

การเลือกใช้งาน Hardware และ Software สำหรับพัฒนาอุปกรณ์ต้นแบบด้าน IoT

Outline

- แนะนำ Hardware สำหรับงานด้าน IoT
- แนะนำ Software สำหรับงานด้าน IoT
- แนะนำ Cloud สำหรับงานด้าน IoT

องค์ประกอบของ IoT

Sensors &
Actuators

Connectivity

People &
Processes

องค์ประกอบของ IoT

- Sensor
 - อุณหภูมิ, ความชื้น, ความเข้มแสง, ความเร็วลม, ควัน, แก๊ส ฯลฯ
- Actuators
 - Relay, Solenoid valve
- Connectivity
 - Wire – Ethernet, RS485, Serial
 - Wireless – WiFi, Bluetooth, Zigbee, Lora, Cellular (3G or 4G)





Hardware สำหรับงานด้าน IoT

○ Arduino Platform

- เป็น Platform (Hardware และ Software) ที่ได้รับความนิยมที่สุดในปัจจุบัน
- ใช้งานง่าย (Plug & Play), ซอฟต์แวร์ที่ใช้พัฒนาเป็น Opensource (Arduino IDE)
- มีโค้ดตัวอย่างเผยแพร่บนอินเทอร์เน็ตเป็นจำนวนมาก
- มีความหลากหลายทางฮาร์ดแวร์ Arduino Board, Arduino Clone และ Arduino Shields



ที่มา : <https://www.arduino.cc/>

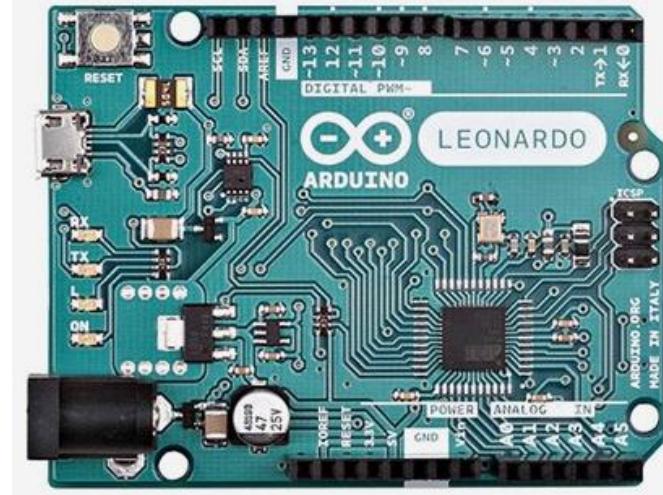
ตัวอย่าง Arduino Board

Arduino UNO : ₧725



ที่มา : <https://goo.gl/Md8EJt>

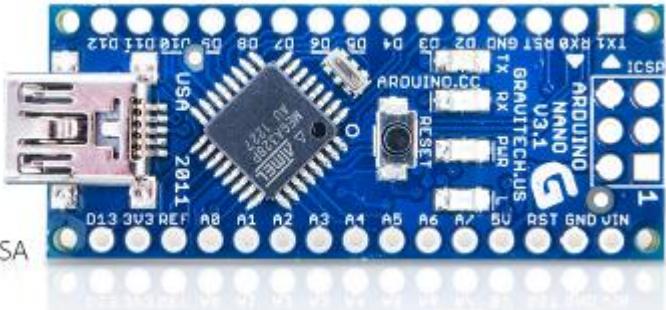
Arduino LEONARDO : ₧690



ที่มา : <https://goo.gl/UnhNya>

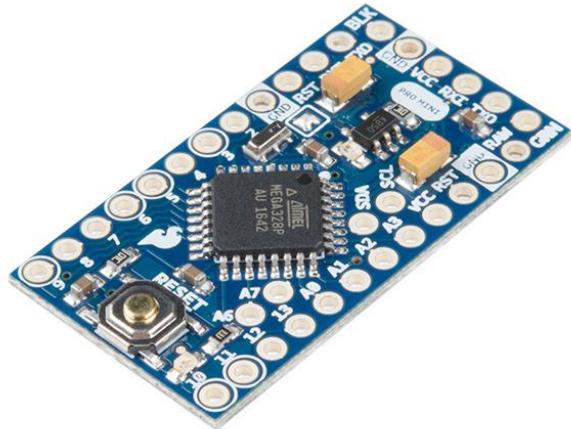
ตัวอย่าง Arduino Board

Arduino Nano 3.1: ₧900



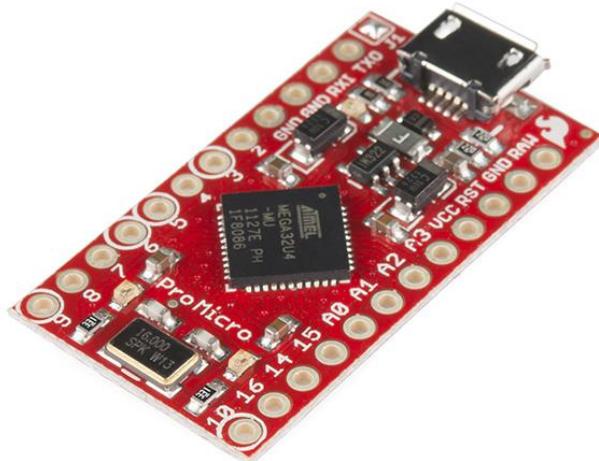
ที่มา : <https://goo.gl/XBoczS>

Arduino Pro Mini : ₧390



ที่มา : <https://goo.gl/cBWKnU>

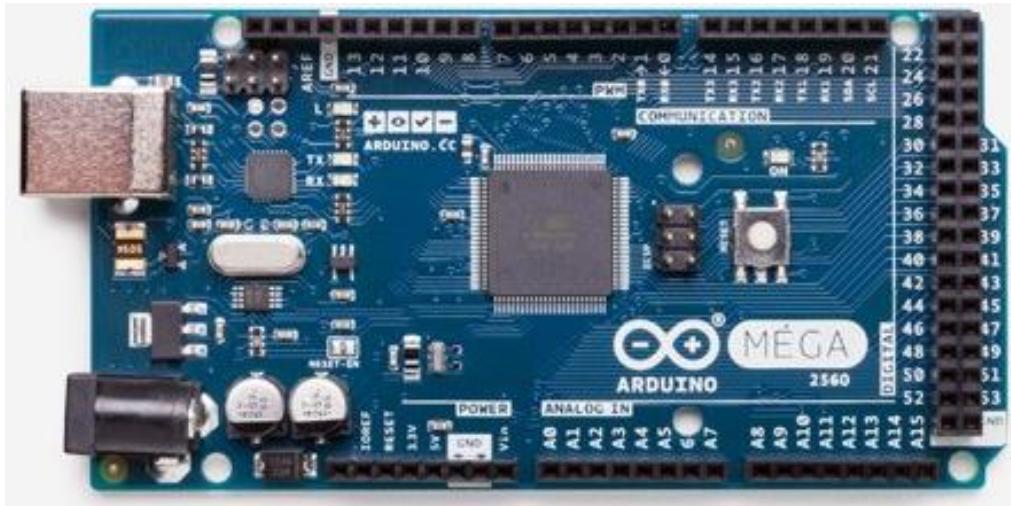
Arduino Pro Micro : ₧690



ที่มา : <https://goo.gl/jBTYZH>

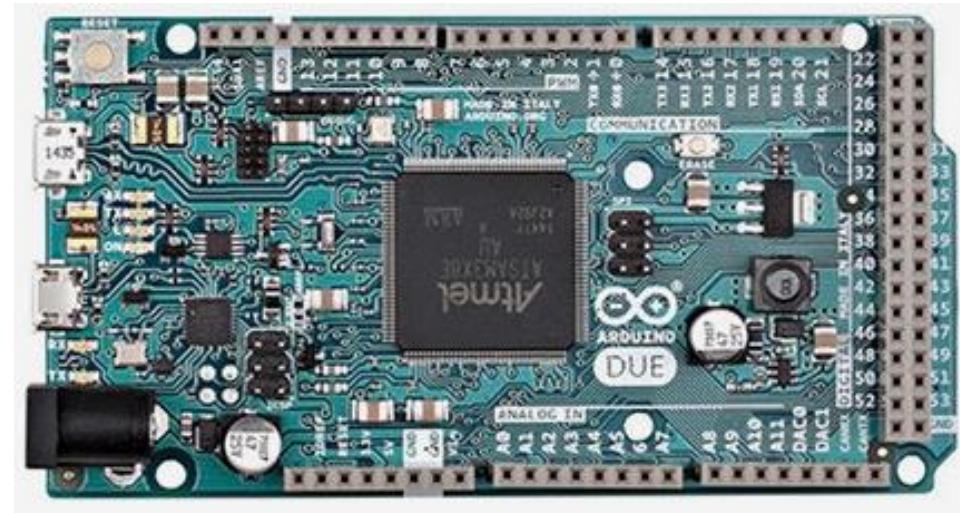
ตัวอย่าง Arduino Board

Arduino MEGA : ₧1650



ที่มา : <https://goo.gl/XbCpQX>

Arduino DUE : ₧1575



ที่มา : <https://goo.gl/6ZEMXV>

ตัวอย่าง Spec Arduino Board

Spec	Arduino UNO	Arduino Pro Micro	Arduino MEGA
MCU	ATmega 328P	ATmega 32U4	ATmega 2560
CPU Speed	16 MHz	16 MHz	16 MHz
Flash Memory	32 KB	32 KB	256 KB
Digital I/O	14 Pins	12 Pins	54 Pins
PWM Output	6 Pins	5 Pins	15 Pins
Analog input	6 Pins	4 Pins	16 Pins
Operating Voltage	5 V	5 V	5 V

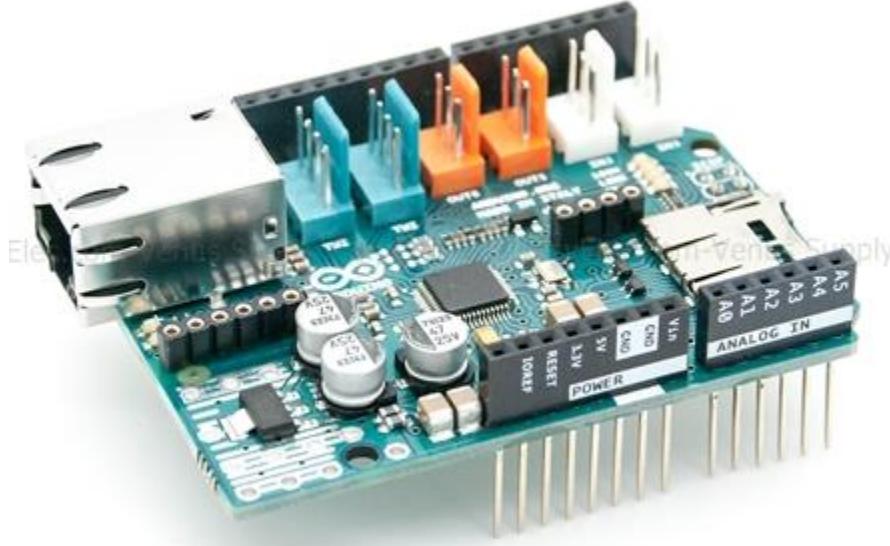
ที่มา : <https://www.arduino.cc/>

เชื่อมต่อ Arduino Board กับ Internet

- Arduino Board + Shield
 - Ethernet Shield, WiFi Shield, GSM Shield
- Arduino Board built-in Ethernet or WiFi
 - Arduino Yun, Arduino Ethernet, Arduino UNO WiFi
- Arduino Board + WiFi Module
 - ATWINC1500 , CC3000, ESP8266-01

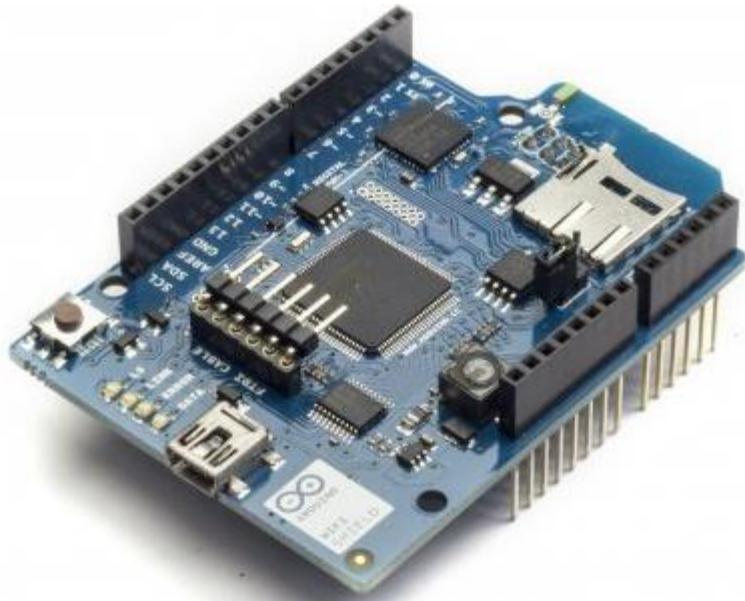
ตัวอย่าง Arduino Shield

Arduino Ethernet Shield : ₧1450



ที่มา : <https://goo.gl/JqsdMy>

Arduino WiFi Shield : ₧3250



ที่มา : <https://goo.gl/bU2Gyv>

Arduino GSM Shield : ₧2650



ที่มา : <https://goo.gl/grvwwE>

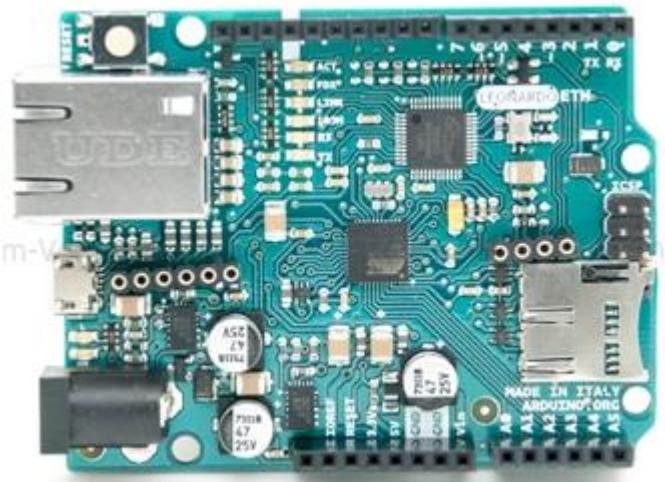
ตัวอย่าง Arduino Board Build-in Ethernet or WiFi

Arduino YUN : ₧2800



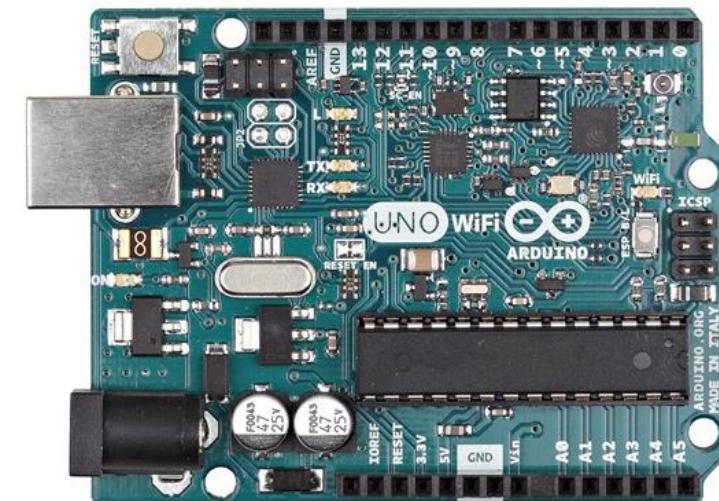
ที่มา : <https://goo.gl/m5V7ZS>

Arduino Ethernet : ₧2350



ที่มา : <https://goo.gl/RELXxG>

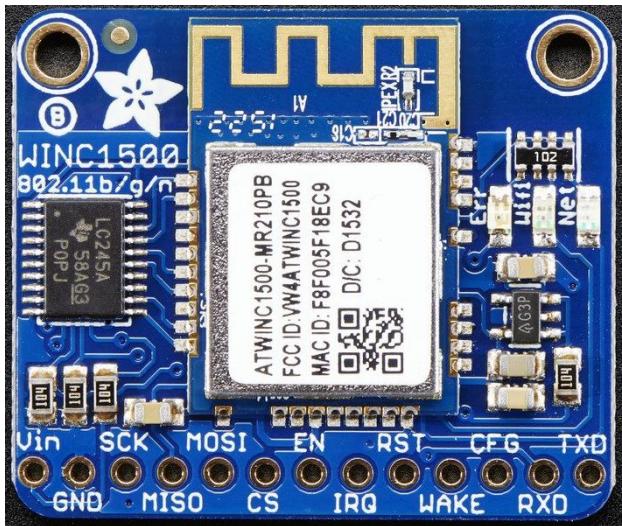
Arduino UNO WiFi : ₧1250



ที่มา : <https://goo.gl/gTFEXY>

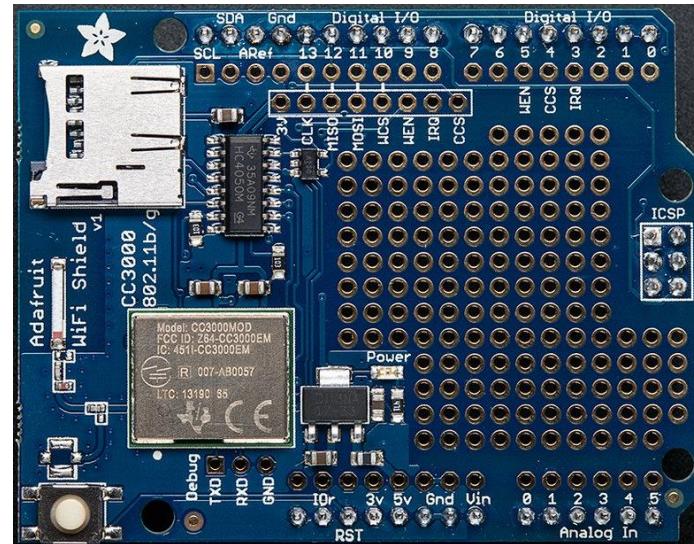
ตัวอย่าง Arduino Board + WiFi Module

Adafruit ATWINC1500 : \$24.95



ที่มา : <https://goo.gl/61Y9JN>

Adafruit HUZZAH CC3000 : \$39.95



ที่มา : <https://goo.gl/T2yc7H>

ESP8266-01 : ₵190



ที่มา : <https://goo.gl/gP6H87>

Hardware สำหรับงานด้าน IoT

- ESP8266 Module : โปรเซสเซอร์ + วงจรสื่อสาร Wireless Transceiver
 - โปรเซสเซอร์ Xtensa LX106 ขนาด 32 บิต ความเร็ว 80MHz ของบริษัท Tensilica
 - สามารถเชื่อมต่อกับ Wi-Fi 2.4GHz (มาตรฐาน IEEE 802.11b/g/n)
 - บริษัท Espressif Systems (Shanghai, China) เป็นผู้ผลิตชิป
 - ได้รับความนิยมอย่างมากในปัจจุบัน



ตัวอย่าง ESP8266 Module



ESP-01



ESP-02



ESP-03



ESP-04



ESP-05



ESP-06



ESP-07



ESP-08



ESP-09



ESP-10



ESP-11

ตัวอย่าง ESP8266 Module

ESP-12



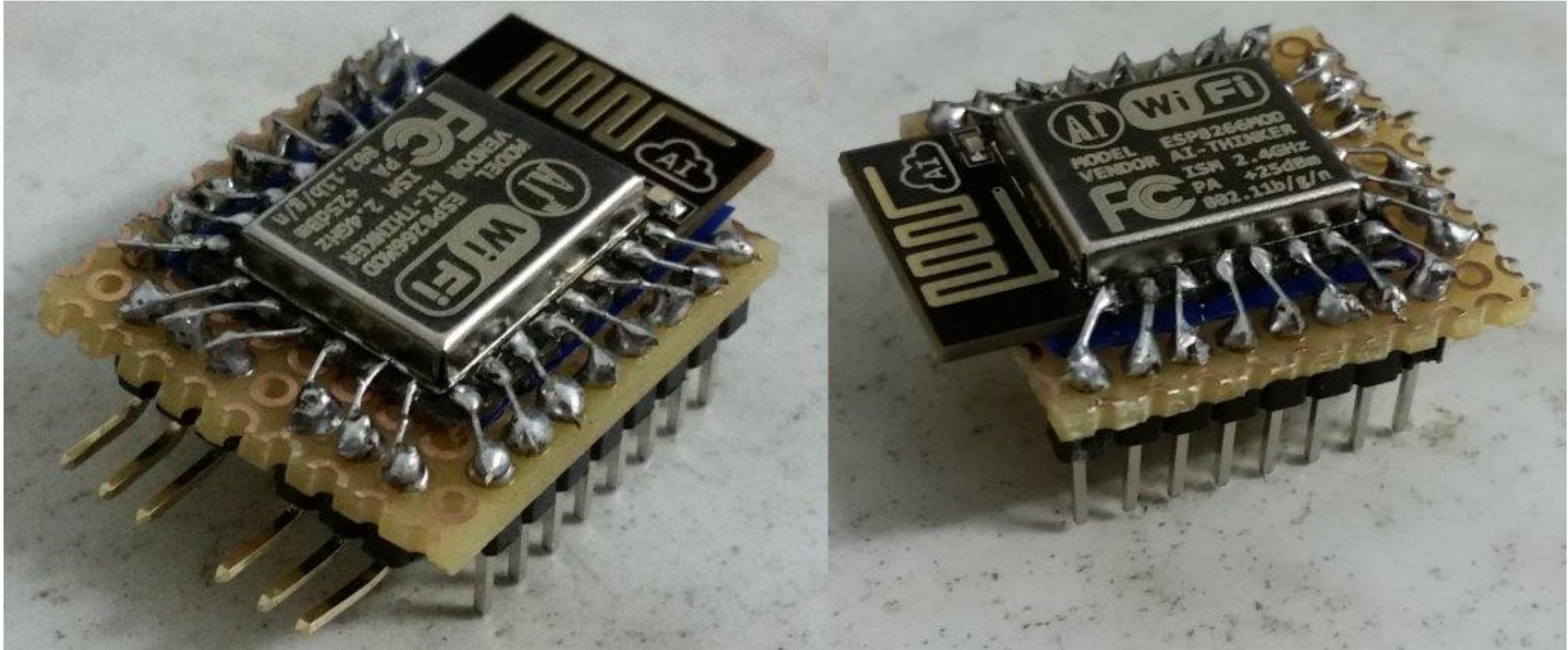
ที่มา : <https://goo.gl/7aZMBB>

ESP-12E

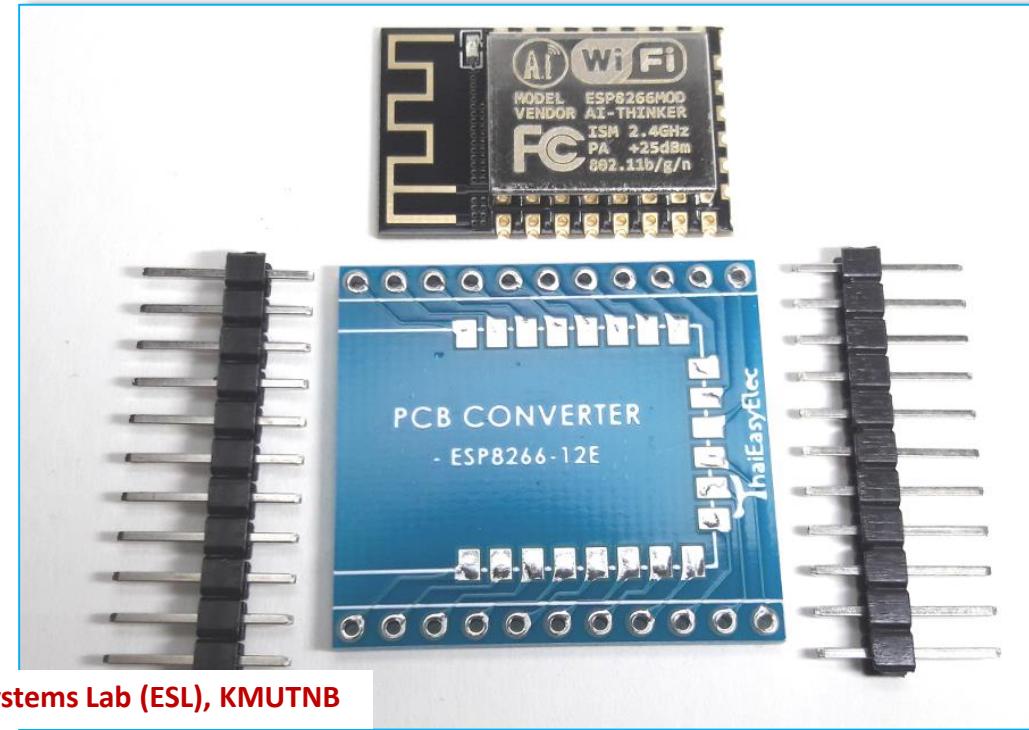
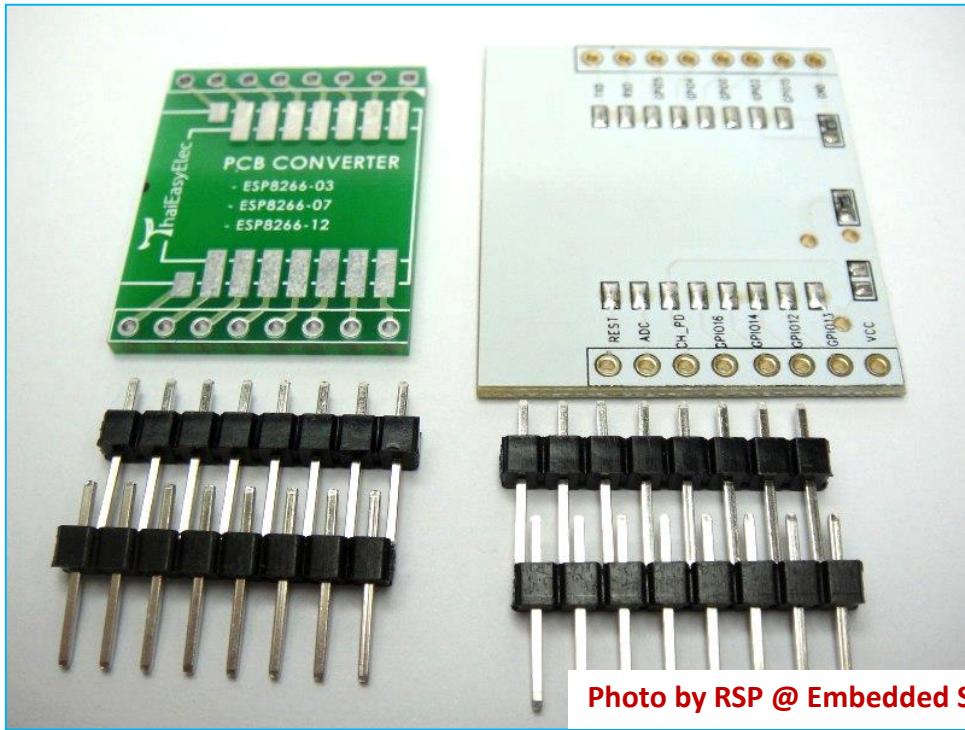


