# ASSIGNMENT-4

|  |  |
| --- | --- |
| **Date** | 24.09.2022 |
| **TeamID** | PNT2022TMID41904 |
| **ProjectName** | IoT Based Safety Gadget for Child Safety Monitoring & Notification |

**Question:**

Write code and connections in wokwi for the ultrasonic sensor. Whenever the distance is lessthan100 cmssend an"alert"totheIBMcloud anddisplayinthedevicerecent events.

Uploaddocumentwith wokwisharelink andimagesofIBMcloud.

# Code:

#include<WiFi.h>#include <WiFiClient.h>#include <PubSubClient.h>const int trigPin = 5;constintechoPin= 18;

//define sound speed in cm/uS#defineSpeed 0.034

#definecm\_to\_inch0.393701longduration;

float distance;floatdistanceInch;

void callback(char\* subscribetopic, byte\* payload, unsignedintpayloadLength);

//-------credentialsofIBMAccounts------

#defineORG" 937txh"//IBMORGANITIONID

#defineDEVICE\_TYPE"ultrasonic"//DevicetypementionedinibmwatsonIOTPlatform

#defineDEVICE\_ID"ultrasonicsensor"//DeviceIDmentionedinibmwatsonIOTPlatform

#defineTOKEN"1234567890" //TokenStringdata3;

//--------Customisetheabovevalues--------

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Namechar publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type ofeventperform andformatinwhich datato besend

char subscribetopic[] = "iot-2/cmd/test/fmt/String";// cmdREPRESENT commandtypeANDCOMMANDISTESTOFFORMAT STRING

char authMethod[] = "use-token-auth";// authentication methodchartoken[] =TOKEN;

charclientId[]="d:"ORG":"DEVICE\_TYPE ":"DEVICE\_ID;//clientid

WiFiClientwifiClient; // creating the instance for wificlientPubSubClientclient(server,1883,callback,wifiClient);

voidsetup(){

**Serial**.begin(115200); // Starts the serial communicationpinMode(trigPin, OUTPUT); // Sets the trigPin as an OutputpinMode(echoPin, INPUT); // Sets the echoPin as an Input**Serial**.println();

wificonnect();mqttconnect();

}

voidloop(){

// Clears the trigPindigitalWrite(trigPin, LOW);delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro secondsdigitalWrite(trigPin,HIGH);

delayMicroseconds(10);digitalWrite(trigPin,LOW);

// Reads the echoPin, returns the sound wave travel time in microsecondsduration= pulseIn(echoPin,HIGH);

// Calculate the distancedistance=duration\*Speed/2;

//Convert toinches

distanceInch=distance\*cm\_to\_inch;

// Prints the distance in the Serial Monitor**Serial**.print("Distance : ");**Serial**.println(distance);

PublishData(distance);delay(1000);

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

}

}

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

/\*

creating theStringininformJSontoupdatethe datato ibmcloud

\*/

Stringpayload="{\"DistanceinCentimeter\":";

payload += centimeter;payload+= "}";

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

**Serial**.println(payload);

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

**Serial**.println("Publish ok");// if it sucessfully upload data on the cloudthen it will print publish ok in Serial monitor or else it will print publishfailed

}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("Connecting...");

WiFi.begin("Wokwi-GUEST","",6);//passingthewificredentialstoestablishtheconnection

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");

}

}

voidcallback(char\*subscribetopic,byte\*payload,unsignedintpayloadLength)

