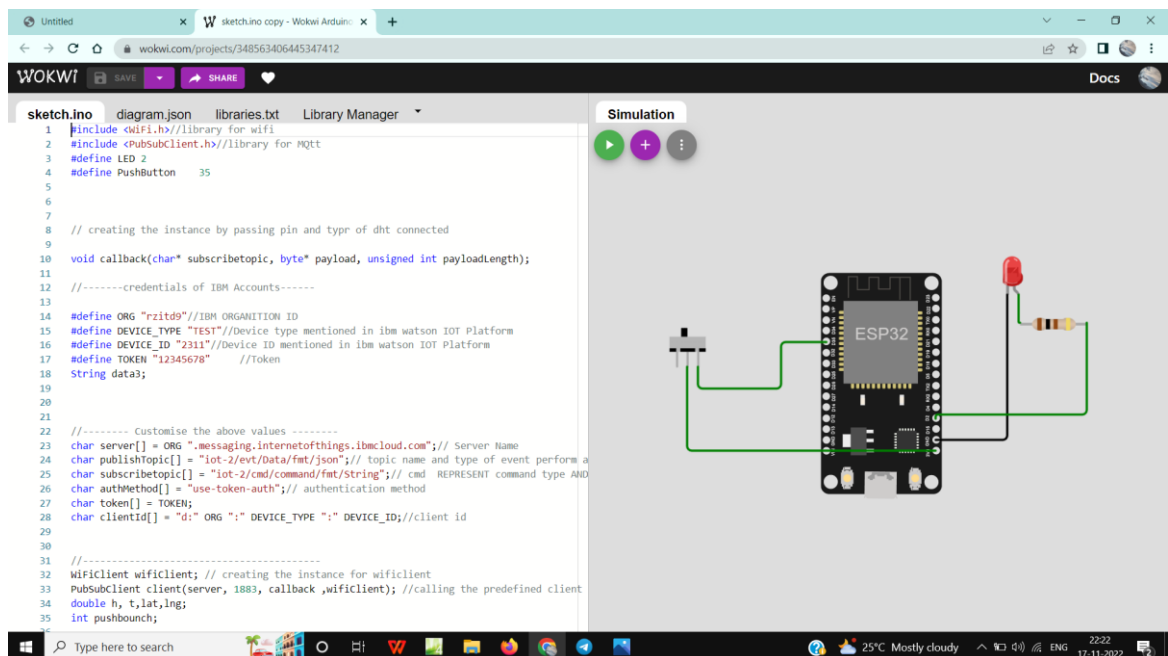


SPRINT 1

Project Topic : IoT based Safety Gadget for child Safety Monitoring and Notification

TeamID : PNT2022TMID46774

- We Connect the ESP32 with a LED and a PUSHBUTTON. and a Resister.
- And Writing Code for that.
- From this Simulation We can get Message in Our IBM Cloud Device on Recent Tabs.



WOKWI

sketch.ino diagram.json libraries.txt Library Manager

```

37 void setup()// configuring the ESP32
38 {
39   Serial.begin(115200);
40   pinMode(LED,OUTPUT);
41   pinMode(PushButton, INPUT);
42   delay(10);
43   Serial.println();
44   wifiConnect();
45   mqttconnect();
46 }
47
48 void loop()// Recursive Function
49 {
50
51   h = 11.12756;
52   t = 7.777666;
53   lat = 9.10232;
54   lng = 10.102323;
55   pushbounch = digitalRead(PushButton);
56
57   PublishData(t, h);
58   delay(1000);
59   if (!client.loop()) {
60     mqttconnect();
61   }
62 }
63
64
65
66 /*.....retrieving to Cloud.....*/
67
68 void PublishData(double LAT, double LONG) {
69   mqttconnect();//function call for connecting to ibm
70   /*
71   | creating the String in in form JSON to update the data to ibm cloud
72   */

