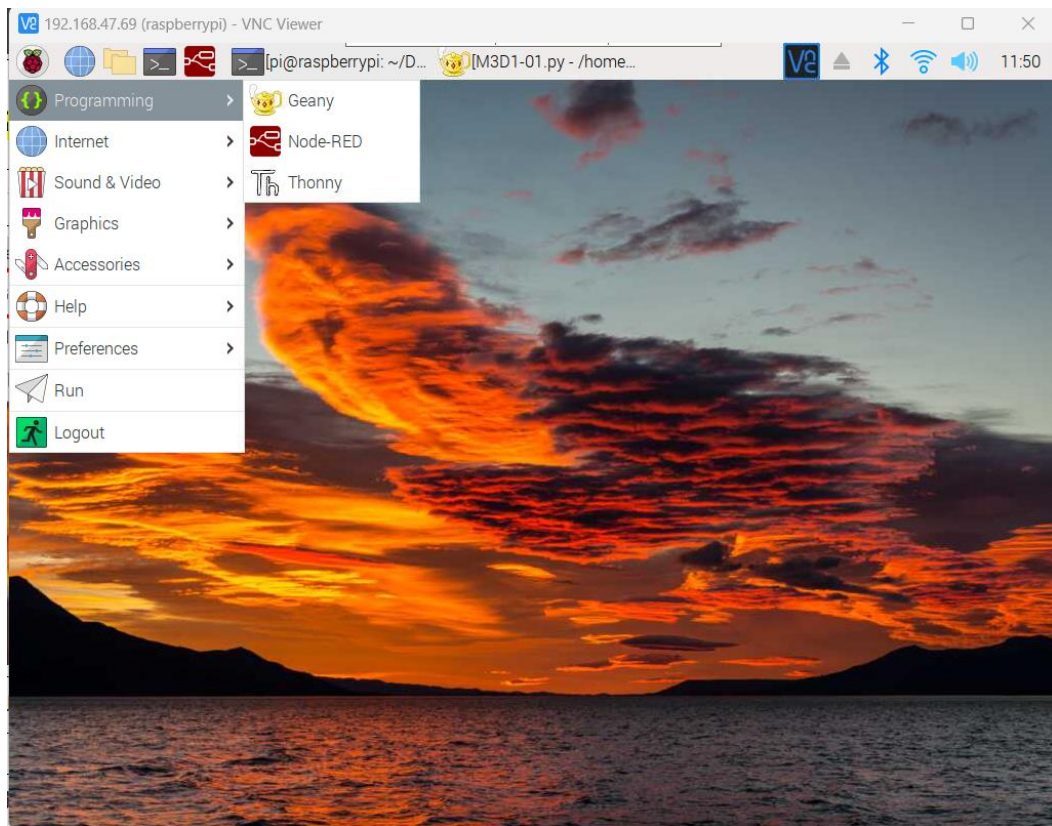


<p>การสร้าง MQTT Server บน Raspberry Pi เพื่อใช้งาน Chatbot LINE ในฟาร์มอัจฉริยะ</p> <p>Chatbot LINE from Raspberry Pi MQTT Server for Smart Farming</p>	
ชื่อ-สกุล : นางสาวขวัญจิรา พันธุ์เกตุ	รหัสนักศึกษา : B6321451
6/6 – คำถามท้ายบทเพื่อทดสอบความเข้าใจ	

Quiz_101 – ทดสอบ RPi4 GPIO with Python

Python.1 - Python Switch control LED >> กดติด ปล่อยดับ

โปรแกรมที่ใช้ทดสอบ



ใช้โปรแกรม Geany

Code Capture

192.168.47.69 (raspberrypi) - VNC Viewer

[pi@raspberrypi: ~/D... M3D1-01.py - /home...

M3D1-01.py - /home/pi/Desktop - Geany

File Edit Search View Document Project Build Tools Help

Symbols

Variables

- LED_pin [3]
- SW_pin [4]

Imports

- GPIO [1]
- RPi [1]
- time [2]

```

1 import RPi.GPIO as GPIO
2 import time
3 LED_pin = 32
4 SW_pin = 36
5
6 GPIO.setmode(GPIO.BOARD)
7 GPIO.setup(LED_pin, GPIO.OUT)
8 GPIO.setup(SW_pin, GPIO.IN, pull_up_down = GPIO.PUD_UP)
9
10 while(True):
11     if(GPIO.input(SW_pin) == 0):
12         GPIO.output(LED_pin, GPIO.HIGH)
13         print("Input = 1, HIGH")
14     else:
15         GPIO.output(LED_pin, GPIO.LOW)
16         print("Input = 0, LOW")
17     time.sleep(0.5)
18

```

Status

11:43:22: This is Geany 1.37.1.
 11:43:22: File /home/pi/Desktop/M3D1-01.py opened (1).
 11:46:40: File /home/pi/Desktop/M3D1-01.py saved.

line: 1 / 18 col: 23 sel: 0 INS TAB mode: LF encoding: UTF-8 filetype: Python scope: unknown

import RPi.GPIO as GPIO

import time

LED_pin = 32

SW_pin = 36

GPIO.setmode(GPIO.BOARD)

GPIO.setup(LED_pin, GPIO.OUT)

GPIO.setup(SW_pin, GPIO.IN, pull_up_down = GPIO.PUD_UP)

```
while(True):
```

```
    if(GPIO.input(SW_pin) == 0):
```

```
        GPIO.output(LED_pin, GPIO.HIGH)
```

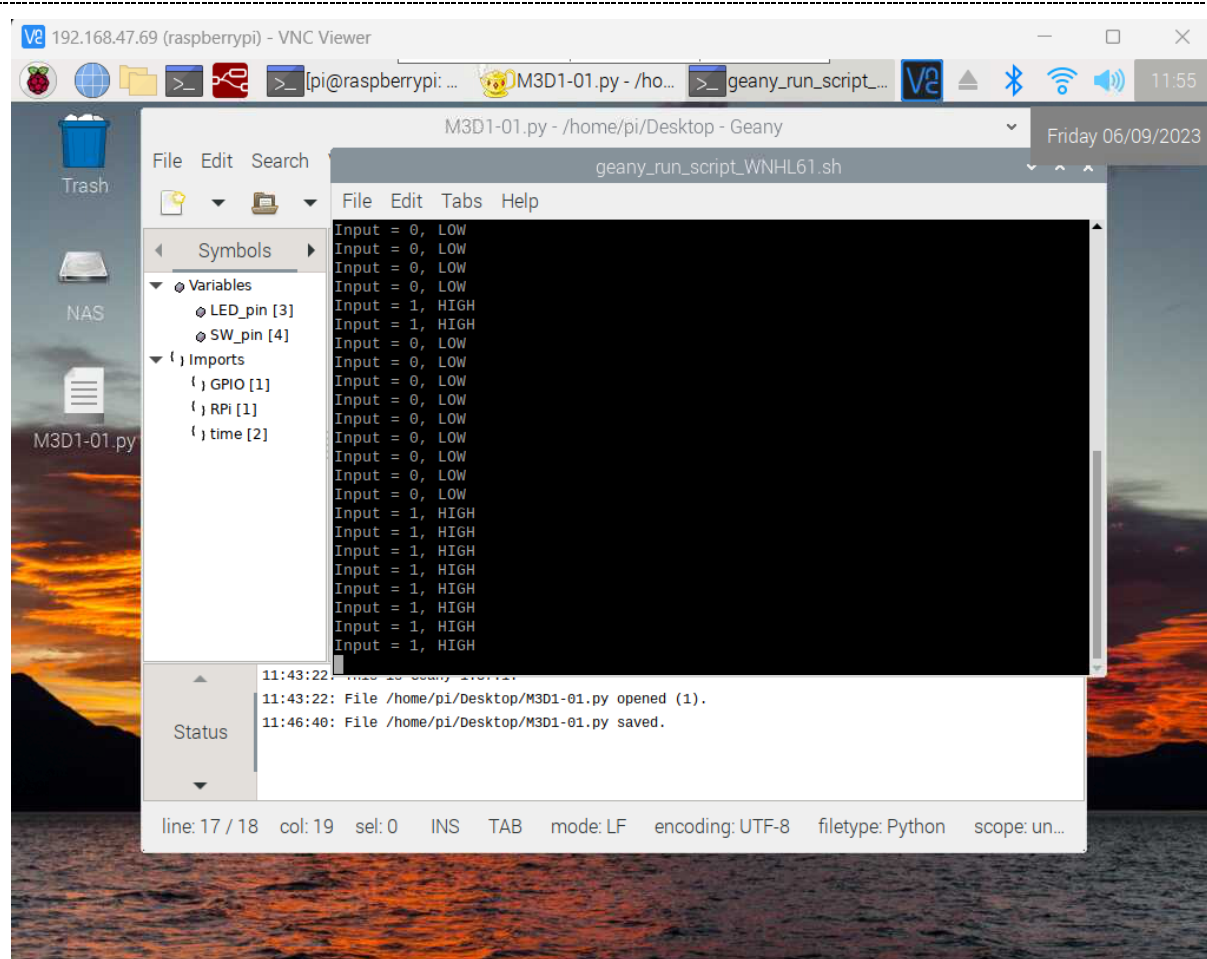
```
        print("Input = 1, HIGH")
```

```
    else:
```

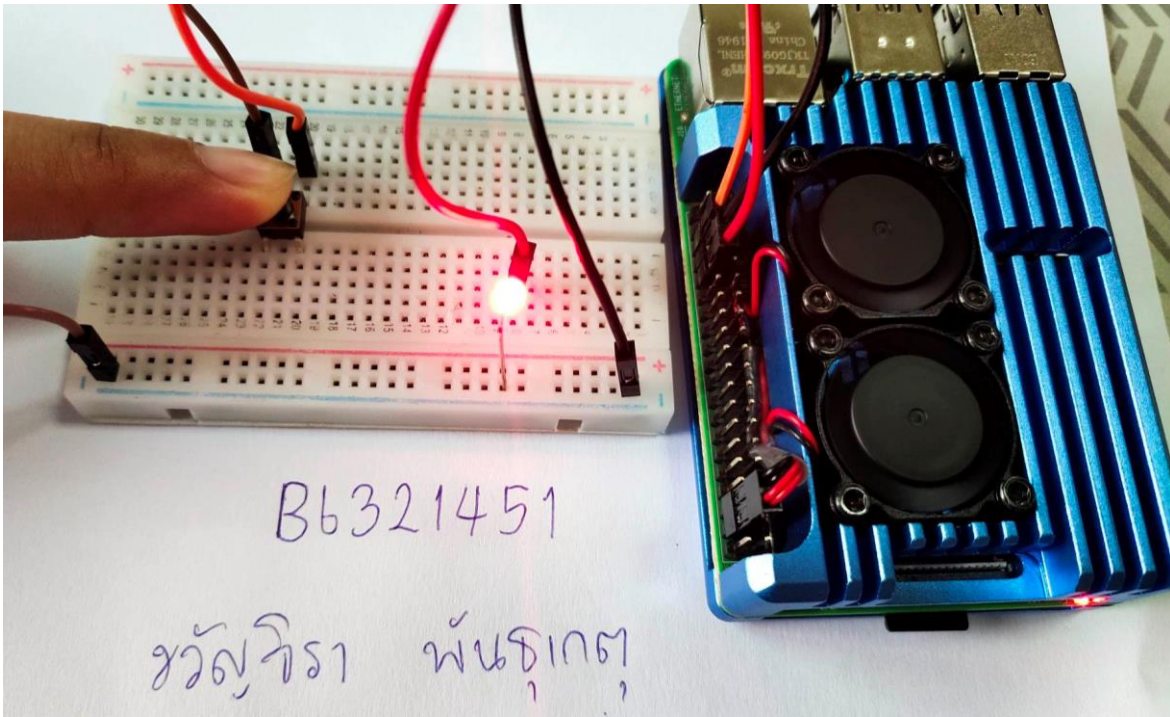
```
        GPIO.output(LED_pin, GPIO.LOW)
```

```
        print("Input = 0, LOW")
```

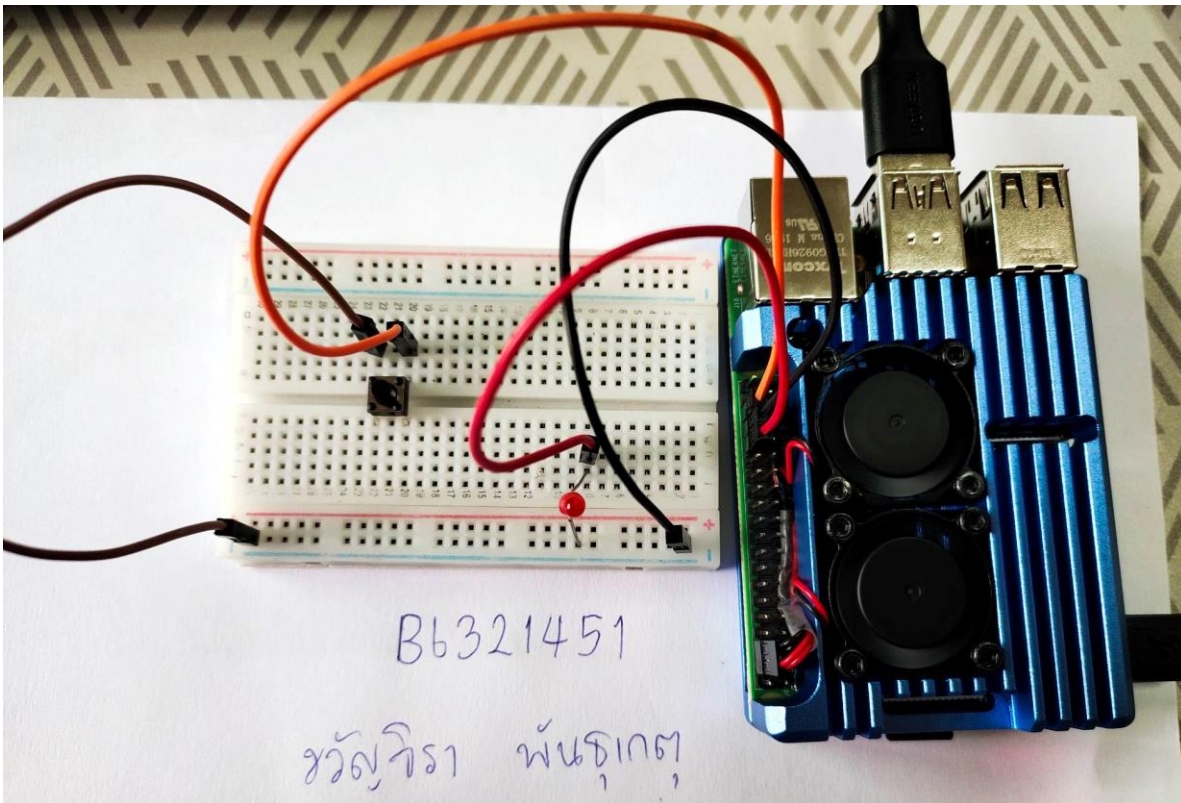
```
    time.sleep(0.5)
```



รูปการทดสอบ 1 [กดติด]