ที่มา : <https://goo.gl/1FS81f>

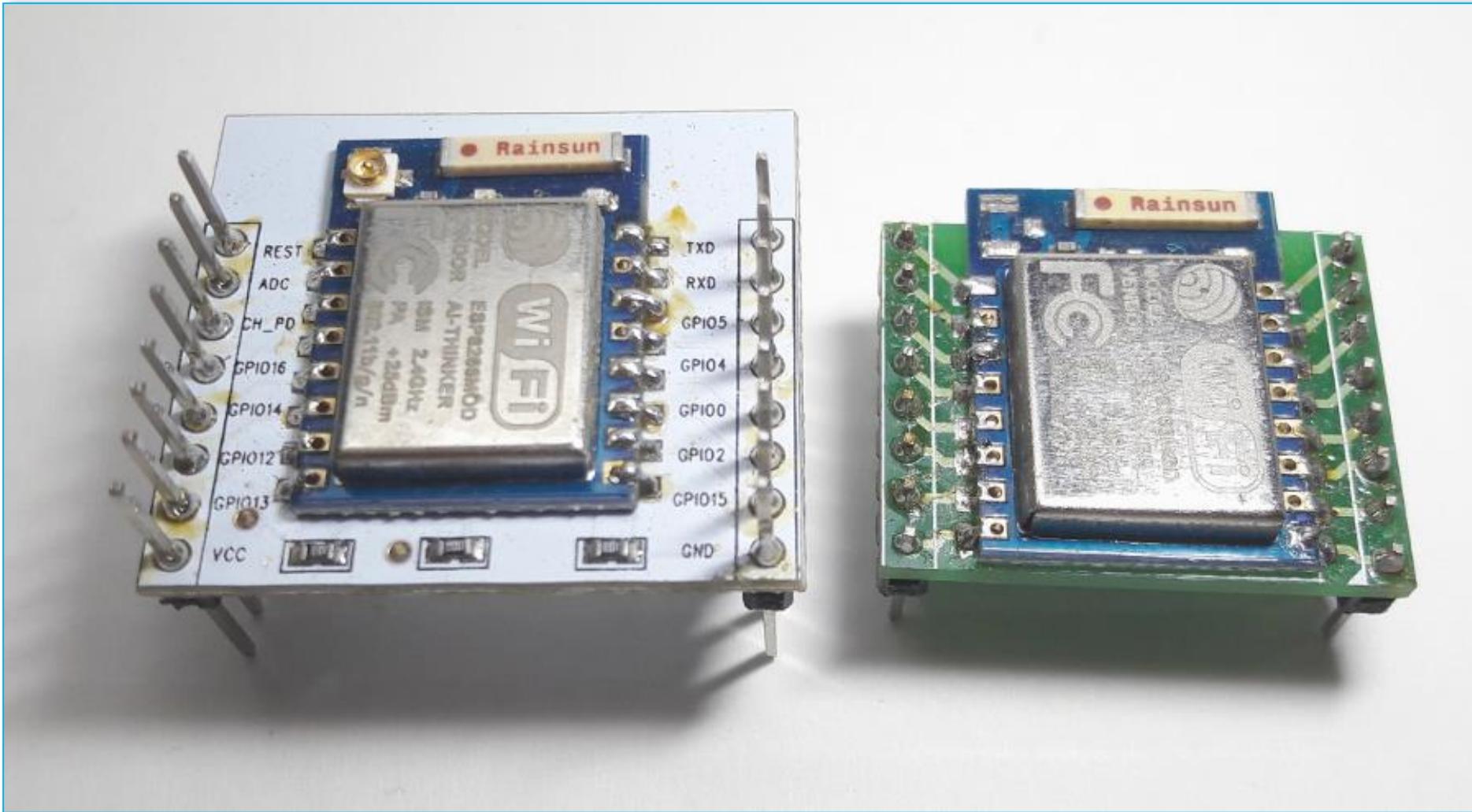
การใช้งาน ESP8266 Module



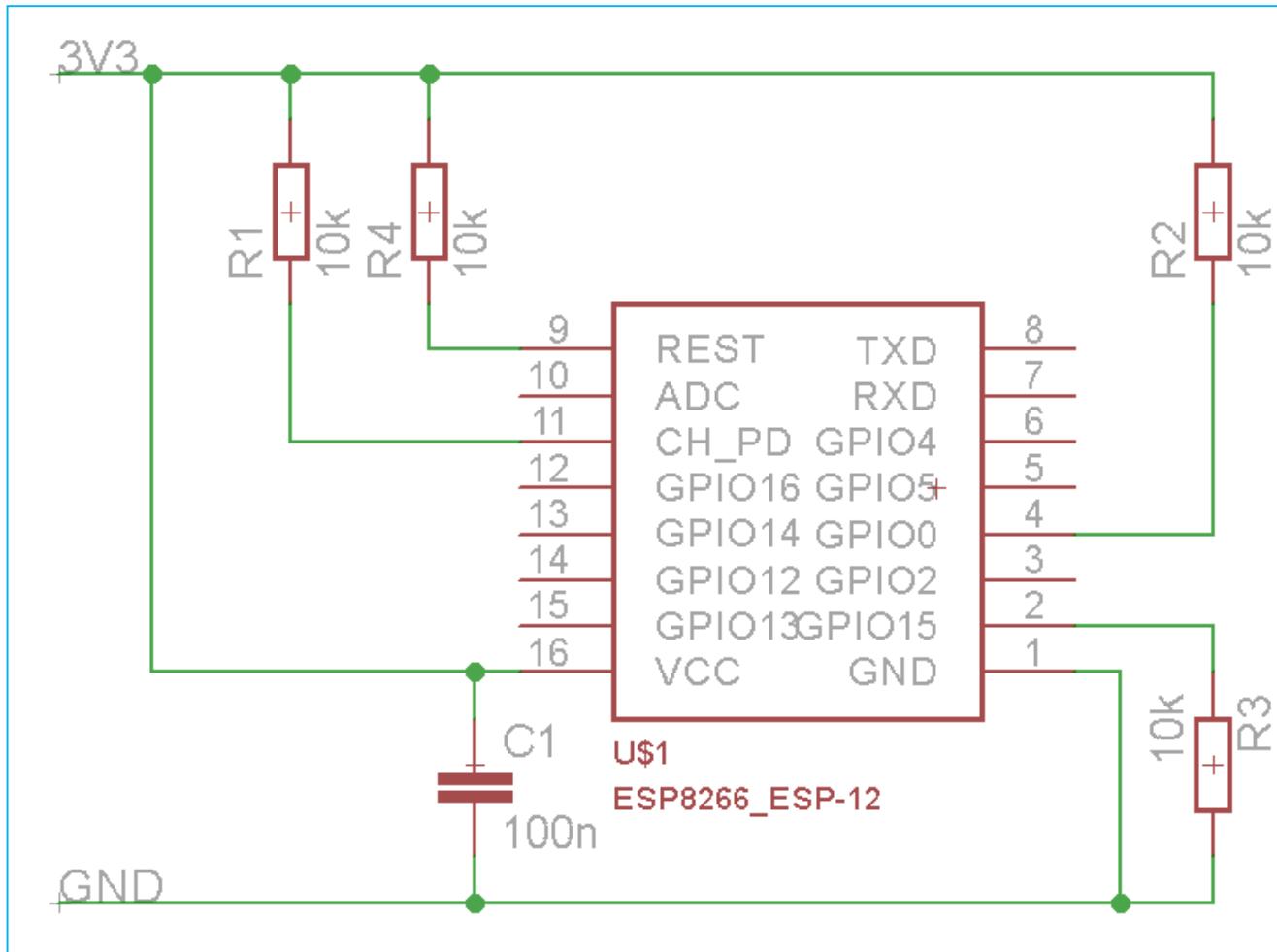
Breakout Board สำหรับ ESP8266 Module



Breakout Board สำหรับ ESP8266 Module

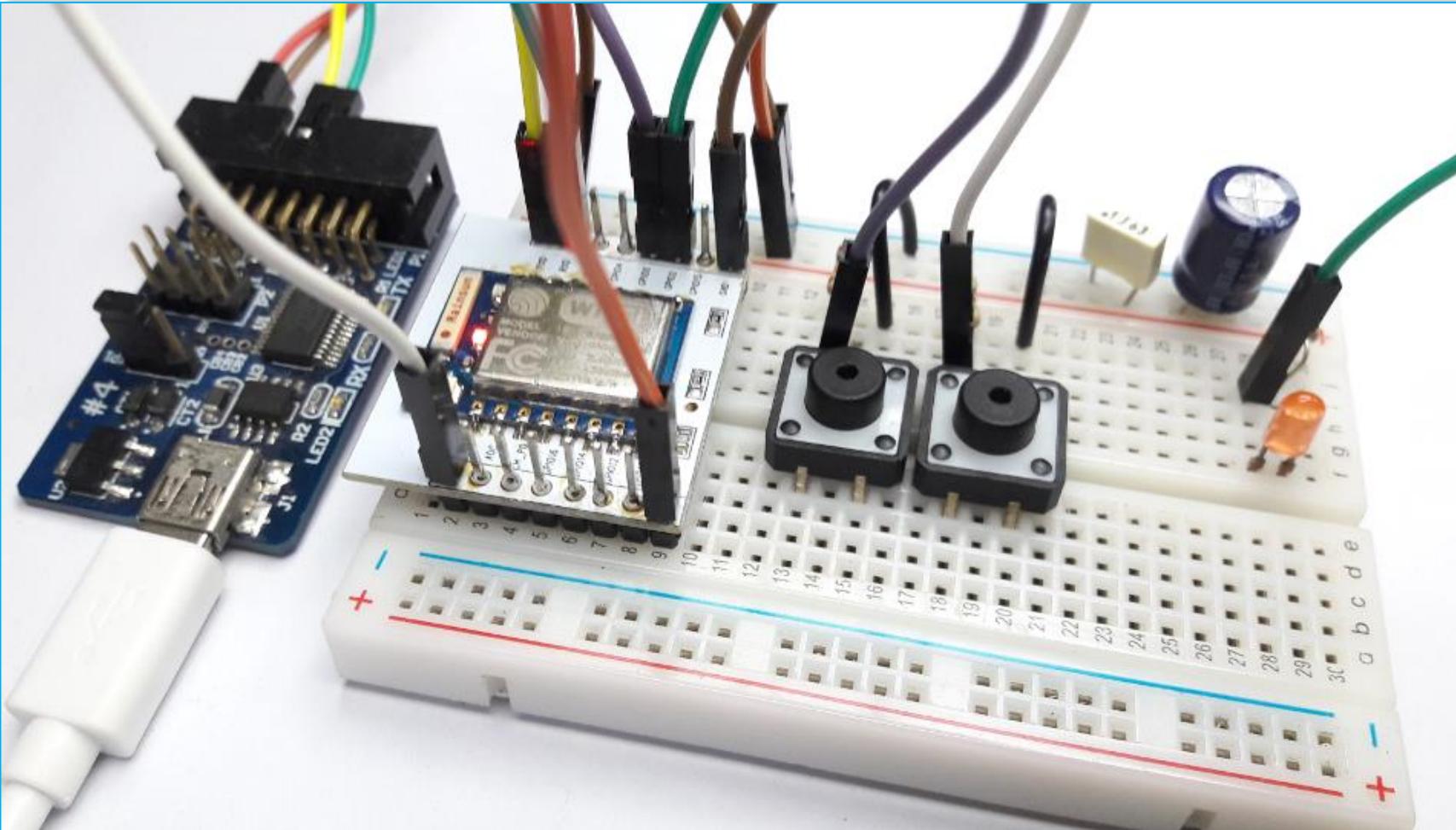


วงจรสำหรับการใช้งาน ESP8266 Module



ที่มา : <http://esp8266.github.io/Arduino/>

การใช้งาน ESP8266 Module



ESP8266 Development Board

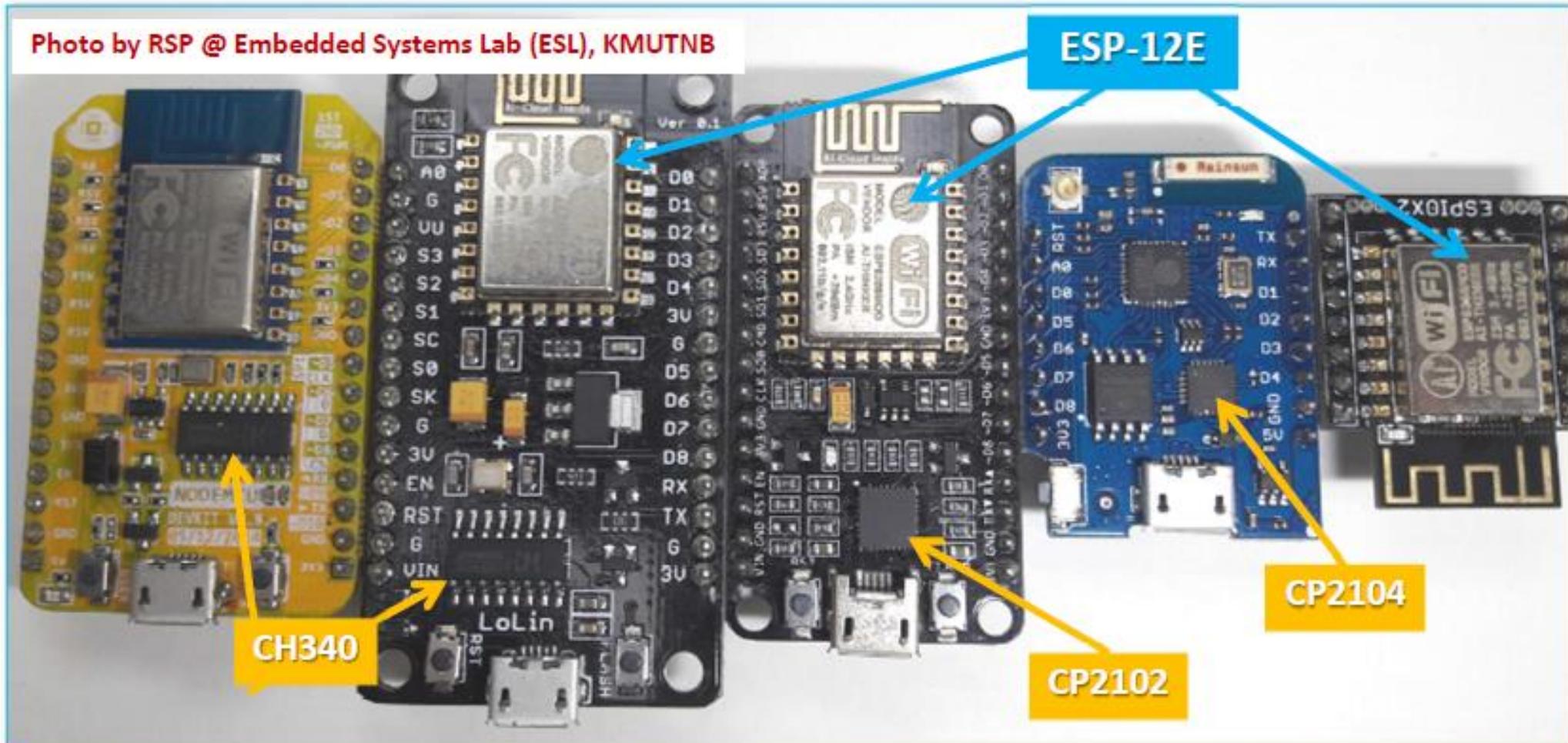
NodeMCU 1st Generation
ESP-12, 4MB flash

LoLin
NodeMCU

NodeMCU 2nd Generation
ESP-12E, 8MB flash

WeMoS D1 Mini Pro
16MB Flash

EXPIOX2
Dev. Board



NodeMCU VS Arduino UNO

	Arduino Uno (ATmega328P 5V/16MHz)	ESP8266EX NodeMCU
MCU / SoC	ATmel ATmega328P 8-bit, RISC	Xtensa LX106 32-bit, RISC
CPU Speed	16 MHz	80 MHz
Data RAM	2 KB	96 KB
Instruction RAM	-	64 KB
On-chip Flash Memory	32 KB	-
External Flash Memory	-	4 or 8 MB
Wi-Fi	-	IEEE 802.11 b/g/n

	Arduino Uno (ATmega328P 5V/16MHz)	NodeMCU (ESP8266EX)
User SPI (HW)	1	1
I2C (HW)	1	1
UART (HW)	1	2
ADC resolution	10-bit	10-bit
#Analog Inputs	6	1
Digital I/O Logic Level	5V	3.3V
#PWM outputs (HW)	6	4
DC current per I/O pin	40 mA (max.)	12 mA (max.)
Device Voltage Supply	4.5V .. 5.5V	3.0V to 3.6V
On-board USB-to-Serial chip	yes	yes

WiFi Development Board

WiPy 2.0 : ₧900



ที่มา : <https://goo.gl/cYkSBS>

Spark Core : \$39.95



ที่มา : <https://goo.gl/VHt4pU>

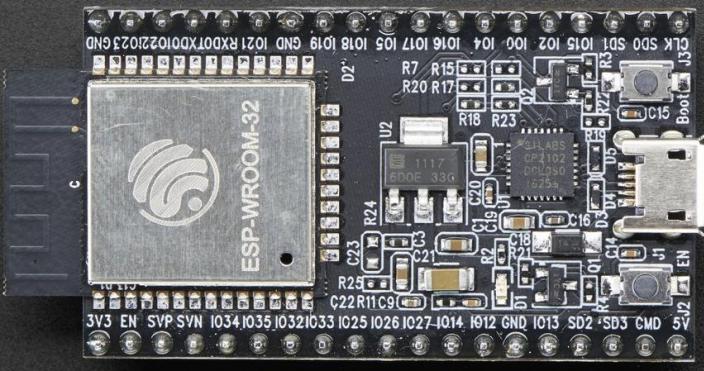
Espruino WiFi : \$34.95



ที่มา : <https://goo.gl/ekKbYB>

WiFi Development Board

ESP32 Dev Board : \$15



ลิ้งค์ : <https://goo.gl/cYkSBS>

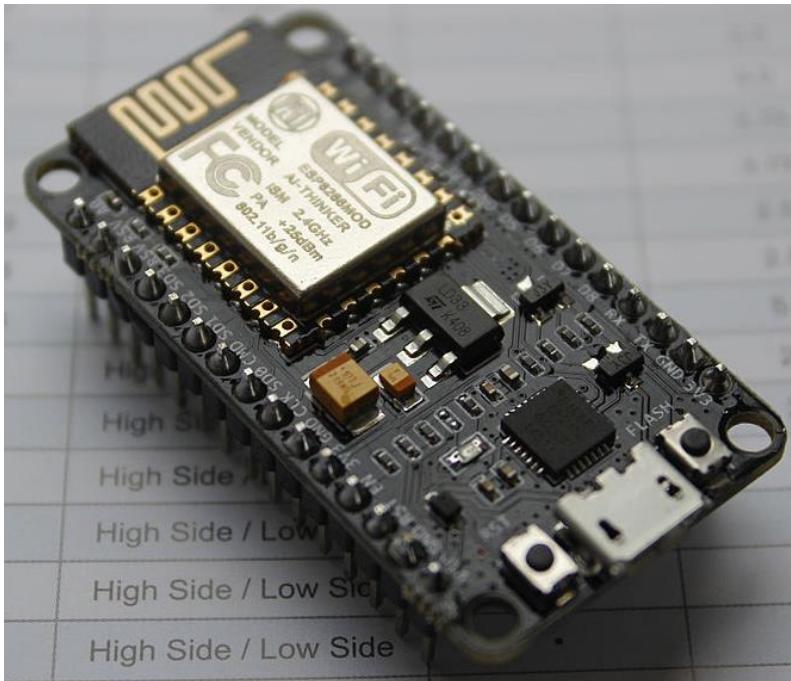
Node32S : ₧690



ลิ้งค์ : <https://goo.gl/SgdnuK>

Hardware สำหรับการอบรม

Node mcu Devkit v1.0 : ₧280 ไม่รวม Vat 7%



ที่มา : <https://goo.gl/92816Y>

Wemos D1 mini : ₧250 ไม่รวม Vat 7%



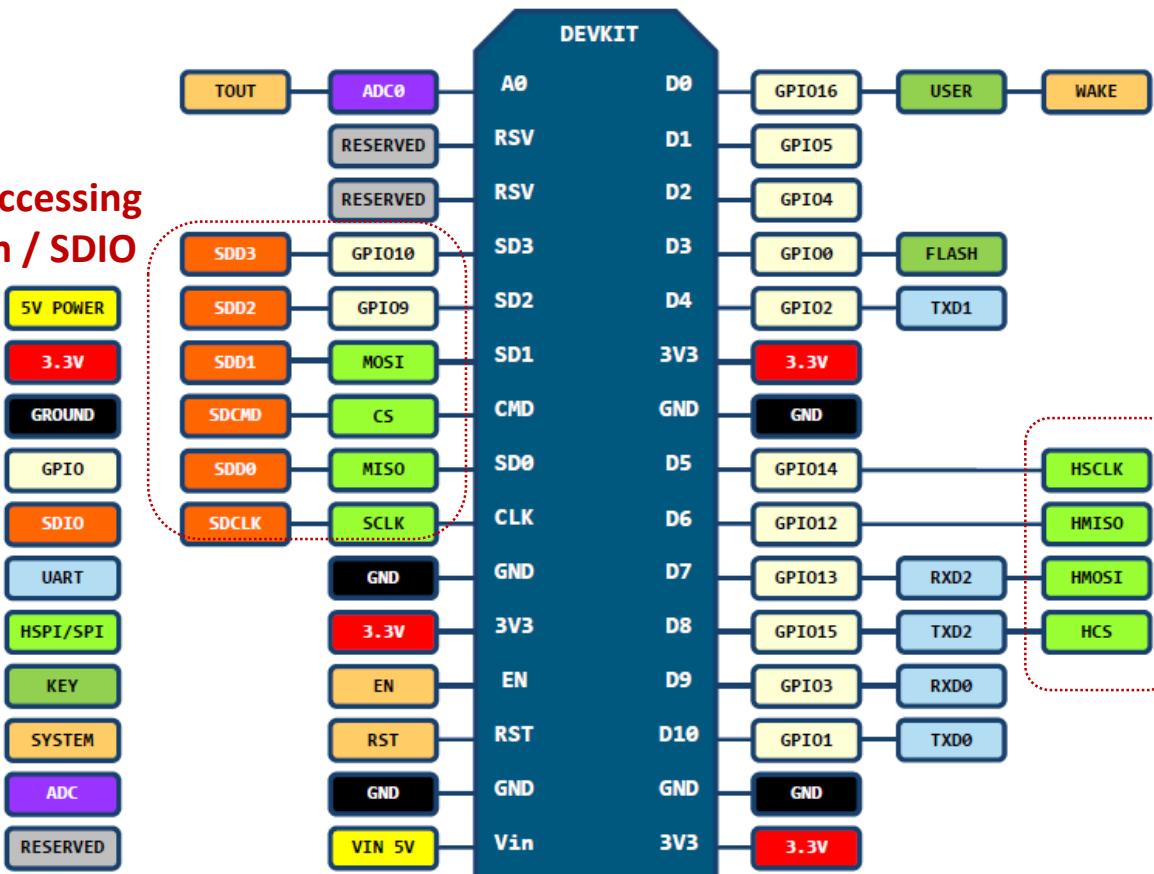
ที่มา : <https://goo.gl/na3TwX>

NodeMCU Pin Map

PIN DEFINITION

Source: <https://github.com/nodemcu/nodemcu-devkit-v1.0>

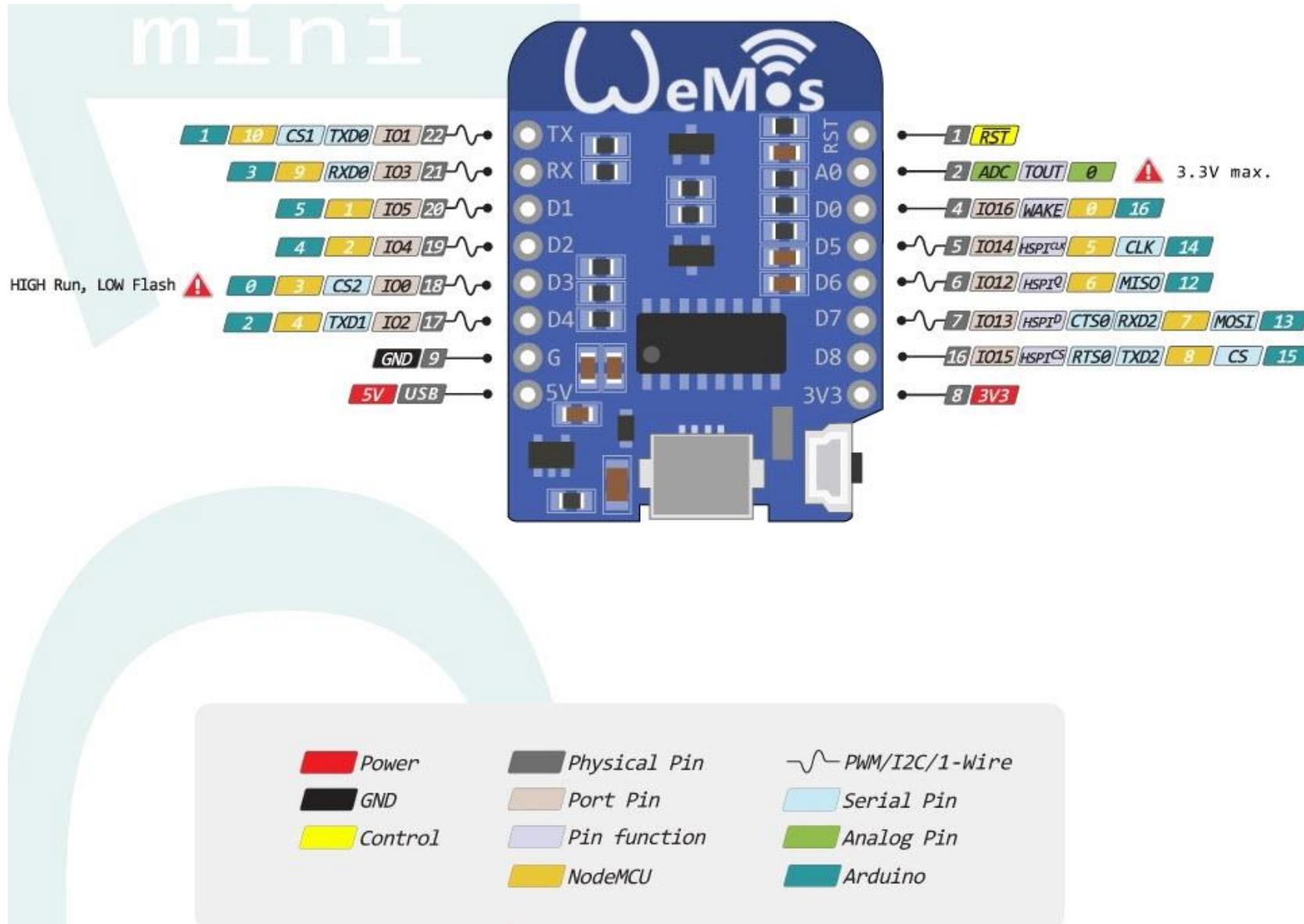
SPI for accessing
SPI Flash / SDIO



HSPI
(User SPI)

D0(GPIO16) can only be used as gpio read/write, no interrupt supported, no pwm/i2c/ow supported.

