[嵌入式系統設計](https://flipclass.stust.edu.tw/course/13883)第十四次作業

班級:五專資工三甲

姓名:陳紋誼

學號:5A9G0006

指導老師:吳建中 老師

1. 題目



1. 程式

/\*\*\*\*\*\*\*\*\*

Rui Santos

Complete project details at https://randomnerdtutorials.com

\*\*\*\*\*\*\*\*\*/

#include <WiFi.h>

#include <PubSubClient.h>

#include <Wire.h>

//#include <Adafruit\_BME280.h>

//#include <Adafruit\_Sensor.h>

// Replace the next variables with your SSID/Password combination

const char\* ssid = "TP-LINK\_6F46";

const char\* password = "aa3632aa";

// Add your MQTT Broker IP address, example:

//const char\* mqtt\_server = "192.168.1.144";

const char\* mqtt\_server = "127.0.0.1:1880";

WiFiClient espClient;

PubSubClient client(espClient);

long lastMsg = 0;

char msg[50];

int value = 0;

//uncomment the following lines if you're using SPI

/\*#include <SPI.h>

#define BME\_SCK 18

#define BME\_MISO 19

#define BME\_MOSI 23

#define BME\_CS 5\*/

//Adafruit\_BME280 bme; // I2C

//Adafruit\_BME280 bme(BME\_CS); // hardware SPI

//Adafruit\_BME280 bme(BME\_CS, BME\_MOSI, BME\_MISO, BME\_SCK); // software SPI

float temperature = 0;

float humidity = 0;

// LED Pin

//const int ledPin = 4;

void setup() {

Serial.begin(115200);

// default settings

// (you can also pass in a Wire library object like &Wire2)

//status = bme.begin();

/\*if (!bme.begin(0x76)) {

Serial.println("Could not find a valid BME280 sensor, check wiring!");

while (1);

}\*/

setup\_wifi();

client.setServer(mqtt\_server, 1883);

client.setCallback(callback);

//pinMode(ledPin, OUTPUT);

}

void setup\_wifi() {

delay(10);

// We start by connecting to a WiFi network

Serial.println();

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

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

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

}

void callback(char\* topic, byte\* message, unsigned int length) {

Serial.print("Message arrived on topic: ");

Serial.print(topic);

Serial.print(". Message: ");

String messageTemp;

for (int i = 0; i < length; i++) {

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

messageTemp += (char)message[i];

}

Serial.println();

// Feel free to add more if statements to control more GPIOs with MQTT

// If a message is received on the topic esp32/output, you check if the message is either "on" or "off".

// Changes the output state according to the message

if (String(topic) == "esp32/output") {

Serial.print("Changing output to ");

if (messageTemp == "on") {

Serial.println("on");

//digitalWrite(ledPin, HIGH);

}

else if (messageTemp == "off") {

Serial.println("off");

//digitalWrite(ledPin, LOW);

}

}

}

void reconnect() {

// Loop until we're reconnected

while (!client.connected()) {

Serial.print("Attempting MQTT connection...");

// Attempt to connect

if (client.connect("ESP8266Client")) {

Serial.println("connected");

// Subscribe

//client.subscribe("esp32/output");

} else {

Serial.print("failed, rc=");

Serial.print(client.state());

Serial.println(" try again in 5 seconds");

// Wait 5 seconds before retrying

delay(5000);

}

}

}

void loop() {

if (!client.connected()) {

reconnect();

}

client.loop();

long now = millis();

if (now - lastMsg > 5000) {

lastMsg = now;

// Temperature in Celsius

//temperature = bme.readTemperature();

temperature = random(18, 32);

// Uncomment the next line to set temperature in Fahrenheit

// (and comment the previous temperature line)

//temperature = 1.8 \* bme.readTemperature() + 32; // Temperature in Fahrenheit

// Convert the value to a char array

char tempString[8];

dtostrf(temperature, 1, 2, tempString);

Serial.print("Temperature: ");

Serial.println(tempString);

client.publish("esp32/temperature", tempString);

//humidity = bme.readHumidity();

humidity = random(45, 80);

// Convert the value to a char array

char humString[8];

dtostrf(humidity, 1, 2, humString);