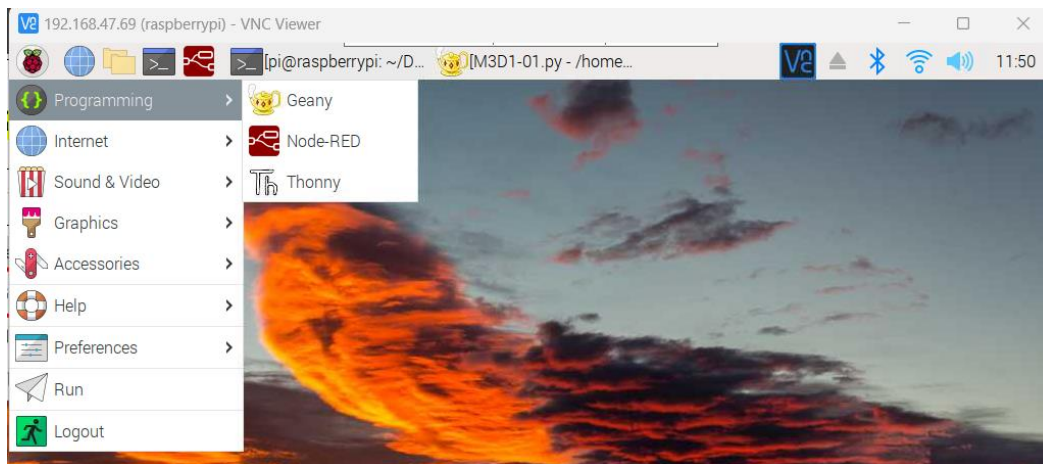


รูปการทดสอบ 2 [ปล่อยดับ]



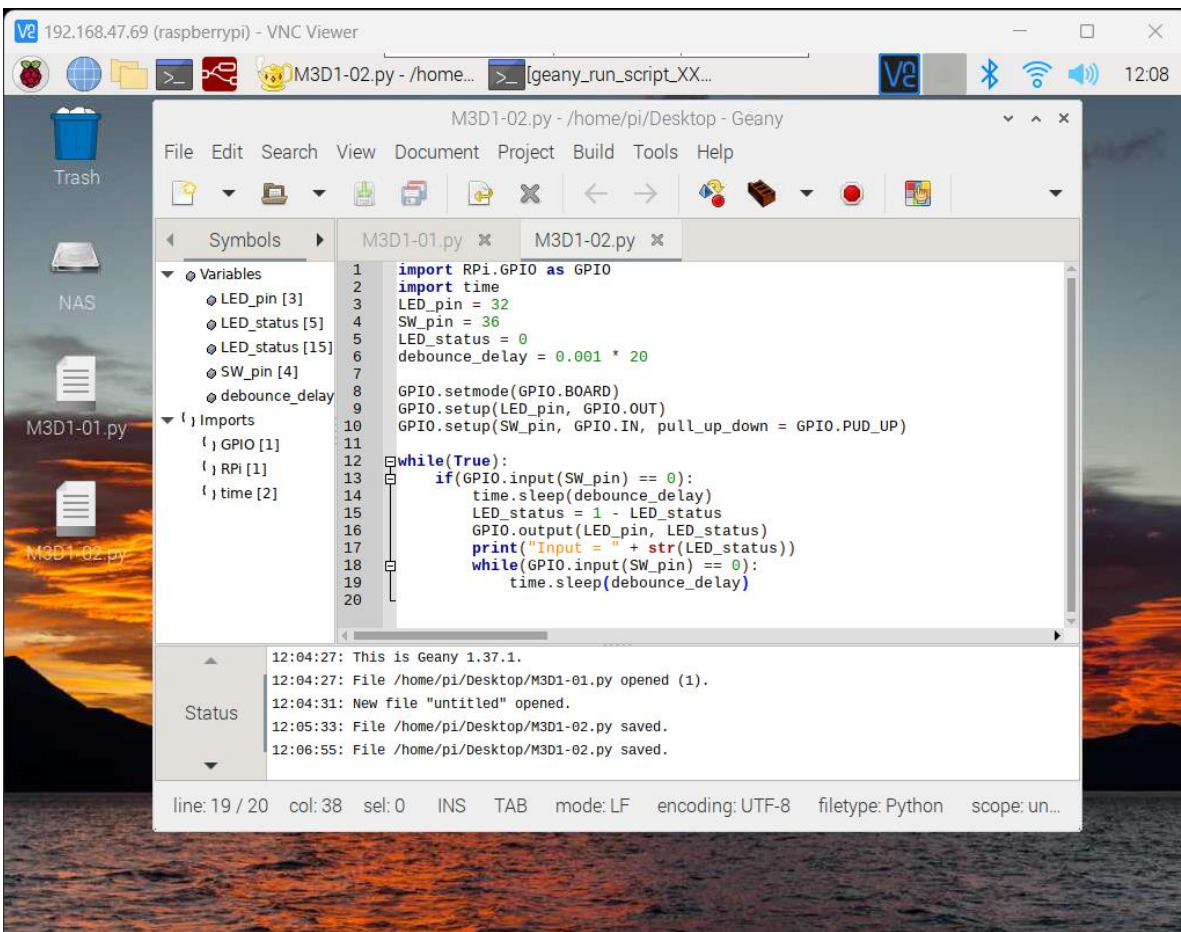
Python.2 - Python Switch control LED >> กดติด กดดับ

โปรแกรมที่ใช้ทดสอบ



ใช้โปรแกรม Geany

รูป Code Capture



```
import RPi.GPIO as GPIO

import time

LED_pin = 32

SW_pin = 36

LED_status = 0

debounce_delay = 0.001 * 20


GPIO.setmode(GPIO.BOARD)

GPIO.setup(LED_pin, GPIO.OUT)

GPIO.setup(SW_pin, GPIO.IN, pull_up_down = GPIO.PUD_UP)


while(True):

    if(GPIO.input(SW_pin) == 0):

        time.sleep(debounce_delay)

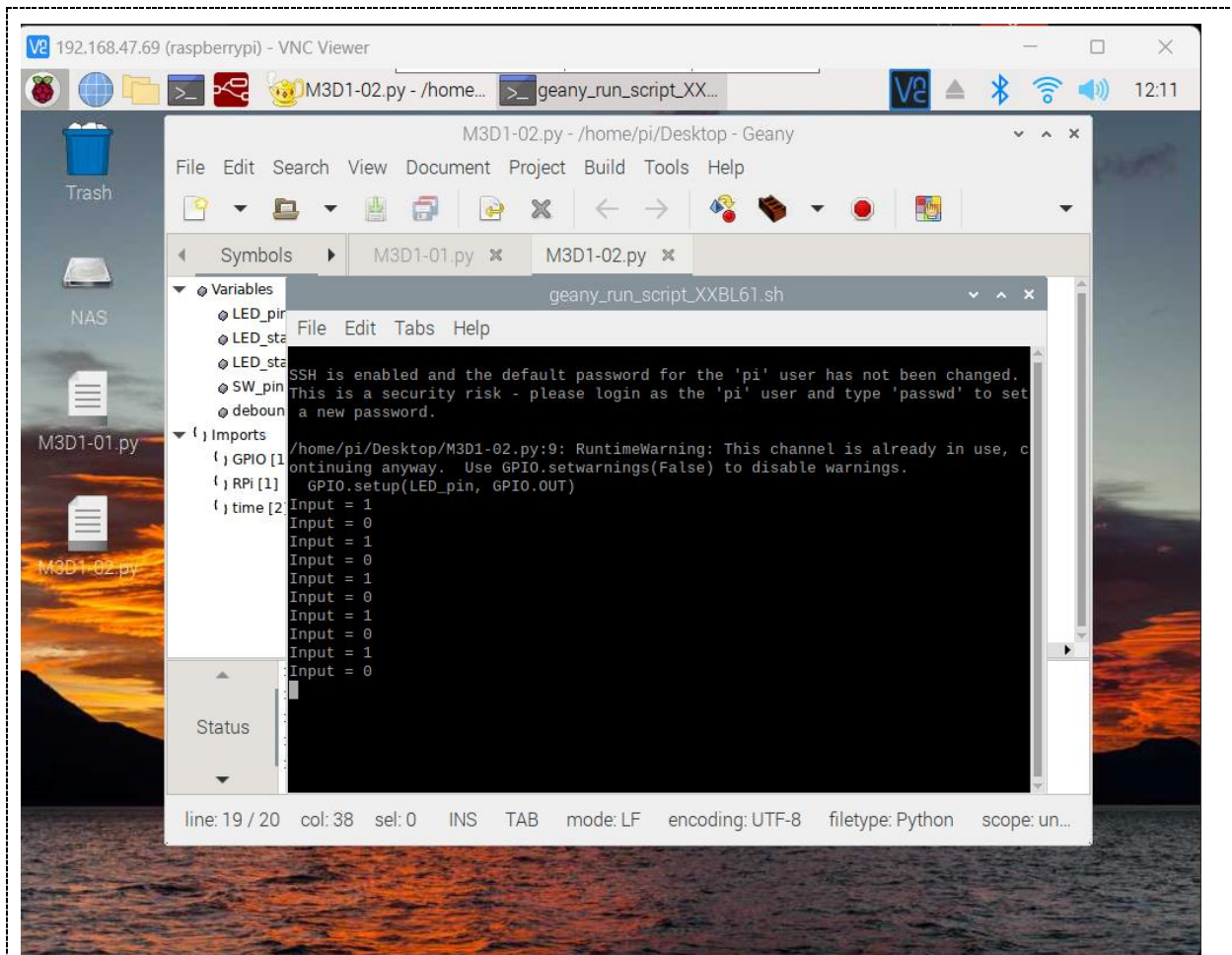
        LED_status = 1 - LED_status

        GPIO.output(LED_pin, LED_status)

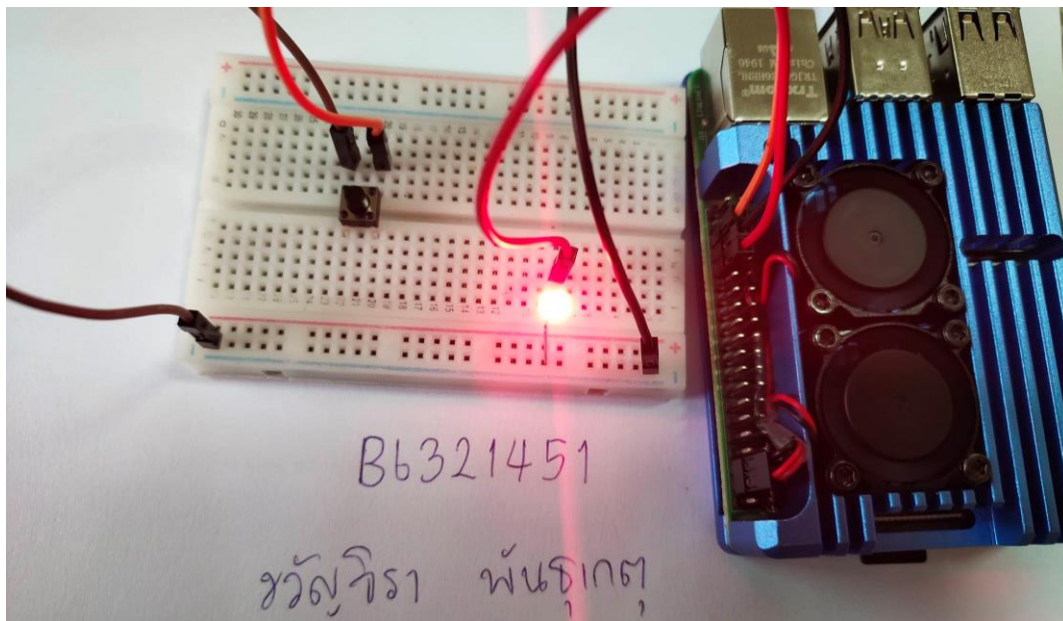
        print("Input = " + str(LED_status))

        while(GPIO.input(SW_pin) == 0):

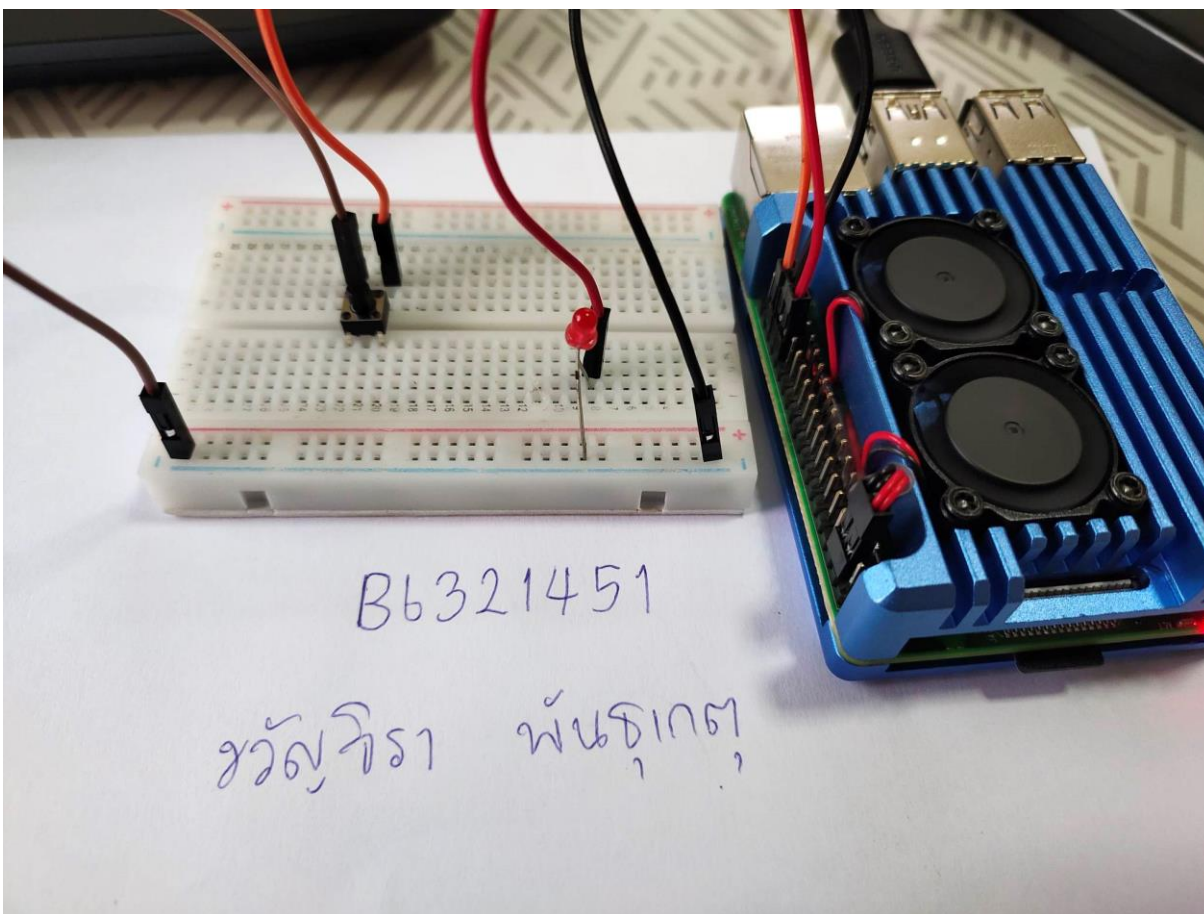
            time.sleep(debounce_delay)
```

