

คู่มือการฝึกอบรม  
**NODE-RED**

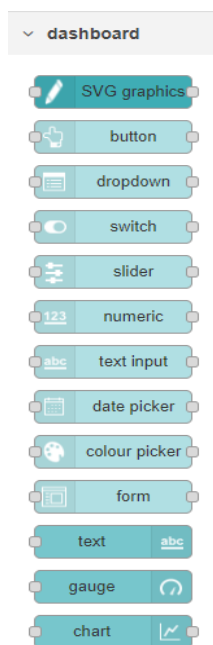


**Node red communication IOT level3**

**NODE-RED DASHBOARD**

## การใช้งาน Dashboard เบื้องต้น

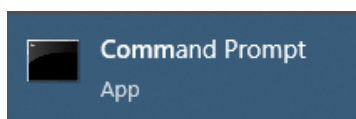
ในโปรแกรม Node red ผู้ใช้งานสามารถสร้างหน้า Dashboard เป็นหน้าเว็บสำหรับการสร้างอินเตอร์เฟซควบคุมสั่งการหรือแสดงข้อมูลสถานะการทำงานในระบบได้ โดยวิธีการใช้งานให้โหลดส่วนเสริม Palette Dashboard เพิ่มเข้ามาในโปรแกรม



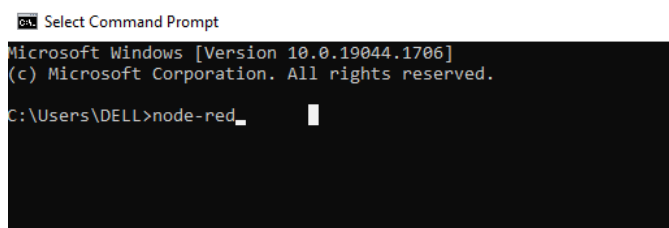
## การเพิ่มส่วนเสริม Dashboard เข้ามาในโปรแกรม Node red

### 1. เปิดโปรแกรม Node red

- กดปุ่ม Windows เปิด Command (CMD) ขึ้นมา



- พิมพ์ node-red ใน Command จากนั้นกด Enter



- หน้า CMD แสดงการเปิดใช้งาน Node red

```
6 Jun 08:45:40 - [info] Node-RED version: v2.2.2
6 Jun 08:45:40 - [info] Node.js version: v16.15.1
6 Jun 08:45:40 - [info] Windows_NT 10.0.19044 x64 LE
6 Jun 08:45:41 - [info] Loading palette nodes
6 Jun 08:45:42 - [info] Settings file : C:\Users\DELL\.node-red\settings.js
6 Jun 08:45:42 - [info] Context store : 'default' [module=memory]
6 Jun 08:45:42 - [info] User directory : \Users\DELL\.node-red
6 Jun 08:45:42 - [warn] Projects disabled : editorTheme.projects.enabled=false
6 Jun 08:45:42 - [info] Flows file : \Users\DELL\.node-red\flows.json
6 Jun 08:45:42 - [info] Server now running at http://127.0.0.1:1880/
6 Jun 08:45:42 - [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.
-----

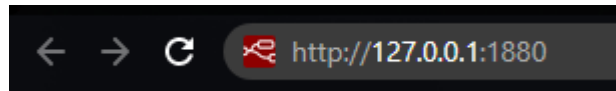
6 Jun 08:45:42 - [info] Starting flows
6 Jun 08:45:42 - [info] Started flows
6 Jun 08:45:42 - [info] [mqtt-broker:ddf764380ded56cc] Connection failed to broker: 6adc437c-808a-4c80-93ee-b5cefe0b889a
@mqtt://mqtt.netpie.io:1883
```

- เปิดหน้าเว็บเบราว์เซอร์ (Edge, Chrome) จากนั้นให้พิมพ์ URL ตามข้อความด้านล่างนี้

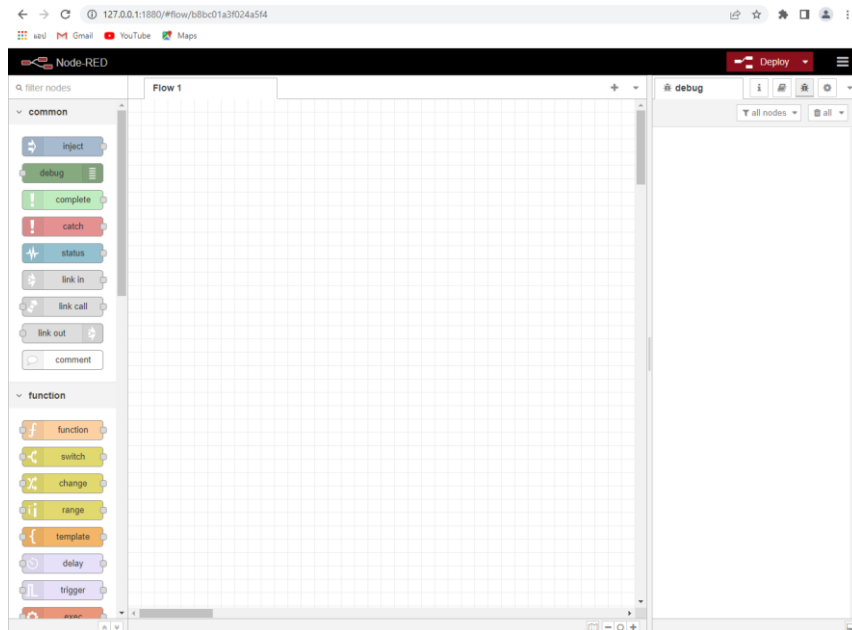
“

<http://127.0.0.1:1880>

”

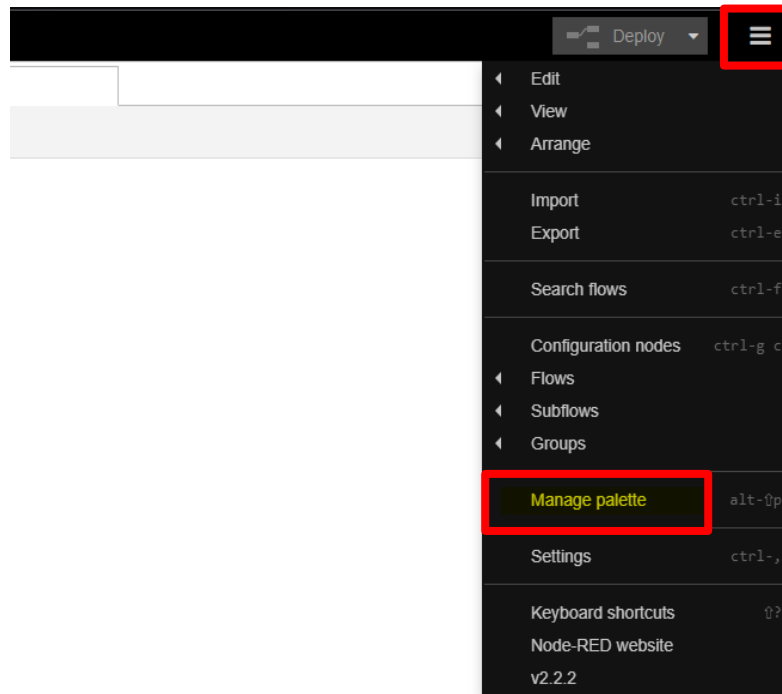


- หน้าเริ่มต้นของโปรแกรม Node red

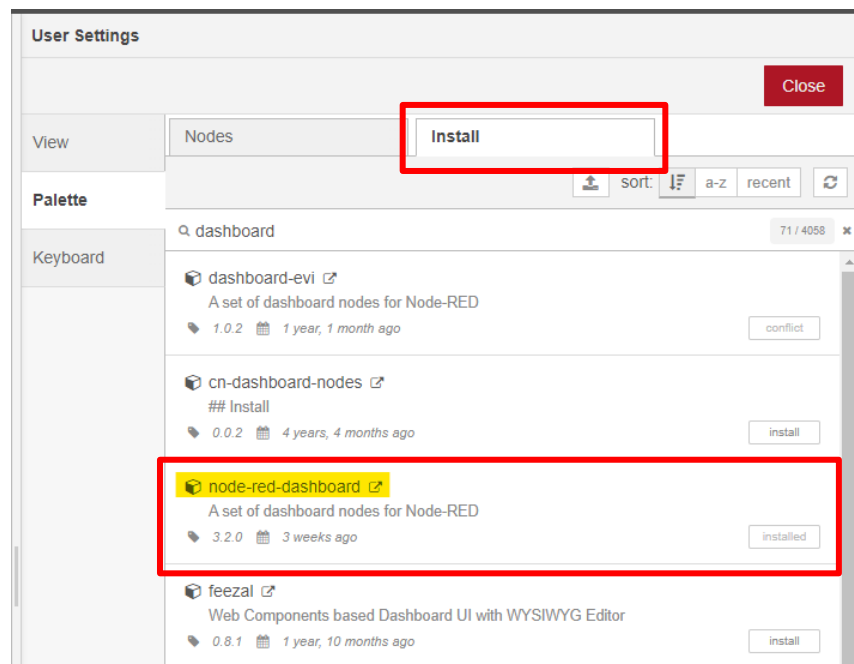


## 2. ติดตั้งส่วนเสริม Dashboard

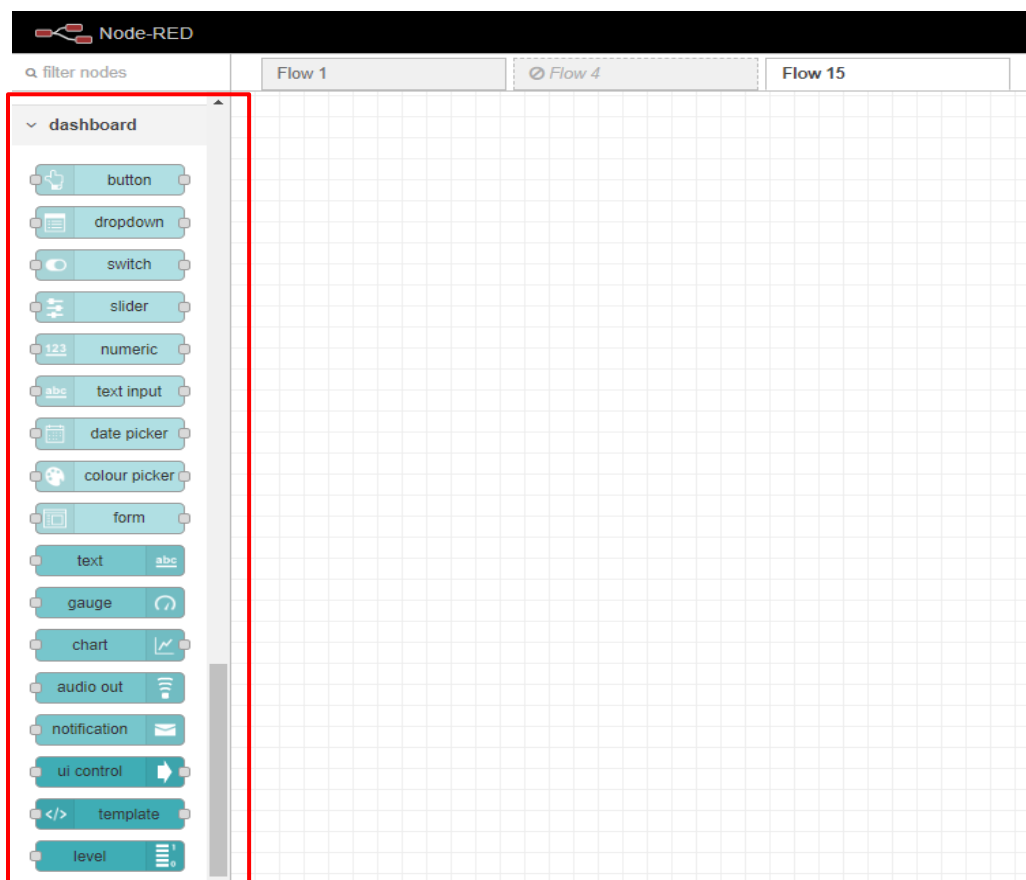
- เข้าไปที่ Manage Palette Node



- ค้นหา Dashboard จากนั้นให้ติดตั้ง node-red-dashboard

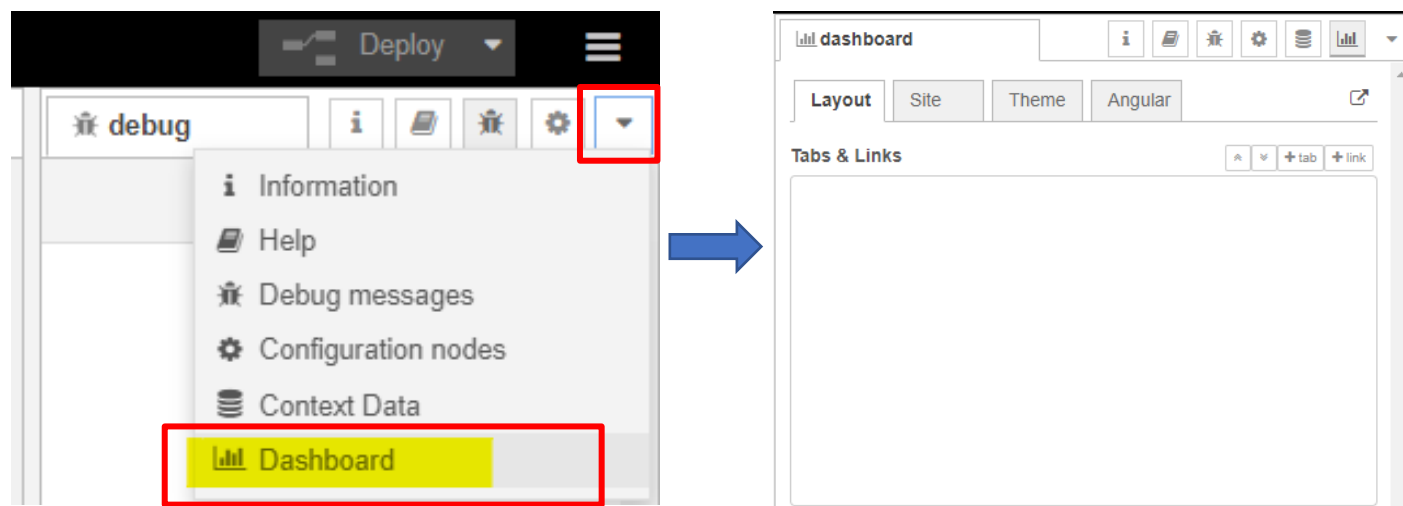


- ตรวจสอบ Node Dashboard ที่เพิ่มเข้ามาใหม่ เลื่อนลงมาด้านล่าง สังเกตหัวข้อ Dashboard

