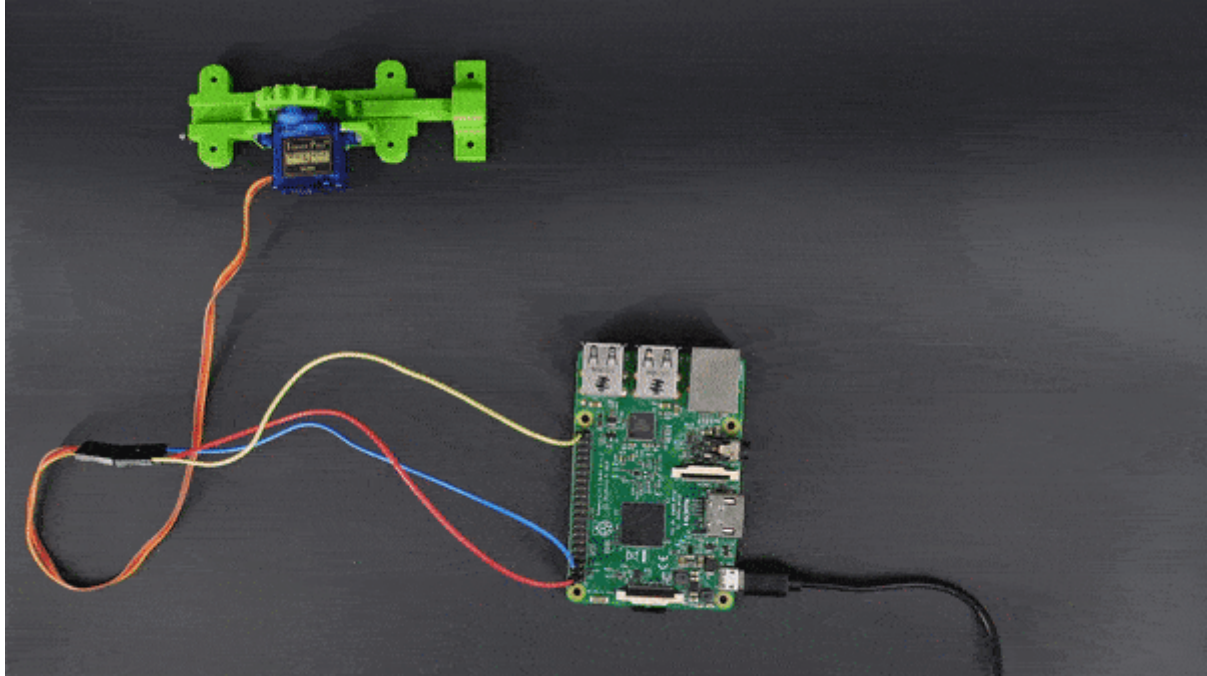


التحكم بمحرك سيرفو باستخدام النود-ريد (Node-RED)

تستخدم المحركات بشكل شائع في مجال الروبوتات، و يمكننا أن نتحكم بهذه الروبوتات عن طريق ربطها بالانترنت، في هذا المشروع نقوم بتوصيل محرك سيرفو مع راسبيري باي Raspberry Pi والتحكم به من خلال صفحة ويب باستخدام النود-ريد (Node-RED) سيتم إنشاء صفحة منقولة لعرض زاوية المحرك و مزلاج من درجة 0-180 و ترسل قيم الزاوية المراد تحريكها إلى راسبيري باي (Raspberry Pi) للتحكم بمحرك سيرفو ،



المواد و الأدوات



1X راسپيري باي



1X ذاكرة (8 قيفا كحد أدنى)



1X محول طاقة (5V-2A)



1X محرك سيرفو

تثبيت النودريد (Node-RED)

يتم تثبيت النود-ريد (Node-RED) عند تثبيت نظام التشغيل راسبين (Raspbian) النسخة الكاملة على راسبيري باي، لكن إذا ثبتت نظام (Raspbian Lite) فإنه لا يتضمن على النود-ريد (Node-RED) فلذلك في البداية ستحتاج إلى اتباع الخطوات التالية. ابدأ بتثبيت أدوات (git) ، بالإضافة إلى أدوات البناء الأساسية التي ستكون مطلوبة في الخطوة التالية. عن طريق الأمر :

```
sudo apt install build-essential git
```

قم بتشغيل أمر (bash) لتثبيت نسخة جديدة من (Node-RED) عن طريق السطر التالي :

```
bash <(curl -sL  
https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs  
-and-nodered)
```

