Spectrum Technology

V1.0 2021/8/19

抜粋版

UPS電源開発キット

～ソーラーパネル＋ポータブル電源を使ってUPS電源化し、自宅のパソコンで常用、CO2排出削減に貢献～

実践編

AWS IoT

Raspberry Pi

スペクトラム・テクノロジー株式会社

https://spectrum-tech.co.jp

sales@spectrum-tech.co.jp

all rights reserved 2021 spectrum technology co.

Spectrum Technology

UPS電源開発キット　目次

Pi運用マニュアル　ページ

1. RaspberryPiについて　3
2. Linux基本コマンド　4
3. 基本操作　5
4. 日常運用（ウイルススキャン、更新）　6

AWS　IoT開発　ページ

① メニュー　8
② AWS設定　9
③ AWS　IoT設定　16
④ UPS電源

- 全体構成　29
- 接続構成　30
- 単体試験　32
- AWS IoTルール設定　33
- プログラム設定　39
- IoT折り返し試験　45
- iot総合試験　46
- DynamoDB確認　47
- リアルタイム表示設定、確認　49
- IAMユーザ追加　51
- CSVデータ取得　55
- Cloudwatch使い方　57

抜粋版のため
ページと一致しま
せん

all rights reserved 2021 spectrum technology co.

RaspberryPi運用マニュアル

Raspberry Pi

1. Raspberry Piについて

 既に全世界で1000万台以上販売された手のひらサイズのコンピュータです。LinuxベースのRasbianOSで動作しております。

2. Linux基本コマンド

① システム関係

- 起動：電源を入れると自動で起動します。
- 再起動：$ reboot

  又は、menu>shutdown>reboot；左上のメニューから
- 終了： $ shutdown

  又は、menu>shutdown>shutdown；左上のメニューから
- ログアウト $ exit

  又は、menu>shutdown>logout；左上のメニューから
- 日本語／英語の入力切替：キーボードのCTLとjを同時に押します（コントロール：左下とj）

all rights reserved 2021 spectrum technology co.

RaspberryPi運用マニュアル

Raspberry Pi

2. Linux基本コマンド

② ディレクトリ操作、コピー、移動、削除

pi@raspberrypi:~$ cd /home/pi/Documents　ディレクトリの切り替え
pi@raspberrypi:/home/pi/Documents$ ls　ファイルとディレクトリの表示（表示したら操作したいファイルを右クリックでコピペして操作します
pi@raspberrypi:~$ cp ファイル名　ディレクトリ　配下のディレクトリのファイルを別のディレクトリへコピー
pi@raspberrypi:~$ mv ファイル名　ディレクトリ　配下のディレクトリのファイルを別のディレクトリへ移動
pi@raspberrypi:~$ sudo rm ファイル名　ファイルの削除
便利な機能　rm –help　コマンドのオプションが分からない場合は、ヘルプで問い合わせる。すべてのコマンド共通(マイナスを２個とhelp)

③ ユーザ権限、プロセス他

pi@raspberrypi:~ $ su -　スーパーユーザ（root)に切り替え、パスワードを入力
pi@raspberrypi:~$ ps a　現状の動いているプロセスを表示
pi@raspberrypi:~$ kill　特定のプロセスを強制終了
pi@raspberrypi:~$ sudo apt-get install pkg　パッケージのインストールなどに使用
pi@raspberrypi:~$ date　日付、時間の設定を行います。
pi@raspberrypi:~$ sudo leafpad /etc/network/interfaces　インタフェースに記述してい内容を変更します。Viよりも使いやすいです。

④ モジュール、usb、メモリ、HDDなどの表示

pi@raspberrypi:~$ lsmod　linuxのモジュールリスト表示
pi@raspberrypi:~$ lsusb　usbのデバイス表示
pi@raspberrypi:~$ free –mt　メモリ使用状態表示
pi@raspberrypi:~$ df　HDD（マイクロSD）の使用状態表示

