NODE MCU - MQTT

Download and install Nodejs

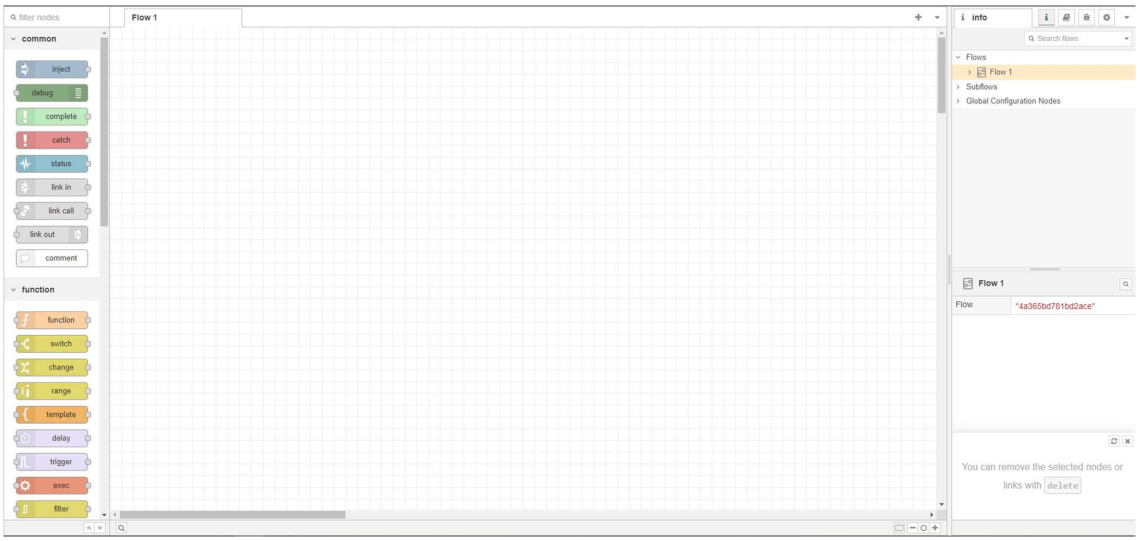
https://nodejs.org/en/download

Install nod-red

```
npm install -g --unsafe-perm node-red
```

```
Microsoft Windows [Version 10.0.10586]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\MyPC>npm install -g --unsafe-perm node-red

added 292 packages in 23s

40 packages are looking for funding
  run `npm fund` for details
npm notice
npm notice New minor version of npm available! 9.5.0 -> 9.6.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.6.2
npm notice Run npm install -g npm@9.6.2 to update!
npm notice

C:\Users\MyPC>node-red
22 Mar 08:51:36 - [info]

Welcome to Node-RED
===================

22 Mar 08:51:36 - [info] Node-RED version: v3.0.2
22 Mar 08:51:36 - [info] Node.js  version: v18.15.0
22 Mar 08:51:36 - [info] Windows_NT 10.0.10586 x64 LE
22 Mar 08:51:38 - [info] Loading palette nodes
22 Mar 08:51:39 - [info] Settings file  : C:\Users\MyPC\.node-red\settings.js
22 Mar 08:51:39 - [info] Context store  : 'default' [module=memory]
22 Mar 08:51:39 - [info] User directory : C:\Users\MyPC\.node-red
22 Mar 08:51:39 - [warn] Projects disabled : editorTheme.projects.enabled=false
22 Mar 08:51:39 - [info] Flows file     : C:\Users\MyPC\.node-red\flows.json
22 Mar 08:51:39 - [info] Creating new flow file
22 Mar 08:51:39 - [warn]

---------------------------------------------------------------------
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
---------------------------------------------------------------------

22 Mar 08:51:39 - [info] Server now running at http://127.0.0.1:1880/
22 Mar 08:51:39 - [warn] Encrypted credentials not found
22 Mar 08:51:39 - [info] Starting flows
22 Mar 08:51:39 - [info] Started flows
```

Giao diện:

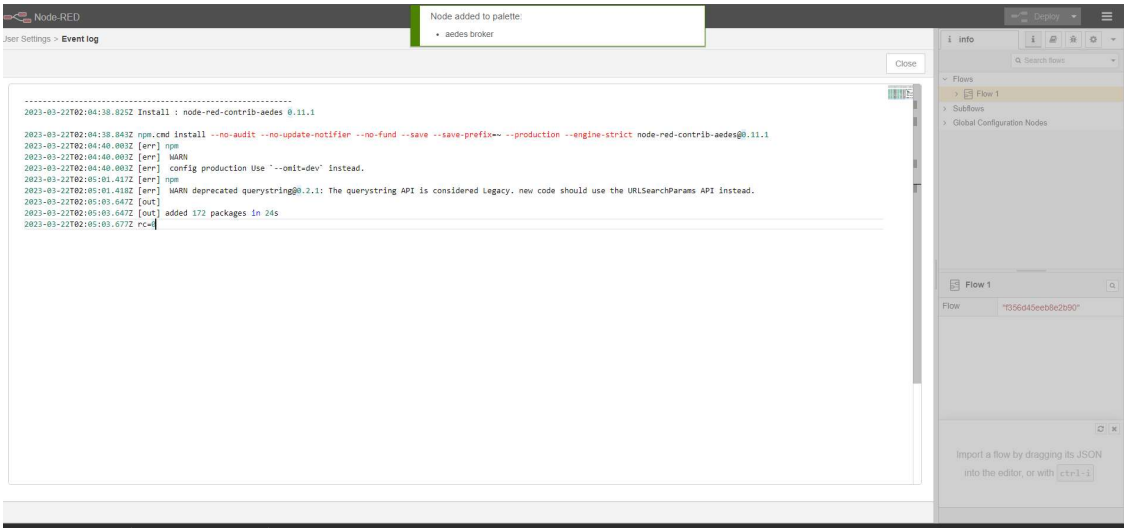


Cài đặt gói thư viện:

node-red-dashboard 3.4.0

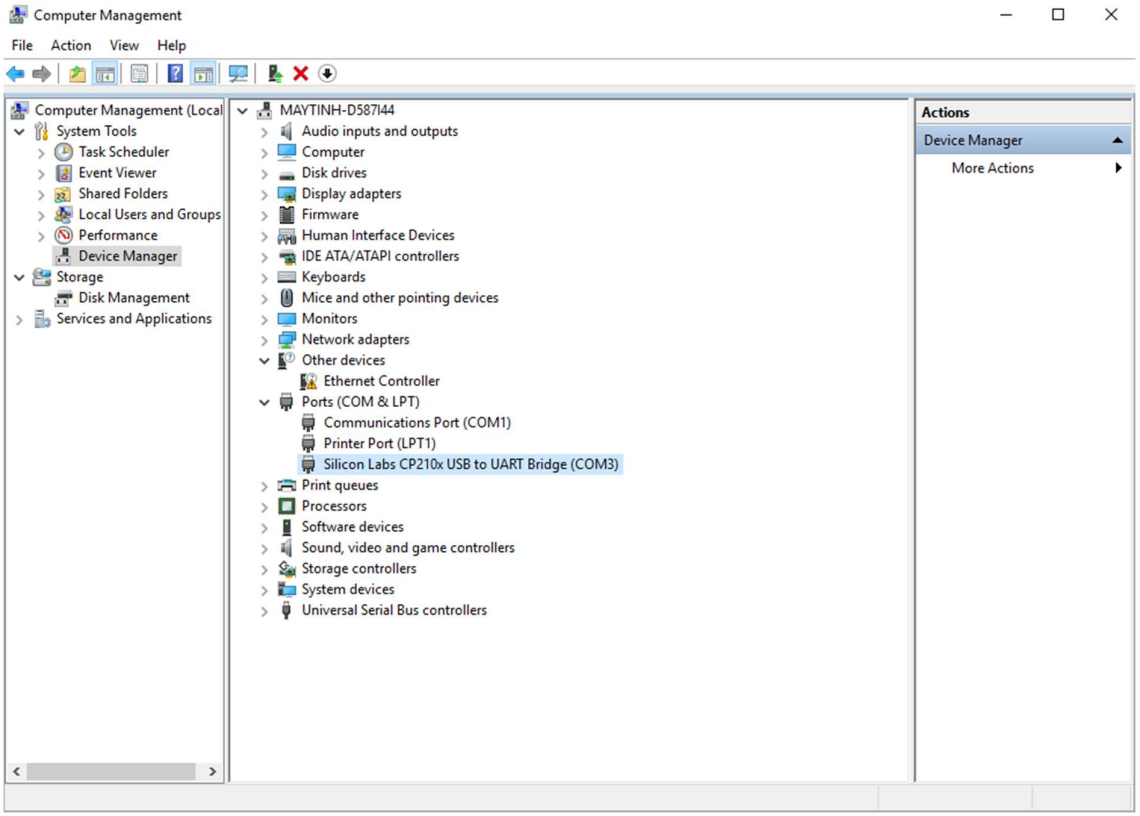
```
npm i node-red-dashboard
```

gói aedes làm MQTT

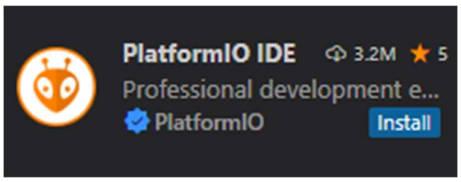


Download drive foe ESP8266 -

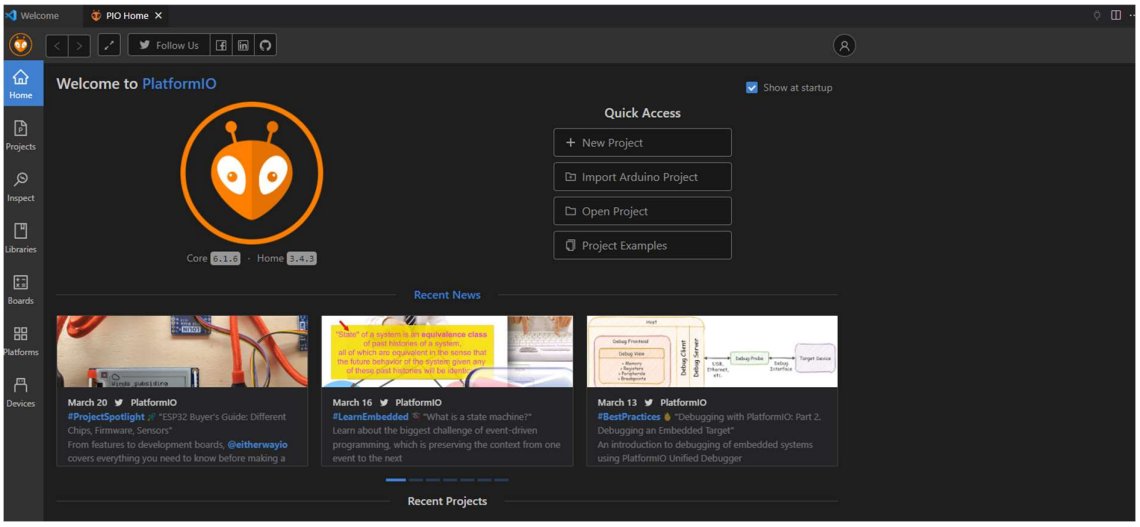
[CP210x Windows Drivers]



