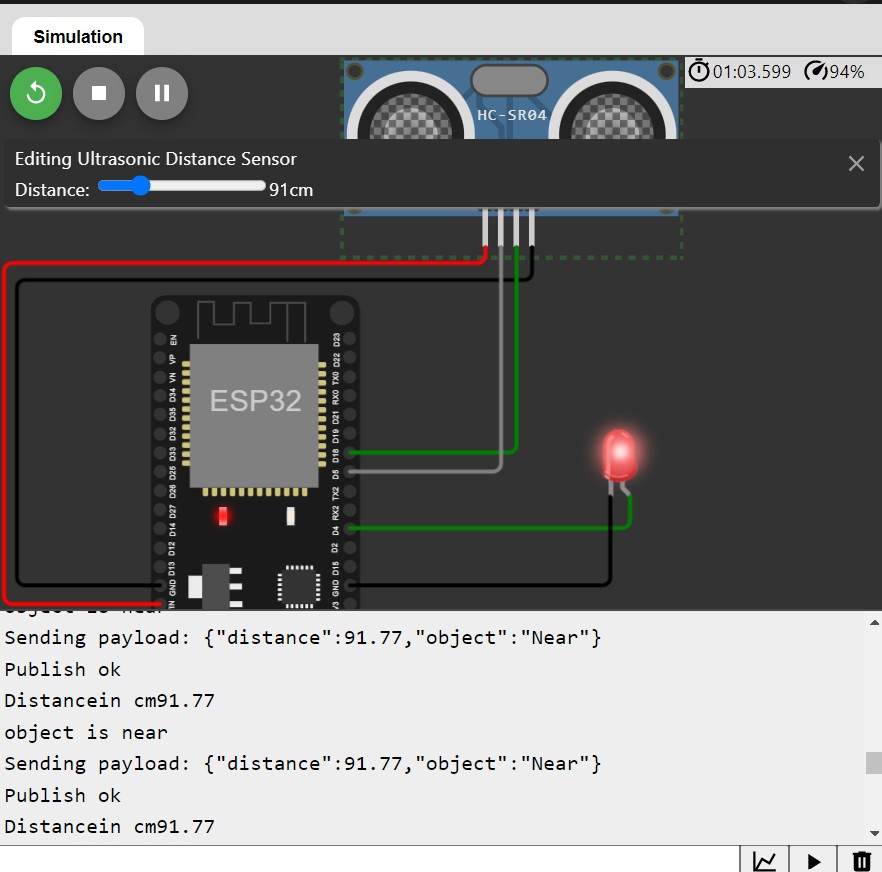
**Whenobjectisnotneartotheultrasonicsensor**



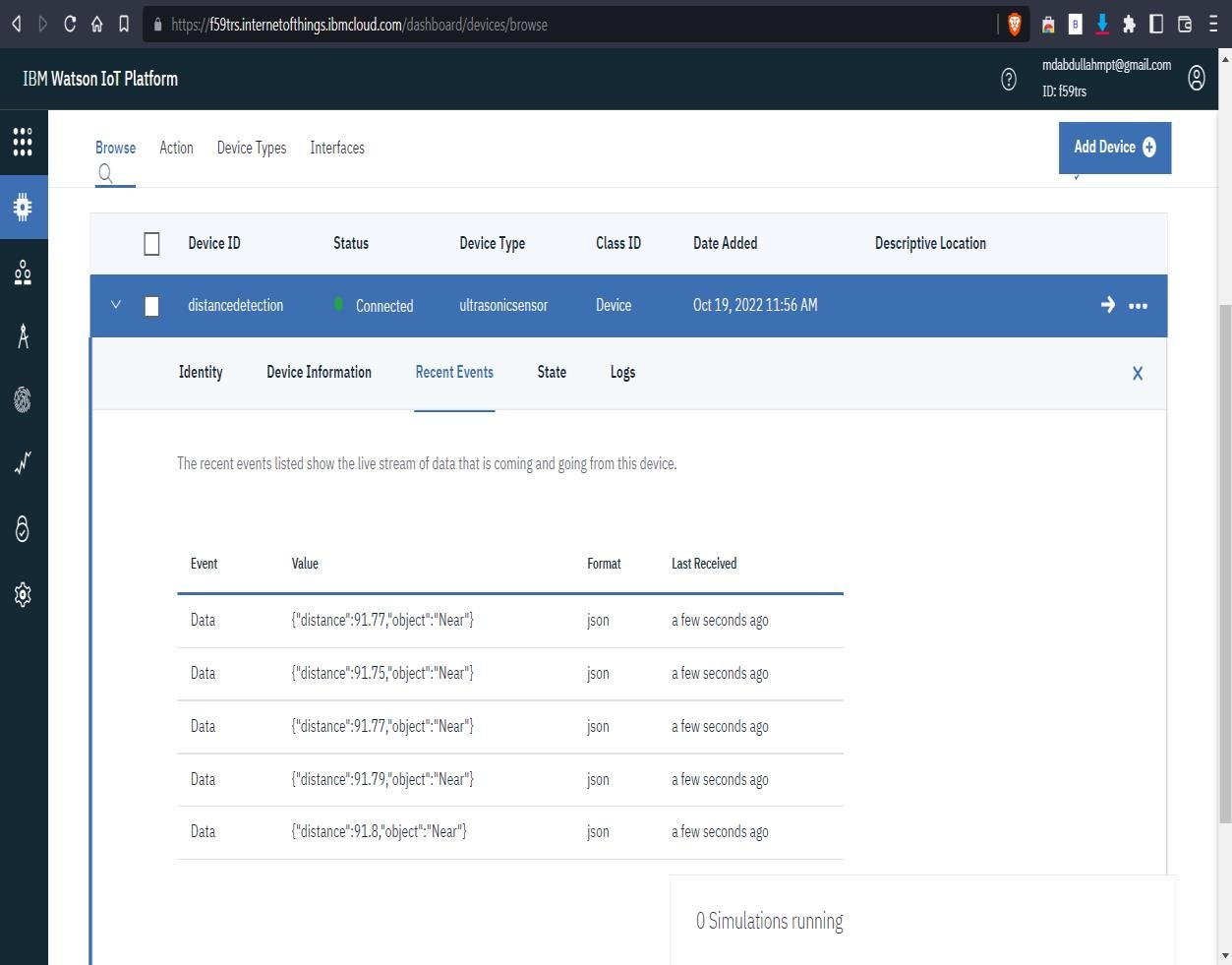
**DatasenttotheIBMclouddevice whentheobjectis far**



**Whenobjectisnearertotheultrasonicsensor**



**DatasenttotheIBMclouddevice whentheobjectisnear**



https://wokwi.com/projects/345964118720643668