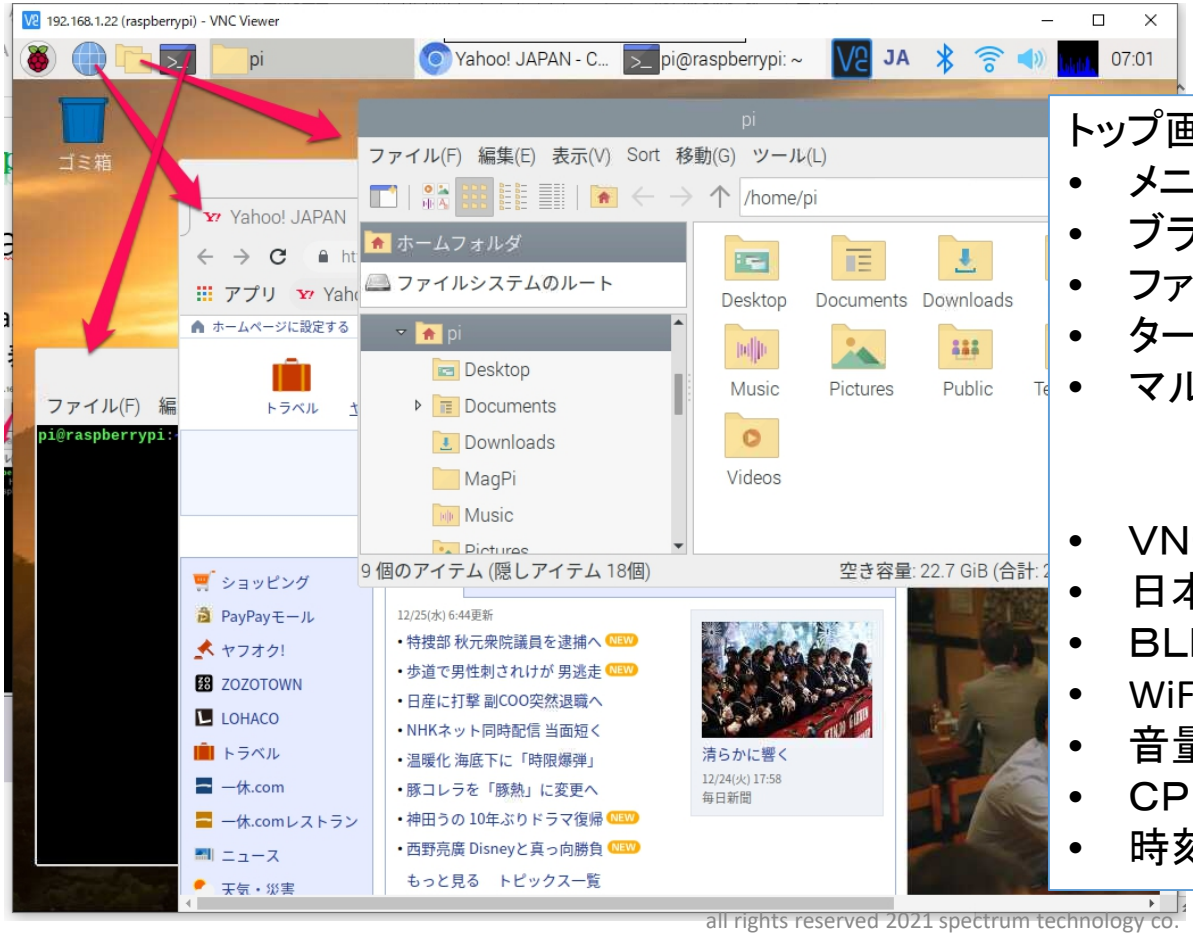
all rights reserved 2021 spectrum technology co.

RaspberryPi運用マニュアル

3.　Raspberry Piの基本操作

①　表示画面と内容

デスクトップ上によく使うコマンド.txtがあります。
コピペして使ってください



トップ画面（上段のタスクバーで選択）

- メニュー
- ブラウザ
- ファイルマネジャ
- ターミナル
- マルチ画面選択

- VNC
- 日本語入力
- BLE
- WiFi
- 音量
- CPU使用率
- 時刻

all rights reserved 2021 spectrum technology co.

RaspberryPi運用マニュアル

4. 日常運用

① セキュリティ対策（アンチウイルス更新、スキャン）

- アンチウイルス対策として無料のclamAVをインストールしてます。
- 手動での運用を基本としてます。

パターンファイル更新
手動スキャン時に更新されます
手動でスキャン
$ sudo clamscan --infected --remove --recursive
自動化可能ですが、バックグランドで重くなる可能性大。コマンド入力後約5分位かかります。

```
pi@raspberrypi: ~
ファイル(F)  編集(E)  タブ(T)  ヘルプ(H)
ERROR: /var/log/clamav/freshclam.log is locked by another
ERROR: Problem with internal logger (UpdateLogFile = /var/
og).
root@raspberrypi:~# leafpad /etc/clamav/freshclam.conf
/clamav/freshclam.log
root@raspberrypi:~# freshclam
t Fri J[redacted]
main.cvd is up to date (version: 57, sigs: [redacted] mishh
ammer)
daily.cvd is up to date (version: 21862, sigs: 394456, f-level: 63, builder: neo
)
bytecode.cvd is up to date (version: 283, sigs: 53, f-level: 63, builder: neo)
root@raspberrypi:~# clamscan --infected --remove --recursive

Known viruses: 4607906
Engine version: 0.99.2
Scanned directories: 264
Scanned files: 2063
Infected files: 0
Data scanned: 61.31 MB
Data read: 49.02 MB (ratio 1.25:1)
Time: 71.844 sec (1 m 11 s)
root@raspberrypi:~#
```

