

**Node-RED**

# Node-red リチカ

2022. 3. 1

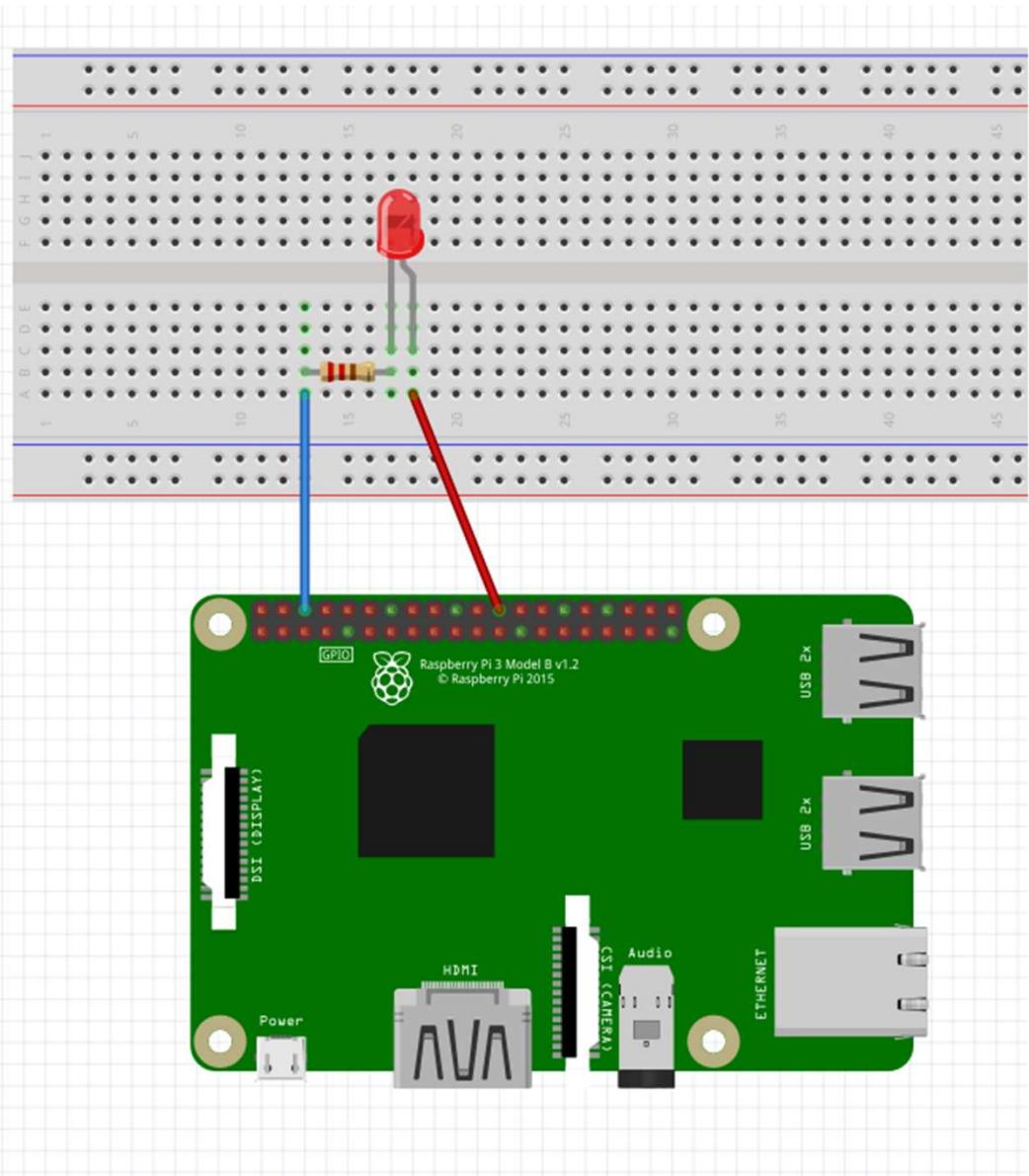
# LEDを配線

- Raspberry Pi 3 B
- ブレッドボード( 4 0 0 穴)
- LED(何色でも可)
- 抵抗(150Ω)
- ジャンパーウイヤ

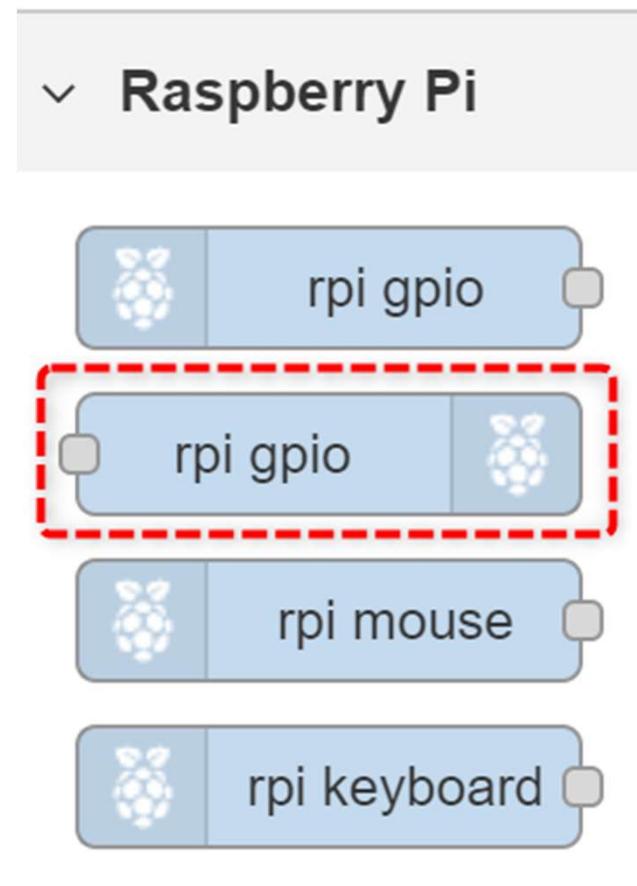
# Node-REDで制御

Node-REDを使い、LEDを点灯したり、消灯したり制御してみます。

プラス側は24番ピンに繋いでいます。



# Raspberry Pi グループノード



## ノードの設置

injectノードと、rpi gpioノードを配置します。