Wemos D1 Mini Pin Map



Software สำหรับงานด้าน IoT

- ภาษาที่ใช้พัฒนา NodeMCU
 - Native C / ESP8266 SDK
 - <https://github.com/pfalcon/esp-open-sdk>
 - C/C++ / Arduino ESP8266
 - <https://github.com/esp8266/Arduino>
 - Lua for ESP8266 (Scripting Language)
 - <https://nodelua.org/>
 - Micro-Python for ESP8266 (Scripting Language)
 - <http://micropython.org/>
 - JavaScript (Espruino Firmware)
 - <http://www.espruino.com/>
- แต่ที่นิยมแพร่หลายคือ การเขียนโปรแกรมด้วย Arduino IDE

ตัวอย่างโค้ด Lua : Blink LED

```
led_pin = 4                                -- use NodeMCU IO index 4 => GPIO2 pin
gpio.mode( led_pin, gpio.OUTPUT )             -- output mode
state = false
count = 0
repeat
    gpio.write( led_pin, gpio.HIGH )          -- update output to high
    tmr.delay( 500000 )                      -- delay for 0.5 sec
    gpio.write( led_pin, gpio.LOW )           -- update output to low
    tmr.delay( 500000 )                      -- delay for 0.5 sec
    count = count + 1                        -- increment counter by 1
    tmr.wdclr()                            -- reset watchdog timer
until count > 100
```

ตัวอย่างโค้ด Micro-Python : Blink LED

```
import time, machine
from machine import Pin
p = Pin(2, machine.Pin.OUT)          # GPIO-2 as output
for i in range(10):
    p.low()                         # output low
    time.sleep_ms(500)               # sleep 500 msec
    p.high()                        # output high
    time.sleep_ms(500)               # sleep 500 msec
```

ตัวอย่างโค้ด JavaScript : Blink LED

```
var led=2;          //D4
var state = false;
var count=0;
pinMode(led, 'output');
var t=setInterval( function() {
    state = !state;
    digitalWrite(led, state);
    console.log( 'state: ' + ((state) ? 'T':'F')
                 + ', count: ' + count++ );
    if (count > 10) {
        clearInterval(t);
        console.log('LED flashing stopped..');
    }
}, 500);
```

ตัวอย่างโค้ด Arduino IDE : Blink LED

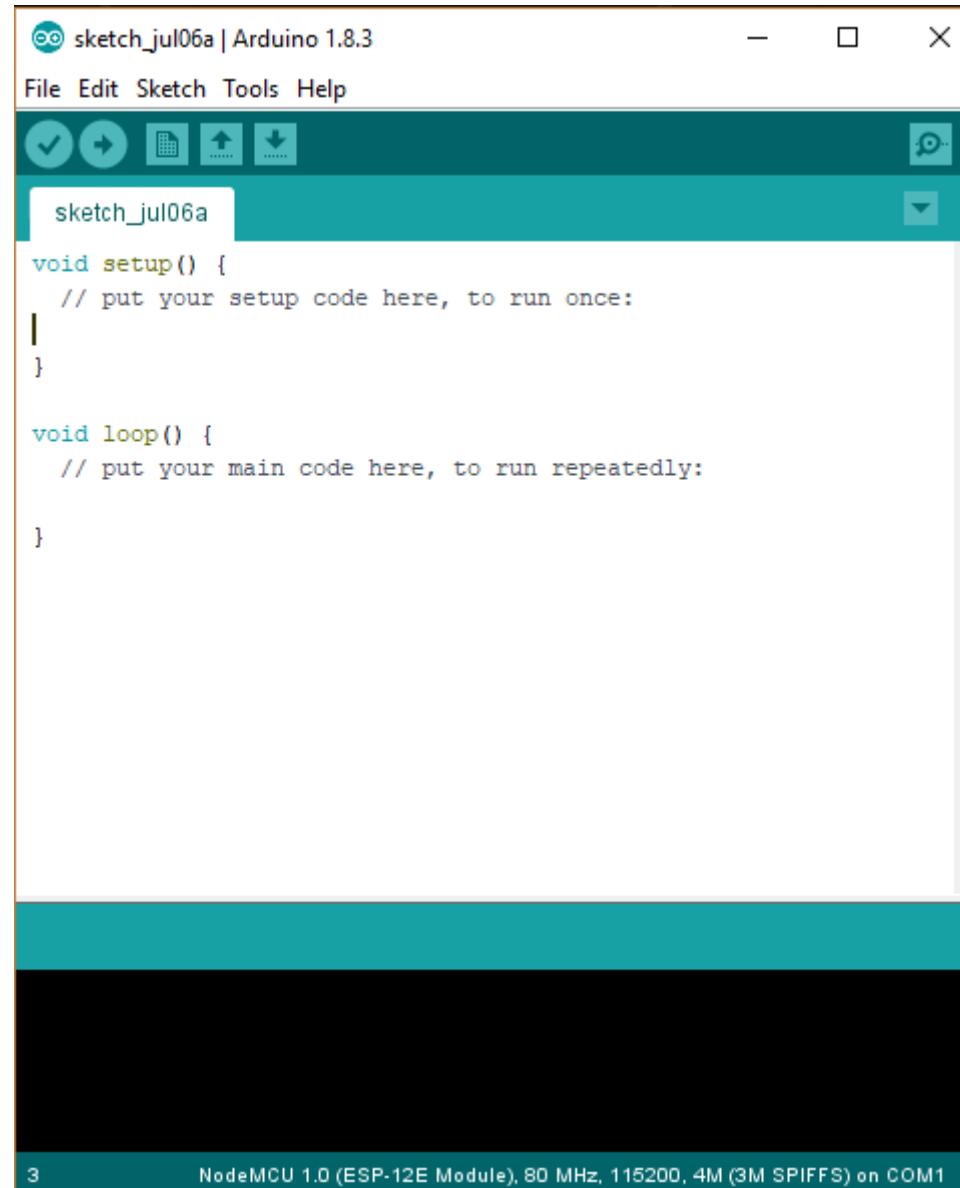
```
#define LED_PIN 2 // GPIO 2 or D4 pin (built-in LED on ESP-12E module)

int state = 0;

void setup() {
    Serial.begin(115200); // Open Serial Port
    Serial.println("\n\n\n\n");
    pinMode(LED_PIN, OUTPUT); // set output mode
}

void loop() {
    Serial.println(state); // show the current value of the state variable
    digitalWrite(LED_PIN, state); // update the output using the state variable
    state ^= 1; // toggle the state variable
    delay(500); // wait for 500 ms
}
```

Aduino IDE สำหรับ ESP8266



The screenshot shows the Arduino IDE 1.8.3 interface with a sketch titled "sketch_jul06a". The code consists of two main functions: setup() and loop(). The setup() function contains a single line of code: "put your setup code here, to run once:". The loop() function also contains a single line of code: "put your main code here, to run repeatedly:". The IDE has a dark theme with teal highlights for the tabs and status bar.

```
sketch_jul06a | Arduino 1.8.3
File Edit Sketch Tools Help
sketch_jul06a
void setup() {
  // put your setup code here, to run once:
}

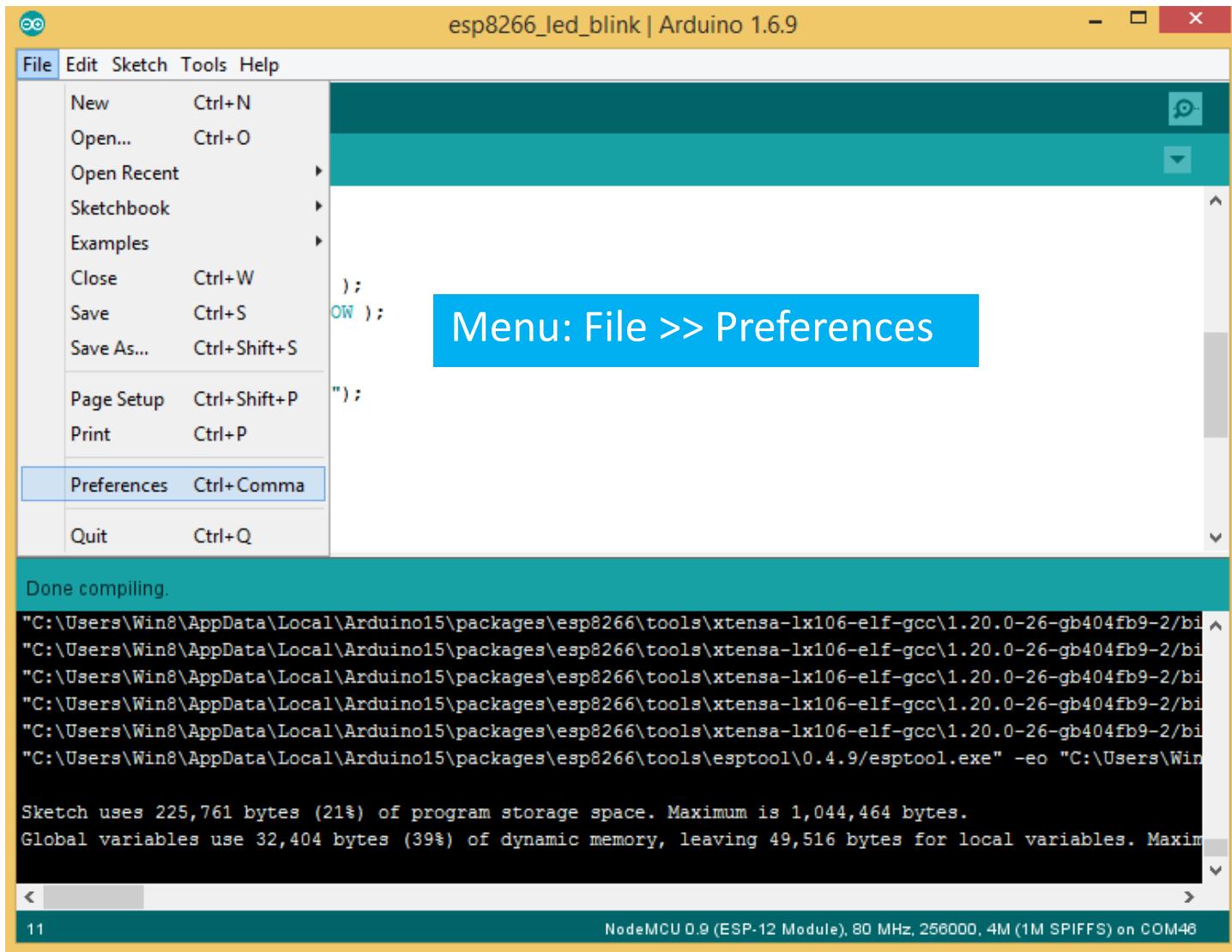
void loop() {
  // put your main code here, to run repeatedly:
}
```

3 NodeMCU 1.0 (ESP-12E Module), 80 MHz, 115200, 4M (3M SPIFFS) on COM1

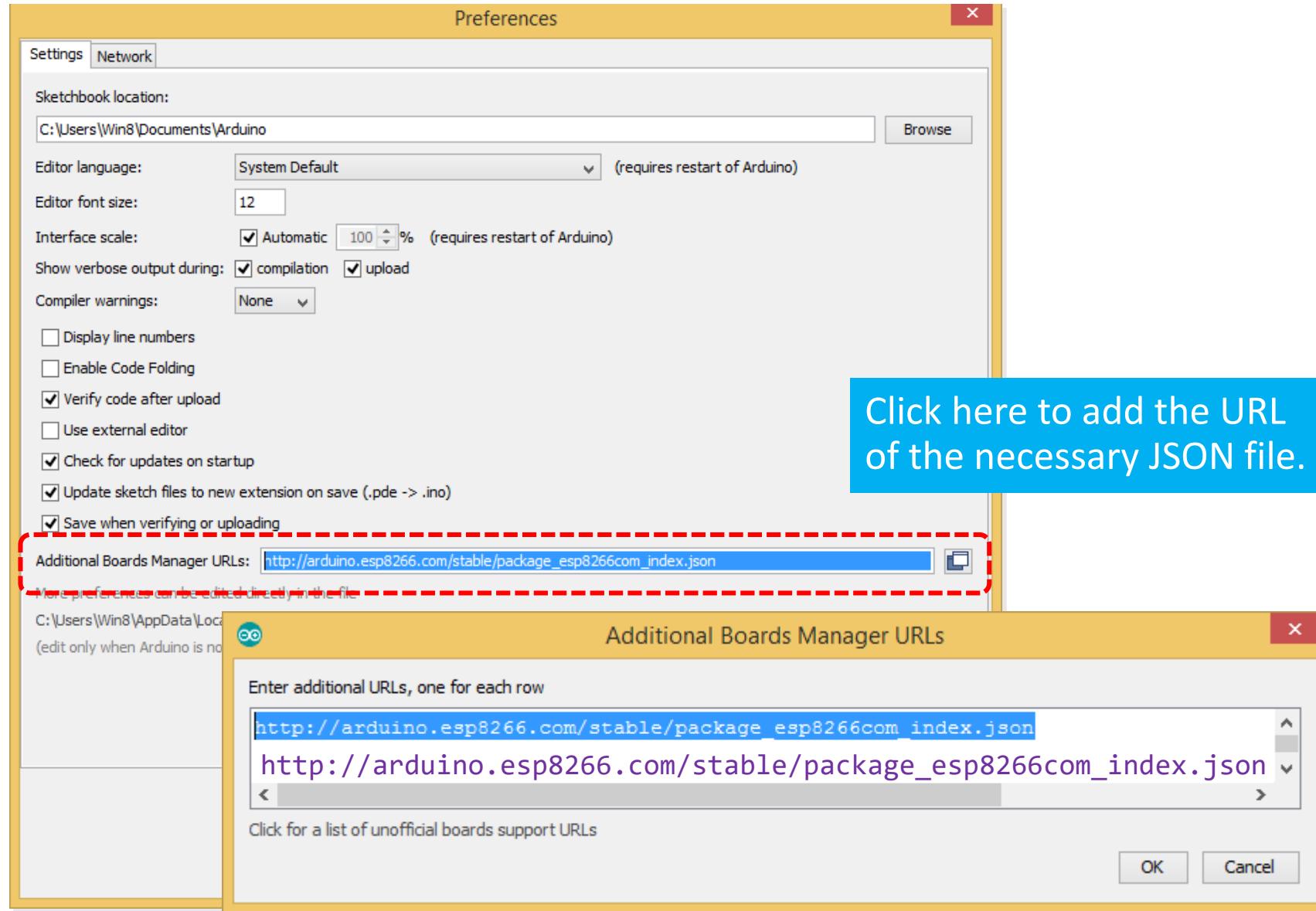
API และ Library สำหรับ ESP8266

- Digital I/O : pinMode(), digitalWrite(), digitalRead()
- Analog Input : analogRead()
- PWM : analogWrite()
- Serial (up to 115200) : Serial.begin(), Serial.println()
- SPI, i2c (Wire) & Software Serial
- Servo, WiFi, EEPROM, Ticker, ESP-specific APIs

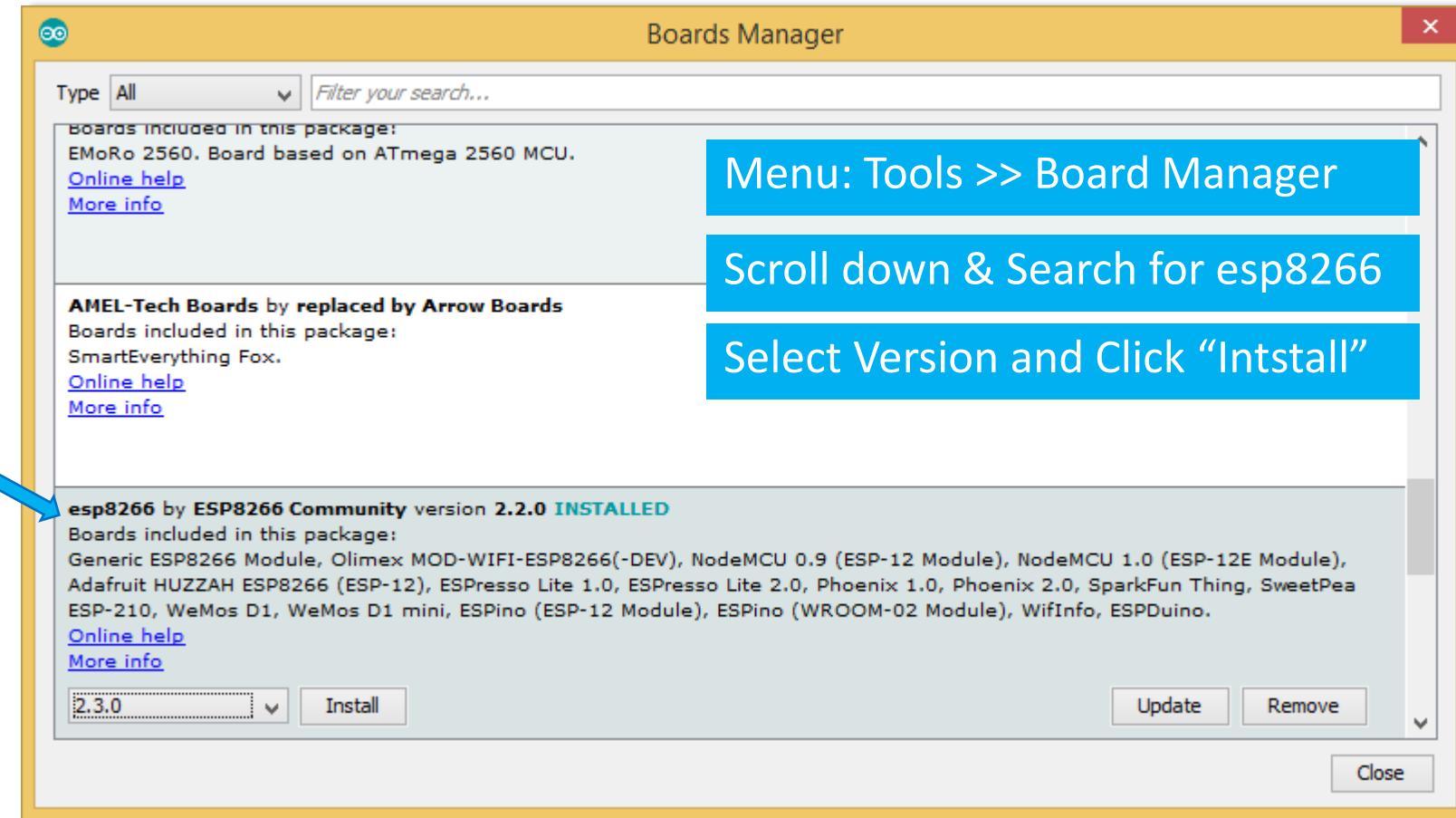
ติดตั้ง ESP8266 Board Manager สำหรับ Arduino IDE



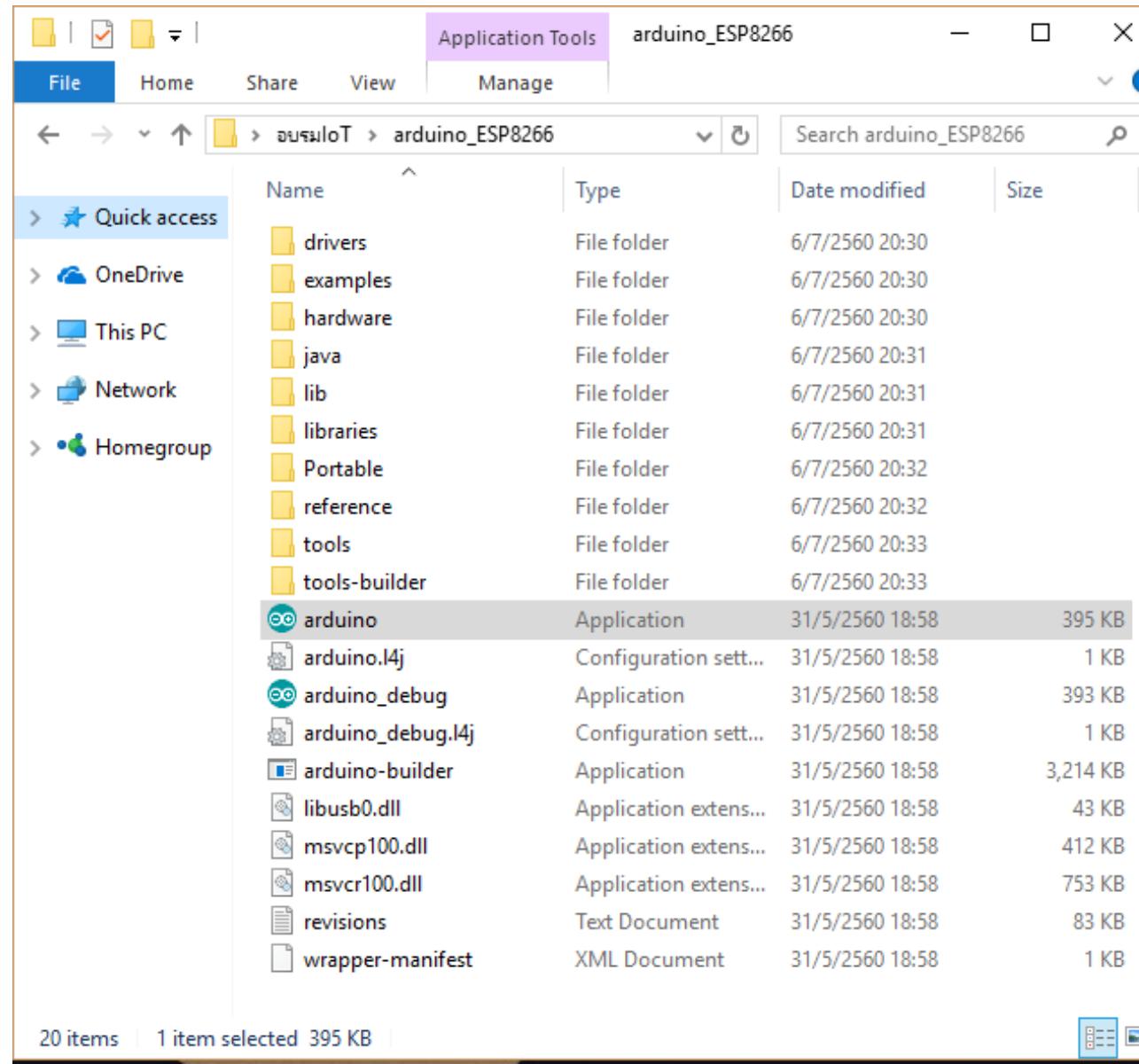
ติดตั้ง ESP8266 Board Manager สำหรับ Arduino IDE



ติดตั้ง ESP8266 Board Manager สำหรับ Arduino IDE



Aduino IDE สำหรับการอบรม



Load file ที่ใช้ในการอบรม

○ เปิด browser เข้าที่ <https://github.com/batman1292/IoTWorkshop-Code>

The screenshot shows the GitHub repository page for 'IoTWorkshop-Code' owned by 'batman1292'. The repository has 3 commits, 1 branch, 0 releases, and 1 contributor. It has 1 unwatched star and 0 forks. The 'Clone or download' button is highlighted in green. A modal window titled 'Clone with HTTPS' is open, showing the URL <https://github.com/batman1292/IoTWorkshop-Code>. The commit history lists several files including esp8266_bh1750_lib_demo-1, esp8266_bh1750_lib_demo-2, esp8266_blink_LED, esp8266_connect_WiFi_AP, esp8266_dht11_lib_demo-1, esp8266_dht11_lib_demo-2, esp8266_pcf8574a_lcd_lib_demo-1, and esp8266_pcf8574a_lcd_lib_demo-2.

No description, website, or topics provided.