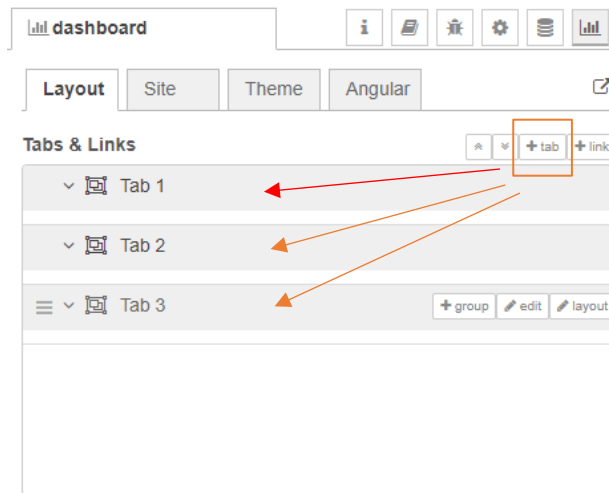


### 3. การเพิ่มหัวข้อหลักหน้า Dashboard ก่อนการใช้งาน

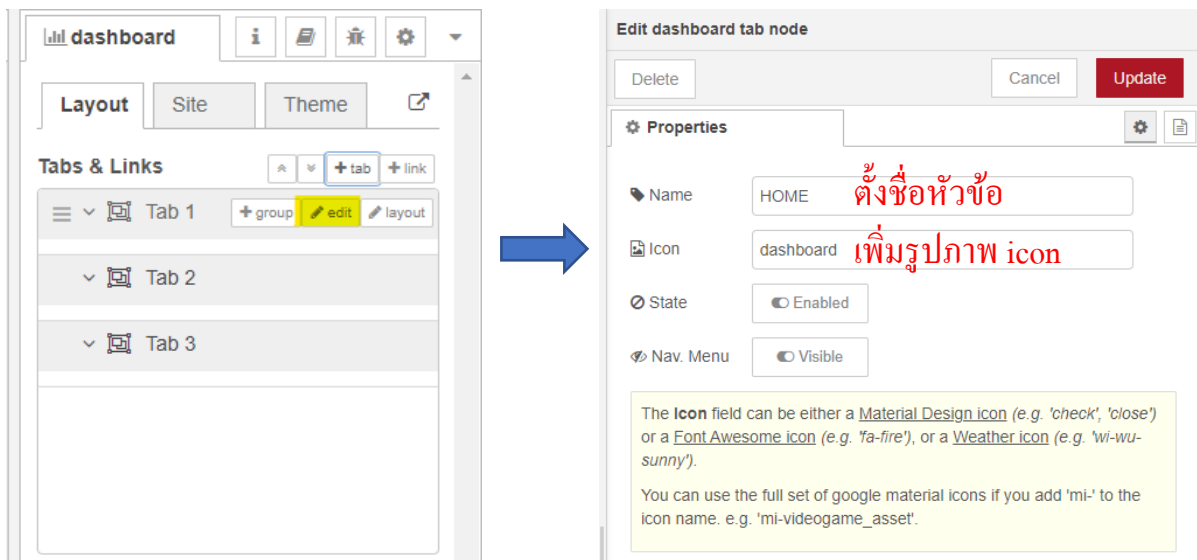
ในการจะใช้งานหน้า Dashboard จะต้องทำการเพิ่มหน้า Tap หัวข้อให้กับ Node Dashboard ก่อน เพื่อเป็นการระบุที่อยู่ของกราฟฟิคที่ผู้ใช้งานสร้างขึ้นว่าต้องการให้ไปอยู่ตำแหน่งใดในหน้า Dashboard



- เพิ่มหัวข้อ Tap ในแต่ละ Tap จะเป็นชื่อหัวข้อหลักในหน้าจอ Dashboard



- สามารถเข้าไปแก้ไขหัวข้อได้ใน edit



- การเพิ่มรูปภาพ icon ในหัวข้อ tab

เปิดหน้าเว็บเบราว์เซอร์ (Edge, Chrome) จากนั้นให้ค้นหา <https://fontawesome.com/v4/icons/>




ในเว็บจะมีรูป icon ให้ใช้งานมากมาย สามารถกดเลือกนำมาใช้แสดงในหน้า Dashboard ได้

- เลือก icon home มาใช้งานในหัวข้อแรก (สามารถกดค้นหาได้ด้วยคำสั่ง CTRL+F )

hard-of-hearing (alias)

♥ heart

 home

⌚ hourglass-2 (alias)

⌚ hourglass-o

- กดคลิกเข้าไปใน home



After you get **up and running**, you can place Font Awesome icons just about anywhere with the `<i>` tag:

fa-home

## ชื่อที่ใช้ระบุภาพใน ICON

- Copy fa-home ไปใส่ใน icon dashboard เมื่อตั้งค่าเสร็จแล้วให้กด Update

### Edit dashboard tab node

DeleteCancelUpdate

Properties

Name

HOME

Icon

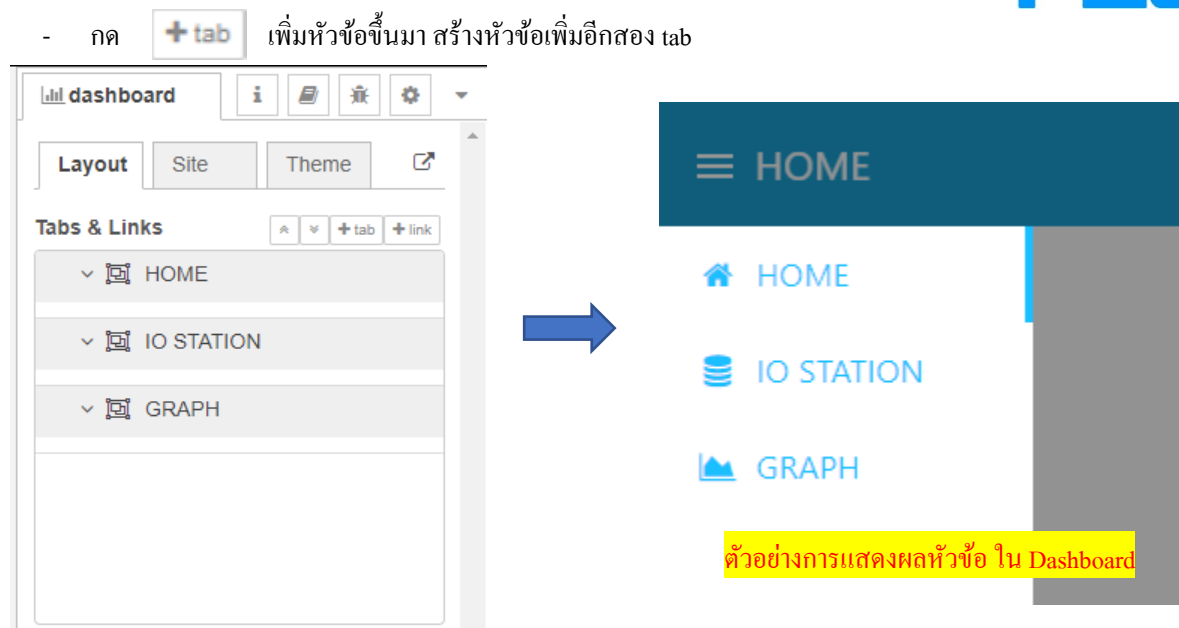
fa-home

State

☒ Enabled

Nav. Menu

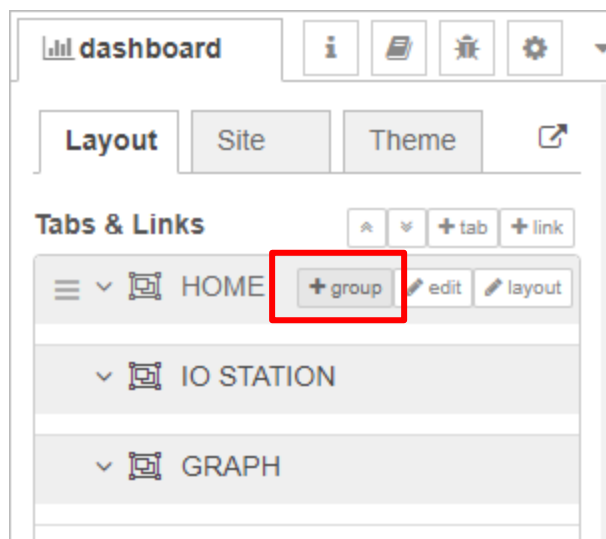
☒ Visible



\*\*การเปิดหน้า Dashboard ได้นั้นจำเป็นต้องมีการใช้งาน Dashboard node ในโปรแกรม node red ก่อนจึงจะสามารถแสดงผลได้

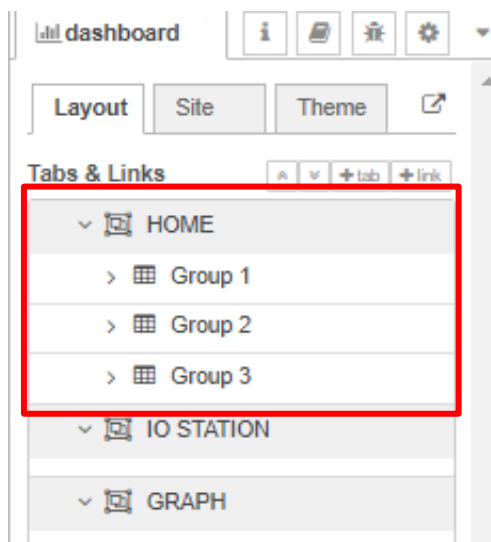
#### 4. การเพิ่ม Group ภายในหัวข้อ Tab

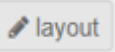
ภายในแต่ละหัวข้อใน Dashboard จะมีการแบ่ง Group หัวข้อด้านในออกเป็นบล็อก เราสามารถเพิ่ม Group ได้ โดยการเลื่อนเมาส์เข้าไปที่หัวข้อหลักจากนั้นกด **+ group**



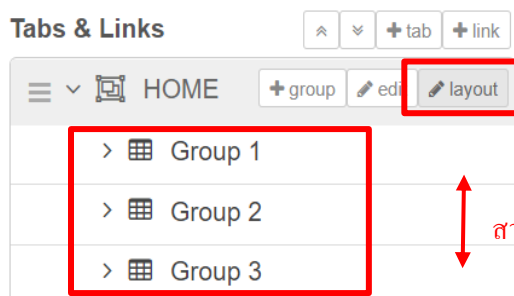


- เพิ่ม Group ใน HOME ออกเป็น 3 บล็อก

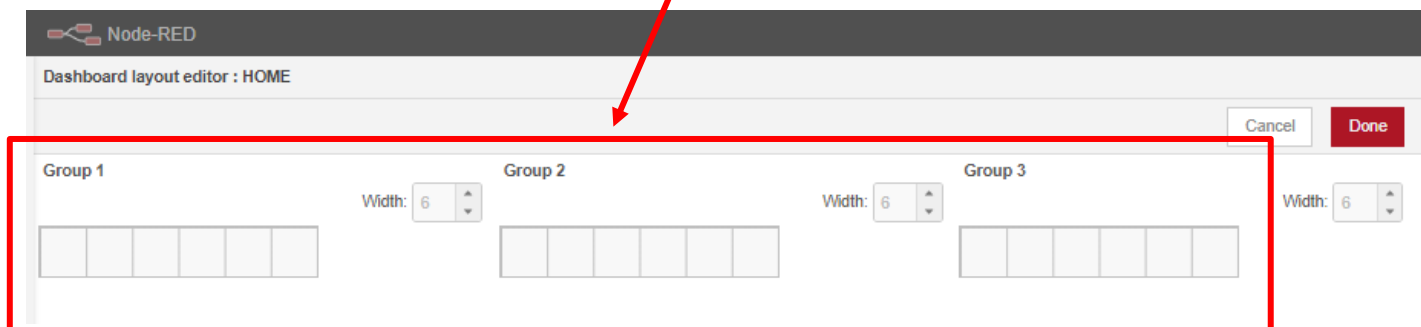



การแสดงผล Dashboard node red จะแบ่งการแสดงผลเป็นบล็อก โดยหัวข้อที่อยู่บนสุดจะอยู่ด้านซ้ายของหน้า Dashboard เสมอสามารถ  
การจัดเรียง Group ได้โดยการเลื่อนเมาส์เข้าไปที่หัวข้อหลัก จากนั้นกด 

