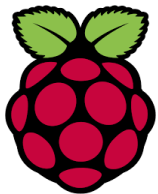


WLOラズパイ倶楽部

Grove I2Cでデバイス接続



2017年10月12日

- Raspberry PiのGPIO I2C端子にGrove I2Cデバイスを接続

 - 物理接続

 - Raspbianの設定

- Pythonで値を読んでみる

 - ライブラリの導入

 - センサー用スクリプトの導入

- Node-REDで値を読んでみる

 - カスタムノードの導入

 - node-red-dashboardでグラフ描画



・I²C (Inter-Integrated Circuit)

フィリップス社が提唱した周辺デバイスとのシリアル通信の方式で、主にEEPROMメモリICなどとの高速通信を実現する方式です。

<http://www.picfun.com/c15.html>

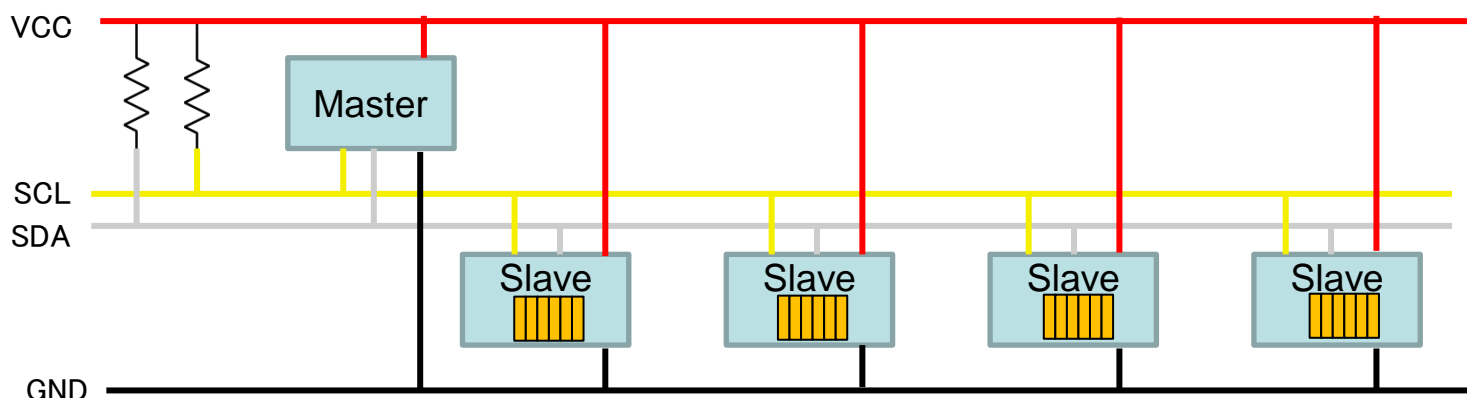
読み方: あい・すくえあ・しー

・特徴

マスタースレーブ: 1台のマスタと1台以上(最大112台)のスレーブを2線の信号線で接続

クロック同期通信: 100kbit/s のstandard mode と400kbit/s のFast modeがある

一つのスレーブ内に1バイトのレジスタアドレスを持ち、マスタから指定して読み書き可能





・Grove System

深圳を代表するハードウェアMaker企業「Seeed Studio」が
提唱するモジュール式の試作システム。

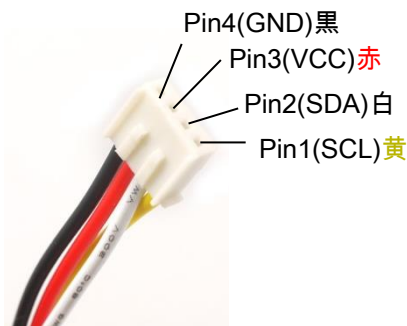
http://wiki.seeed.cc/Grove_System/

読み方:ぐろーぶ しすてむ

・特徴

標準化された4pin(2mmピッチ)のコネクタ、ケーブル、基板サイズを持つ
ベースユニットはArduino/BeagleBone/IOIO/Particle/NodeMCU/Raspberry Piに接続し
そこにDigital/Analog/UART/I2Cでデバイスを接続してシステムを構成できる

・Grove I2C ピン配置



pin	Function	Note
pin1	SCL	I2C Clock
pin2	SDA	I2C Data
pin3	VCC	Power for Grove Module, 5V/3.3V
pin4	GND	Ground

