

Assignment - 4

Name	Jaganaath A
Team ID	PNT2022TMID43363
Project Name	Hazardous Area Monitoring for Industrial Plant Powered by IoT
Maximum Marks	4 Marks

Program

```
#include <WiFi.h>
#include<PubSubClient.h>

void callback(char* subscribetopic,byte* payload,unsigned int payloadLength);
#define ORG "4wj0mx"
#define DEVICE_TYPE "TempAndHumid"
#define DEVICE_ID "TH-01"
#define TOKEN "1234567890"
#define SOUND_SPEED 0.034
#define CM_TO_INCH 0.393701

const int trigPin = 5;
const int echoPin = 18;

long duration;
float distanceCm;
float distanceInch;
String data;

char server[]=ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char subscribeTopic[] = "iot-2/cmd/command/fmt/String";
char authMethod[] = "use-token-auth";
char token[] =TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;

WiFiClient wificlient;
PubSubClient client(server,1883,callback,wificlient);

void setup() {
    Serial.begin(115200); // Starts the serial communication
    pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
    pinMode(echoPin, INPUT); // Sets the echoPin as an Input
    wificlient.connect();
    mqttconnect();
}
```

```

void loop() {
    // Clears the trigPin
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    // Sets the trigPin on HIGH state for 10 micro seconds
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

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

    // Calculate the distance
    distanceCm = duration * SOUND_SPEED/2;

    // Convert to inches
    distanceInch = distanceCm * CM_TO_INCH;

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

    delay(1000);

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

void PublishData(float distanceCm)
{
    mqttconnect ();
    String payload;

```

```

if(distanceCm<100.0)
{
    payload = "{\"Alert\":\"";
    payload += distanceCm;
    payload += "\"}";
}
else
{
    payload = "{\"distanceCm\":\"";
    payload += distanceCm;
    payload += "\"}";
}

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

if(client.publish(publishTopic , (char*) payload.c_str())){
    Serial.println("Publish ok");}
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()
{
    Serial.println();
    Serial.print("Connecting to");

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

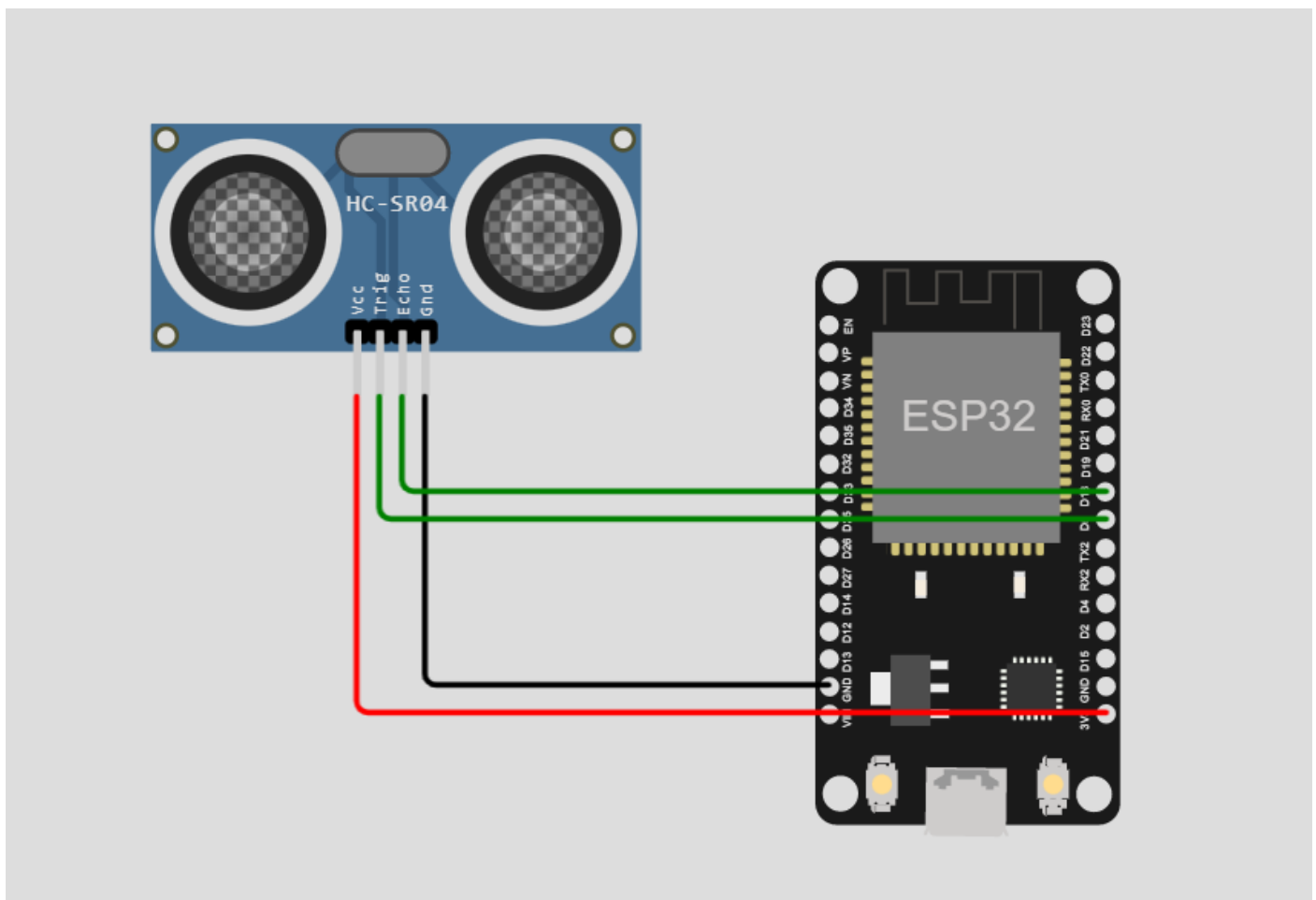
```

```

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++)
  {
    data +=(char)payload[i];
  }
  Serial.println("data: "+data);
  if(data>(char)100)
  {
    Serial.println("Alert!");
  }*/
}