Add topics

3 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

batman1292 fix code

- esp8266_bh1750_lib_demo-1 frist commit
- esp8266_bh1750_lib_demo-2 fix code
- esp8266_blink_LED frist commit
- esp8266_connect_WiFi_AP frist commit
- esp8266_dht11_lib_demo-1 frist commit
- esp8266_dht11_lib_demo-2 fix code
- esp8266_pcf8574a_lcd_lib_demo-1 frist commit
- esp8266_pcf8574a_lcd_lib_demo-2 fix code

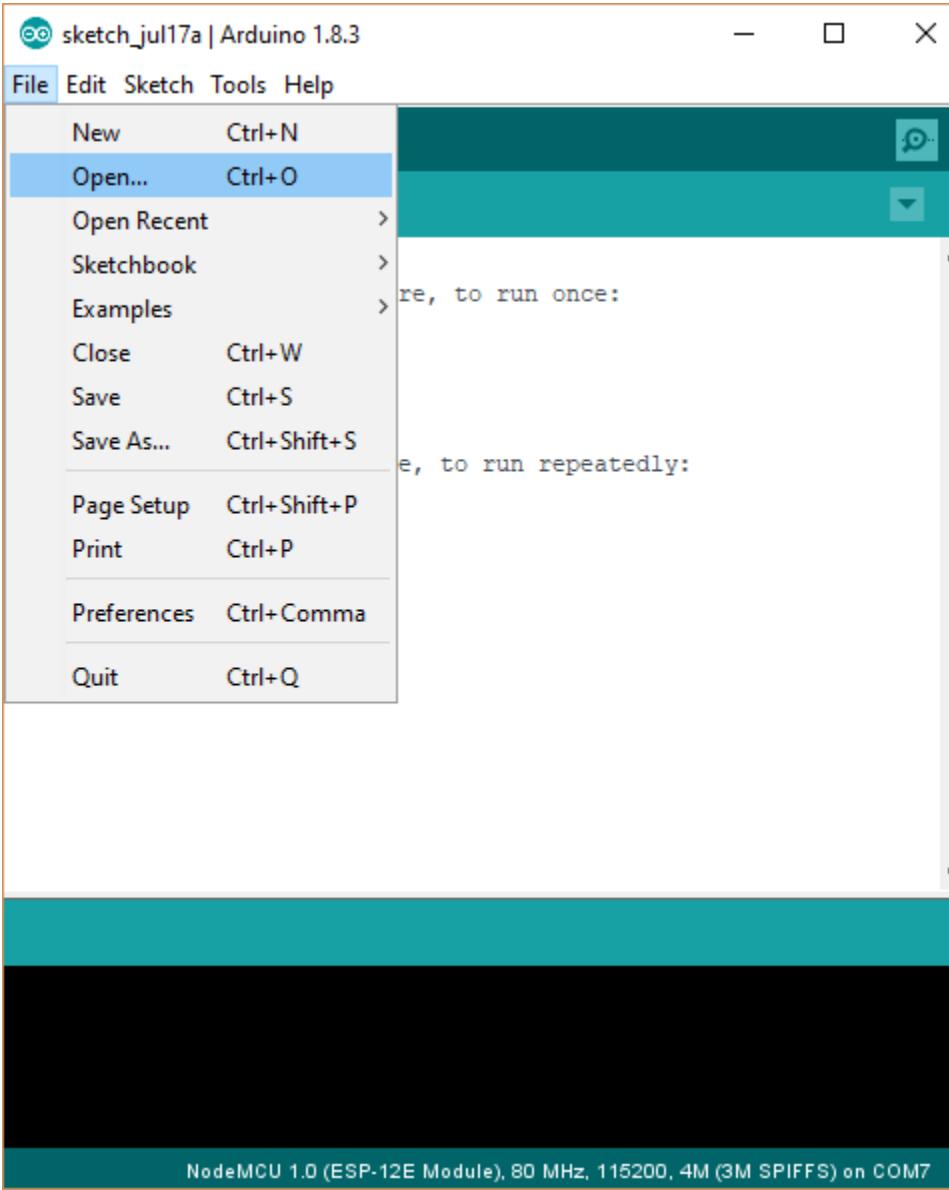
Clone with HTTPS Use SSH
Use Git or checkout with SVN using the web URL.
<https://github.com/batman1292/IoTWorkshop-Code>

Open in Desktop Download ZIP 13 minutes ago

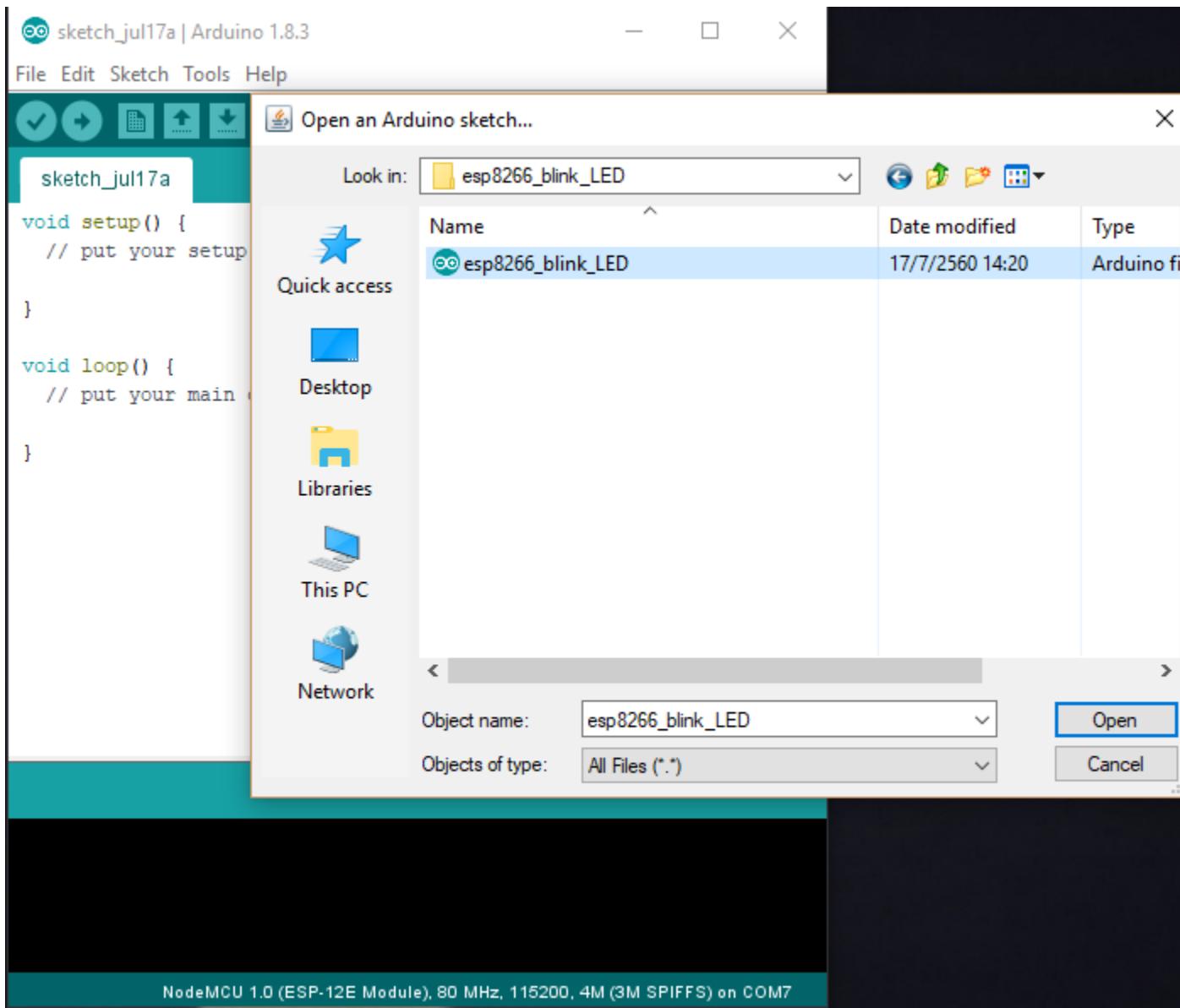
13 minutes ago a minute ago 13 minutes ago a minute ago

<https://github.com/batman1292/IoTWorkshop-Code/archive/master.zip>

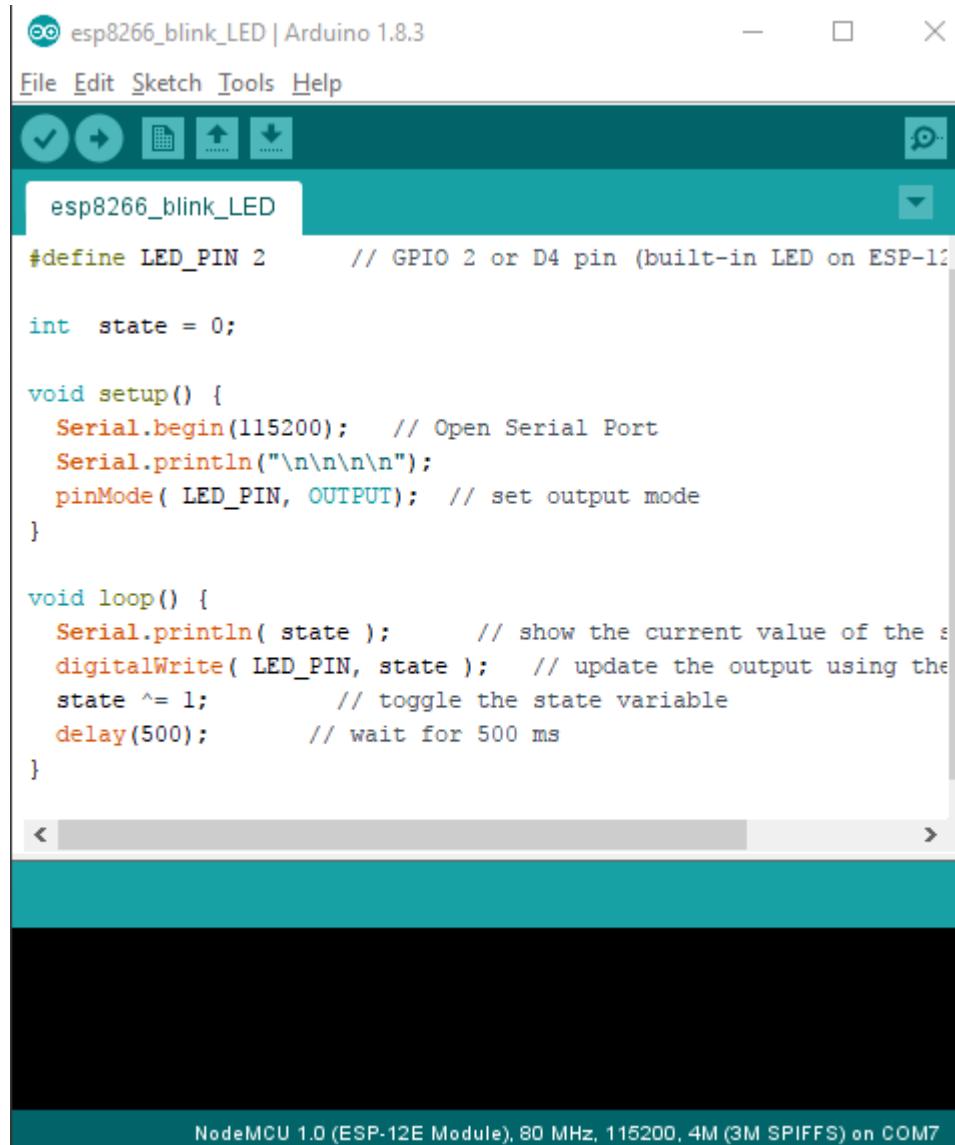
เปิด Code จาก Arduino IDE



ทดสอบการใช้งาน Blink LED



ทดสอบการใช้งาน Blink LED



The screenshot shows the Arduino IDE interface with the following details:

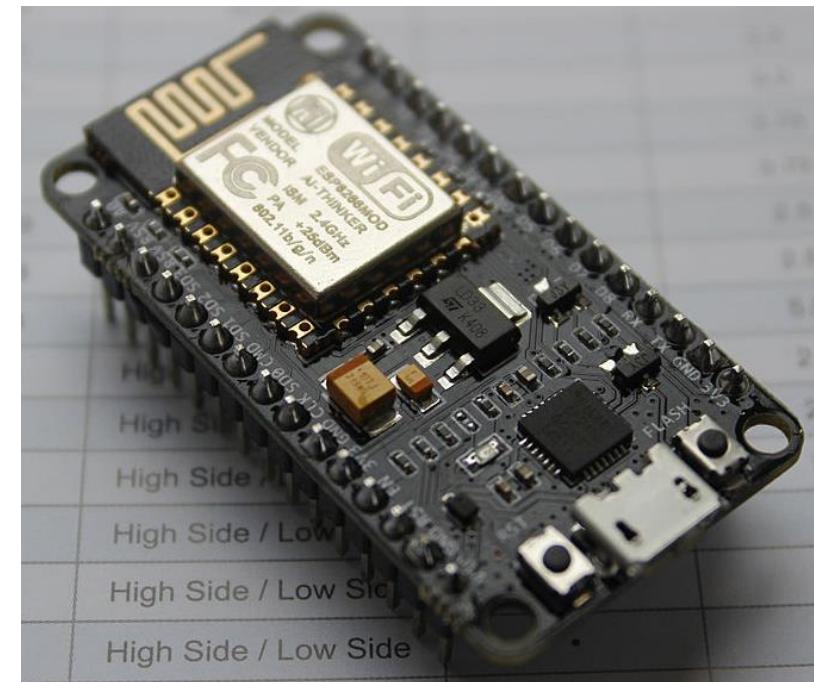
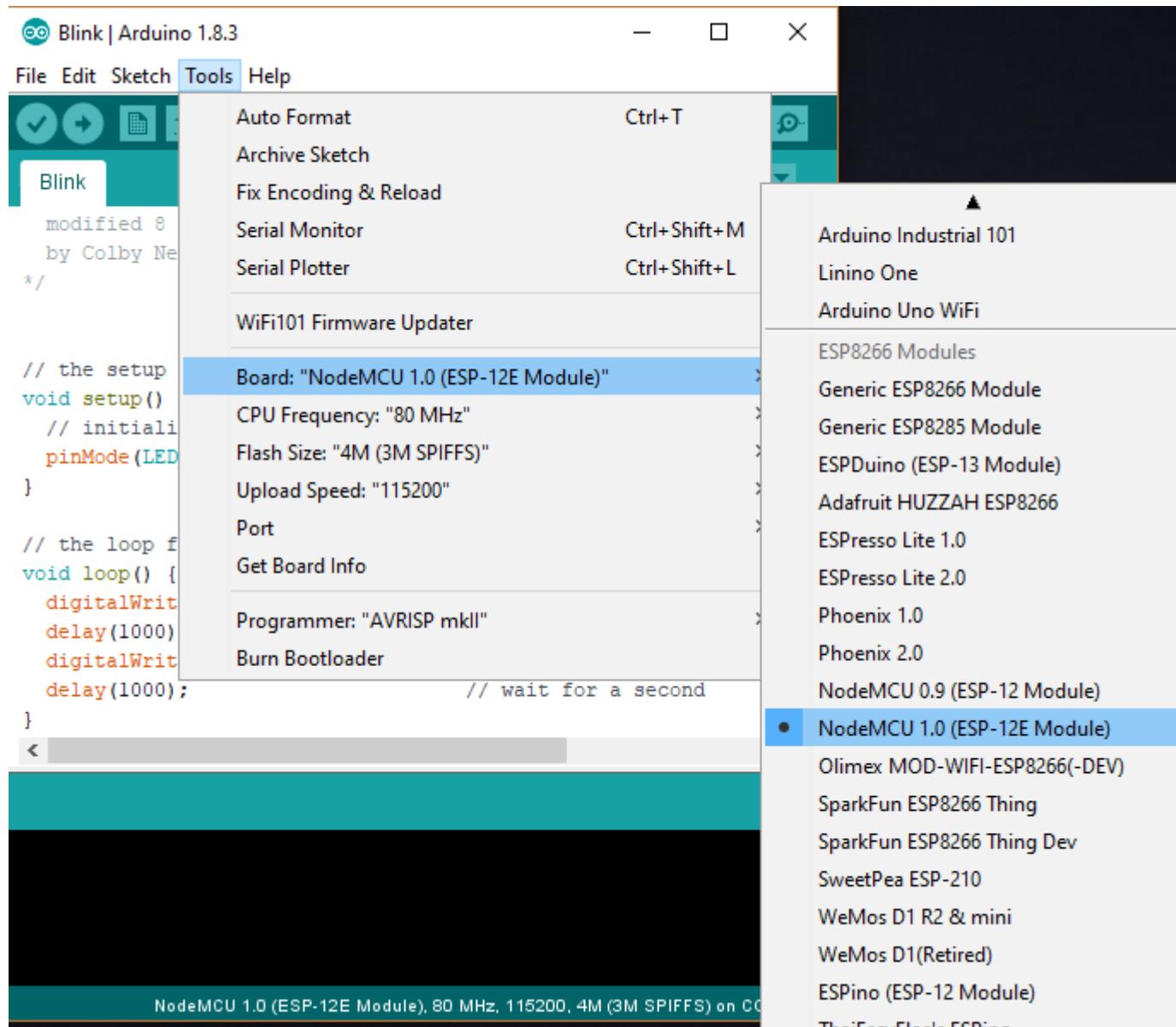
- Title Bar:** esp8266_blink_LED | Arduino 1.8.3
- Menu Bar:** File Edit Sketch Tools Help
- Sketch Area:** The code for the sketch is displayed:

```
#define LED_PIN 2      // GPIO 2 or D4 pin (built-in LED on ESP-12E)
int state = 0;

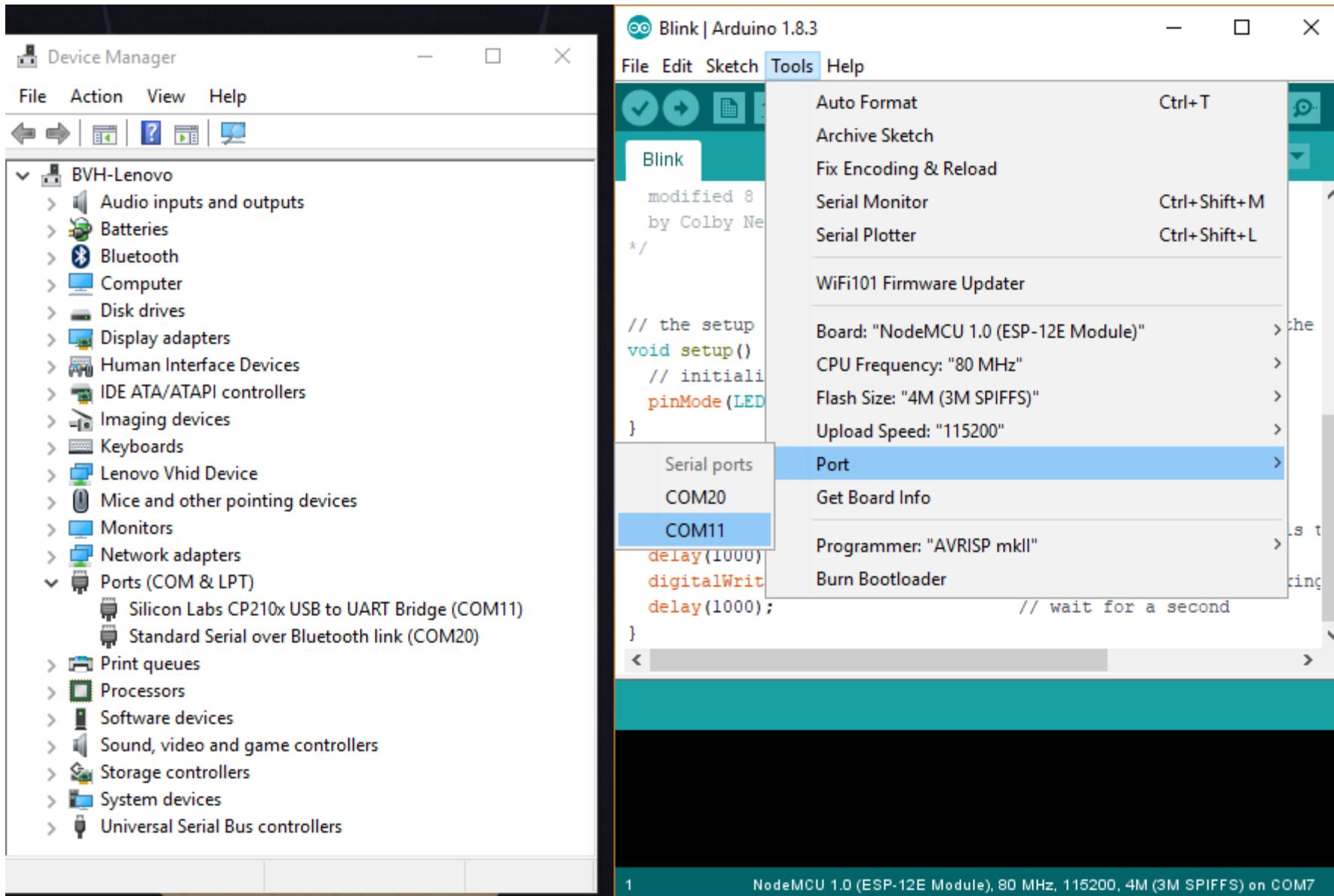
void setup() {
    Serial.begin(115200);    // Open Serial Port
    Serial.println("\n\n\n");
    pinMode(LED_PIN, OUTPUT); // set output mode
}

void loop() {
    Serial.println(state);    // show the current value of the state
    digitalWrite(LED_PIN, state); // update the output using the state variable
    state ^= 1;              // toggle the state variable
    delay(500);              // wait for 500 ms
}
```
- Status Bar:** NodeMCU 1.0 (ESP-12E Module), 80 MHz, 115200, 4M (3M SPIFFS) on COM7

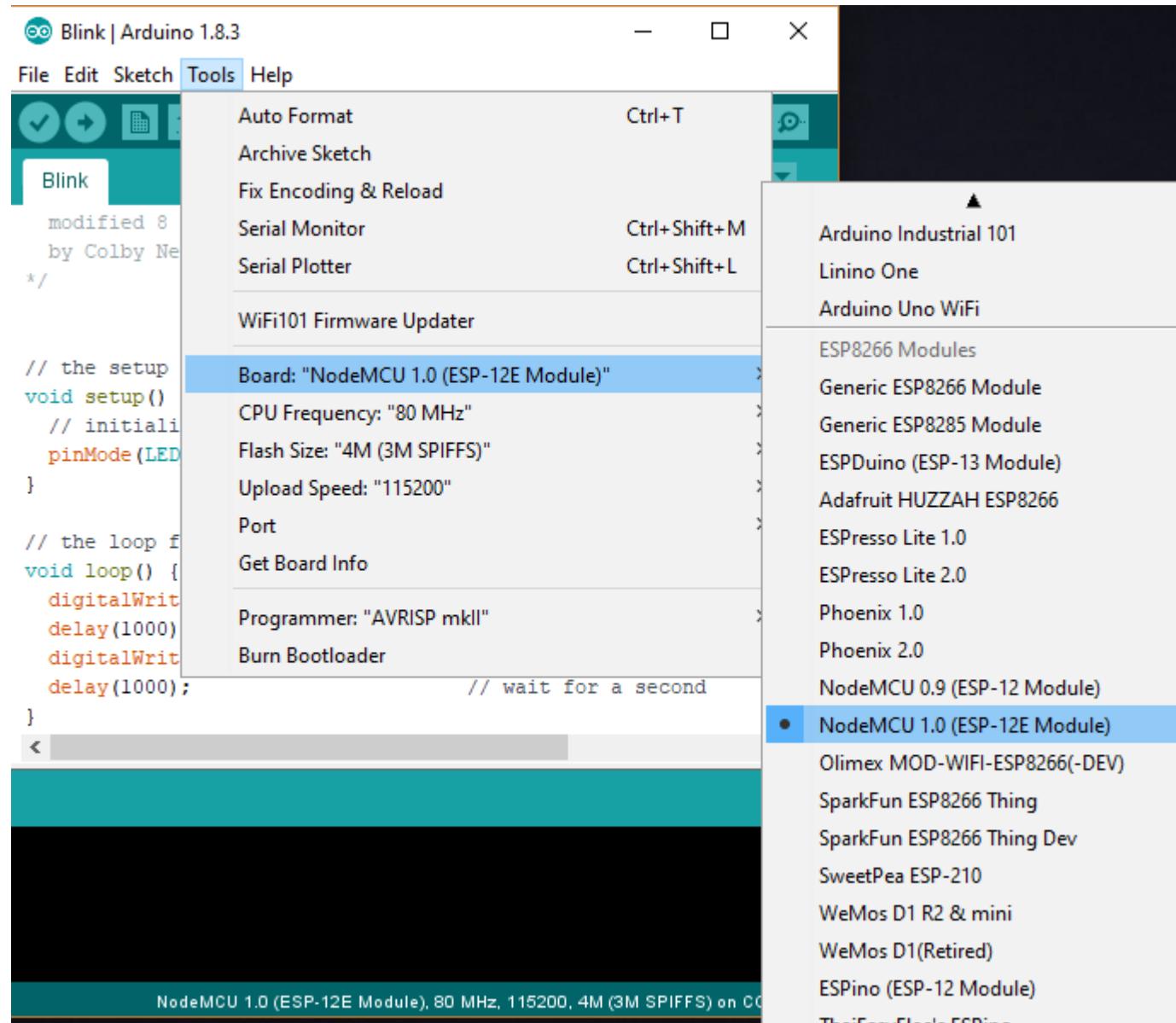
เลือกบอร์ด (NodeMCU)