รูปการทดสอบ 1

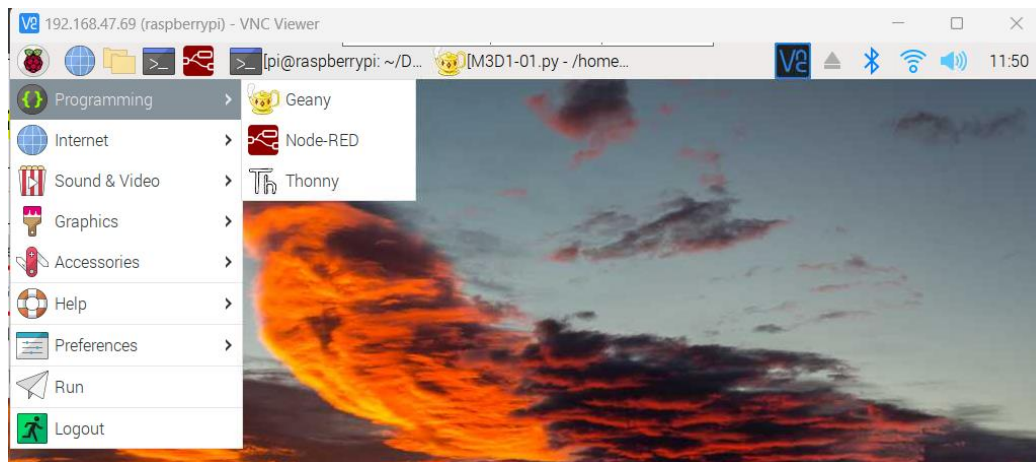


รูปการทดสอบ 2



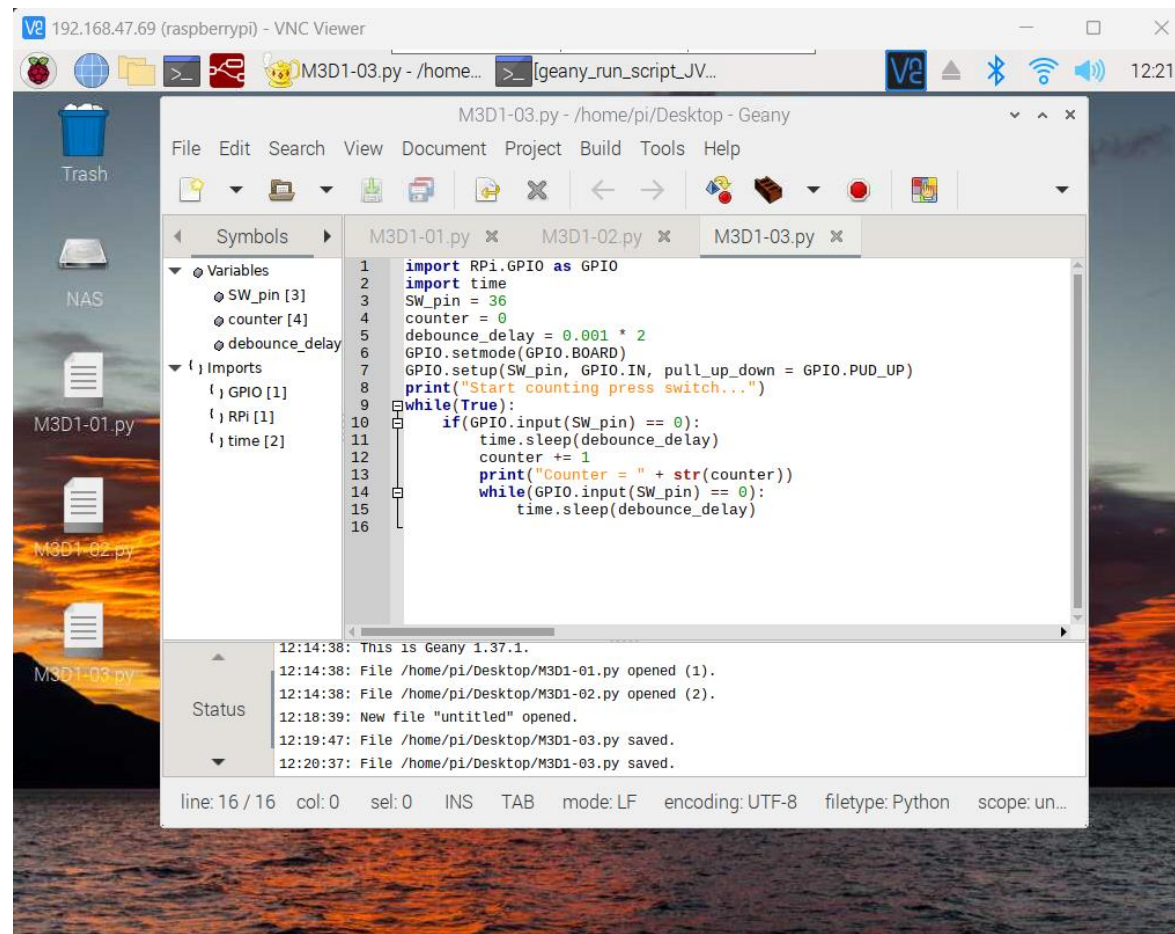
POython.3 - Python Switch >> Switch Counter

โปรแกรมที่ใช้ทดสอบ



ใช้โปรแกรม Geany

รูป Code Capture



```
import RPi.GPIO as GPIO

import time

SW_pin = 36

counter = 0

debounce_delay = 0.001 * 2

GPIO.setmode(GPIO.BOARD)

GPIO.setup(SW_pin, GPIO.IN, pull_up_down = GPIO.PUD_UP)

print("Start counting press switch...")

while(True):

    if(GPIO.input(SW_pin) == 0):

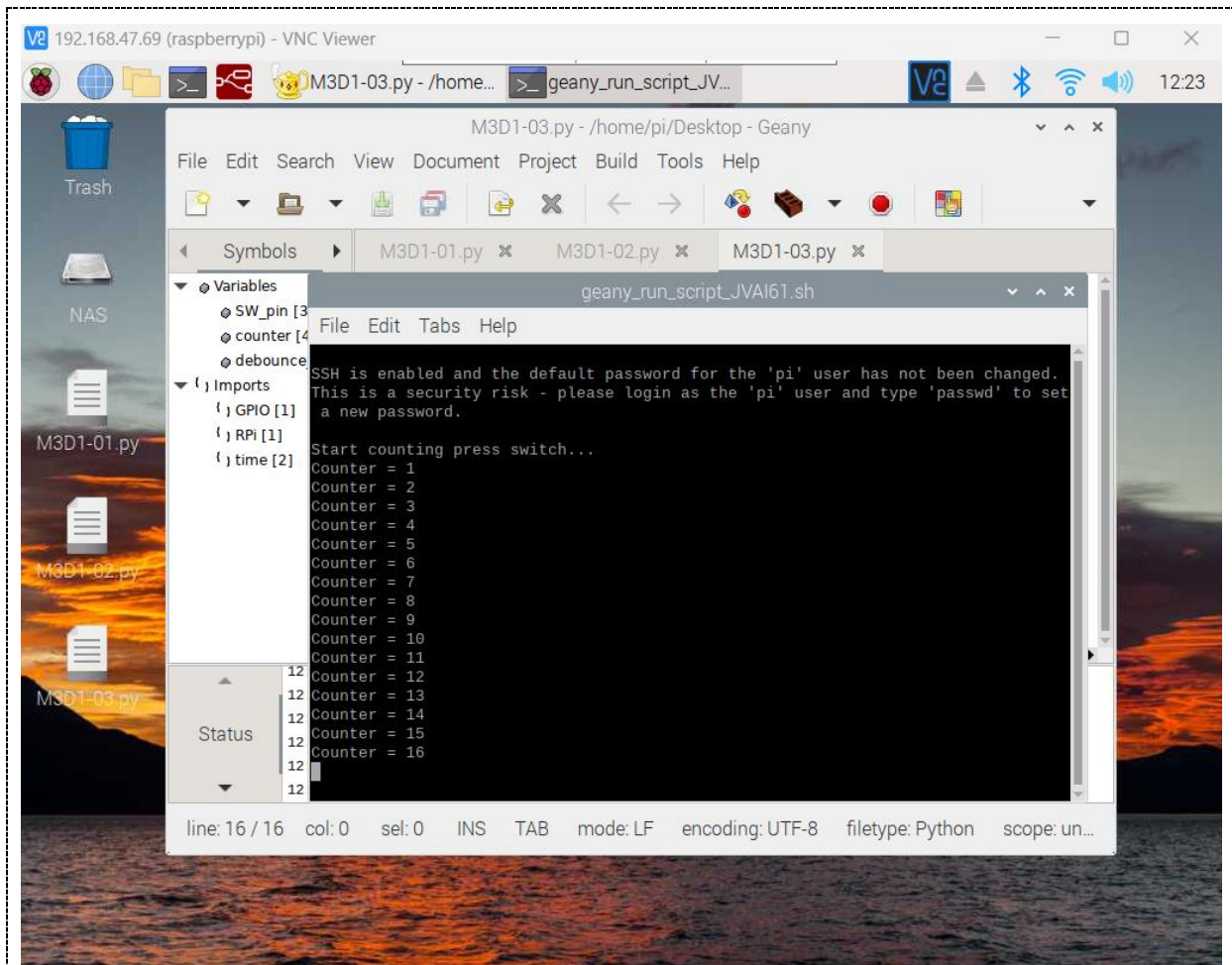
        time.sleep(debounce_delay)

        counter += 1

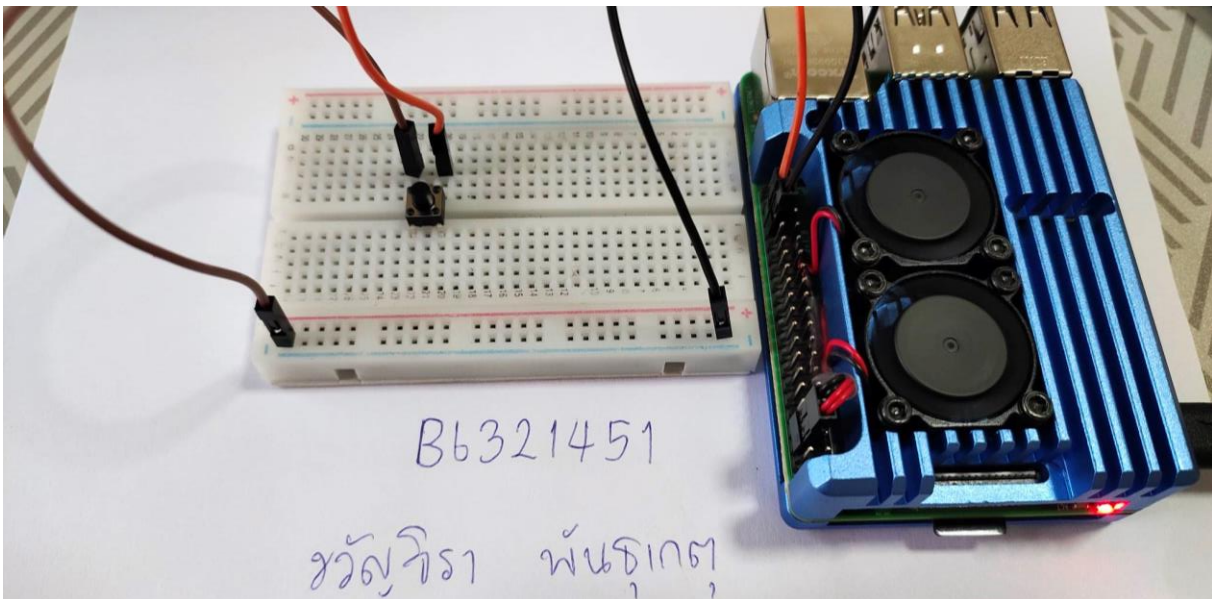
        print("Counter = " + str(counter))

        while(GPIO.input(SW_pin) == 0):

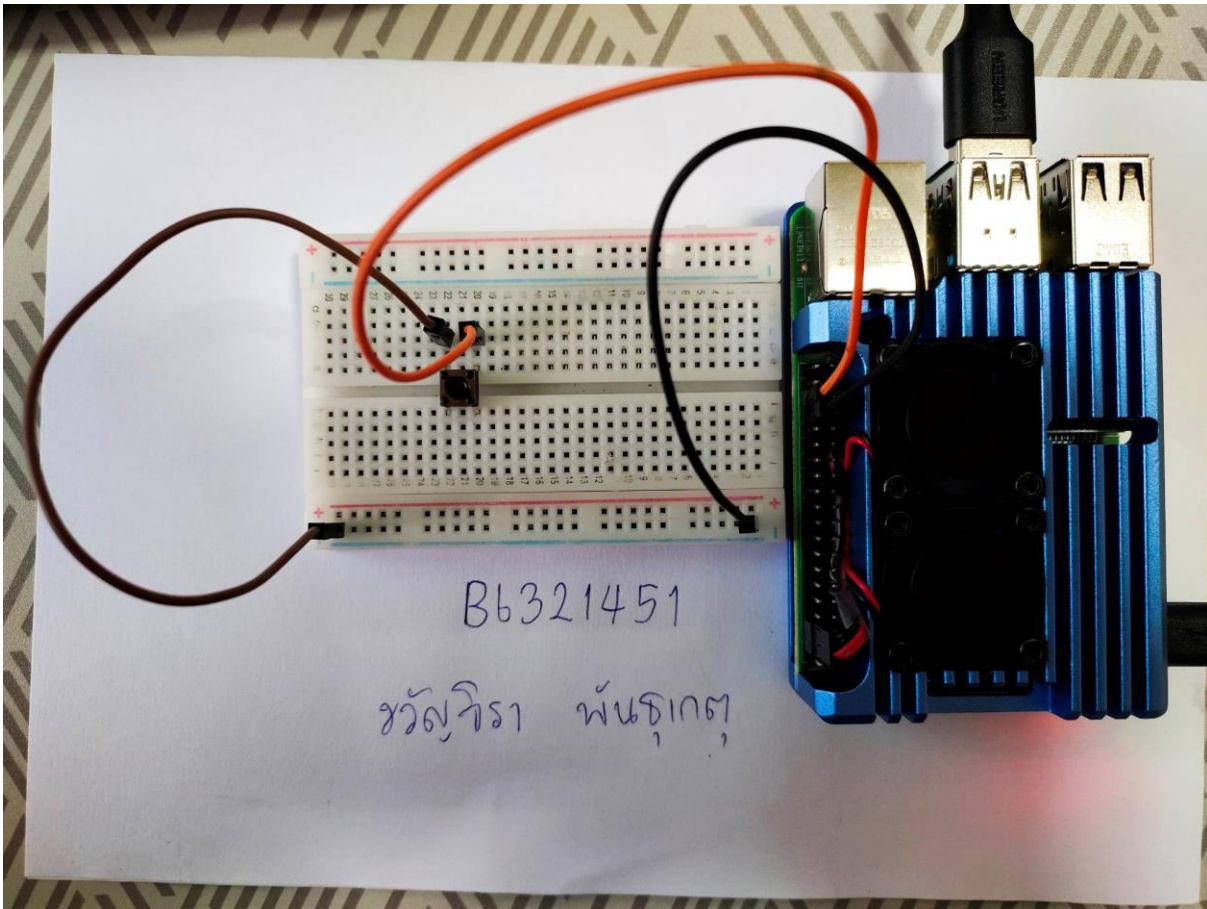
            time.sleep(debounce_delay)
```