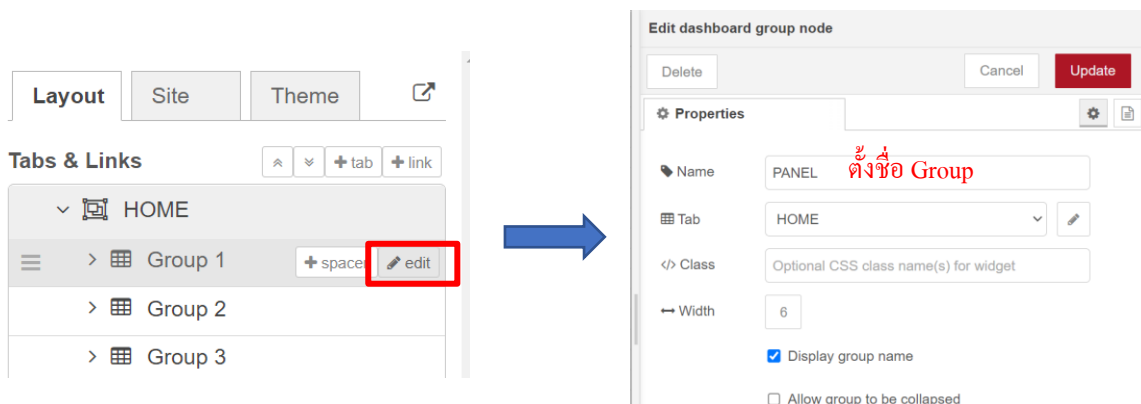
สามารถคลิกหัวข้อ Group ขึ้นหรือลงได้ เพื่อใช้จัดเรียงหน้า Dashboard หัวข้อที่อยู่บนสุดจะอยู่ด้านซ้าย



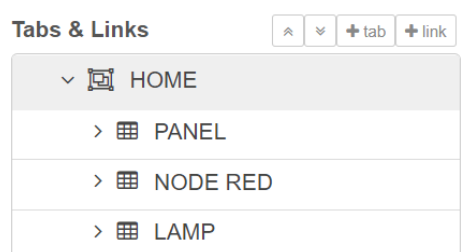
สามารถคลิกค้างย้าย Group เพื่อเรียงหน้า dashboard



- แก้ไขชื่อ Group ได้ด้วยการเลื่อนเมาส์ไปที่ Group ที่ต้องการแก้ไข จากนั้นกด 



- ตั้งชื่อใหม่ให้กับ Group ทั้งหมด



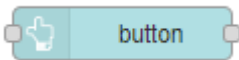
\*\*กรณีที่ Group นั้น ไม่ได้ถูกเลือกใช้งานจาก Dashboard Node จะทำให้ Dashboard บนหน้าเว็บ ไม่แสดงผลชื่อ Group นั้นขณะใช้งาน

## Dashboard Node

เป็นส่วนเสริมที่มีความสามารถในการสร้างปุ่มกดหรือการแสดงผลบนหน้าจอ Dashboard แล้วนำมาควบคุมการสั่งการ Node อื่นใน Node red ได้ทำให้ผู้ใช้งานสามารถควบคุมสั่งการ หรือดูสถานะของข้อมูลผ่านหน้าจอ Dashboard เพิ่มรูปแบบการใช้งานที่สะดวกและง่ายมากยิ่งขึ้น

**\*\*จำเป็นต้องมีการใช้งาน Dashboard Node อันใดอันหนึ่งขึ้นมาก่อนจึงจะสามารถแสดงผลการใช้งานบนหน้า Dashboard ได้**

### 1. Button node



ปุ่มกดสั่งงานในหน้า Dashboard สามารถตั้งค่าข้อมูลที่ในส่งได้ การจะสั่งงานจำพวก Dashboard Node จะต้องไปตั้งในหน้าเว็บ UI จึงจะสามารถสั่งงานได้

- เพิ่ม Button node กดเข้าไปใน node เพื่อตั้งค่าการสั่งงาน

Delete
Cancel
Done

Properties

Group
Add new dashboard group...

Size
auto
กำหนดขนาดของปุ่ม

Icon
optional icon
กำหนดรูป icon

Label
button
ตั้งชื่อ node

Tooltip
optional tooltip

Color
optional text/icon color
ตั้งค่าสีตัวอักษร (HTML Color Codes)

Background
optional background color
ตั้งค่าสีปุ่ม (HTML Color Codes)

When clicked, send:

Payload
a-z
ตั้งค่า output ที่จะส่งออกไปขณะปุ่มกดทำงาน

Topic
msg. topic

If msg arrives on input, emulate a button click:
☐

Name
Name

## - เพิ่มที่อยู่ Group

เมื่อเราตั้งค่า Group เอาไว้แล้วสามารถกด

Add new dashboard group...

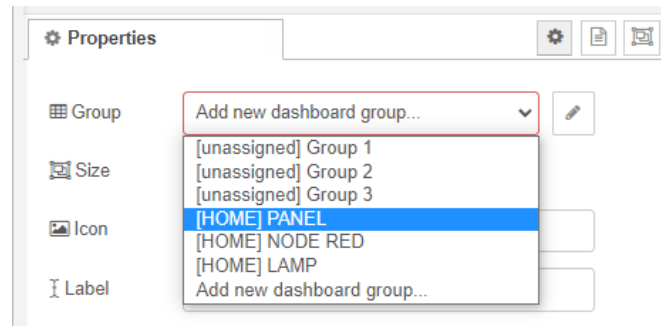


เพื่อหา Group ที่เราสร้างเอาไว้ หรือในกรณีที่

ยังไม่ได้สร้าง Group สามารถสร้างใหม่ในหัวข้อนี้ได้โดยการกด



เพื่อสร้าง Group ใหม่ขึ้นมา



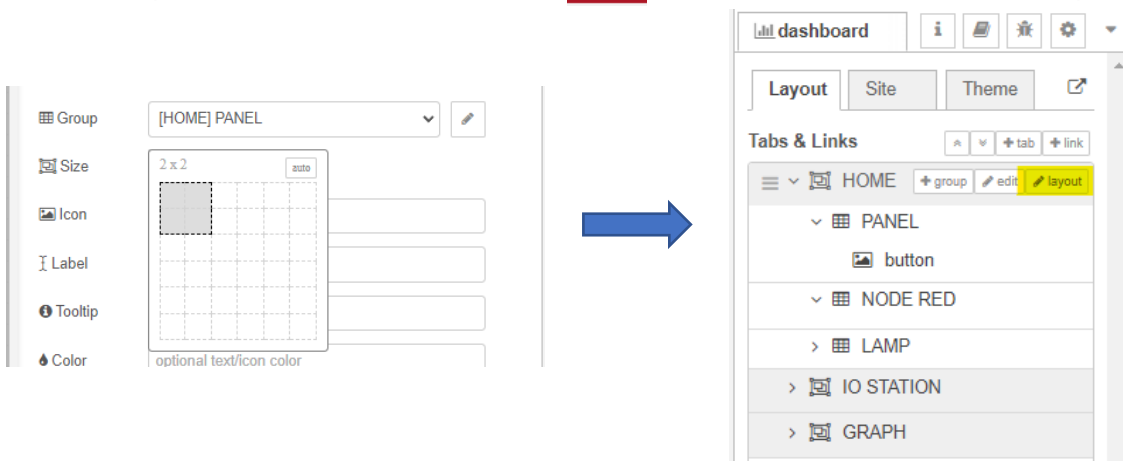
## - การกำหนดขนาดของปุ่มกด (Size)

Dashboard Node red จะกำหนดขนาดเป็นช่องสี่เหลี่ยม สามารถตรวจสอบขนาดได้ใน layout ของ Group นั้น

ทดสอบขนาดของปุ่ม โดย สร้างบล็อกขนาด 2x2 จากนั้นกด

Done

ยืนยันการตั้งค่า สามารถตรวจสอบขนาดปุ่มกดได้ใน layout

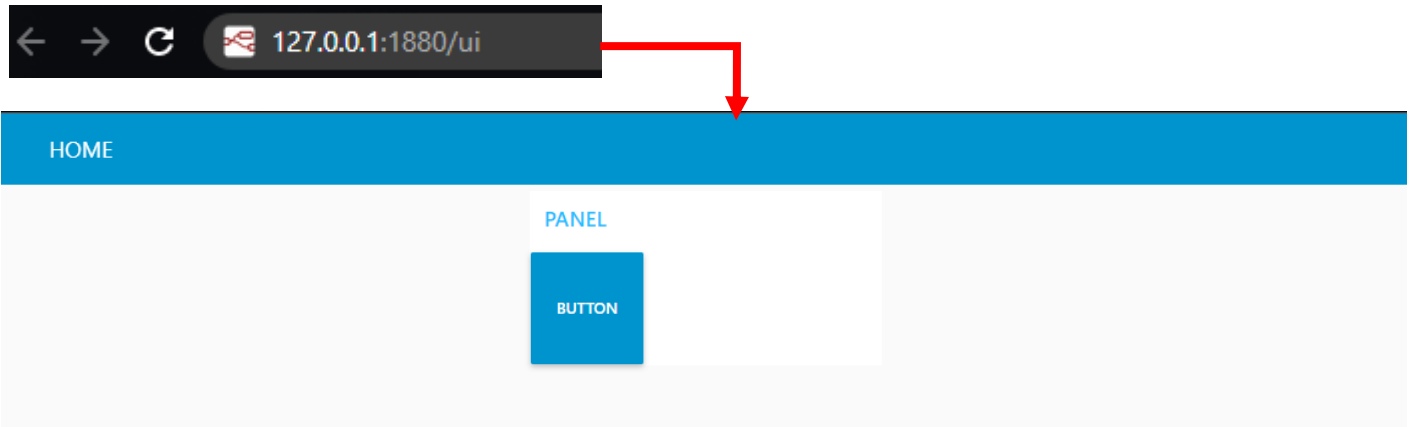


## - ตรวจสอบขนาดของปุ่มกดในหน้า layout



- การเปิดหน้า UI Dashboard

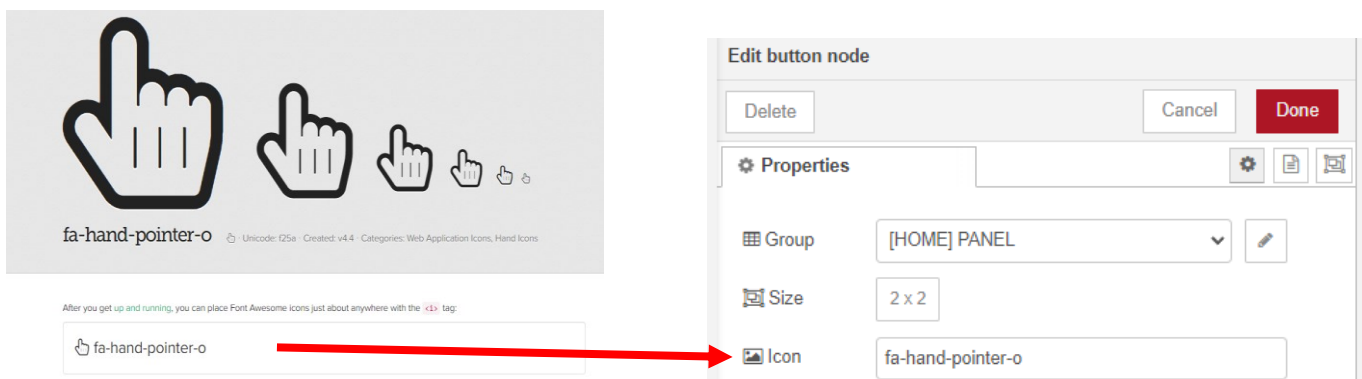
ดาวน์โหลดโปรแกรม node red เมื่อผู้ใช้งานได้สร้าง Button node แล้วเพิ่มที่อยู่ Group เรียบร้อย จะสามารถเข้าใช้งานหน้าเว็บ Dashboard ได้ โดยการเข้าไปที่เว็บไซต์ <http://127.0.0.1:1880/ui>



ในหน้า Dashboard จะเห็นได้ว่าตำแหน่งปุ่มอยู่ตรงกลางหน้าจอเนื่องจากตัว Node red Dashboard จะตั้งให้ Group ตัวแรกอยู่ตำแหน่งกลางเสมอ ยกเว้นมีการใช้งาน Group ที่มากขึ้น จึงจะมีการเรียง Group กันตามการตั้งค่า layout ตัวที่อยู่หัวข้อบนสุดจะอยู่ด้านซ้าย

- การใส่รูป Icon

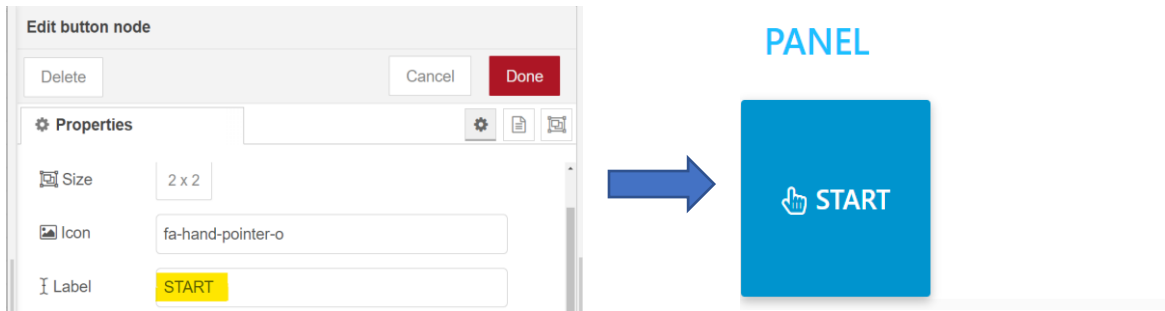
เข้าไปที่โปรแกรม node red เพิ่มรูป icon ใน Button node สามารถเข้าไปหา icon ได้ที่เว็บไซต์ <https://fontawesome.com/v4/icons/>



ดาวน์โหลดโปรแกรม Node red จากนั้นเข้าไปที่หน้า UI Dashboard เพื่อดูความเปลี่ยนแปลง