http://wiki.seeed.cc/Grove_System/#grove-i2c

<https://www.switch-science.com/catalog/list/?keyword=grove+i2c>



GROVE - I2C ハブ
432 円 在庫：多数

1 カートに追加



GROVE - I2C ADC
1,803 円 在庫：多数

1 カートに追加



GROVE - I2C タッチセンサ
2,365 円 在庫切れ

1 カートに追加



GROVE - I2C モータードライバ
2,365 円 在庫：多数

1 カートに追加



GROVE - I2C ミニモータードライバ
1,522 円 在庫：14

1 カートに追加



I2C 8x8 LEDマトリックス基板 (MFT版) ---
2,160 円 在庫：多数

1 カートに追加



Xadow - GROVEアダプタ
2,084 円 在庫：8

1 カートに追加



GROVE - ジェスチャー
1,922 円 在庫：多数

1 カートに追加



GROVE - 気圧センサ (BMP280)
1,155 円 在庫：多数

1 カートに追加



GROVE - ベースケース
2,775 円 在庫：6

1 カートに追加



GROVE - I2C 3軸ジャイロセンサ
2,775 円 在庫：多数

1 カートに追加



GROVE - I2C カラーセンサ--在庫限り
1,387 円 在庫切れ

1 カートに追加



GROVE - I2C デジタル光センサ
1,382 円 在庫：多数

1 カートに追加



GROVE - I2C OLEDディスプレイ
128x64
2,084 円 在庫：6

1 カートに追加



GROVE - I2C 三軸加速度センサ
1,382 円 在庫：多数

1 カートに追加



GROVE - 温度・気圧センサ (BME280)
2,462 円 在庫：3

1 カートに追加



ESPr8 Developer用 GROVEシールド
432 円 在庫：多数

1 カートに追加



GROVE - マルチチャネルガスセンサ
5,983 円 在庫：多数

1 カートに追加



GROVE - 高精度RTC
1,242 円 在庫：多数

1 カートに追加



GROVE - IMU 10DOF v2.0
2,192 円 在庫：多数

