

ASSIGNMENT 4

Date	24 October 2022
Team ID	PNT2022TMID52874
Project Name	Project -Real time river water quality monitoring and Control System
Maximum Marks	4 Marks

Project Title: Real Time River water quality monitoring and Control system

Team ID: PNT2022TMID52874

Team Members:

1. Shiranjeevi A - Team Leader
2. Sanjay- Team Member
3. Sherje e l- Team Member
4. Praveen bharathi- Team Member
5. Praneeth- Team Member

QUESTION:

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.

CODE:

```
#include <WiFi.h> //library for wifi
#include <WiFiClient.h>
#include <PubSubClient.h> //library for MQTT
// creating the instance by passing pin and typr of dht connected
float distance;
#define sound_speed 0.034
int trigpin=18;
int echopin=19;
int led=5;
int LED=9;
long duration;
String message; // creating the instance by passing pin and typr of dht connected

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-----credentials of IBM Accounts-----

#define ORG "f5r12v" //IBM ORGANITION ID
#define DEVICE_TYPE "IOT_Device_1" //Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "12345" //Device ID mentioned in ibm watson IOT Platform
#define TOKEN "kpX9+5HoaSWQsQszVJ" //Token
String data3;
float h, t;
```

```

//----- Customise the above values -----
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event
perform and format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT command type
AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id

//-----
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined
client id by passing parameter like server id,portand wificredential
void setup()// configureing the ESP32
{
    Serial.begin(115200);
    pinMode(trigpin,OUTPUT);
    pinMode(echopin,INPUT);
    pinMode(led,OUTPUT);
    delay(10);
    Serial.println();
    wificonnect();
    mqttconnect();
}

void loop()// Recursive Function
{

digitalWrite(trigpin,LOW);
digitalWrite(trigpin,HIGH);
delay(1000);
digitalWrite(trigpin,LOW);
duration=pulseIn(echopin,HIGH);
distance=duration*sound_speed/2;
Serial.println("distance"+String(distance)+"cm");
if(distance<100)
{
    message="Alert";
    digitalWrite(led,HIGH);
} else
{
    message="No problem";
    digitalWrite(led,LOW);
}
delay(1000);
PublishData(distance,message);
// if (!client.loop()) {
//     mqttconnect();
// }
}

```

```

/*.....retrieving to
Cloud.....*/

void PublishData(float d, String a) {
  mqttconnect();//function call for connecting to ibm
  /*
    creating the String in in form JSon to update the data to ibm cloud
  */
  String payload = "{\"distance\":";
  payload += d; payload += "}";
  payload += "," "{\"message\":";
  payload += a;
  payload += "}";

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

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

void mqttconnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }

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

void wificonnect() //function defination 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(".");
  }
}

