**2190151 Computer Programming Lab**

**Lab 7 M5Stack IoT**

**Objective:**

1.Student is able to use IoT platform such as NETPIE

2.Student is able to use MQTT protocol

3.Student is able to use basic HTML5

**Background Theory**

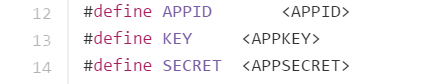
**MQTT**[[2]](https://en.wikipedia.org/wiki/MQTT#cite_note-2) (**Message Queuing Telemetry Transport**) is an [ISO standard](https://en.wikipedia.org/wiki/International_Organization_for_Standardization) (ISO/IEC PRF 20922)[[3]](https://en.wikipedia.org/wiki/MQTT#cite_note-ISO-3) [publish-subscribe](https://en.wikipedia.org/wiki/Publish%E2%80%93subscribe_pattern)-based messaging protocol. It works on top of the [TCP/IP protocol](https://en.wikipedia.org/wiki/TCP/IP). It is designed for connections with remote locations where a "small code footprint" is required or the network bandwidth is limited. The [publish-subscribe messaging pattern](https://en.wikipedia.org/wiki/Publish%E2%80%93subscribe_pattern) requires a [message broker](https://en.wikipedia.org/wiki/Message_broker).

NETPIE is a cloud platform for IoT solution development. NETPIE lets you connect everything seamlessly and takes care of all tedious work, like infrastructure administration, so you can spend time on your innovative IoT applications.

NETPIE is not managed MQTT brokers. We created our own publish-subscribe communication model that allows instant messaging among IoT devices (aka "things"). This communication model can support both the MQTT protocol and the HTTP REST protocol. To make things talk over the MQTT protocol, you would need help from NETPIE libraries (aka "Microgears"). Our Microgears are open-source and available on [github](https://github.com/netpieio" \t "_bank).

**Task 1 IoT monitoring**

1. Following guideline in <https://netpie.io/getstarted>
2. Download Microgears for hardware or operating systems of your choice from [github](https://github.com/netpieio" \t "_bank).
3. Find the example code folder. Find the following lines. (They are different in different types of Microgears.)
4. In Arduino Microgear,
5. Edit the KEY, SECRET and APPID fields according to what you have created on the NETPIE web. All variables should have the string markers "". For example, to program your light sensor in the kitchen, you would use.



1. Create new application in netpie.io, in order to get “Application ID”, “Application key” and “Application Secret”
2. Within the same application obtain 2 types of Application key, that is “Device Key” and “Session Key”
3. Connect DHT11 sensor to M5Stack
4. Implement this code in Arduino M5Stack

#include <MicroGear.h>

#include <M5Stack.h>

#include <SimpleDHT.h>

const char\* ssid = "Your ssid";

const char\* password = "Your WiFi password";

#define APPID "Your App ID"

#define KEY "Your App Key"

#define SECRET "Your App Secret"

#define ALIAS "NodeMCU1"

#define TargetWeb "HTML\_web"

#define D4 5 // TXD1

#define DHTPIN D4 // what digital pin we're connected to

#define DHTTYPE DHT11 // DHT 11

SimpleDHT11 dht;

WiFiClient client;

MicroGear microgear(client);

void onMsghandler(char \*topic, uint8\_t\* msg, unsigned int msglen)

{

Serial.print("Incoming message --> ");

msg[msglen] = '\0';

Serial.println((char \*)msg);

}

void onConnected(char \*attribute, uint8\_t\* msg, unsigned int msglen)

{

Serial.println("Connected to NETPIE...");

microgear.setAlias(ALIAS);

}

void setup()

{

/\* Event listener \*/

microgear.on(MESSAGE,onMsghandler);

microgear.on(CONNECTED,onConnected);

dht.begin();

Serial.begin(115200);

Serial.println("Starting...");

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED)

{

delay(250);

Serial.print(".");

}

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

microgear.init(KEY,SECRET,ALIAS);

microgear.connect(APPID);

}

Figure 1

Figure 2

void loop()

{

if (microgear.connected())