أثناء عملية التثبيت سيتطلب أن تجيب بنعم مرتين و ذلك بكتابة (Y) سيستغرق التثبيت بعض الوقت و بعد اكتمال التثبيت تظهر معلومات حول إعدادات نود-ريد (Node-RED) في وحدة التحكم.

```
pi@nodered: ~
Running Node-RED install for user pi at /home/pi on raspbian

This can take 20-30 minutes on the slower Pi versions - please wait.

Stop Node-RED ✓
Remove old version of Node-RED ✓
Remove old version of Node.js ✓
Install Node.js LTS ✓ Node v12.18.2 Npm 6.14.6
Clean npm cache ✓
Install Node-RED core ✓ 1.1.1
Move global nodes to local -
Install extra Pi nodes ✓
Npm rebuild existing nodes -
Add shortcut commands ✓
Update systemd script ✓

Any errors will be logged to /var/log/nodered-install.log
All done.
You can now start Node-RED with the command node-red-start
or using the icon under Menu / Programming / Node-RED
Then point your browser to localhost:1880 or http://{your_pi_ip-address}:1880

Started Wed 8 Jul 23:52:45 BST 2020 - Finished Thu 9 Jul 00:00:10 BST 2020

pi@nodered:~ $
```

إذا كنت تستخدم راسبيري باي (Raspberrypi 4) الذي يمتلك ذاكرة عشوائية أكثر من 2 غيغابايت يمكن البدء بخدمة نود-ريد (Node-RED). بكتابة الأمر التالي:

```
node-red-start
```

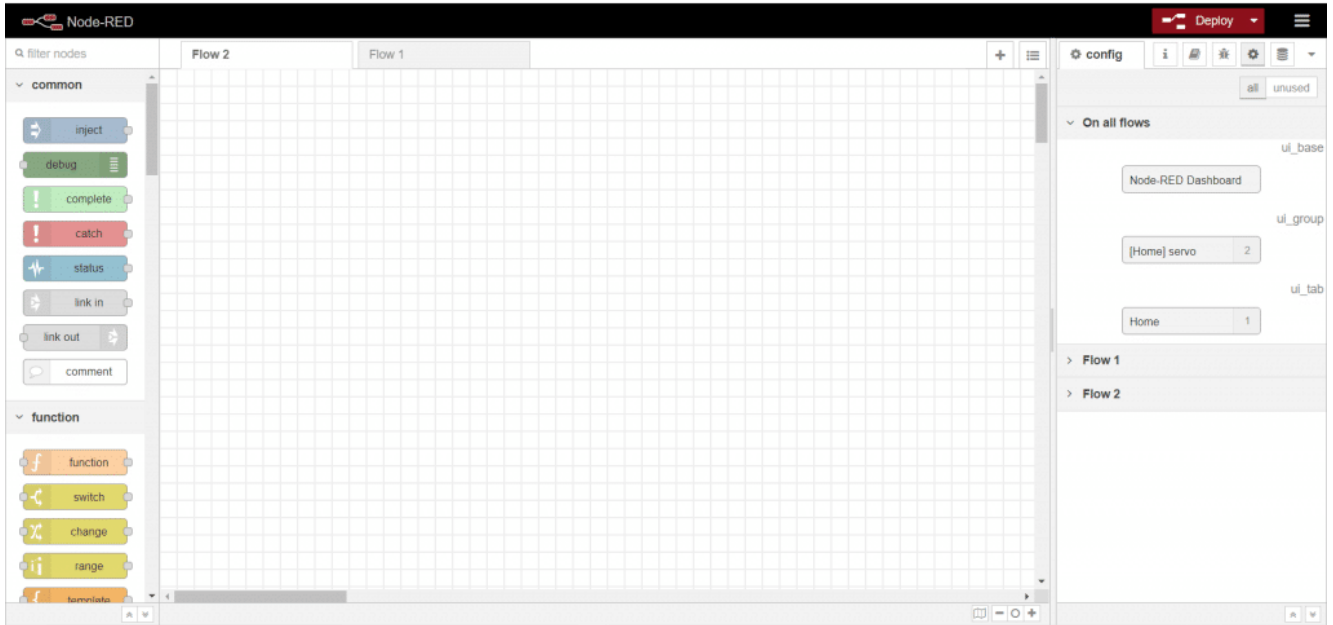
بالنسبة لإصدارات راسبيري باي (Raspberry Pi) الأقدم من الأفضل تحديد مقدار ذاكرة الوصول العشوائي المتاحة يمكن استخدام الأمر التالي:

```
node-red-pi --max-old-space-size=256
```

