การสร้าง MQTT Server บน Raspberry Pi เพื่อใช้งาน Chatbot LINE ในฟาร์มอัจฉริยะ
Chatbot LINE from Raspberry Pi MQTT Server for Smart Farming

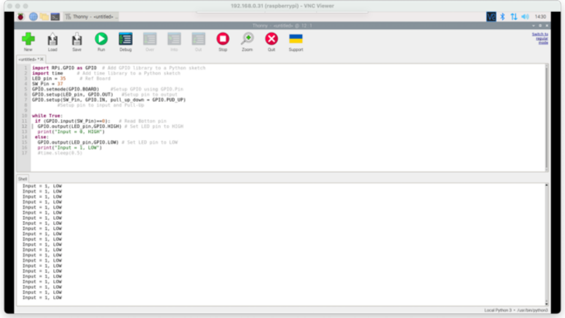
ขื่อ-สกุล : B6304577 นายภานุพงศ์ แคนอินทร์

6/6 – คำถามท้ายบทเพื่อทดสอบความเข้าใจ

Quiz_101 – ทดสอบ RPi4 GPIO with Python

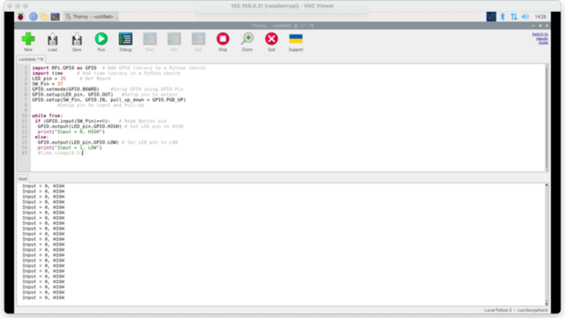
Python.1 - Python Switch control LED >> กดติด ปล่อยดับ

โปรแกรมที่ใช้ทดสอบ



```
import RPi.GPIO as GPIO  # Add GPIO library to a Python sketch
import time      # Add time library to a Python sketch
LED_pin = 35      # Ref Board
SW_Pin = 37
GPIO.setmode(GPIO.BOARD)    #Setup GPIO using GPIO.Pin
GPIO.setup(LED_pin, GPIO.OUT)   #Setup pin to output
GPIO.setup(SW_Pin, GPIO.IN, pull_up_down = GPIO.PUD_UP)
         #Setup pin to input and Pull-Up

while True:
 if (GPIO.input(SW_Pin)==0):   # Read Botton pin
  GPIO.output(LED_pin,GPIO.HIGH) # Set LED pin to HIGH
  print("Input = 0, HIGH")
 else:
  GPIO.output(LED_pin,GPIO.LOW) # Set LED pin to LOW
  print("Input = 1, LOW")
  #time.sleep(0.5)
```

```
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
Input = 1, LOW
```



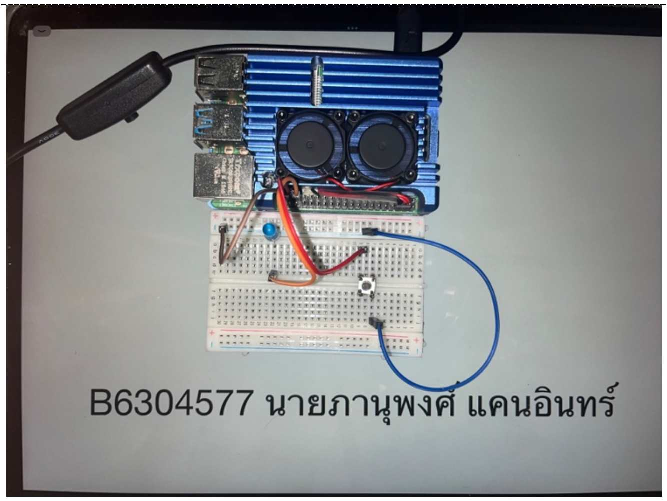
```
import RPi.GPIO as GPIO  # Add GPIO library to a Python sketch
import time      # Add time library to a Python sketch
LED_pin = 35      # Ref Board
SW_Pin = 37
GPIO.setmode(GPIO.BOARD)    #Setup GPIO using GPIO.Pin
GPIO.setup(LED_pin, GPIO.OUT)   #Setup pin to output
GPIO.setup(SW_Pin, GPIO.IN, pull_up_down = GPIO.PUD_UP)
         #Setup pin to input and Pull-Up

while True:
 if (GPIO.input(SW_Pin)==0):   # Read Botton pin
  GPIO.output(LED_pin,GPIO.HIGH) # Set LED pin to HIGH
  print("Input = 0, HIGH")
 else:
  GPIO.output(LED_pin,GPIO.LOW) # Set LED pin to LOW
  print("Input = 1, LOW")
  #time.sleep(0.5)
```

```
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
Input = 0, HIGH
```