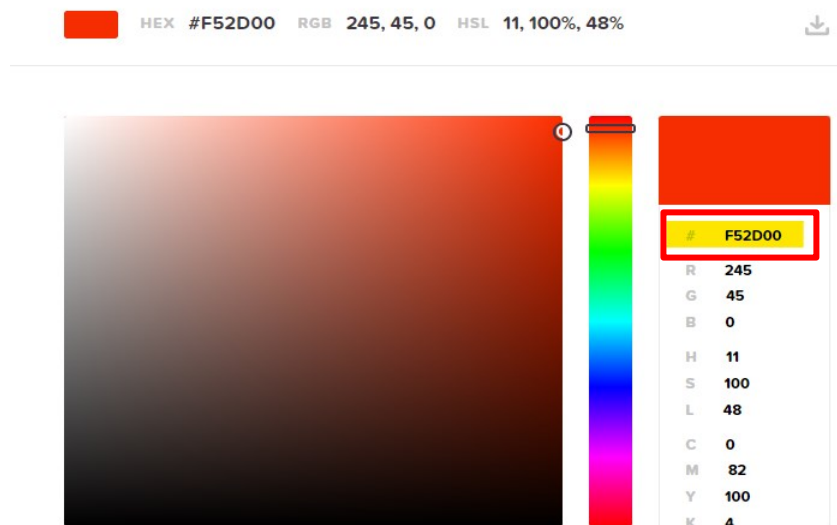


- การตั้งชื่อปุ่ม Label เปลี่ยนแปลงชื่อปุ่มใน UI Dashboard



- การตั้งค่าสี UI Dashboard

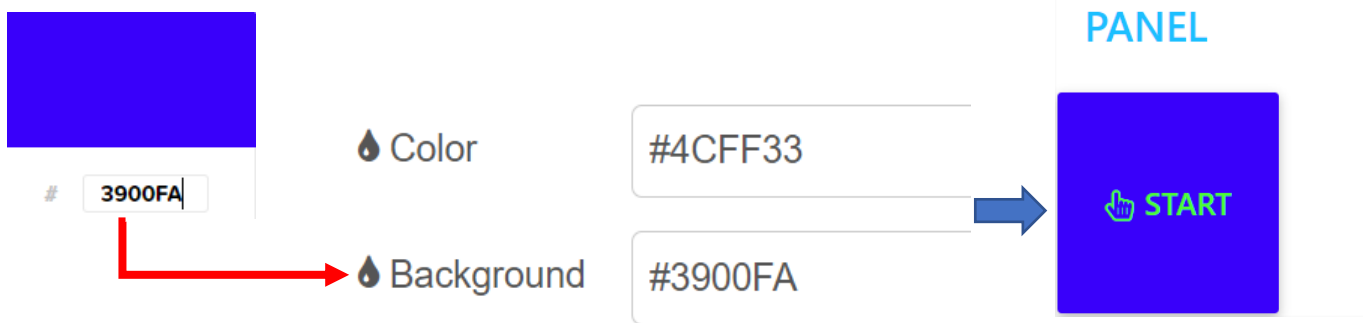
การเปลี่ยนสีของ UI Dashboard จะอ้างอิงโค้ดสี HTML color codes (#XXXXXX) สามารถตรวจสอบโค้ดสีได้ที่เว็บไซต์ <https://htmlcolorcodes.com/>



ทดสอบการเปลี่ยนสีตัวอักษร

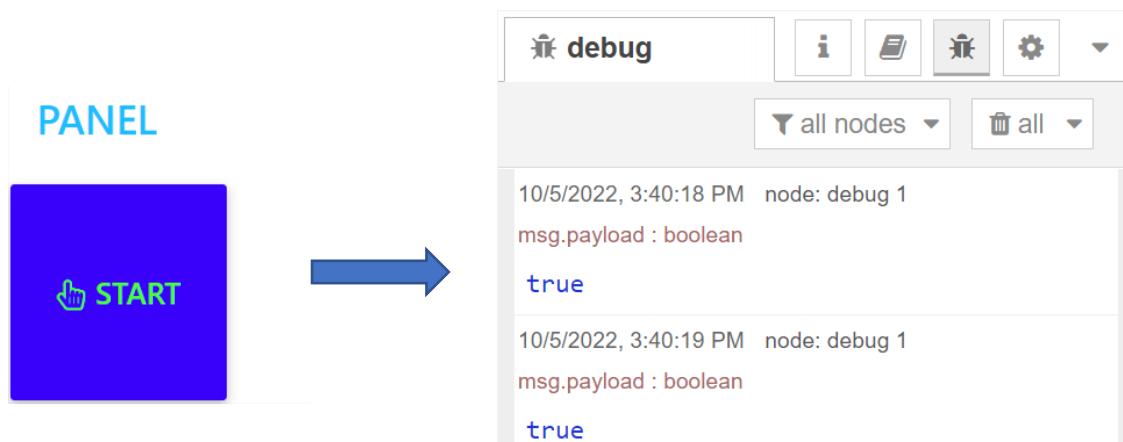
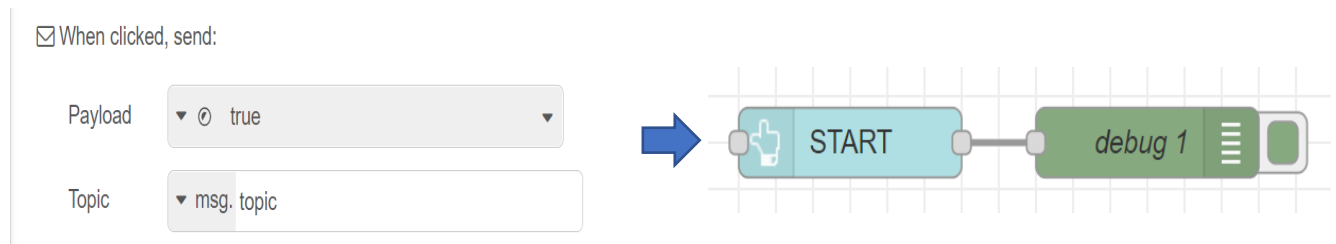


## ทดสอบการเปลี่ยนสีปุ่มกด



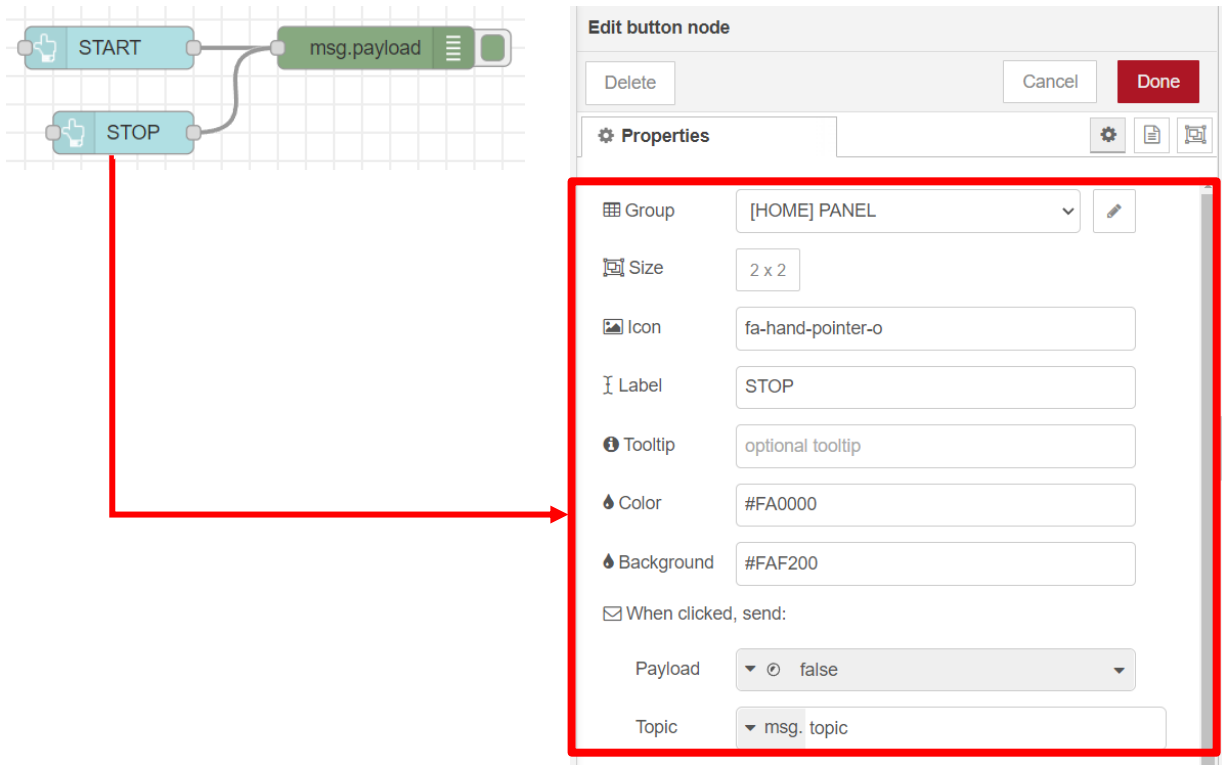
- ตั้งค่า output ที่จะส่งออกไปขณะกดปุ่มทำงาน

กำหนดค่า Payload ทดสอบการส่งค่า Boolean = True จากนั้นเชื่อมต่อ output กับ debug node ทดสอบการส่งค่ามายัง debug

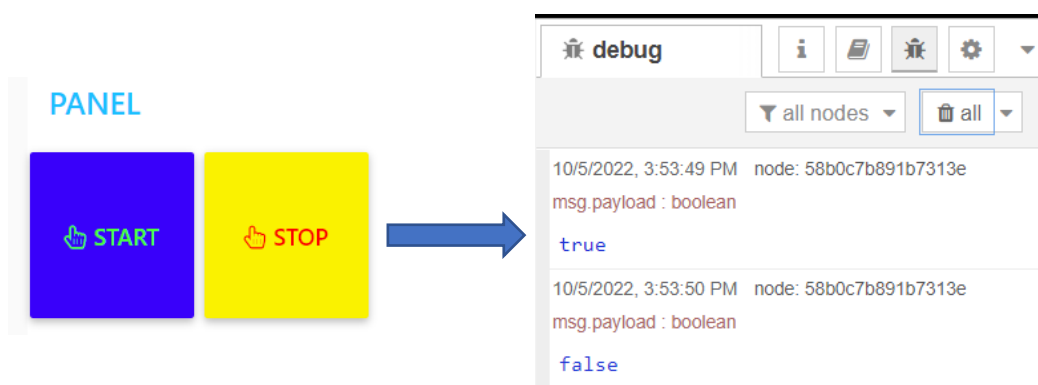


## การใช้งาน Button node เบื้องต้น

1. เพิ่มปุ่มกด STOP ใน UI Dashboard สั่งงานให้ส่ง boolean False ออกมาจาก output

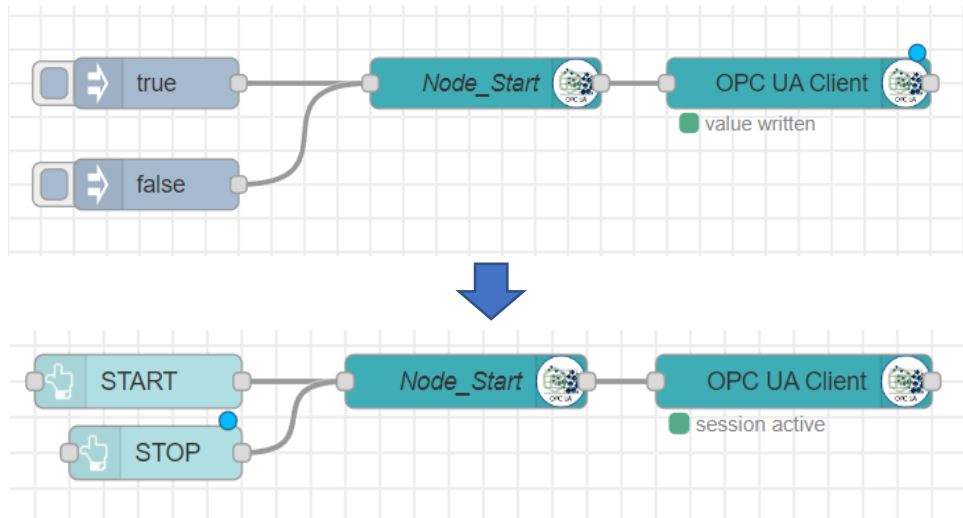


2. คำนวณโหลดโปรแกรมตรวจสอบปุ่มที่สร้างขึ้นใน UI Dashboard จากนั้นทดสอบการกดปุ่มส่งค่าใน debug