في ثواني سوف تبدأ خدمة النود-ريد (Node-RED) اكتب `http://nodered.local:1880` في شريط العنوان الخاص بمتصفحك

اكتب عنوان IP الخاص بالراسبيري باي لديك بدل من "nodered.local"

و ستظهر لديك واجهة النود-ريد

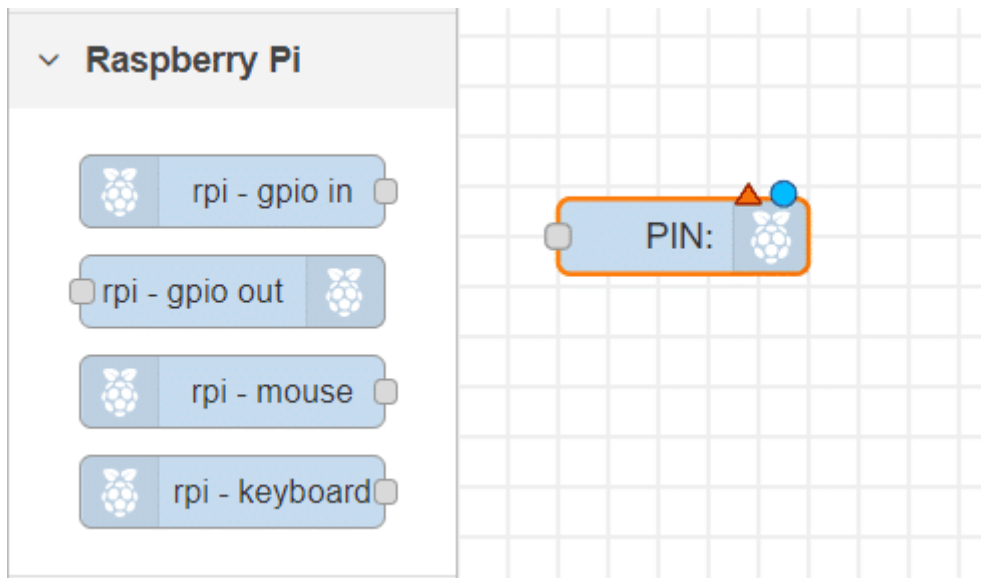


إنشاء التدفق

في البداية يمكنك الاطلاع على السلسلة التعليمية الخاصة بالنود-ريد (Node-RED) من الرابط

أولاً: الاتصال بمنفذ الإدخال و الإخراج (GPIO) في لوحة الراسبيري باي :

من قسم عقد راسبيري باي نسحب المخرجات "rpi gpio" و التي تكون رمز الراسبيري على اليمين



حدد المنفذ الذي قمت بتوصيل محرك سيرفو معه

Edit rpi-gpio out node

Delete

Cancel

Done

⚙ Properties



Ground - 9 <input type="radio"/>	<input type="radio"/> 10 - GPIO15 - RxD
GPIO17 - 11 <input type="radio"/>	<input type="radio"/> 12 - GPIO18
GPIO27 - 13 <input type="radio"/>	<input type="radio"/> 14 - Ground
GPIO22 - 15 <input type="radio"/>	<input type="radio"/> 16 - GPIO23
3.3V Power - 17 <input type="radio"/>	<input type="radio"/> 18 - GPIO24
MOSI - GPIO10 - 19 <input type="radio"/>	<input type="radio"/> 20 - Ground
MISO - GPIO09 - 21 <input type="radio"/>	<input type="radio"/> 22 - GPIO25
SCLK - GPIO11 - 23 <input type="radio"/>	<input type="radio"/> 24 - GPIO8 - CE0
Ground - 25 <input type="radio"/>	<input type="radio"/> 26 - GPIO7 - CE1
SD - 27 <input type="radio"/>	<input type="radio"/> 28 - SC
GPIO05 - 29 <input type="radio"/>	<input type="radio"/> 30 - Ground
GPIO06 - 31 <input type="radio"/>	<input type="radio"/> 32 - GPIO12
GPIO13 - 33 <input type="radio"/>	<input type="radio"/> 34 - Ground
GPIO19 - 35 <input type="radio"/>	<input type="radio"/> 36 - GPIO16
GPIO26 - 37 <input type="radio"/>	<input type="radio"/> 38 - GPIO20
Ground - 39 <input type="radio"/>	<input checked="" type="radio"/> 40 - GPIO21

Type

Digital output



Initialise pin state?