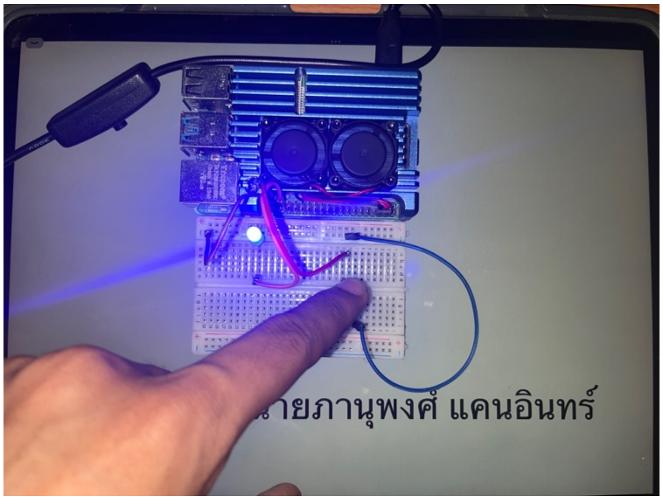
รูป Code Capture

```
import RPi.GPIO as GPIO  # Add GPIO library to a Python sketch
import time     # Add time library to a Python sketch
LED_pin = 35     # Ref Board
SW_Pin = 37
GPIO.setmode(GPIO.BOARD)    #Setup GPIO using GPIO.Pin
GPIO.setup(LED_pin, GPIO.OUT)   #Setup pin to output
GPIO.setup(SW_Pin, GPIO.IN, pull_up_down = GPIO.PUD_UP)
         #Setup pin to input and Pull-Up

while True:
 if (GPIO.input(SW_Pin)==0):   # Read Botton pin
  GPIO.output(LED_pin,GPIO.HIGH) # Set LED pin to HIGH
  print("Input = 0, HIGH")
 else:
  GPIO.output(LED_pin,GPIO.LOW) # Set LED pin to LOW
  print("Input = 1, LOW")
  #time.sleep(0.5)
```

รูปการทดสอบ 1

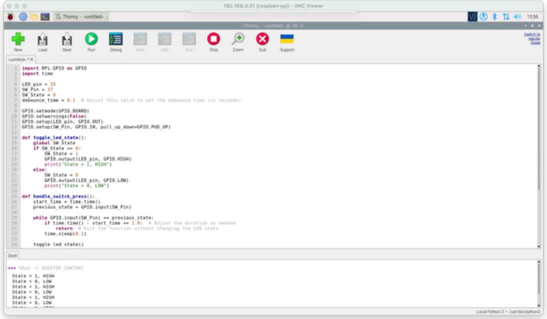


รูปการทดสอบ 2



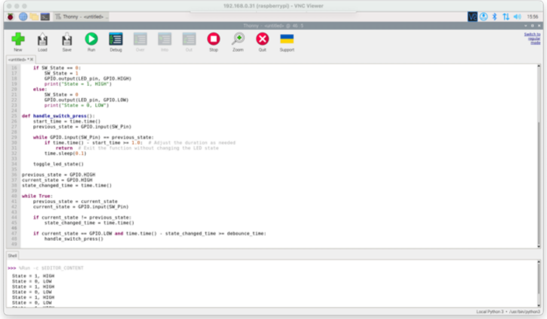
Python.2 - Python Switch control LED >> กดติด กดดับ