1 カートに追加



GROVE - I2C カラーセンサ v2.0
1,404 円 在庫：多数

1 カートに追加



GROVE - I2C 三軸加速度センサ ADXL345搭載
1,382 円 在庫：多数

1 カートに追加



HSTS-ADT7410 高精度I2C温度センサ
1,080 円 在庫：多数

1 カートに追加



I2C接続4x4キーボード
2,160 円 在庫：多数

1 カートに追加



GROVE - RTC
961 円 在庫：多数

1 カートに追加



GROVE - RGB/バックライト液晶モジュール
1,944 円 在庫：多数

1 カートに追加



Intel Joule用 GROVE シールド
1,965 円 在庫：5

1 カートに追加



GROVE - IMU 9DOF v2.0
1,868 円 在庫：多数

1 カートに追加



GROVE - 温湿度センサ (SHT31) v1.0
1,544 円 在庫：多数

1 カートに追加

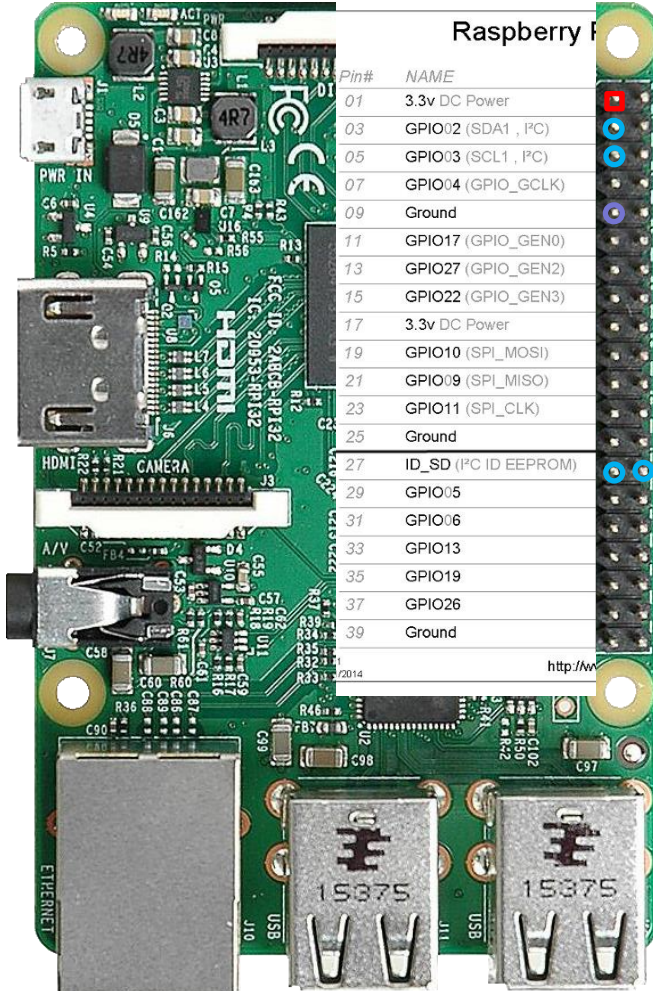


SeedStudio CAN-BUSシールド V2
2,980 円 在庫：多数

1 カートに追加

- ・2ポートのI2Cを持つが、使えるのはGPIOの3/5pinの1ポート

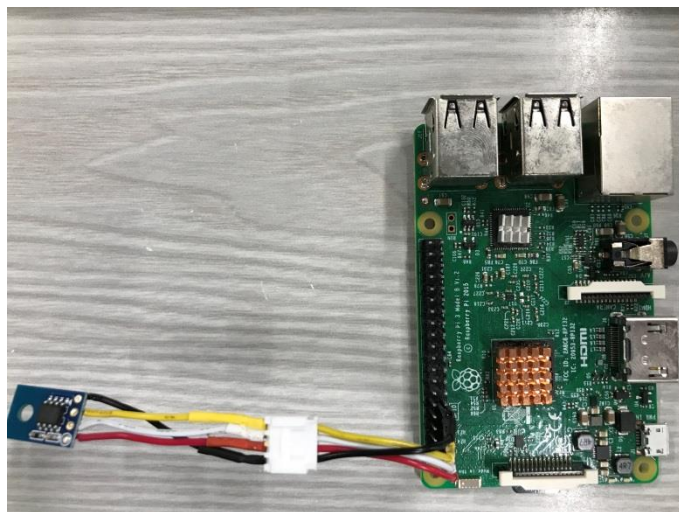
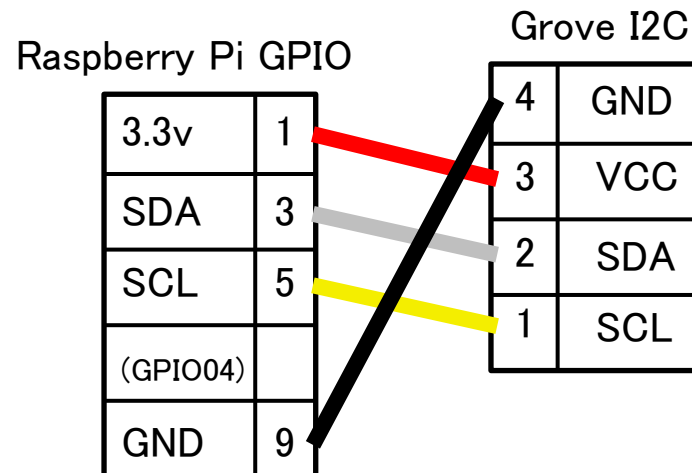
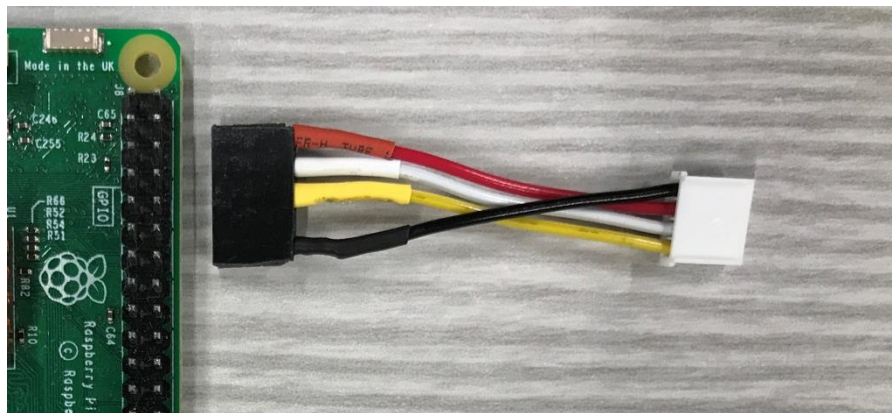
27/28pinはHATのためのEEPROM専用
3.3Vトレラント



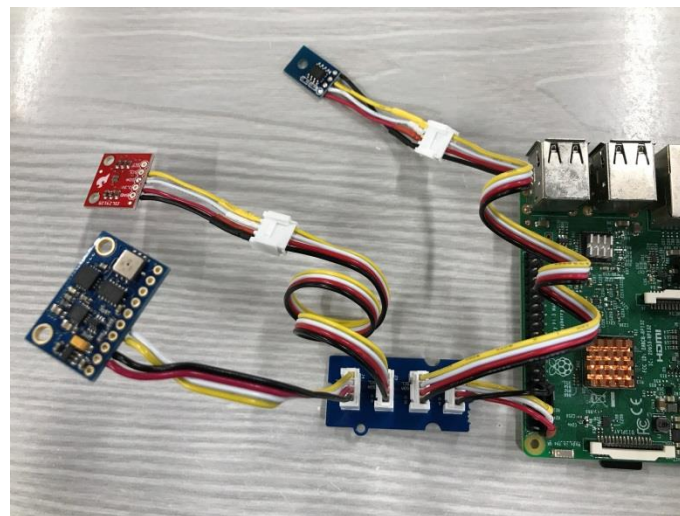
The image shows a Raspberry Pi board with a GPIO header overlay. The overlay is a table with two columns of pin information. The left column lists pins 01 through 39, and the right column lists pins 02 through 40. The table is titled 'Raspberry Pi GPIO Header'.