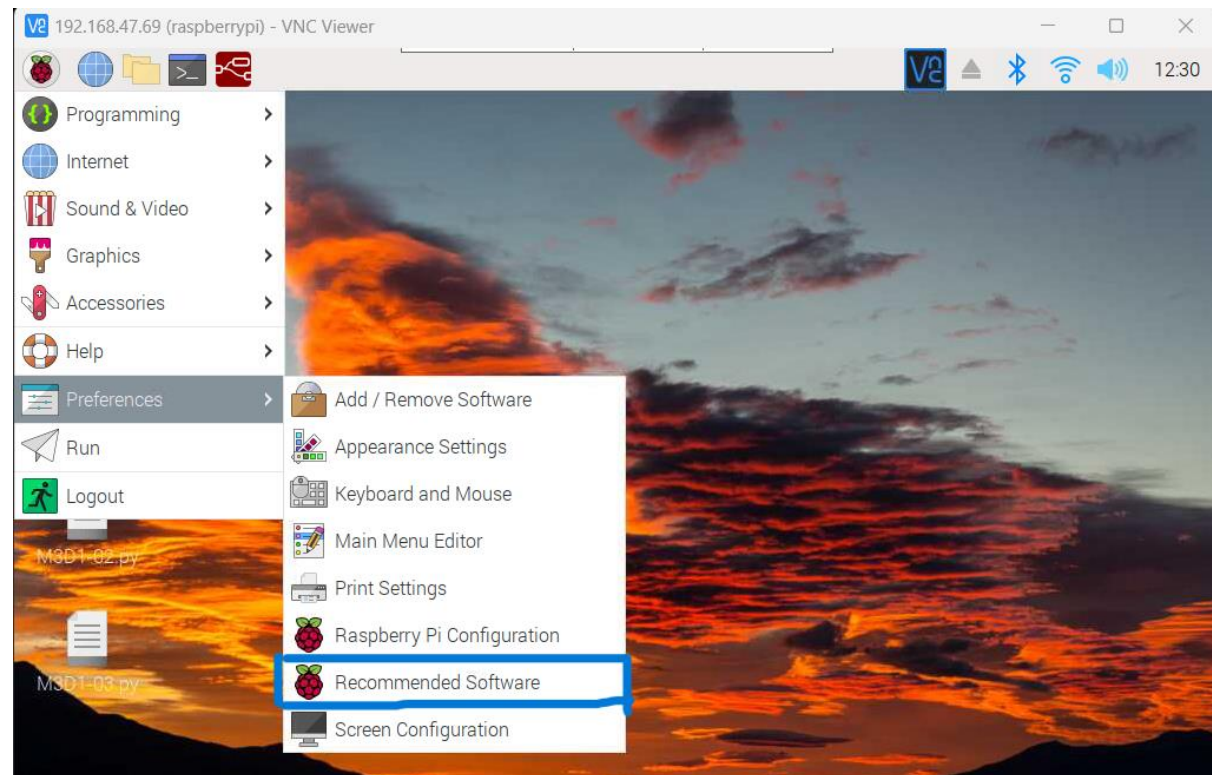


รูปการทดสอบ 1



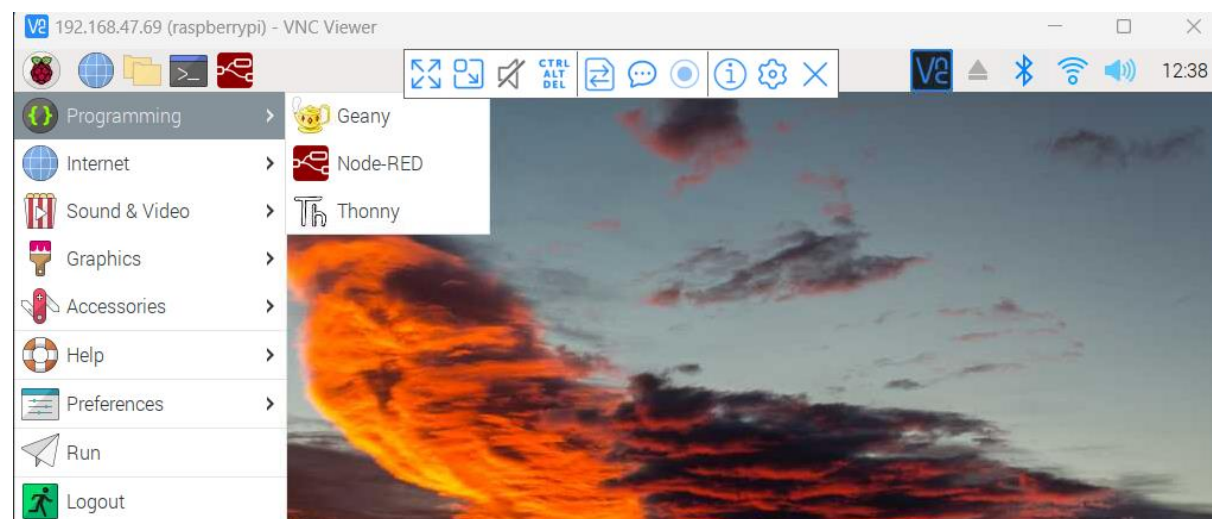
รูปการทดสอบ 2



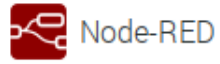
Quiz_102 – ทดสอบ RPi4 GPIO with Node-RED**Node-RED.1 – Node-RED เพื่อควบคุมสวิตช์กดแบบ กดติด กดดับ {Switch-LED 1 คู่}****โปรแกรมที่ใช้ทดสอบ**

เข้าไปที่ > Reccommeded Software ไปที่ช่องค้นหาแล้วพิมพ์ node-red แล้ว ติดตั้ง Node-RED

เมื่อติดตั้งเสร็จแล้วให้เปิด Node-Red โดยไปที่ Programming -> Node-RED



เมื่อเข้าไปที่ Node-RED



ให้นำ link ในกรอบสีแดงไปใส่ใน web browser เพื่อเข้าสู่ Node-Red

```

pi@raspberrypi: Node-RED console
File Edit Tabs Help

Start Node-RED

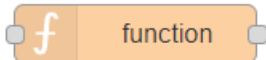
Once Node-RED has started, point a browser at http://192.168.47.69:1880
On Pi Node-RED works better with the Firefox or Chrome browser

Use node-red-stop to stop Node-RED
Use node-red-start to start Node-RED again
Use node-red-log to view the recent log output
Use sudo systemctl enable nodered.service to autostart Node-RED at every boot
Use sudo systemctl disable nodered.service to disable autostart on boot

To find more nodes and example flows - go to http://flows.nodered.org

Starting as a systemd service.
9 Jun 12:42:16 - [warn] [blynk-ws-client:910a50e5d4c756a0] Warning: You are trying to connect to the Blynk Cloud Server (blynk-cloud.com) which has not been shut down permanently as of year end 2022
9 Jun 12:42:16 - [info] [blynk-ws-client:910a50e5d4c756a0] Start ### secure connection: wss://blynk-cloud.com/websockets
9 Jun 12:42:16 - [error] [blynk-ws-client:910a50e5d4c756a0] Websocket Error: getaddrinfo ENOTFOUND blynk-cloud.com
9 Jun 12:42:16 - [info] [blynk-ws-client:910a50e5d4c756a0] Reconnect ### in 5 seconds...
  
```

รูป Code Capture - เลือก



Edit function node

Delete Cancel Done

Properties

Name toggle state 1

Setup On Start **On Message** On Stop

```

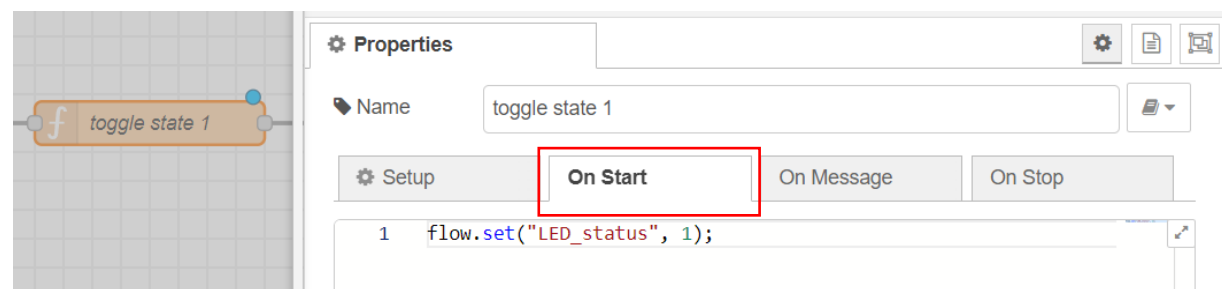
1 var LED_status = 1;
2 if (msg.payload) {
3   flow.set("LED_status", 1 - flow.get("LED_status"));
4 }
5 msg.payload = 1;
6 return msg;
  
```



```

var LED_status = 1;
if (msg.payload) {
  flow.set("LED_status", 1 - flow.get("LED_status"));
}
msg.payload = 1;
return msg;

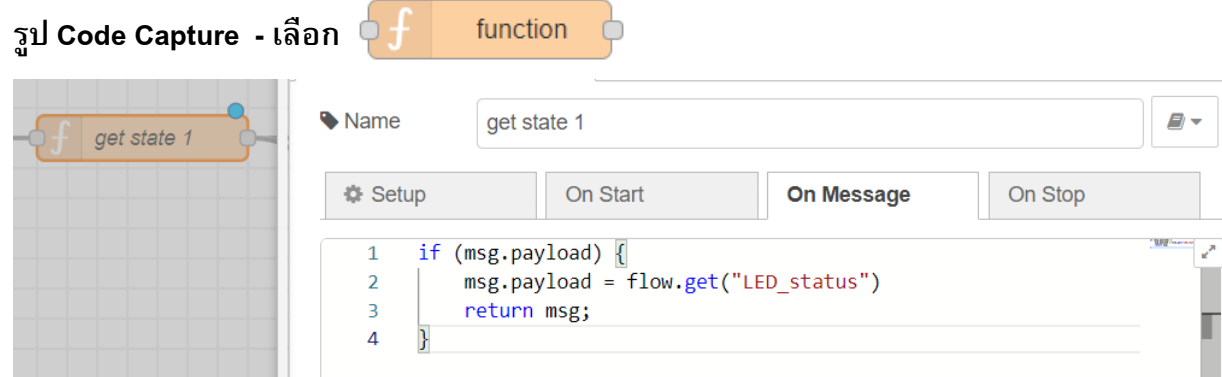
```



```

flow.set("LED_status", 1);

```



```

if (msg.payload) {
  msg.payload = flow.get("LED_status")
  return msg;
}

```

รูป Code Capture - เลือก

rpi - gpio in

Delete Cancel Done

⚙ Properties

GPIO04 - 7	8 - GPIO14 - TxD
Ground - 9	10 - GPIO15 - Rx D
GPIO17 - 11	12 - GPIO18
GPIO27 - 13	14 - Ground
GPIO22 - 15	16 - GPIO23
3.3V Power - 17	18 - GPIO24
MOSI - GPIO10 - 19	20 - Ground
MISO - GPIO09 - 21	22 - GPIO25
SCLK - GPIO11 - 23	24 - GPIO8 - CE0
Ground - 25	26 - GPIO7 - CE1
SD - 27	28 - SC
GPIO05 - 29	30 - Ground
GPIO06 - 31	32 - GPIO12
GPIO13 - 33	34 - Ground
GPIO19 - 35	36 - GPIO16
GPIO26 - 37	38 - GPIO20
Ground - 39	40 - GPIO21

BCM GPIO 16

↑ Resistor? pullup Debounce 25 ms

☐ Read initial state of pin on deploy/restart?

รูป Code Capture - เลือก

rpi - mouse

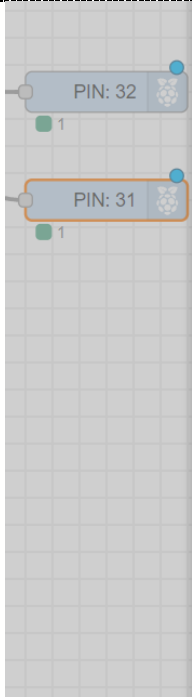
มา 2 node

⚙ Properties

GPIO17 - 11	12 - GPIO18
GPIO27 - 13	14 - Ground
GPIO22 - 15	16 - GPIO23
3.3V Power - 17	18 - GPIO24
MOSI - GPIO10 - 19	20 - Ground
MISO - GPIO09 - 21	22 - GPIO25
SCLK - GPIO11 - 23	24 - GPIO8 - CE0
Ground - 25	26 - GPIO7 - CE1
SD - 27	28 - SC
GPIO05 - 29	30 - Ground
GPIO06 - 31	32 - GPIO12
GPIO13 - 33	34 - Ground
GPIO19 - 35	36 - GPIO16
GPIO26 - 37	38 - GPIO20
Ground - 39	40 - GPIO21

BCM GPIO 12

Type Digital output



Properties

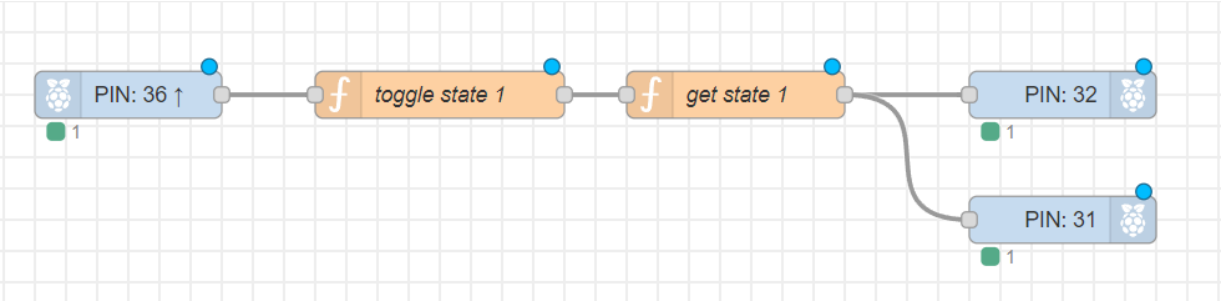
GPIO17 - 11	GPIO18
GPIO27 - 13	14 - Ground
GPIO22 - 15	16 - GPIO23
3.3V Power - 17	18 - GPIO24
MOSI - GPIO10 - 19	20 - Ground
MISO - GPIO09 - 21	22 - GPIO25
SCLK - GPIO11 - 23	24 - GPIO8 - CE0
Ground - 25	26 - GPIO7 - CE1
SD - 27	28 - SC
GPIO05 - 29	30 - Ground
GPIO06 - 31	32 - GPIO12
GPIO13 - 33	34 - Ground
GPIO19 - 35	36 - GPIO16
GPIO26 - 37	38 - GPIO20
Ground - 39	40 - GPIO21

BCM GPIO 6

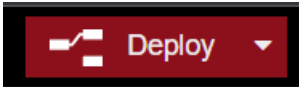
Type Digital output

☐ Initialise pin state?

