



MicroPython and IoT

อุปกรณ์ที่ใช้

Board (เลือกมา 1 ตัว)

- ESP32 รุ่นใดก็ได้
- ESP8266 รุ่นใดก็ได้
- NodeMCU รุ่นใดก็ได้
- PyBoard รุ่นใดก็ได้ (ในไทยไม่น่ามีขาย)
- สาย USB ต่อเข้าบอร์ด

Component

- LCD 16x2 (เอาแบบมี i2c จะดี)
- LED + resistor 220 Ohm
- DHT22/ DHT11 - เอาแบบที่มี 3 ขา
- Switch
- Relay 2 Channel ++
- Computer
- WiFi

ดาวน์โหลดโปรแกรม

1.Fritzing

ดาวน์โหลดที่ <https://www.filecroco.com/download-fritzing/download/>

The screenshot shows the FileCroco website interface. At the top, there's a navigation bar with icons for Home, Science, Fritzing 0.9.3b, Downloading, Facebook, and Twitter. Below the navigation is a large green header featuring a cartoon crocodile and the word "FileCroco". The main content area has a white background. On the left, there's a red box containing the Fritzing logo and the text "Fritzing" followed by "October 16th, 2020 - Free - 100% Safe". It also shows a rating of 3.48 out of 5 based on 181 votes. A red-bordered button says "Choose a mirror" with a "Download now" button inside it. To the right of this is a "Categories" sidebar with a tree graphic. The categories listed are: Antispyware, Antivirus, Audio Tools, CD / DVD / Bluray, Compression and Backup, Desktop Tools, Developer Tools, Digital Photo / Image, Drivers, Email / IM / VoIP, File Sharing, File Transfer, Gaming, Mobile Phone Tools, Multimedia, Networking and Admin, Office Tools, Others, Security and Firewalls.

Fritzing
October 16th, 2020 - Free - 100% Safe
★ ★ ★ ★ ★ (181 votes, average: 3.48 out of 5)

Choose a mirror

Download now

Internal Mirror

About Fritzing

Fritzing is an easy-to-use design application aimed towards designers, engineers, and other creatives to work on their projects and create amazing designs with this design automation tool. If you're a designer or an engineer, then you need to know about Fritzing. It's a comprehensive and functional piece of software that will help you create amazing...

Latest updates

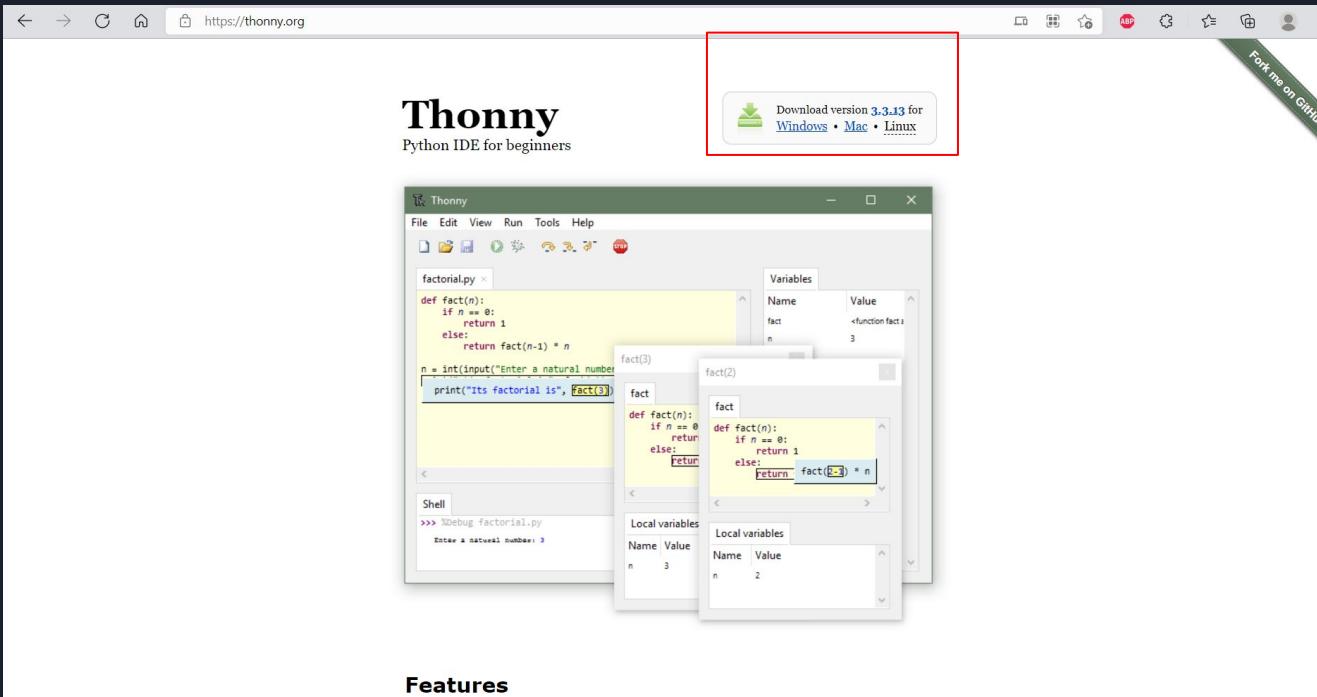
E Estlcam 11.242
S SimplexNumerica 21.1.2.0
W Wabbitemu 1.9.5.22
L Linear Interpolation Calculator 1.2
A 4nec2 5.8.16

Categories

- Antispyware
- Antivirus
- Audio Tools
- CD / DVD / Bluray
- Compression and Backup
- Desktop Tools
- Developer Tools
- Digital Photo / Image
- Drivers
- Email / IM / VoIP
- File Sharing
- File Transfer
- Gaming
- Mobile Phone Tools
- Multimedia
- Networking and Admin
- Office Tools
- Others
- Security and Firewalls

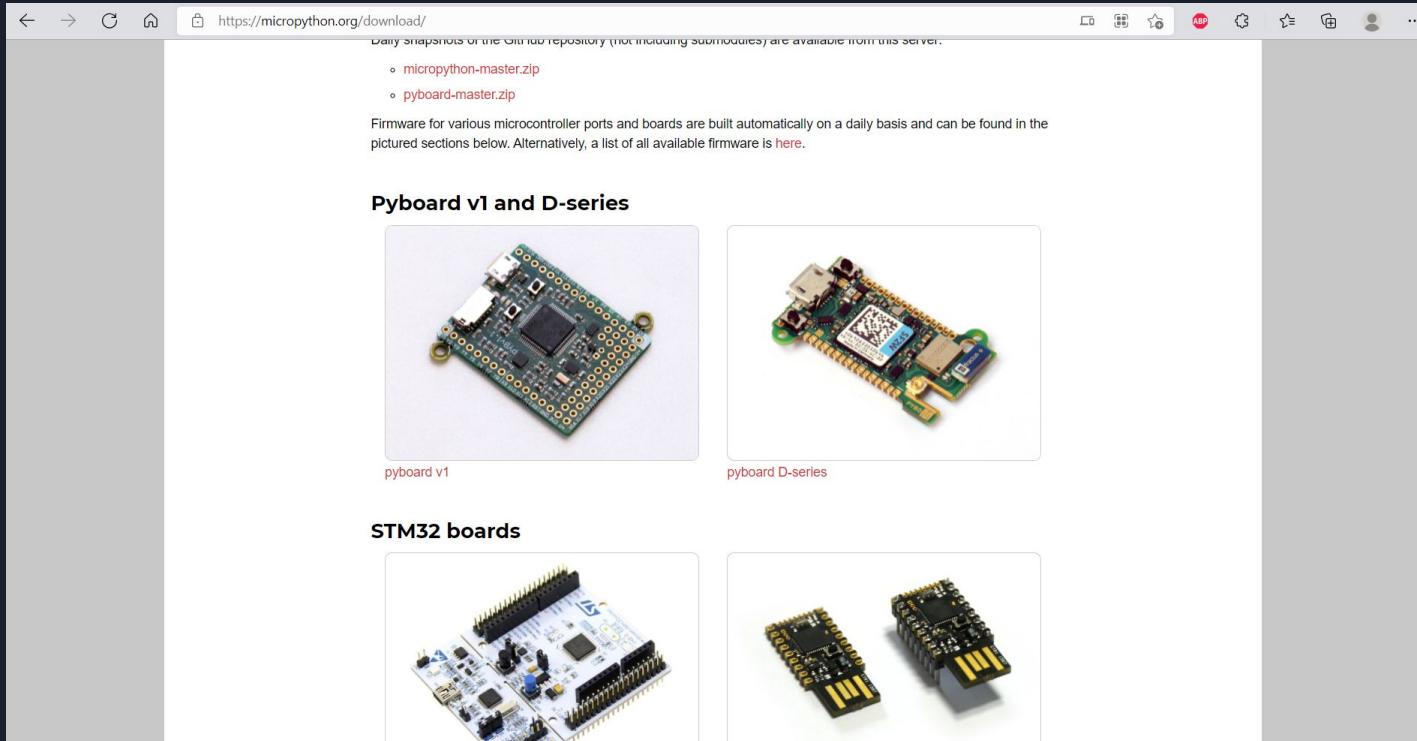
ดาวน์โหลดโปรแกรม

2. ดาวน์โหลดโปรแกรม Thonny ที่ลิ้งนี้ <https://thonny.org/> คลิกที่ตาม OS ของตัวเองที่มุ่งข้างบน และติดตั้งให้เรียบร้อย



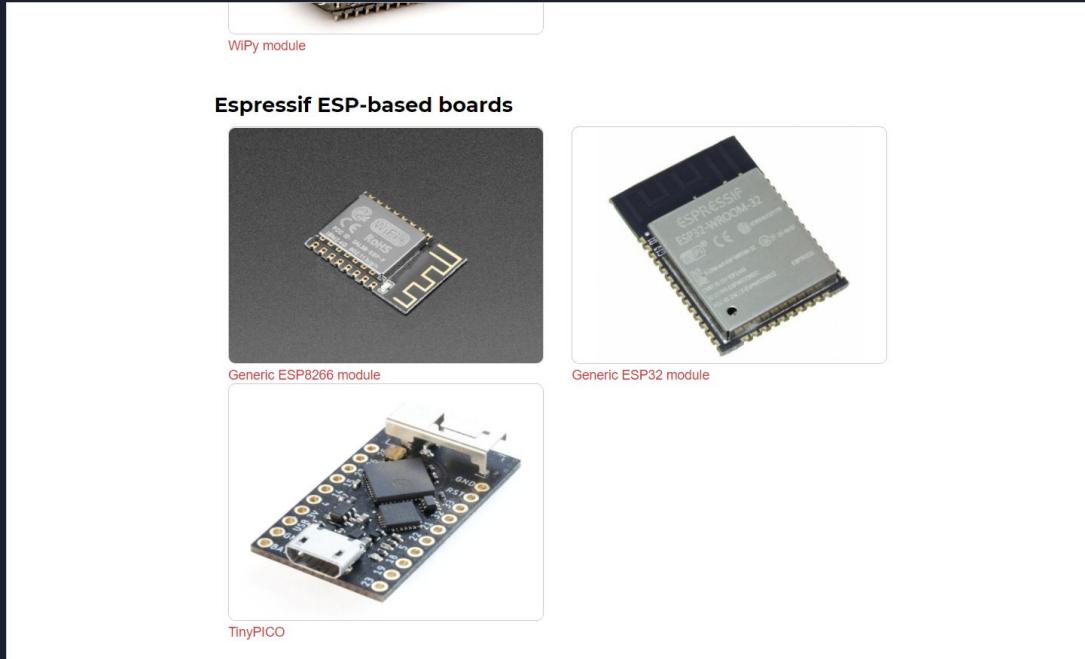
วิธีลง firmware

เข้าไปที่เว็บไซต์ <https://micropython.org/download/>



วิธีลง firmware

คลิกไปที่บอร์ดที่ตัวเองใช้



วิธีลง firmware

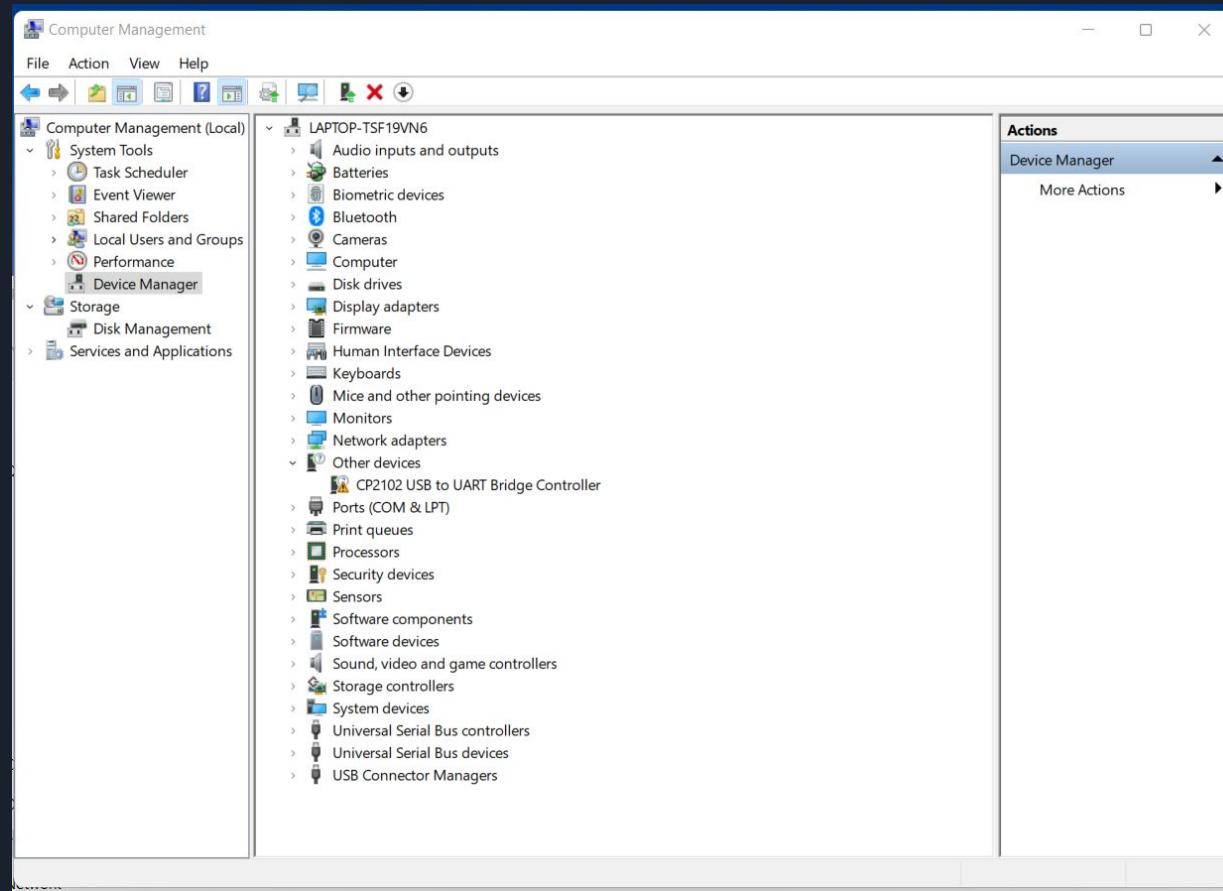
เลือนลงมาหาเวอร์ชั่นของ firmware ตามภาพ และนำเลือกเป็นเวอร์ชั่นที่ stable แล้ว ในที่นี่เลือก
เวอร์ชั่นที่เป็น stable แล้วล่าสุด ตามลูกศร

Firmware with ESP-IDF v4.x

Firmware built with ESP-IDF v4.x, with support for BLE, LAN and PPP.