โปรแกรมที่ใช้ทดสอบ



```python
import RPi.GPIO as GPIO
import time

LED_pin = 35
SW_Pin = 37
SW_State = 0
debounce_time = 0.1  # Adjust this value to set the debounce time (in seconds)

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup(LED_pin, GPIO.OUT)
GPIO.setup(SW_Pin, GPIO.IN, pull_up_down=GPIO.PUD_UP)

def toggle_led_state():
    global SW_State
    if SW_State == 0:
        SW_State = 1
        GPIO.output(LED_pin, GPIO.HIGH)
        print("State = 1, HIGH")
    else:
        SW_State = 0
        GPIO.output(LED_pin, GPIO.LOW)
        print("State = 0, LOW")

def handle_switch_press():
    start_time = time.time()
    previous_state = GPIO.input(SW_Pin)

    while GPIO.input(SW_Pin) == previous_state:
        if time.time() - start_time >= 1.0:  # Adjust the duration as needed
            return  # Exit the function without changing the LED state
        time.sleep(0.1)

    toggle_led_state()
```

```
>>> %Run -c $EDITOR_CONTENT
  State = 1, HIGH
  State = 0, LOW
  State = 1, HIGH
  State = 0, LOW
  State = 1, HIGH
  State = 0, LOW
```



```python
    if SW_State == 0:
        SW_State = 1
        GPIO.output(LED_pin, GPIO.HIGH)
        print("State = 1, HIGH")
    else:
        SW_State = 0
        GPIO.output(LED_pin, GPIO.LOW)
        print("State = 0, LOW")

def handle_switch_press():
    start_time = time.time()
    previous_state = GPIO.input(SW_Pin)

    while GPIO.input(SW_Pin) == previous_state:
        if time.time() - start_time >= 1.0:  # Adjust the duration as needed
            return  # Exit the function without changing the LED state
        time.sleep(0.1)

    toggle_led_state()

previous_state = GPIO.HIGH
current_state = GPIO.HIGH
state_changed_time = time.time()

while True:
    previous_state = current_state
    current_state = GPIO.input(SW_Pin)

    if current_state != previous_state:
        state_changed_time = time.time()

    if current_state == GPIO.LOW and time.time() - state_changed_time >= debounce_time:
        handle_switch_press()
```

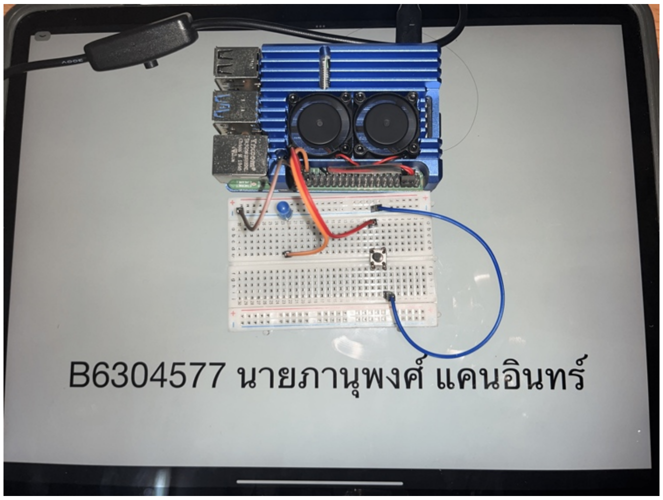
```
>>> %Run -c $EDITOR_CONTENT
  State = 1, HIGH
  State = 0, LOW
  State = 1, HIGH
  State = 0, LOW
  State = 1, HIGH
  State = 0, LOW
```