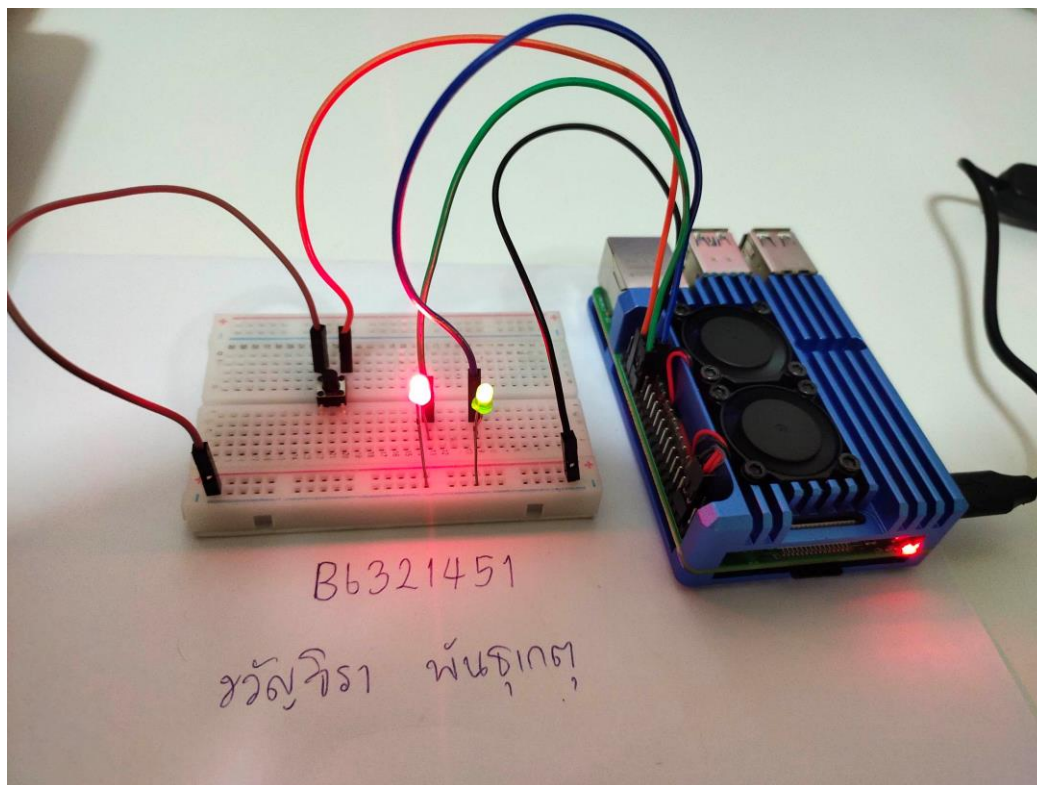
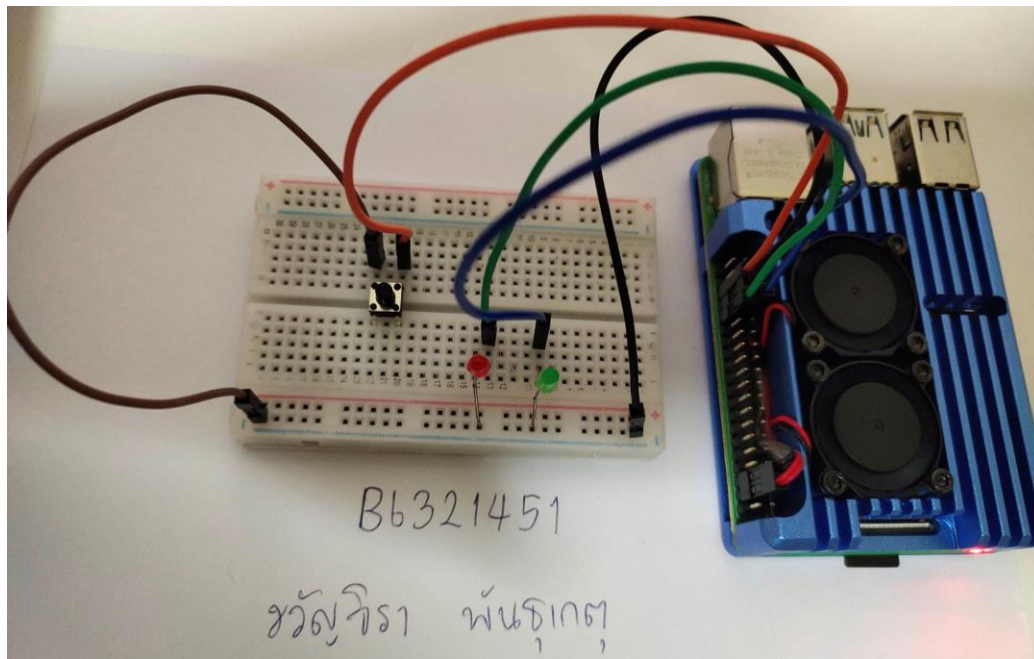
นำทุก node ต่อกัน



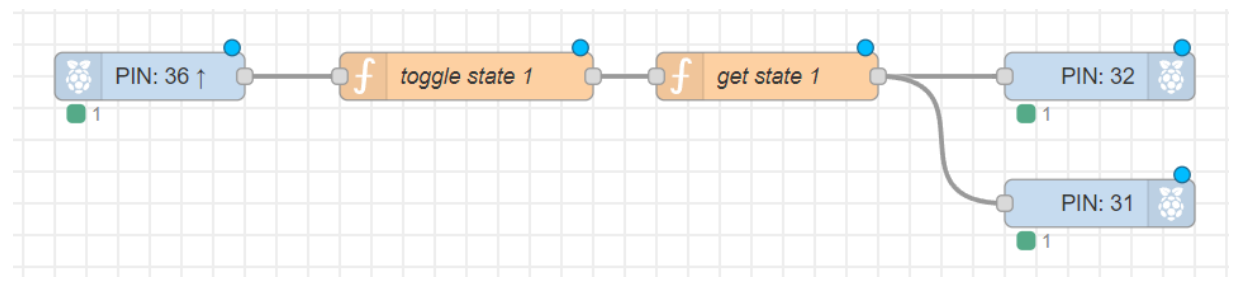
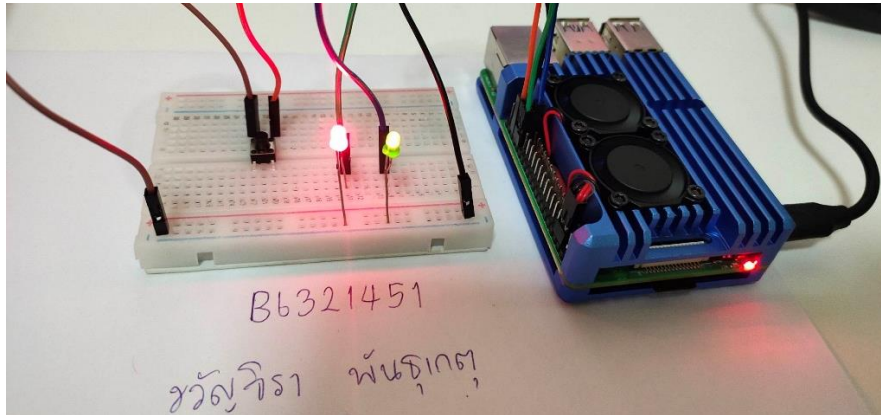
จากนั้นกด



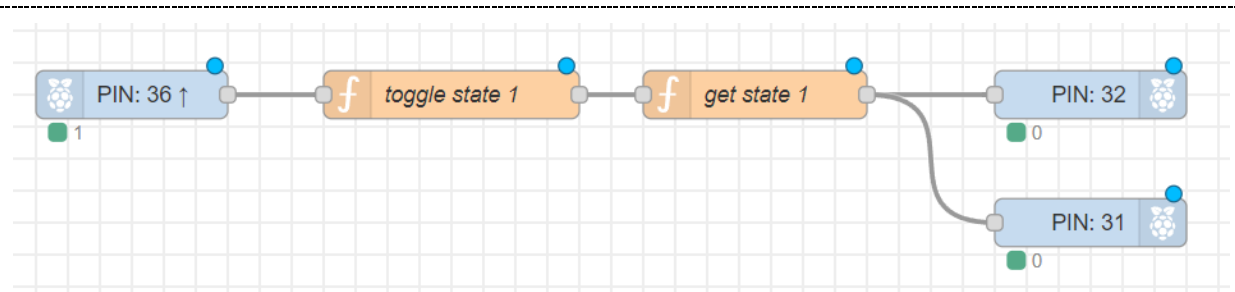
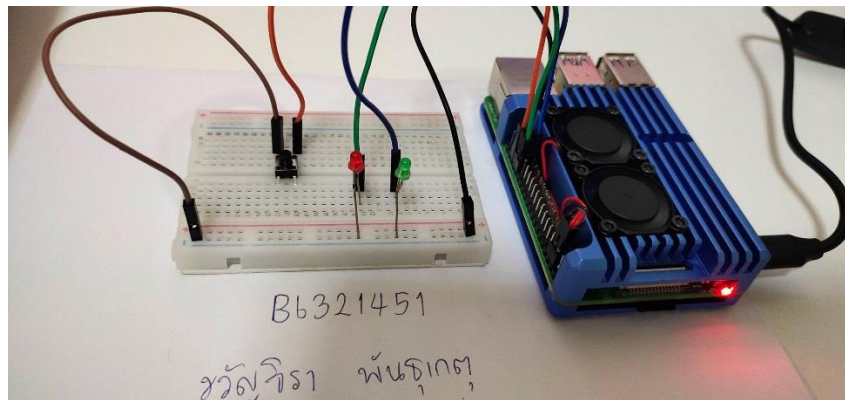
รูปวงจร



รูปการณ์ทดลอง 1 - เมื่อกดสวิตช์แล้วปล่อย LED ติด

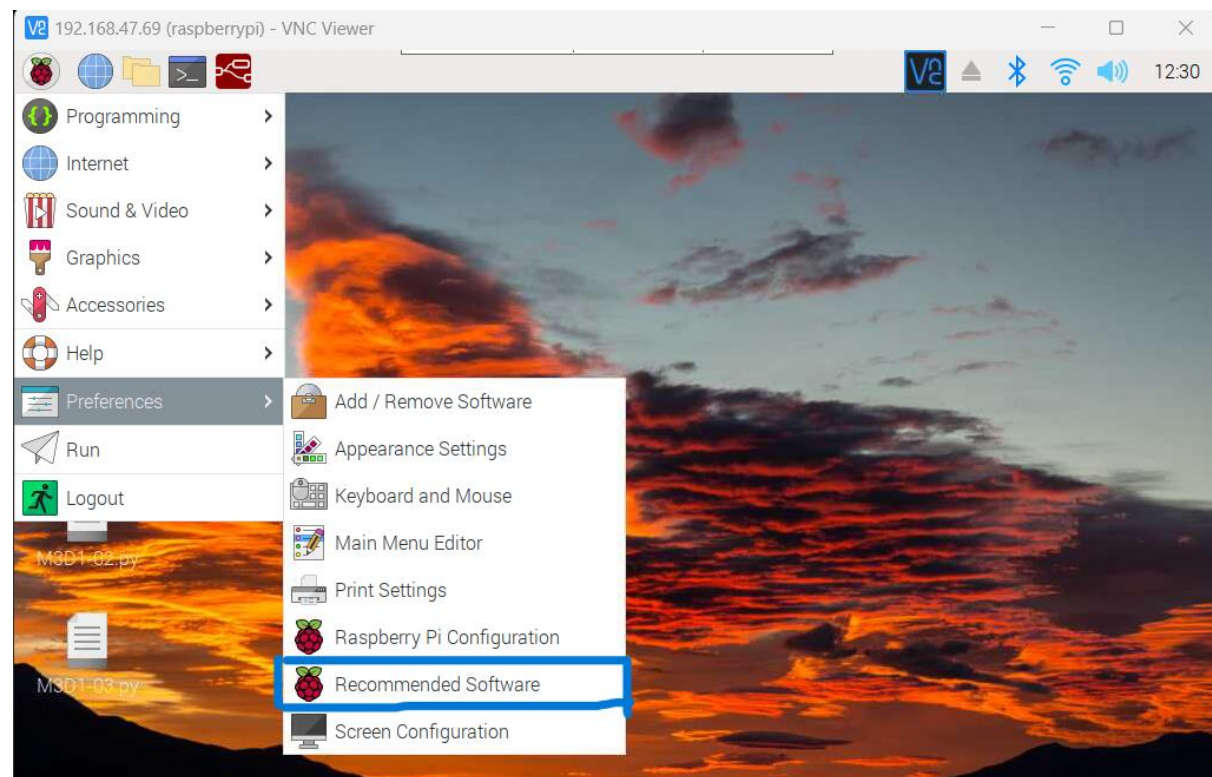


รูปการณ์ทดลอง 1 - เมื่อกดสวิตช์แล้วปล่อย LED ดับ



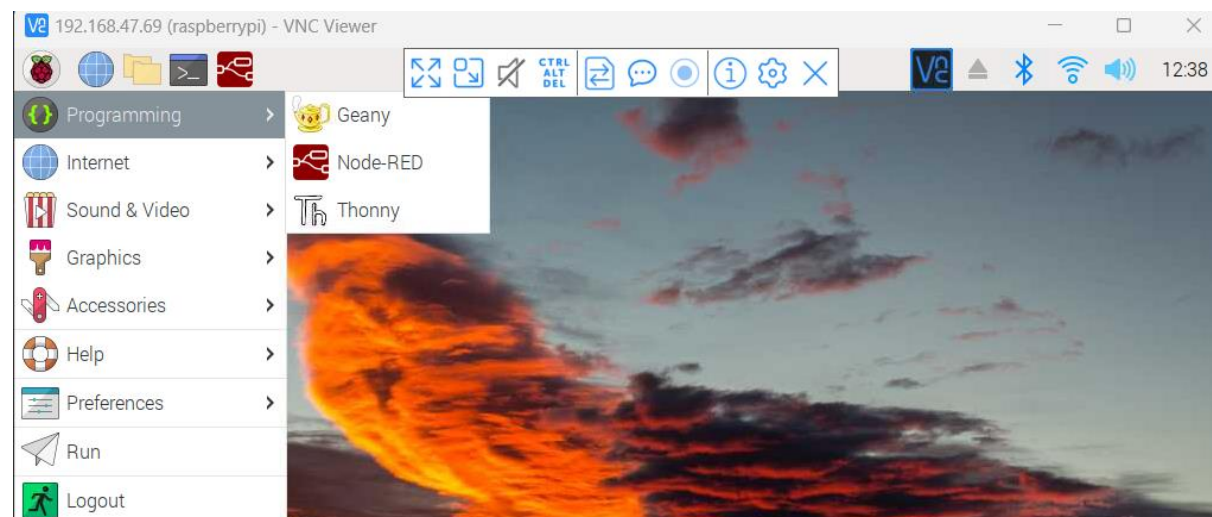
Node-RED.2 - Node-RED เพื่อควบคุมสวิตช์กดแบบ กดติด กดดับ 2 คู่

โปรแกรมที่ใช้ทดสอบ



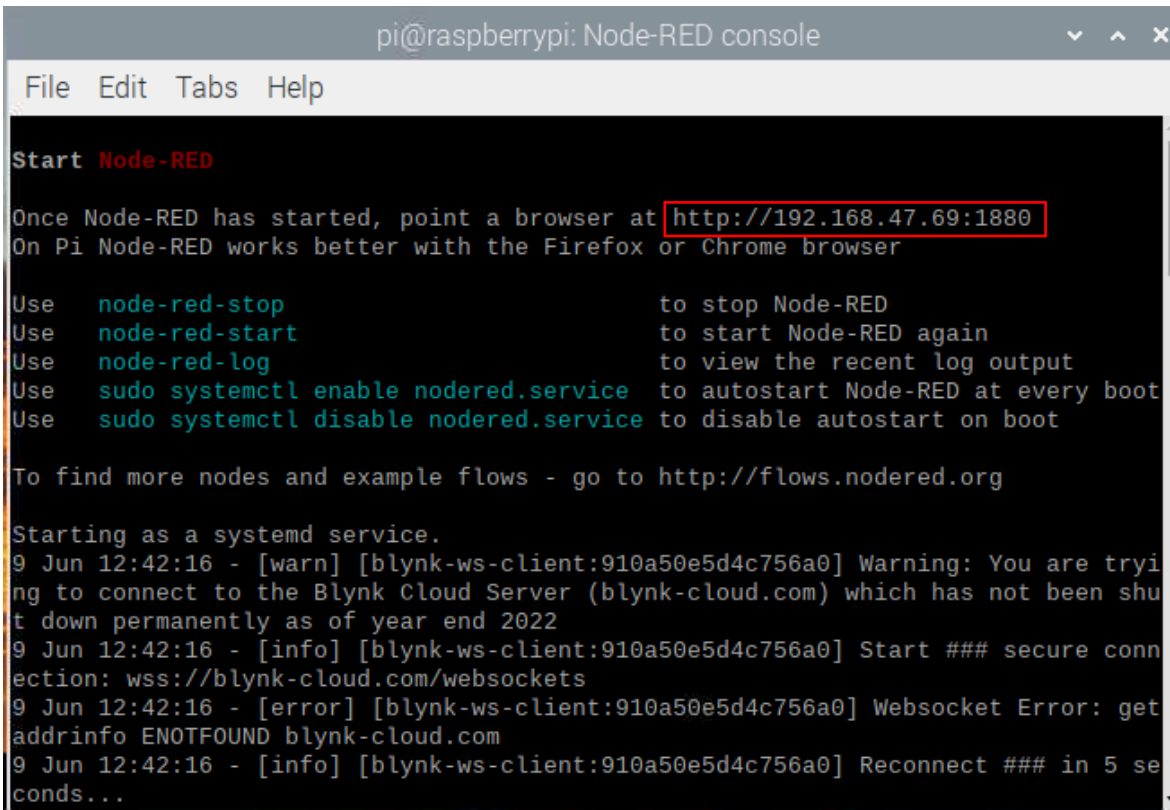
เข้าไปที่ > Recommended Software ไปที่ช่องค้นหาแล้วพิมพ์ node-red แล้ว ติดตั้ง Node-RED