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1, I ² C)	DC Power 5v	04
05	GPIO03 (SCL1, I ² C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I ² C ID EEPROM)	(I ² C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

- ・「特製ケーブル」で接続。



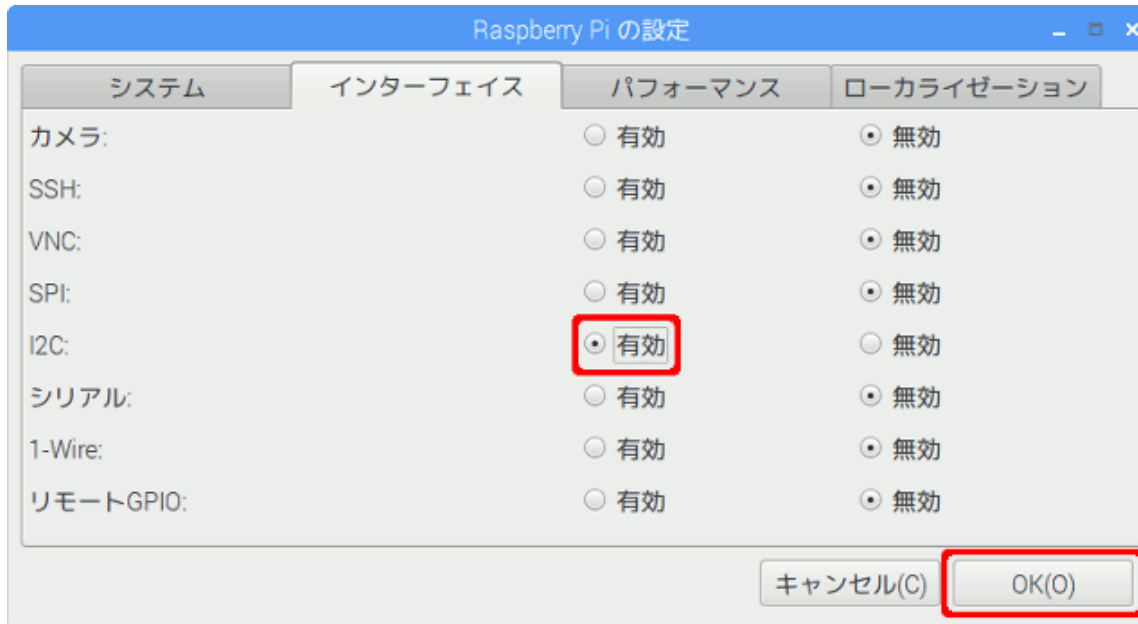
モジュール単独



I2C HUBをつかって複数のモジュール接続

・I2Cを有効にする

メニュー → 設定 → Raspberry Piの設定 → インターフェイスタブ



※再起動が必要

・パッケージの導入

i2c-tools

```
$ sudo apt install i2c-tools
```


- ・i2cdetectコマンドで確認

```
$ i2cdetect -y 1
```

```
pi@raspberrypi:~$ i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  5c  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
pi@raspberrypi:~$
```

認識されたデバイスのアドレスが表示される

※温湿度センサー AM2321はスリープモードに入ると表示されないので
何度か試して確認する必要アリ。

- ・i2cの読み書きに必要なライブラリの導入

```
$ sudo apt install python-smbus
```

- ・デバイスに応じたスクリプトを書く

温湿度センサー AM2321の場合

```
import smbus
import time

i2c = smbus.SMBus(1)
address = 0x5c

try:
    i2c.write_i2c_block_data(address, 0x00, [])
except:
    pass

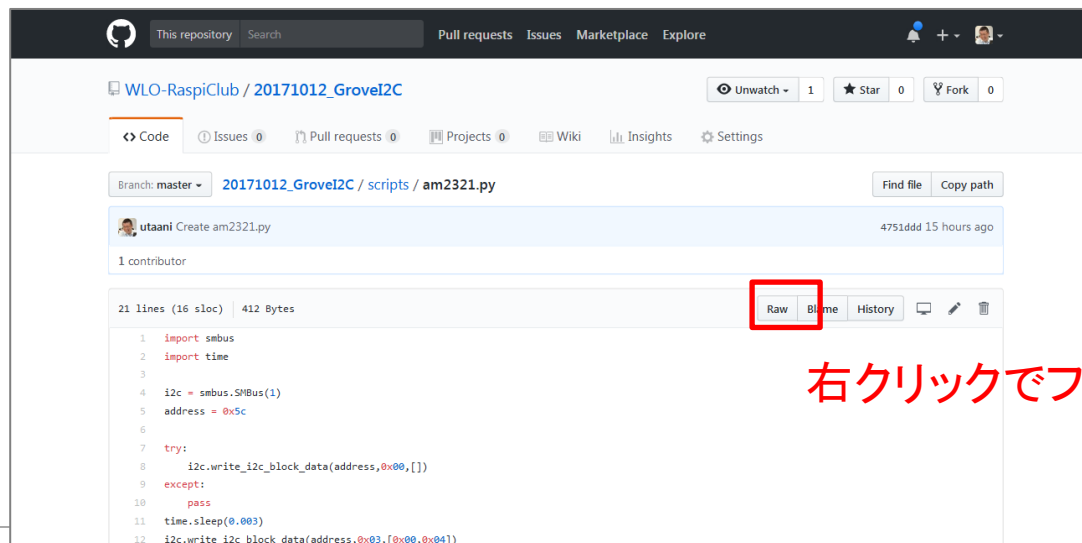
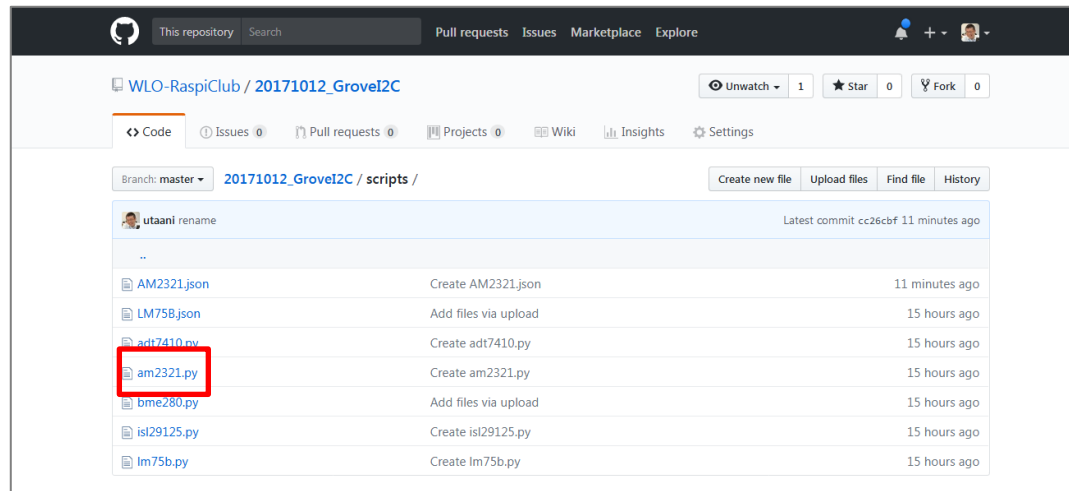
time.sleep(0.003)
i2c.write_i2c_block_data(address, 0x03, [0x00, 0x04])

time.sleep(0.015)
block = i2c.read_i2c_block_data(address, 0, 6)
hum = float(block[2] << 8 | block[3])/10
temp = float(block[4] << 8 | block[5])/10

print('humidity:' + str(hum))
print('temp:' + str(temp))
```