حدد نوع الاشارة و قيمة التردد

SCLK - GPIO11 - 23 <input type="radio"/>	<input type="radio"/> 24 - GPIO8 - CE0
Ground - 25 <input type="radio"/>	<input type="radio"/> 26 - GPIO7 - CE1
SD - 27 <input type="radio"/>	<input type="radio"/> 28 - SC
GPIO05 - 29 <input type="radio"/>	<input type="radio"/> 30 - Ground
GPIO06 - 31 <input type="radio"/>	<input type="radio"/> 32 - GPIO12
GPIO13 - 33 <input type="radio"/>	<input type="radio"/> 34 - Ground
GPIO19 - 35 <input type="radio"/>	<input type="radio"/> 36 - GPIO16
GPIO26 - 37 <input type="radio"/>	<input type="radio"/> 38 - GPIO20
Ground - 39 <input type="radio"/>	<input checked="" type="radio"/> 40 - GPIO21

40

Type

PWM output



Frequency

100

Hz

Name

servo

ثانياً : تعيين قيم التحكم بالخرج

سنقوم بمعادلة قيم الدخل (0-26) مع أعلى و أقل قيمة لزاوية محرك سيرفو التي نود أن يتحرك خلالها و هي (0-180) قسم الدوال نسحب "range"

The image shows the Node-RED web interface. At the top left, there is a search bar labeled "filter nodes". Below it, a sidebar lists node categories: "common" (expanded) and "function" (collapsed). Under "function", several nodes are visible: "function", "switch", "change", "range", "template", "delay", and "trigger". The main workspace, titled "Flow 1", contains a grid with two nodes: a blue "SERVO" node with a Raspberry Pi icon and a yellow "range" node with a Raspberry Pi icon. A small grey button with a left-pointing arrow is located below the "range" node in the sidebar.

نحدد المدى للدخل، وكذلك نحدد المدى للخروج

Edit range node

Delete
Cancel
Done

⚙️ **Properties**

⚙️
📄
📏

⋮ Property

⦿ Action

➔ Map the input range:

from: to:

➔ to the target range:

from: to:

Round result to the nearest integer?