เมื่อติดตั้งเสร็จแล้วให้เปิด Node-Red โดยไปที่ Programming -> Node-RED



เมื่อเข้าไปที่ Node-RED  Node-RED

ให้นำ link ในกรอบสีแดงไปใส่ใน web browser เพื่อเข้าสู่ Node-Red



```

pi@raspberrypi: Node-RED console
File Edit Tabs Help

Start Node-RED

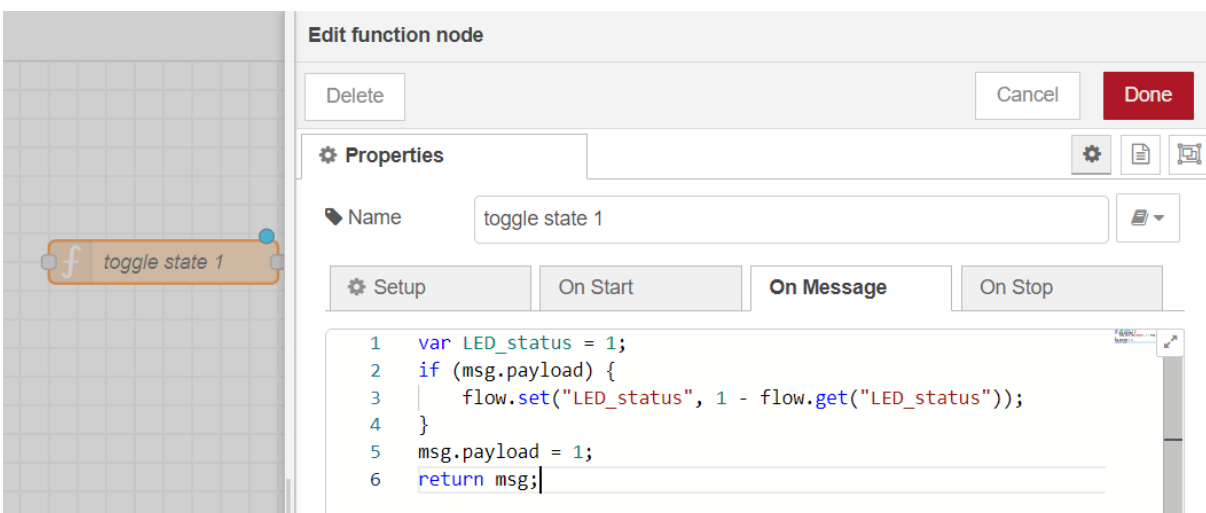
Once Node-RED has started, point a browser at http://192.168.47.69:1880
On Pi Node-RED works better with the Firefox or Chrome browser

Use node-red-stop to stop Node-RED
Use node-red-start to start Node-RED again
Use node-red-log to view the recent log output
Use sudo systemctl enable nodered.service to autostart Node-RED at every boot
Use sudo systemctl disable nodered.service to disable autostart on boot

To find more nodes and example flows - go to http://flows.nodered.org

Starting as a systemd service.
9 Jun 12:42:16 - [warn] [blynk-ws-client:910a50e5d4c756a0] Warning: You are trying to connect to the Blynk Cloud Server (blynk-cloud.com) which has not been shut down permanently as of year end 2022
9 Jun 12:42:16 - [info] [blynk-ws-client:910a50e5d4c756a0] Start ### secure connection: wss://blynk-cloud.com/websockets
9 Jun 12:42:16 - [error] [blynk-ws-client:910a50e5d4c756a0] Websocket Error: getaddrinfo ENOTFOUND blynk-cloud.com
9 Jun 12:42:16 - [info] [blynk-ws-client:910a50e5d4c756a0] Reconnect ### in 5 seconds...
  
```

รูป Code Capture – เลือก  function  มา 2 node



```

Edit function node
Delete Cancel Done

Properties
Name toggle state 1

Setup On Start On Message On Stop

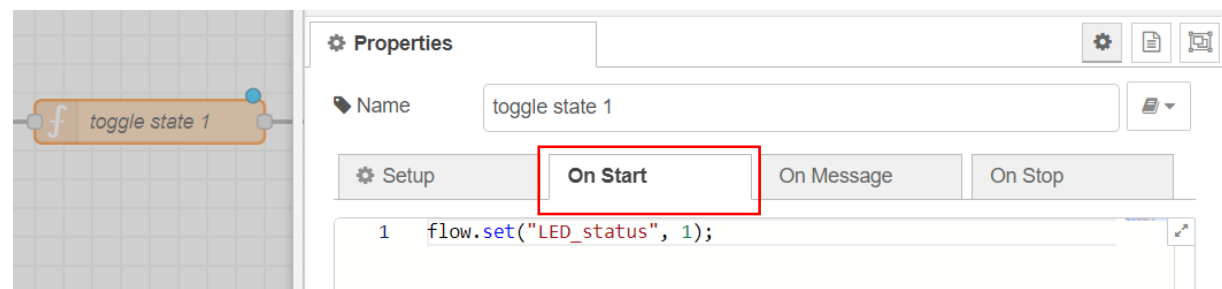
1 var LED_status = 1;
2 if (msg.payload) {
3   flow.set("LED_status", 1 - flow.get("LED_status"));
4 }
5 msg.payload = 1;
6 return msg;
  
```



```

var LED_status = 1;
if (msg.payload) {
    flow.set("LED_status", 1 - flow.get("LED_status"));
}
msg.payload = 1;
return msg;

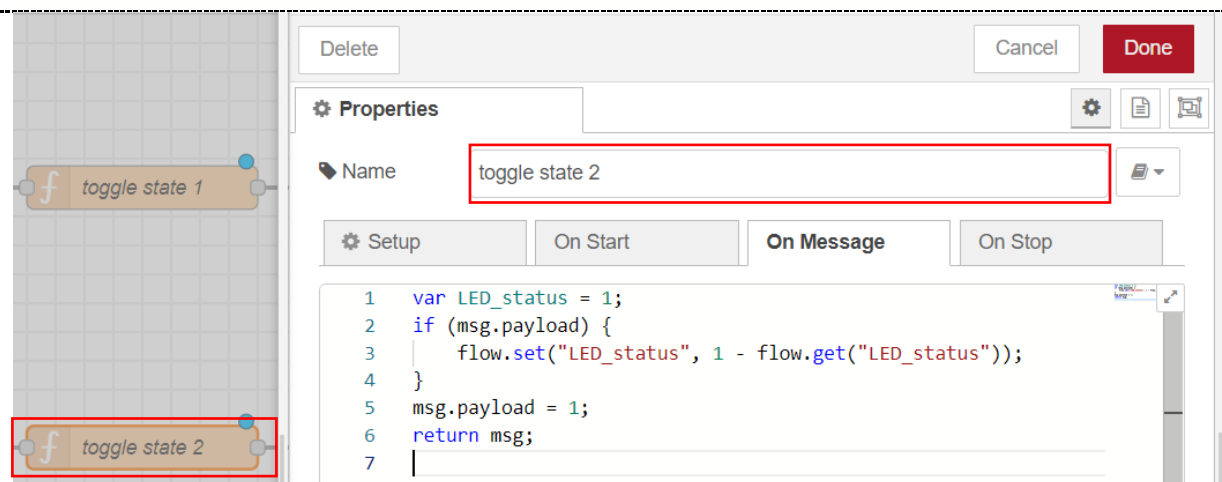
```



```

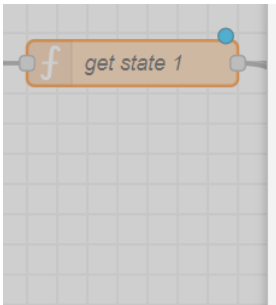
flow.set("LED_status", 1);

```



รูป Code Capture - เลือก

function



Name

get state 1

Setup

On Start

On Message

On Stop

```

1  if (msg.payload) {
2    msg.payload = flow.get("LED_status")
3    return msg;
4  }

```

```

if (msg.payload) {
  msg.payload = flow.get("LED_status")
  return msg;
}

```

รูป Code Capture - เลือก

function

Properties

Name

get state 2

Setup

On Start

On Message

On Stop

```

1  if (msg.payload) {
2    msg.payload = flow.get("LED_status")
3    return msg;
4  }

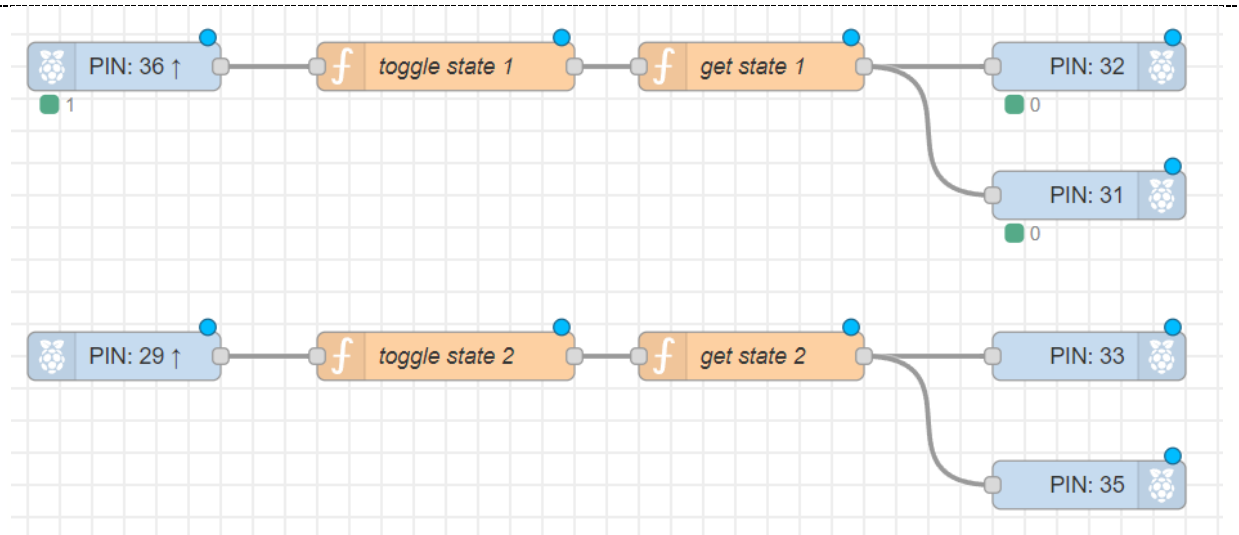
```

```

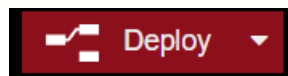
if (msg.payload) {
  msg.payload = flow.get("LED_status")
  return msg;
}

```

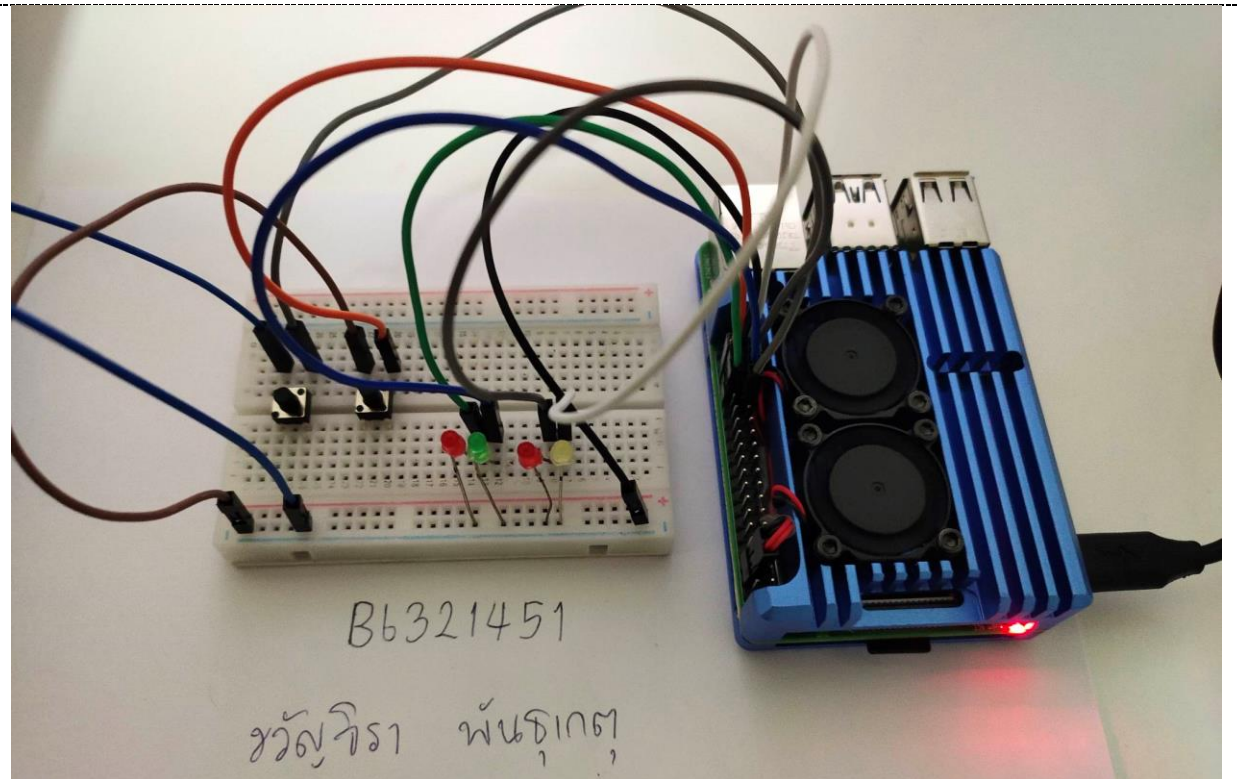
รูป Code Capture -- หน้า node มาต่อรวมกัน