・スクリプトをgithubからダウンロード

https://github.com/WLO-RaspiClub/20171012_Grovel2C/tree/master/scripts



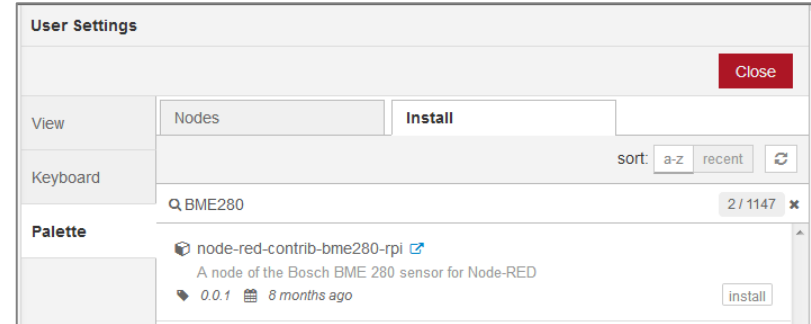
右クリックでファイル保存

<http://git.io/wlopi>

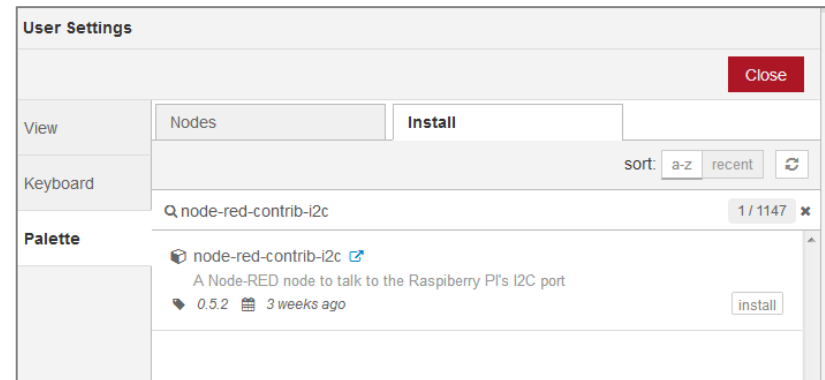
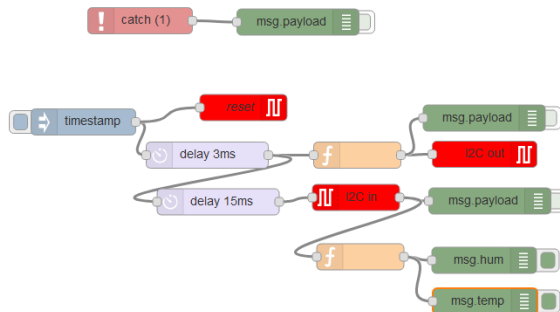
・3つの方法