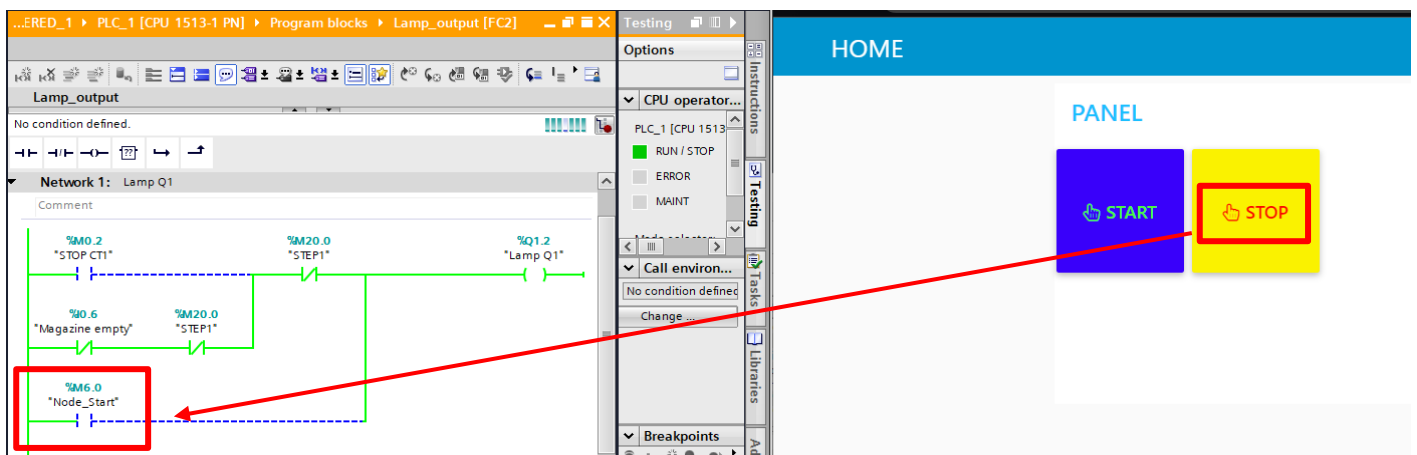
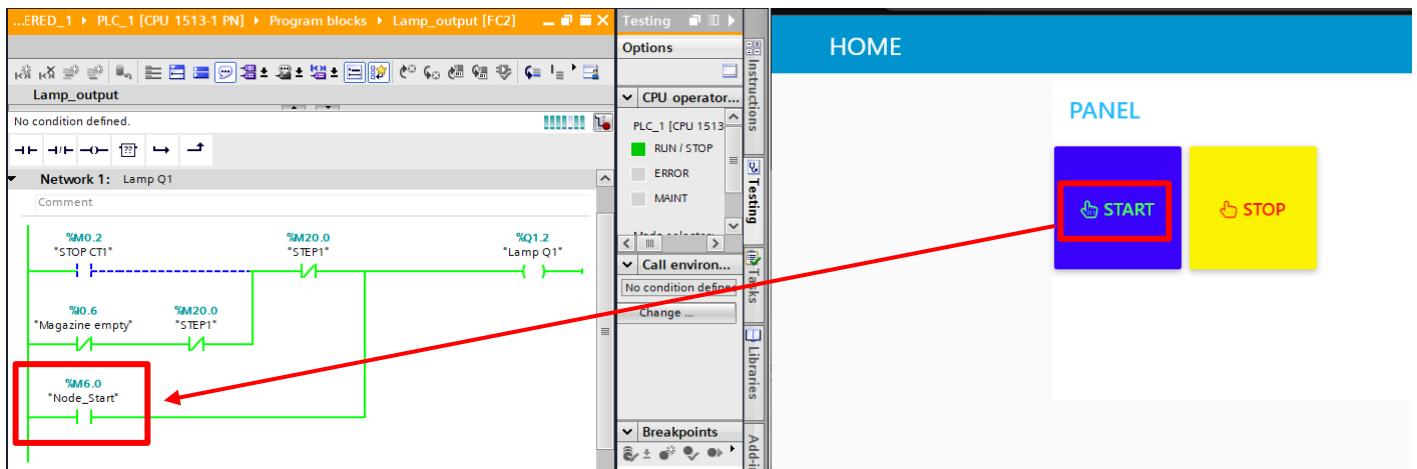




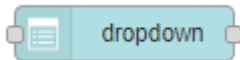
3. สร้างปุ่มกด Button node แทนที่ปุ่ม inject node สร้าง OPCUA Node\_Start ทำงาน (ใบงานที่ 4 หน้า 15)



ทดสอบการทำงานกดปุ่ม START/STOP ใน UI Dashboard ตรวจสอบการส่งค่าไป OPCUA Server สร้างให้ หลอดไฟ LAMP\_Q1 ทำงานในโปรแกรม Tia Portal



## 2. Dropdown node



ปุ่มเลือกหัวข้อการสั่งงาน สามารถกด Dropdown ในหน้า UI Dashboard เพื่อเลือกหัวข้อการสั่งงานได้เมื่อกดเลือกหัวข้อ จะมีข้อมูลที่เรารวบรวมค่าส่งออกมาซึ่ง Output สามารถเลือก Type ที่ส่งออกมาได้

- เพิ่ม Dropdown node ในหน้าจอ node red กดเข้าไปเพื่อตั้งค่าการใช้งาน

- การกำหนดค่า output ที่จะส่งออกไปขณะเลือกหัวข้อ

## การใช้งาน Dropdown node เบื้องต้น

เพิ่มหัวข้อการทำงาน เมื่อกดเลือกหัวข้อจะส่งค่าตัวเลขออกไปใช้งานจากนั้นส่งค่าไปยัง PLC

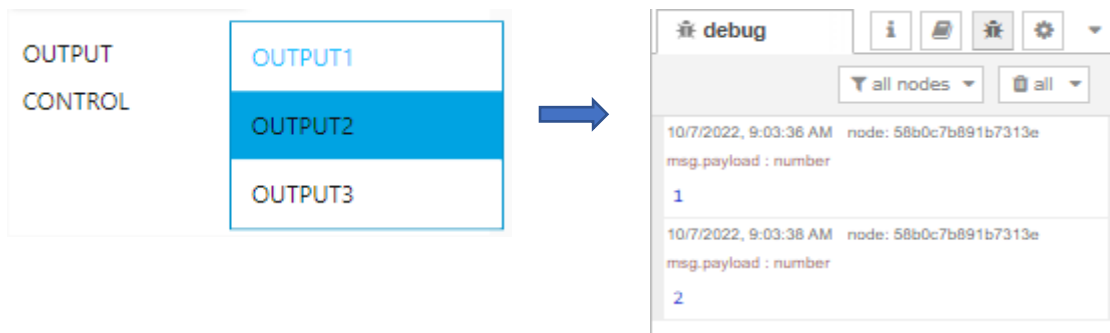
- ตั้งค่าส่งค่าตัวเลขและกำหนดชื่อของข้อมูล



- เชื่อมต่อ Dropdown node เข้ากับ Debug node จากนั้นดาวน์โหลดโปรแกรม ทดสอบการทำงานที่หน้า UI Dashboard



- กดเลือกหัวข้อการสั่งงาน Select option ตรวจสอบค่าที่ออกมาใน Debug

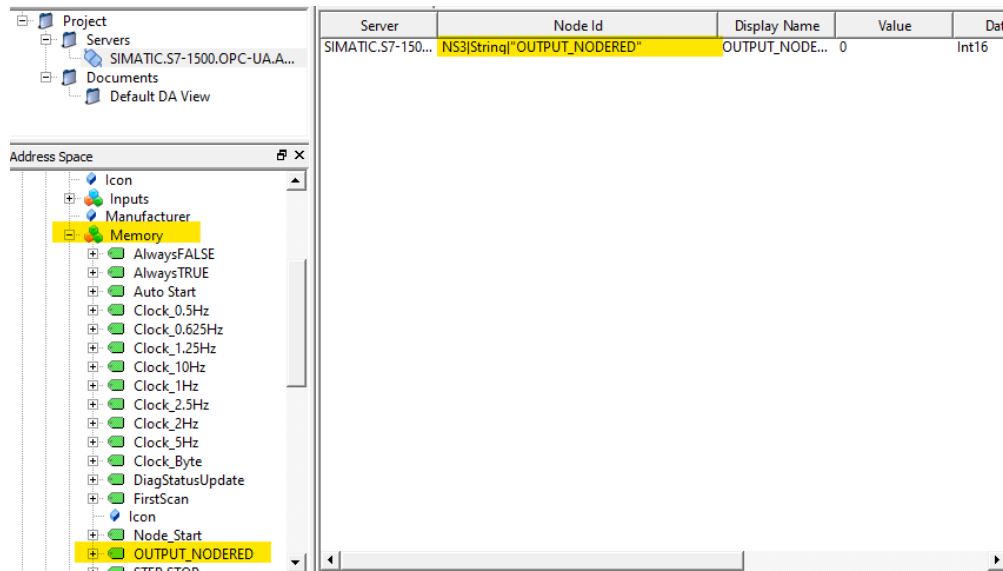


**\*\*ค่า output จะออกมาก็ต่อเมื่อทำการกดเลือกหัวข้อเท่านั้นและจะส่งออกมาเพียงแค่ครั้งเดียวจนกว่าจะมีการเลือกหัวข้อใหม่**

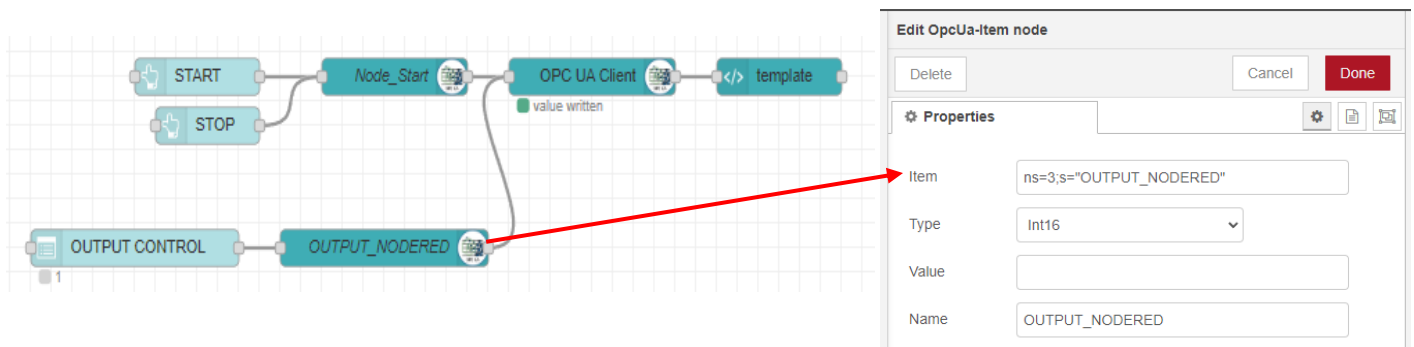
- เข้าไปที่โปรแกรม Tia Portal เพิ่ม Global Memory ที่รับค่าจำนวนเต็ม int ใน Default tag table

PLC tags	48	STOP CT1	Bool	%M0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Show all tags	49	Start OPCUA	Bool	%M0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Add new tag table	50	Node_Start	Bool	%M6.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Default tag table [98]	51	OUTPUT_NODERED	Int	%MW10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PLC data types	52	<Add new>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

- คำนวณโหลดโปรแกรม PLC ตรวจสอบชื่อ item ใน UaExpert



- เพิ่ม OpcUa-item ตั้งชื่อ item OUTPUT\_NODERED จากนั้นเชื่อมต่อ node เข้ากับ OPCUA Client READ



- ทดสอบการส่งค่าใน UI Dashboard ตรวจสอบการส่งค่าไปยัง PLC ใน Watch and force tables

OUTPUT CONTROL

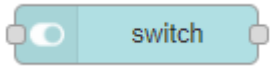
OUTPUT1

The screenshot shows the 'Watch and force tables' section of the UI Dashboard. The 'Watch table\_1' is expanded, showing the following data:

PLC data types	Address	Variable	Value	Comment
19	"Indicator light Start on"	%Q1.0	Bool	FALSE
20	"Indicator light (Reset)"	%Q1.1	Bool	FALSE
21	"Lamp Q1"	%Q1.2	Bool	TRUE
22	"Lamp Q2"	%Q1.3	Bool	FALSE
23	"OUTPUT_NODERED"	%MW10	DEC+/-	1
24		<Add new>		

A red arrow points from the 'OUTPUT1' dropdown menu to the '1' value in the 'OUTPUT\_NODERED' row of the Watch table.

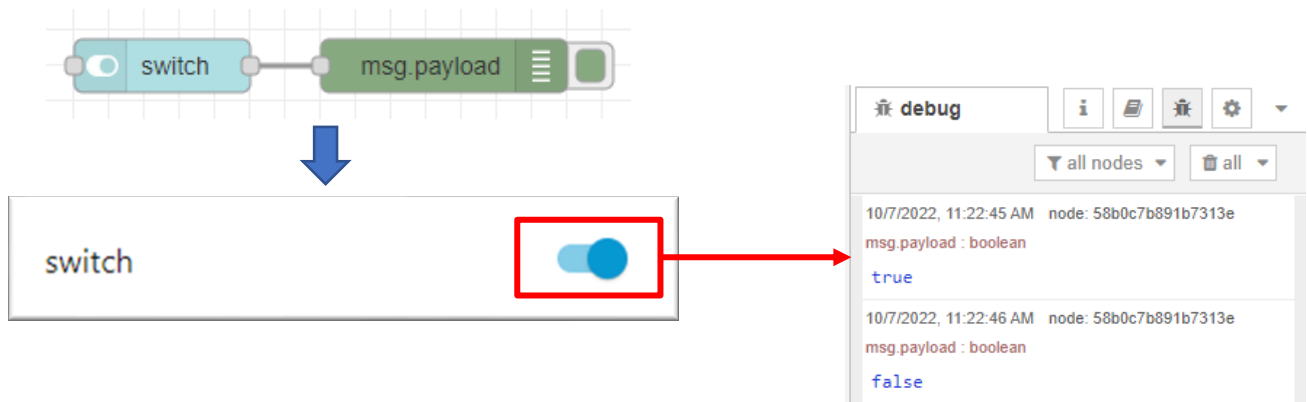
### 3. Switch node



การทำงานของ Switch node คล้ายกับ Button node แต่จะมีความสามารถในการเปิดหรือปิดการสั่งงานได้ ซึ่งต่างจาก Button node ที่สั่งงานแค่กดปุ่มทำงานเพียงอย่างเดียว เมื่อกดเปิดจะส่งค่าออกไปยัง output เมื่อกดปิดใช้งานจะส่งอีกค่าหนึ่งออกไปอีกครั้ง

