

Assignment -4

Assignment Date	26.10.2022
Student Name	Mr.Stephenraj.W
Student Roll Number	821919104027
Maximum Marks	2 Marks

Question :

Write code and connections in wokwi for the ultrasonic sensor. Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in the device recent events. Upload document with wokwi share link and images of IBM cloud.

Wokwi Link:

<https://wokwi.com/projects/new/esp32>

Code:

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "uzesig"
#define DEVICE_TYPE "Assignment4"
#define DEVICE_ID "12345"
#define TOKEN "12345678"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/data/fmt/json";
char topic[] = "iot-2/cmd/home/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);
void publishData();
const int trigpin = 5;
const int echopin = 18;
String command;
String data = "";
long duration;
float dist;

void setup()
```

```

{
Serial.begin(115200);
pinMode(led, OUTPUT);
pinMode(trigpin, OUTPUT);
pinMode(echopin, INPUT);
wifiConnect();
mqttConnect();
}
void loop() {
bool isNearby = dist < 100;
digitalWrite(led, isNearby);
publishData();
delay(500);
if (!client.loop()) {
mqttConnect();
}
}
void wifiConnect() {
Serial.print("Connecting to "); Serial.print("Wifi");
WiFi.begin("Wokwi-GUEST", "", 6);
while (WiFi.status() != WL_CONNECTED) {
delay(500);
Serial.print(".");
}
Serial.print("WiFi connected, IP address: ");
Serial.println(WiFi.localIP());
}
void mqttConnect() {
if (!client.connected()) {
Serial.print("Reconnecting MQTT client to "); Serial.println(server);
while (!client.connect(clientId, authMethod, token)) {
Serial.print(".");
delay(500);
}
initManagedDevice();
Serial.println();
}
}
void initManagedDevice() {
if (client.subscribe(topic)) {
// Serial.println(client.subscribe(topic));
Serial.println("IBM subscribe to cmd OK");
} else {
Serial.println("subscribe to cmd FAILED");
}
}
void publishData()
{

```

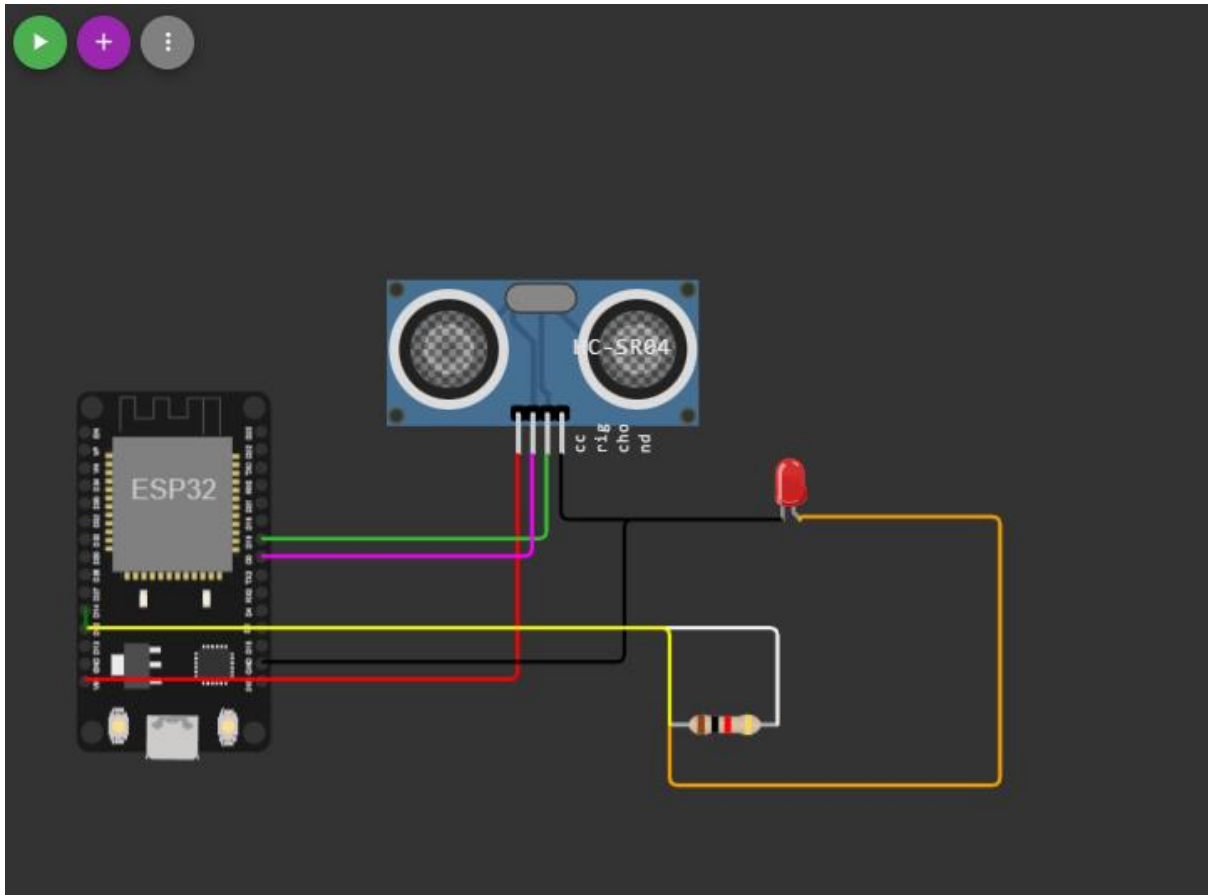
```