センサー専用のカスタムノードがある場合

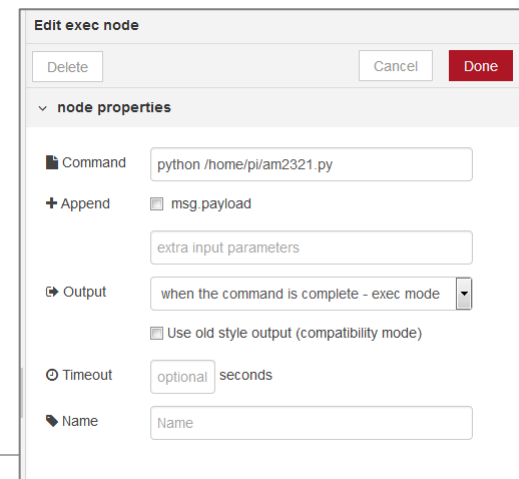
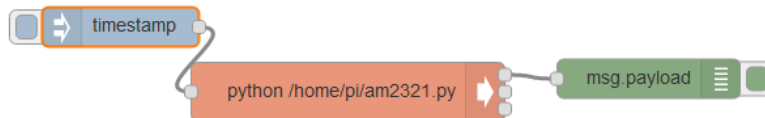
例:BME280



汎用のI2Cドライバを使ってFlowを書く

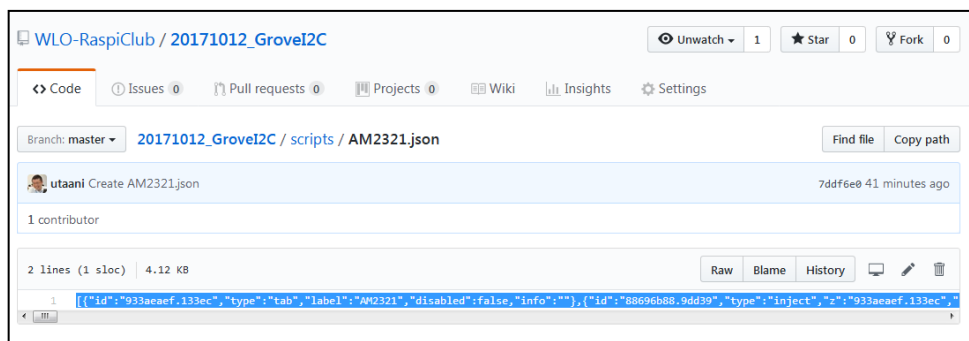
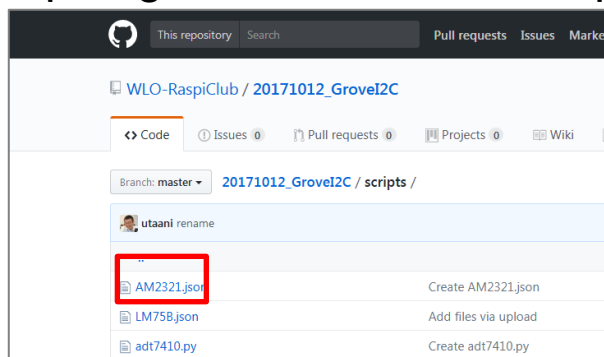


EXECノードを使って
Pythonスクリプトを呼び出す



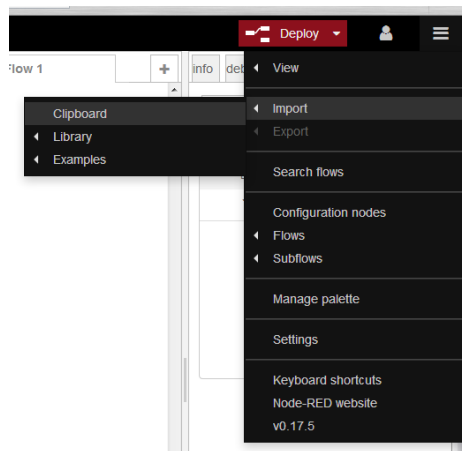
・スクリプトをgithubからコピー

https://github.com/WLO-RaspiClub/20171012_Grovel2C/tree/master/scripts

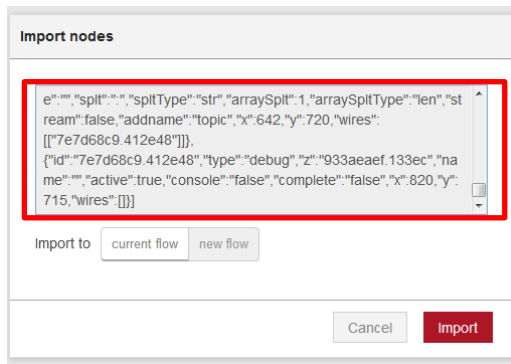


jsonをトリプルクリックで全選択して、コピー

- ・スクリプトをFlowEditorにペーストしてインポート



右上のハンバーガーメニューから
Import→クリップボードを選択
(日本語だと「読み込み」→クリップボード)