รูป Code Capture

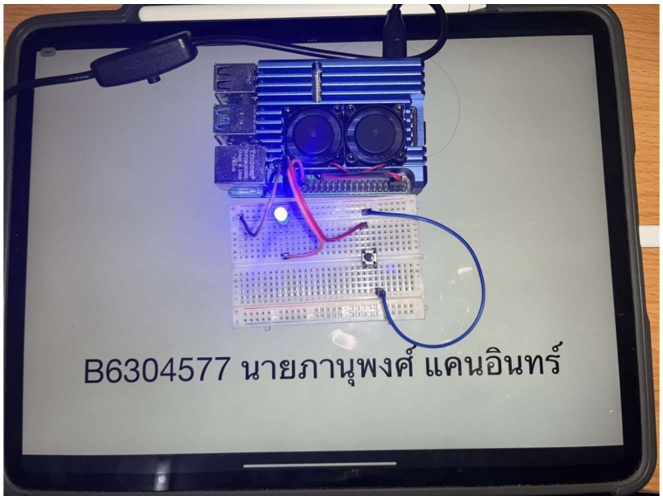
```
import RPi.GPIO as GPIO
import time

LED_pin = 35
SW_Pin = 37
SW_State = 0
debounce_time = 0.1  # Adjust this value to set the debounce time (in seconds)

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup(LED_pin, GPIO.OUT)
GPIO.setup(SW_Pin, GPIO.IN, pull_up_down=GPIO.PUD_UP)

def toggle_led_state():
    global SW_State
    if SW_State == 0:
        SW_State = 1
        GPIO.output(LED_pin, GPIO.HIGH)
        print("State = 1, HIGH")
    else:
        SW_State = 0
        GPIO.output(LED_pin, GPIO.LOW)
        print("State = 0, LOW")

def handle_switch_press():
    start_time = time.time()
    previous_state = GPIO.input(SW_Pin)

    while GPIO.input(SW_Pin) == previous_state:
        if time.time() - start_time >= 1.0:  # Adjust the duration as needed
            return  # Exit the function without changing the LED state
        time.sleep(0.1)

    toggle_led_state()

previous_state = GPIO.HIGH
current_state = GPIO.HIGH
state_changed_time = time.time()
```

```
while True:
    previous_state = current_state
    current_state = GPIO.input(SW_Pin)

    if current_state != previous_state:
        state_changed_time = time.time()

    if current_state == GPIO.LOW and time.time() - state_changed_time >= debounce_time:
        handle_switch_press()
```

รูปการทดสอบ 1

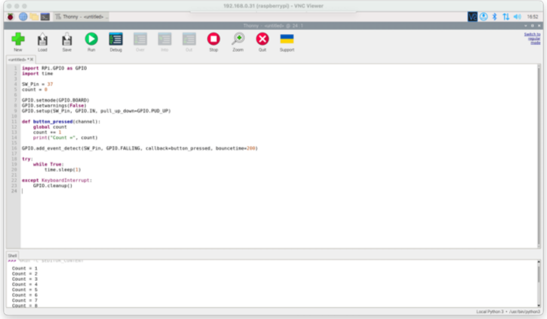


รูปการทดสอบ 2



POython.3 - Python Switch >> Switch Counter

โปรแกรมที่ใช้ทดสอบ



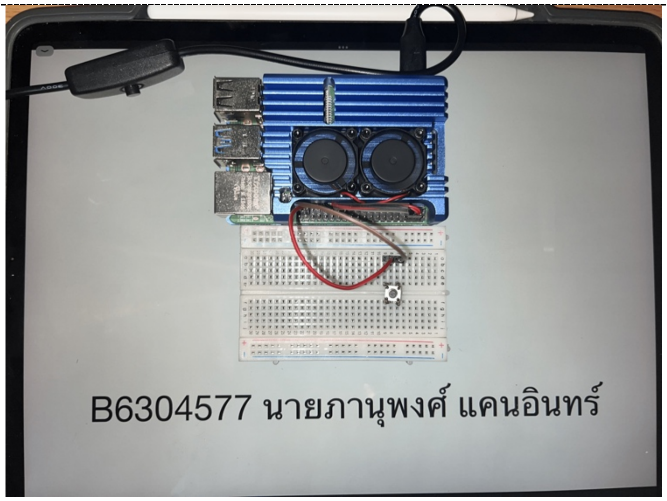
```
import RPi.GPIO as GPIO
import time

SW_Pin = 37
count = 0

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup(SW_Pin, GPIO.IN, pull_up_down=GPIO.PUD_UP)

def button_pressed(channel):
    global count
    count += 1
    print("Count =", count)

GPIO.add_event_detect(SW_Pin, GPIO.FALLING, callback=button_pressed, bouncetime=200)

try:
    while True:
        time.sleep(1)

except KeyboardInterrupt:
    GPIO.cleanup()
```

```
>>> %Run -c $EDITOR_CONTENT
 Count = 1
 Count = 2
 Count = 3
 Count = 4
 Count = 5
 Count = 6
 Count = 7
 Count = 8
```