```

```

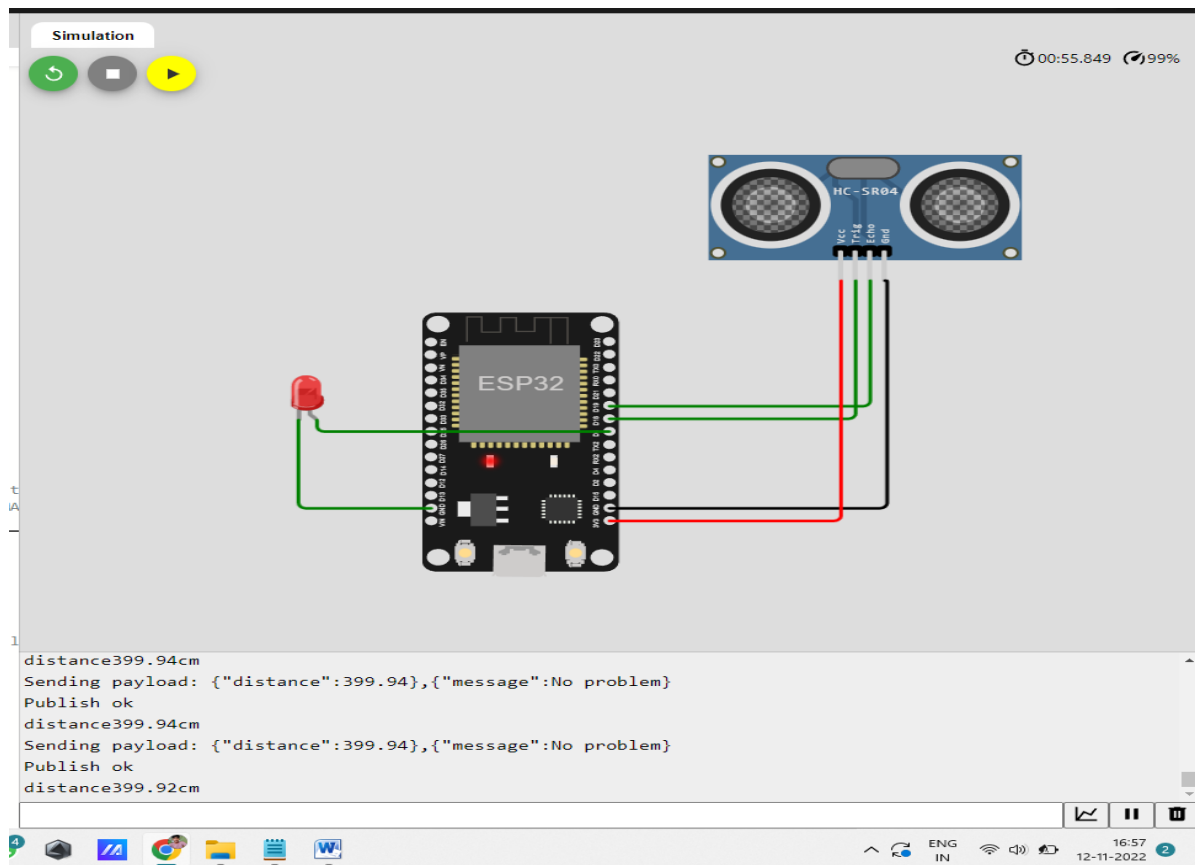
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());
}

void initManagedDevice() {
    if (client.subscribe(subscribetopic)) {
        Serial.println((subscribetopic));
        Serial.println("subscribe to cmd OK");
    } else {
        Serial.println("subscribe to cmd FAILED");
    }
}

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=="lighton")
    {
        Serial.println(data3);
        digitalWrite(LED,HIGH);
    }
    else
    {
        Serial.println(data3);
        digitalWrite(LED,LOW);
    }
    data3="";
}

```


SCHEMATIC/CIRCUIT DIAGRAM:





IBM CLOUD OUTPUT:

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Search by Device ID

Device Simulator 

<input type="checkbox"/>	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By	Device Class
>	<input type="checkbox"/>	 Disconnected	IOT_Device_1	Device	Nov 12, 2022 5:09 PM		2004207ec@cit.edu.in	
▼	<input checked="" type="checkbox"/>	 Connected	IOT_Device_1	Device	Nov 12, 2022 5:06 PM		2004207ec@cit.edu.in	→ ...

Identity

Device Information

Recent Events

State

Logs

Device ID

shiruv12

Device Type

IOT_Device_1

Date Added

Nov 12, 2022 5:06 PM

Added By

2004207ec@cit.edu.in

Connection Status


Connected
Connection Time: Nov 12, 2022 5:12 PM
Client Address: 50.31.197.64 Insecure

Items per page: 50 | 1-2 of 2 items

1 of 1 page

2 Simulations running

Search by Device ID

<input type="checkbox"/>	Device ID	Status	Device Type	Class ID	Date Added	Device Class
▼	<input checked="" type="checkbox"/>	 Connected	IOT_Device_1	Device	Nov 12, 2022 5:06 PM	

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	{"type":"Buffer","data":[123,34,100,105,115,11...	json	a few seconds ago
Data	{"type":"Buffer","data":[123,34,100,105,115,11...	json	a few seconds ago
Data	{"type":"Buffer","data":[123,34,100,105,115,11...	json	a few seconds ago
Data	{"type":"Buffer","data":[123,34,100,105,115,11...	json	a few seconds ago
Data	{"type":"Buffer","data":[123,34,100,105,115,11...	json	a few seconds ago

WOKWI LINK:

<https://wokwi.com/projects/348125060530176595>