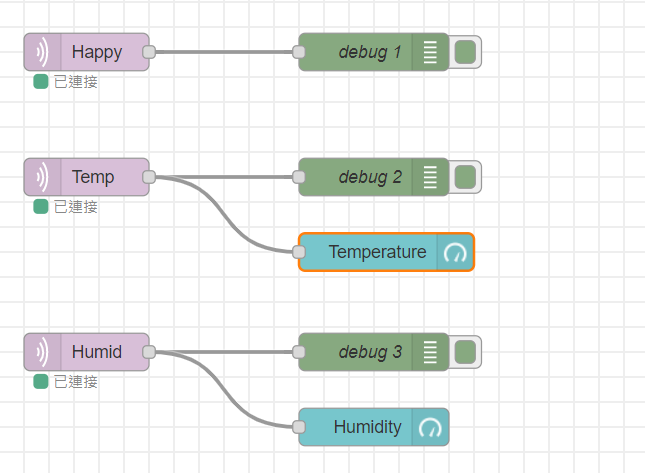
Serial.print("Humidity: ");

Serial.println(humString);

client.publish("esp32/humidity", humString);

}

}



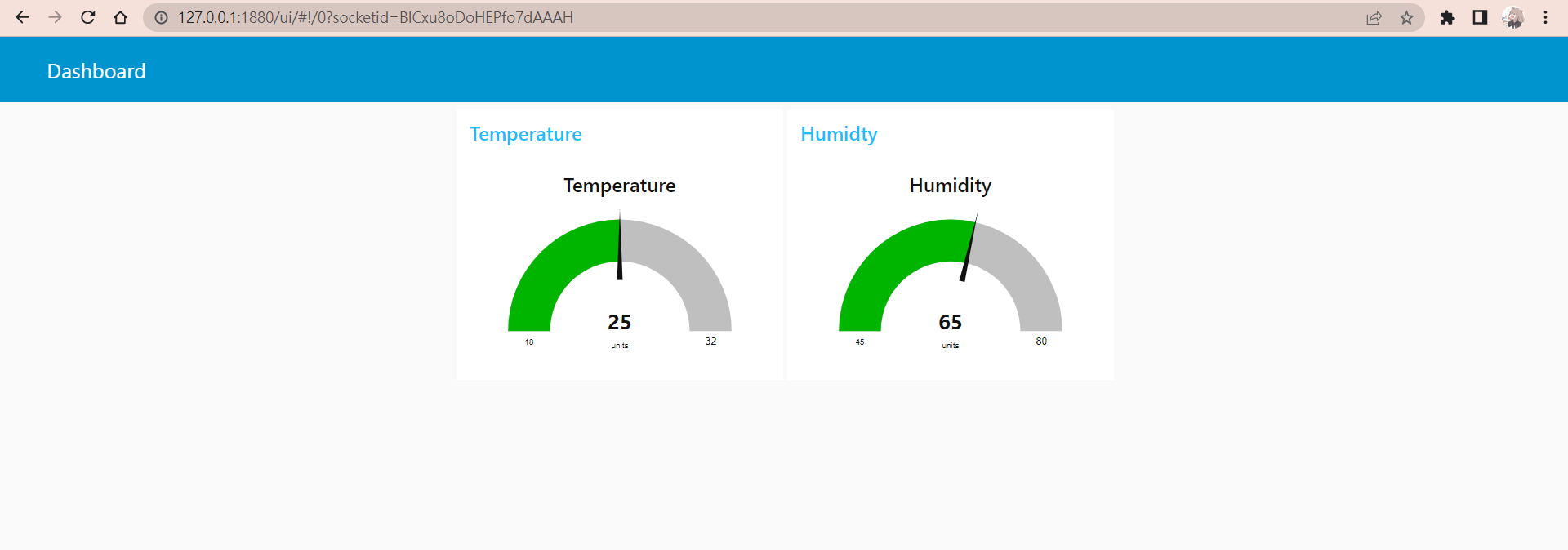
1. 程式說明

將溫度、濕度的資料傳送到自己架設的MQTT server

1. 執行結果

一張含有 文字 的圖片

自動產生的描述



影片連結：