**SPRINT 3**

|  |  |
| --- | --- |
| Date | 12 November 2022 |
| Team ID | PNT2022TMID26330 |
| Project Name | Smart Farmer – IoT Enabled Farming Application |
| Maximum Marks | 8 Marks |

**SENDING SENSOR DATA FROM WOKWI TO IBM WATSON IOT PLATFORM:**

**PROGRAM FOR SENDING TEMPERATURE AND HUMIDITY VALUES USING MQTT PROTOCOL:**

#include <WiFi.h>

#include <PubSubClient.h>

#include "DHT.h"

#define DHTPIN 15

#define DHTTYPE DHT22

#define LED 2

DHT dht (DHTPIN, DHTTYPE); void callback(char\* subscribetopic, byte\* payload, unsigned int payloadLength);

#define ORG "tu4jce"//IBM ORGANITION ID

#define DEVICE\_TYPE "NodeMCU"//Device type

#define DEVICE\_ID "12345"//Device ID

#define TOKEN "2W?\*d5U83t+ICiNhyJ" //Token String data3; float h, t;

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); dht.begin(); pinMode(LED,OUTPUT); delay(10); **Serial**.println(); wificonnect(); mqttconnect();

}

void loop()

{

h = dht.readHumidity(); t = dht.readTemperature();

**Serial**.print("temp:");

**Serial**.println(t);

**Serial**.print("Humid:");

**Serial**.println(h);

PublishData(t, h); delay(1000); if (!client.loop()) { mqttconnect();

}

}

void PublishData(float temp, float humid) { mqttconnect(); String payload = "{\"temp\":"; payload += temp; payload += "," "\"Humid\":"; payload += humid; payload += "}";

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

**Serial**.println(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++) { //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);

}

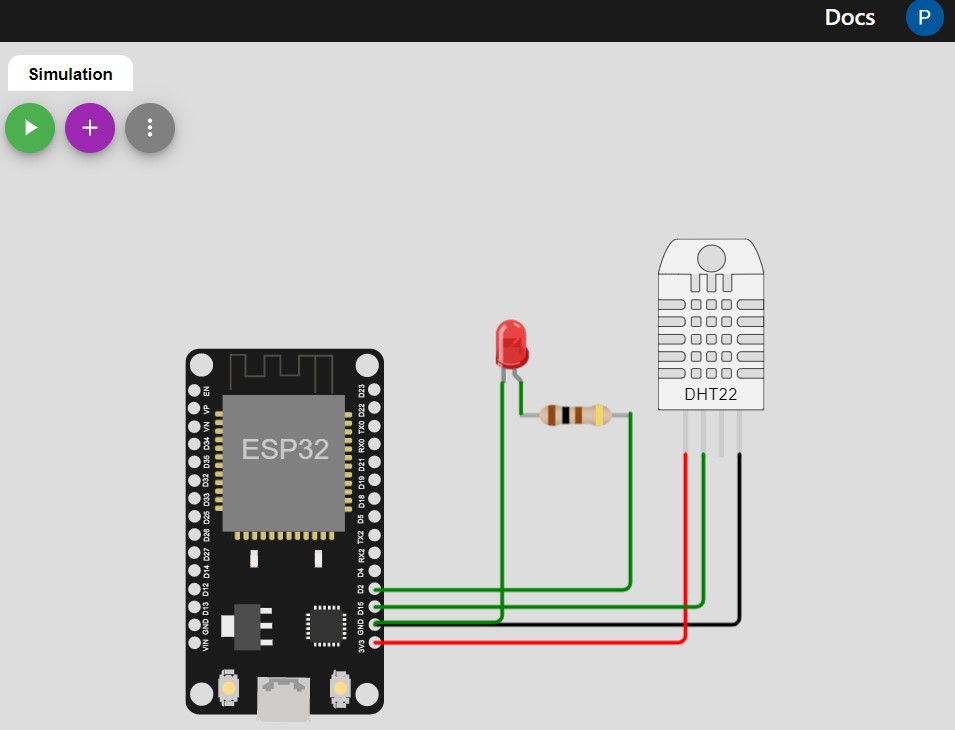
data3="";