```

Simulation

WOKWI

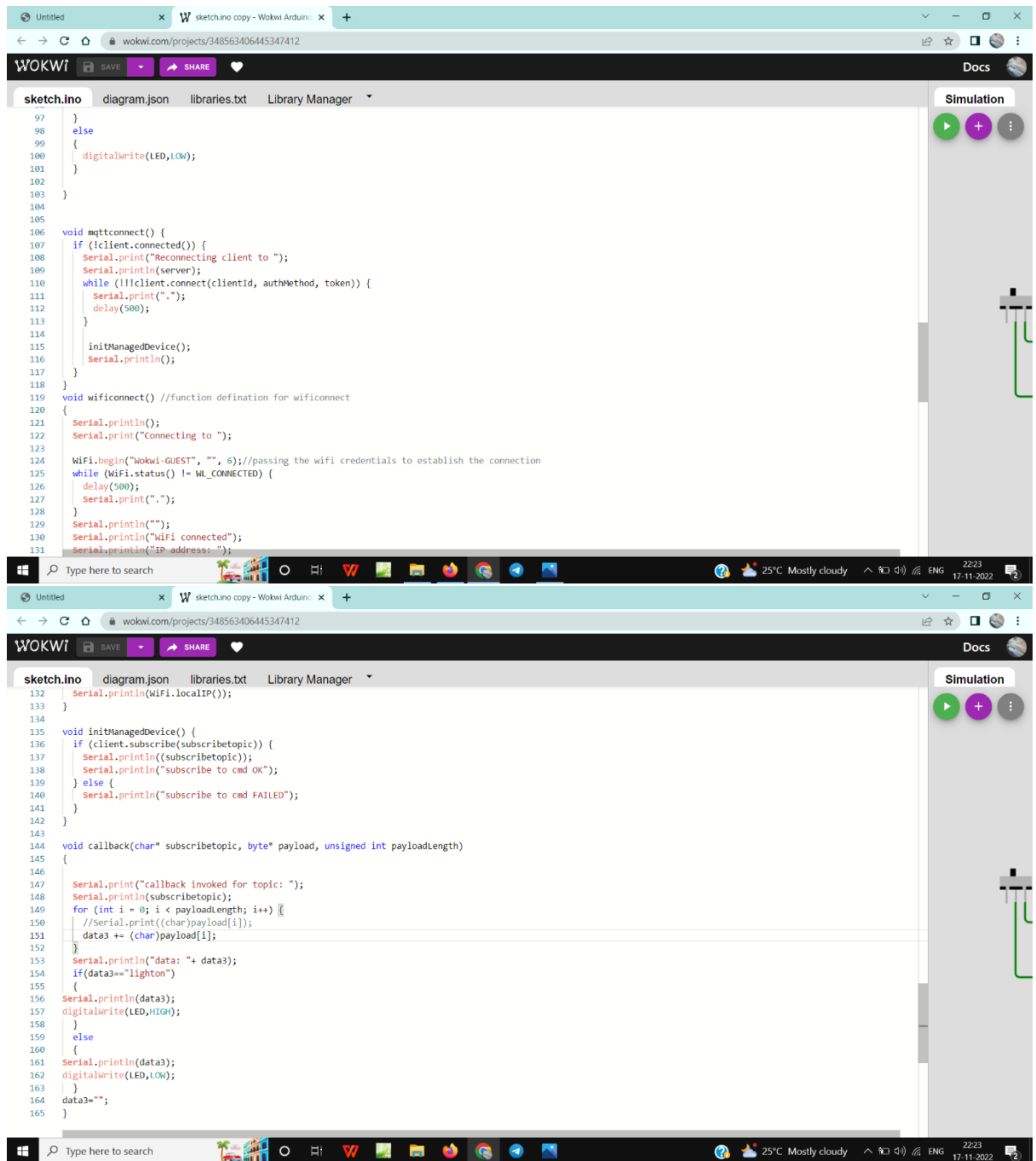
sketch.ino diagram.json libraries.txt Library Manager

```

66 /*.....retrieving to Cloud.....*/
67
68 void PublishData(double LAT, double LONG) {
69   mqttconnect();//function call for connecting to ibm
70   /*
71   | creating the String in in form JSON to update the data to ibm cloud
72   */
73   if(pushbounch == 1)
74   {
75
76     if(h != lat)
77     {
78       String payload = "{\"Latitude\":\"";
79       payload += LAT;
80       payload += "\",\"Longitude\":\"";
81       payload += LONG;
82       payload += "\"}";
83       Serial.print("Sending payload: ");
84       Serial.println(payload);
85
86       digitalWrite(LED,HIGH);
87
88       if (client.publish(publishTopic, (char*) payload.c_str()))
89       {
90         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
91       }
92       else
93       {
94         Serial.println("Publish failed");
95       }
96     }
97   }
98   else
99   {
100     digitalWrite(LED,LOW);

```

Simulation



After Simulation the Device is getting Connected to the IBM Cloud . And Whenever we ON,

- the Push Button the LED gets ON ,
- the Alert Message will received by the Cloud on the Recent Events.