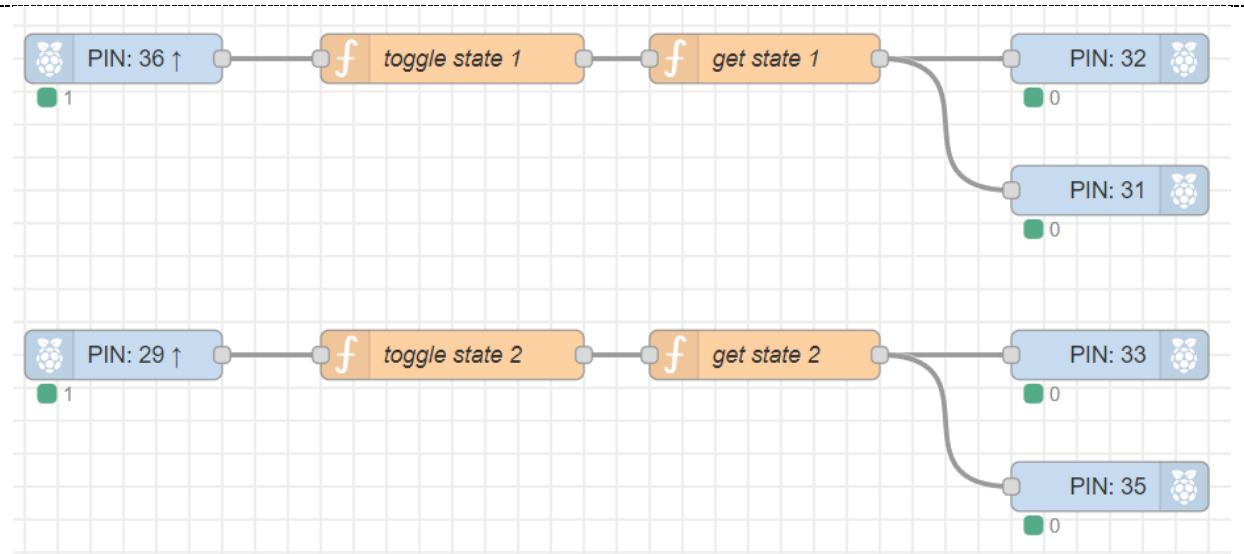
จากนั้นกด



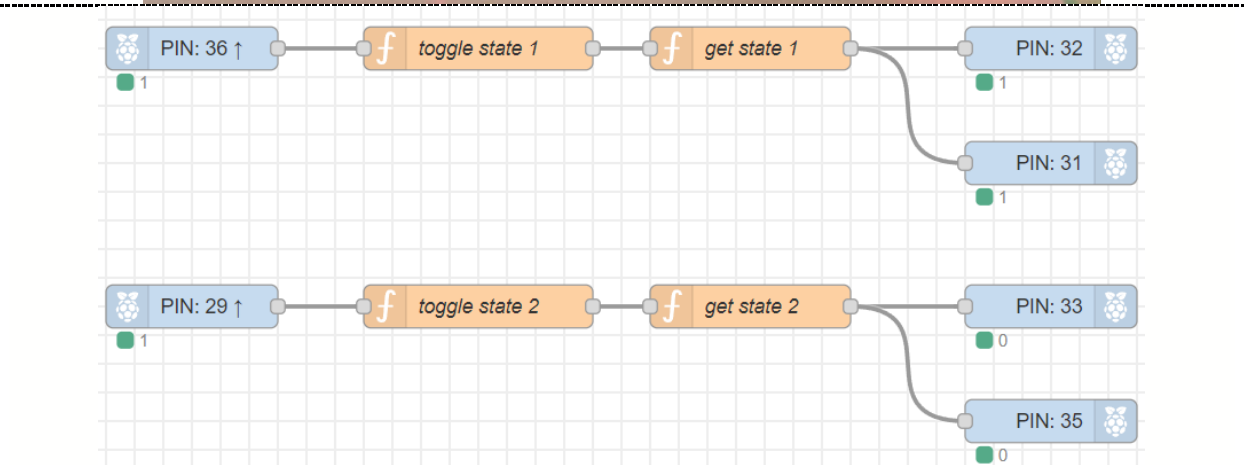
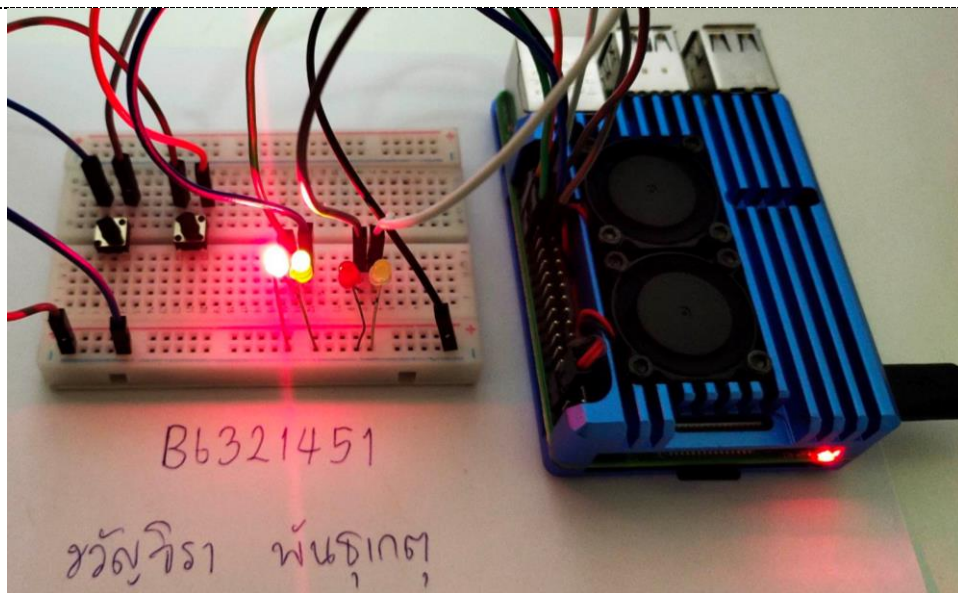
รูปวงจร และการทดสอบ 1



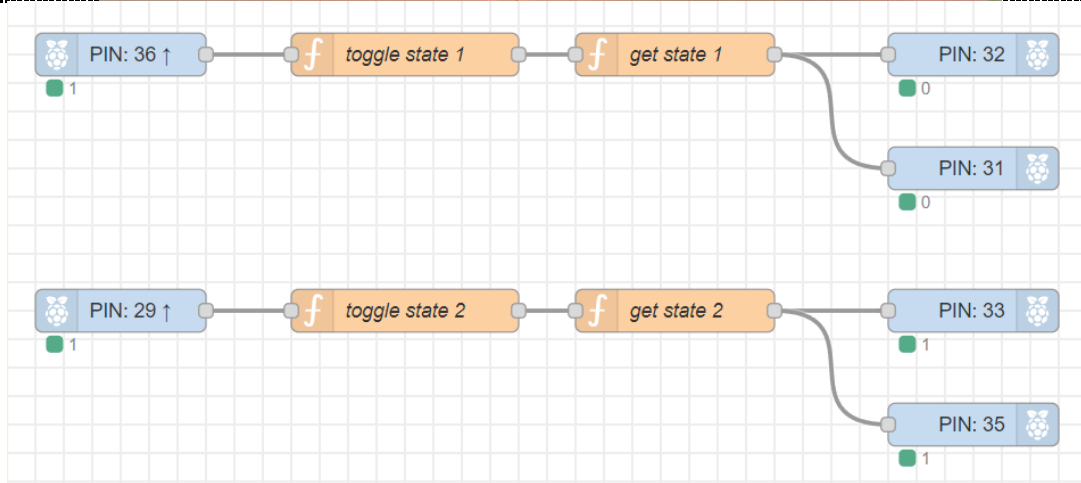
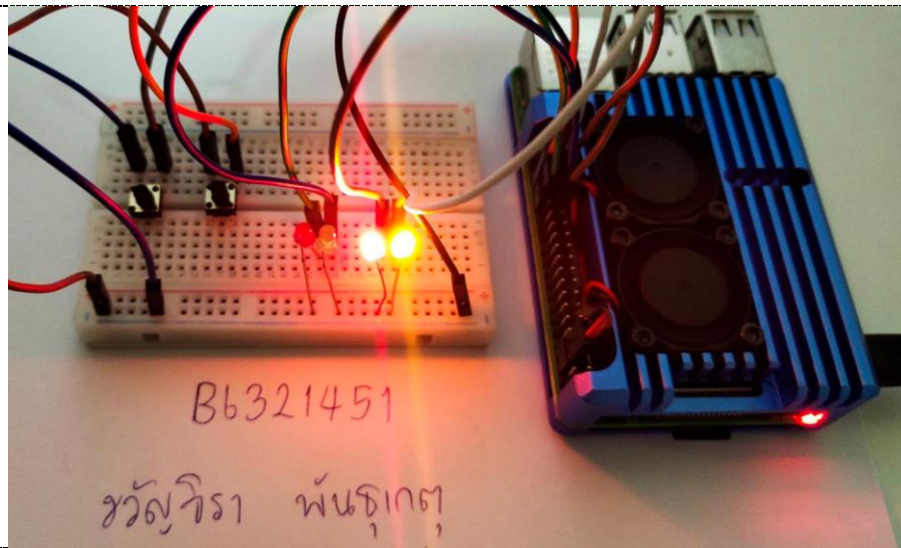
รูปวงจร และการทดสอบ 1 – ไม่กดสวิช LED ดับ



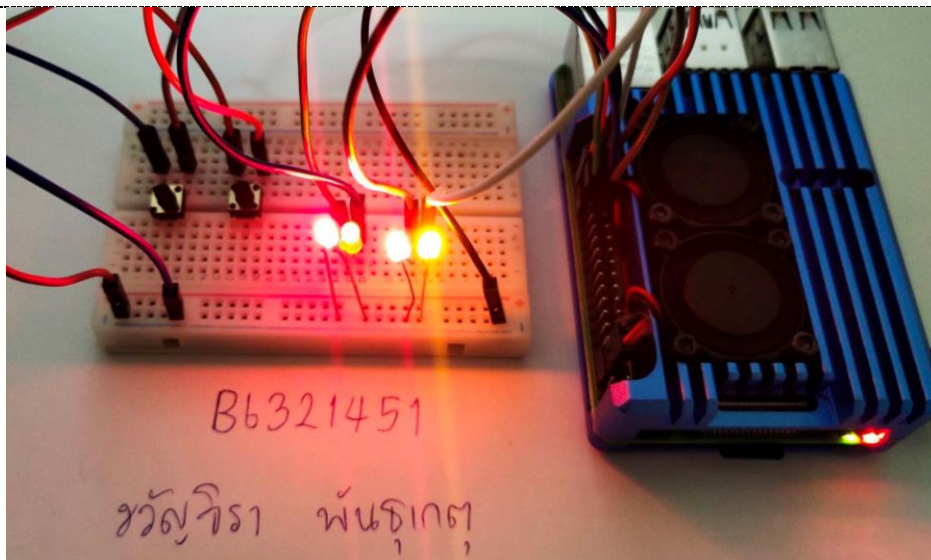
รูปวงจร และการทดสอบ 2 – กดสวิชที่ 1

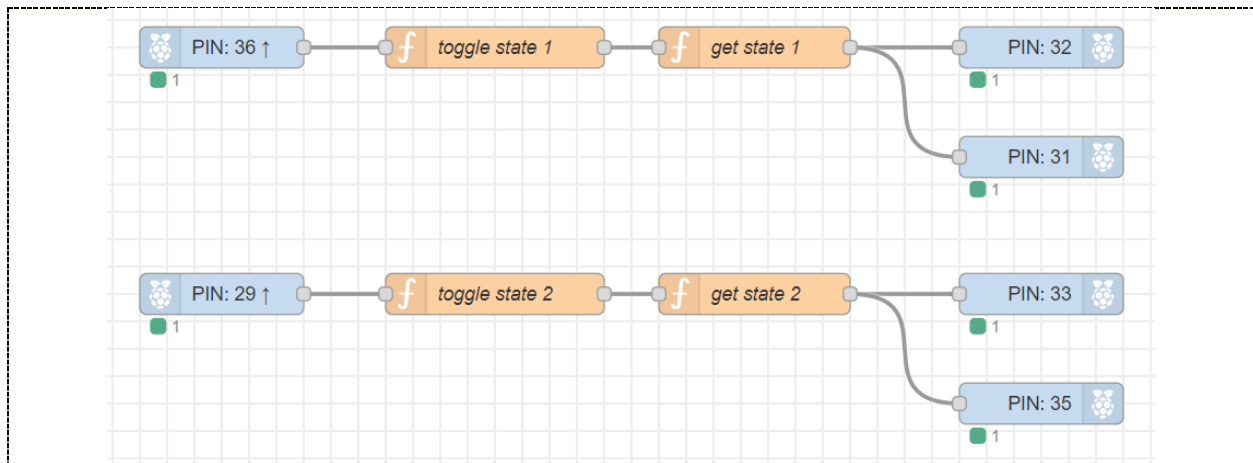


รูปวงจร และการทดสอบ 3 – กดสวิชที่ 2



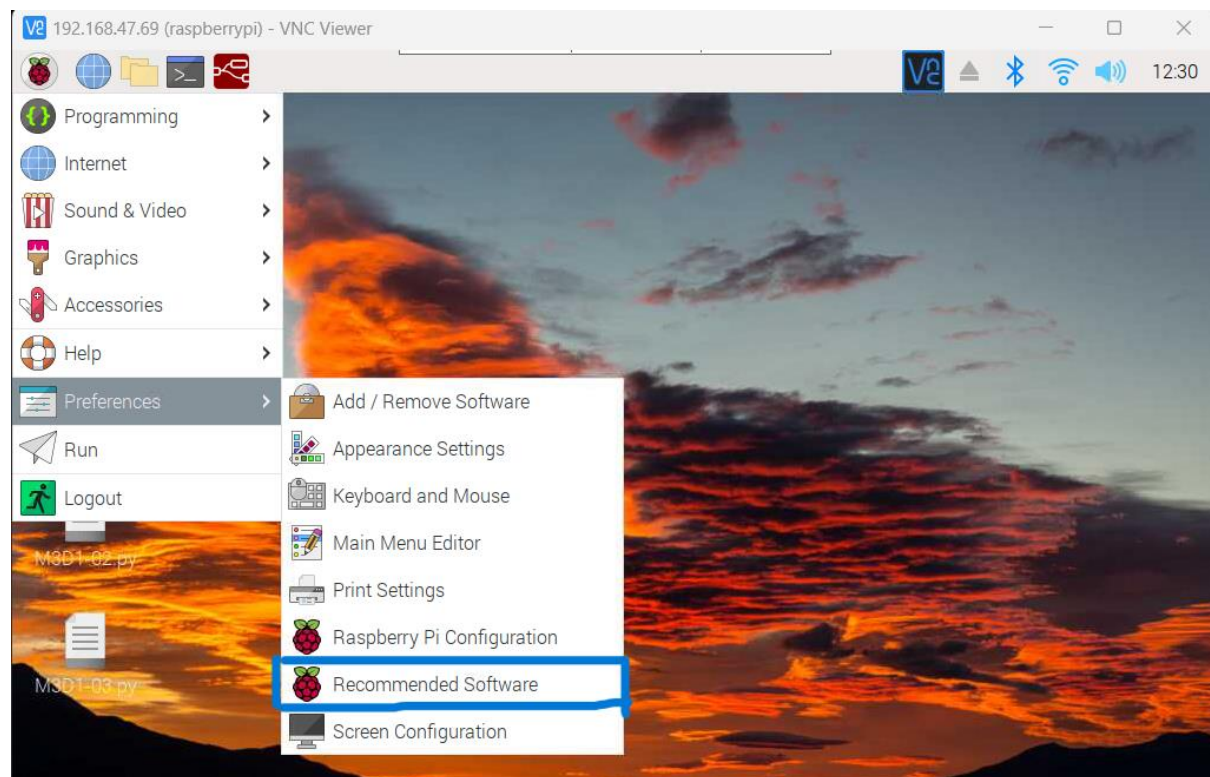
รูปวงจร และการทดสอบ 4 – กดสวิช ทั้ง 2





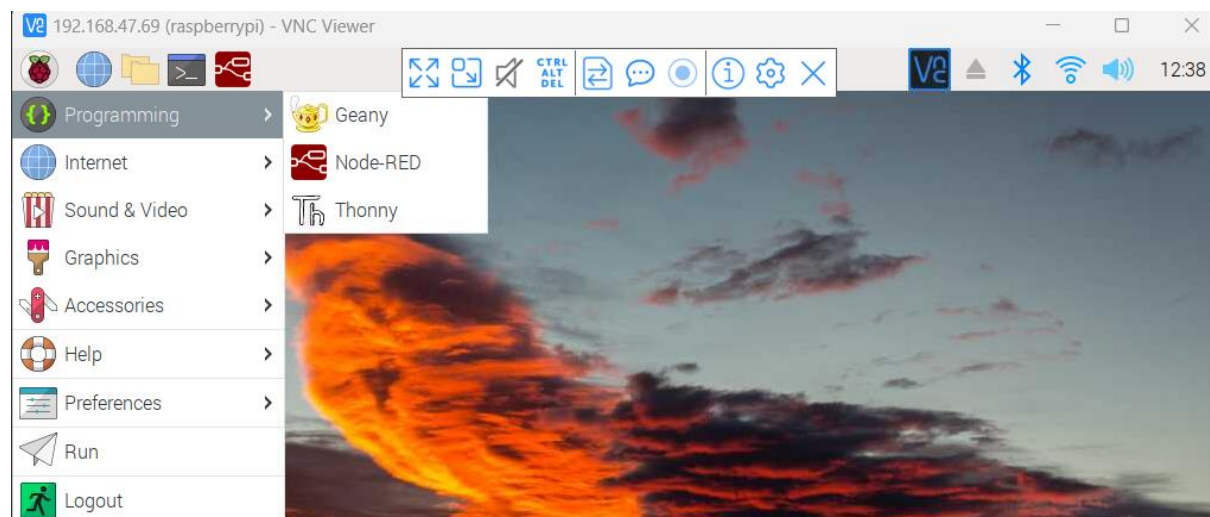
Node-RED.3 - Node-RED เพื่ออ่าน DHT-22 Sensor