รูป Code Capture

```
import RPi.GPIO as GPIO
import time

SW_Pin = 37
count = 0

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup(SW_Pin, GPIO.IN, pull_up_down=GPIO.PUD_UP)

def button_pressed(channel):
    global count
    count += 1
    print("Count =", count)

GPIO.add_event_detect(SW_Pin, GPIO.FALLING, callback=button_pressed, bouncetime=200)

try:
    while True:
        time.sleep(1)

except KeyboardInterrupt:
    GPIO.cleanup()
```

รูปการทดสอบ 1

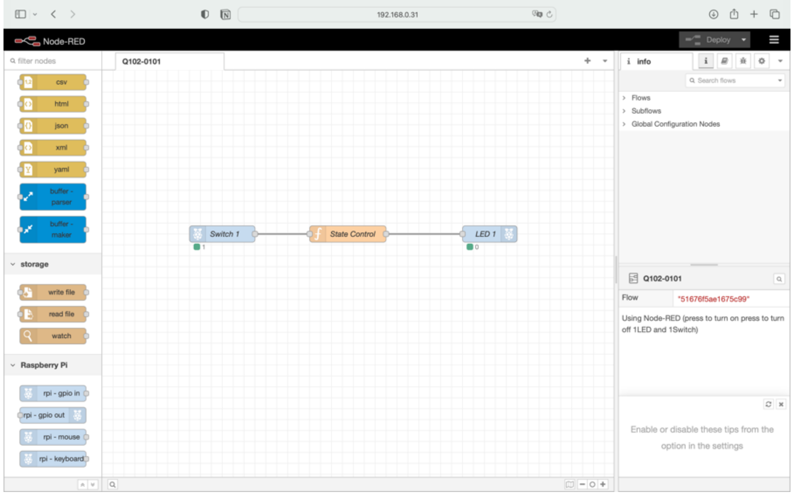


รูปการทดสอบ 2

Quiz_102 – ทดสอบ RPi4 GPIO with Node-RED

Node-RED.1 – Node-RED เพื่อควบคุมสวิตซ์กดแบบ กดติด กดดับ {Switch-LED 1 คู่}

โปรแกรมที่ใช้ทดสอบ



Flow "51676f5ae1675c99"

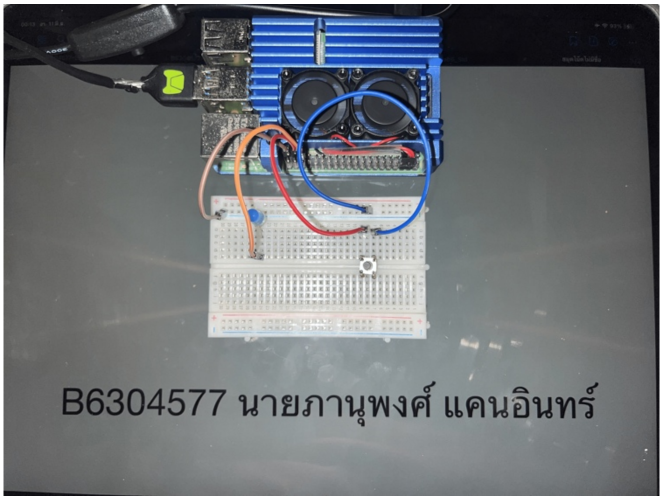
Using Node-RED (press to turn on press to turn off 1LED and 1Switch)