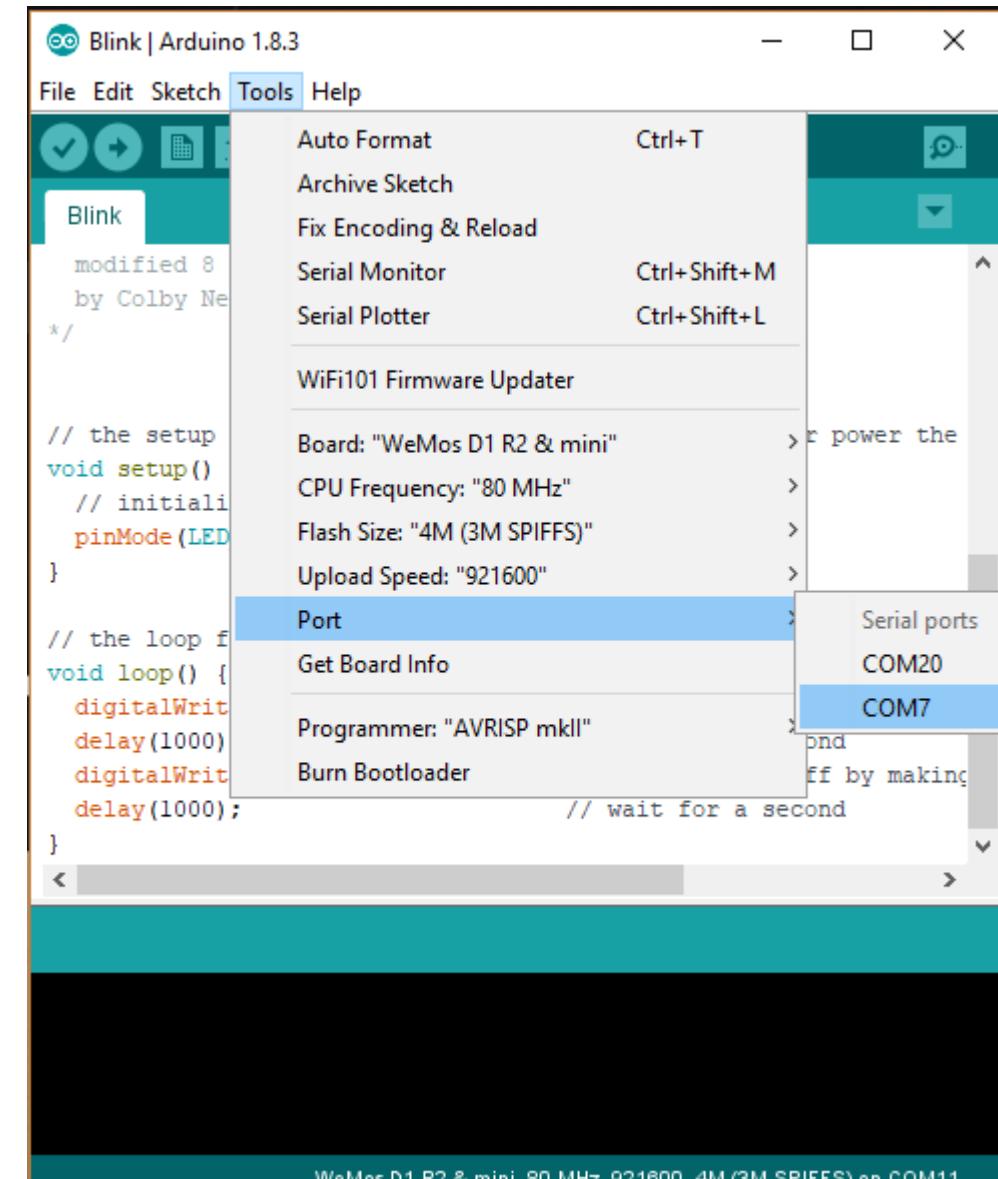
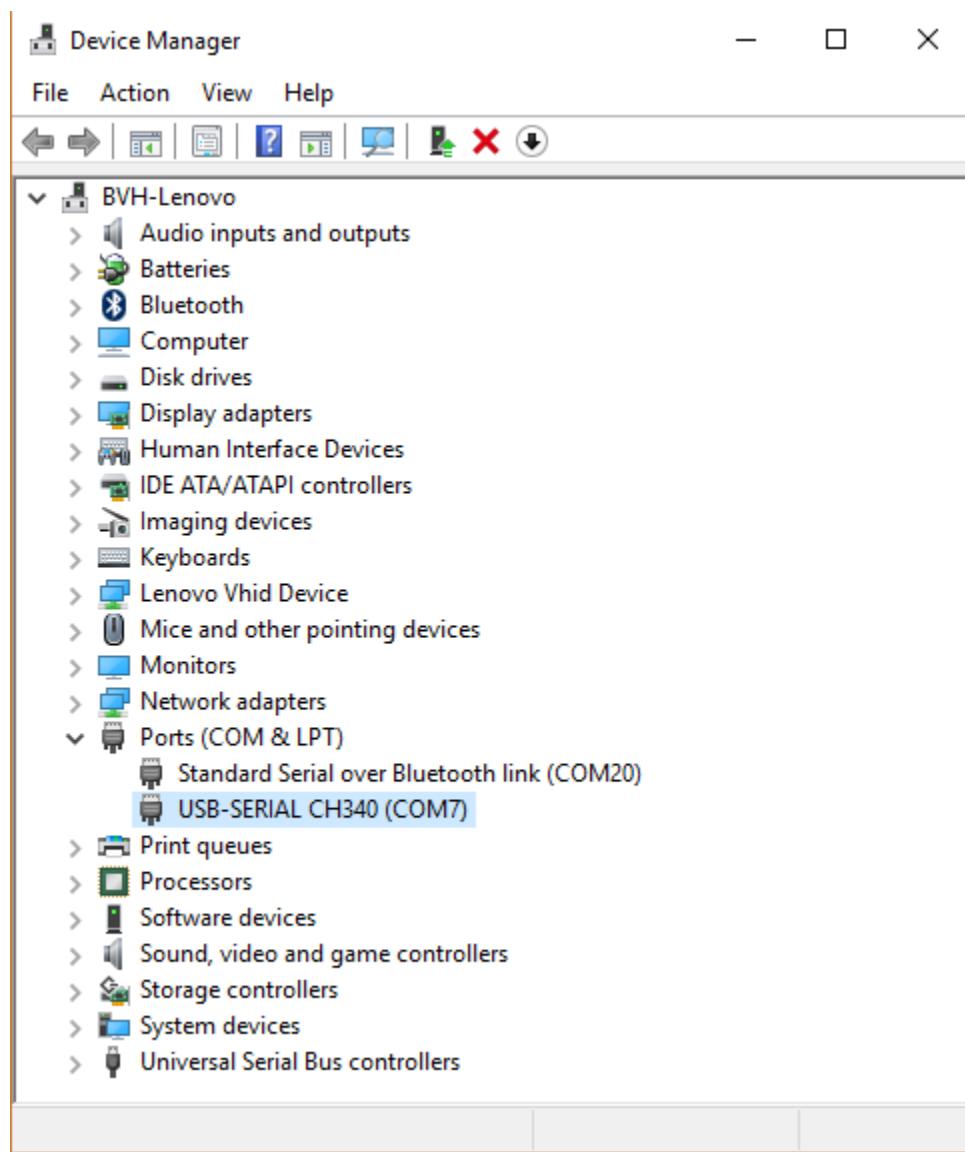
ตรวจสอบ และ เลือก Port (NodeMCU)



เลือกบอร์ด (WeMos D1 Mini)



ตรวจสอบ และ เลือก Port (WeMos D1 Mini)

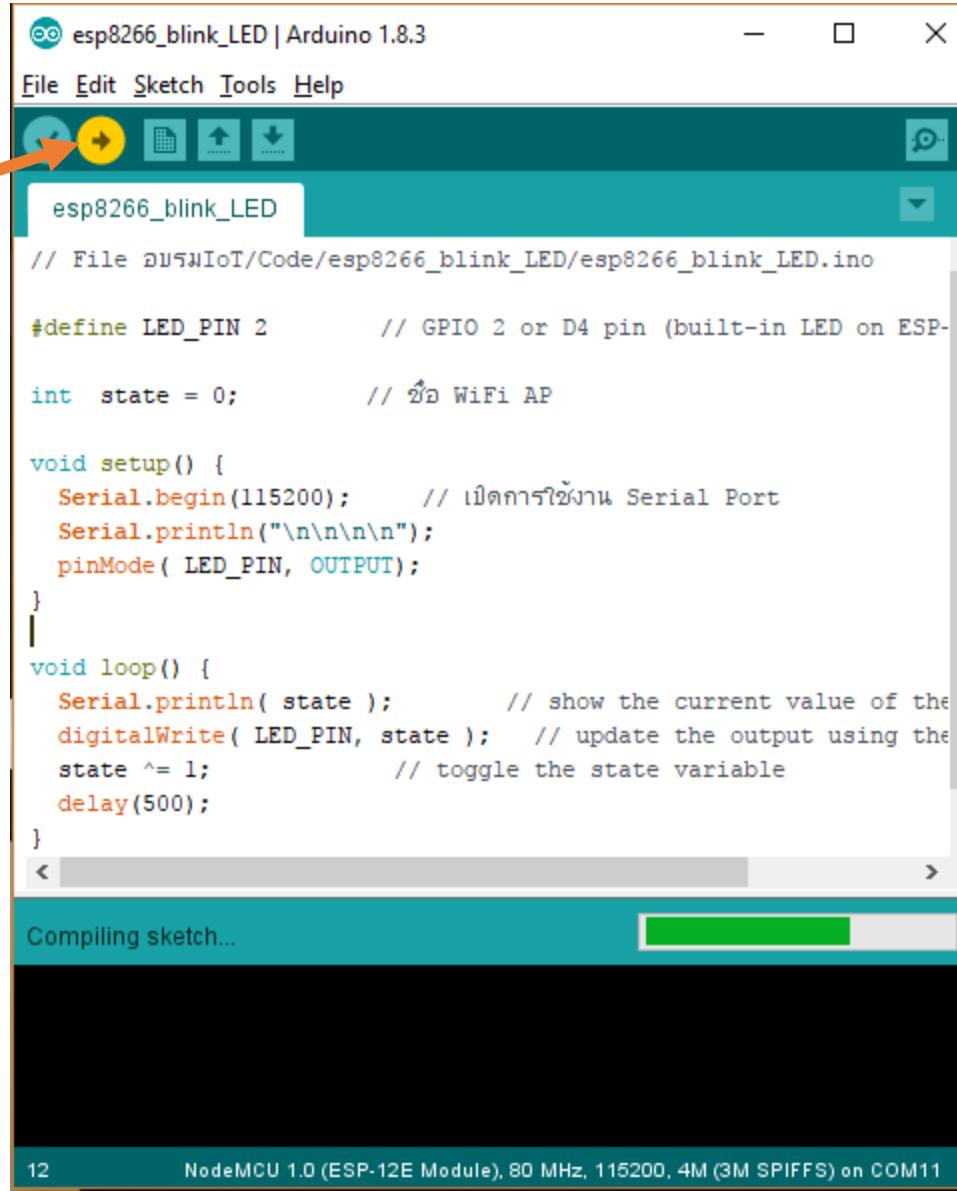


Upload File

กดปุ่ม

หรือใช้ Key ลัด

Ctrl + u



The screenshot shows the Arduino IDE interface with the title bar "esp8266_blink_LED | Arduino 1.8.3". The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar features icons for Open, Save, Upload (highlighted with a yellow arrow), Download, and Magnifying Glass. The code editor displays the "esp8266_blink_LED" sketch. The code itself is as follows:

```
// File อบรมIoT/Code/esp8266_blink_LED/esp8266_blink_LED.ino

#define LED_PIN 2          // GPIO 2 or D4 pin (built-in LED on ESP-12E)

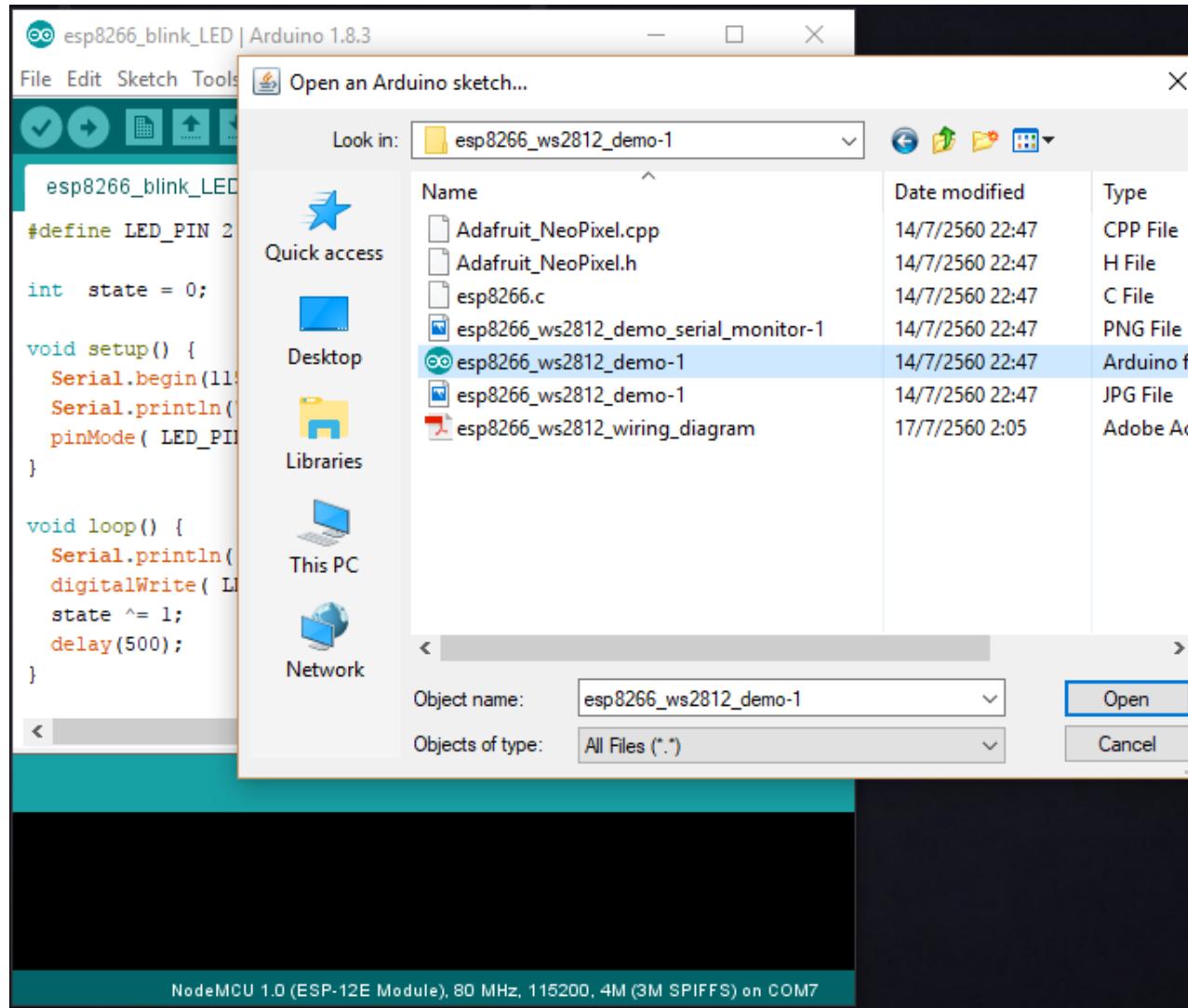
int state = 0;           // ชื่อ WiFi AP

void setup() {
    Serial.begin(115200); // เปิดการใช้งาน Serial Port
    Serial.println("\n\n\n");
    pinMode(LED_PIN, OUTPUT);
}

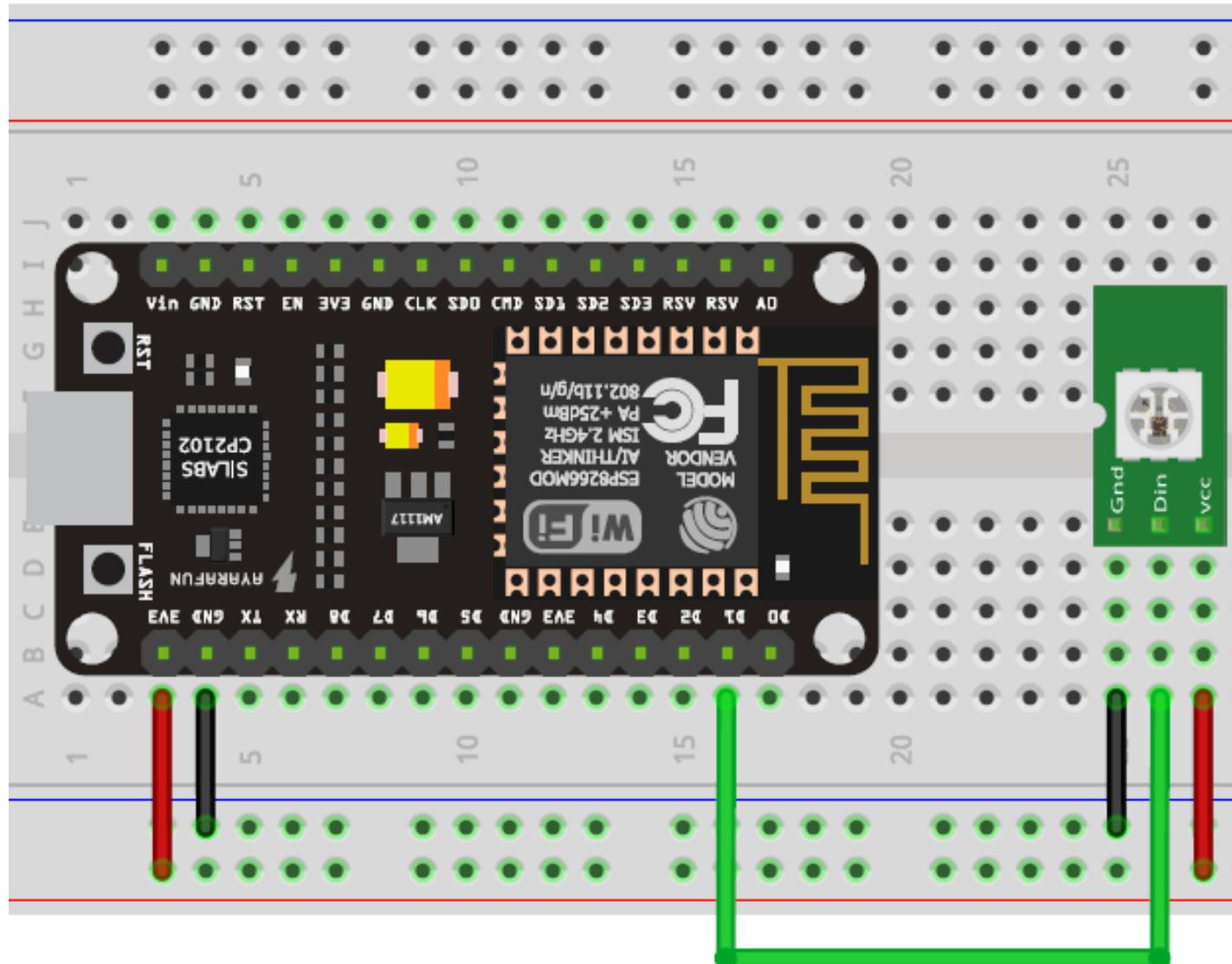
void loop() {
    Serial.println(state); // show the current value of the
    digitalWrite(LED_PIN, state); // update the output using the
    state ^= 1;               // toggle the state variable
    delay(500);
}
```

The status bar at the bottom indicates "Compiling sketch..." with a progress bar.

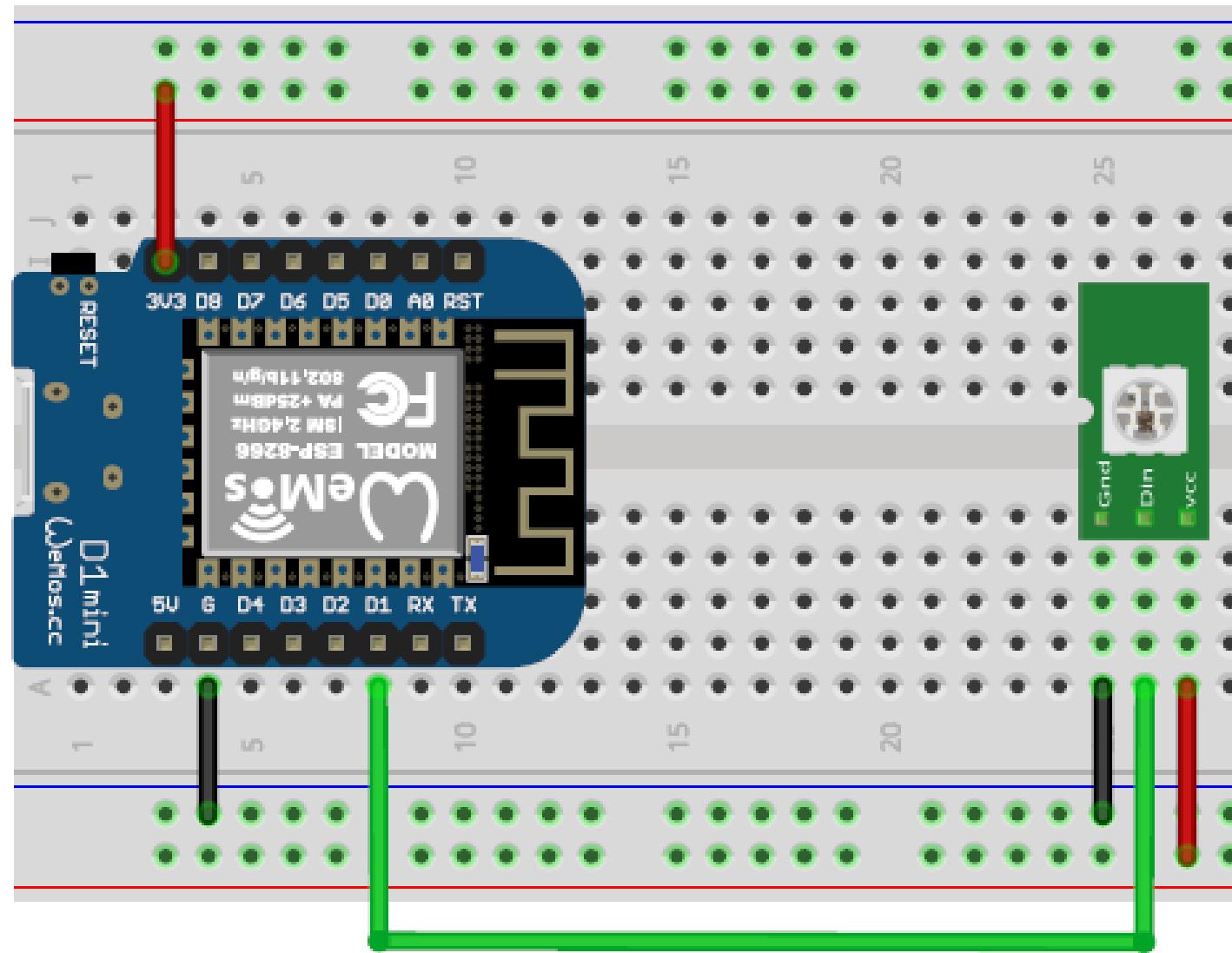
เปิดไฟล์ WS 2812 Demo 1



NodeMCU LED WS2812 Wiring Diagram



Wemos LED WS2812 Wiring Diagram



esp8266_ws2812_demo-1

```
#include "Adafruit_NeoPixel.h" // use the Adafruit NeoPixel library
// see: https://github.com/adafruit/Adafruit_NeoPixel

// User-defined pin definitions for ESP8266 Board
#define DATA_PIN D1 // GPIO-5 / D1 pin
// Note: use Vcc = 5V (or 3.3V, but this works in some cases) for the WS2812

#define NUM_PIXELS 1 // Number of WS2812 RGB LEDs: We use only one WS2812 LED.
#define DELAY_MSEC 2000

// create Adafruit_NeoPixel object and specify the WS2812 type and clock frequency
Adafruit_NeoPixel pixels = Adafruit_NeoPixel( NUM_PIXELS, DATA_PIN, NEO_GRB + NEO_KHZ800 );
```

esp8266_ws2812_demo-1

```
// constants
const uint32_t TEST_COLORS [ ] = {
    pixels.Color(0,0,0), // off
    pixels.Color(255,0,0), // red
    pixels.Color(0,255,0), // green
    pixels.Color(0,0,255), // blue
    pixels.Color(255,255,0),
    pixels.Color(0,255,255),
    pixels.Color(255,0,255),
};

const int NUM_TEST_COLORS = sizeof(TEST_COLORS) / sizeof(uint32);

// global variables
uint32_t ts;      // used to save timestamp
int color_index = 0; // used to select the test color
String str;
char sbuf[32];
```

esp8266_ws2812_demo-1

```
void setup() {
    Serial.begin(115200);
    Serial.println("\n\n\n\n");
    pinMode( DATA_PIN, OUTPUT );

    Serial.println( "ESP8266 WS2812 Demo..." );
    pixels.begin(); // initialize the WS2812 module
    ts = millis();
    pixels.setPixelColor( 0 /*index of WS2812 */, 255 /*R*/, 255 /*G*/, 255 /*B*/ );
    pixels.show();
    delay(100);
    pixels.setPixelColor( 0 /*index of WS2812 */, 0 );
    delay(1000);
}
```

esp8266_ws2812_demo-1

```
void loop() {
    if ( millis() - ts >= DELAY_MSEC ) {
        ts += DELAY_MSEC;
        pixels.setPixelColor( 0 /*index of WS2812 */, TEST_COLORS[color_index] /* color value */ );
        pixels.show();

        str = "Color index: ";
        str += color_index;
        str += ", color value (uint32_t): ";
        sprintf( sbuf, "%06X", pixels.getPixelColor( 0 ) );
        str += sbuf;
        Serial.println( str );

        color_index = (color_index+1) % NUM_TEST_COLORS;
    }
}
```

esp8266_connect_WiFi_AP

```
#include <ESP8266WiFi.h>

#define wifi_ssid "ECC_IoTWorkshop"
#define wifi_password "iotworkshop@ecc"

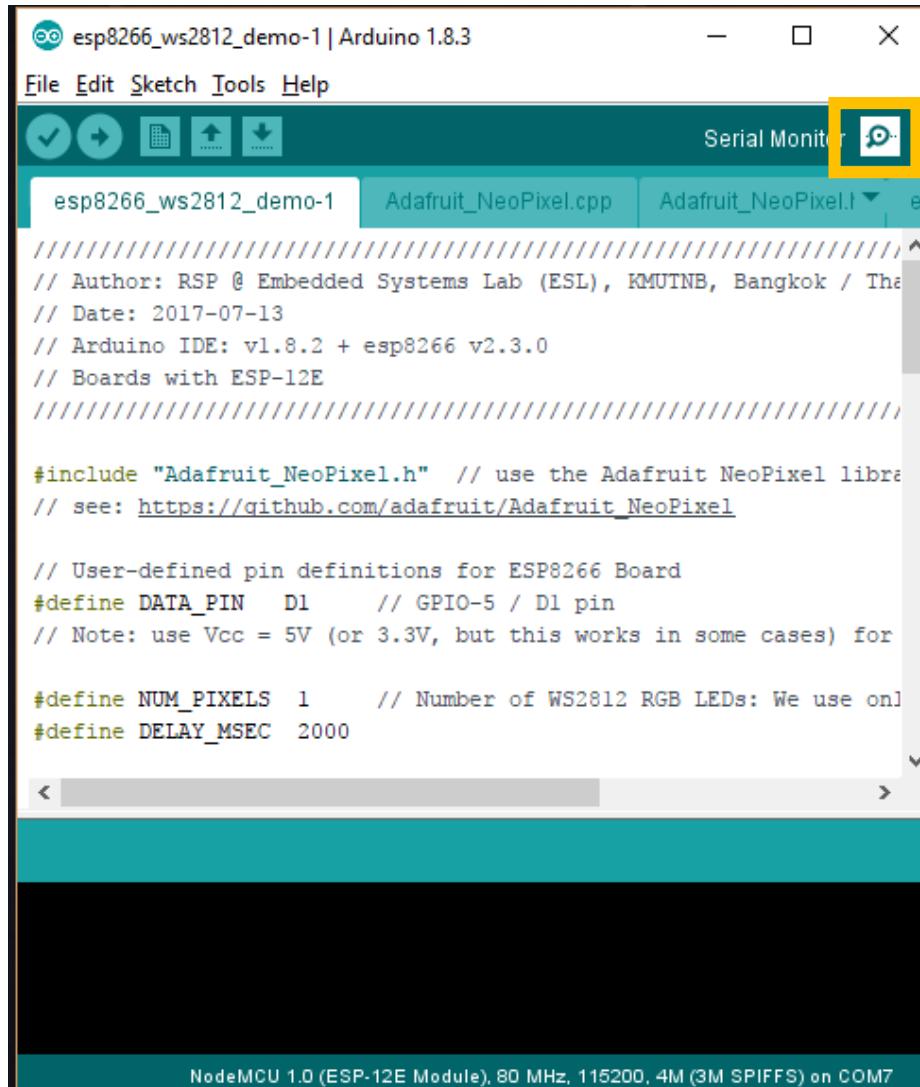
void setup() {
    Serial.begin(115200);          // เปิดการใช้งาน Serial Port
    Serial.println("\n\n\n\n");
    setup_wifi();
}

void loop() {
```