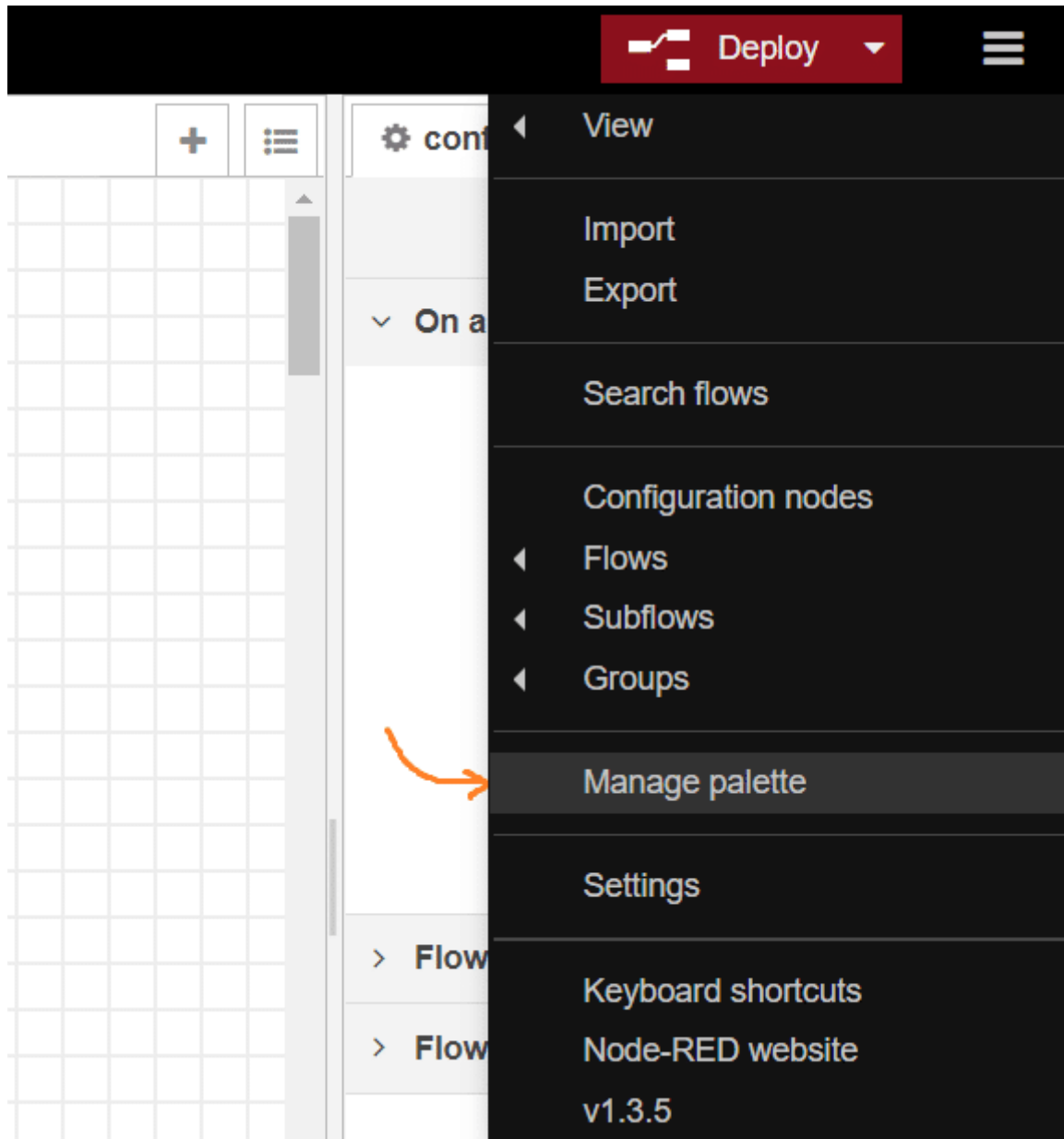
🏷️ Name

Tip: This node ONLY works with numbers.

ثالثاً: واجهة المستخدم

واجهة المستخدم سوف يظهر في واجهة المستخدم عنصرين الأول المزلاج للتحكم بالسيرفو العنصر الثاني شكل المنقلة لإظهار زاوية محرك سيرفو

للتمكن من انشاء واجهة مستخدم تحتاج إلى تنزيل مكتبة لوحة التحكم (Dashboard)



ستظهر لك قائمة يمكن من خلالها البحث عن المكتبة و تنزيلها

User Settings

Close

View: Nodes **Install**

Keyboard: sort: a-z recent

Palette: 49 / 3322 ✕

- cn-dashboard-nodes**
Install
0.0.2 2 years, 11 months ago installed
- node-red-dashboard**
A set of dashboard nodes for Node-RED
2.29.3 2 days ago installed
- feezal**
Web Components based Dashboard UI with WYSIWYG Editor
0.8.0 5 months ago install
- node-red-contrib-dashboard-average-bars**
Calculate and display the average values of msg.payload in a bar chart.
0.0.6 2 years, 10 months ago install

اسحب "slider" من قائمة عقد لوحة التحكم (Dashboard)

The image shows the Node-RED web interface. At the top, there is a black header with the Node-RED logo and the text "Node-RED". Below the header, there is a search bar labeled "filter nodes". On the left side, there is a sidebar with a "dashboard" section expanded, showing a list of nodes: button, dropdown, switch, slider, numeric, text input, date picker, colour picker, and form. The main workspace is titled "Flow 1" and contains a grid of nodes. A "slider" node is connected to a "SERVO" node (with a Raspberry Pi icon) and a "0 - 180" numeric node. The "slider" node is highlighted with an orange border.


بالنقر على العقدة ستظهر لنا قائمة تساعدنا في التحكم باعدادات العقدة

بالبداية قم باضافة مجموعة

Edit slider node

Delete Cancel Done

Properties [Settings] [List] [Preview]

Group Add new ui_group... [Add] 

Size auto

Label slider

Tooltip optional tooltip

Range min 0 max 10 step 1

Output continuously while sliding

→ If msg arrives on input, pass through to output:

انقر على اشارة القلم ثم انشئ مجموعة لواجهة المستخدم

Cancel

Add

⚙️ Properties

🏷️ Name

servo |

📄 Tab

Home

↔️ Width

6

Display group name

Allow group to be collapsed

حدد المجموعة و المدى من (0-26)