{

microgear.loop();

Serial.println("connected");

float Humidity = \_\_\_\_\_\_\_\_\_

float Temp = \_\_\_\_\_\_\_\_// Read temperature as Celsius (the default)

String data = "/" + String(Humidity) + "/" + String(Temp);

char msg[128];

data.toCharArray(msg,data.length());

Serial.println(msg);

microgear.chat(TargetWeb , msg);

}

else

{

Serial.println("connection lost, reconnect...");

microgear.connect(APPID);

}

delay(1500);

}

10, Open editor and type the following HTML5 code in Figure 3

11. Check the results in serial monitor and run HTML5 code on a browser

12. Explain and justify the results

13. Answer these questions:

* 1. Which command does M5Stack send the string stored in variable msg to Microgear HTML5, giver that the HTML5 browser is Hweb
  2. Which command M5Stack uses to setup KEY, SECRET, ALIAS , APPID
  3. Which command use to check that Microgear on M5Stack is connecting with NETPIE
  4. Which part in HTML code use to receive message sent to Microgear HTML5
  5. Which part in HTML is used to declare Microgear HTML5

Figure 3

<!DOCTYPE html>

<html>

<body>

<script src="https://cdn.netpie.io/microgear.js"></script>

<script>

const APPID = "Your App ID";

const KEY = "Your App Key";

const SECRET = "Your App Secret";

const ALIAS = "HTML\_web"; // ชื่อตัวเอง

const thing1 = "M5Stack"; // ชื่อเพื่อนที่จะคุย

var microgear = Microgear.create({

key: KEY,

secret: SECRET,

alias : ALIAS

});

microgear.on('message',function(topic,msg) {

document.getElementById("raw\_data").innerHTML = "Data from M5Stack = " + msg;

document.getElementById("get\_topic").innerHTML = "Topic = " + topic;

var split\_msg = msg.split("/"); //String data = "/" +String(Humidity) + "/" + String(Temp);

console.log(msg); // for debug

if(typeof(split\_msg[0])!='undefined' && split\_msg[0]==""){

document.getElementById("Humidity\_temp").innerHTML = "Humidity = " + split\_msg[1] + " % ,Temp = " + split\_msg[2] + " c";

}

});

microgear.on('connected', function() {

microgear.setAlias(ALIAS);

document.getElementById("connected\_NETPIE").innerHTML = "Connected to NETPIE"

});

microgear.on('present', function(event) {

console.log(event);

});

microgear.on('absent', function(event) {

console.log(event);

});

microgear.resettoken(function(err) {

microgear.connect(APPID);

});

</script>

<h1 id="connected\_NETPIE"></h1>

<p id="raw\_data"></p>

<p id="get\_topic"></p>

<strong id="Humidity\_temp"></strong>

</body>

</html>

**Task 2: Control LED from the Internet**

1. Create new application in netpie.io, in order to get “Application ID”, “Application key” and “Application Secret”
2. Within the same application obtain 2 types of Application key, that is “Device Key” and “Session Key”
3. Connect LED to port 16, and implement the following code in Arduino M5Stack

Figure 4

#include <MicroGear.h>

#include <M5Stack.h>

// constants won't change. They're used here to

// set pin numbers:

#define D0 16 // USER LED Wake

#define ledPin D0 // the number of the LED pin

const char\* ssid = "Your SSID";

const char\* password = "Your WiFi password";

#define APPID "Your App ID"

#define KEY "Your App Key"

#define SECRET "Your App Secret"

#define ALIAS "M5Stack"

#define TargetWeb "DigitalOUTPUT\_HTML\_web"

WiFiClient client;

MicroGear microgear(client);

void onMsghandler(char \*topic, uint8\_t\* msg, unsigned int msglen)

{

Serial.print("Incoming message --> ");

Serial.print(topic);

Serial.print(" : ");

char strState[msglen];

for (int i = 0; i < msglen; i++)

{

strState[i] = (char)msg[i];

Serial.print((char)msg[i]);

}

Serial.println();

String stateStr = String(strState).substring(0, msglen);

if(stateStr == "ON")