Wokwi

sketch.ino diagram.json libraries.txt Library Manager

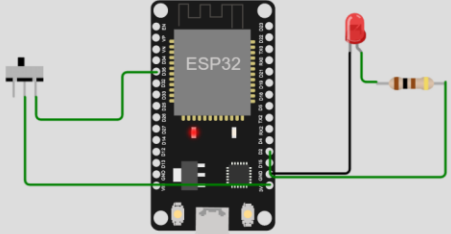
```

1 #include <WiFi.h> //library for wifi
2 #include <PubSubClient.h> //library for MQTT
3 #define LED 2
4 #define PushButton 35
5
6
7
8 // creating the instance by passing pin and type of dht connected
9
10 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
11
12 //-----credentials of IBM Accounts-----
13
14 #define ORG "rzitd9" //IBM ORGANIZATION ID
15 #define DEVICE_TYPE "TEST" //Device type mentioned in ibm watson IOT Platform
16 #define DEVICE_ID "2311" //Device ID mentioned in ibm watson IOT Platform
17 #define TOKEN "12345678" //Token
18 String data3;
19
20
21 //----- Customise the above values -----
22
23 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // Server Name
24 char publishTopic[] = "iot-2/evt/data/fmt/json"; // topic name and type of event performed
25 char subscribetopic[] = "iot-2/cmd/command/fmt/String"; // cmd REPRESENT command type
26 char authMethod[] = "use-token-auth"; // authentication method
27 char token[] = TOKEN;
28 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
29
30 //-----
31
32 WiFiClient wificlient; // creating the instance for wificlient
33 PubSubClient client(server, 1883, callback, wificlient); //calling the predefined client
34 double h, t, lat, lng;
35 int pushbounc;

```

Simulation

00:02.997 104%



Connecting to
 Wifi connected
 IP address:
 10.10.0.2
 Reconnecting client to rzitd9.messaging.internetofthings.ibmcloud.com

Wokwi

sketch.ino diagram.json libraries.txt Library Manager

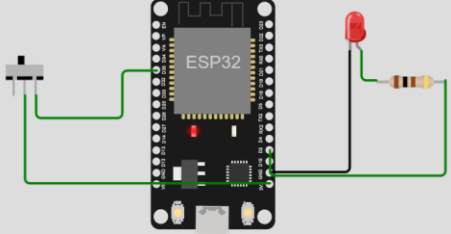
```

1 #include <WiFi.h> //library for wifi
2 #include <PubSubClient.h> //library for MQTT
3 #define LED 2
4 #define PushButton 35
5
6
7
8 // creating the instance by passing pin and type of dht connected
9
10 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
11
12 //-----credentials of IBM Accounts-----
13
14 #define ORG "rzitd9" //IBM ORGANIZATION ID
15 #define DEVICE_TYPE "TEST" //Device type mentioned in ibm watson IOT Platform
16 #define DEVICE_ID "2311" //Device ID mentioned in ibm watson IOT Platform
17 #define TOKEN "12345678" //Token
18 String data3;
19
20
21 //----- Customise the above values -----
22
23 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // Server Name
24 char publishTopic[] = "iot-2/evt/data/fmt/json"; // topic name and type of event performed
25 char subscribetopic[] = "iot-2/cmd/command/fmt/String"; // cmd REPRESENT command type
26 char authMethod[] = "use-token-auth"; // authentication method
27 char token[] = TOKEN;
28 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
29
30 //-----
31
32 WiFiClient wificlient; // creating the instance for wificlient
33 PubSubClient client(server, 1883, callback, wificlient); //calling the predefined client
34 double h, t, lat, lng;
35 int pushbounc;

```

Simulation

00:04.229 99%



Wifi connected
 IP address:
 10.10.0.2
 Reconnecting client to rzitd9.messaging.internetofthings.ibmcloud.com
 iot-2/cmd/command/fmt/String
 subscribe to cmd OK

Wokwi Arduino IDE interface showing a sketch and a simulation.

Sketch (sketch.ino):

```
1 #include <WiFi.h> //library for wifi
2 #include <PubSubClient.h> //library for MQTT
3 #define LED 2
4 #define PushButton 35
5
6
7
8 // creating the instance by passing pin and type of dht connected
9
10 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
11
12 //-----credentials of IBM Accounts-----
13
14 #define ORG "rzitd9" //IBM ORGANIZATION ID
15 #define DEVICE_TYPE "TEST" //Device type mentioned in ibm watson IOT Platform
16 #define DEVICE_ID "2311" //Device ID mentioned in ibm watson IOT Platform
17 #define TOKEN "12345678" //Token
18 String data3;
19
20
21 //----- Customise the above values -----
22
23 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // Server Name
24 char publishTopic[] = "iot-2/evt/data/fmt/json"; // topic name and type of event performed
25 char subscribetopic[] = "iot-2/cmd/command/fmt/string"; // cmd REPRESENT command type
26 char authMethod[] = "use-token-auth"; // authentication method
27 char token[] = TOKEN;
28 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
29
30
31 //-----
32 WiFiClient wificlient; // creating the instance for wificlient
33 PubSubClient client(server, 1883, callback, wificlient); //calling the predefined client
34 double h, t, lat, lng;
35 int pushbounc;
36
```

Simulation:

ESP32 board connected to a push button and an LED. The simulation shows the board sending payloads to the MQTT server.