- เพิ่ม Switch node ในหน้าจอ node red กดเข้าไปเพื่อตั้งค่าการใช้งาน

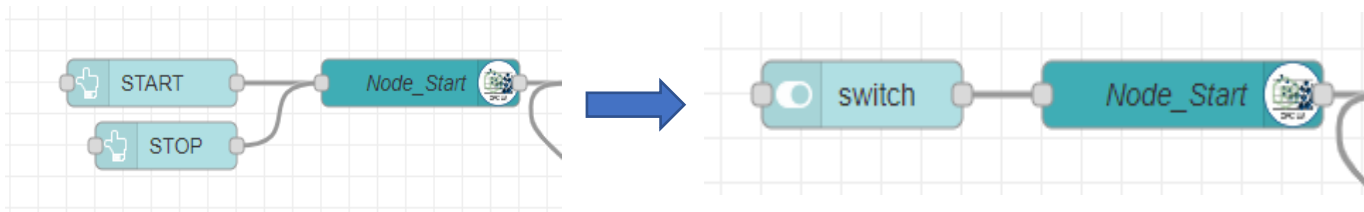
- เชื่อมต่อ Switch node เข้ากับ Debug node จากนั้นดาวน์โหลดโปรแกรม ทดสอบการสั่งงานผ่าน UI Dashboard



## การใช้งาน Switch node เบื้องต้น

เชื่อมต่อ Switch node แทนที่ Button node สั่งงาน ให้ Node\_Start ให้ทำงาน

- เชื่อมต่อ Switch node แทนที่ Button node



- ทดสอบสั่งงานให้ Node\_Start ทำงาน

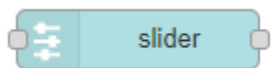
switch

PLC data types	Address	Variable Name	Address	Variable Type	Value
	22	"Lamp Q2"	%Q1.3	Bool	FALSE
	23	"OUTPUT_NODERED"	%MW10	DEC+/-	3
	24	"Node_Start"	%M6.0	Bool	TRUE
	25		<Add new>		

switch

PLC data types	Address	Variable Name	Address	Variable Type	Value
	22	"Lamp Q2"	%Q1.3	Bool	FALSE
	23	"OUTPUT_NODERED"	%MW10	DEC+/-	3
	24	"Node_Start"	%M6.0	Bool	FALSE
	25		<Add new>		

## 4. Slider node



การทำงานของ Slider node จะเป็นแถบเลื่อนซ้าย-ขวา ใช้ในการเพิ่มหรือลดค่าตัวเลข สามารถกำหนดค่าต่ำสุดหรือสูงสุดในแถบเลื่อนได้ เหมาะสำหรับการใช้เพิ่มหรือลดค่าที่เป็นตัวเลขก่อนจะส่งข้อมูลออกมา output

- เพิ่ม Slider node ในหน้าจอ node red กดเข้าไปเพื่อตั้งค่าการใช้งาน

**Edit slider node**

Buttons: Delete, Cancel, Done

**Properties**

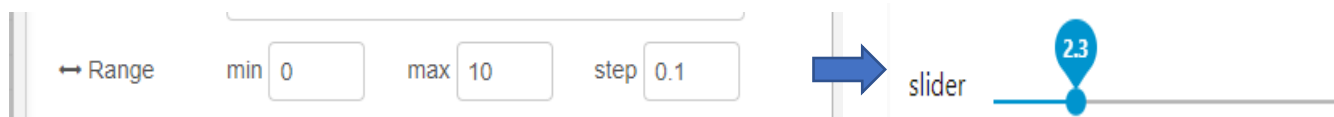
- Group: [HOME] PANEL (Annotation: ที่อยู่ Group)
- Size: auto (Annotation: กำหนดขนาดของ)
- Label: slider (Annotation: ตั้งชื่อใน UI Dashboard)
- Tooltip: optional tooltip
- Range: min 0 max 10 step 1 (Annotation: ตั้งค่า output กำหนด min/max จำนวน)
- Output: continuously while sliding (Annotation: ตั้งค่ารูปแบบการส่ง output)
- If msg arrives on input, pass through to output: ☒
- When changed, send:
  - Payload: Current value
  - Topic: msg. topic
  - Class: Optional CSS class name(s) for widget
  - Name: (Annotation: ตั้งชื่อ node)

- การตั้งค่า output กำหนด min/max

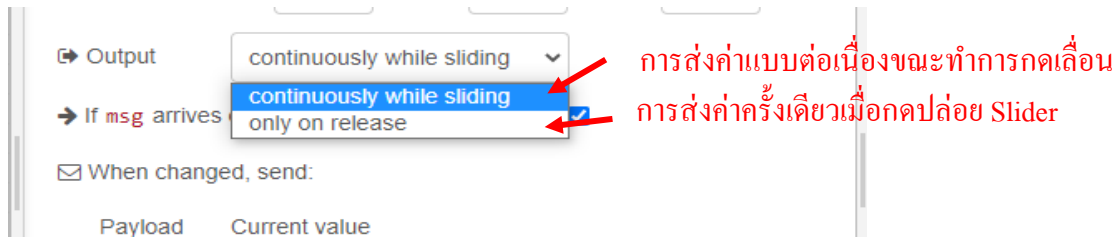
การกำหนดค่าสามารถใช้ได้ทั้งเลขจำนวนเต็ม หรือเป็นทศนิยมได้ การใช้งานส่งค่า OPCUA ก็จำเป็นต้องส่งค่าที่ Type ตรงกัน  
จำนวนเต็ม = int , จำนวนทศนิยม = Real (Data Type ใน PLC กับ OPCUA จะใช้ชื่อที่ต่างกัน)

Step = จำนวนค่าตัวเลขที่จะใช้ในการเลื่อนแถบหนึ่งครั้ง ยิ่งกำหนดค่าน้อยความละเอียดในการกำหนดค่าก็จะมากขึ้น

ตัวอย่างการตั้งค่า ตั้งแต่ 0-10 โดยที่จะเพิ่มค่าทีละ 0.1 ทำให้ค่าที่ออกมาเป็นเลข ทศนิยม



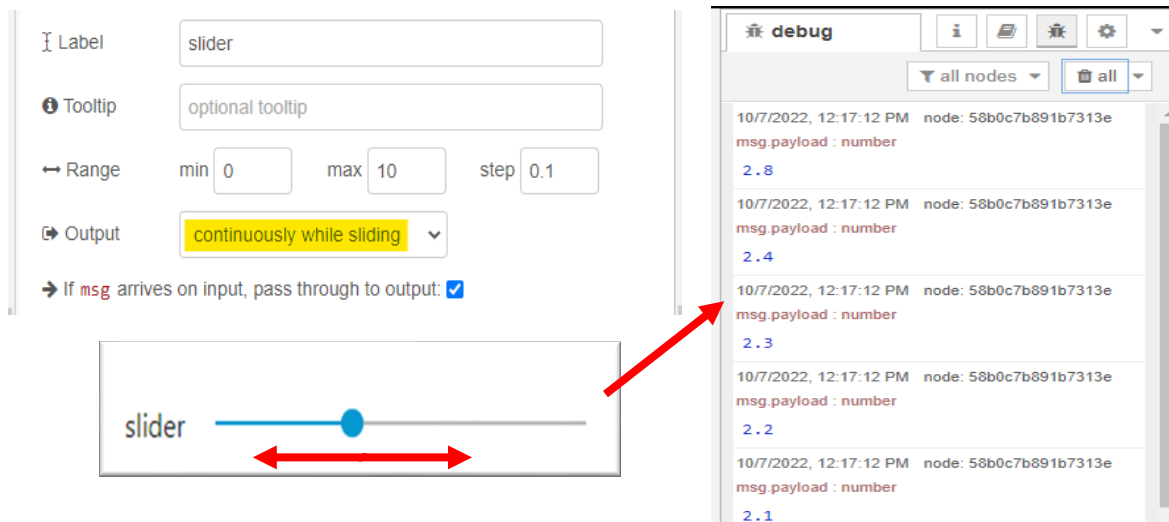
- การตั้งค่ารูปแบบการส่ง output



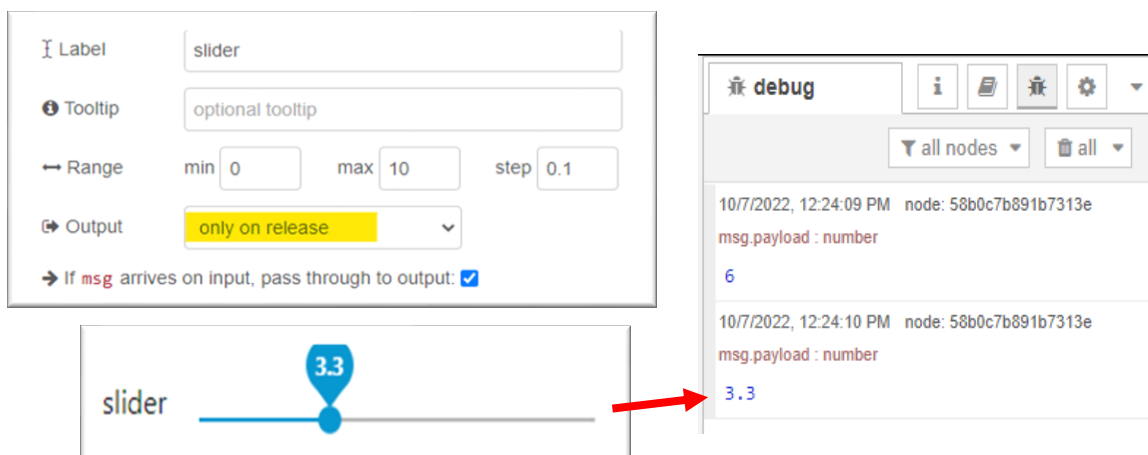
ทดสอบการทำงาน เชื่อมต่อ Slider node เข้ากับ debug node



1. ตั้งค่าการส่งข้อมูลแบบต่อเนื่องขณะทำการเลื่อน Slider คำนวณโหนดโปรแกรมจากนั้นทดลองกดเลื่อน Slider



2. ตั้งค่าการส่งข้อมูลแบบครั้งเดียวเมื่อกดปล่อย Slider คำนวณโหนดโปรแกรมจากนั้นทดลองกดเลื่อน Slider



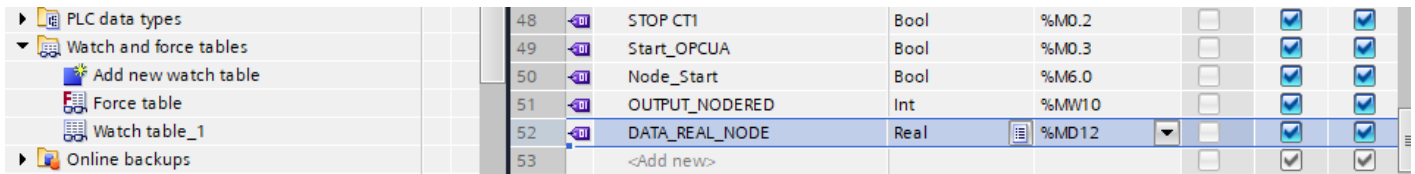


## การใช้งาน Slider node เบื้องต้น

ส่งค่าจาก Slider node ไปยัง PLC แล้วรับค่าจาก PLC มาเปลี่ยนค่าของ Slider

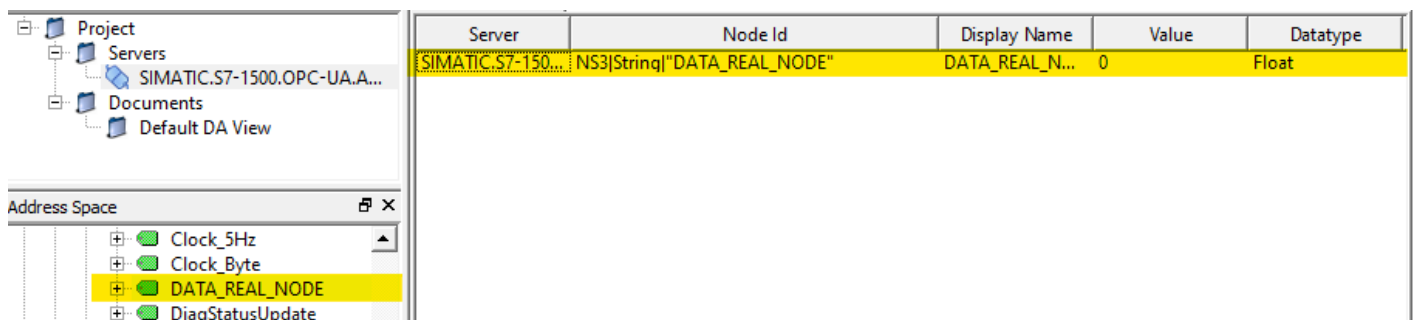
\*\*ในตัวของ Slider node สามารถรับค่าเข้าที่ input มาเปลี่ยนแปลงค่าของ Slider node ได้

1. เข้าไปที่โปรแกรม Tia Portal สร้าง Global memory ที่เก็บค่าแบบตัวเลขทศนิยม Real จากนั้นดาวน์โหลดโปรแกรม PLC



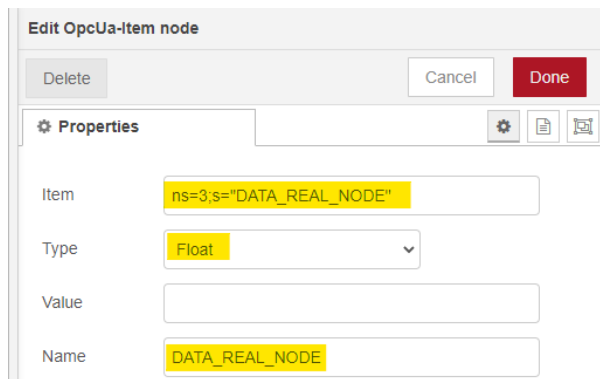
Address	Symbol	Data Type	Address	Symbol	Data Type	Address	Symbol	Data Type
48	STOP_CT1	Bool	%M0.2					
49	Start_OPCUA	Bool	%M0.3					
50	Node_Start	Bool	%M6.0					
51	OUTPUT_NODERED	Int	%MW10					
52	DATA_REAL_NODE	Real	%MD12					
53	<Add new>							