- GENERIC : [esp32-20211008-unstable-v1.17-74-gd42cba0d2.bin](#)
- GENERIC : [esp32-20210924-unstable-v1.17-68-g35fb90bd5.bin](#)
- GENERIC : [esp32-20210921-unstable-v1.17-63-g67d1dca9c.bin](#)
- GENERIC : [esp32-20210921-unstable-v1.17-59-g782d5b2e5.bin](#)
- GENERIC : [esp32-20210902-v1.17.bin](#) ←
- GENERIC : [esp32-20210623-v1.16.bin](#)
- GENERIC : [esp32-20210418-v1.15.bin](#)
- GENERIC : [esp32-idf4-20210202-v1.14.bin](#)
- GENERIC : [esp32-idf4-20200902-v1.13.bin](#)
- GENERIC : [esp32-idf4-20191220-v1.12.bin](#)
- GENERIC-SPIRAM : [esp32spiram-20211008-unstable-v1.17-74-gd42cba0d2.bin](#)
- GENERIC-SPIRAM : [esp32spiram-20210924-unstable-v1.17-68-g35fb90bd5.bin](#)
- GENERIC-SPIRAM : [esp32spiram-20210921-unstable-v1.17-63-g67d1dca9c.bin](#)
- GENERIC-SPIRAM : [esp32spiram-20210921-unstable-v1.17-59-g782d5b2e5.bin](#)
- GENERIC-SPIRAM : [esp32spiram-20210902-v1.17.bin](#)
- GENERIC-SPIRAM : [esp32spiram-20210623-v1.16.bin](#)
- GENERIC-SPIRAM : [esp32spiram-20210418-v1.15.bin](#)
- GENERIC-SPIRAM : [esp32spiram-idf4-20210202-v1.14.bin](#)
- GENERIC-SPIRAM : [esp32spiram-idf4-20200902-v1.13.bin](#)
- GENERIC-SPIRAM : [esp32spiram-idf4-20191220-v1.12.bin](#)

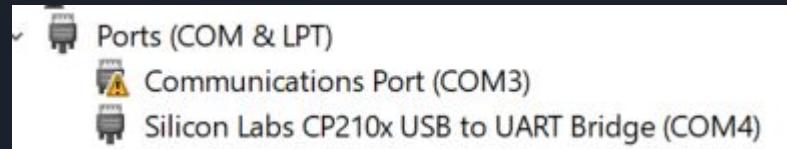
Manage



วิธีลง firmware

ขั้นตอนต่อไปให้ทำการเสียบบอร์ด เข้ากับ Computerแล้วทำการตามขั้นตอนต่อไปนี้

1. ให้เช็ค port ว่าที่เสียบบอร์ดเข้า Computer อยู่ port ที่เท่าไหร่ เช็คโดยการ คลิกขวาที่ This PC > เลือก Manage > เลือก Device Manager เลื่อนหา Ports(COM & LPT)



หากมองไม่เห็น port ให้โหลดตามลิ้งนี้

<https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers>

ดาวน์โหลดและติดตั้งให้เรียบร้อยแล้วไปเช็คที่ Ports(COM & LPT)อีกครั้ง

วิธีลง firmware ผ่าน CMD / Terminal

2. เปิด cmd แล้ว cd เข้าไปในโฟลเดอร์ที่โหลด firmware ไว้ แล้วใช้คำสั่งตามนี้

- ก่อนรันคำสั่งให้ติดตั้ง pip install esptool ก่อนเลย (mac ใช้ pip3 install esptool)
ESP32
Erase(ลบ firmware เก่าที่ลงไว้ในบอร์ด)
Erase(ลบ firmware เก่าที่ลงไว้ในบอร์ด)

```
python -m esptool --chip esp32 --port COM6 erase_flash
## ตรง -- port ให้ใส่ port ตามเครื่องตัวเองที่เช็คในขั้นตอนที่ 1
```

Flash Bootloader(ลง firmware ใหม่)

```
python -m esptool --chip esp32 --port COM6 --baud 460800 write_flash -z 0x1000
esp32-20220117-v1.18.bin
## สีส้มคือให้ copy ชื่อของไฟล์ firmware ที่ดาวน์โหลดมาใส่
```



วิธีลง firmware

ESP8266

Command for Flash Bootloader

Erase

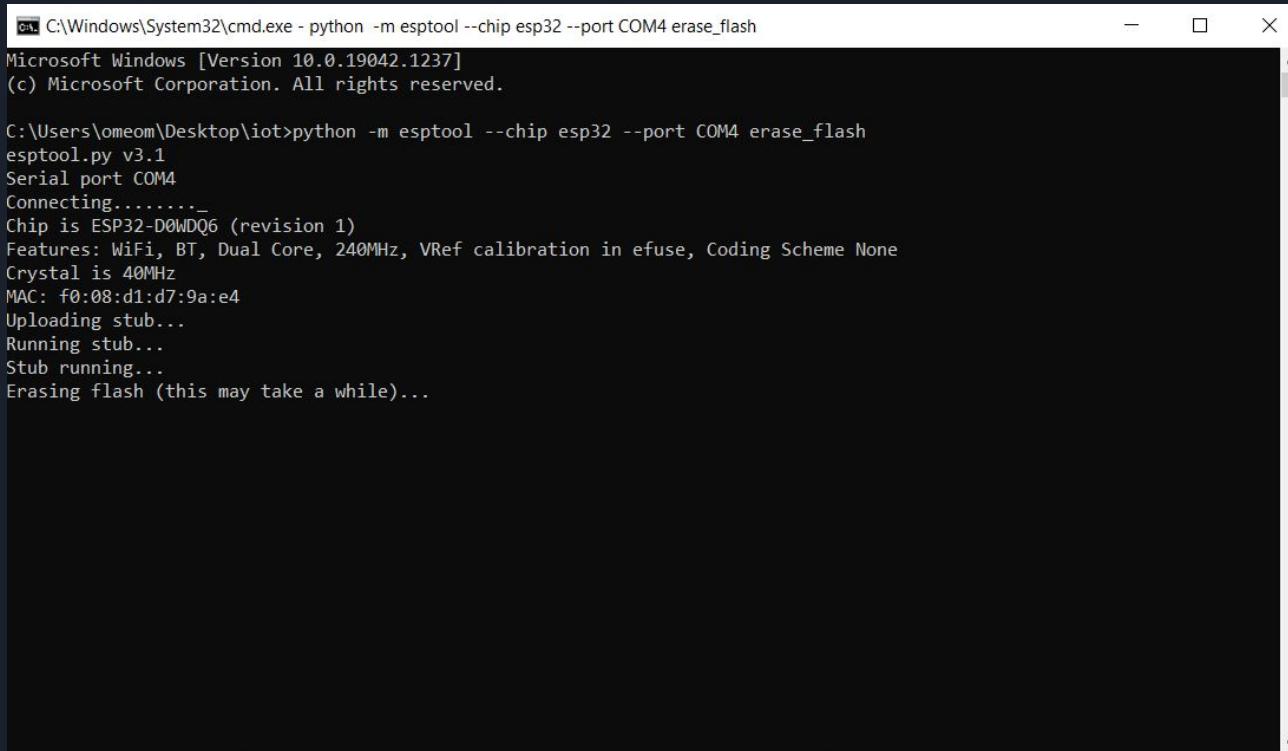
```
python -m esptool --chip esp8266 --port COM8 erase_flash  
## ตรง -- port ให้ใส่ port ตามเครื่องตัวเองที่เช็คในขั้นตอนที่ 1
```

Flash Bootloader

```
python -m esptool --chip esp8266 --port COM8 --baud 115200 write_flash  
--flash_size=detect -fm dio 0 esp8266-20200911-v1.13.bin  
## สีล้มคือให้ copy ชื่อของไฟล์ fiemware ที่ดาวน์โหลดมาใส่
```

วิธีลง firmware

ถ้าขึ้นตามภาพแสดงว่า Erase เรียบร้อย ให้ใส่คำสั่ง Flash Bootloader ต่อ



```
C:\Windows\System32\cmd.exe - python -m esptool --chip esp32 --port COM4 erase_flash
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\omeom\Desktop\iot>python -m esptool --chip esp32 --port COM4 erase_flash
esptool.py v3.1
Serial port COM4
Connecting.....
Chip is ESP32-D0WDQ6 (revision 1)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse, Coding Scheme None
Crystal is 40MHz
MAC: f0:08:d1:d7:9a:e4
Uploading stub...
Running stub...
Stub running...
Erasing flash (this may take a while)...
```

วิธีลง firmware

ถ้าขึ้นตามภาพแสดงว่า Flash Bootloader เรียบร้อย



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

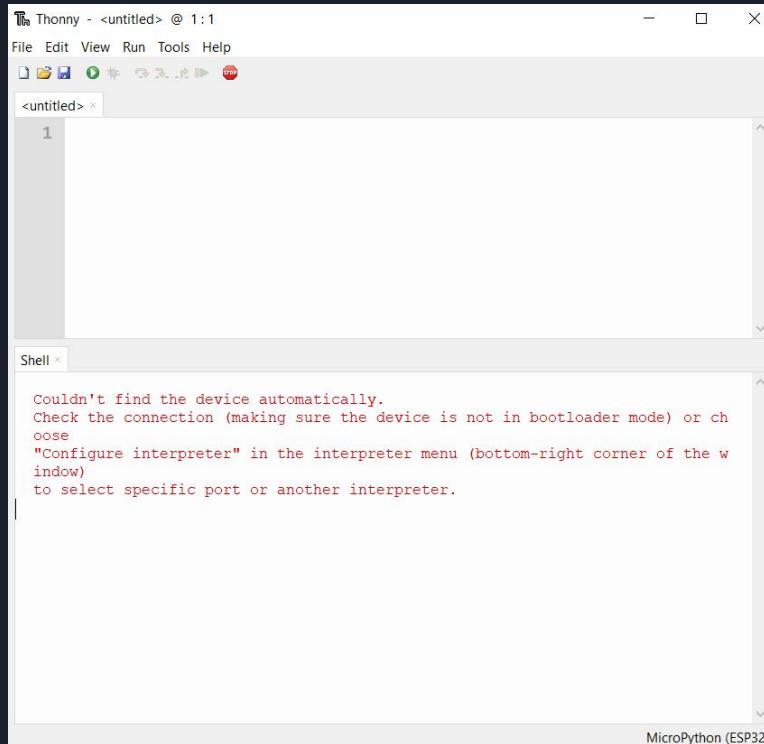
C:\Users\omeom\Desktop\iot>python -m esptool --chip esp32 --port COM4 --baud 460800 write_flash -z 0x1000 esp32-20210902-v1.17.bin
esptool.py v3.1
Serial port COM4
Connecting....
Chip is ESP32-D0WDQ6 (revision 1)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse, Coding Scheme None
Crystal is 40MHz
MAC: f0:08:d1:d7:9a:e4
Uploading stub...
Running stub...
Stub running...
Changing baud rate to 460800
Changed.
Configuring flash size...
Flash will be erased from 0x00001000 to 0x00175fff...
Compressed 1527504 bytes to 987584...
Wrote 1527504 bytes (987584 compressed) at 0x00001000 in 25.8 seconds (effective 473.1 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