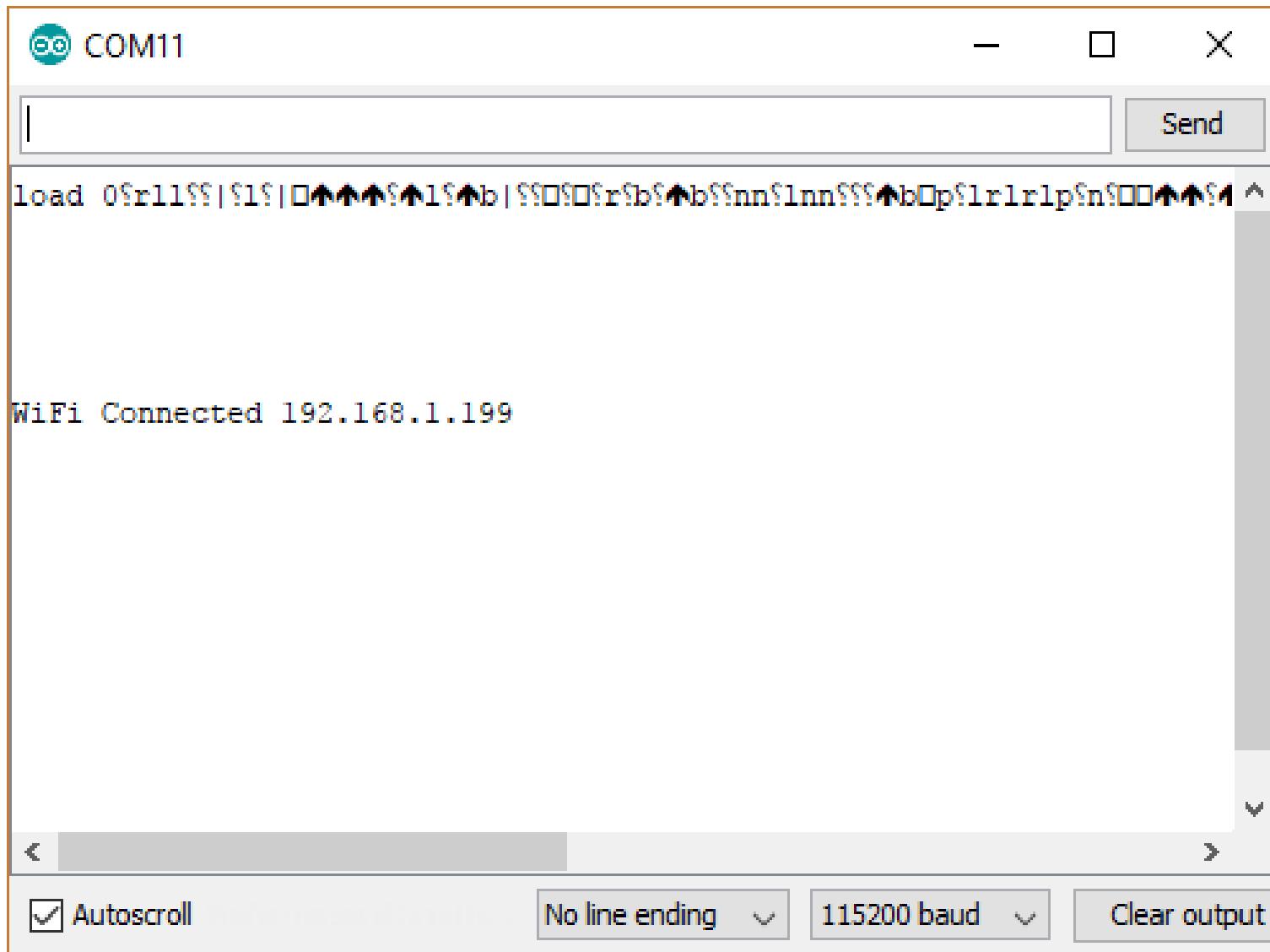
esp8266_connect_WiFi_AP

```
void setup_wifi() {  
    delay(10);  
    // We start by connecting to a WiFi network  
    Serial.println();  
    Serial.print("Connecting to ");  
    Serial.println(wifi_ssid);  
  
    WiFi.begin(wifi_ssid, wifi_password);  
  
    while (WiFi.status() != WL_CONNECTED) {  
        delay(500);  
        Serial.print(".");  
    }  
  
    Serial.println("");  
    Serial.println("WiFi connected");  
    Serial.println("IP address: ");  
    Serial.println(WiFi.localIP());  
}
```

เปิด Serial Monitor



ESP8266 Connect WiFi Result



ຕັ້ງຄ່າ esp8266_ws2812_demo-2

```
#include <ESP8266WiFi.h>
#include "PubSubClient.h"
#include "Adafruit_NeoPixel.h"

#define wifi_ssid "ECC_IoTWorkshop"
#define wifi_password "iotworkshop@ecc"

// Config MQTT Server
#define mqtt_server "m12.cloudmqtt.com"
#define mqtt_port 11419
#define mqtt_user "esp8266_X"
#define mqtt_password "asdf1234"
int esp_id = X;

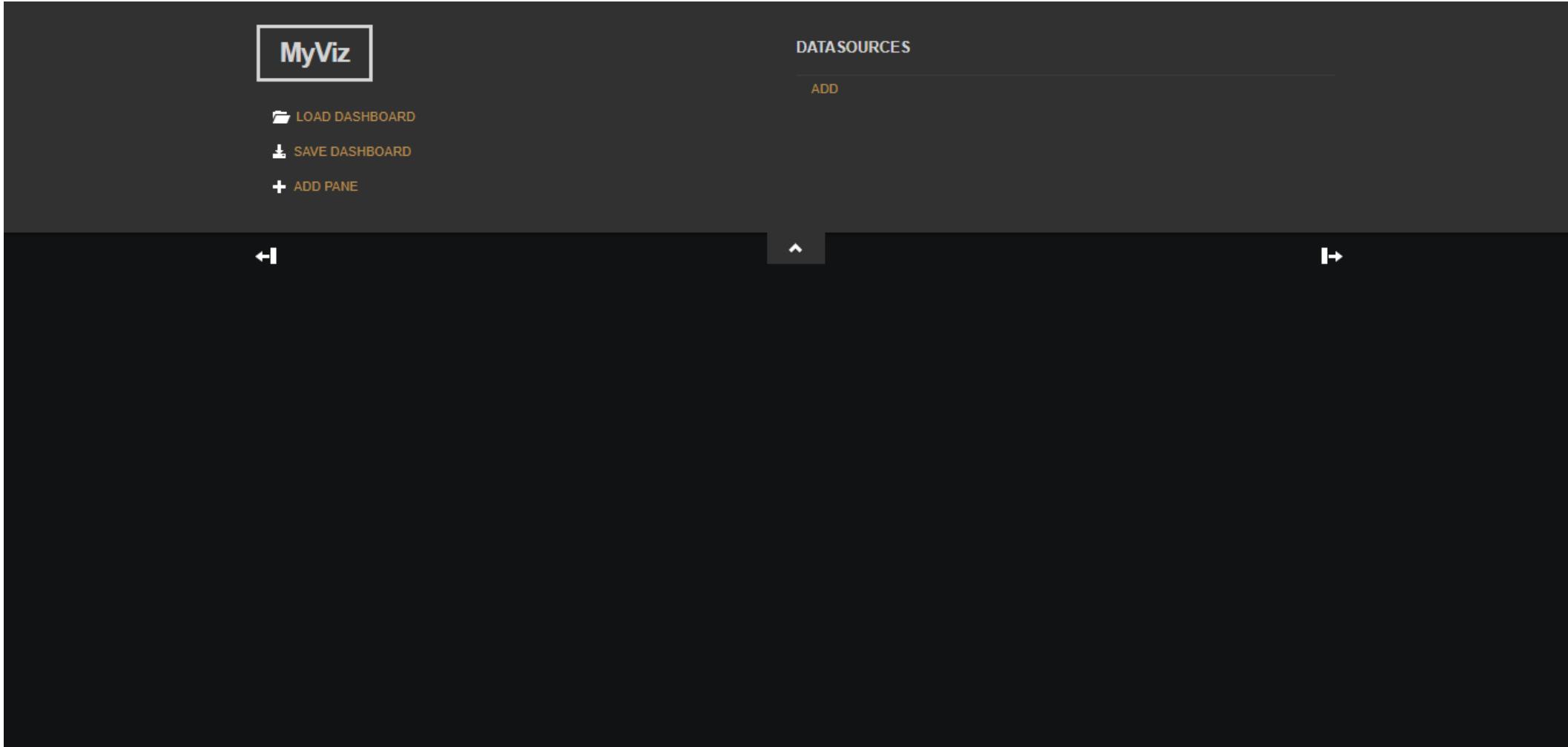
WiFiClient espClient;
PubSubClient client(espClient);
```

ຕັ້ງຄ່າ esp8266_ws2812_demo-2

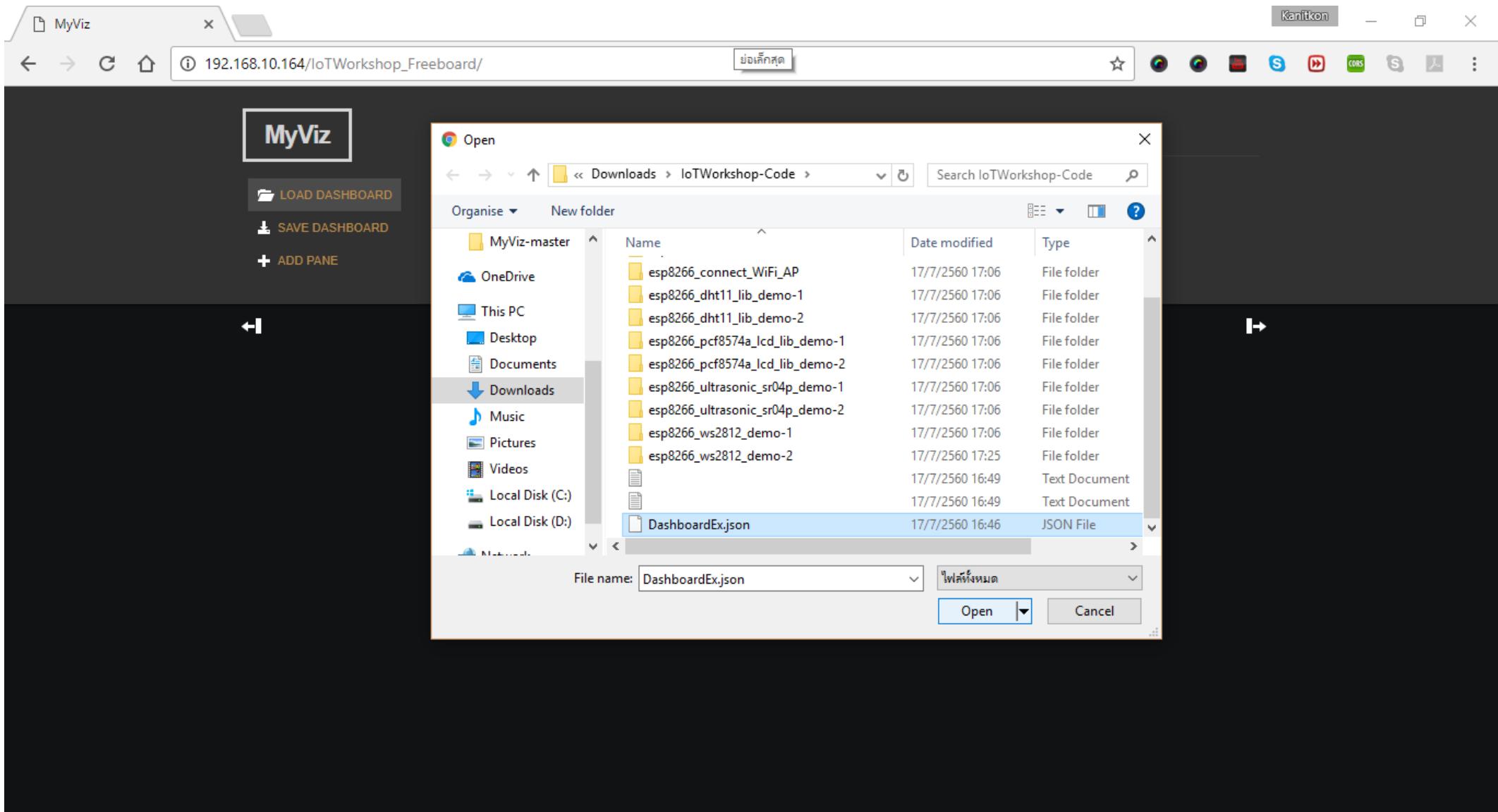
	ເລຂ 1-5	ເລຂ 6-10	ເລຂ 11-15	ເລຂ 16-20
mqtt_server	m12.cloudmqtt.com	m11.cloudmqtt.com	m13.cloudmqtt.com	m12.cloudmqtt.com
mqtt_port	11419	19226	14939	14072

ใช้งาน Freeboard ควบคุม LED

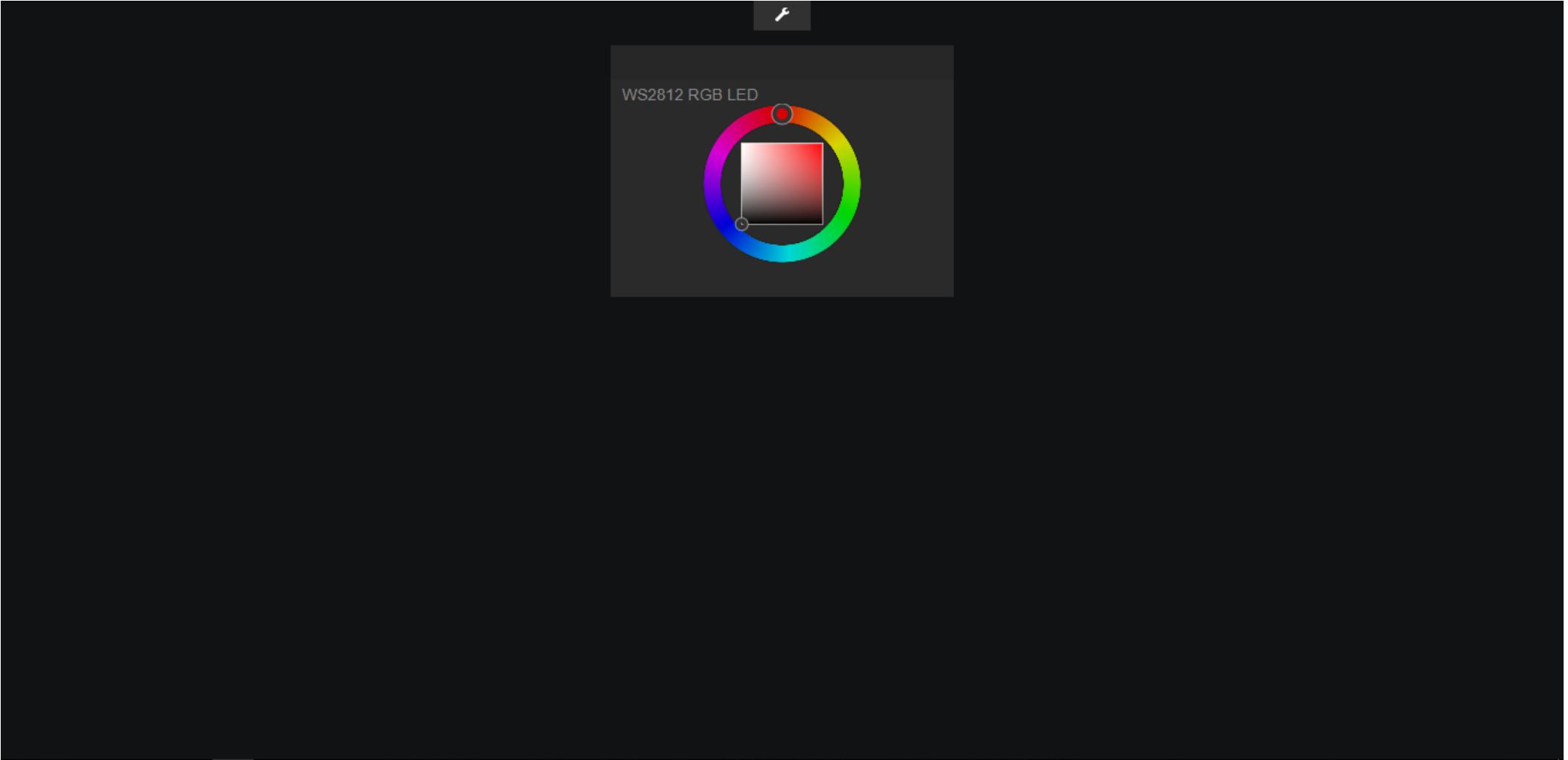
○ เข้าเว็บ browser ที่ url : http://192.168.10.164/IoTWorkshop_Freeboard/