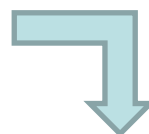
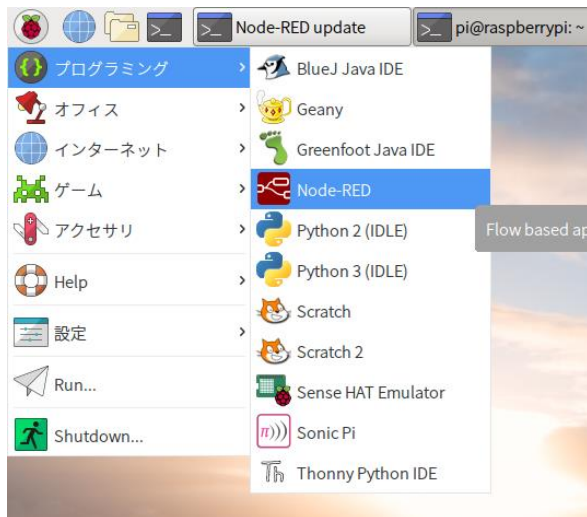
jsonをペーストしてImport

新しいタブにFlowが読み込まれる



・Raspberry Piではじめる

Raspbian 2015-11-23版より、Node-REDが標準でインストールされている。



コンソールが開いて
サービスが起動する

```
Node-RED console
ファイル(F) 編集(E) タブ(T) ヘルプ(H)

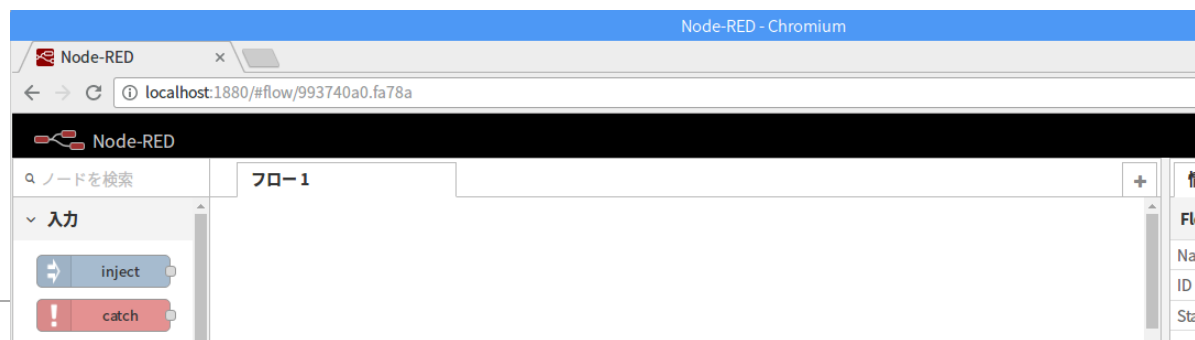
Start Node-RED

Once Node-RED has started, point a browser at http://192.168.0.20:1880
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.
Started Node-RED graphical event wiring tool..
27 Jul 08:48:04 - [info]
Welcome to Node-RED
=====
27 Jul 08:48:04 - [info] Node-RED version: v0.17.5
27 Jul 08:48:04 - [info] Node.js version: v6.11.1
27 Jul 08:48:04 - [info] Linux 4.9.35-v7+ arm LE
27 Jul 08:48:06 - [info] Loading palette nodes
27 Jul 08:48:10 - [info] Settings file : /home/pi/.node-red/settings.js
```



ブラウザで
http://localhost:1880/
にアクセス

<http://git.io/wlopi>

・Raspberry PiにおけるNode-REDの更新

```
pi@raspberrypi: ~  
ファイル(F) 編集(E) タブ(T) ヘルプ(H)  
pi@raspberrypi:~$ update-nodejs-and-nodered  
  
This script will remove any pre-installed versions of node.js and Node-RED  
and replace them with node.js 6.x LTS (boron) and the latest Node-RED from Npm.  
To do this it runs commands as root - please satisfy yourself that this will  
not damage your Pi, or otherwise compromise your configuration.  
  
Doing this may also be 'a bad thing' if you have installed lots of extra nodes.  
Especially if they have any native binary component. Some nodes in your  
~/node-red directory will probably need to be re-installed afterwards, some  
may need you to run npm update, and some may require you to run npm rebuild.  
  
There may be a period of frustration ahead to get back to where you were...  
  
Are you really, really sure you want to do this ? y  
rm: '/var/log/nodered-install.log' を削除できません: そのようなファイルやディレ  
クトリはありません  
  
This can take 20-30 minutes on a Pi 1 - please wait.  
  
Stop Node-RED ✓  
Remove old version of node.js ✓  
Install node.js for Armv7 ✓ Node v6.11.1 Npm 3.10.10  
Clean npm cache ✓  
Install Node-RED core ✓  
Install extra nodes ✓  
Install serialport node ✓  
Npm rebuild existing nodes ✓  
Add menu shortcut ✓  
Update systemd script ✓  
Update update 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 2017年 7月 27日 木曜日 08:34:55 JST - Finished 2017年 7月 27日 木  
曜日 08:45:12 JST  
pi@raspberrypi:~$
```