C:\Users\omeom\Desktop\iot>
```

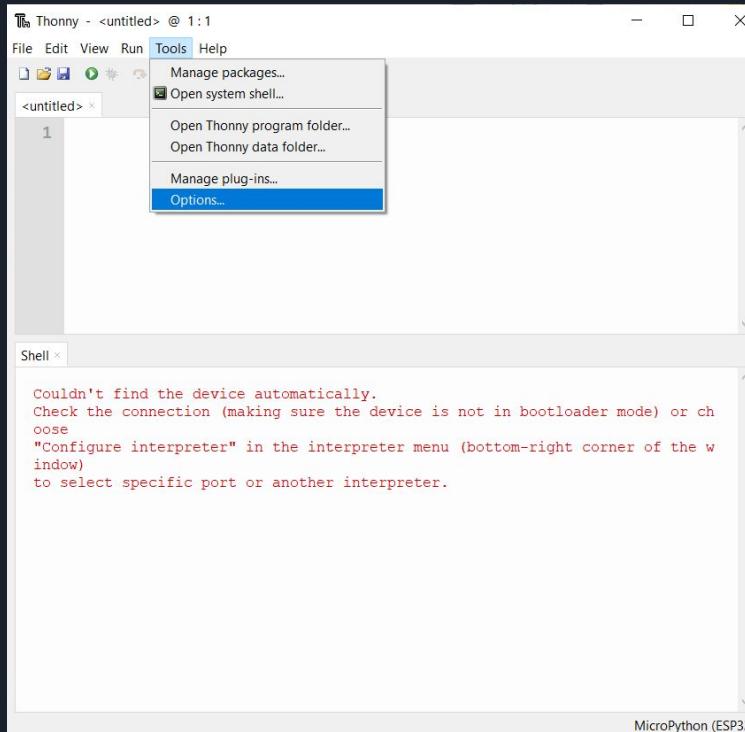
วิธีลง firmware ด้วย thonny

เปิดโปรแกรม thonny



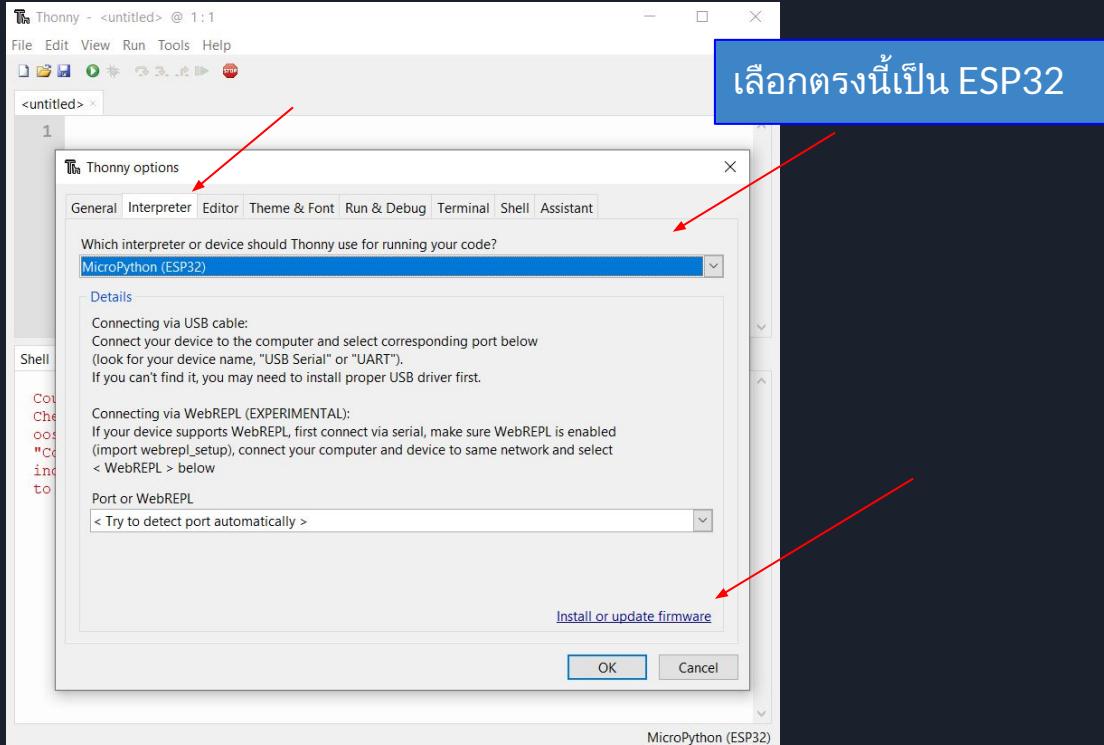
วิธีลง firmware ด้วย thonny

ไปที่ Tools > options



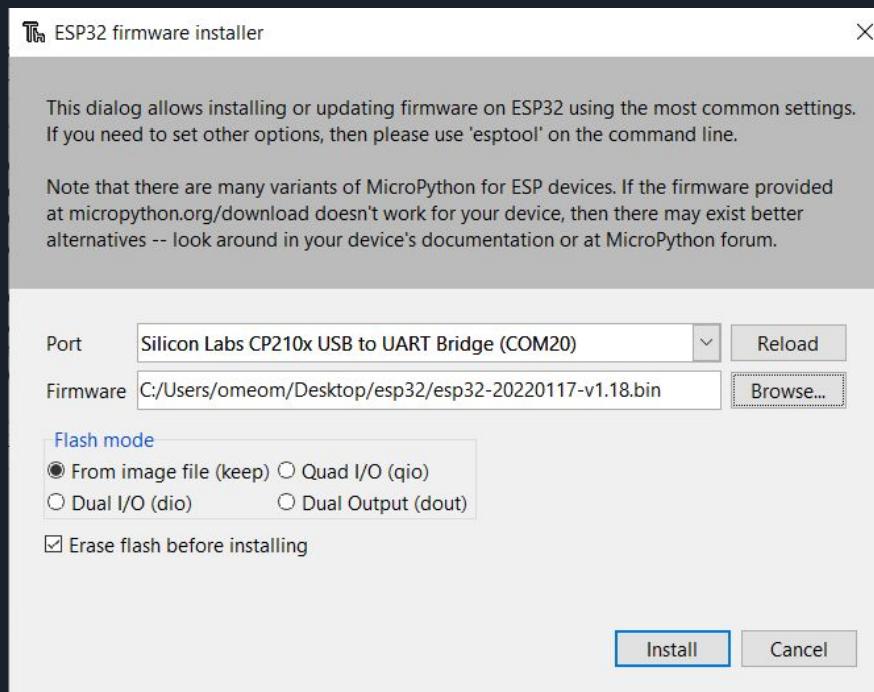
วิธีลง firmware ด้วย thonny

ไปที่ Interpreter คลิก Install ot update firmware



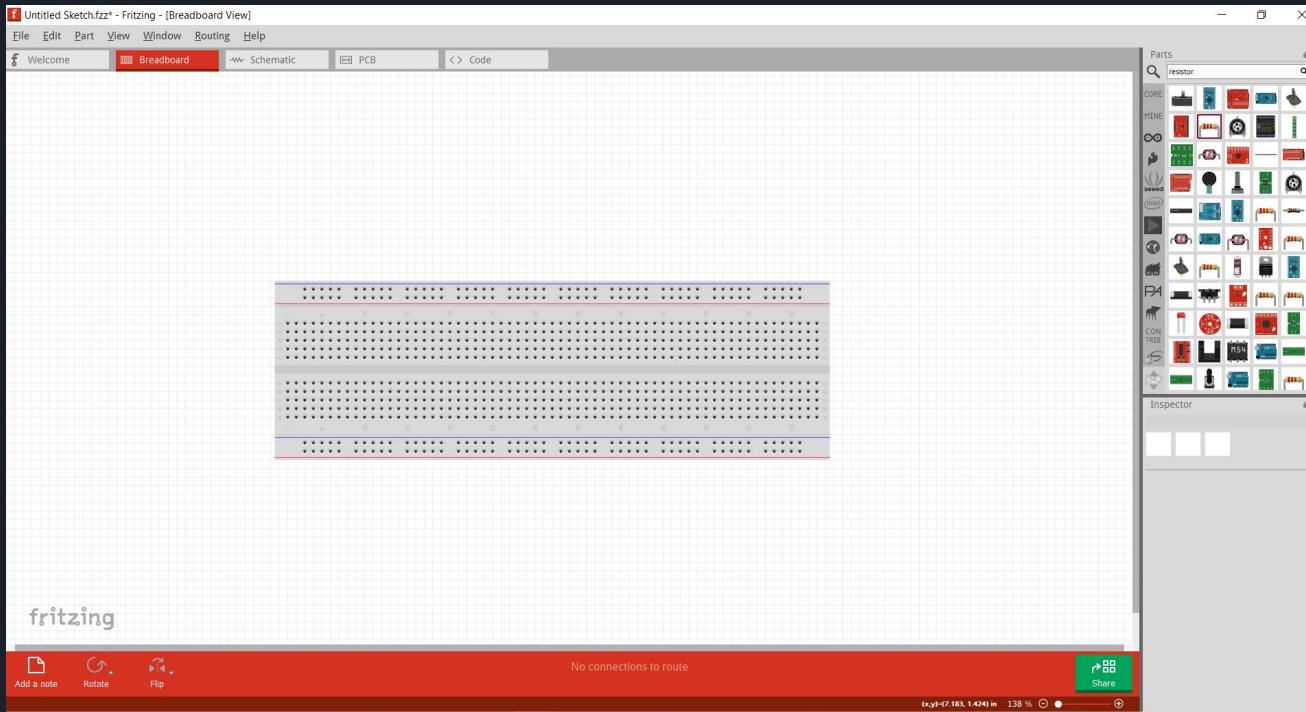
วิธีลง firmware ด้วย thonny

ตรงส่วน Port ให้เลือกไปยัง Port ที่เลี้ยง ESP32 อยู่ และตรง frimware ให้ Browse ไปที่ไฟล์ Firmware ที่โหลดมา แล้วกด Install รอจนเสร็จเป็นอันเรียบร้อย



ทดลองใช้โปรแกรม Fritzing ออกแบบบอร์ด LED

1. เปิด Fritzing เลือกไปที่ Breadboard



ทดลองใช้โปรแกรม Fritzing ออกแบบบอร์ด LED

2. ให้ดาวน์โหลดบอร์ดที่เราจะใช้

ESP32

<https://forum.fritzing.org/uploads/default/original/2X/8/8d99adbb2e8f966178bdb46f5cffa7bd72ce2c2a.fzpz>

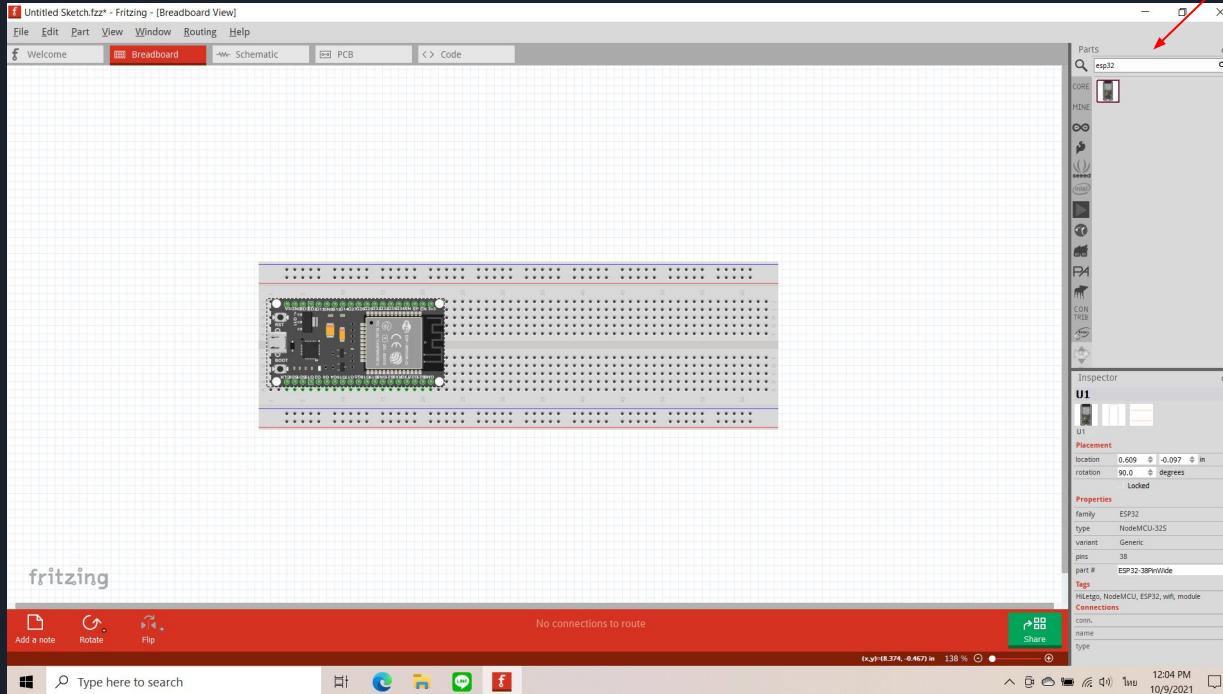
ESP8266

<https://github.com/roman-minyaylov/nodemcu-v3-fritzing>

ทดลองใช้โปรแกรม Fritzing ออกแบบบอร์ด LED

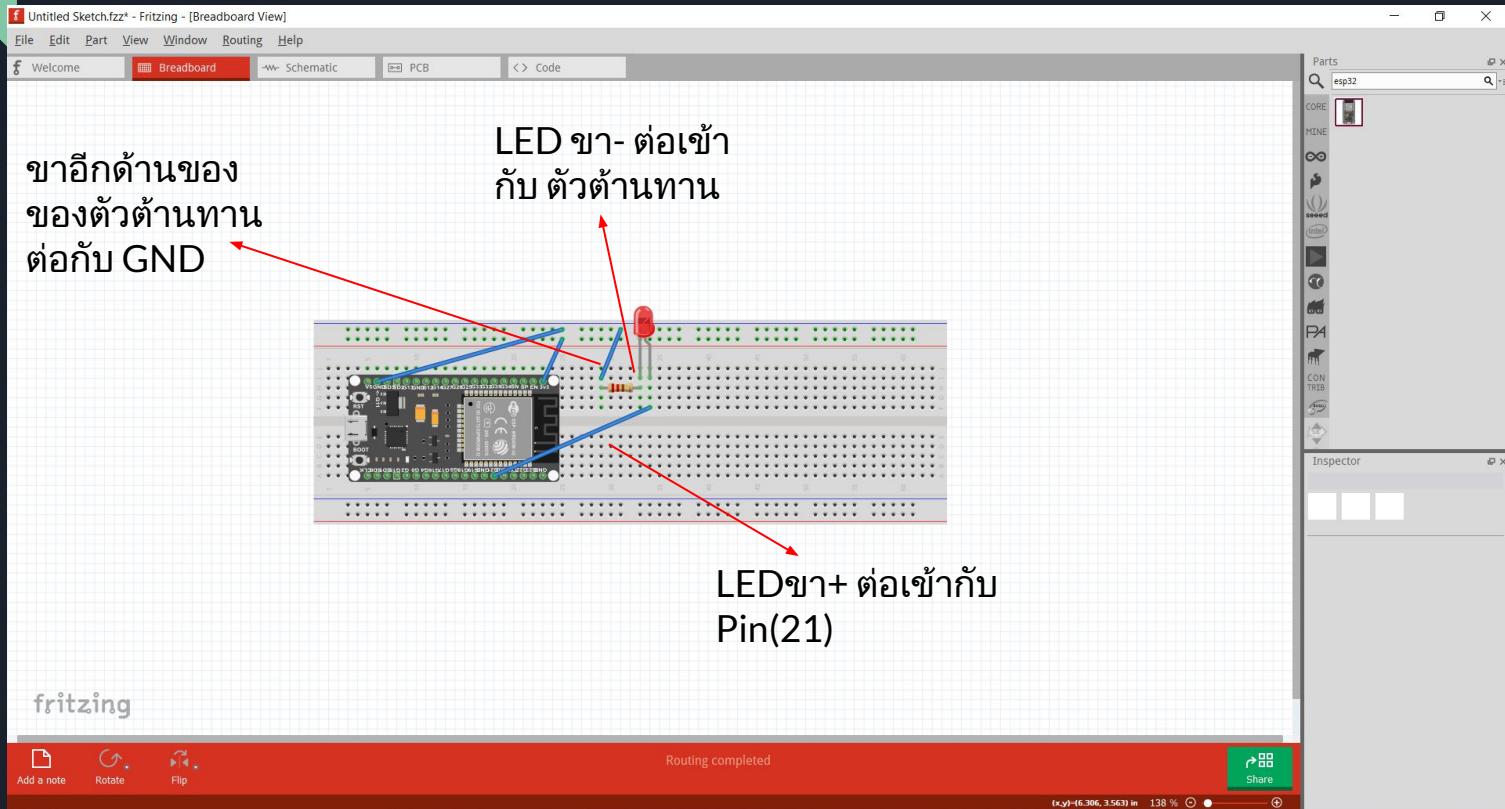
ช่องค้นหา

3. ดาวน์โหลดเสร็จแล้ว ให้พิมพ์หน้าบอร์ดตัวเองในช่องค้นหาตามภาพ แล้วลากมาไว้ที่ protoboard

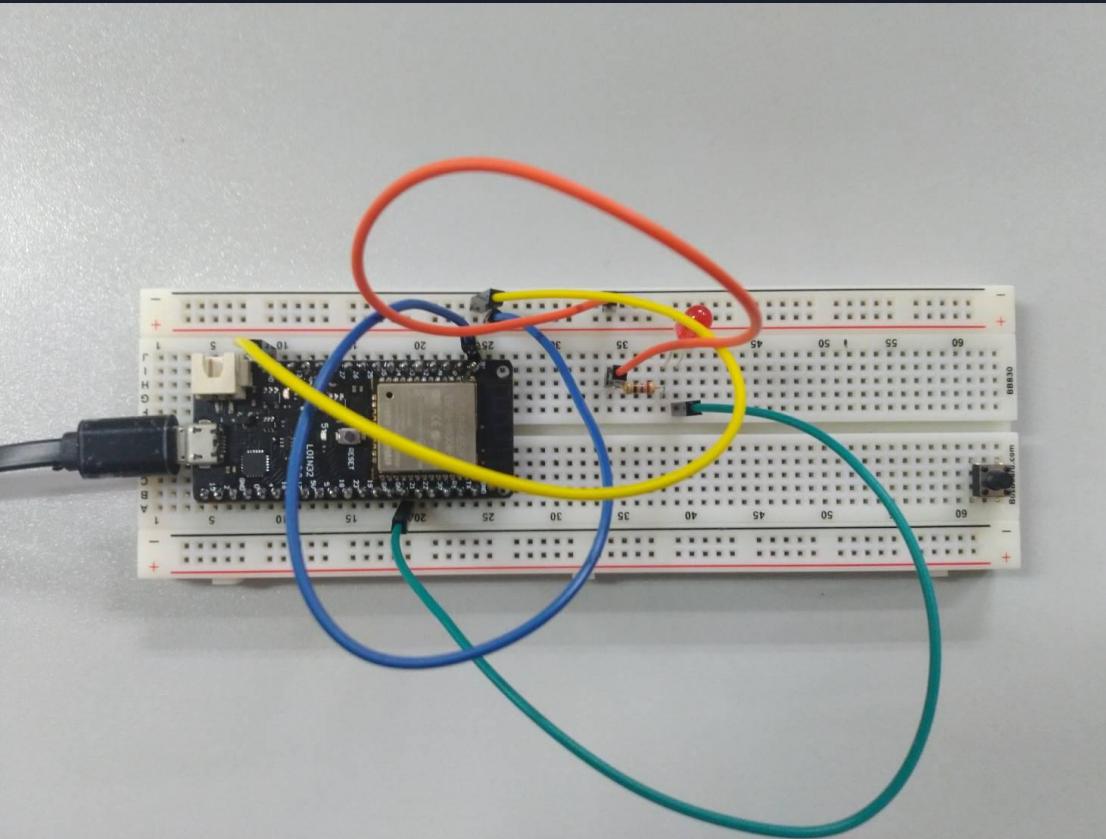


ทดลองใช้โปรแกรม Fritzing ออกแบบบอร์ด LED

4. เริ่มทดลองออกแบบ



ลองต่ออุปกรณ์จริง ตามที่ได้ออกแบบในโปรแกรม Fritzing





เริ่ม run คำสั่งในบอร์ด ESP32 / ESP8266

Diagram Board: ESP32 - Lolin32

ESP32 WeMos LOLIN32 PINOUT

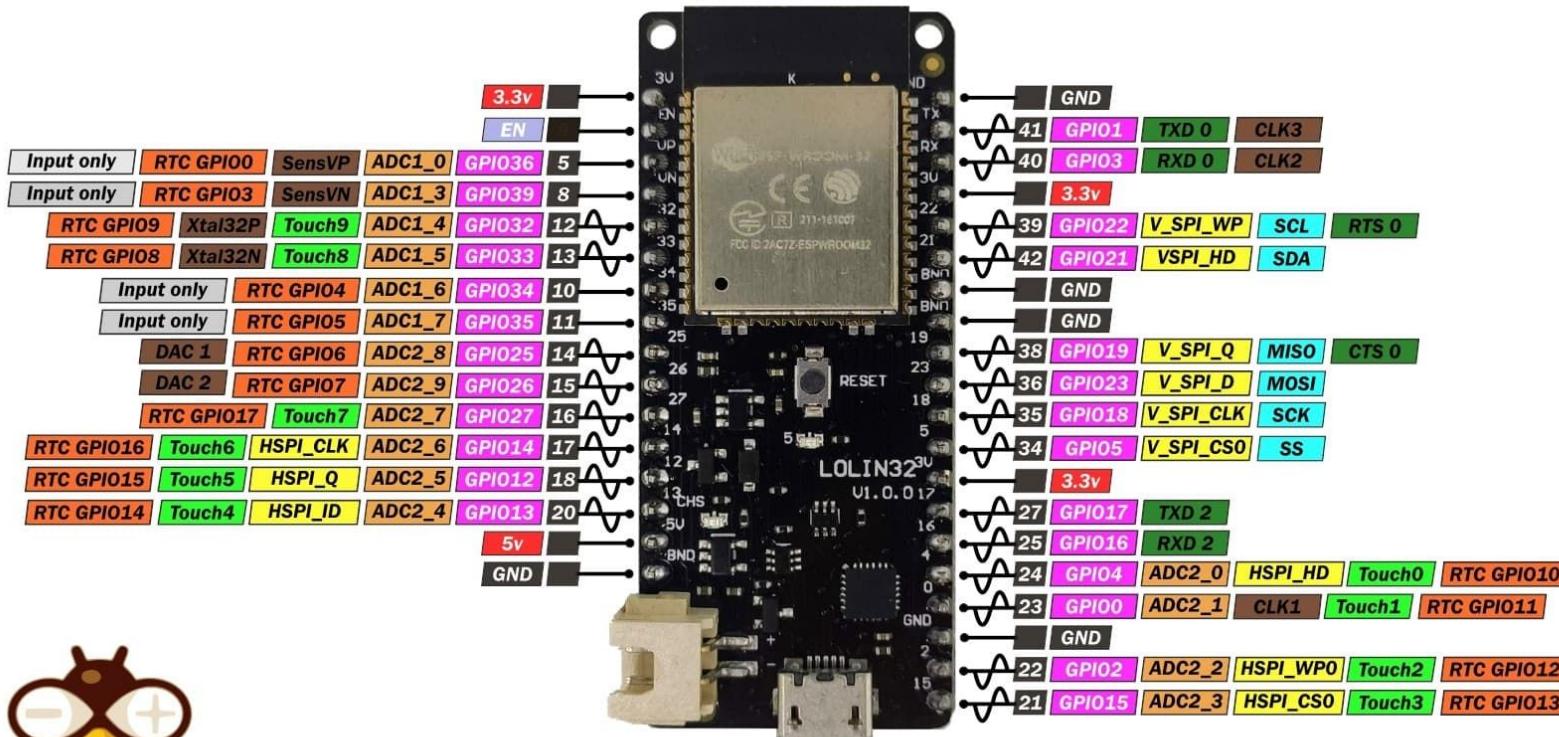
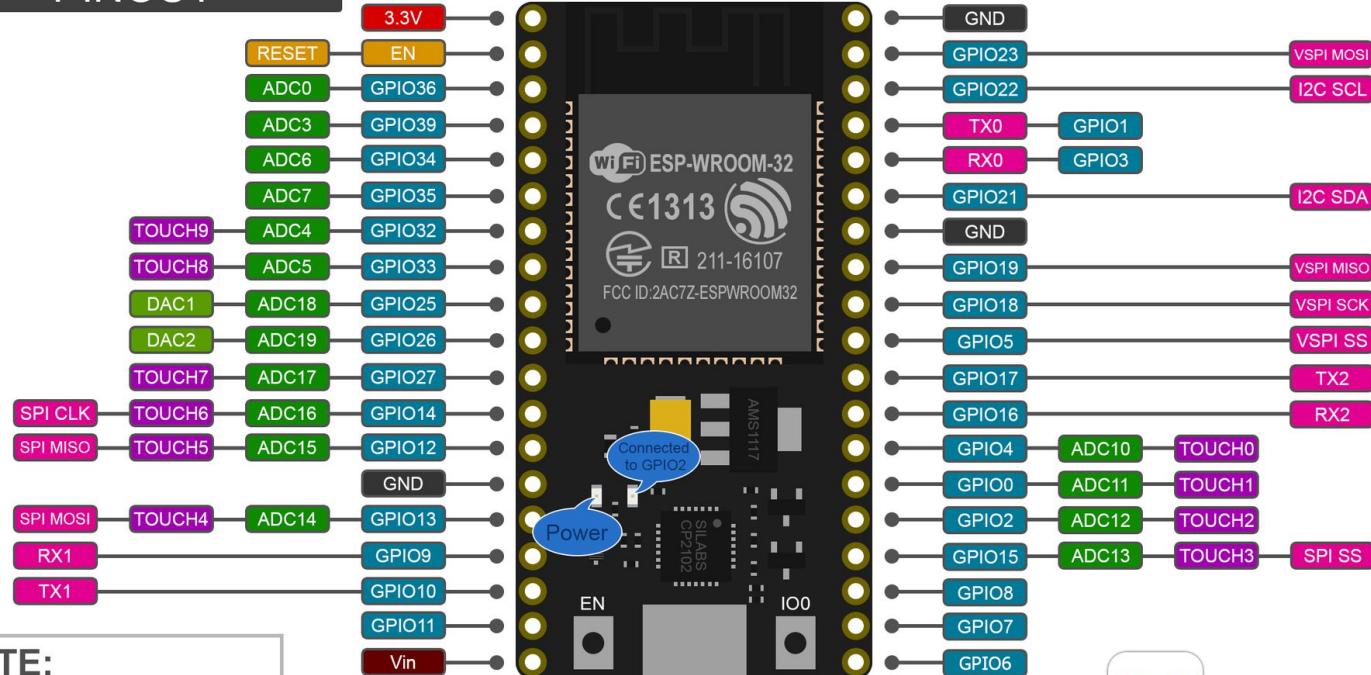


Diagram Board: ESP32-S / ESP32

NodeMCU-32S

PINOUT



NOTE:

All pin supported PWM and I2C

Pin current 6mA (Max. 12mA)

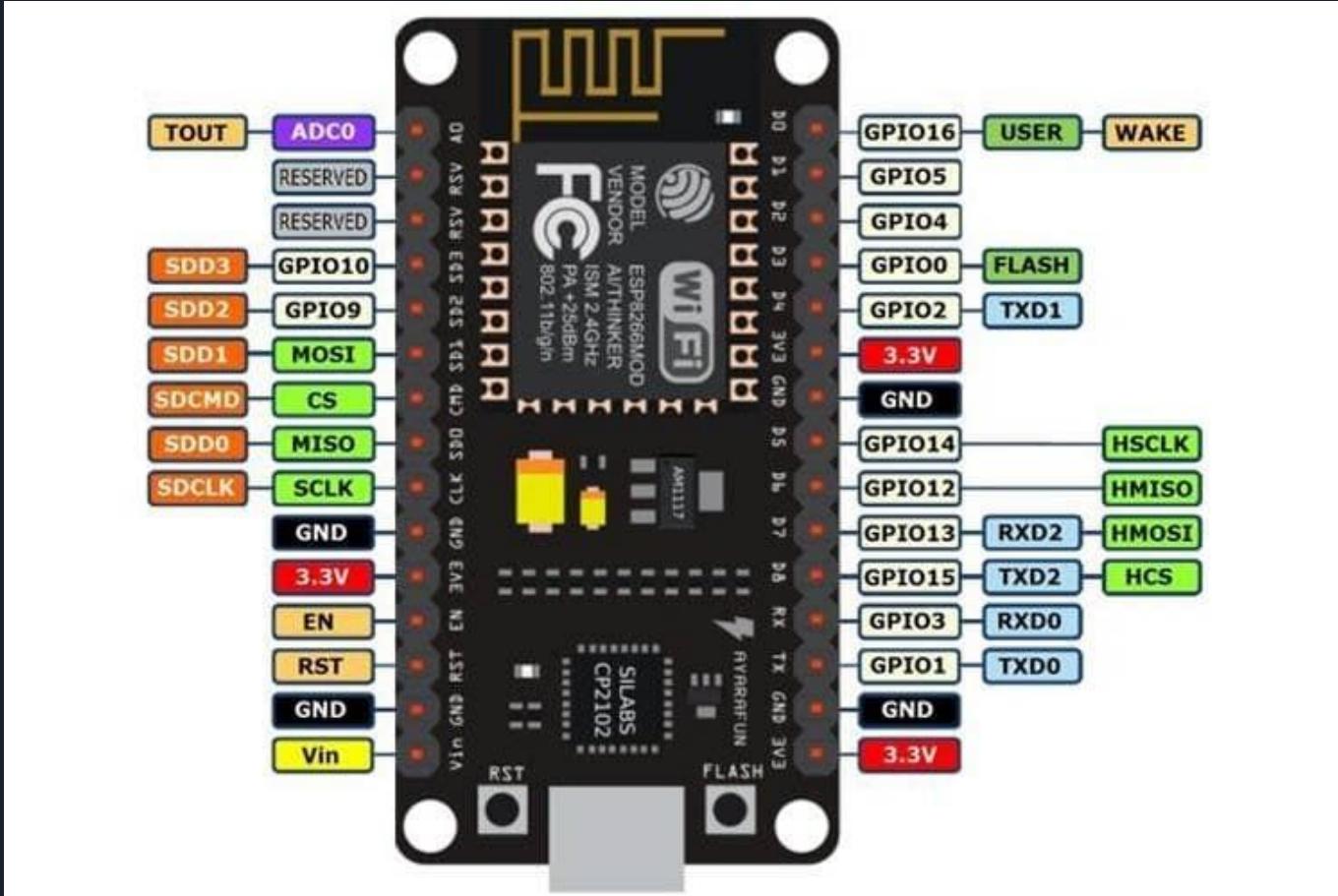


CC BY 4.0

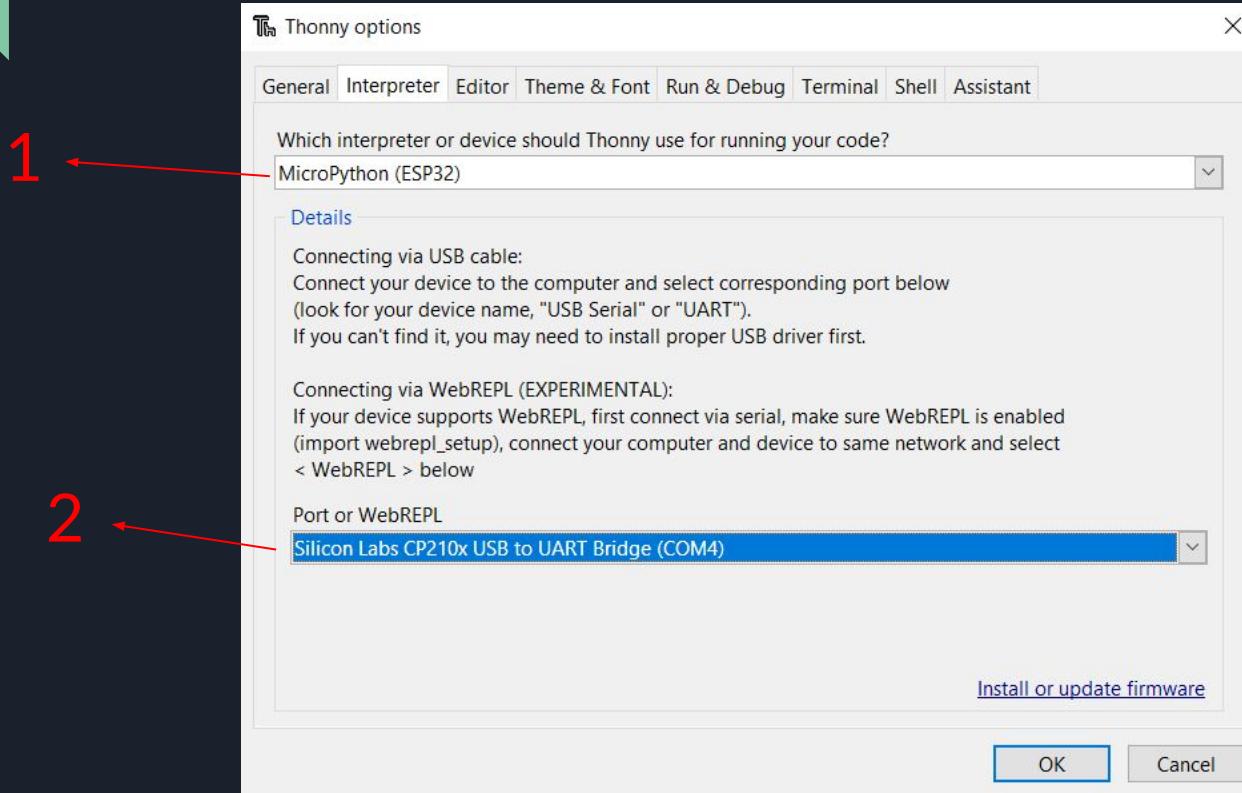


www.ioxhop.com

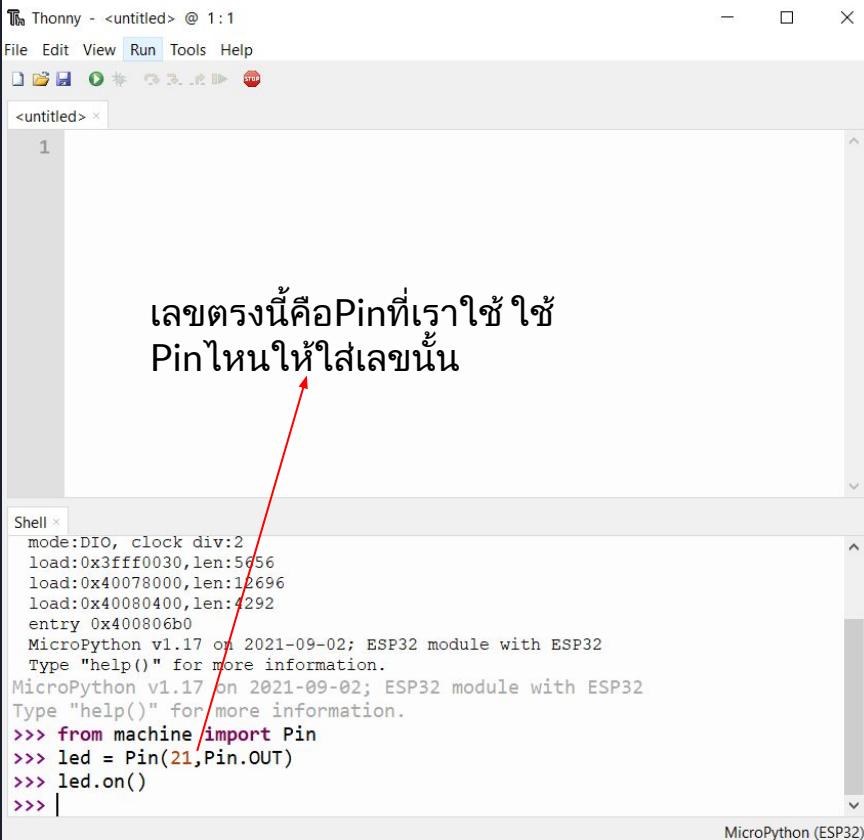
Diagram Board: ESP12-F



เปิดโปรแกรม Thonny ไปที่ Tools > Options > Interpreter ให้เลือก ตรง ช่องที่ 1 เป็น MicroPython(ชื่อบอร์ดที่เราใช้) ช่องที่ 2 เป็น Port ของบอร์ดที่เราเสียบ



ทดลองพิมพ์คำสั่งเพื่อให้ LED ทำงานตามภาพ พิมพ์ไปทีละบรรทัด



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

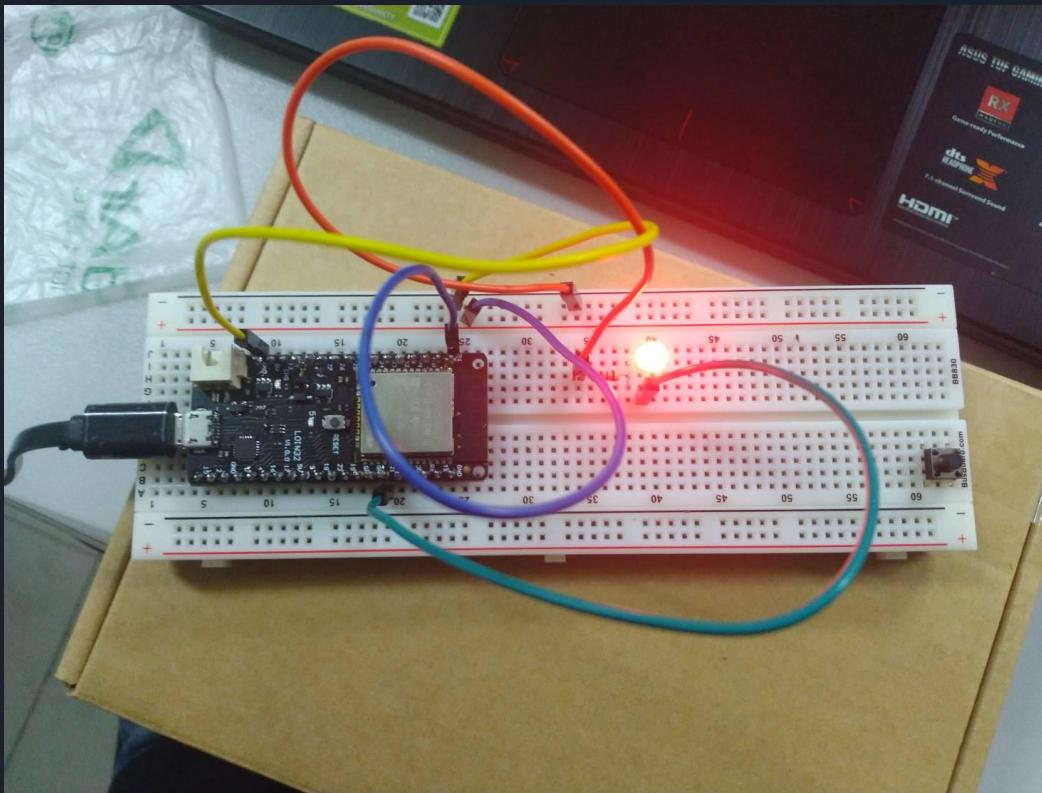
- Title Bar:** Thonny - <untitled> @ 1:1
- Menu Bar:** File Edit View Run Tools Help
- Toolbar:** Includes icons for Open, Save, Run, Stop, and others.
- Code Editor:** A single tab labeled <untitled> containing the number "1".
- Shell:** A terminal window showing the MicroPython environment and the execution of the following code.
- Code in Shell:**