โปรแกรมที่ใช้ทดสอบ



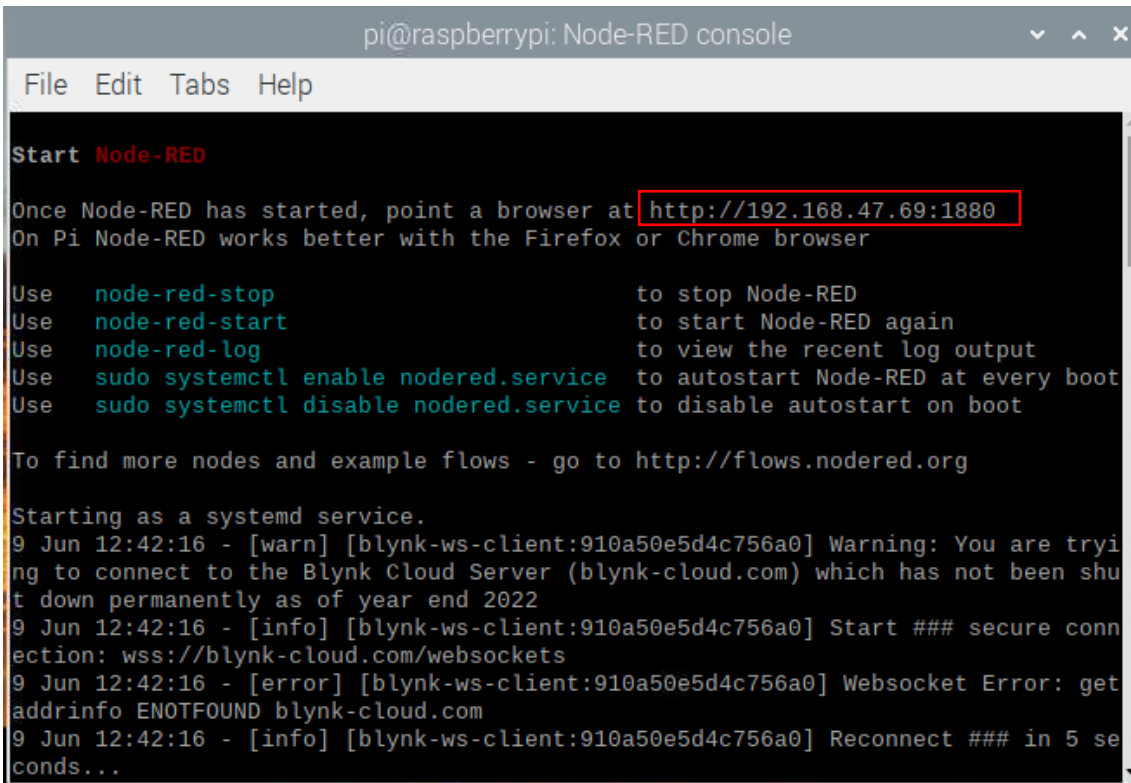
เข้าไปที่ > Recommended Software ไปที่ช่องค้นหาแล้วพิมพ์ node-red แล้ว ติดตั้ง Node-RED

เมื่อติดตั้งเสร็จแล้วให้เปิด Node-Red โดยไปที่ Programming -> Node-RED



เมื่อเข้าไปที่ Node-RED  Node-RED

ให้นำ link ในกรอบสีแดงไปใส่ใน web browser เพื่อเข้าสู่ Node-Red



```

pi@raspberrypi: Node-RED console
File Edit Tabs Help

Start Node-RED

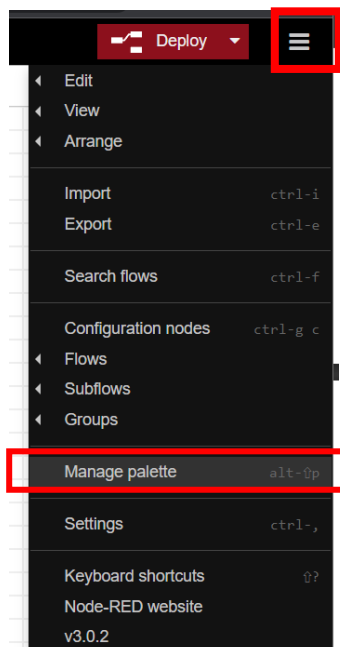
Once Node-RED has started, point a browser at http://192.168.47.69:1880
On Pi Node-RED works better with the Firefox or Chrome browser

Use node-red-stop to stop Node-RED
Use node-red-start to start Node-RED again
Use node-red-log to view the recent log output
Use sudo systemctl enable nodered.service to autostart Node-RED at every boot
Use sudo systemctl disable nodered.service to disable autostart on boot

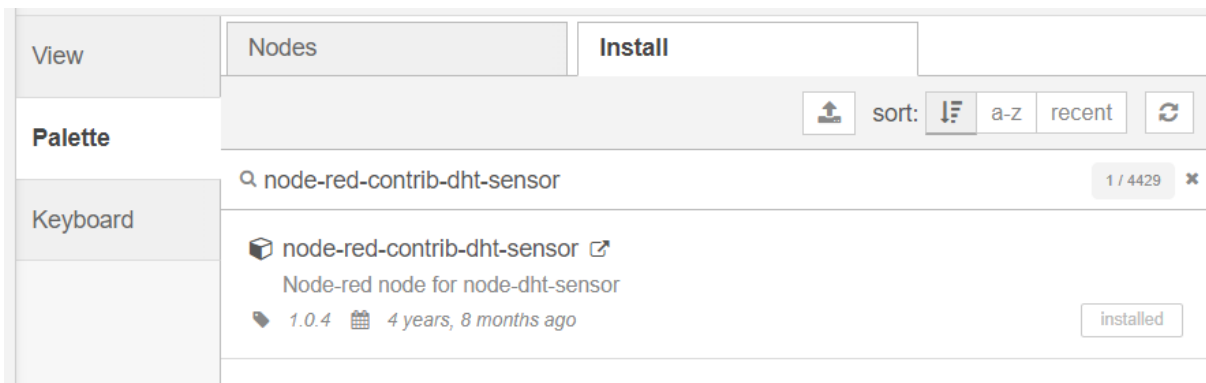
To find more nodes and example flows - go to http://flows.nodered.org

Starting as a systemd service.
9 Jun 12:42:16 - [warn] [blynk-ws-client:910a50e5d4c756a0] Warning: You are trying to connect to the Blynk Cloud Server (blynk-cloud.com) which has not been shut down permanently as of year end 2022
9 Jun 12:42:16 - [info] [blynk-ws-client:910a50e5d4c756a0] Start ### secure connection: wss://blynk-cloud.com/websockets
9 Jun 12:42:16 - [error] [blynk-ws-client:910a50e5d4c756a0] Websocket Error: getaddrinfo ENOTFOUND blynk-cloud.com
9 Jun 12:42:16 - [info] [blynk-ws-client:910a50e5d4c756a0] Reconnect ### in 5 seconds...
  
```

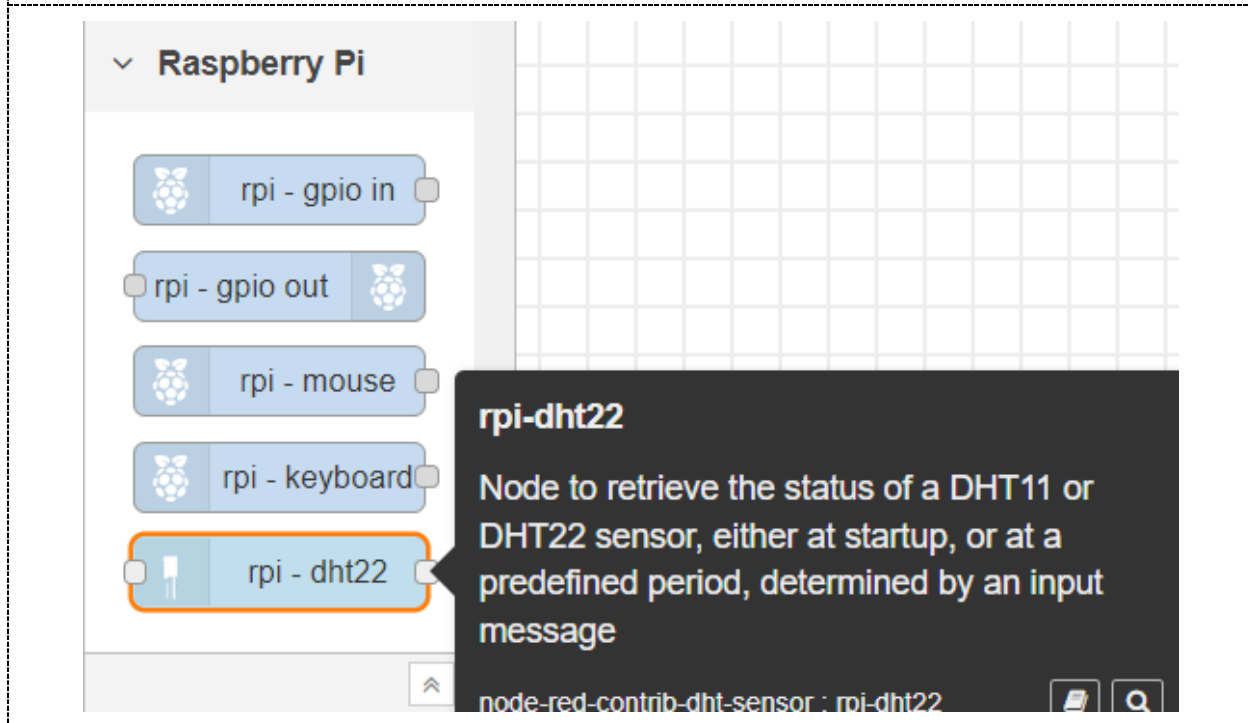
เมื่อเข้ามาที่ Node-RED ตามลิ้งค์ URI → ติดตั้ง palette สำหรับ DHT22 กดไปที่ ด้านบนขวา



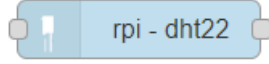
ไปที่ Install -> ค้นหา node-red-contrib-dht-sensor แล้วกดติดตั้ง



เมื่อติดตั้งเสร็จที่ tab ด้านซ้ายจะได้ node ใหม่เพิ่มเข้ามา



รูป Code Capture – เลือก



แล้วตั้งค่าตามนี้ จากนั้นกด Done

Edit rpi-dht22 node

Delete Cancel Done

Properties

Topic rpi-dht22

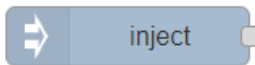
Sensor model DHT22

Pin numbering BCM GPIO

Pin number 11

Name Name

รูป Code Capture – เลือก



แล้วตั้งค่าตามนี้ จากนั้นกด Done

Edit inject node

Delete Cancel Done

Properties

Name Name

msg. payload = timestamp

msg. topic = a_z

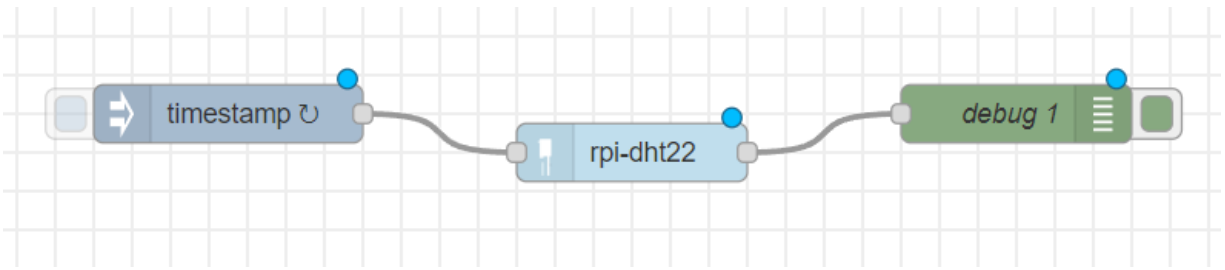
+ add inject now

☐ Inject once after 0.1 seconds, then

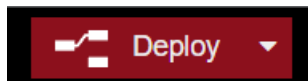
Repeat interval

every 30 seconds

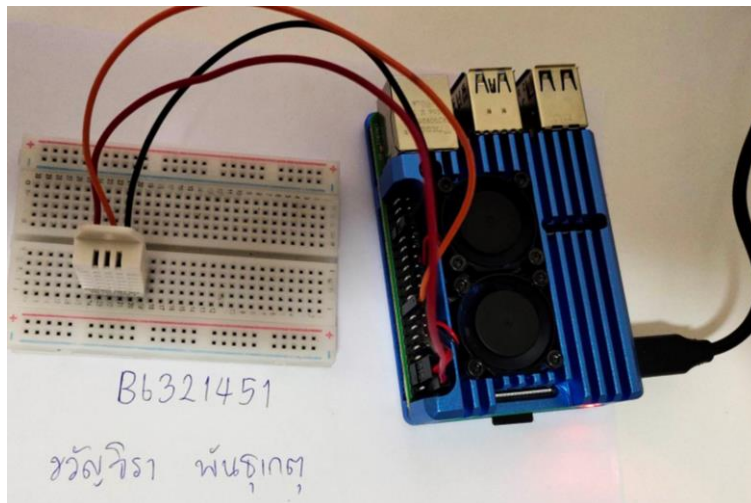
รูป Code Capture – เลือก ☐ debug แล้วนำทุก node มาต่อตามนี้

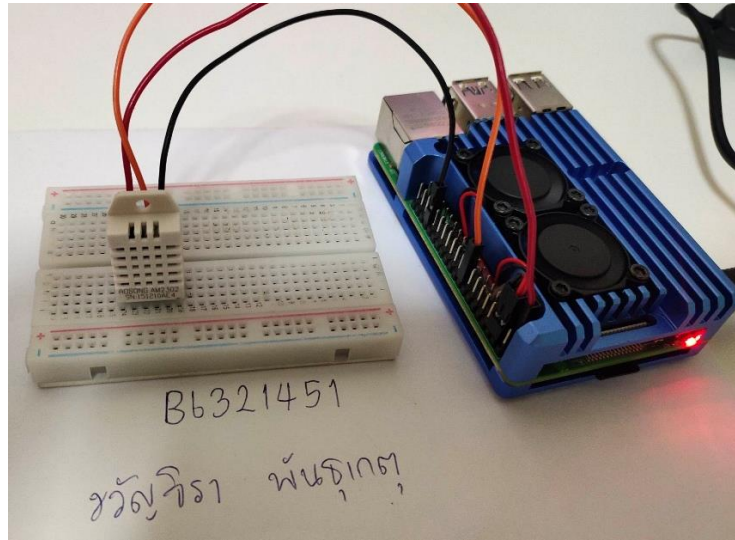


จากนั้นก็กด



รูปวงจร





ติดปัญหา DHT อ่านอุณหภูมิไม่ได้

The screenshot shows a Node-RED flow diagram with three nodes: 'timestamp', 'rpi-dht22', and 'msg.payload'. The flow is as follows: 'timestamp' → 'rpi-dht22' → 'msg.payload'.

The console output shows the following messages:

```
6/10/2023, 11:03:33 PM node: msg.payload  
rpi-dht22 : msg.payload : string[4]  
"0.00"  
6/10/2023, 11:04:03 PM node: msg.payload  
rpi-dht22 : msg.payload : string[4]  
"0.00"  
6/10/2023, 11:04:33 PM node: msg.payload  
rpi-dht22 : msg.payload : string[4]  
"0.00"
```