2. ตรวจสอบ item ที่เพิ่มขึ้นมาใหม่ใน UaExpert จะเห็นว่า Type ที่ OPCUA จะเป็น Float แทน



Server	Node Id	Display Name	Value	Datatype
SIMATIC.S7-1500...	NS3[String]"DATA_REAL_NODE"	DATA_REAL_N...	0	Float

3. เพิ่ม OpcUa-item DATA\_REAL\_NODE ขึ้นมา สังเกตชื่อ Node id และ Data type ใน UaExpert



**Edit OpcUa-item node**

Delete Cancel Done

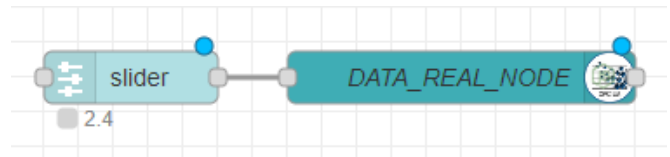
**Properties**

Item: ns=3;s="DATA\_REAL\_NODE"

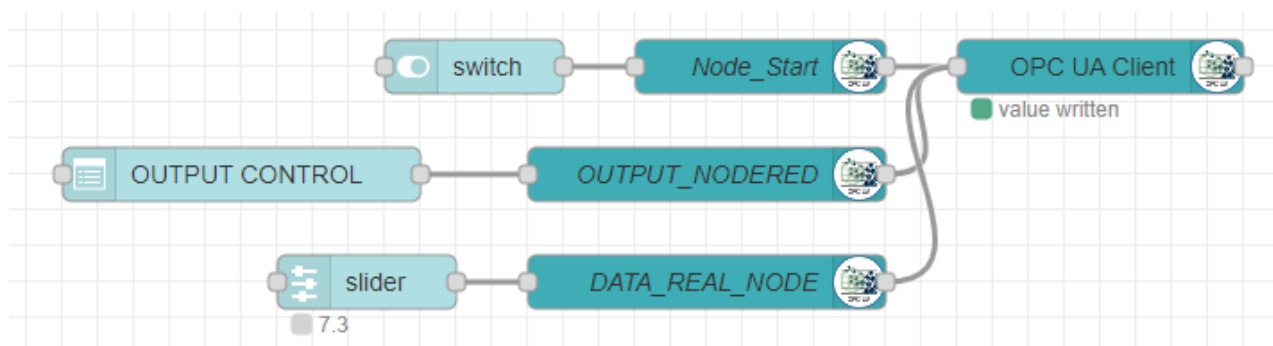
Type: Float

Value:

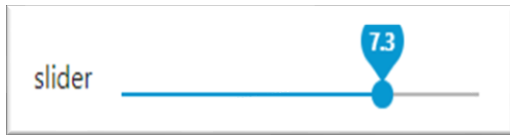
Name: DATA\_REAL\_NODE



4. ส่งค่า DATA\_REAL\_NODE ไปยัง OPCUA Client WRITE

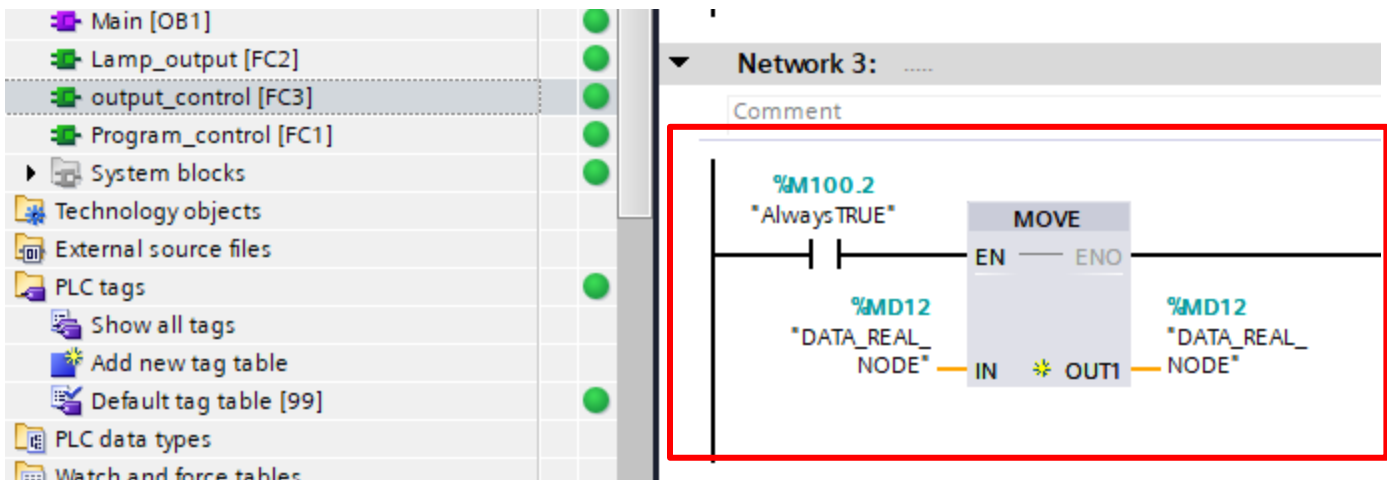


5. ตรวจสอบค่าที่ส่งไป PLC ใน Watch and force tables

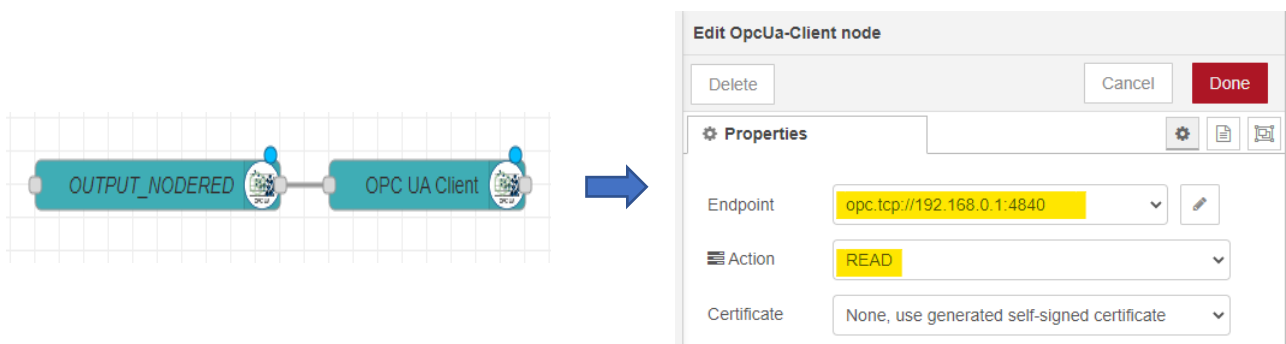


▼ Watch and force tables	23	*OUTPUT_NODERED*	%MW10	DEC+/-	3
✦ Add new watch table	24	*Node_Start*	%M6.0	Bool	<input type="checkbox"/> FALSE
📊 Force table	25	*DATA_REAL_NODE*	%MD12	Floating-point nu...	7.3
📊 Watch table_1	26		<Add new>		

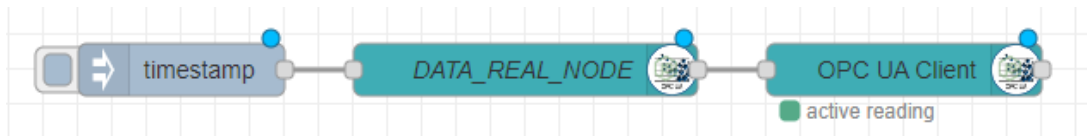
6. สร้าง MOVE ใน Tia Portal ใช้สำหรับทดสอบเปลี่ยนค่า DATA\_REAL\_NODE จากนั้นดาวน์โหลดโปรแกรม



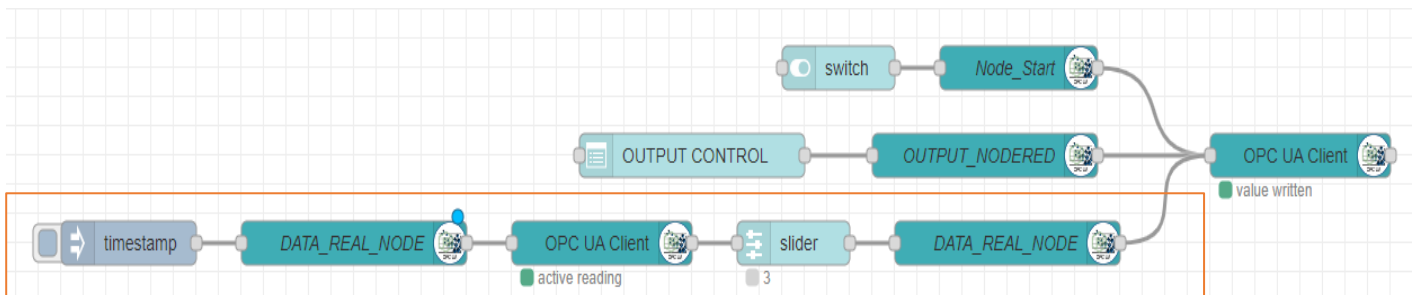
7. ทำการ Copy OpcUa-item DATA\_REAL\_NODE ขึ้นมาหนึ่ง Node จากนั้นเพิ่ม OPC UA Client READ ขึ้นมาอีกหนึ่ง Node



8. เพิ่ม inject node เชื่อมต่อเข้ากับ input OpcUa-item DATA\_REAL\_NODE ตามรูป



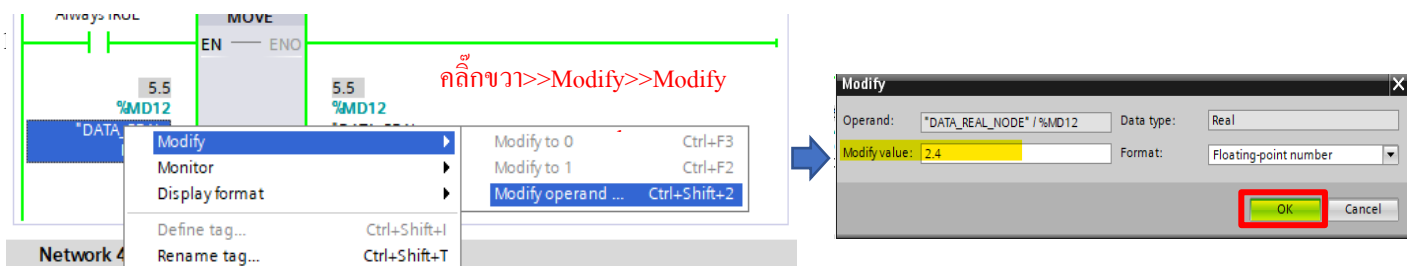
9. เชื่อมต่อ node ทั้งหมด เข้ากับ input Slider node



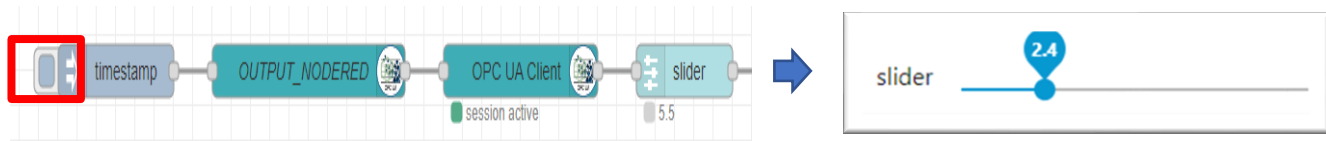
ทดสอบการทำงานสั่งงานเปลี่ยนค่า DATA\_REAL\_NODE โดยเลื่อน Slider node



- เปลี่ยนค่าตัวเลข DATA\_REAL\_NODE ใน Tia Portal

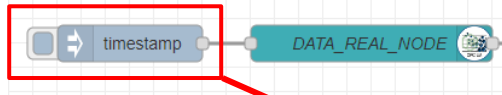


- กด inject ใน node red ค่าตัวเลขที่เปลี่ยนแปลงของ Slider node



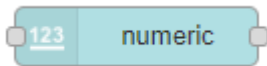
การตั้งค่า inject node ให้ทำงานส่งค่าแบบต่อเนื่อง

กดเข้าไปที่ตั้งค่า inject node ตั้งค่า Repeat ส่งค่าแบบ interval ทุก ๆ 1 วินาที



\*\*ดาวน์โหลดโปรแกรมทดลองส่งค่าและรับค่าอีกครั้งหนึ่ง

## 5. Numeric node

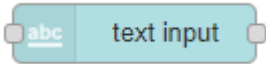


หลักการทำงาน Numeric node จะคล้ายกับ Slider node แต่จะเป็นการกดคลิก ขึ้นหรือลง เพิ่มค่าหรือลดค่าตัวเลขแทน สามารถรับค่า input เปลี่ยนแปลงค่าได้ เหมือนกับ Slider node