Edit slider node

Properties

Group [Home] servo

Size auto

Label PWM

Tooltip optional tooltip

Range min max step

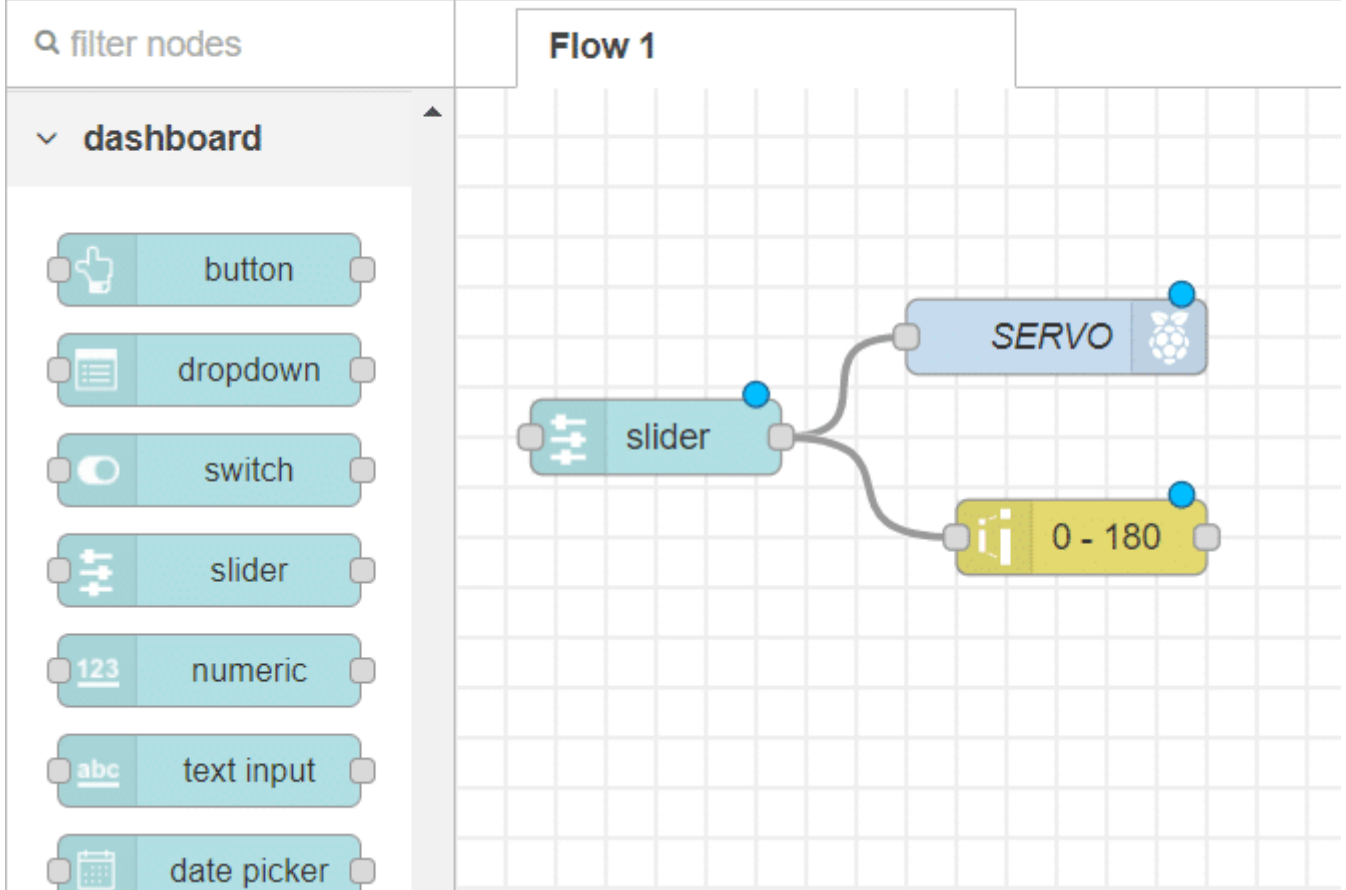
Output continuously while sliding

If msg arrives on input, pass through to output:

When changed, send:

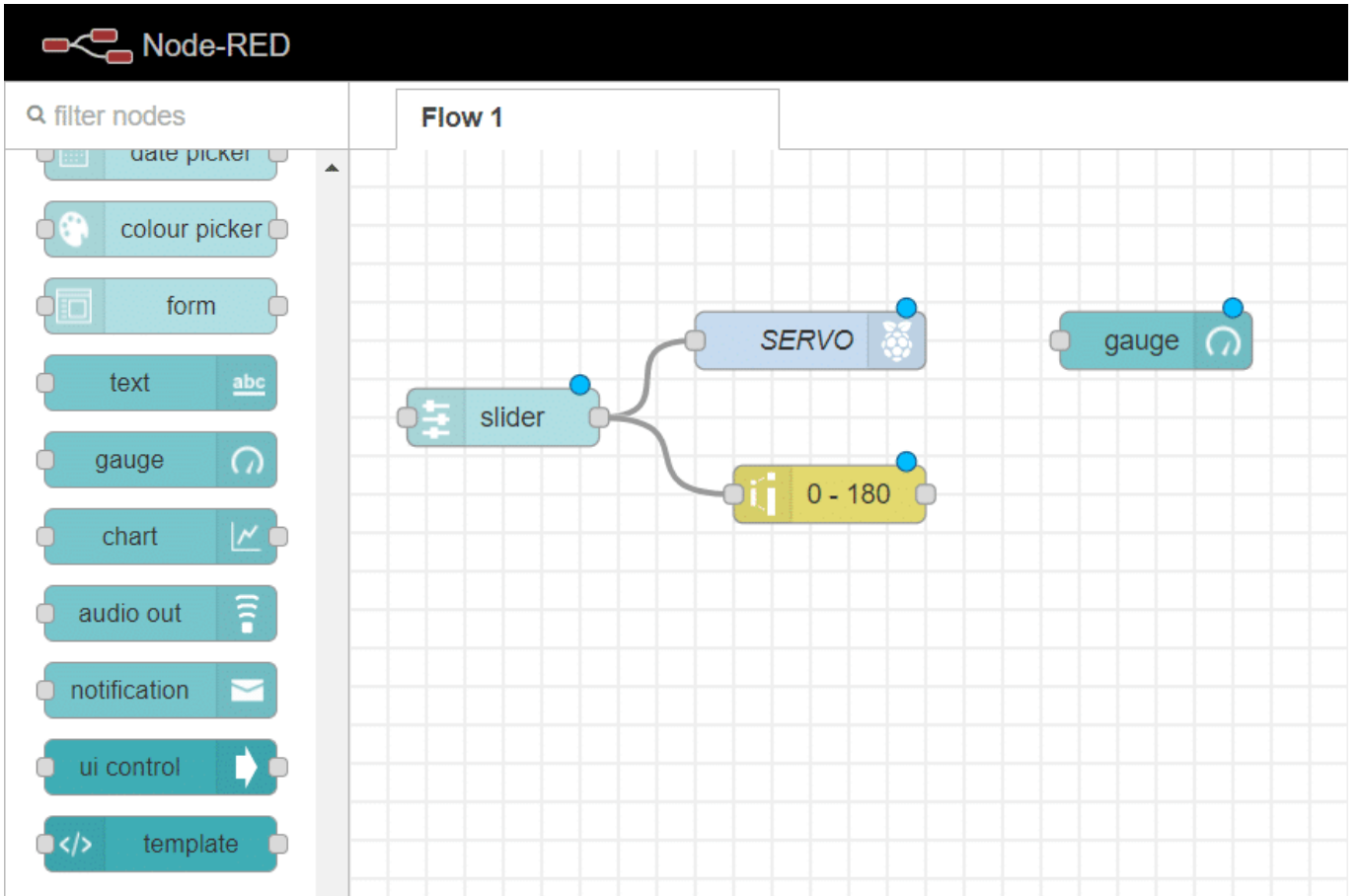
Payload	Current value
Topic	<input type="button" value="Dropdown"/> msg. topic
Name	<input type="text"/>

الآن نقوم بتوصيل العقد معا لإنشاء التدفق المطلوب. من النقط الرمادية على الجانب الأيمن من عقد الإدخال نقوم بسحب سلكه بعقدة الإخراج .



نظيف لواجهة المستخدم شكل المنقلة الذي يشير للزاوية التي يوجد عندها محرك سيرفو

نسحب "gauge" من قسم "Dashboard"



نحدد القيمة التي سيمثلها الشكل

Edit gauge node

Delete

Cancel

Done

⚙ Properties



🗪 Group

[Home] servo



📐 Size

auto

☰ Type

Gauge



📏 Label

Degree

📏 Value format

{{value}}

📏 Units

degree

Range

min

0

max

180

Colour gradient



Sectors

0

...

optional

...