手動でスキャン

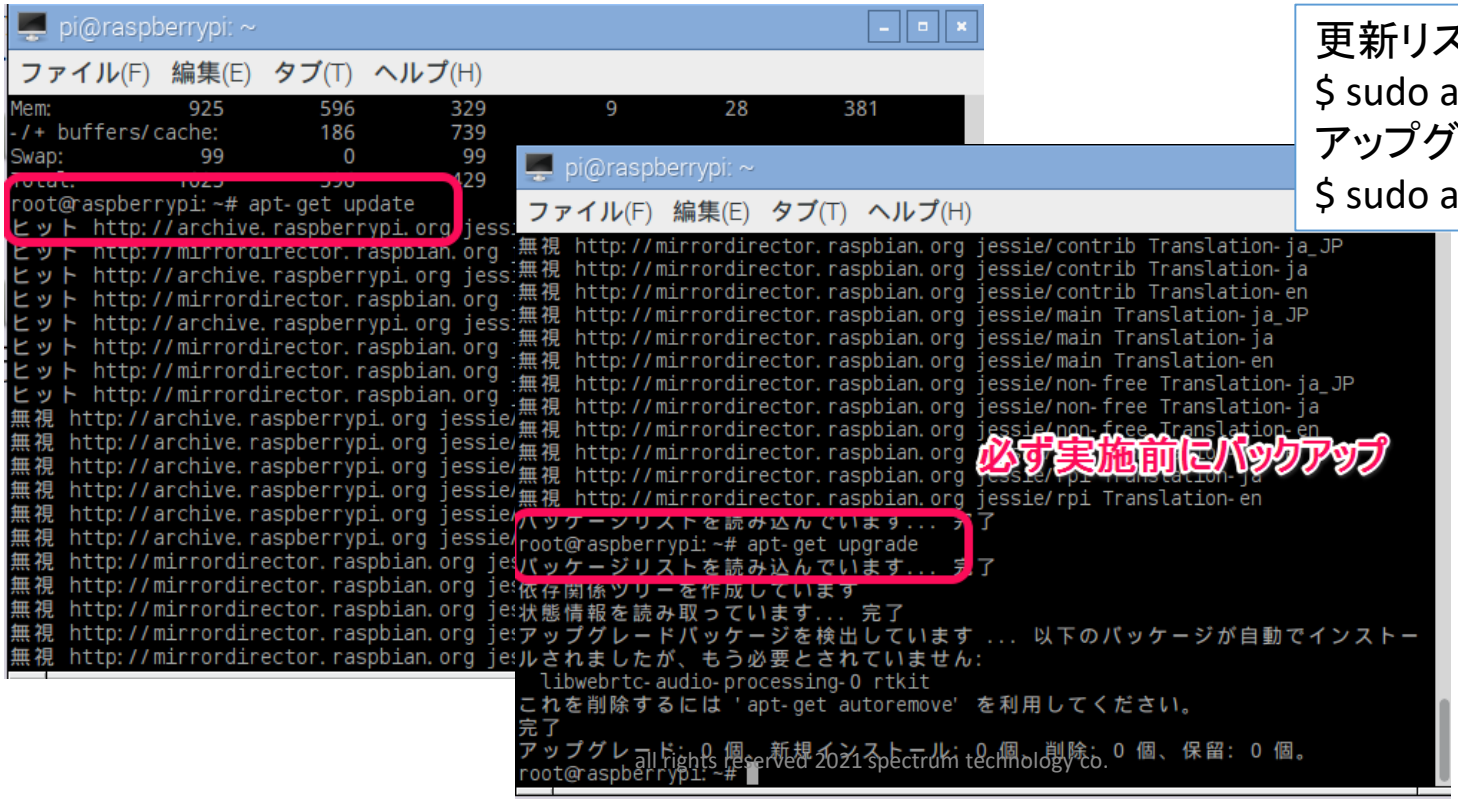
all rights reserved 2021 spectrum technology co.