- เพิ่ม Numeric node ในหน้าจอ node red กดเข้าไปเพื่อตั้งค่าการใช้งาน

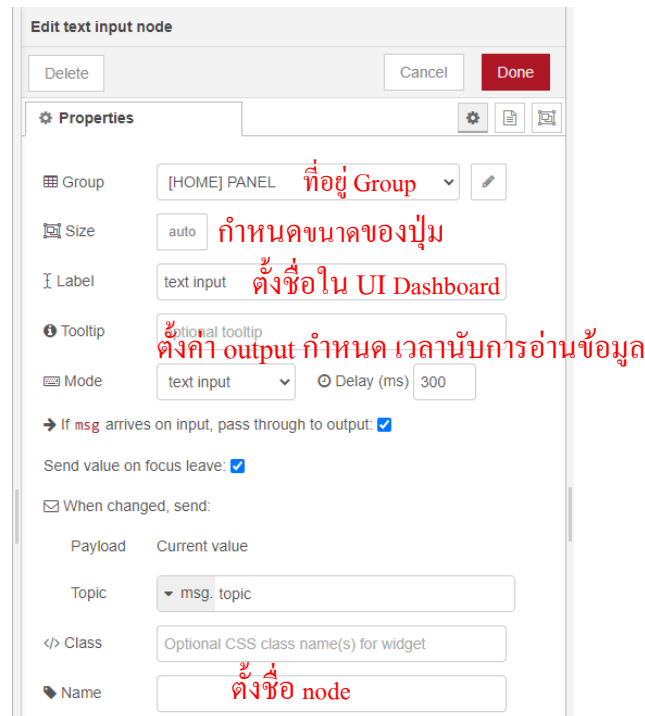


## 6. Text input node



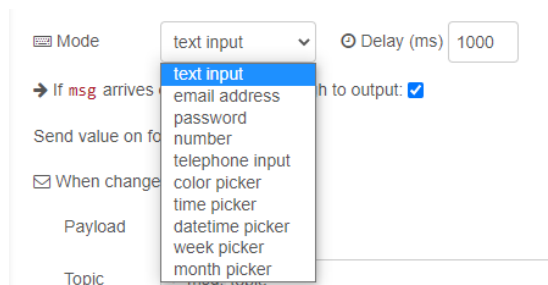
หลักการทำงานของ Text input node จะเป็นช่อง UI ให้ผู้ใช้งานใส่ตัวอักษร หรือตัวเลข ที่ต้องการ จากนั้นจะมีการนับเวลาใน Text input node เมื่อนับเวลาครบจะทำการส่งค่า output ออกไปหนึ่งครั้ง ค่าข้อมูลจะส่งไปอีกครั้งเมื่อมีการใส่ค่าข้อมูลใหม่หรือมีสัญญาณเข้าที่ input ค่าของ Text input node ตามเปลี่ยนแปลงตามค่าที่เข้ามาจาก input

- เพิ่ม Text input node ในหน้าจอ node red กดเข้าไปเพื่อตั้งค่าการใช้งาน



- การตั้งค่า output และการกำหนดเวลาในการอ่านข้อมูล

การตั้งค่าโหมดการทำงาน มีการตั้งค่าใช้งานหลายประเภท สามารถเลือกใช้งานโหมดการทำงานที่ต้องการได้



การกำหนดเวลา Delay (ms) เมื่อมีการเขียนข้อมูลใน UI Dashboard จะทำให้ Delay เริ่มนับเวลา เมื่อนับเวลาครบจะส่งข้อมูลออกทันที

## การใช้งาน Text input node เบื้องต้น

ส่งค่าตัวอักษรจาก PLC (Real) Text input node (String) ไปยัง Text input node (String) ใน UI Dashboard

1. เข้าไปตั้งค่าใน Text input node ตั้งค่า Mode=text input และ Delay=1000 (ms)

Group: [HOME] PANEL

Size: auto

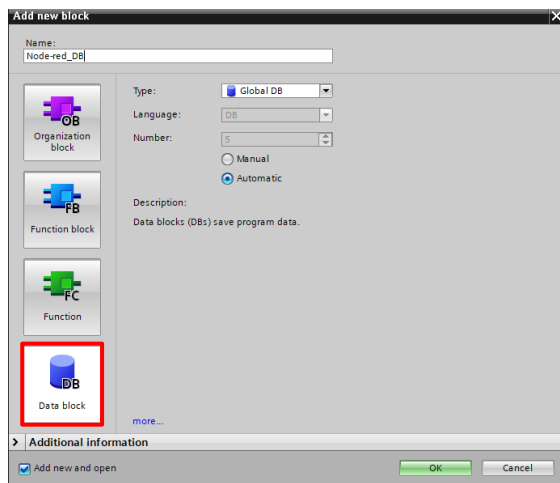
Label: text input

Tooltip: optional tooltip

Mode: text input

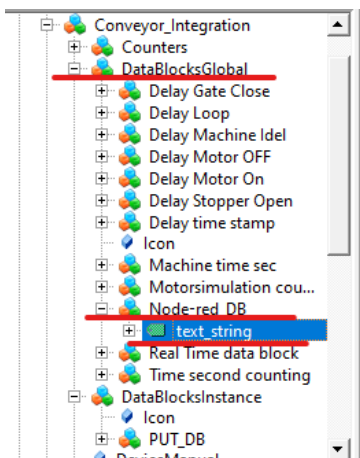
Delay (ms): 1000

2. การใช้ Data type String จำเป็นต้องประกาศใช้งานใน Data block จึงจะสามารถใช้งานได้



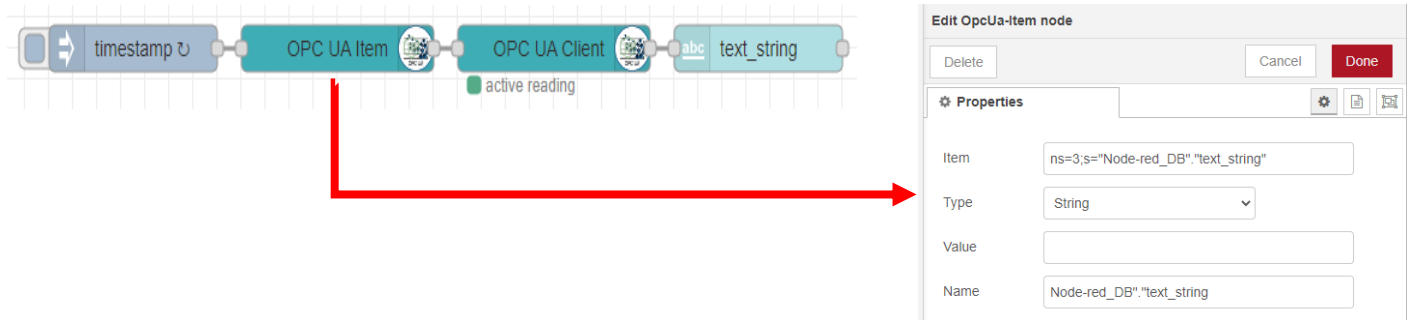
Node-red_DB							
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...
1	Static						
2	text_string	String			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

3. ค้นหา ITEM OPCUA ของ Text input ที่สร้างขึ้นมาใหม่ใน UaExpert



Default DA View				
Server	Node Id	Display Name	Value	Datatype
SIMATIC.S7-150...	NS3[String]"Node-red_DB"."text_string"	text_string		String

## 4. เชื่อมต่อ Node id Text input กับ OPCUA Node READ



## 5. ทดสอบการส่งข้อมูลโดยส่งค่าตัวอักษรจาก PLC มาแสดงในหน้า Dashboard text input node

HOME

### TEXT\_Status

text\_string  
nodered-test

YORSTATIONconveyor\_pickandplaceVolker\_conveyor\_pickandplaceVolker\_conveyor\_pickandplace

Window Help

Volker\_conveyor\_pickandplace > Conveyor\_Integration [CPU 1516-3 PN/DP] > Program blocks > Node-red\_DB [DB1]

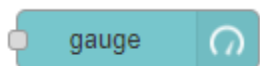
Keep actual values Snapshot Copy snapshots to start values Load start values as actual values

Node-red\_DB

	Name	Data type	Start value	Monitor value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Sup...
1	Static									
2	text_string	String	"	'nodered-test'		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	<Add new>					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	



## 7. Gauge node



การทำงานของเกจวัดจะแสดงค่าตัวเลขตามค่าที่ส่งเข้ามา ผู้ใช้งานสามารถส่งค่าที่เป็นตัวเลขจำนวนเต็ม หรือทศนิยม ให้กับเกจวัดได้ ค่าของตัวเลขจะเปลี่ยนแปลงตามการเปลี่ยนแปลงที่ส่งค่าเข้ามา

- เพิ่ม Gauge node ในหน้าจอ node red กดเข้าไปเพื่อตั้งค่าการใช้งาน

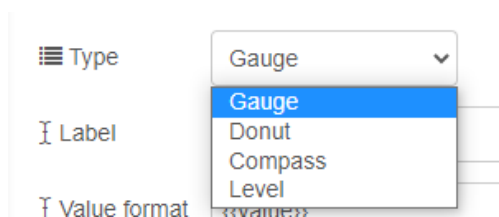
- Gauge node สามารถเปลี่ยนแปลง Type การแสดงผลได้หลายรูปแบบดังนี้

Gauge = เกจวัดแบบครึ่งวงกลม

Donut = เกจวัดแบบวงกลม

Compass = เกจวัดแบบเข็มทิศ

Level = เกจวัดแบบเพิ่มระดับ

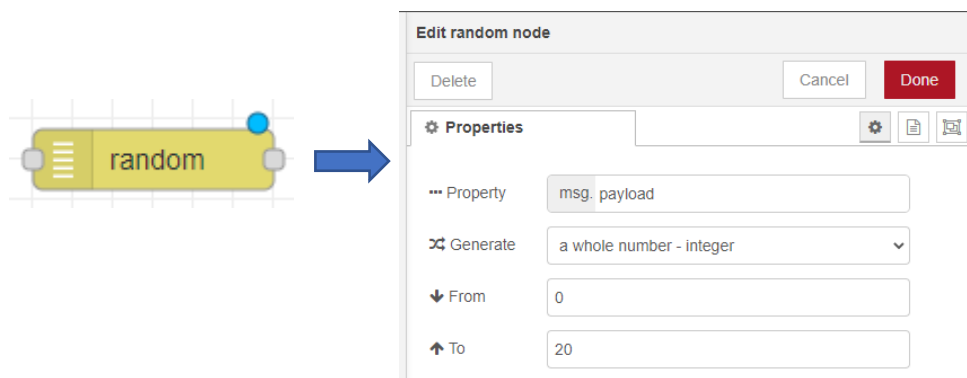


## การใช้งาน Gauge node เบื้องต้น

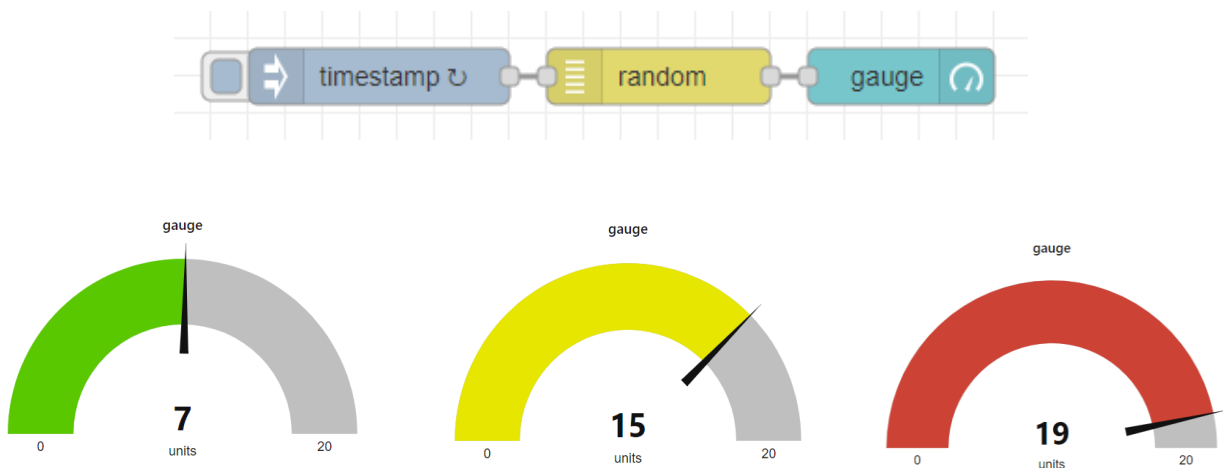
การส่งค่าตัวเลขไปแสดงผลบน Gauge node

1. เข้าไปตั้งค่าใน Gauge node กำหนดค่า min, max ของข้อมูล

2. ทำการทดสอบการส่งค่าตัวเลขโดยจะใช้ Random node ในการสุ่มค่าตัวเลข 0-20



3. เชื่อมต่อ Random node เข้ากับ inject node ให้ทำงานส่งค่าแบบต่อเนื่อง ทุก ๆ 1 วินาที (Interval = 1 seconds)



## 7. Chart node



ใช้ในการแสดงผลแบบกราฟ ผู้ใช้งานสามารถส่งค่าที่เป็นตัวเลขจำนวนเต็ม หรือทศนิยม ไปแสดงผลบนกราฟได้ สามารถกำหนดช่วงเวลาของการอ่านค่าข้อมูล หรือค่า min, max ของข้อมูลได้

Delete
Cancel
Done

Properties

Group
[HOME] Group
ที่อยู๋ Group

Size
auto
กำหนดขนาดของปุ่ม

Label
chart
ตั้งชื่อใน UI Dashboard

Type
Line chart
enlarge points

X-axis
last 1 hours OR 1000 points
กำหนดช่วงเวลาในการแสดงผลข้อมูล

X-axis Label
HH:mm:ss
as UTC

Y-axis
min max
ตั้งค่า min, max ของข้อมูล

Legend
None
Interpolate linear

Series Colours
กำหนดสีของข้อมูลที่แสดงผล

Blank label
display this text before valid data arrives

Class
Optional CSS class name(s) for widget

Name
Name
กำหนดชื่อ node

## การใช้งาน Chart node เบื้องต้น

การส่งค่าตัวเลขไปแสดงผลบน Chart node

1. ตั้งค่าช่วงเวลาในการแสดงผล ในช่วงเวลา 10 วินาที และกำหนดค่า min, max ของข้อมูล 0-20

2. เชื่อมต่อ Chart node เข้ากับ Random node ทดสอบการแสดงผลของกราฟใน Ui dashboard