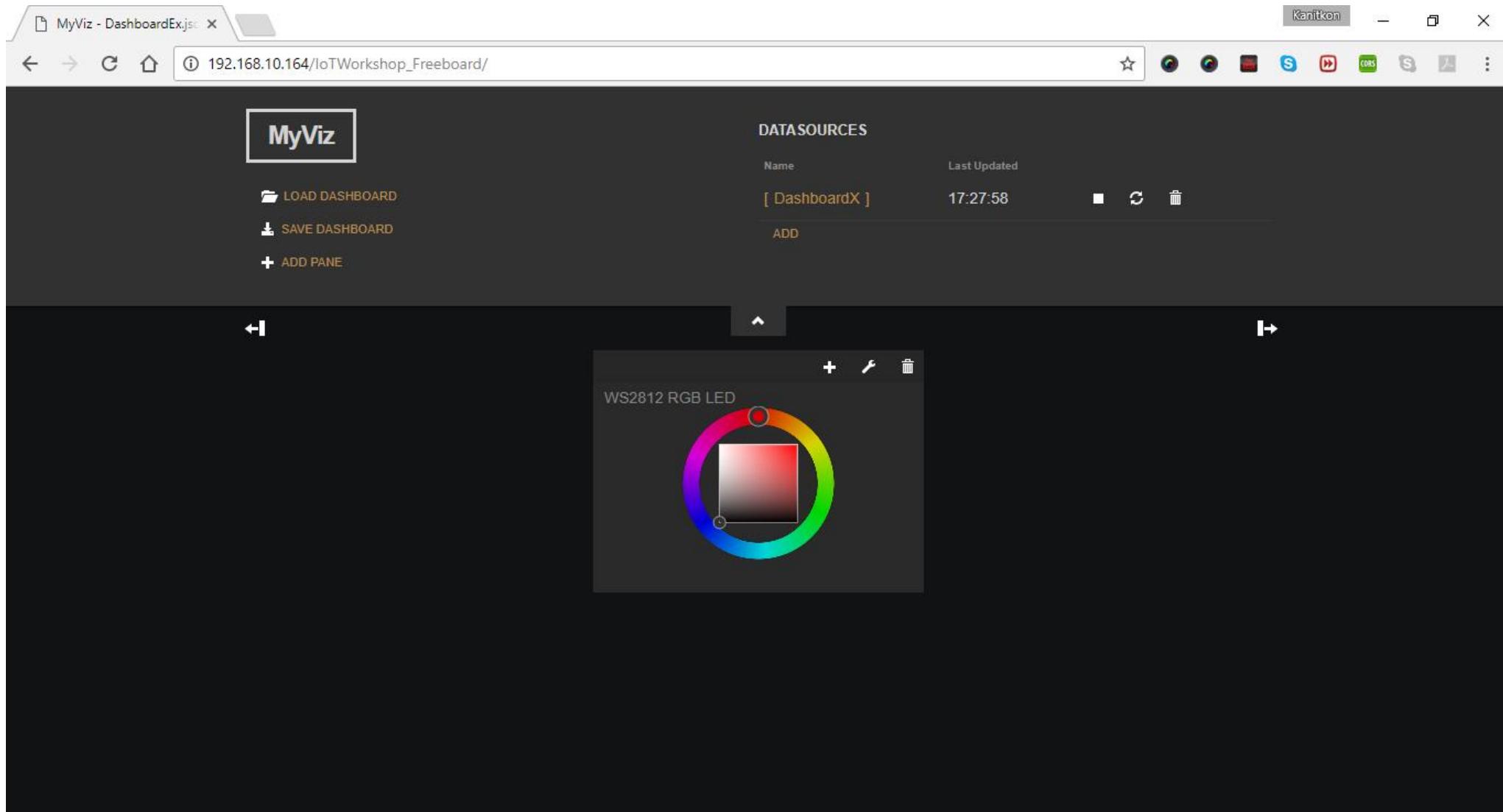
ใช้งาน Freeboard ควบคุม LED



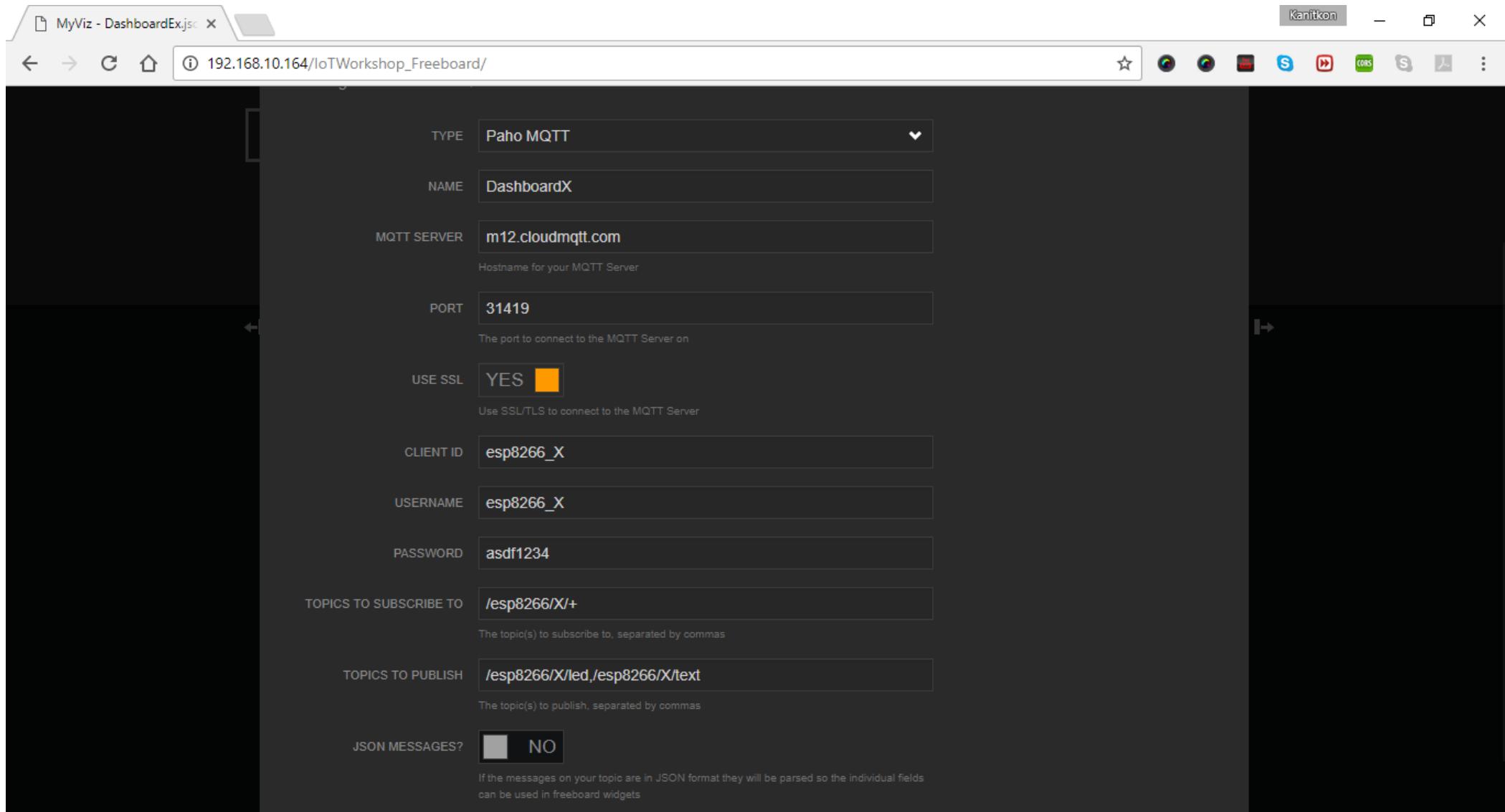
ใช้งาน Freeboard ควบคุม LED



ใช้งาน Freeboard ควบคุม LED



ໃຊ້ງານ Freeboard គາບគຸມ LED

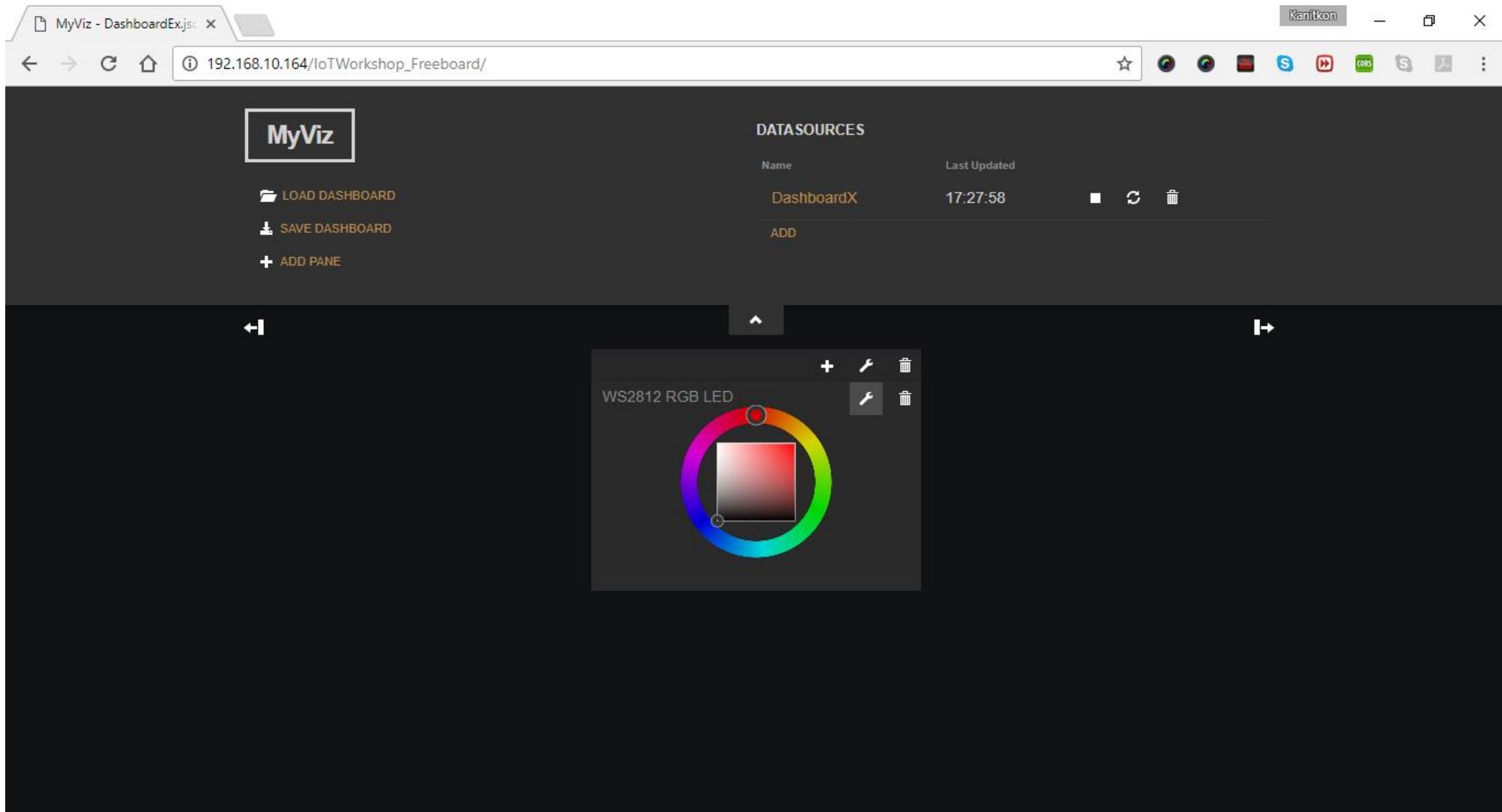


ใช้งาน Freeboard ควบคุม LED

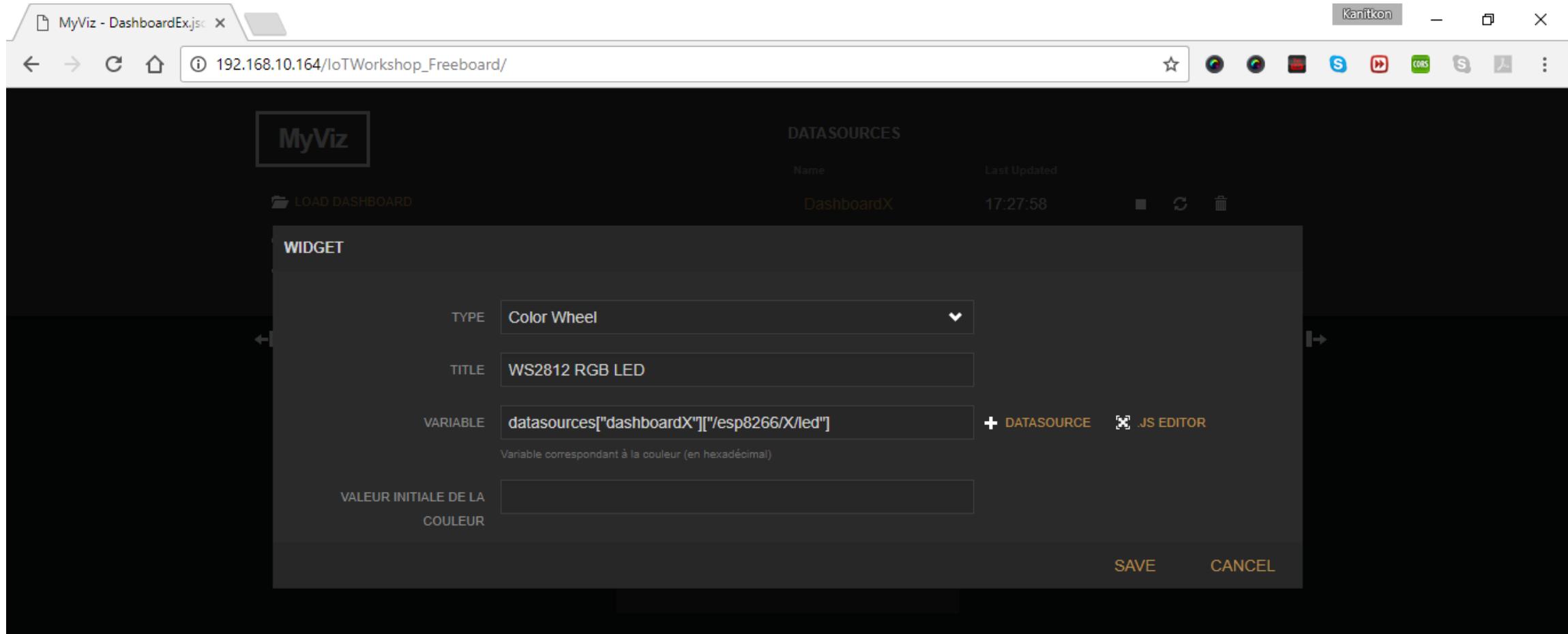
- เปลี่ยนตัว X ให้เป็นเลขตามถุงใส่สายไฟ
 - ตัวอย่าง /esp8266/7 เปลี่ยนตัว X เป็นเลข 7
- เปลี่ยน MQTT Server และ Port ตามตาราง

	เลข 1-5	เลข 6-10	เลข 11-15	เลข 16-20
mqtt_server	m12.cloudmqtt.com	m11.cloudmqtt.com	m13.cloudmqtt.com	m12.cloudmqtt.com
mqtt_port	31419	39226	34939	34072

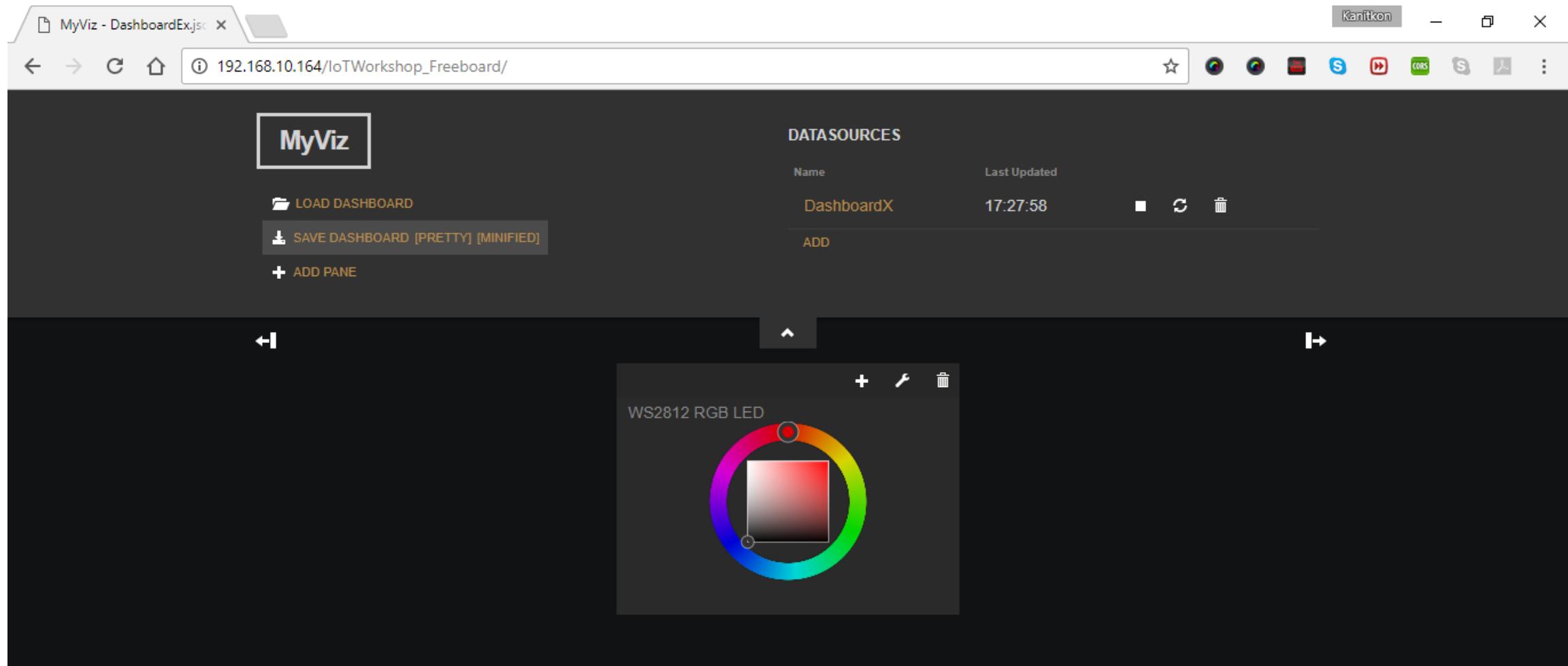
ໃຊ້ງານ Freeboard គາບគຸມ LED



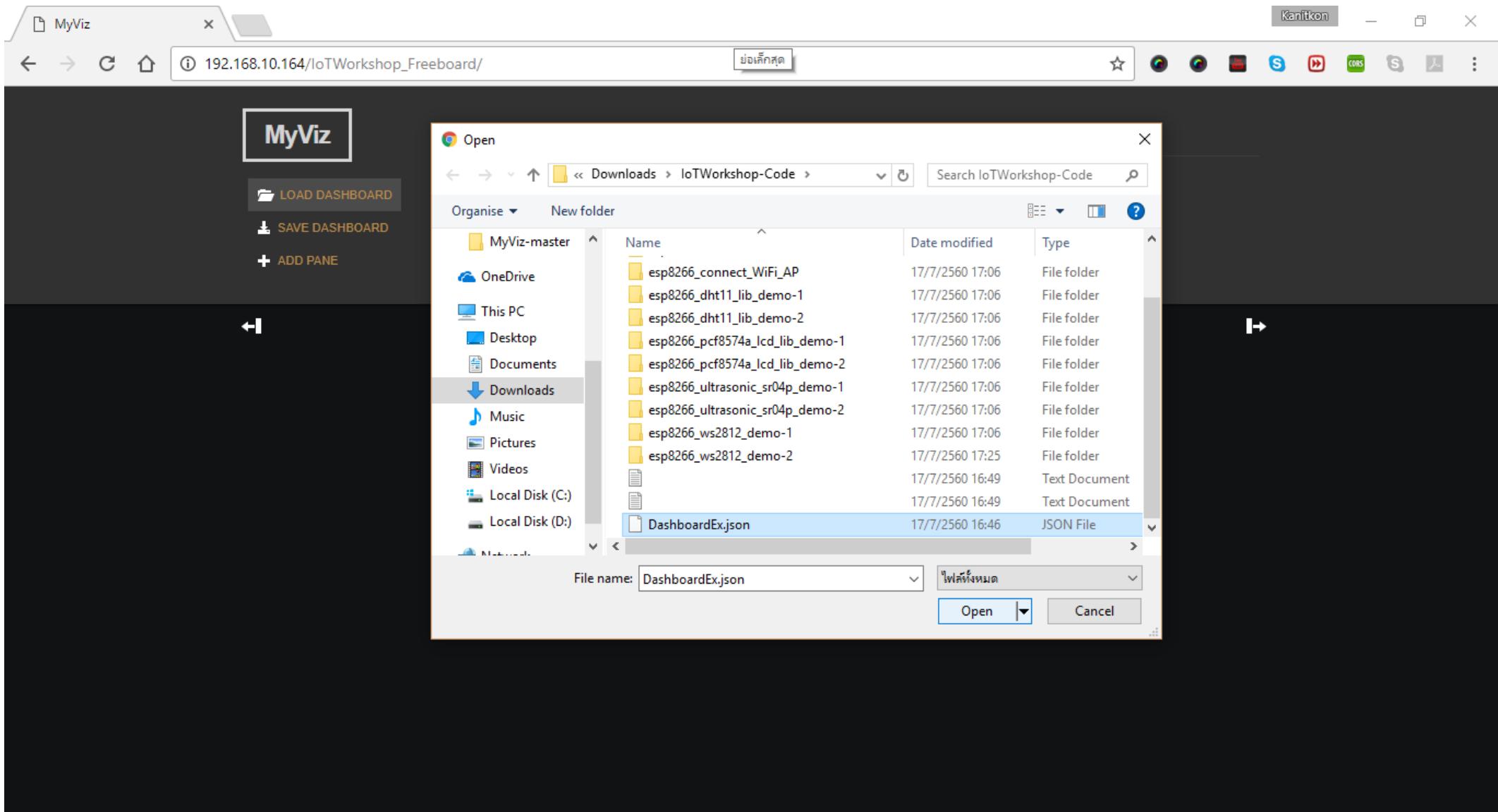
ໃຊ້ງານ Freeboard គາບគຸມ LED



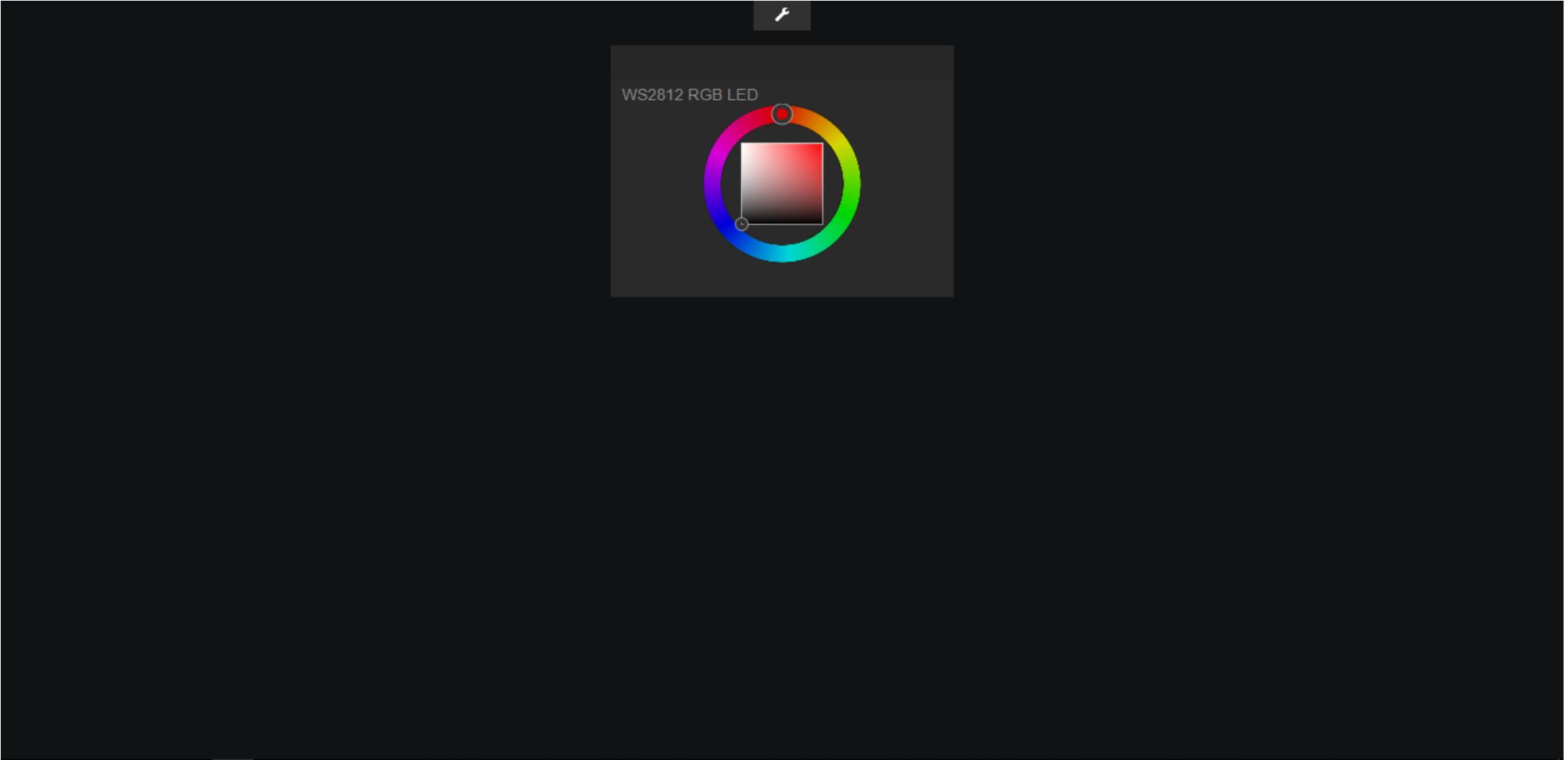
ใช้งาน Freeboard ควบคุม LED



ใช้งาน Freeboard ควบคุม LED



ใช้งาน Freeboard ควบคุม LED



esp8266_ws2812_demo-2

```
// On a Trinket or Gemma we suggest changing this to 1
#define PIN      5

// How many NeoPixels are attached to the Arduino?
#define NUMPIXELS 1
Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);

void setup() {
    pixels.begin(); // This initializes the NeoPixel library.

    Serial.begin(115200);
    delay(10);

    setup_wifi();
    client.setServer(mqtt_server, mqtt_port);
    client.setCallback(callback);

    pixels.setPixelColor(0, pixels.Color(127, 127, 127)); // Moderately bright green color.

    pixels.show(); // This sends the updated pixel color to the hardware.
}
```

esp8266_ws2812_demo-2

```
void reconnect() {
    // Loop until we're reconnected
    while (!client.connected()) {
        Serial.print("Attempting MQTT connection...");
        // Attempt to connect
        // If you do not want to use a username and password, change next line to
        String clientId = "ESP8266Client-"+String(esp_id);
        clientId += String(random(0xffff), HEX);
        if (client.connect(clientId.c_str(), mqtt_user, mqtt_password)) {
            char inTopic[64];
            sprintf(inTopic, "/esp8266/%d/+", esp_id);
            Serial.println("connected");
            client.subscribe(inTopic);
        } else {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" try again in 5 seconds");
            // Wait 5 seconds before retrying
            delay(5000);
        }
    }
}
```

esp8266_ws2812_demo-2

```
void loop() {
    if (!client.connected()) {
        reconnect();
    }
    client.loop();
}

void callback(char* topic, byte* payload, unsigned int length) {
    const char s[2] = "/";
    char *token;
    token = strtok(topic, s);
    token = strtok(NULL, s);
    token = strtok(NULL, s);
    Serial.println(token);
    Serial.write(payload, length);
    Serial.println("");
    if(!strcmp(token, "led")){
        setWS2812(payload, length);
    }
}
```

esp8266_ws2812_demo-2

```
void setWS2812(byte* payload, unsigned int length){  
// Serial.write(payload, length);  
// Serial.println(" ");  
char hexval[2];  
char *ptr;  
  
hexval[0] = payload[1];  
hexval[1] = payload[2];  
int red_val = strtol(hexval, &ptr, 16);  
  
hexval[0] = payload[3];  
hexval[1] = payload[4];  
int green_val = strtol(hexval, &ptr, 16);  
  
hexval[0] = payload[5];  
hexval[1] = payload[6];  
int blue_val = strtol(hexval, &ptr, 16);  
  
pixels.setPixelColor(0, pixels.Color(red_val, green_val, blue_val)); // Moderately bright green color.  
  
pixels.show(); // This sends the updated pixel color to the hardware.  
}
```

บริการ Cloud สำหรับงานด้าน IoT

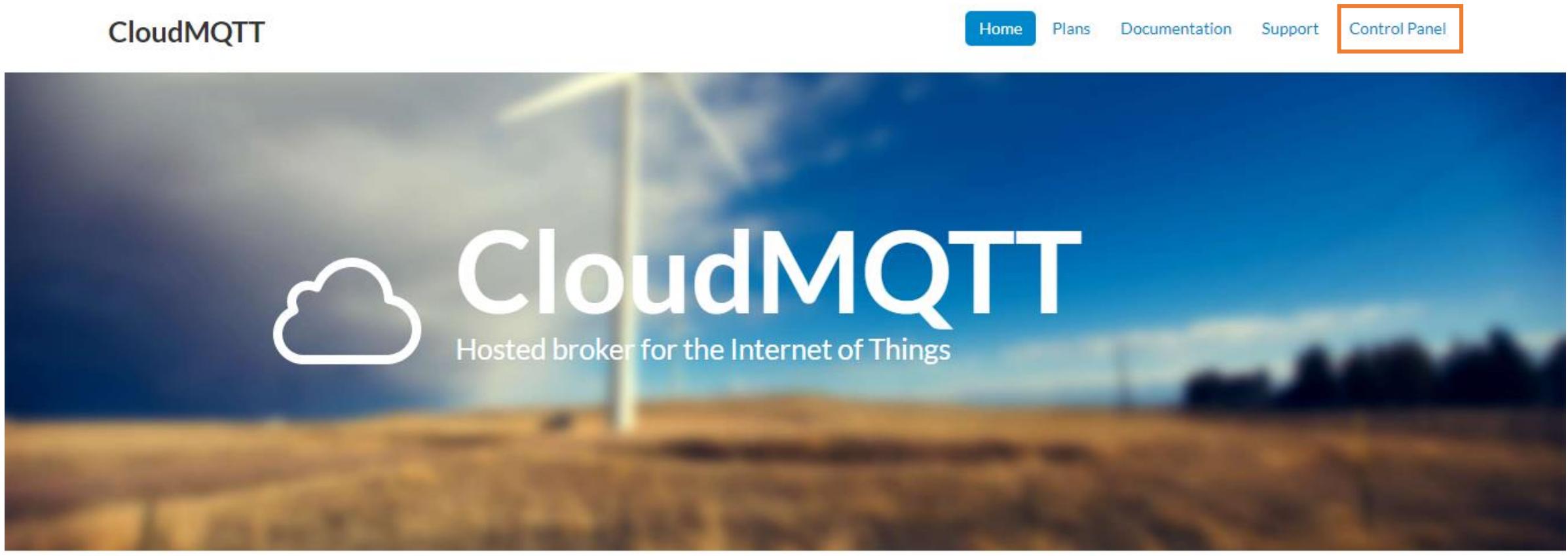
- เป็นบริการในการรับส่งข้อมูลระหว่างอุปกรณ์ต่างๆผ่าน Internet
- นิยมใช้ HTTP และ MQTT เป็นโปรโตคอลในการรับส่งข้อมูล
- ส่วนใหญ่มี Dashboard สำหรับเรียกดูข้อมูล หรือ กำหนดค่าต่างๆ
- บางบริการสามารถติดตั้งลงใน Local Server ได้

ผู้ให้บริการ Cloud สำหรับงานด้าน IoT

- Thingspeak (REST API) *
- Google FireBase (REST API)
- data sparkfun (REST API) *
- HiveMQ (MQTT)
- AWS IoT (MQTT)
- adafruit.io (MQTT) *
- Cloud MQTT (MQTT)
- ─ NETPIE.io (MQTT) #Made in Thailand
- ─ ANTO.io (MQTT) #Made in Thailand

การสมัคร Cloud MQTT

- เปิด Browser เข้าที่ <https://www.cloudmqtt.com/>



การสมัคร Cloud MQTT

○ กรอก email ที่ช่อง Sign up และคลิกปุ่ม Signup

CloudMQTT

Home Plans Documentation Support Control Panel

Login to access your account

E-mail:

Password: Login 

Keep me logged in for two weeks



Sign up If you don't have an account yet

E-mail: Sign up 

Terms of service
Program policies
Privacy policy
Imprint

powered by


การสมัคร Cloud MQTT

○ ตรวจสอบ email จาก Cloud MQTT เมื่อได้รับแล้วให้คลิกที่ลิงค์ดังกล่าว

The screenshot shows an email in the Gmail inbox. The subject line is "[CloudMQTT] Confirm e-mail". The sender is "CloudMQTT <support@cloudmqtt.com>". The timestamp is "17:18 (0 นาทีที่ผ่านมา)". The recipient's name is "ถึง ฉัน". The message content is in Thai and reads: "Please confirm your e-mail address by clicking on this link: https://customer.cloudmqtt.com/customer/confirm?email=a5830050a@gmail.com&token=f81c38816293da6970f5683b7840777e0023ac23". There are language and reading mode buttons at the bottom of the email view.

Please confirm your e-mail address by clicking on this link:

<https://customer.cloudmqtt.com/customer/confirm?email=a5830050a@gmail.com&token=f81c38816293da6970f5683b7840777e0023ac23>

การสมัคร Cloud MQTT

○ กรอกข้อมูลลงไป ให้ครบทุกช่อง ยกเว้น ช่อง VAT ID งานนี้ให้กด submit

CloudMQTT

Home Plans Documentation Support Control Panel

Create an account

E-mail:

Company name

Address
(incl. zip and town)

Country

VAT ID Required only for EU companies

Password: ······

Confirm password: ······



EN

การสมัคร Cloud MQTT

○ เมื่อสมัครเสร็จสิ้นได้ผลดังนี้

CloudMQTT

Home Plans Documentation Support Control Panel

CloudMQTT Instances

for a5830050a@gmail.com ([Log out](#))

Name	Plan	Region
a5830050a@gmail.com kmutnb	Not set up.	

[+ Create](#)

You don't have any instances yet, do you want to [create one?](#)

[Terms of service](#)
[Program policies](#)
[Privacy policy](#)
[Imprint](#)

powered by  amazon web services

สร้าง Broker ของ Cloud MQTT

○ กดปุ่ม +Create

CloudMQTT

Home Plans Documentation Support Control Panel

CloudMQTT Instances for a5830050a@gmail.com (Log out)

Name	Plan	Region	Actions
freeboard02	Cat	US	Details Edit Delete
freeboard01	Cat	US	Details Edit Delete

[+ Create](#)

Account ([Edit](#)) PayPal ([Change](#)) Invoices ([All](#))

a5830050a@gmail.com Not set up.

