Name: CHINNASAMY K **RegNo:** GCTC1918107

Degree & Branch: B.Tech-FinalYear-InformationTechnology

College: GovernmentCollegeofTechnology,Coimbatore–641013

Subject: ProfessionalReadinessforInnovation,Employability&Entrepreneurship(NalaiyaThiran)

Assignment–4ESP32withUltrasonicSenor andIoTWatson

Task–1:

Write code and connections in wokwi for ultrasonic sensor. 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 ofibmcloud

Solution:

Program:

```
#include<WiFi.h>//libraryforwifi#include<Pu
bSubClient.h>//libraryforMQtt
#define TRIG_PIN 4// ESP32 pin GIOP23 connected to Ultrasonic Sensor's TRIG
pin#define ECHO_PIN 2// ESP32 pin GIOP22 connected to Ultrasonic Sensor's ECHO
pin#defineDHTTYPEDHT11//define typeofsensorDHT11

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

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

#define ORG "svzstn"//IBM ORGANITION ID
#define DEVICE_TYPE "chinna635111"//Device type mentioned in ibmwatson IOT Platform
#define DEVICE_ID "samy635111"//Device ID mentioned in ibmwatson IOT Platform
#define TOKEN "samy635111"     //Token
String data3;
floatduration_us, distance_cm;


charserver[]=ORG".messaging.internetofthings.ibmcloud.com";//ServerName
charpublishTopic[]="iot-2/evt/Data/fmt/json";//topicnameandtypeofeventperformandformatin
whichdatatobesend
charsubscribetopic[] = "iot-2/cmd/test/fmt/String";// cmdREPRESENT command type
ANDCOMMANDISTESTOFFORMATSTRING
charauthMethod[] = "use-token-auth";// authentication
methodchartoken[] =TOKEN;
charclientId[]="d:"ORG":"DEVICE_TYPE ":"DEVICE_ID;//clientid


//-----------------------------------------
WiFiClientwifiClient;//creating theinstanceforwificlient
PubSubClientclient(server,1883,callback,wifiClient);//callingthepredefinedclientidbypassi
ng parameterlikeserverid,portandwificredential
```

```
voidsetup()//configureingtheESP32
{
  Serial.begin(115200);pinMo
  de(TRIG_PIN,
  OUTPUT);pinMode(ECHO_PIN,I
  NPUT);
  delay(10);Serial.
  println();wificon
  nect();mqttconnec
  t();
}


voidloop()//RecursiveFunction
{
  digitalWrite(TRIG_PIN,
  HIGH);delayMicroseconds(10);d
  igitalWrite(TRIG_PIN,LOW);
  // measure duration of pulse from ECHO
  pinduration_us=pulseIn(ECHO_PIN, HIGH);
  // calculate the
  distancedistance_cm=0.017*duration_
  us;
  // print the value to Serial
  MonitorSerial.print("distance:
  ");Serial.print(distance_cm);Serial.
  println("cm");
  delay(500);if(distan
  ce_cm<100){
    PublishData(distance_cm);
  }
  delay(1000);
  if(!client.loop())
    {mqttconnect();
  }
}

/*.....retrievingtoCloud ....... */

voidPublishData(floatdistance_cm)
{
  mqttconnect();//functioncallforconnectingtoibm
  /*
     creating theStringininformJSontoupdatethe datato ibmcloud
  */
  String payload =
  "{\"Alert\":\"ON\"";payload += ","
  "\"Distance_cm\":";payload+=
  distance_cm;
  payload+= "}";

  Serial.print("Sendingpayload:");
  Serial.println(payload);
  if(client.publish(publishTopic,(char*)payload.c_str())){
```