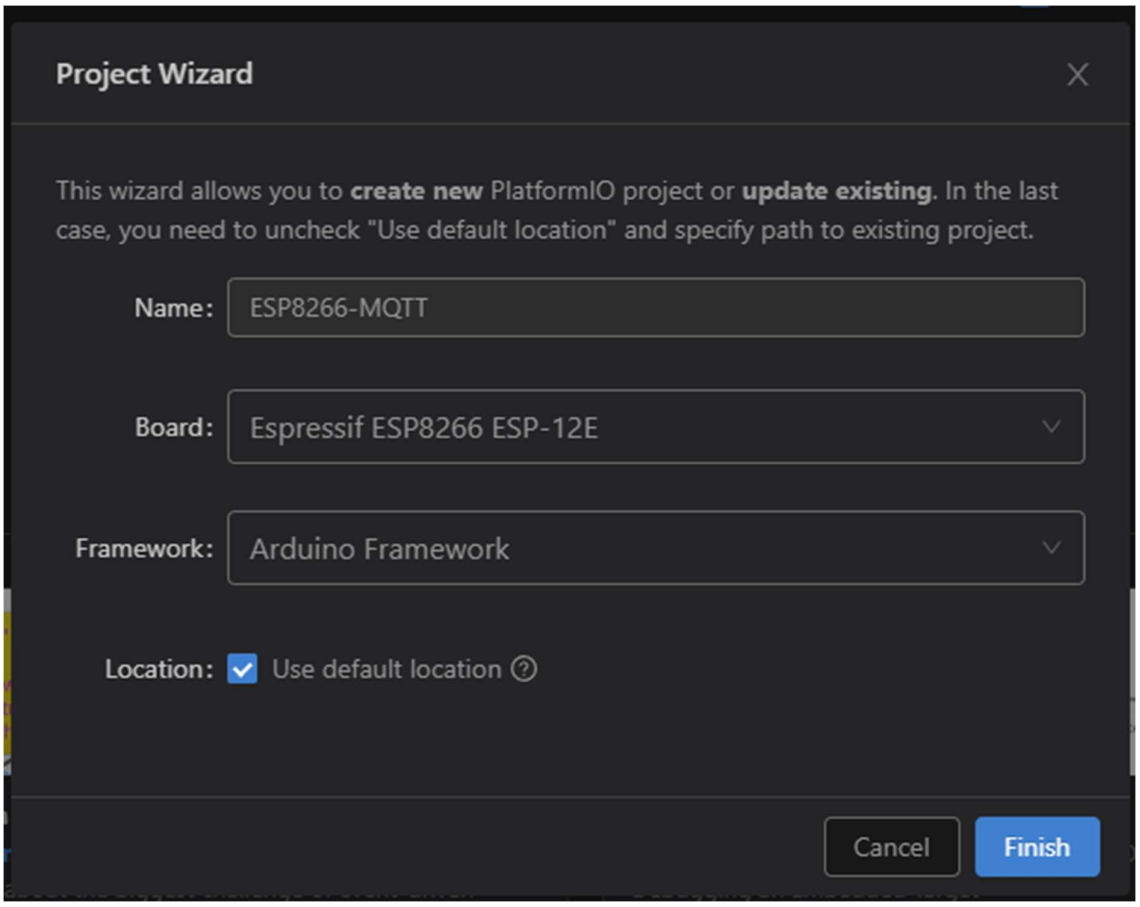
Open Visual Code download extensions: PlatformIO IDE đề lập trình Arduino IDE.



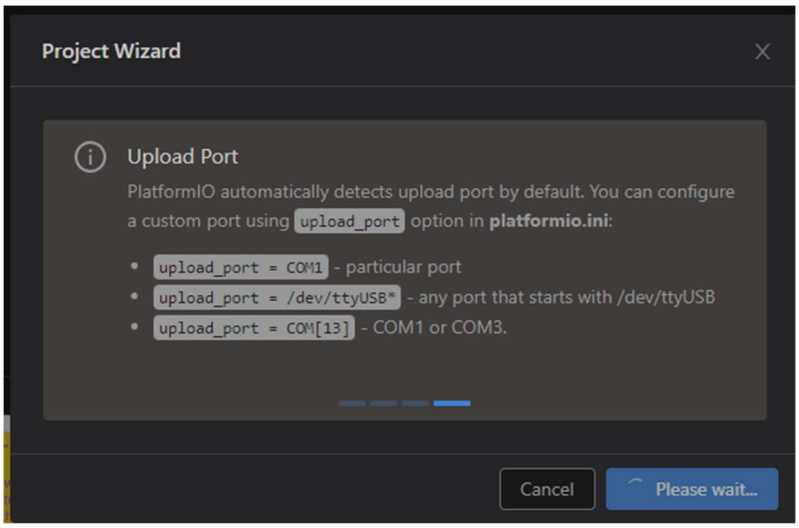
Tạo project:



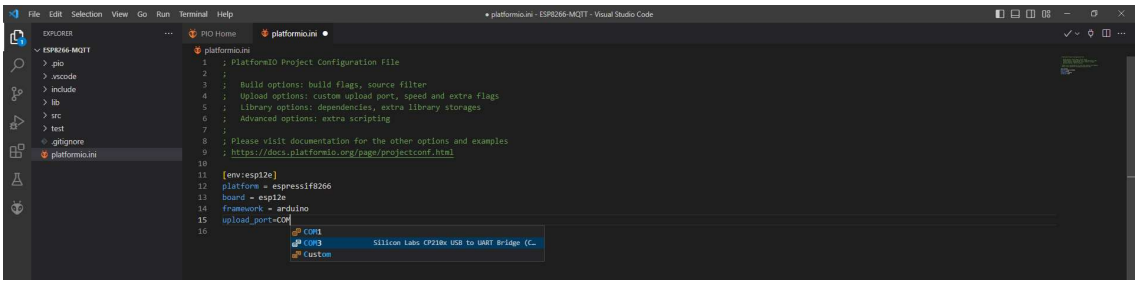
Làm việc với ESP8266



Khi làm việc lần đầu đợi download library



Khởi động chương trình:



Bài 1: Code demo with Led

```
#include<Arduino.h>
#define LED D1 // Led in NodeMCU at pin GPIO16 (D0).
void setup() {
pinMode(LED, OUTPUT); // set the digital pin as output.
}
void loop() {
digitalWrite(LED, HIGH);// turn the LED off.(Note that LOW is the voltage
level but actually
                        //the LED is on; this is because it is acive low on the
ESP8266.
delay(1000);            // wait for 1 second.
digitalWrite(LED, LOW); // turn the LED on.
delay(1000);           // wait for 1 second.
}
```

Bài 3: thực hiện tương tự với DHT11

Bài 3: Code mẫu kết nối ESP8266 to MQTT Server:

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

// WiFi credentials
const char* ssid = "your_SSID";
const char* password = "your_PASSWORD";

// MQTT broker details
const char* mqtt_server = "your_MQTT_broker_IP_address";
const int mqtt_port = 1883;
const char* mqtt_username = "your_MQTT_username";
const char* mqtt_password = "your_MQTT_password";

// MQTT client name
const char* client_id = "ESP8266Client";

WiFiClient wifi_client;
```

```
PubSubClient mqtt_client(wifi_client);

void setup() {
  // Connect to WiFi network
  Serial.begin(9600);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");

  // Connect to MQTT broker
  mqtt_client.setServer(mqtt_server, mqtt_port);
  while (!mqtt_client.connected()) {
    Serial.println("Connecting to MQTT broker...");
    if (mqtt_client.connect(client_id, mqtt_username, mqtt_password)) {
      Serial.println("Connected to MQTT broker");
    } else {
      Serial.print("Failed to connect to MQTT broker, rc=");
      Serial.print(mqtt_client.state());
      Serial.println(" retrying in 5 seconds");
      delay(5000);
    }
  }

  // Subscribe to a topic
  mqtt_client.subscribe("your_topic");
}

void loop() {
  // Reconnect to MQTT broker if connection lost
  if (!mqtt_client.connected()) {
    Serial.println("Connection lost, reconnecting to MQTT broker...");
    if (mqtt_client.connect(client_id, mqtt_username, mqtt_password)) {
      Serial.println("Connected to MQTT broker");
      mqtt_client.subscribe("your_topic");
    } else {
      Serial.print("Failed to connect to MQTT broker, rc=");
      Serial.print(mqtt_client.state());
      Serial.println(" retrying in 5 seconds");
      delay(5000);
    }
  }

  // Publish a message
  mqtt_client.publish("your_topic", "Hello from ESP8266!");
```

```
  // Wait for a few seconds
  delay(5000);
}
```