Terms of service Program policies Privacy policy Imprint

powered by  amazon web services

สร้าง Broker ของ Cloud MQTT

○ กำหนดชื่อ Data center และ Plan

CloudMQTT

Home Plans Documentation Support Control Panel

Create new CloudMQTT Instance

PayPal not enabled. Please [enable PayPal](#) if you want to subscribe to a paid plan

Name

Data center

Plan

To learn more about the different plans please visit: www.cloudmqtt.com/plans.html

สร้าง Broker ของ Cloud MQTT

○ ในส่วนของ Plan นั้นเราจะเลือกตามคุณภาพดังนี้

CloudMQTT Plans

Home **Plans** Documentation Support Control Panel



Cute Cat

- 10 connections
- 10 Kbit/s

[Try now for Free](#)



Keen Koala

- 100 connections
- 100 Kbit/s
- Support by e-mail
- Support by chat

[Try now for \\$19/month](#)



Loud Leopard

- 1 000 connections
- 1 Mbit/s
- Support by e-mail
- Support by chat

[Try now for \\$99/month](#)



Power Pug

- 10 000 connections
- 10 Mbit/s
- Support by e-mail
- Support by chat
- 24/7 phone support

[Try now for \\$299/month](#)

สร้าง Broker ของ Cloud MQTT

อคลิ๊กไปที่ detail เพื่อเข้าไปดูข้อมูลข้างใน และสร้าง user

CloudMQTT

Home Plans Documentation Support Control Panel

CloudMQTT Instances for a5830050a@gmail.com (Log out)

Name	Plan	Region	Actions
Test	Cat	US	Details Edit Delete
freeboard02	Cat	US	Details Edit Delete
freeboard01	Cat	US	Details Edit Delete

[+ Create](#)

Account (Edit) PayPal (Change) Invoices (All)

a5830050a@gmail.com
kmutnb

Not set up.

Terms of service Program policies Privacy policy Imprint

powered by  amazon webservices

สร้าง Broker ของ Cloud MQTT

หน้านี้จะมีข้อมูลของ broker อยู่ และส่วนถัดมาจะให้เราสร้าง user โดยกรอกชื่อและรหัสลงไป

CloudMQTT Console

Overview Websocket UI Server log Statistics Restart

Instance info

Server	
User	
Password	
Port	
SSL Port	24939
Websockets Port (TLS only)	34939
Connection limit	10

Users and ACL Bridges Amazon Kinesis Stream

Manage Users

user	Save
------	-------	------

สร้าง Broker ของ Cloud MQTT

- เมื่อสร้างเสร็จจะต้องกำหนด Topic สำหรับ user

The screenshot shows the 'Manage Users' page from the AWS IoT console. At the top, there are three tabs: 'Users and ACL' (selected), 'Bridges', and 'Amazon Kinesis Stream'. Below the tabs, the title 'Manage Users' is displayed. A single user entry is listed:

- username: user
- password: [REDACTED]
- Actions: Delete (red button)

At the bottom of the list, there are two input fields: 'username' and 'password', followed by a green 'Save' button.

สร้าง Broker ของ Cloud MQTT

○สร้าง topic ขึ้นมาเพื่อที่เราจะได้เข้า subscribe และ publish ได้

User	Topic	Read	Write
------	-------	------	-------

New Rule

User: user

Topic: /user/+

Read Access?

Write Access?

Save

สร้าง Broker ของ Cloud MQTT

User	Topic	Read	Write	
user	/user/+	true	true	<button>Delete</button>

New Rule

User:

Topic:

Read Access?

Write Access?

Save

การสมัคร Thinkspeak

- เข้าไปยังหน้าเว็บ <https://thingspeak.com> และดูเลือกคำว่า Sign Up เพื่อสมัครสมาชิก



Collect

Analyze

Act

การสมัคร Thinkspeak

- กรอกข้อมูลที่ใช้ในการสมัครสมาชิก

ThingSpeak™ Channels Apps Community Support ▾ How to Buy Log In Sign Up

Sign up for ThingSpeak

The ThingSpeak service is operated by MathWorks. In order to sign up for ThingSpeak, you must create a new MathWorks Account or log in to your MathWorks Account.

Create MathWorks Account

Email Address !
Missing required information

User ID ?
Password ?
Thailand ▼
First Name
Last Name

By clicking continue, you agree to our [privacy policy](#)

DATA AGGREGATION AND ANALYTICS
ThingSpeak™

SMART CONNECTED DEVICES

MATLAB®

ALGORITHM DEVELOPMENT
SENSOR ANALYTICS

การสมัคร Thinkspeak

- เมื่อกรอกข้อมูลครบกดปุ่ม Continue

ThingSpeak™ Channels Apps Community Support ▾ How to Buy Log In Sign Up

Create MathWorks Account

a5830033@gmail.com ✓
a5830033 ✓ ?
..... ✓ ⓘ
Thailand
jessada ✓
weeradetkumpon ✓

By clicking continue, you agree to our privacy policy

Cancel Continue

DATA AGGREGATION AND ANALYTICS

ThingSpeak™

SMART CONNECTED DEVICES

MATLAB®

ALGORITHM DEVELOPMENT SENSOR ANALYTICS

Already have a ThingSpeak account?

การสมัคร Thinkspeak

○ ตรวจสอบ email และกดปุ่ม Continue

ThingSpeak™ Channels Apps Community Support ▾ How to Buy Log In Sign Up

Verify Your MathWorks Account

To finish creating your account, complete the following steps:

1. Go to your inbox for a5830033@gmail.com.
2. Click the link in the email we sent you.

Once you've done this, click Continue.

Didn't get the email?

1. Check your spam folder.
2. [Send me the email again](#).
3. Contact [Customer Support](#) if you still do not have the email

The diagram illustrates the data flow and integration between ThingSpeak, MATLAB, and smart connected devices. At the center is a cloud icon labeled "DATA AGGREGATION AND ANALYTICS" containing the "ThingSpeak™" logo. To the left, several small icons representing "SMART CONNECTED DEVICES" are shown, each with a Wi-Fi signal and a connection line pointing to a central hub. From this hub, a blue arrow points down to the ThingSpeak cloud. Another blue arrow points from the ThingSpeak cloud to the right, where a monitor icon represents "MATLAB". Below the monitor, the text "ALGORITHM DEVELOPMENT SENSOR ANALYTICS" is displayed. A double-headed blue arrow connects the central hub to the MATLAB monitor.

Cancel Continue

Already have a ThingSpeak account?

การสมัคร Thinkspeak

○ เมื่อทำการยืนยันแล้วจะสามารถเริ่มใช้งานได้



Sign-up successful

Congratulations, you have successfully linked your MathWorks account to ThingSpeak. Use the following email ID and its associated MathWorks account password on all subsequent logins to ThingSpeak.

Email ID: a5830033@gmail.com

Welcome to ThingSpeak!

OK

การสร้าง Channel Thinkspeak

○ ก่อนเริ่มใช้งานกดปุ่ม Agree to Terms



ThingSpeak Terms of Use

ThingSpeak Terms of Use have changed. We require that you agree to the [Terms of Use](#) and [Privacy Policy](#) before continuing.

[Agree to Terms](#)

[Decline and Sign Out](#)

การสร้าง Channel Thinkspeak

○ กดปุ่ม New Channel ในการสร้าง channel สำหรับใช้งาน



My Channels

New Channel

Help

Collect data in a ThingSpeak channel from a device, from another channel, or from the web.

Click [New Channel](#) to create a new ThingSpeak channel.

Click on the column headers of the table to sort by the entries in that column.

Learn to [create channels](#), explore and transform data.

Learn more about [ThingSpeak Channels](#).

Examples

- [Arduino](#)
- [Arduino MKR1000](#)
- [ESP8266](#)
- [Raspberry Pi](#)
- [Netduino Plus](#)

Upgrade

Need to send more data faster?

Need to use ThingSpeak for a commercial project?

Upgrade

การสร้าง Channel Thinkspeak

○ กรอกข้อมูลของ channel ที่ต้องการใช้งาน

The screenshot shows the 'New Channel' creation page on the ThingSpeak website. The top navigation bar includes links for 'Channels', 'Apps', 'Community', 'Support', 'How to Buy', 'Account', and 'Sign Out'. The main form on the left is titled 'New Channel' and contains fields for 'Name' (set to 'Test'), 'Description' (empty), and eight 'Field' sections (Field 1 to Field 8), each with a 'Field Label' input and a checked checkbox. Below the fields is a 'Metadata' input field. On the right, a 'Help' section provides information about channels and their settings, listing various configuration options like Channel Name, Description, Fields, Metadata, Tags, Latitude, Longitude, Elevation, and Publicity.

New Channel

Name: Test

Description:

Field 1: Field Label 1

Field 2:

Field 3:

Field 4:

Field 5:

Field 6:

Field 7:

Field 8:

Metadata:

Help

Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.

Channel Settings

- **Channel Name:** Enter a unique name for the ThingSpeak channel.
- **Description:** Enter a description of the ThingSpeak channel.
- **Field#:** Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
- **Metadata:** Enter information about channel data, including JSON, XML, or CSV data.
- **Tags:** Enter keywords that identify the channel. Separate tags with commas.
- **Latitude:** Specify the position of the sensor or thing that collects data in decimal degrees. For example, the latitude of the city of London is 51.5072.
- **Longitude:** Specify the position of the sensor or thing that collects data in decimal degrees. For example, the longitude of the city of London is -0.1275.
- **Elevation:** Specify the position of the sensor or thing that collects data in meters. For example, the elevation of the city of London is 35.052.
- **Make Public:** If you want to make the channel publicly available, check this box.
- **URL:** If you have a website that contains information about your ThingSpeak channel, specify the URL.

การสร้าง Channel Thinkspeak

- เมื่อกรอกข้อมูลเสร็จลิ้นทำการกดปุ่ม Save Channel

The screenshot shows the configuration interface for a new ThingSpeak channel. The top navigation bar includes links for Channels, Apps, Community, Support, How to Buy, Account, and Sign Out. The main form contains the following fields:

- Make Public**: A checkbox with an unchecked state.
- URL**: An input field containing a URL.
- Elevation**: An input field containing an elevation value.
- Show Location**: A checkbox with an unchecked state.
- Latitude**: An input field containing a latitude value (0.0).
- Longitude**: An input field containing a longitude value (0.0).
- Show Video**: A checkbox with an unchecked state.
- YouTube**: A radio button selected.
- Vimeo**: A radio button unselected.
- Video ID**: An input field for entering a video ID.
- Show Status**: A checkbox with an unchecked state.
- Save Channel**: A large green rectangular button at the bottom of the form.

A note above the "Show Location" section states: "can then visualize data and transform it using [ThingSpeak Apps](#)". Below the "Show Location" section, there is a link to "Learn More".

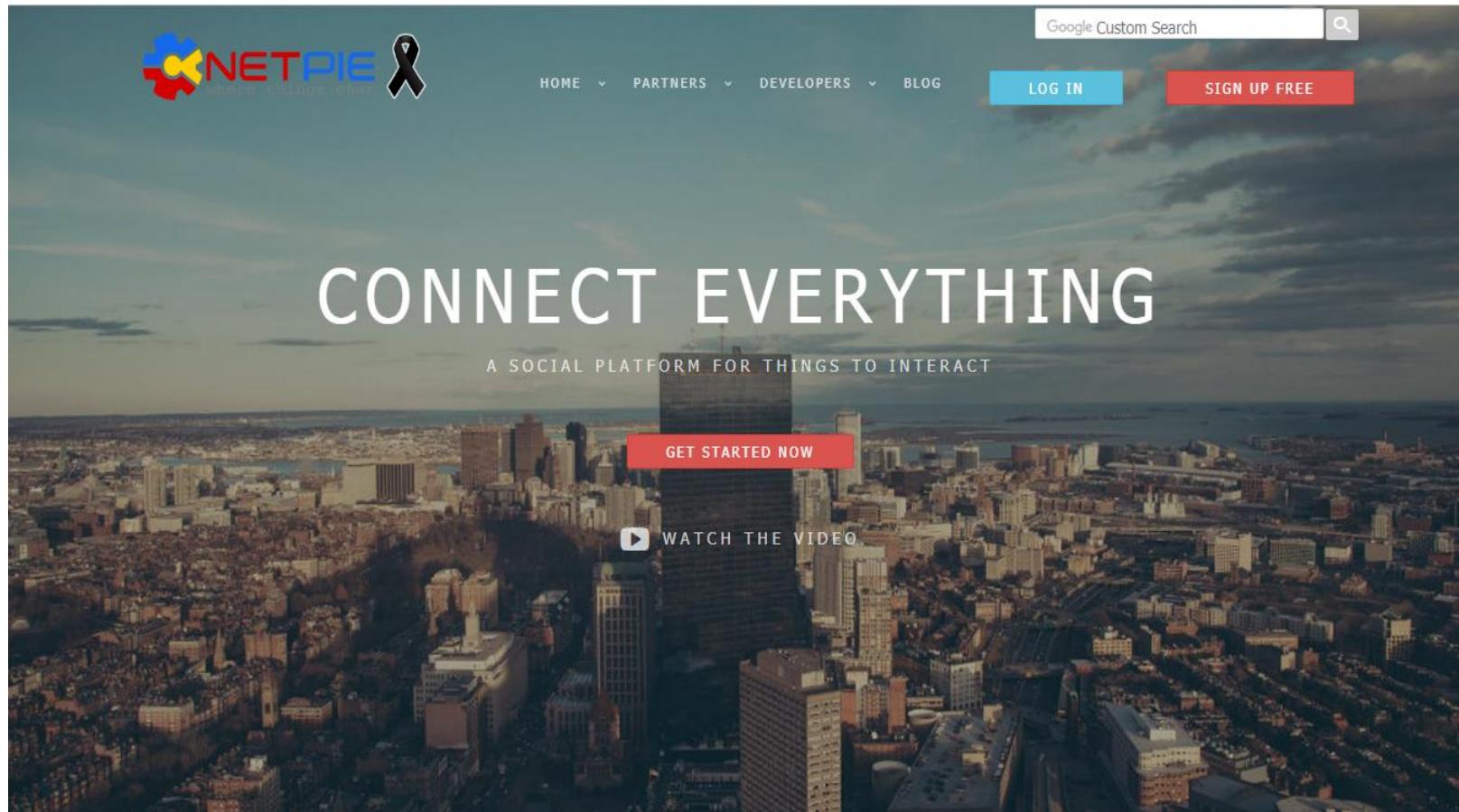
การสร้าง Channel Thinkspeak

○ หน้า Channel ที่สร้างสำเร็จ

The screenshot shows the ThingSpeak channel 'Test' interface. At the top, there is a navigation bar with links for ThingSpeak™, Channels, Apps, Community, Support, How to Buy, Account, and Sign Out. Below the navigation bar, the channel name 'Test' is displayed. Underneath the channel name, it shows the Channel ID: 300934, Author: a5830033, and Access: Private. There are tabs for Private View (selected), Public View, Channel Settings, API Keys, and Data Import / Export. Below these tabs are buttons for Add Visualizations, Data Export, MATLAB Analysis, and MATLAB Visualization. The 'Channel Stats' section indicates the channel was created and updated less than a minute ago, and has 0 entries. A chart titled 'Field 1 Chart' shows a single data point labeled 'Test'. The chart has a legend on the left labeled 'Field Label 1'.

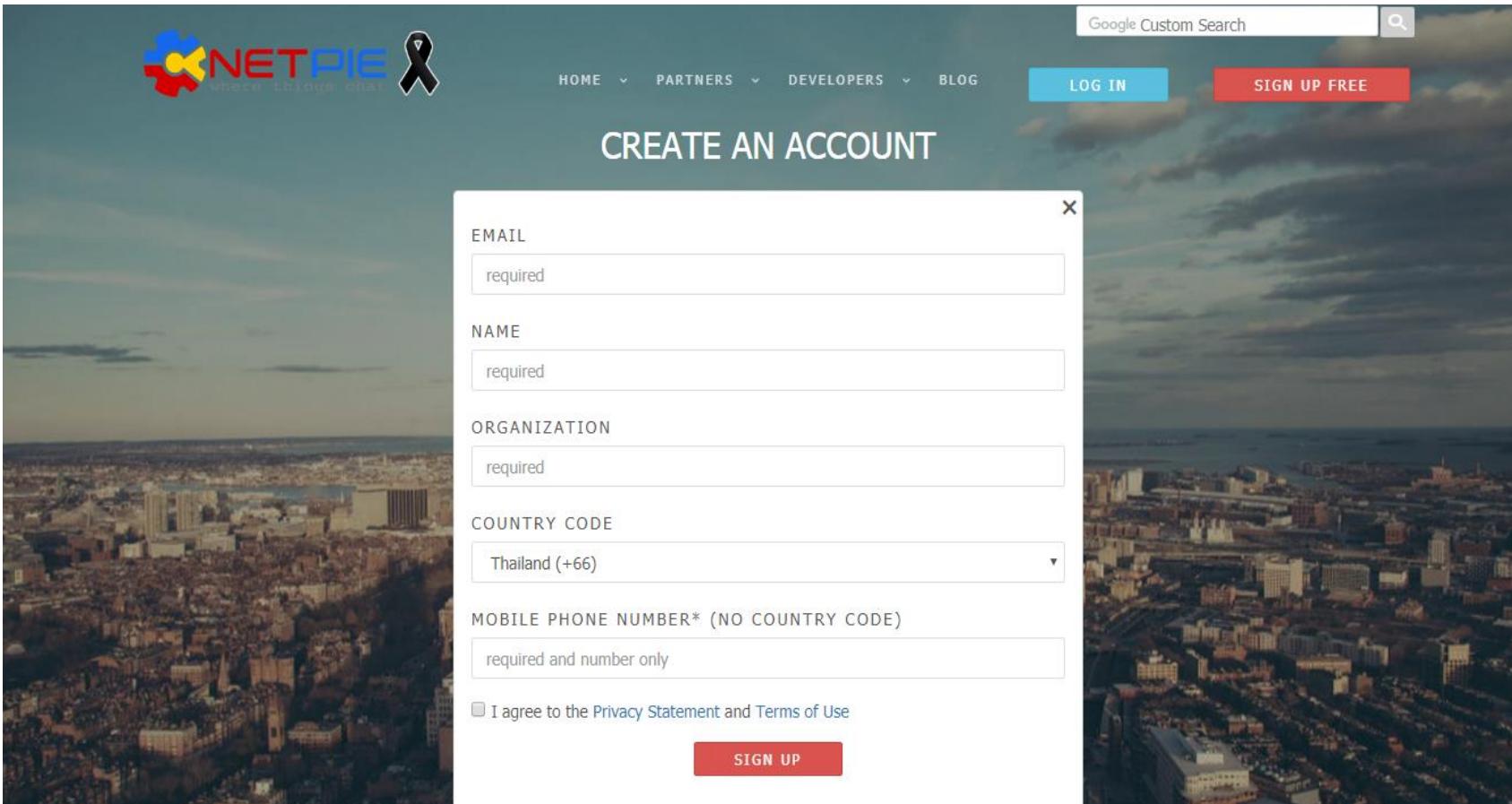
การใช้งาน NETPIE.io

○ เข้าไปยังหน้าเว็บ <https://netpie.io/> และกดปุ่ม SIGNUP FREE เพื่อสมัครสมาชิก



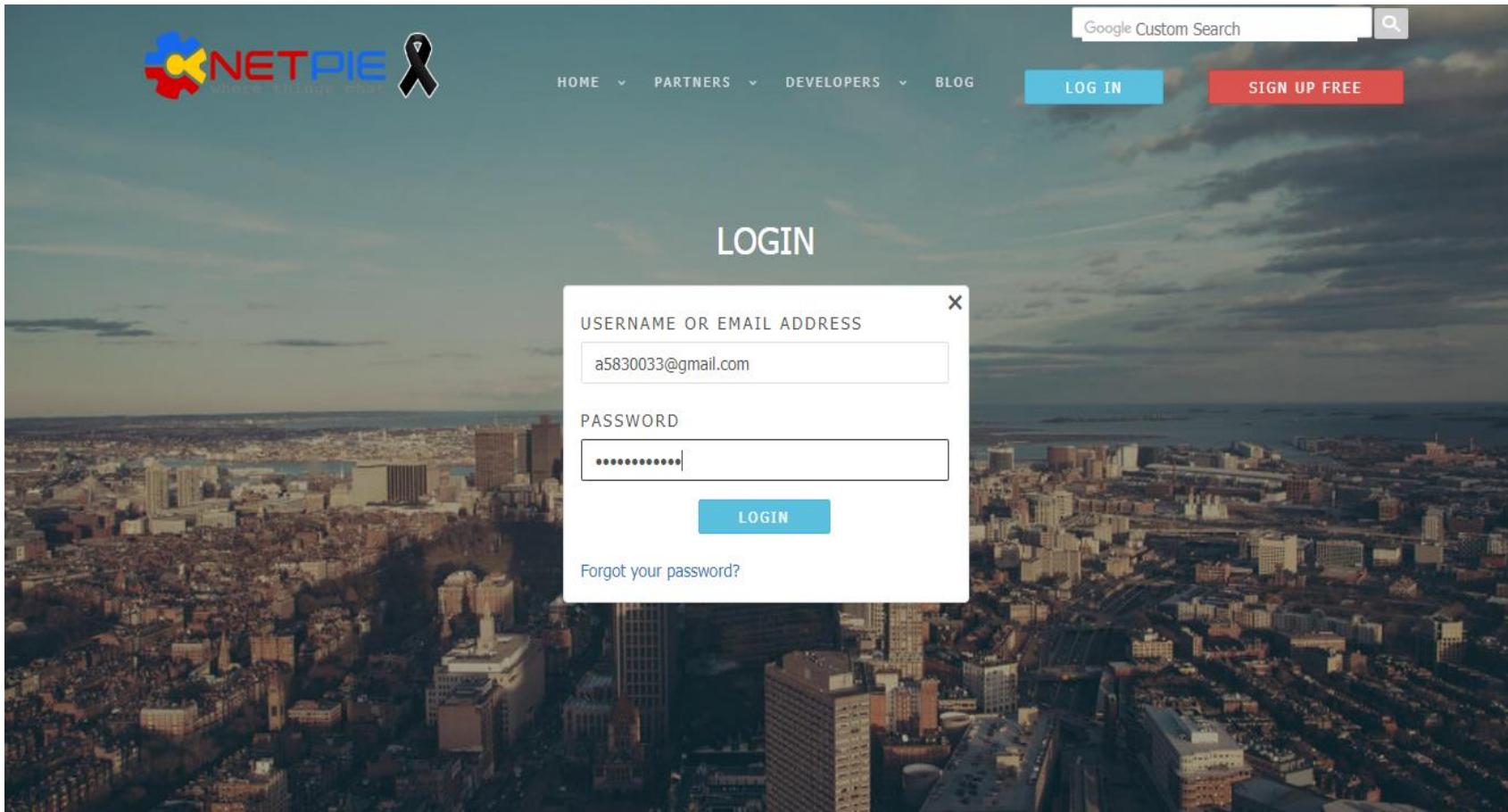
การใช้งาน NETPIE.io

○ ทำการกรอกข้อมูล เมื่อกรอกเสร็จแล้วกดปุ่ม SIGN UP



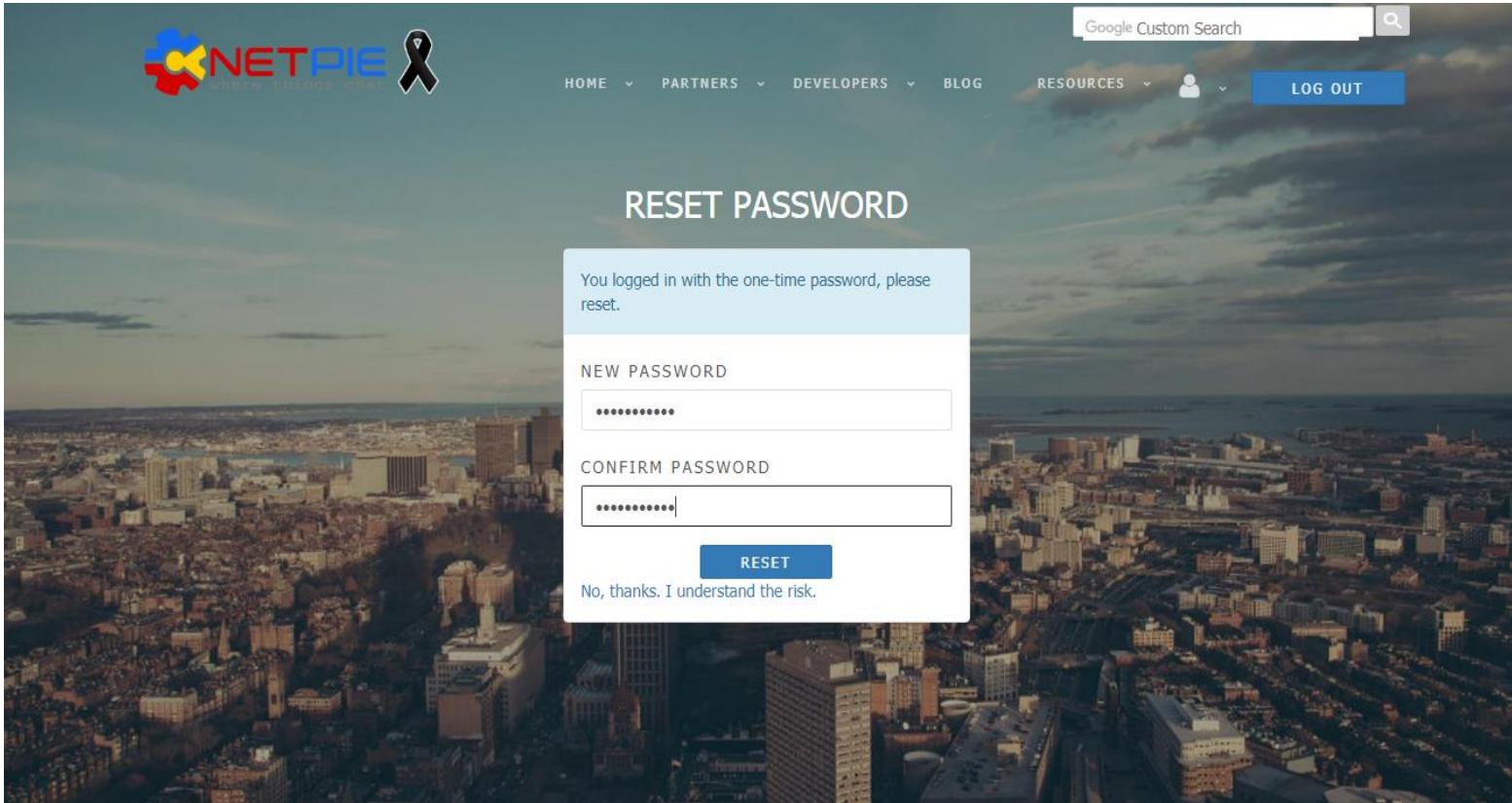
การใช้งาน NETPIE.io

- เมื่อสมัครเสร็จแล้วจะมีรหัสในการเข้าใช้งานครั้งแรกส่งไปยังเมล์อิเมลให้นำรหัสนั้นมาใช้ในการ Log In เข้าใช้งาน



การใช้งาน NETPIE.io

- เมื่อเข้าใช้งานครั้งแรกก็สามารถเปลี่ยนรหัสผ่านที่จะใช้ได้



- วิธีการใช้งานสามารถดูได้จาก <https://netpie.io/getstarted>