RaspberryPi運用マニュアル

4. 日常運用

② インストール済パッケージの更新リスト、アップグレード

- Linuxの場合は、頻繁に更新が発生します。アップグレードを定期的に実施してください。
- 更新前には、バックアップを取ることをお勧めします。特にアップグレードはまれに動作不良、戻せない状態が発生します。自己責任で実施してください。



更新リスト取得
$ sudo apt-get update
アップグレード実施
$ sudo apt-get upgrade

all rights reserved 2021 spectrum technology co.

AWS　IoT開発
①．メニュー

- AWS設定
  AWS　IoTを実施するためのAWSの設定
  AWSの詳細説明は以下のURLへ
  https://aws.amazon.com/jp/cloud/?nc2=h_l2_cc
- AWS　IoT設定
  AWS　IoTのデバイス、証明書、ポリシー、ルールなどの設定を行います。
  開発者ガイトは以下のURLへ
  https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/what-is-aws-iot.html

（参考）

- Githubなど
  - https://github.com/aws/aws-iot-device-sdk-python3（一部のプログラムはTLSエラーで動きません）
  - https://github.com/eclipse/paho.mqtt.python3
- MQTT
  - https://www.ibm.com/developerworks/jp/iot/library/iot-mqtt-why-good-for-iot/（IBMの資料でわかりやすい）

all rights reserved 2021 spectrum technology co.

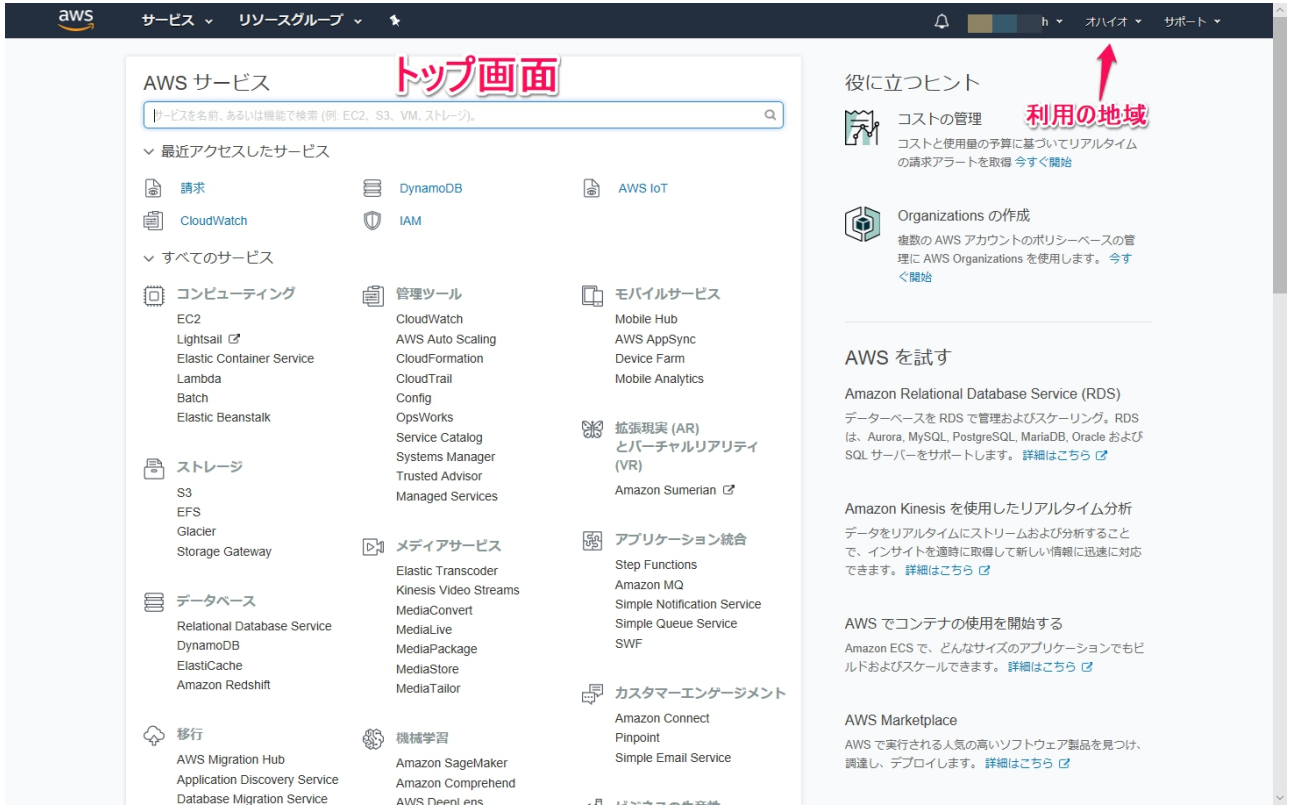
AWS　IoT開発
②. AWS設定

- 手順1：アカウントの作成
  - https://aws.amazon.com/jp/register-flow/
  - 必要なメールアドレス、パスワードなどを入力します。

all rights reserved 2021 spectrum technology co.