digitalWrite(trigpin, LOW);
digitalWrite(trigpin, HIGH);
delayMicroseconds(10);
digitalWrite(trigpin, LOW);
duration = pulseIn(echopin, HIGH);
dist = duration * speed / 2;
if (dist < 100) {
String payload = "{\"Normal Distance\":";
payload += dist;
payload += "}";
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) {
Serial.println("Publish OK");
}
}
if (dist > 101 ) {
String payload = "{\"Alert distance\":";
payload += dist;
payload += "}";
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) {
Serial.println("Warning crosses 110cm -- it automaticaly of the loop");
digitalWrite(led, HIGH);
} else {
Serial.println("Publish 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++) {
dist += (char)payload[i];
}
Serial.println("data:" + data3);
if (data3 == "lighton") {
Serial.println(data3);
digitalWrite(led, HIGH);
}
data3 = "";
}

```

Circuit Diagram:



Output:

WOKWI

SAVE SHARE

Docs

sketch.ino • diagram.json • libraries.txt • Library Manager

Simulation

00:35.035 57%

```

1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 WiFiClient wificlient;
4 String data;
5 #define ORG "uassig"
6 #define DEVICE_TYPE "Assignmenta"
7 #define DEVICE_ID "12345"
8 #define TOKEN "12345678"
9 #define speed 0.034
10 #define led 34
11 char server[] = ORG "messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/data/fmt/json";
13 char topic[] = "iot-2/cmd/home/fmt/string";
14 char authMethod[] = "use-token-auth";
15 char token[] = TOKEN;
16 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server, 1883, wificlient);
18 void publishData();
19 const int trigpin = 5;
20 const int echopin = 10;
21 String command;
22 String data = "";
23 long duration;
24 float dist;
25
26 void setup()
27 {
28   Serial.begin(115200);
29   pinMode(led, OUTPUT);
30   pinMode(trigpin, OUTPUT);
31   pinMode(echopin, INPUT);
32   wifiConnect();
33   mqttConnect();
34 }
35
36 void loop() {
37   bool isNearby = dist < 100;
38   digitalWrite(led, isNearby);
39   publishData();
40 }
  
```

IBM Watson IoT Platform

?

821919104027@smartinternz.com
ID: uzesig

Browse

Action

Device Types

Interfaces

Add Device +

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	{"Normal Distance":25.94}	json	a few seconds ago
data	{"Normal Distance":25.94}	json	a few seconds ago
data	{"Normal Distance":25.94}	json	a few seconds ago
data	{"Normal Distance":66.96}	json	21 minutes ago
data	{"Normal Dis	0 Simulations running	