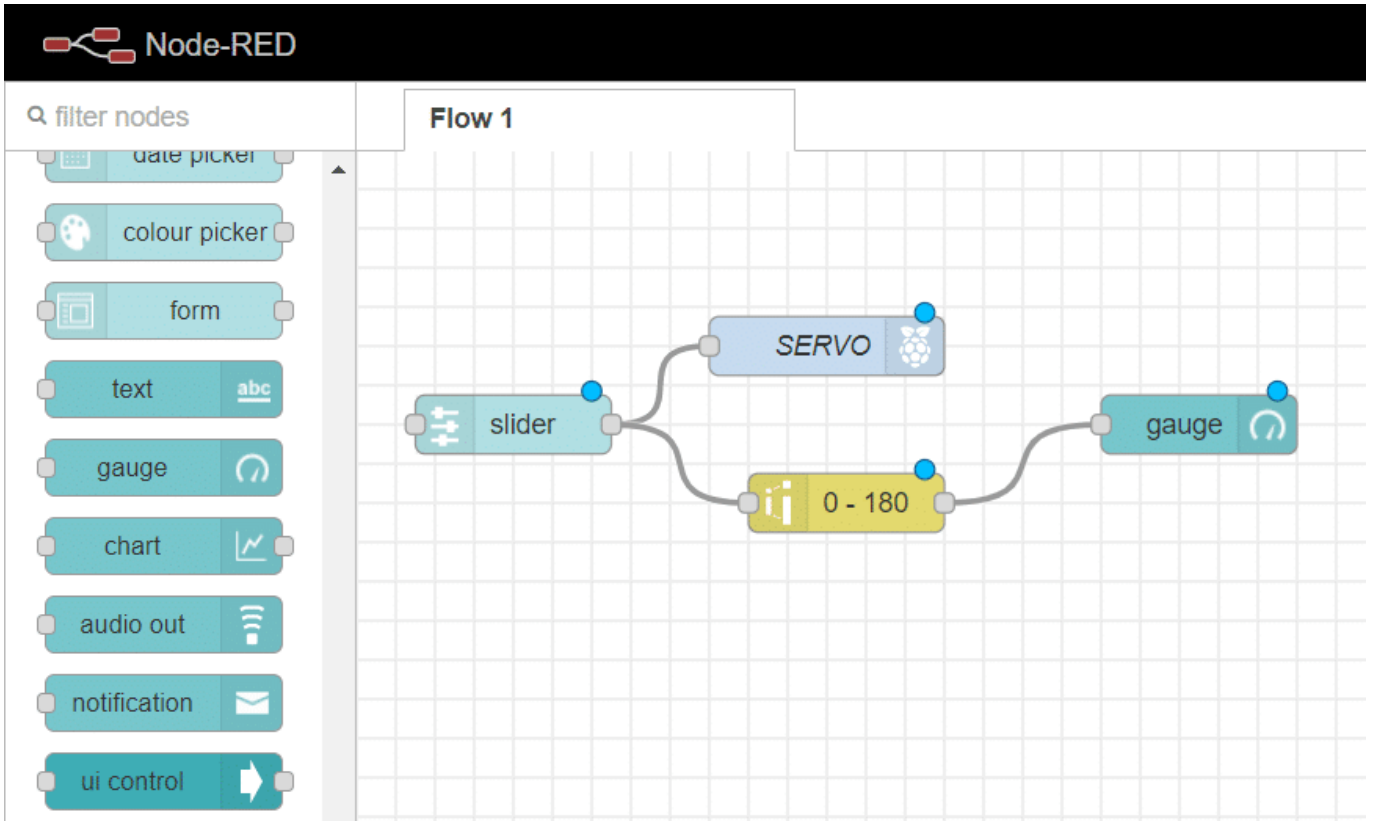
optional

...

180

🏷 Name

نوصل العقد معا لإنشاء التدفق المطلوب



أخيرًا: نشر التدفق

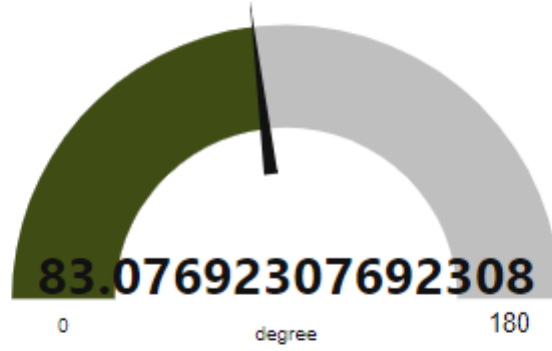
تم الانتهاء من التدفق، حتى تتمكن من نشره. انقر على زر Deploy الأحمر الكبير أعلى يمين الشاشة. ستظهر رسالة في الأعلى تقول تم نشرها بنجاح "Successfully deployed"



اكتب <http://nodered.local:1880/ui> في شريط العنوان الخاص بمتصفحك اكتب عنوان IP الخاص راسبييري باي لديك بدل من "nodered.local" لتظهر واجهة المستخدم التي قمت ببنائها لتتحكم بمحرك سيرفو

servo

Degree



PWM



يمكنك التحكم من خلال الواجهة التي أنشأتها بالنود-ريد (NodeRED) بمحرك سيرفو