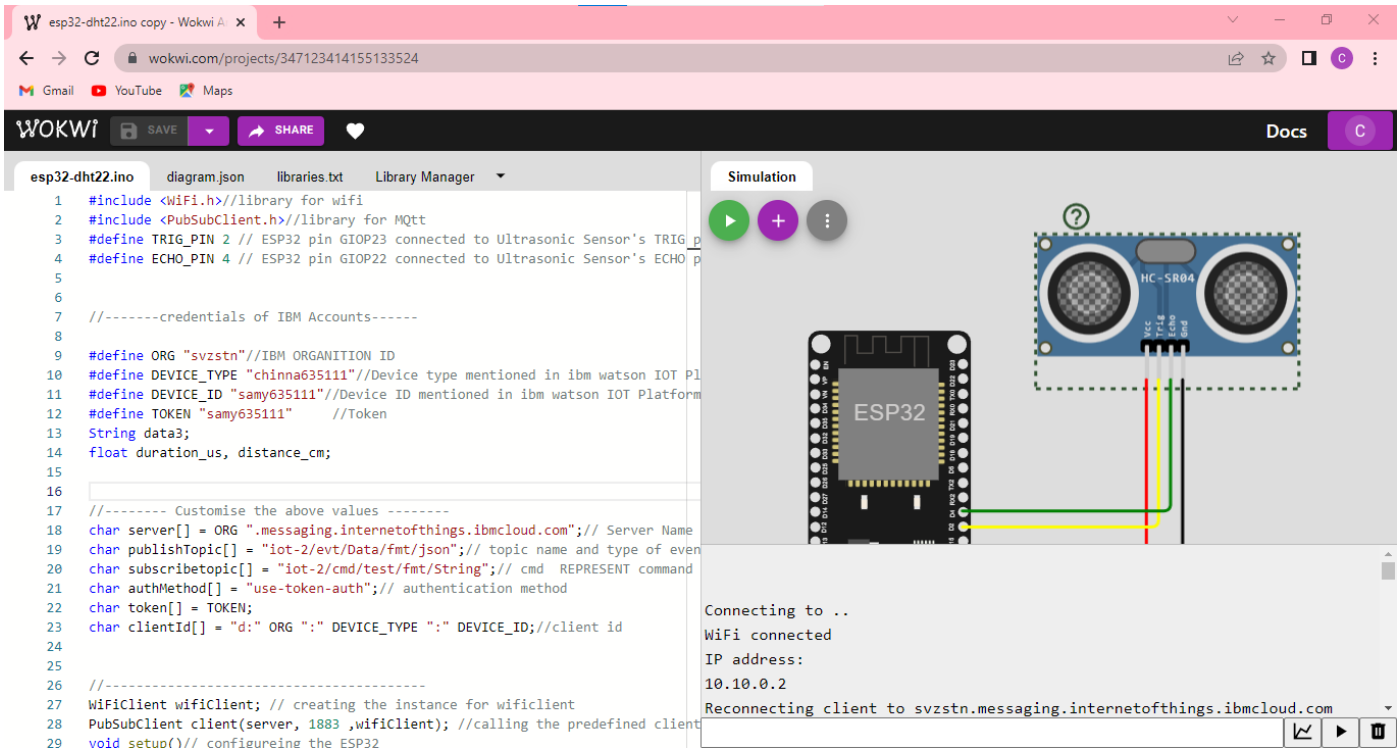
```
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then
itwillprintpublishok inSerialmonitoror elseitwill printpublishfailed
  }else{
    Serial.println("Publishfailed");
  }
}


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


voidwificonnect()//functiondefinationforwificonnect
{
  Serial.println();Serial.print("Conn
  ectingto");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish
theconnection
  while(WiFi.status() != WL_CONNECTED)
    {delay(500);
    Serial.print(".");
  }
  Serial.println("");Serial.println
  ("WiFi
  connected");Serial.println("IP
  address:
  ");Serial.println(WiFi.localIP())
  ;
}
```

OutputofProgram:

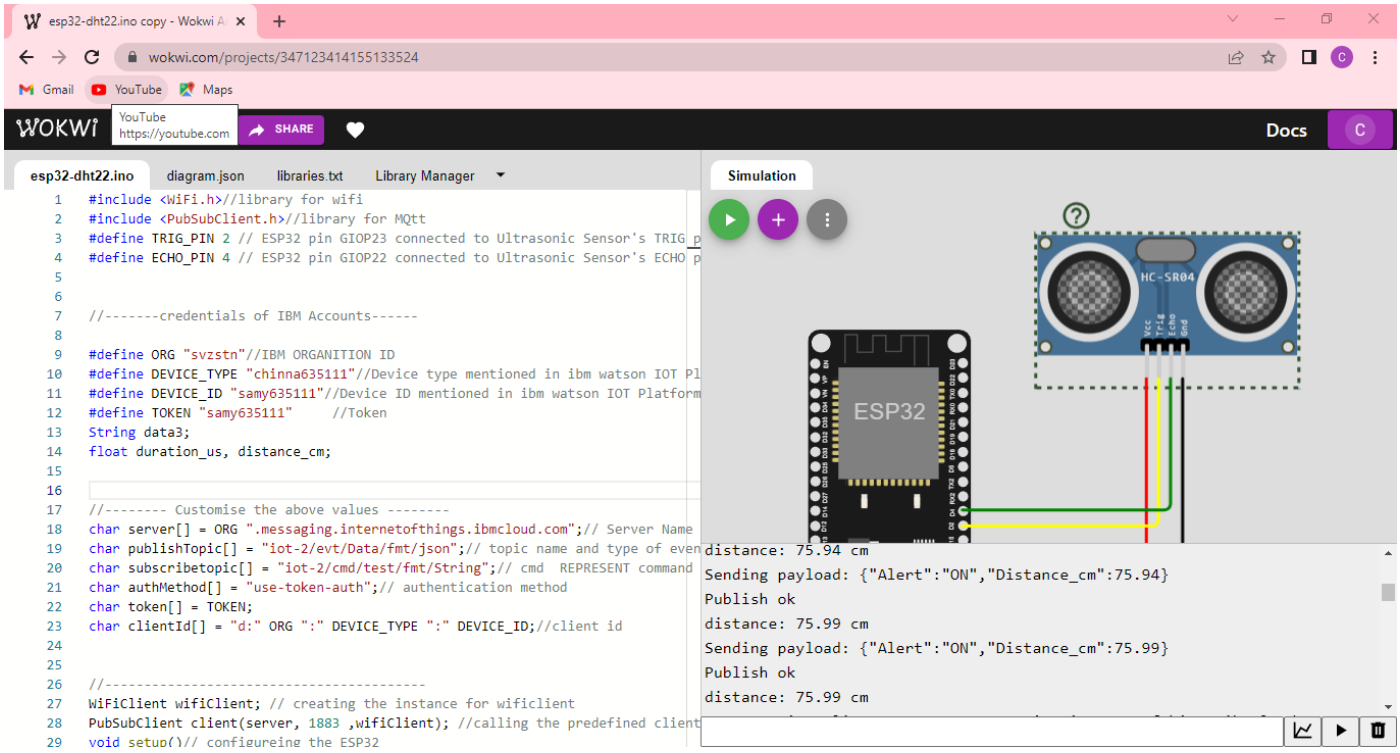
WokwiProjectLink:

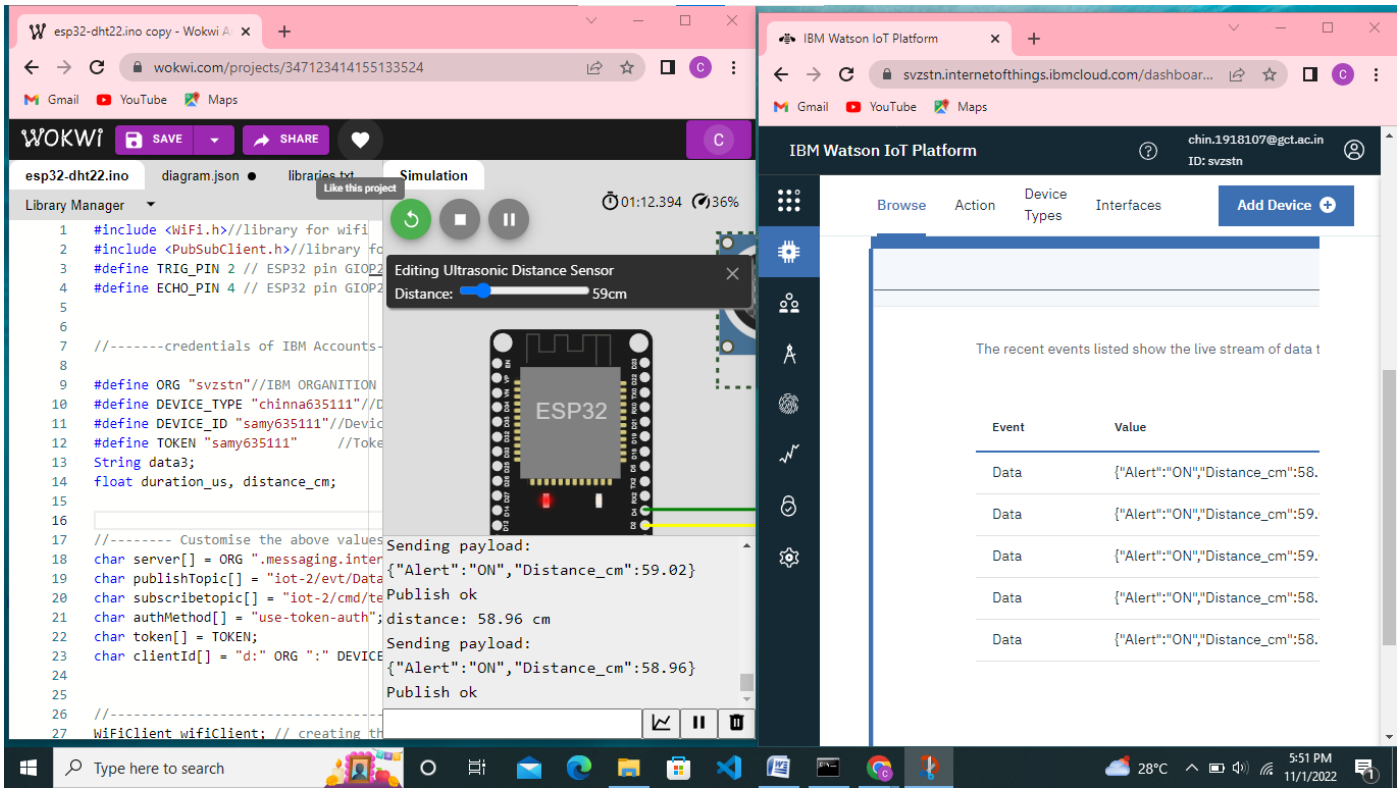
https://wokwi.com/projects/347123414155133524