```
mode:DIO, clock div:2
load:0xffff0030,len:5656
load:0x40078000,len:12696
load:0x40080400,len:4292
entry 0x400806b0
MicroPython v1.17 on 2021-09-02; ESP32 module with ESP32
Type "help()" for more information.
MicroPython v1.17 on 2021-09-02; ESP32 module with ESP32
Type "help()" for more information.
>>> from machine import Pin
>>> led = Pin(21,Pin.OUT)
>>> led.on()
>>> |
```
- Status Bar:** MicroPython (ESP32)

A red arrow points from the text "Pin 21" in the explanatory text below to the "21" in the line `led = Pin(21,Pin.OUT)`.

เลขตรงนี้คือPinที่เราใช้ ใช้ Pinไหนให้ใส่เลขนั้น

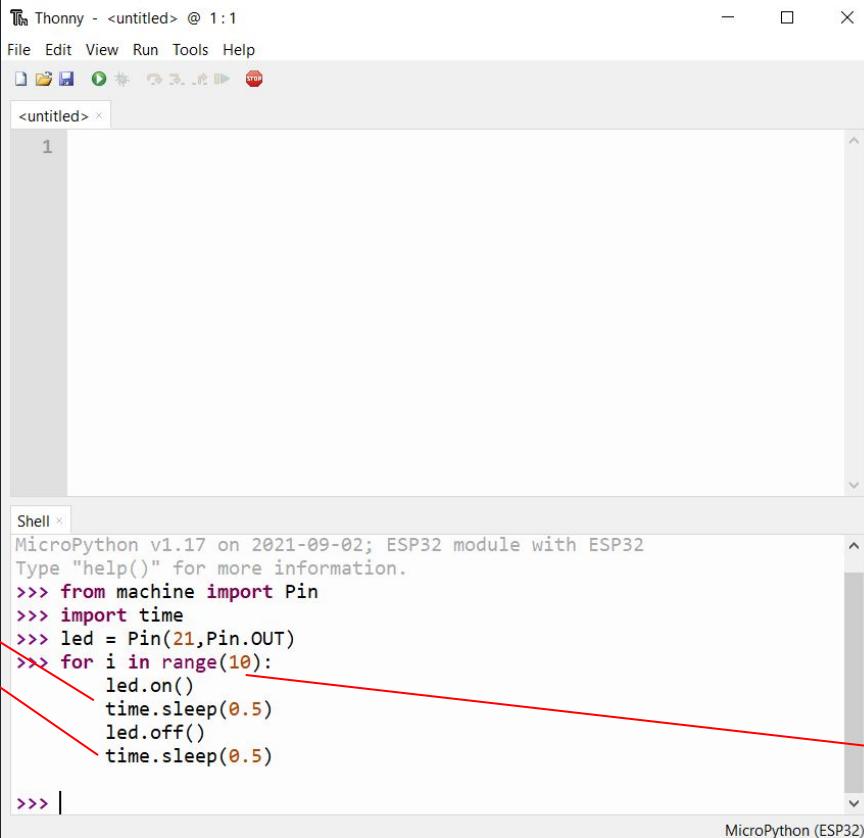
ผลลัพธ์



ปิดโดยการพิมพ์คำสั่ง

`led.off()`

ทดลองทำไฟกระพริบโดยคำสั่งต่อไปนี้



The screenshot shows the Thonny IDE interface with a dark theme. At the top is a menu bar with File, Edit, View, Run, Tools, and Help. Below the menu is a toolbar with icons for file operations like Open, Save, Run, and Stop. The main window has a tab labeled <untitled>. In the code editor area, there is a single digit '1' on the left margin. The code in the editor is:

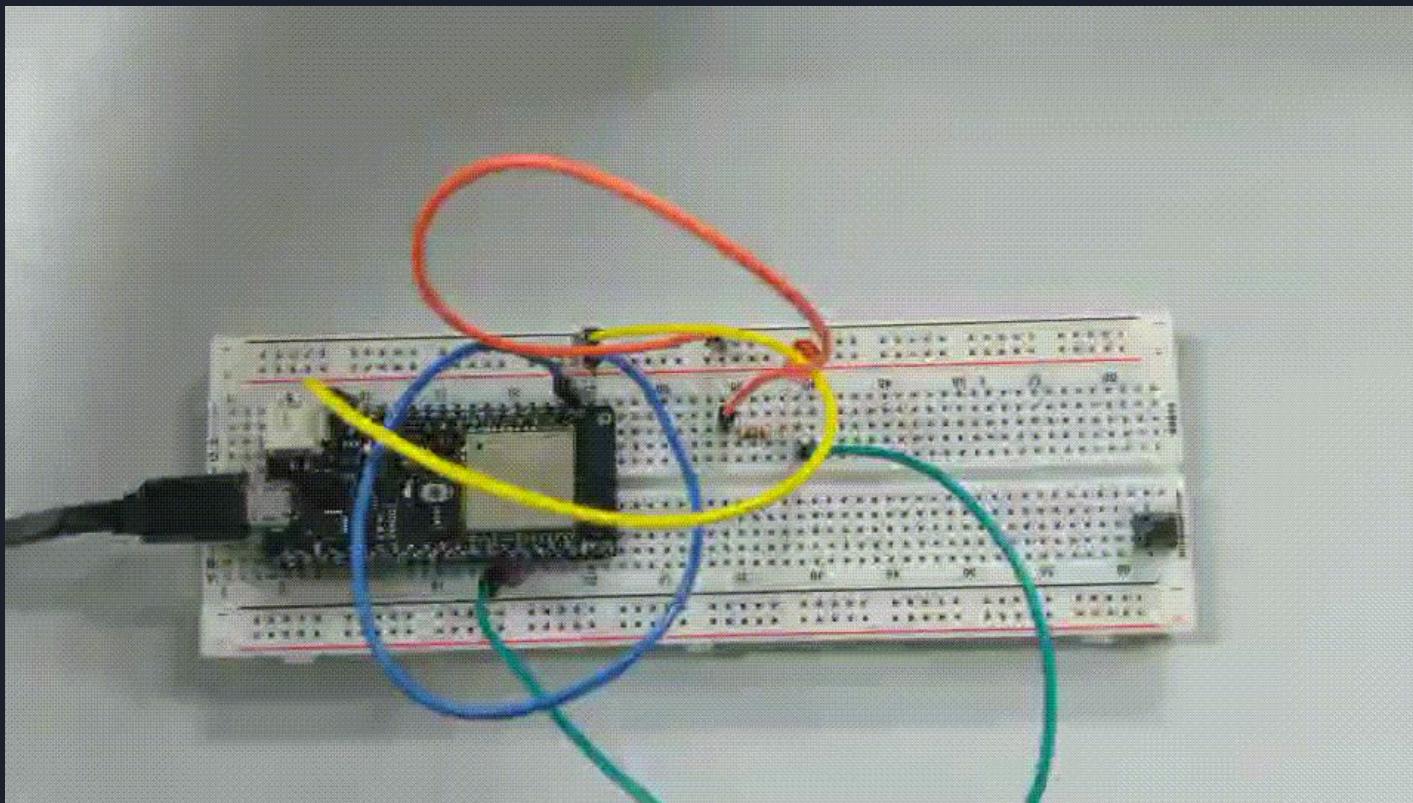
```
>>> from machine import Pin  
>>> import time  
>>> led = Pin(21,Pin.OUT)  
>>> for i in range(10):  
    led.on()  
    time.sleep(0.5)  
    led.off()  
    time.sleep(0.5)
```

Below the code editor is a shell window titled "Shell x". It displays the MicroPython version and build information, followed by the executed code. The shell window has a scroll bar on the right. At the bottom of the shell window, it says "MicroPython (ESP32)".

ระยะเวลา
รอที่รัน
คำสั่งถัดไป

จำนวนครั้งในการ
ทำการสั่ง

ผลลัพธ์

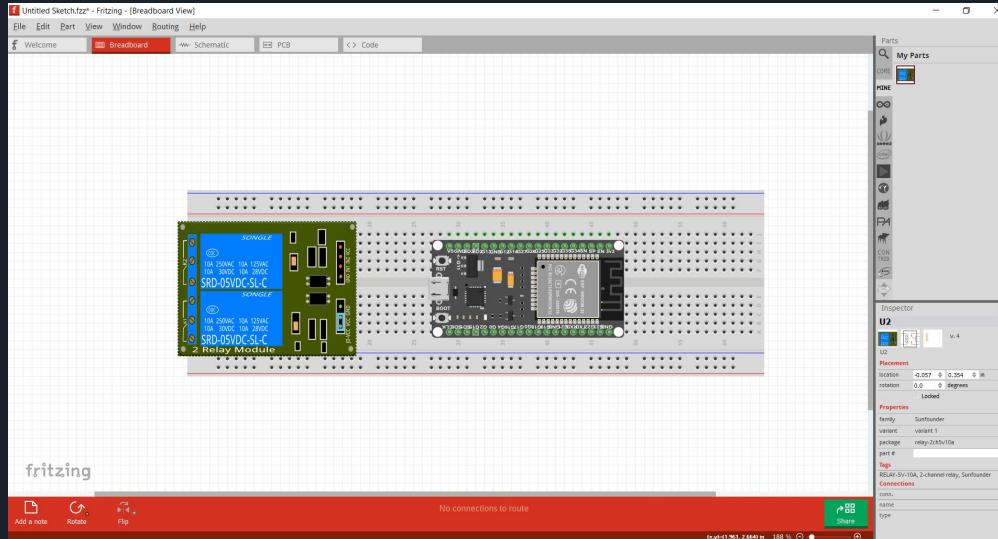


ทดลองตัว Relay

1. โหลดโมเดลใน Fritzing ของตัว relay ได้ที่ลิ้งนี้

<https://timgolisch.wordpress.com/2015/02/22/fritzing-2-channel-relay-part/>

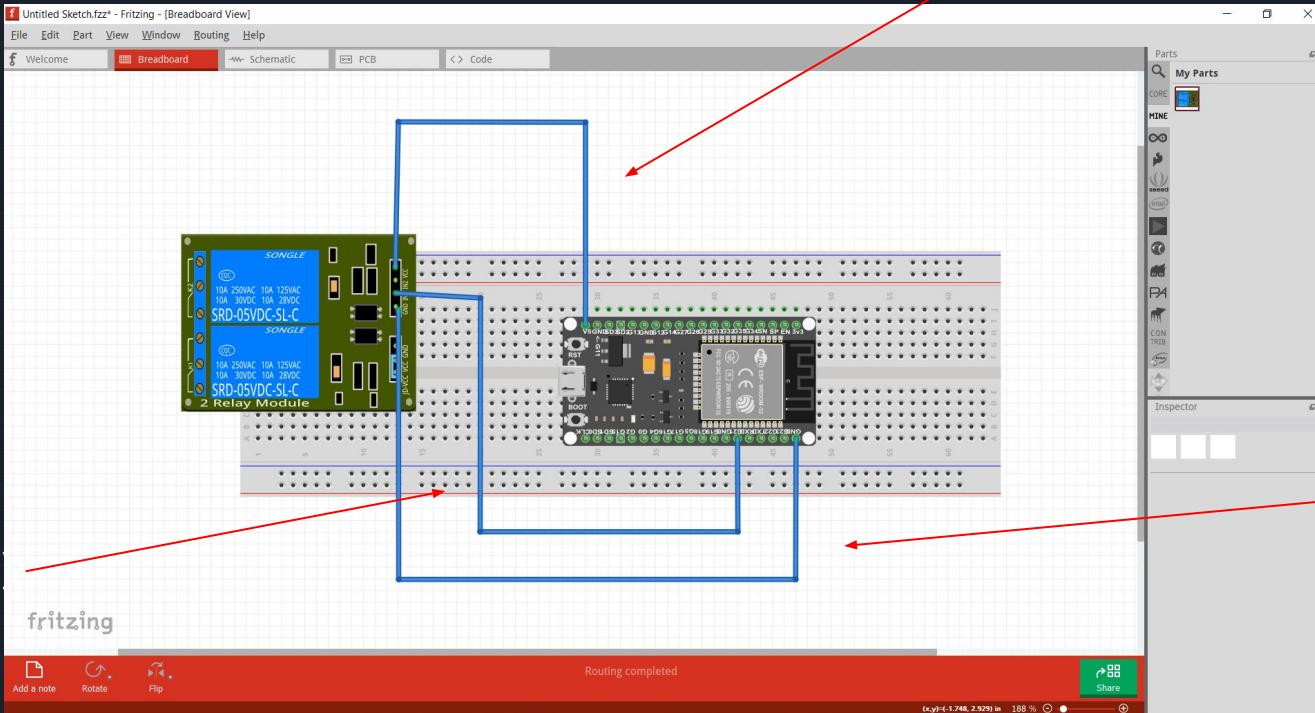
2. ดาวน์โหลดเสร็จแล้วให้ลากมาไว้ใน Fritzing แล้ว ดึงมาใช้งาน



ทดลองตัว Relay

3.ทดลองออกแบบวงจรของ relay

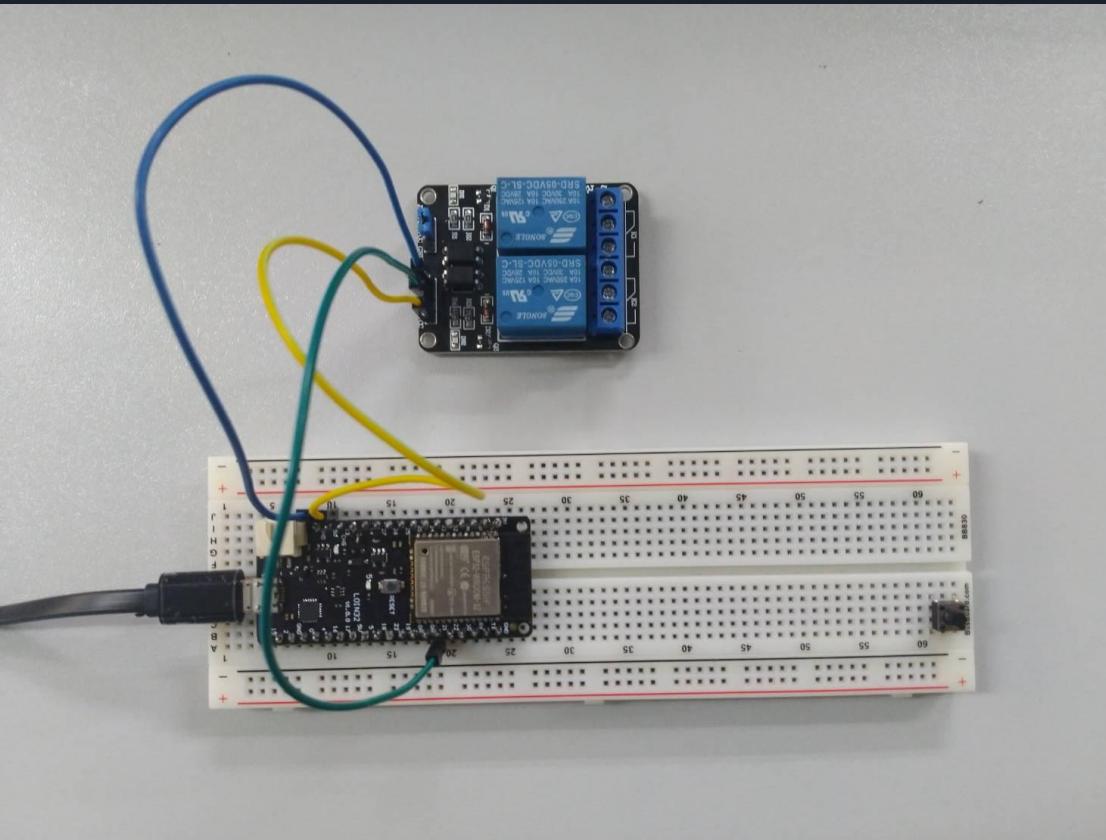
VCC ต่อ กับ 5V



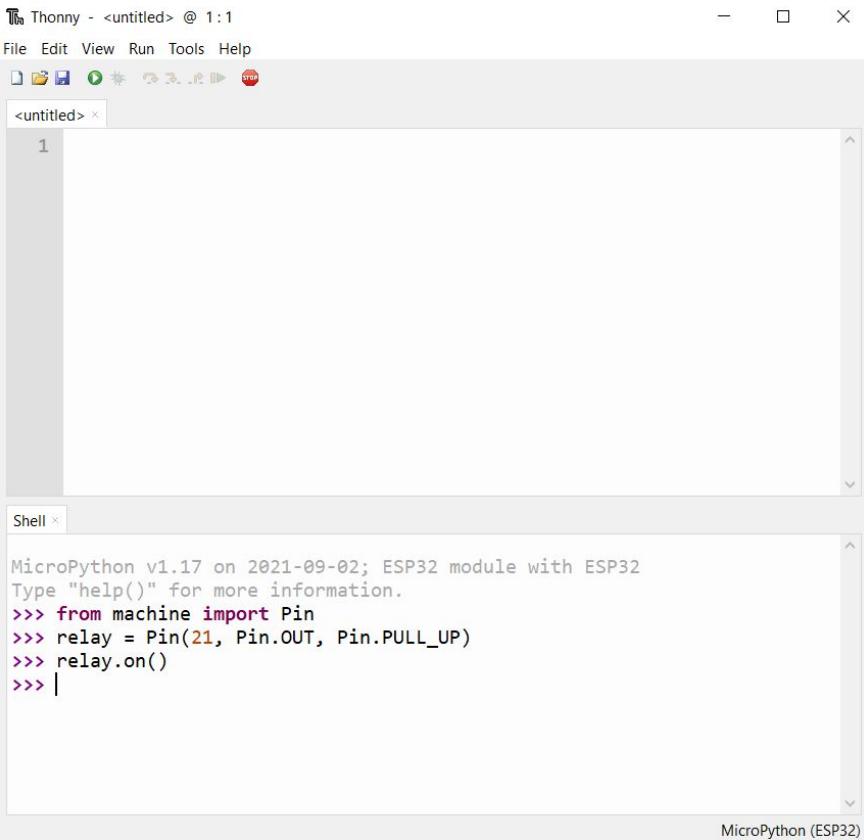
GND ต่อ GND

IN1 ต่อ Pin(21)

ลองต่ออุปกรณ์ด้วยสาย ตามที่ได้ออกแบบในโปรแกรม Fritzing



ทดลองพิมพ์คำสั่งเพื่อให้ LED ทำงานตามภาพ พิมพ์ไปทีละบรรทัด



The screenshot shows the Thonny IDE interface with two main panes: an editor and a shell.

Editor: The title bar says "Thonny - <untitled> @ 1:1". The menu bar includes File, Edit, View, Run, Tools, and Help. Below the menu is a toolbar with icons for file operations and a stop button. The code editor window has tabs for "<untitled>" and "Shell". The code in the editor is:

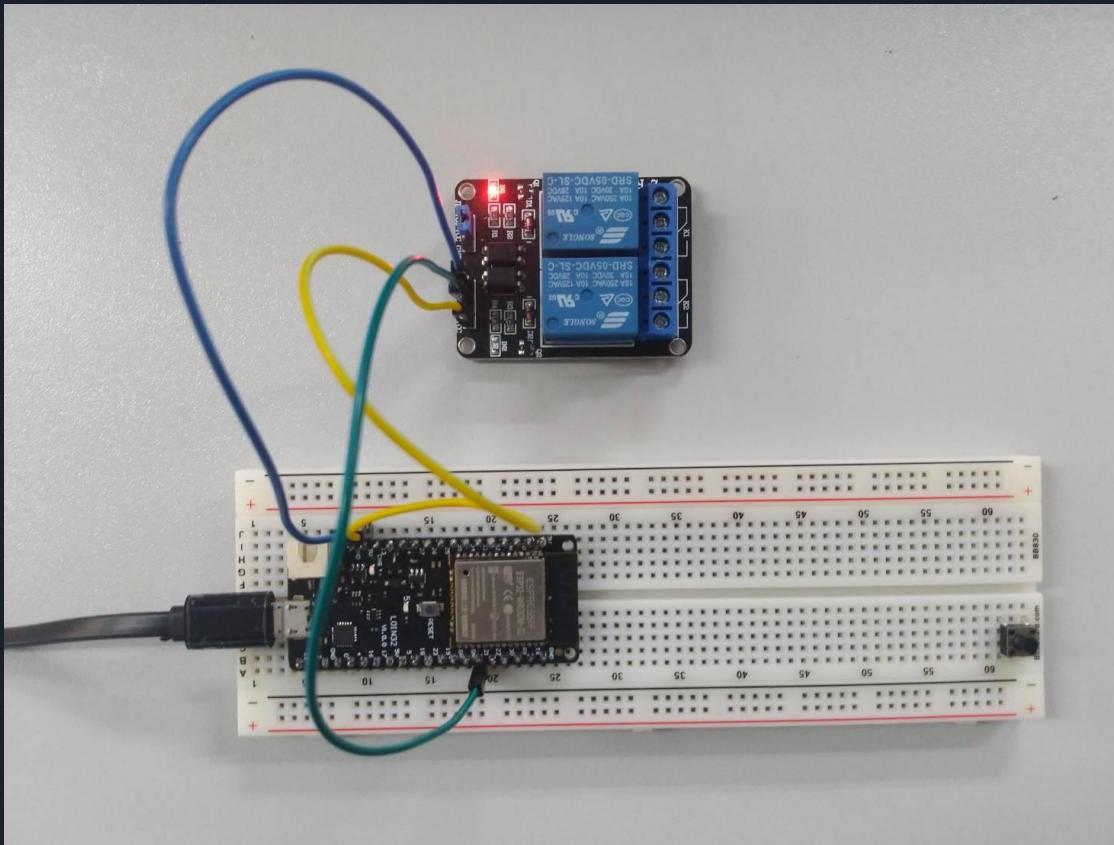
```
<untitled>
1
```

Shell: The title bar says "Shell x". It displays the MicroPython environment information and the entered code:

```
MicroPython v1.17 on 2021-09-02; ESP32 module with ESP32
Type "help()" for more information.
>>> from machine import Pin
>>> relay = Pin(21, Pin.OUT, Pin.PULL_UP)
>>> relay.on()
>>> |
```

The status bar at the bottom right of the shell window says "MicroPython (ESP32)".

ผลลัพธ์ ไฟติดแสดงว่า Relay ใช้งานได้

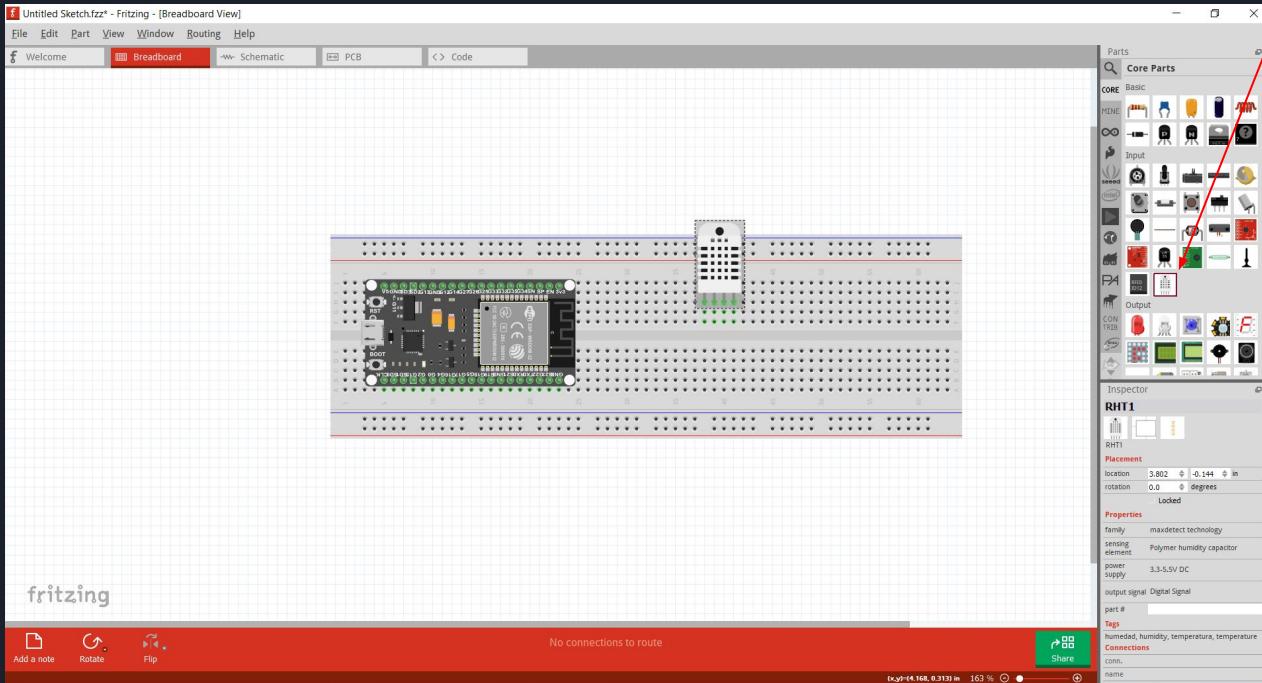


ปิดโดยการพิมพ์คำสั่ง
`relay.off()`

ทดลองตัว DHT22

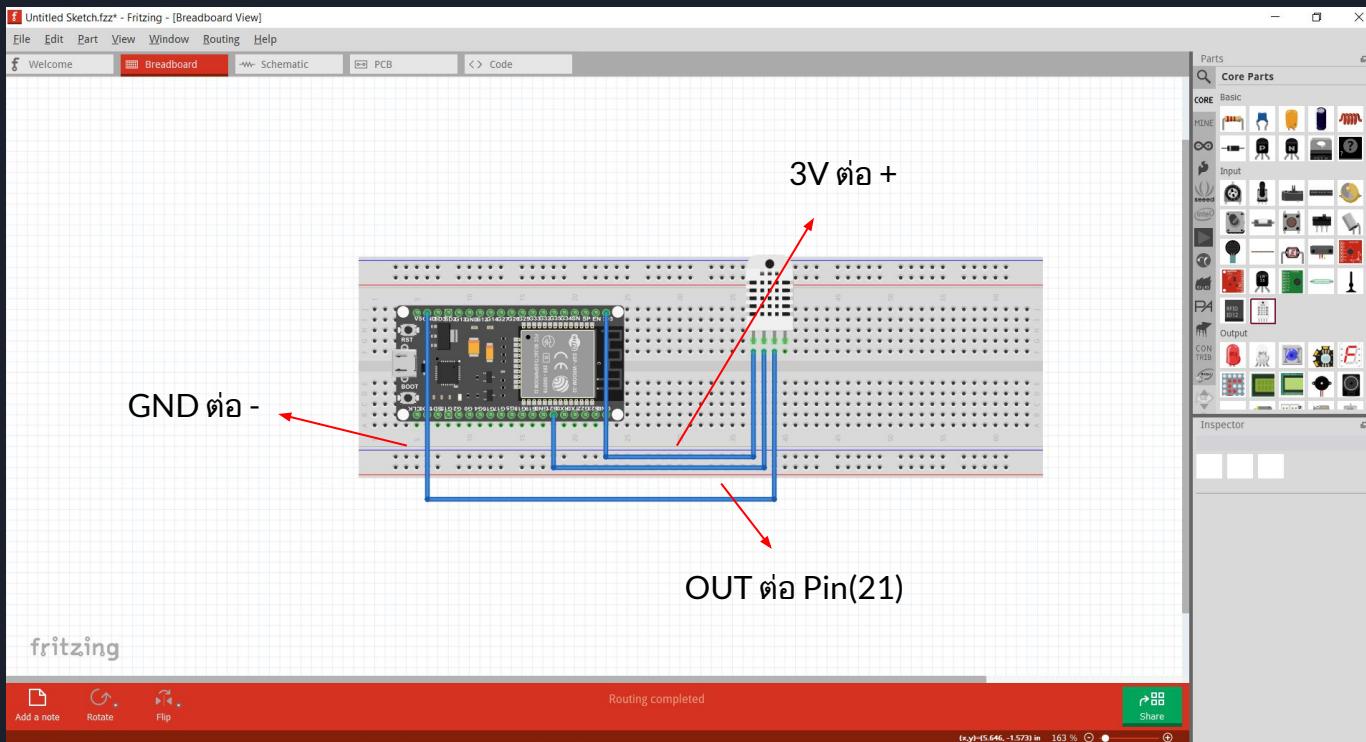
DHT22

1. ใน Fritzing จะไม่มี dht22 3ขาแบบที่เราใช้นั้นเราจะใช้ตัว 4 ขา ที่โปรแกรมมีลอง
ออกแบบวงจร ให้ไปที่คำว่า core ในแทน basic อันสุดท้าย ลากลงมาไว้ที่บอร์ด

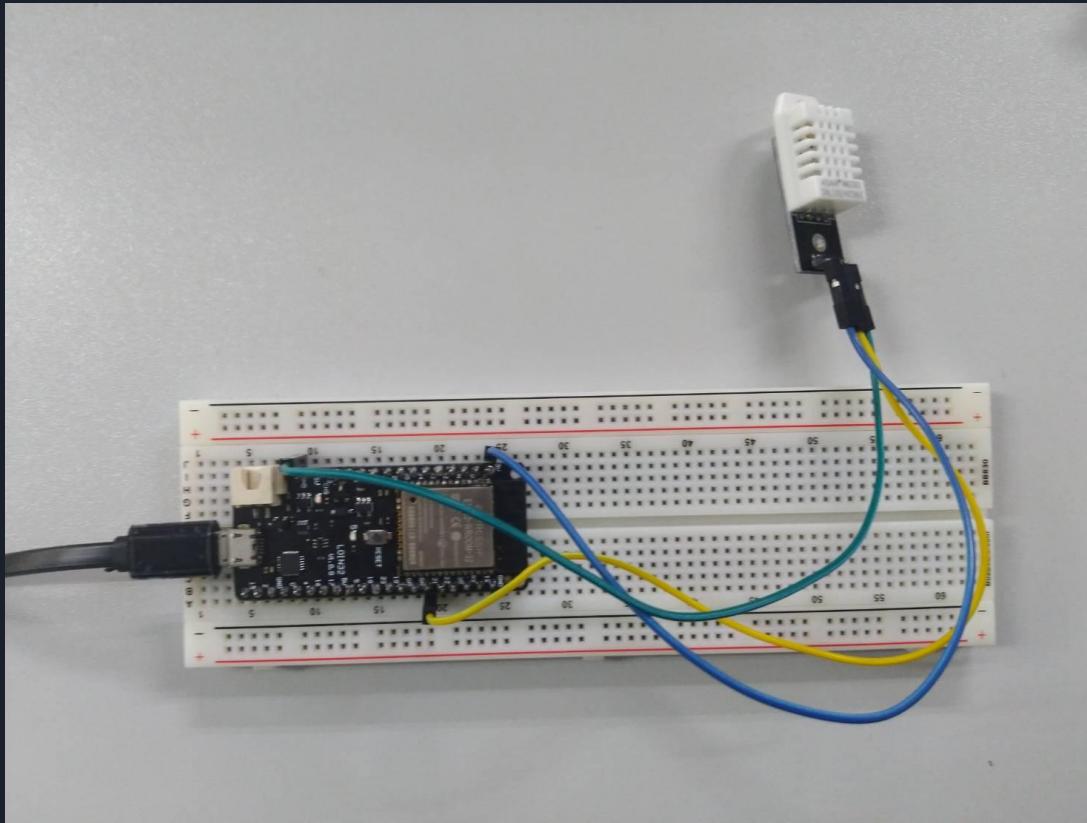


ทดลองตัว Relay

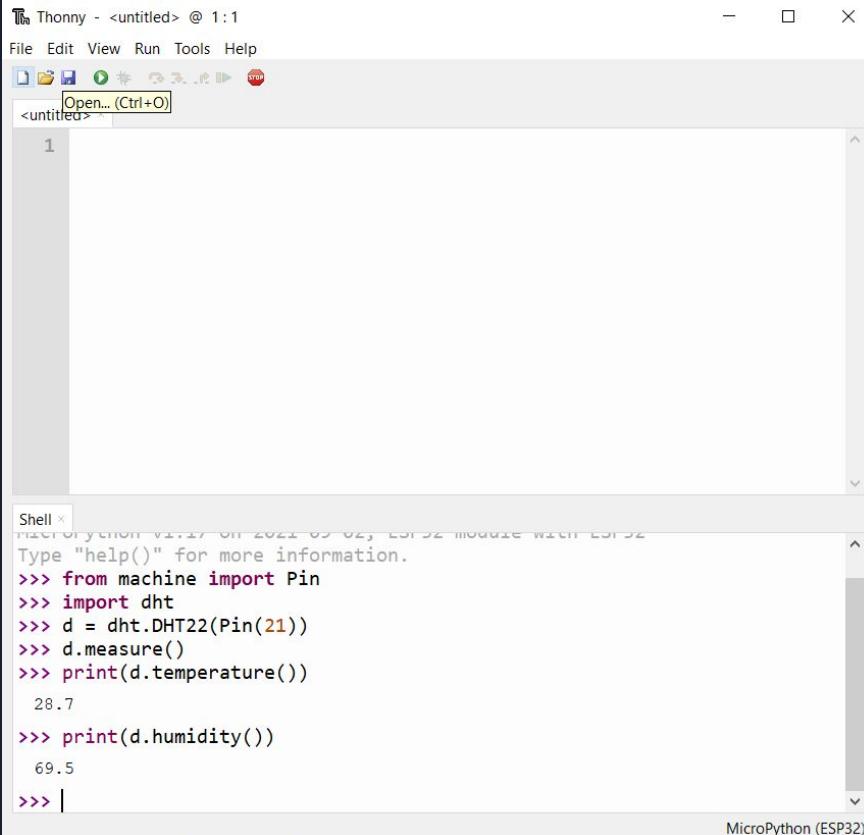
3.ทดลองออกแบบวงจรของ DHT22 (ในที่นี้เรายังใช้ 3 ขาแต่ในโปรแกรมมี 4 ขาใช้แทนกัน ก่อน)



ลองต่ออุปกรณ์จริง ตามที่ได้ออกแบบในโปรแกรม Fritzing



ทดลองพิมพ์คำสั่งเพื่อให้ DHT22 ทำงานโดยการลองเช็ค อุณหภูมิและความชื้น ถ้า อุณหภูมิและความชื้นแสดงค่าได้แบบนี้แสดงว่าใช้งานได้



The screenshot shows the Thonny IDE interface with a code editor and a shell window.

Code Editor:

```
<untitled>
1
```

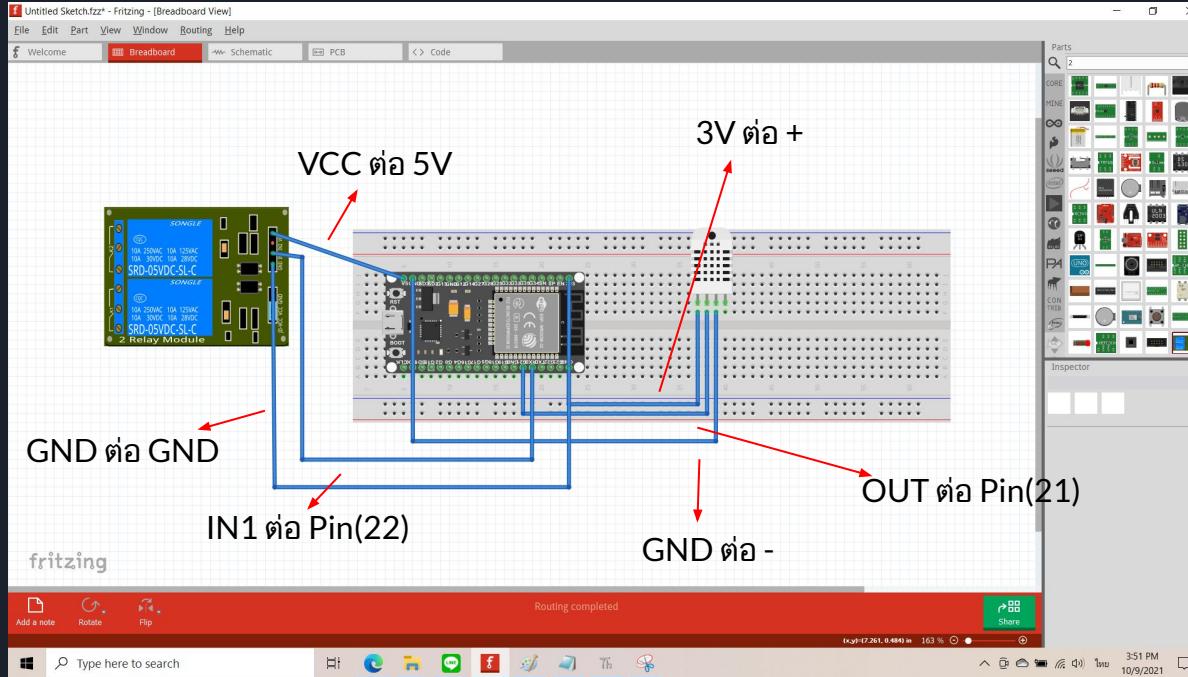
Shell:

```
MicroPython v1.17 on 2021-05-02, ESP32 module with ESP32
Type "help()" for more information.
>>> from machine import Pin
>>> import dht
>>> d = dht.DHT22(Pin(21))
>>> d.measure()
>>> print(d.temperature())
28.7
>>> print(d.humidity())
69.5
>>> |
```

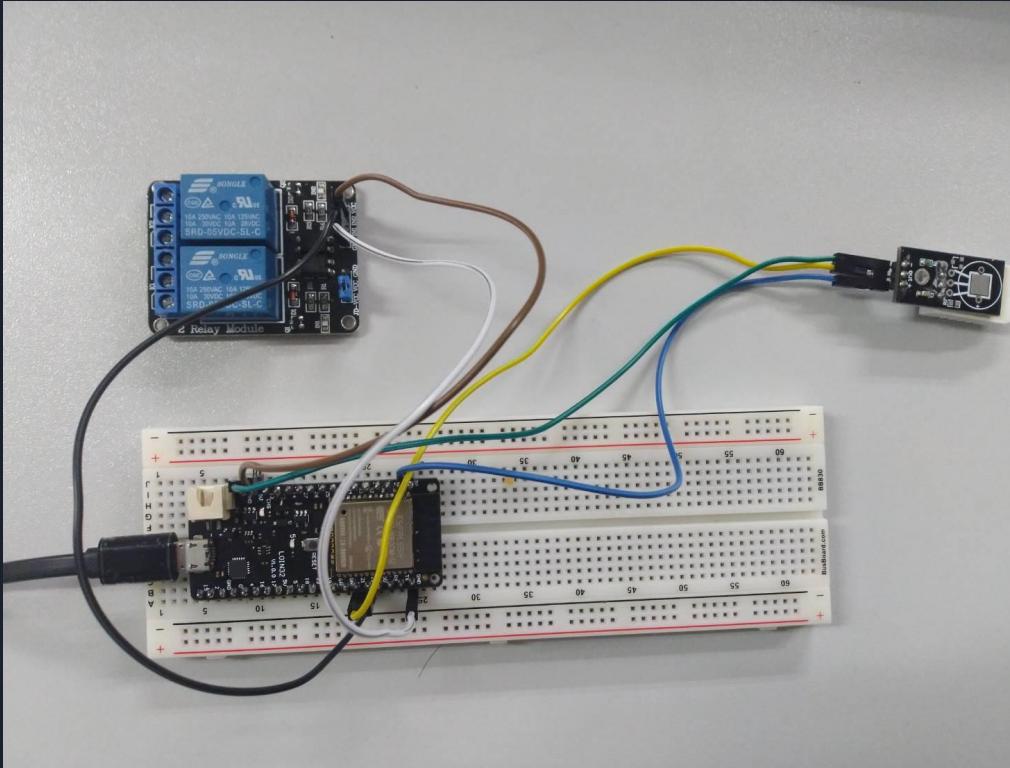
The shell output shows the temperature as 28.7 and the humidity as 69.5.

ทดลองตัว Relay+DHT22

ทดลองใช้ Relay+DHT22 โจทย์คือเมื่ออุณหภูมิลดลงถึงระดับนึงให้ Relay off และเมื่ออุณหภูมิกลับขึ้นมาให้ Relay on ลองออกแบบวงจรใน Fritzing

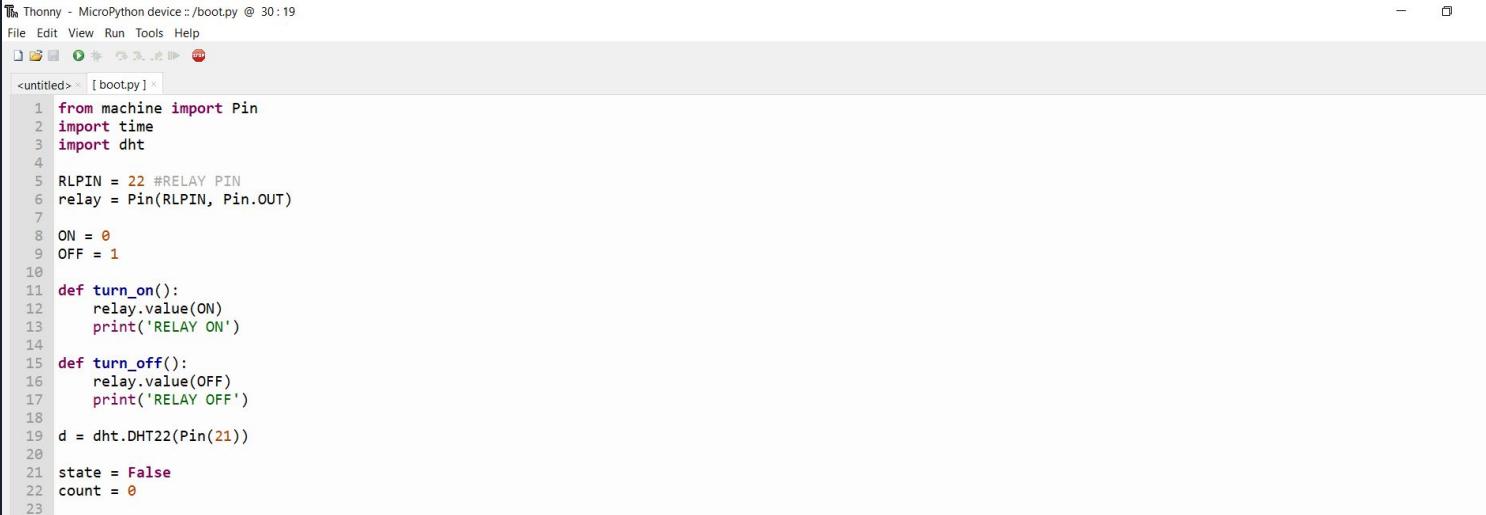


ลองต่ออุปกรณ์จริง ตามที่ได้ออกแบบในโปรแกรม Fritzing



โค้ดสำหรับทำโจทย์

ขั้นตอนแรกให้เปิด Thonny กด Ctrl+O เลือก MicroPython Device จะเจอไฟล์เดอร์ boot.py ให้เปิด แล้วใส่โค้ดตามด้านภาพต่อไปนี้