# GPIOノードの設定

rpi gpioノードをダブルクリックし設定を行います。

ラズパイの24番ピンに繋ぎましたので、24番を選択します。



# Injectノードの設定

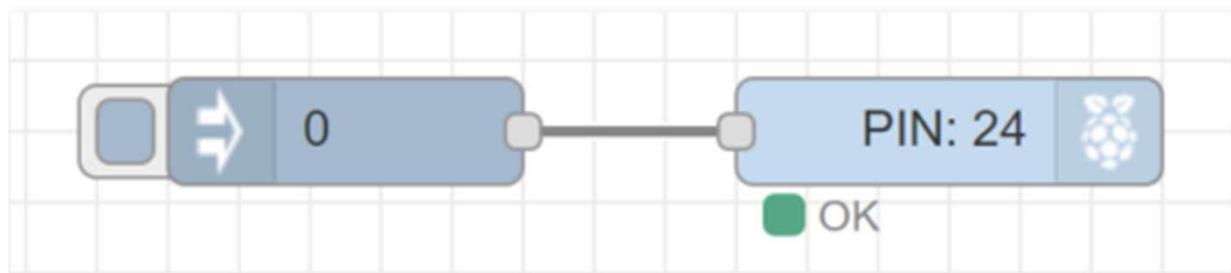
同じくinjectノードをダブルクリックし設定を行います。

ペイロードに数値の0を指定しました。



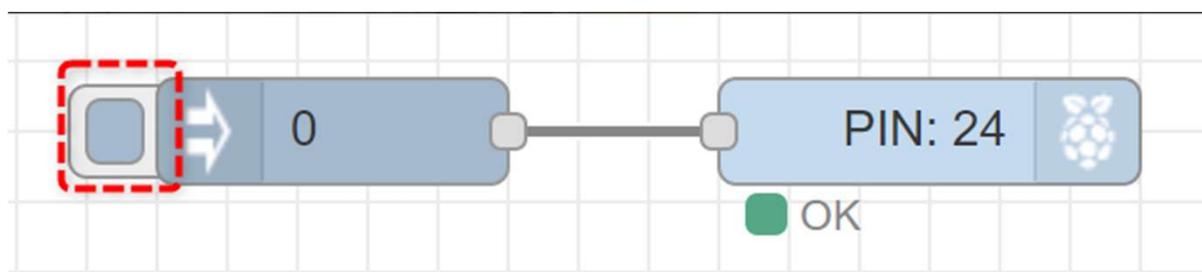
# LEDの振る舞い

デプロイすると、何もしなくてもLEDが光ります。



•[rpi gpioノードに0: LEDが消える](#)