WokwiWebsiteScreenshot:



IBMCloudScreenshot:




esp32-dht22.ino copy - Wokwi A
wokwi.com/projects/347123414155133524
WOKWI
SAVE
SHARE
esp32-dht22.ino
diagram.json
libraries.txt
Simulation
Library Manager
Like this project
01:12.394
36%
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library fo
#define TRIG_PIN 2 // ESP32 pin GIOP2
#define ECHO_PIN 4 // ESP32 pin GIOP2
//-------credentials of IBM Accounts-
#define ORG "svzstn"//IBM ORGANITION
#define DEVICE_TYPE "chinna635111"//D
#define DEVICE_ID "samy635111"//Devic
#define TOKEN "samy635111" //Toke
String data3;
float duration_us, distance_cm;
//-------- Customise the above values
char server[] = ORG ".messaging.inter
char publishTopic[] = "iot-2/evt/Data
char subscribetopic[] = "iot-2/cmd/te
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE
WiFiClient wifiClient; // creating th
Editing Ultrasonic Distance Sensor
Distance:
59cm
ESP32
Sending payload:
{"Alert":"ON","Distance_cm":59.02}
Publish ok
distance: 58.96 cm
Sending payload:
{"Alert":"ON","Distance_cm":58.96}
Publish ok
IBM Watson IoT Platform
svzstn.internetofthings.ibmcloud.com/dashboar...
chin.1918107@gct.ac.in
ID: svzstn
Browse
Action
Device Types
Interfaces
Add Device
The recent events listed show the live stream of data t
Event
Value
Data
{"Alert":"ON","Distance_cm":58.
Data
{"Alert":"ON","Distance_cm":59.
Data
{"Alert":"ON","Distance_cm":59.
Data
{"Alert":"ON","Distance_cm":58.
Data
{"Alert":"ON","Distance_cm":58.
Type here to search
28°C
5:51 PM
11/1/2022

ExplanationofProgram:

Initially, we have imported the <Wifi.h> and <PubSubClient.h> header files as they are needed to connectwifiandMQTTProtocol.Then,DefineTriggerpinandEchopinvalueswheretheultrasonicsensorisconnectedwith ESP32 module. Then, Define the IBM Account Credentials such as ORG, Device_Type, Device_ID and Token. Alsodefine the server, publishTopic, SubscribeTopic, authMethod, Token and ClientID. Create Object for WifiClient andPubSubClient.

Then, Start the void Setup() Function, Begin the Serial Monitor and set PinMode of Trigger Pin as Outputand Echo Pin as Input and call wificonnect() and mqttconnect() to initialize wifi and mqtt Connection and Definetheirmethods tomaketheConnection.

Then,Beginthevoidloop()function,digitalWriteHIGHtoTriggerPinandcreateadelayof10microsecondsandwritebackLOW.Then,usepulseIn()functionwithEchoPintocalculatetheDurationandcalculatethedistance.IftheDistanceislessthan100cm callPublishData() Functiontopublish theDatatoIoTWatsonDevice.

Finally, Define the PublishData() function with message as parameter. Then Define string that contains thepayloadwiththemessagetobesentintheJsonFormat.CallClient.publish()functionwithpublishTopicandpayloadas parameter. Also define the wificonnect() and mqttconnect() to make intial connection with wifi and mqttconnection.