```
Thonny - MicroPython device :: /boot.py @ 30 : 19
File Edit View Run Tools Help
<untitled> [ boot.py ] <-->
1 from machine import Pin
2 import time
3 import dht
4
5 RLPIN = 22 #RELAY PIN
6 relay = Pin(RLPIN, Pin.OUT)
7
8 ON = 0
9 OFF = 1
10
11 def turn_on():
12     relay.value(ON)
13     print('RELAY ON')
14
15 def turn_off():
16     relay.value(OFF)
17     print('RELAY OFF')
18
19 d = dht.DHT22(Pin(21))
20
21 state = False
22 count = 0
23
```

ต่อหน้าถัดไป



โค้ดสำหรับทำโจทย์

```
24 while True:  
25     d.measure()  
26     temp = d.temperature()  
27     humid = d.humidity()  
28     print(count)  
29     print('TEMP: {} HUMID : {}'.format(temp,humid))  
30     if temp < 26.5:  
31         state = True  
32         #print('TURN ON')  
33     else:  
34         state = False  
35         #print('TURN OFF')  
36  
37     if state == True:  
38         turn_on()  
39     else:  
40         turn_off()  
41  
42     print('-----')  
43     count += 1  
44     time.sleep(3)
```

Save และกดปุ่ม Reset ที่บอร์ดหนึ่งครั้ง จาก Code ถ้าอุณหภูมิต่ำกว่า 26.5 Relay จะ On ถ้ามากกว่า Relay จะ Off รอดูผลลัพธ์

```
Shell x
-----
7
TEMP: 26.7 HUMID : 74.0
RELAY OFF
-----
8
TEMP: 26.6 HUMID : 74.0
RELAY OFF
-----
9
TEMP: 26.5 HUMID : 74.0
RELAY OFF
10
TEMP: 26.4 HUMID : 74.20001
RELAY ON
-----
11
TEMP: 26.4 HUMID : 74.4
RELAY ON
-----
```

Relay off เมื่ออุณหภูมิมากกว่า 26.5

Relay on เมื่ออุณหภูมิต่ำกว่า 26.5

วิธีเชื่อมต่อ WIFI ให้ใส่คำลั่งตามนี้ ทีละบรรทัด

```
import network
```

```
sta_if = network.WLAN(network.STA_IF)
```

```
sta_if.active(True)
```

```
sta_if.scan()
```

```
sta_if.connect("ชื่อWIFI", "รหัสWIFI")
```

```
sta_if.isconnected()
```