{

digitalWrite(ledPin, LOW);

microgear.chat(TargetWeb, "ON");

}

else if (stateStr == "OFF")

{

digitalWrite(ledPin, HIGH);

microgear.chat(TargetWeb, "OFF");

}

}

void onConnected(char \*attribute, uint8\_t\* msg, unsigned int msglen)

{

Serial.println("Connected to NETPIE...");

microgear.setAlias(ALIAS);

}

void setup()

{

/\* Event listener \*/

microgear.on(MESSAGE,onMsghandler);

microgear.on(CONNECTED,onConnected);

Serial.begin(115200);

Serial.println("Starting...");

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED)

{

delay(250);

Serial.print(".");

}

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

microgear.init(KEY,SECRET,ALIAS);

microgear.connect(APPID);

pinMode(ledPin,OUTPUT);

digitalWrite(ledPin,HIGH); // Turn off LED

}

void loop()

{

if(microgear.connected())

{

microgear.loop();

Serial.println("connect...");

}

else

{

Serial.println("connection lost, reconnect...");

microgear.connect(APPID);

}

delay(250);

}

Figure 5

void onConnected(char \*attribute, uint8\_t\* msg, unsigned int msglen)

{

Serial.println("Connected to NETPIE...");

microgear.setAlias(ALIAS);

}

void setup()

{

/\* Event listener \*/

microgear.on(MESSAGE,onMsghandler);

microgear.on(CONNECTED,onConnected);

Serial.begin(115200);

Serial.println("Starting...");

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED)

{

delay(250);

Serial.print(".");

}

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

microgear.init(KEY,SECRET,ALIAS);

microgear.connect(APPID);

pinMode(ledPin,OUTPUT);

digitalWrite(ledPin,HIGH); // Turn off LED

}

void loop()

{

if(microgear.connected())

{

microgear.loop();

Serial.println("connect...");

}

else

{

Serial.println("connection lost, reconnect...");

microgear.connect(APPID);

}

delay(250);

}

1. Open editor and type the HTML5 code in Figure6

Figure 7

<script src="https://cdn.netpie.io/microgear.js"></script>

<script>

const APPID = "Your App ID";

const KEY = "Your App Key";

const SECRET = "Your App Secret";

const ALIAS = "DigitalOUTPUT\_HTML\_web"; // ชื่อตัวเอง

const thing1 = "M5Stack"; // ชื่อเพื่อนที่จะคุย

function switchPress(logic){

if(logic == 1 ){

microgear.chat(thing1,"ON");

}else if(logic == 0 ){

microgear.chat(thing1,"OFF");

}

}

var microgear = Microgear.create({

key: KEY,

secret: SECRET,

alias : ALIAS

});

microgear.on('message', function(topic,data) {

if(data=="ON"){

document.getElementById("Status").innerHTML = "Load is ON.";

}else if(data=="OFF"){

document.getElementById("Status").innerHTML = "Load is OFF.";

}

});

microgear.on('connected', function() {

microgear.setAlias(ALIAS);

document.getElementById("connected\_NETPIE").innerHTML = "Connected to NETPIE"

});

microgear.on('present', function(event) {

console.log(event);

});

microgear.on('absent', function(event) {

console.log(event);

});

microgear.resettoken(function(err) {

microgear.connect(APPID);

});

</script>

<center>

<h1 id="connected\_NETPIE"></h1>

<button type="button" onclick="switchPress(1)">Turn ON</button>

<button type="button" onclick="switchPress(0)">Turn OFF</button>

<p><strong id="Status">Load is OFF.</strong></p>

</center>

8 Check the results in serial monitor and run HTML5 code on a browser

9.Explain and justify the results

**Section\_\_\_\_\_\_\_\_\_\_\_\_\_date\_\_\_\_\_\_\_\_\_\_**

**Name\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Student ID\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Laboratory 7 M5Stack IoT:**

**Task1: IoT monitoring**

Graded by \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Time\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Task 2: Control LED from the Internet**

Graded by \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Time\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**What do you learn from this lab?**