Enable or disable these tips from the option in the settings

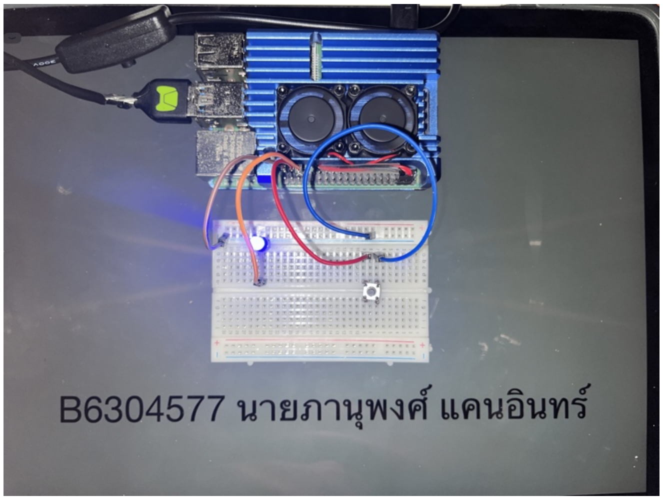
รูป Code Capture

```
[
    {
        "id": "51676f5ae1675c99",
        "type": "tab",
        "label": "Q102-0101",
        "disabled": false,
        "info": "Using Node-RED (press to turn on press to turn off 1LED and 1Switch) ",
        "env": []
    },
    {
        "id": "eec1119ce2081360",
        "type": "rpi-gpio in",
        "z": "51676f5ae1675c99",
        "name": "Switch 1",
        "pin": "26",
        "intype": "up",
        "debounce": "25",
        "read": false,
        "bcm": true,
        "x": 220,
        "y": 300,
        "wires": [
            [
                "016c3e9be89901e3"
            ]
        ]
    },
    {
        "id": "06cb6823737d2dfc",
        "type": "rpi-gpio out",
        "z": "51676f5ae1675c99",
        "name": "LED 1",
        "pin": "19",
        "set": "",
        "level": "0",
        "freq": "",
        "out": "out",
        "bcm": true,
        "x": 710,
        "y": 300,
        "wires": []
    },
    {
        "id": "016c3e9be89901e3",
        "type": "function",
        "z": "51676f5ae1675c99",
        "name": "State Control",
        "func": "context.state = context.state | false;\ncontext.state = 
!context.state\n\nvar myContext = context.state;\nvar count = 
context.get(\"count\")||0;\ncount += 1;\ncontext.set(\"count\",count);\nmsg.count = 
count;\n\nfunction isOdd(num) { \n    return num % 2;\n}\n\nif(myContext === true && 
isOdd((count+1)/2) ===1){\n    msg.payload = 1;\n    return msg;\n} else if (myContext === 
true && isOdd((count+1)/2) ===0){\n    msg.payload = 0;\n    return msg;\n}",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 450,
        "y": 300,
        "wires": [
            [
                "06cb6823737d2dfc"
            ]
        ]
    }
]
```

รูปการทดสอบ 1

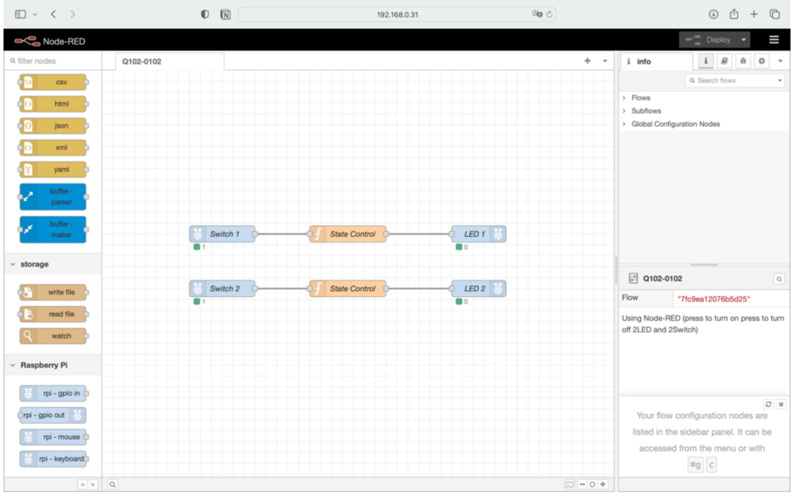


รูปการทดสอบ 2



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

โปรแกรมที่ใช้ทดสอบ



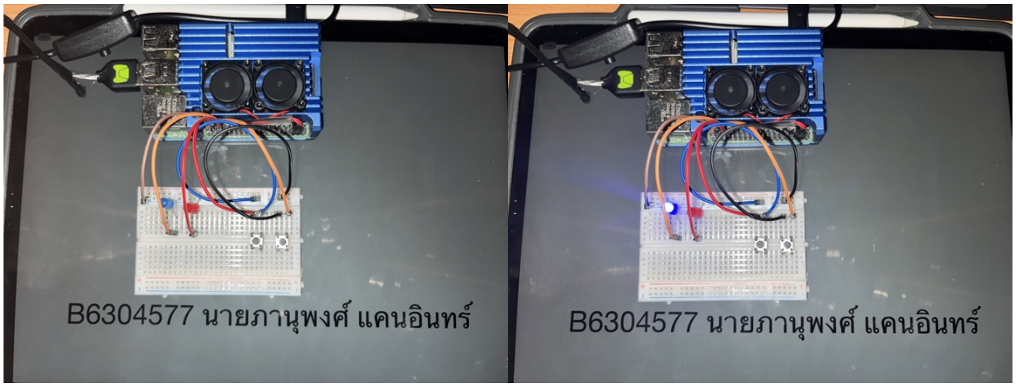
Using Node-RED (press to turn on press to turn off 2LED and 2Switch)

