SMART BRIDGE EXTERNSHIP

ASSIGNMENT – 2

Name: N. Sriya Setty

Registration Number: 20BCI7020

Institute: VIT-AP

==========

In wokwi connect push button and upload 0 and 1 to ibm cloud

Code:

Sketch.ino

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#define button 4
#define LED 5
int buttonPin;
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
#define ORG "9f8wlx"//IBM ORGANITION ID
#define DEVICE_TYPE "abcd"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "1234"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678"     //Token
String data3;
 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
```

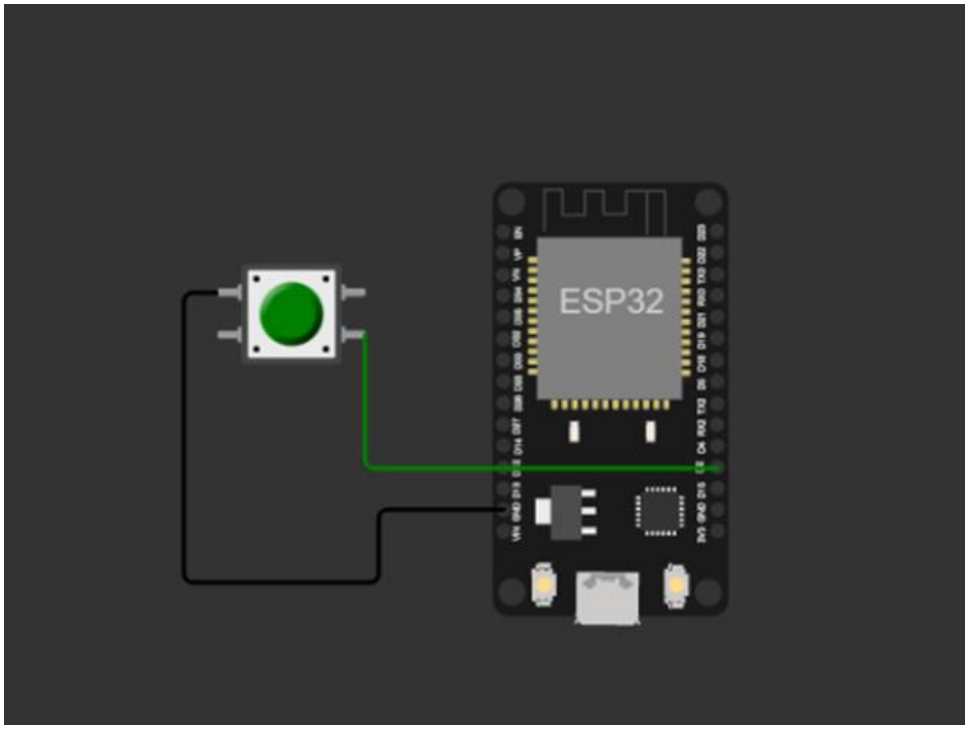
```
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing parameter like server id,portand wificredential
void setup() {
  pinMode(buttonPin, INPUT_PULLUP);
  Serial.begin(9600);
  wificonnect();
  mqttconnect();
}

void loop() {
  int buttonState = digitalRead(buttonPin);

  if (buttonState == HIGH) {
  Serial.println("Button state: 1");
   } else {
  Serial.println("Button state: 0");
  }


  delay(100);
  if (!client.loop()) {
    mqttconnect();
_} // Adjust delay as needed
}

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="";
}
```

Diagram.json

```
{
_"version": 1,
_"author": "JYOTI PRAKASH BEHURA 20BCE7355",
_"editor": "wokwi",
_"parts": [
__{ "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 0, "left": 0, "attrs": {} },
__{
___"type": "wokwi-pushbutton",
___"id": "btn1",
___"top": 38.73,
___"left": -124.27,
___"attrs": { "color": "green" }
__}
_],
_"connections": [
__[ "esp:TX0", "$serialMonitor:RX", "", [] ],
__[ "esp:RX0", "$serialMonitor:TX", "", [] ],
__[ "esp:D2", "btn1:2.r", "green", [ "h0" ] ],
__[ "btn1:1.l", "esp:GND.2", "black", [ "h-14.53", "v130", "h87.73", "v-32.73" ] ]
_],
_"dependencies": {}
}
```

Diagram:



Output:

```
Button state: 1
Button state: 1
Button state: 1
```