

Assignment -4
Python Programming

Assignment Date	07 OCTOBER 2022
Student Name	MANISHA S
Student Roll Number	621319106053
Maximum Marks	2 Marks

Question-1:

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

Solution:

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
WiFiClient wifiClient;
#define ORG "9tg03j"
#define DEVICE_TYPE "RaspberryPi"
#define DEVICE_ID "12345"
#define TOKEN "12345678"
#define speed 0.034
char server[] = ORG
".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/status1/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=19;
String command;
String data="";
String name="Alert";
String icon="";
long duration;
int dist;
void setup()
{
  Serial.begin(115200);
  pinMode(trigpin, OUTPUT);
  pinMode(echopin, INPUT);
```

```
wifiConnect();
mqttConnect();
}
void loop() {
  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(".");
      Serial.print("*");
      delay(1000);
    }
    initManagedDevice();
    Serial.println();
  }
}
void initManagedDevice() {
  if (client.subscribe(topic)) {
    Serial.println(client.subscribe(topic));
    Serial.println("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){
dist=100-dist;
icon="Not-Crashed";
}
else{
dist=0;
icon="Crashed";
}
DynamicJsonDocument doc(1024);
String payload;
doc["Name"]=name;
doc["Impact"]=icon;
doc["Distance"]=dist;
serializeJson(doc, payload);
delay(3000);
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str()))
{
Serial.println("Publish OK");
}
else {
Serial.println("Publish FAILED");
}
}
```

Browse

Action

Device Types

Interfaces

12345

Connected

Rasp

Device

Oct 14, 2022 9:55 AM

→

...

Identity

Device Information

Recent Events

State

Logs

X

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
status1	{"Name":"Alert","Icon":"trash","FillPercent":0}	json	a few seconds ago
status1	{"Name":"Alert","Icon":"trash","FillPercent":0}	json	a few seconds ago
status1	{"Name":"Alert","Icon":"trash","FillPercent":0}	json	a few seconds ago
status1	{"Name":"Alert","Icon":"trash","FillPercent":0}	json	a few seconds ago

WOKWI

SAVE

SHARE

sketch.ino

Docs

sketch.ino

diagram.json

libraries.btt

Library Manager

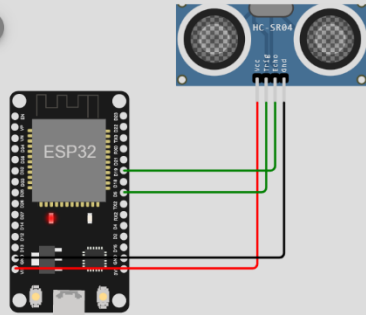
```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include <ArduinoJson.h>
4 WiFiClient wifiClient;
5 #define ORG "stgo3j"
6 #define DEVICE_TYPE "RaspberryPi"
7 #define DEVICE_ID "12345"
8 #define TOKEN "12345678"
9 #define speed 0.034
10 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
11 char publishTopic[] = "iot-2/evt/status1/fmt/json";
12 char topic[] = "iot-2/cmd/home/fmt/String";
13 char authMethod[] = "use-token-auth";
14 char token[] = TOKEN;
15 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
16 PubSubClient client(server, 1883, wifiClient);
17 void publishData();
18 const int trigpin=5;
19 const int echopin=19;
20 String command;
21 String data="";
22 String name="Alert";
23 String icon="";
24 long duration;
25 int dist;
26 void setup()
27 {
28   Serial.begin(115200);
29   pinMode(trigpin, OUTPUT);
30   pinMode(echopin, INPUT);
31   wifiConnect();
32   mqttConnect();
33 }
```

Simulation

▶

■

⏸



00:14.047

100%

Sending payload: {"Name":"Alert","Impact":"Crashed","Distance":0}

Publish OK

Sending payload: {"Name":"Alert","Impact":"Crashed","Distance":0}

Publish OK

76°F

Cloudy

Search

ENG

IN

10:31 PM

04-Nov-22