Sending payload: {"Latitude":7.78,"Longitude":11.13}
Publish ok
Sending payload: {"Latitude":7.78,"Longitude":11.13}
Publish ok
Sending payload: {"Latitude":7.78,"Longitude":11.13}
Publish ok

Wokwi Arduino IDE interface showing a sketch and a simulation.

Sketch (sketch.ino):

```
1 #include <WiFi.h> //library for wifi
2 #include <PubSubClient.h> //library for MQTT
3 #define LED 2
4 #define PushButton 35
5
6
7
8 // creating the instance by passing pin and type of dht connected
9
10 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
11
12 //-----credentials of IBM Accounts-----
13
14 #define ORG "rzitd9" //IBM ORGANIZATION ID
15 #define DEVICE_TYPE "TEST" //Device type mentioned in ibm watson IOT Platform
16 #define DEVICE_ID "2311" //Device ID mentioned in ibm watson IOT Platform
17 #define TOKEN "12345678" //Token
18 String data3;
19
20
21 //----- Customise the above values -----
22
23 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // Server Name
24 char publishTopic[] = "iot-2/evt/data/fmt/json"; // topic name and type of event performed
25 char subscribetopic[] = "iot-2/cmd/command/fmt/string"; // cmd REPRESENT command type
26 char authMethod[] = "use-token-auth"; // authentication method
27 char token[] = TOKEN;
28 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
29
30
31 //-----
32 WiFiClient wificlient; // creating the instance for wificlient
33 PubSubClient client(server, 1883, callback, wificlient); //calling the predefined client
34 double h, t, lat, lng;
35 int pushbounc;
36
```

Simulation:

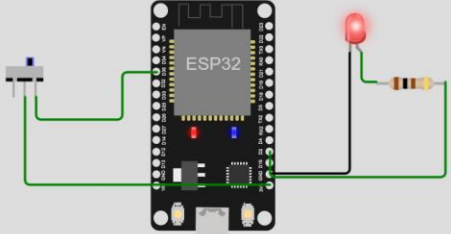
ESP32 board connected to a push button and an LED. The simulation shows the board sending payloads to the MQTT server.

Publish ok
Sending payload: {"Latitude":7.78,"Longitude":11.13}
Publish ok
Sending payload: {"Latitude":7.78,"Longitude":11.13}
Publish ok
Sending payload: {"Latitude":7.78,"Longitude":11.13}
Publish ok

sketch.ino

```
1 #include <WiFi.h> //library for wifi
2 #include <PubSubClient.h> //library for MQTT
3 #define LED 2
4 #define PushButton 35
5
6
7
8 // creating the instance by passing pin and type of dht connected
9
10 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
11
12 //-----credentials of IBM Accounts-----
13
14 #define ORG "rzitd9"//IBM ORGANIZATION ID
15 #define DEVICE_TYPE "TEST"//device type mentioned in ibm watson IOT Platform
16 #define DEVICE_ID "2311"//Device ID mentioned in ibm watson IOT Platform
17 #define TOKEN "12345678" //Token
18 String data3;
19
20
21
22 //----- Customise the above values -----
23 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
24 char publishTopic[] = "iot-2/evt/data/fmt/json";// topic name and type of event perfo
25 char subscribetopic[] = "iot-2/cmd/command/fmt/string";// cmd REPRESENT command type
26 char authMethod[] = "use-token-auth";// authentication method
27 char token[] = TOKEN;
28 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
29
30
31 //-----
32 WiFiClient wifiClient; // creating the instance for wifiClient
33 PubSubClient client(server, 1883, callback, wifiClient); //calling the predefined cli
34 double h, t,lat,lng;
35 int pushbounc;
```

Simulation



Publish ok

Sending payload: {"Latitude":7.78,"Longitude":11.13}

Publish ok

Sending payload: {"Latitude":7.78,"Longitude":11.13}

Publish ok

Sending payload: {"Latitude":7.78,"Longitude":11.13}

Publish ok

IBM Watson IoT Platform

← Back

Device Drilldown - 2311

Connection Information

Recent Events

Connection Information

Basic connection information about this device.

Device ID

2311

Device Type

TEST

Date Added

Nov 10, 2022 5:41 PM

Added By

nandhinivaradharajan4@gmail.com

Connection Status

Disconnected

Last Connected

Nov 17, 2022 10:28 PM

Client Address

145.40.93.209 Insecure

Duration

a few seconds

Data Transferred

286 B

Recent Events

1 Simulation running