```

Circuit Diagram



Wokwi Simulation

sketch.ino

diagram.json

Ultrasonic.h

libraries.txt

Ultrasonic.cpp

Library Manager

```
1 #include <WiFi.h>
2 #include<PubSubClient.h>
3
4
5 void callback(char* subscribetopic,byte* payload,unsigned int payloadLength);
6 #define ORG "4wj0mx"
7 #define DEVICE_TYPE "TempAndHumid"
8 #define DEVICE_ID "TH-01"
9 #define TOKEN "1234567890"
10 #define SOUND_SPEED 0.034
11 #define CM_TO_INCH 0.393701
12
13 const int trigPin = 5;
14 const int echoPin = 18;
15
16 long duration;
17 float distanceCm;
18 float distanceInch;
19 String data;
20
21 char server[]=ORG ".messaging.internetofthings.ibmcloud.com";
22 char publishTopic[] = "iot-2/evt/Data/fmt/json";
23 char subscribeTopic[] = "iot-2/cmd/command/fmt/String";
24 char authMethod[] = "use-token-auth";
25 char token[] =TOKEN;
26 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
27
28 WiFiClient wifiClient;
29 PubSubClient client(server,1883,callback,wifiClient);
30
31 void setup() {
32   Serial.begin(115200); // Starts the serial communication
33   pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
34   pinMode(echoPin, INPUT); // Sets the echoPin as an Input
35   wifiConnect();
36   mqttConnect();
37 }
38
```

Simulation

00:25.098 62%

Distance (cm): 394.98
Sending payload: {"distanceCm":394.98}Publish ok
Distance (cm): 394.96
Sending payload: {"distanceCm":394.96}Publish ok
Distance (cm): 394.94
Sending payload: {"distanceCm":394.94}Publish ok
Distance (cm): 394.94
Sending payload: {"distanceCm":394.94}Publish ok

WOKWI

SAVE

SHARE

Docs

sketch.ino

diagram.json

Ultrasonic.h

libraries.txt

Ultrasonic.cpp

Library Manager

```
1 #include <WiFi.h>
2 #include<PubSubClient.h>
3
4
5 void callback(char* subscribetopic,byte* payload,unsigned int payloadLength);
6 #define ORG "4wj0mx"
7 #define DEVICE_TYPE "TempAndHumid"
8 #define DEVICE_ID "TH-01"
9 #define TOKEN "1234567890"
10 #define SOUND_SPEED 0.034
11 #define CM_TO_INCH 0.393701
12
13 const int trigPin = 5;
14 const int echoPin = 18;
15
16 long duration;
17 float distanceCm;
18 float distanceInch;
19 String data;
20
21 char server[]=ORG ".messaging.internetofthings.ibmcloud.com";
22 char publishTopic[] = "iot-2/evt/Data/fmt/json";
23 char subscribeTopic[] = "iot-2/cmd/command/fmt/String";
24 char authMethod[] = "use-token-auth";
25 char token[] =TOKEN;
26 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
27
28 WiFiClient wifiClient;
29 PubSubClient client(server,1883,callback,wifiClient);
30
31 void setup() {
32   Serial.begin(115200); // Starts the serial communication
33   pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
34   pinMode(echoPin, INPUT); // Sets the echoPin as an Input
35   wifiConnect();
36   mqttConnect();
37 }
38
```

Simulation

00:16.382 85%

Editing Ultrasonic Distance Sensor
Distance: 395cm

Connecting to..
WiFi connected
IP ADDRESS
10.10.0.2
Reconnecting client to 4wj0mx.messaging.internetofthings.ibmcloud.com
iot-2/cmd/command/fmt/String
subscribe to cmd OK

Distance (cm): 200.01
Sending payload: {"distanceCm":200.01}Publish ok
Distance (cm): 199.94
Sending payload: {"distanceCm":199.94}Publish ok
Distance (cm): 115.96
Sending payload: {"distanceCm":115.96}Publish ok
Distance (cm): 92.97
Sending payload: {"Alert":92.97}Publish ok
Distance (cm): 92.97
Sending payload: {"Alert":92.97}Publish ok
Distance (cm): 165.97
Sending payload: {"distanceCm":165.97}Publish ok
Distance (cm): 394.94
Sending payload: {"distanceCm":394.94}Publish ok

IBM Watson Cloud

IBM Watson IoT Platform
jagangoguinla@gmail.com
ID: 4wj0mx

-
- Device Drilldown - TH-01**
-
-
-
-
-
-
-

[< Back](#)

Device Drilldown - TH-01

Connection Information

Recent Events

State

Device Information

Metadata

Diagnostics

Connection Logs

Device Actions

Event	Value	Format	Last Received
Data	{"distanceCm":394.94}	json	a few seconds ago
Data	{"distanceCm":394.94}	json	a few seconds ago
Data	{"distanceCm":394.96}	json	a few seconds ago
Data	{"distanceCm":394.94}	json	a few seconds ago
Data	{"distanceCm":394.94}	json	a few seconds ago

State

This table shows a list of data points that are reported by this device.

Showing Raw Data | No Interfaces Available

Property	Value	Type	Event	Last Received
distanceCm	394.94	Number	Data	a few seconds ago

Project Link - <https://wokwi.com/projects/348669654654255699>