injectノードのボタンをクリックし、rpi gpioノードに0を送ります。



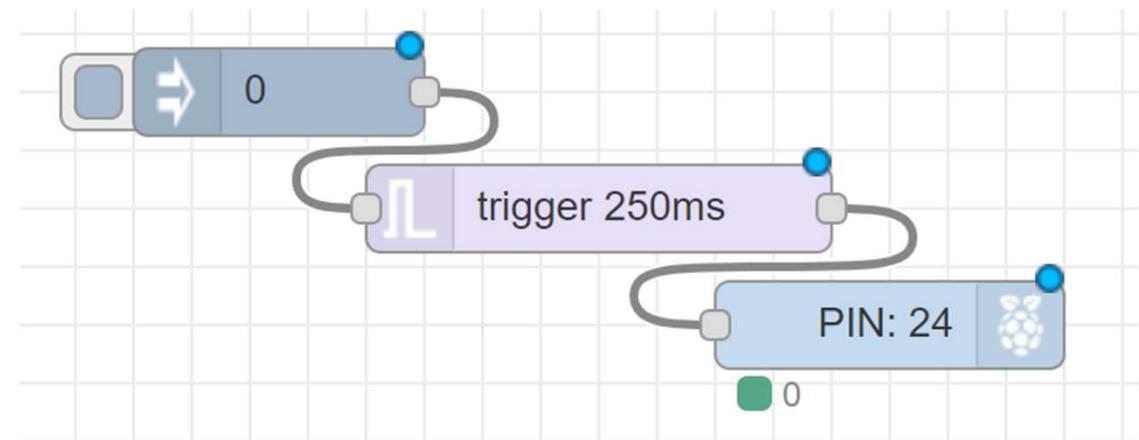
•[rpi gpioノードに1: LEDが光る](#)

# LED点滅

- 今度は、0.5秒間隔で点滅させたいと思います。  
**triggerノード**を使用します。



フローはこんな感じです。



# 点灯間隔の設定

- injectノードをダブルクリックし設定を行います。  
1秒間隔で実行するため、  
以下のように設定しました。  
ちなみにペイロードは、  
次のtriggerノードで書き  
換えるため適当で良いで  
す。



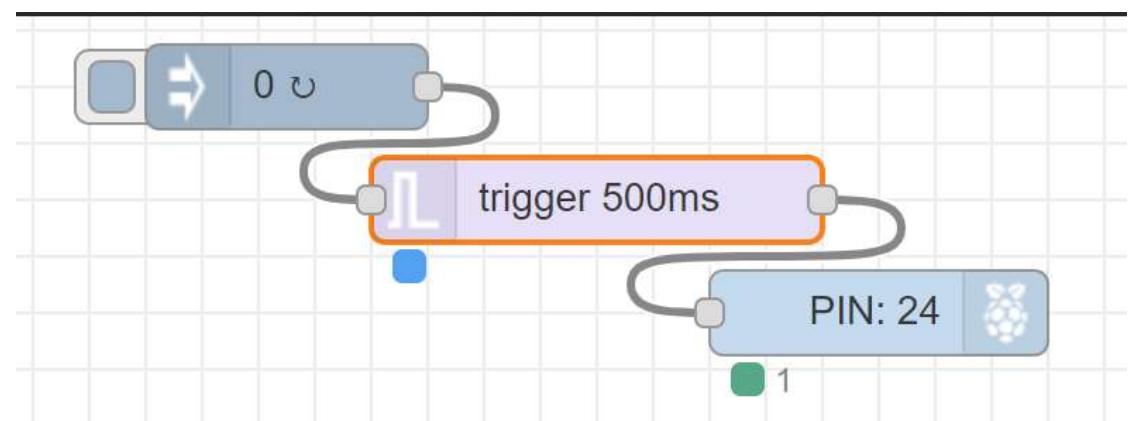
# triggerノードの設定

- triggerノードをダブルクリックし設定を行います。  
injectノードからデータを受け取ったら、1を次のノードへ送信し、500ミリ秒後に0を再送します。
- これにより、LEDを光らせた500ミリ秒後に消すといった具合になります。



## フロー図

- フローは以下のようになりました。
- デプロイして確かめましょう。



500ミリ秒ごとにLEDが点滅しました。

ピン番号を指定するため、GPIO.BOARDにしています。  
今回は24番ピンに繋ぎましたので、24を指定しています。

```
import RPi.GPIO as GPIO
import time

pin_no = 24

# BOARD: ピン番号
# BCM: GPIO番号
GPIO.setmode(GPIO.BOARD)
GPIO.setup(pin_no, GPIO.OUT)
try:
    while True:
        GPIO.output(pin_no, True)
        time.sleep(0.5)
        GPIO.output(pin_no, False)
        time.sleep(0.5)
except KeyboardInterrupt:
    pass

GPIO.cleanup()
```

## Pythonでの プログラム例

左記のPythonコードを実行します。

python3 led\_on\_off.py  
0.5秒間隔でLEDが点滅します。  
(止めるときはCtrl+Cで止める)