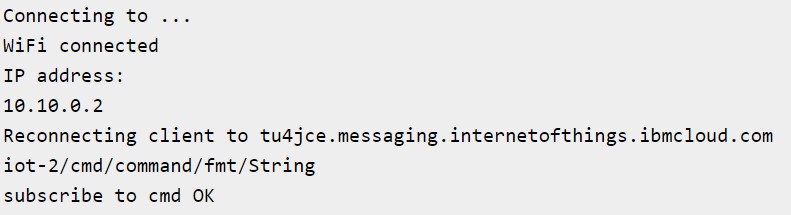
}

**CIRCUIT:**

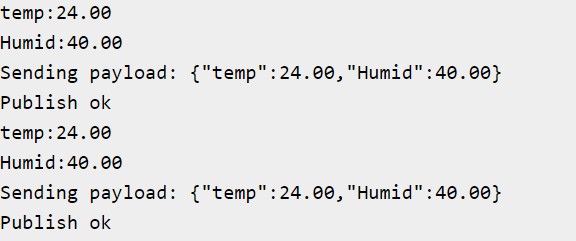
**WOKWI SERIAL M**

**ONITOR:**



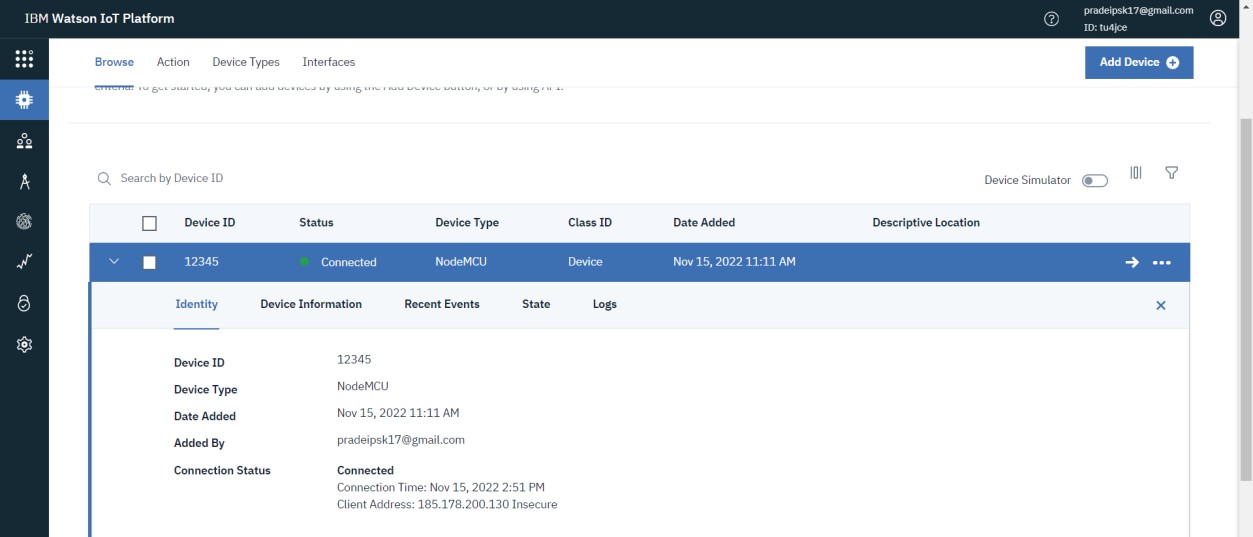


**Connecting to IBM Watson IoT platform**

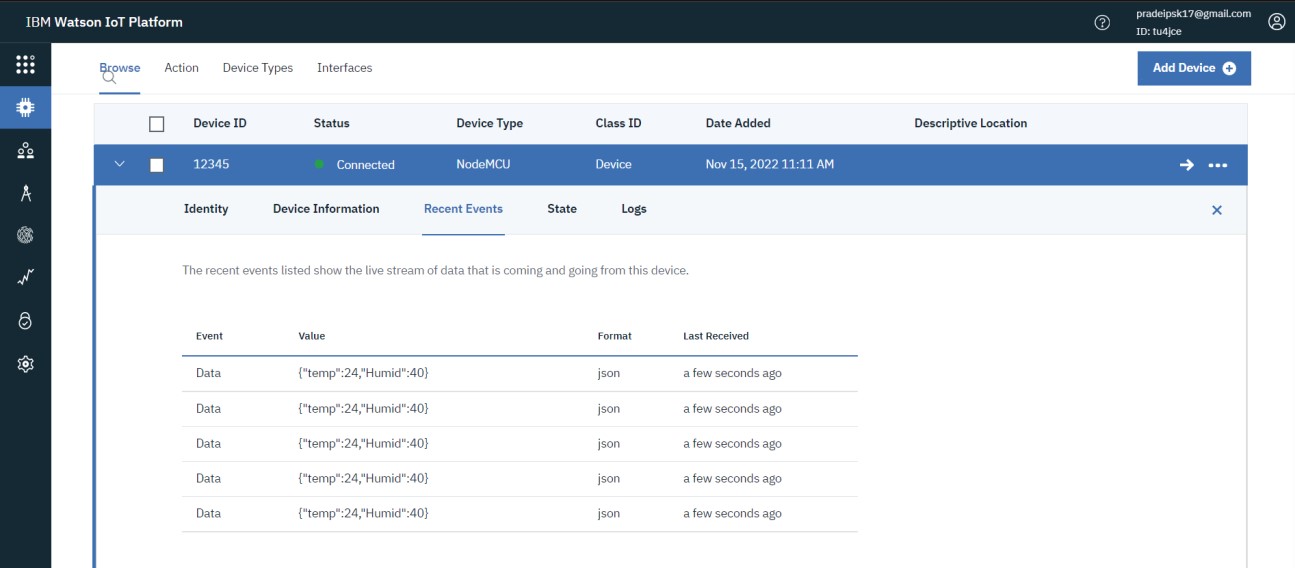


**Publishing temperature and humidity values to the IBM Watson IoT platform**

**IBM Watson IoT platform:**



**Connected Status in IBM Watson IoT platform**



**Recent Events in IBM Watson IoT platform**

Once the sensor data like temperature and humidity gets updated in the IBM Watson IoT platform, those sensor data’s will be available under recent events.