{

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

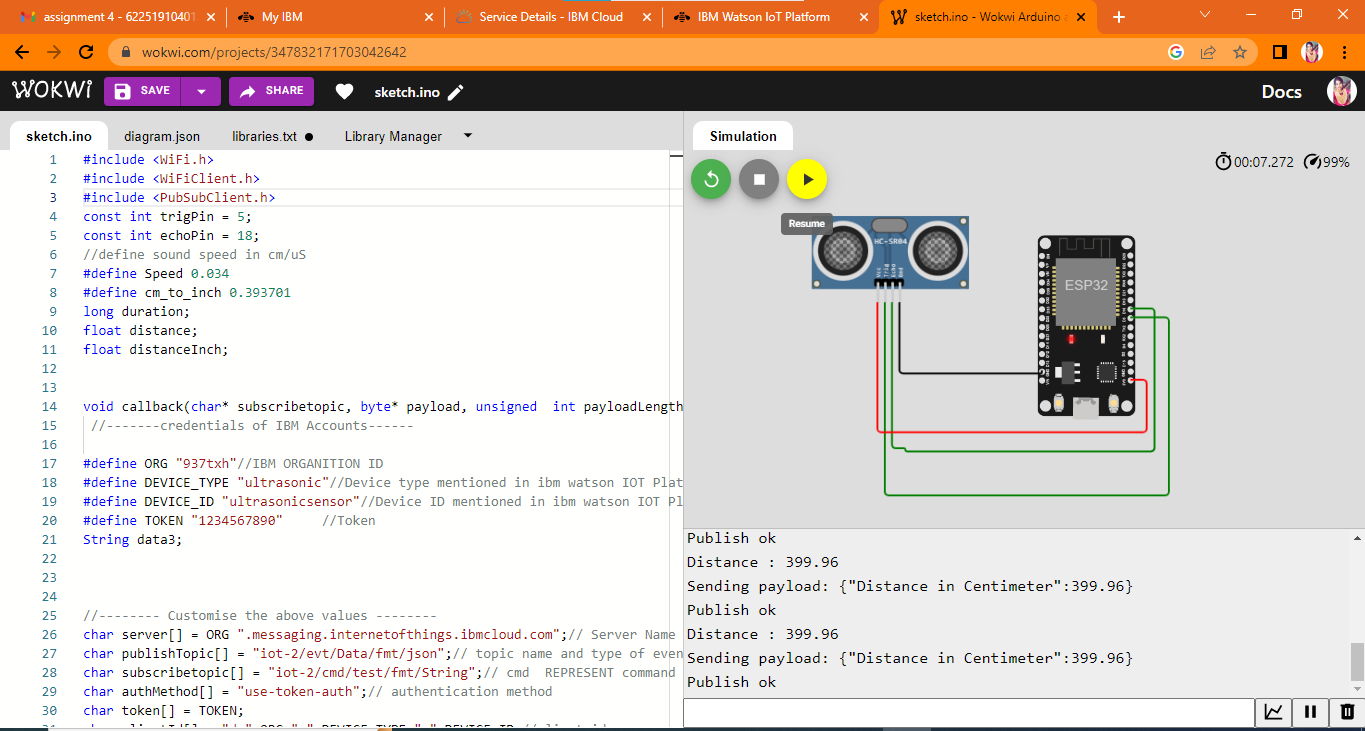
**Serial**.println(subscribetopic);

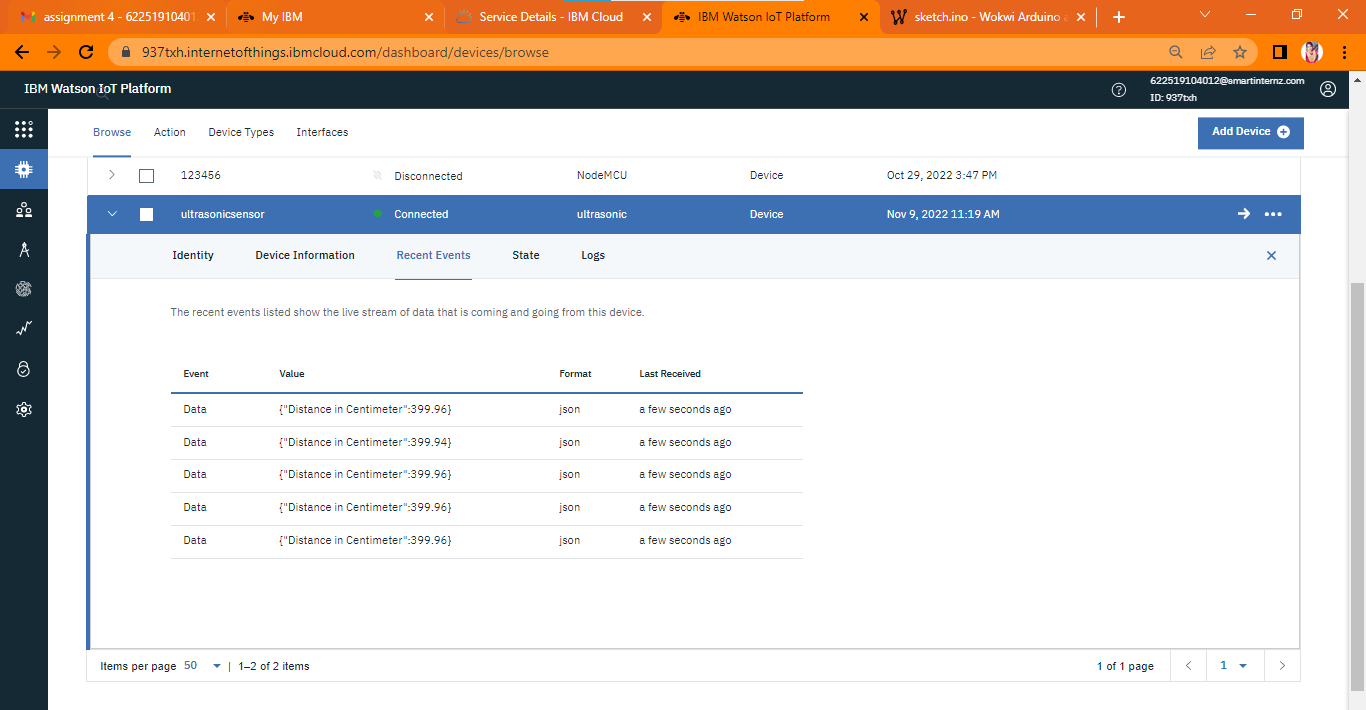
for(inti=0;i<payloadLength;i++){

//Serial.print((char)payload[i]);data3+=(char)payload[i];

}

}



** WokwiSharelink:**

https://wokwi.com/projects/347832171703042642