Your flow configuration nodes are listed in the sidebar panel. It can be accessed from the menu or with

รูป Code Capture

```
[
    {
        "id": "7fc9ea12076b5d25",
        "type": "tab",
        "label": "Q102-0102",
        "disabled": false,
        "info": "Using Node-RED (press to turn on press to turn off 2LED and 2Switch) ",
        "env": []
    },
    {
        "id": "5e0e47b70cfa15b3",
        "type": "rpi-gpio in",
        "z": "7fc9ea12076b5d25",
        "name": "Switch 1",
        "pin": "26",
        "intype": "up",
        "debounce": "25",
        "read": false,
        "bcm": true,
        "x": 220,
        "y": 300,
        "wires": [
            [
                "e9a1afc0ccd95ad2"
            ]
        ]
    },
    {
        "id": "ece0b86753b4806e",
        "type": "rpi-gpio out",
        "z": "7fc9ea12076b5d25",
        "name": "LED 1",
        "pin": "19",
        "set": "",
        "level": "0",
        "freq": "",
        "out": "out",
        "bcm": true,
        "x": 690,
        "y": 300,
        "wires": []
    },
    {
        "id": "e9a1afc0ccd95ad2",
        "type": "function",
        "z": "7fc9ea12076b5d25",
        "name": "State Control",
        "func": "context.state = context.state | false;\ncontext.state = 
!context.state\n\nvar myContext = context.state;\nvar count = 
context.get(\"count\")||0;\ncount += 1;\ncontext.set(\"count\",count);\nmsg.count = 
count;\n\nfunction isOdd(num) { \n    return num % 2;\n}\n\nif(myContext === true && 
isOdd((count+1)/2) ===1){\n    msg.payload = 1;\n    return msg;\n} else if (myContext === 
true && isOdd((count+1)/2) ===0){\n    msg.payload = 0;\n    return msg;\n}",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 450,
        "y": 300,
        "wires": [
            [
                "ece0b86753b4806e"
            ]
        ]
    },
    {
        "id": "0c4853c44002d61b",
        "type": "rpi-gpio out",
        "z": "7fc9ea12076b5d25",
        "name": "LED 2",
        "pin": "16",
        "set": "",
        "level": "0",
        "freq": "",
        "out": "out",
        "bcm": true,
        "x": 690,
        "y": 400,
        "wires": []
    },
    {
        "id": "6efb796920ac0f04",
        "type": "rpi-gpio in",
        "z": "7fc9ea12076b5d25",
        "name": "Switch 2",
        "pin": "21",
        "intype": "up",
        "debounce": "25",
        "read": false,
        "bcm": true,
        "x": 220,
        "y": 400,
        "wires": [
            [
                "aad96eb4d550ef39"
            ]
        ]
    },
    {
        "id": "aad96eb4d550ef39",
        "type": "function",
        "z": "7fc9ea12076b5d25",
        "name": "State Control",
        "func": "context.state = context.state | false;\ncontext.state = !context.state\nvar myContext = context.state;\nvar count = context.get(\"count\")||0;\ncount += 1;\ncontext.set(\"count\",count);\nmsg.count = count;\nfunction isOdd(num) { return num % 2;}\n\nif(myContext === true && isOdd((count+1)/2) ===1){\n    msg.payload = 1;\n    return msg;\n} else if (myContext === true && isOdd((count+1)/2) ===0){\n    msg.payload = 0;\n    return msg;\n}",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 450,
        "y": 400,
        "wires": [
            [
                "0c4853c44002d61b"
            ]
        ]
    }
]
```