Raspbianで標準サポートされているミドルウェアが古い場合があり、そのままでは安定して動作しない。

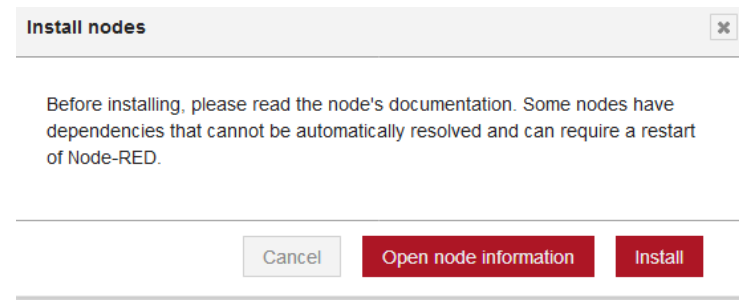
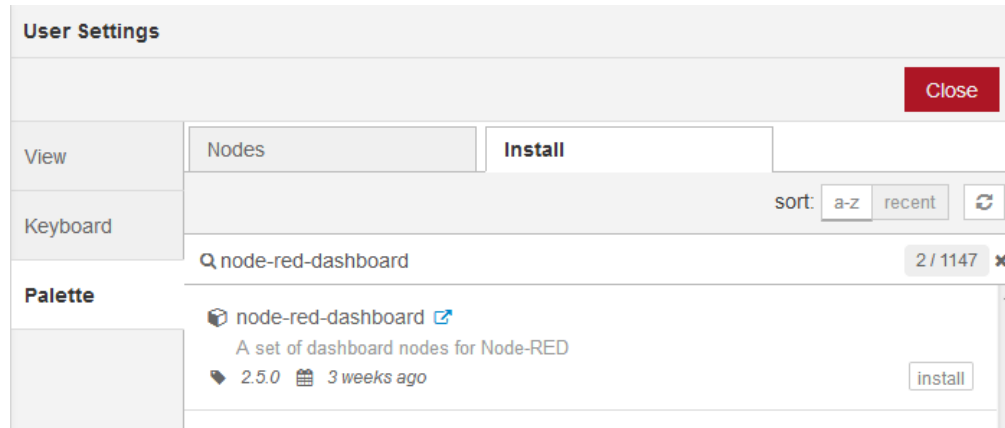
⇒更新用スクリプト

「update-nodejs-and-nodered」を用いてNode.jsとNode-REDを最新版に更新する(sudo不要、所要時間約12分)

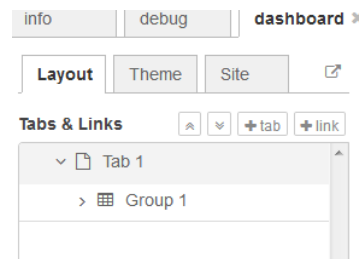
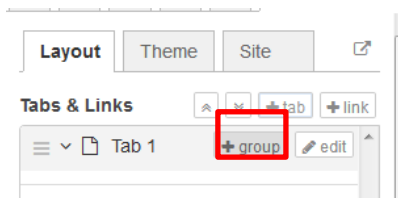
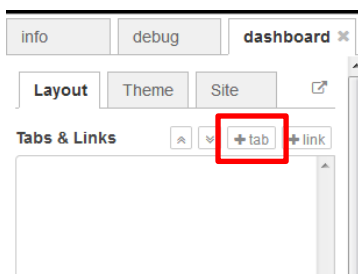
途中で止まると再実行できない
以下の手順で復旧できる

```
$ wget https://git.io/wloup2  
$ bash wloup2
```

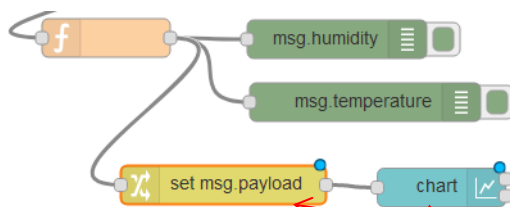
・node-red-dashboardのインストール



・dashboardのTabとgroupを作る

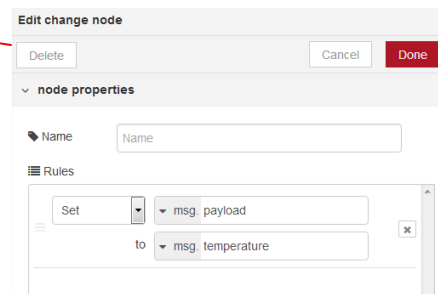
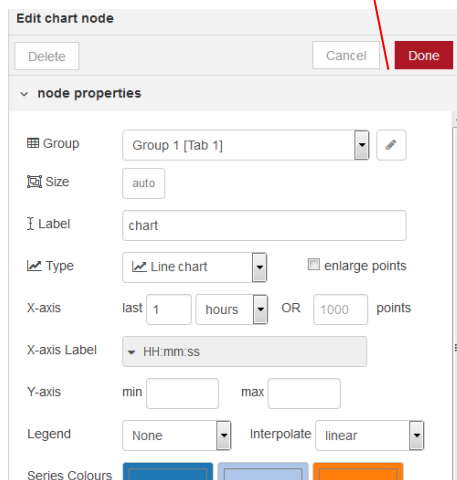


・chartノードへのプロパティをchangeノードで変換



chartノードはmsg.payloadのデータをグラフ化するのでプロパティを複製する

・chartノードの設定



- dashboardにアクセスしてみる

<http://IPアドレス:1880/ui>