รูปการทดสอบ 1

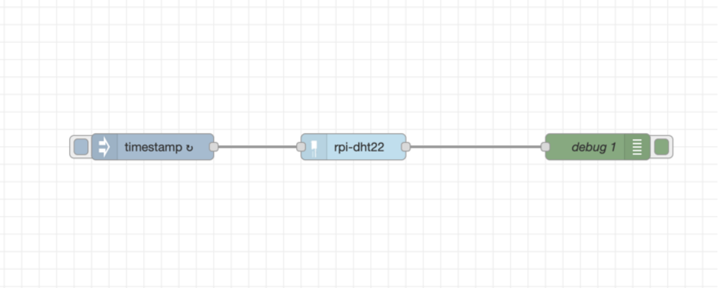

B6304577 นายภานุพงศ์ แคนอินทร์
B6304577 นายภานุพงศ์ แคนอินทร์

รูปการทดสอบ 2

B6304577 นายภานุพงศ์ แคนอินทร์
B6304577 นายภานุพงศ์ แคนอินทร์

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

โปรแกรมที่ใช้ทดสอบ

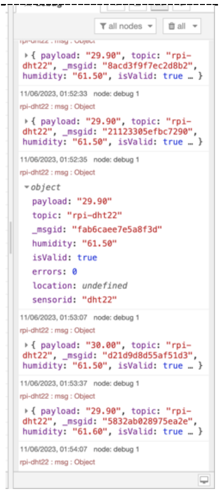


รูป Code Capture

```
[
    {
        "id": "d6b35559496318d0",
        "type": "tab",
        "label": "Q102-0301",
        "disabled": false,
        "info": "",
        "env": []
    },
    {
        "id": "bcfc5be92bd00b6f",
        "type": "rpi-dht22",
        "z": "d6b35559496318d0",
        "name": "",
        "topic": "rpi-dht22",
        "dht": 22,
        "pintype": "0",
        "pin": "4",
        "x": 440,
        "y": 260,
        "wires": [
            [
                "ae17d2026071014f"
            ]
        ]
    },
    {
        "id": "1ff0c6a3eade9b3c",
        "type": "inject",
        "z": "d6b35559496318d0",
        "name": "",
        "props": [
            {
                "p": "payload"
            },
            {
                "p": "topic",
                "vt": "str"
            }
        ],
        "repeat": "30",
        "crontab": "",
        "once": false,
        "onceDelay": 0.1,
        "topic": "",
        "payload": "",
        "payloadType": "date",
        "x": 210,
        "y": 260,
        "wires": [
            [
                "bcfc5be92bd00b6f"
            ]
        ]
    },
    {
        "id": "ae17d2026071014f",
        "type": "debug",
        "z": "d6b35559496318d0",
        "name": "debug 1",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "true",
        "targetType": "full",
        "statusVal": "",
        "statusType": "auto",
        "x": 720,
        "y": 260,
        "wires": []
    }
]
```

รูปการทดสอบ 1

รูปการทดสอบ 2



```
{ payload: "29.90", topic: "rpi-dht22", _msgid: "8acd3f9f7ec2d8b2", humidity: "61.50", isValid: true … }

11/06/2023, 01:52:33   node: debug 1
rpi-dht22 : msg : Object
{ payload: "29.90", topic: "rpi-dht22", _msgid: "21123305efbc7290", humidity: "61.50", isValid: true … }

11/06/2023, 01:52:35   node: debug 1
rpi-dht22 : msg : Object
object
  payload: "29.90"
  topic: "rpi-dht22"
  _msgid: "fab6caee7e5a8f3d"
  humidity: "61.50"
  isValid: true
  errors: 0
  location: undefined
  sensorid: "dht22"

11/06/2023, 01:53:07   node: debug 1
rpi-dht22 : msg : Object
{ payload: "30.00", topic: "rpi-dht22", _msgid: "d21d9d8d55af51d3", humidity: "61.50", isValid: true … }

11/06/2023, 01:53:37   node: debug 1
rpi-dht22 : msg : Object
{ payload: "29.90", topic: "rpi-dht22", _msgid: "5832ab028975ea2e", humidity: "61.60", isValid: true … }

11/06/2023, 01:54:07   node: debug 1
rpi-dht22 : msg : Object
```