AWS　IoT開発
②. AWS設定

- AWSトップ画面
  - 利用する場合に、地域を意識して設定してください。地域毎に料金が変わったり、利用できるサービスが限定されている場合があります。

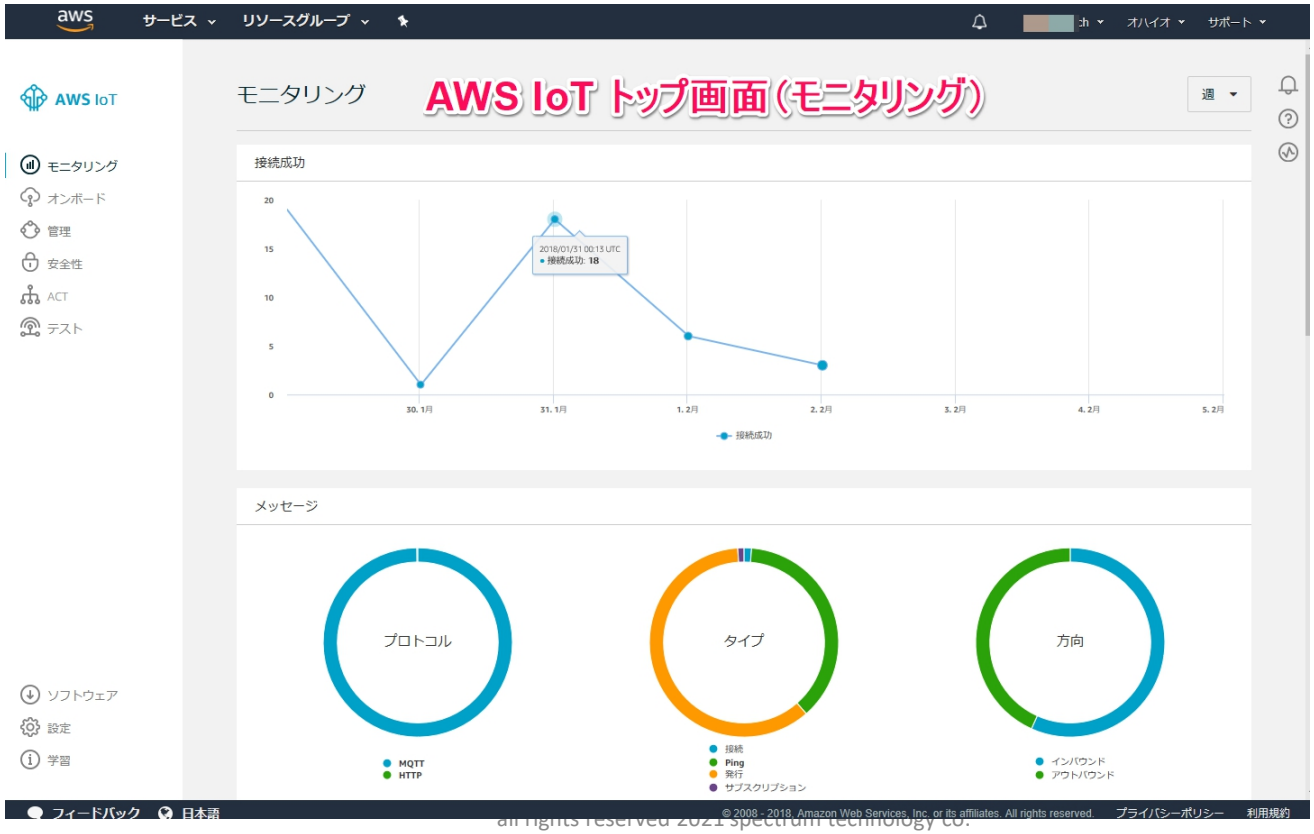


all rights reserved 2021 spectrum technology co.

AWS　IoT開発
③. AWS　IoT設定

AWS IoT Core

- 画面説明1
  - AWSトップ画面からAWS　IoTのサービスを選択します。
  - モニタリング：接続状況がひとめで分かります

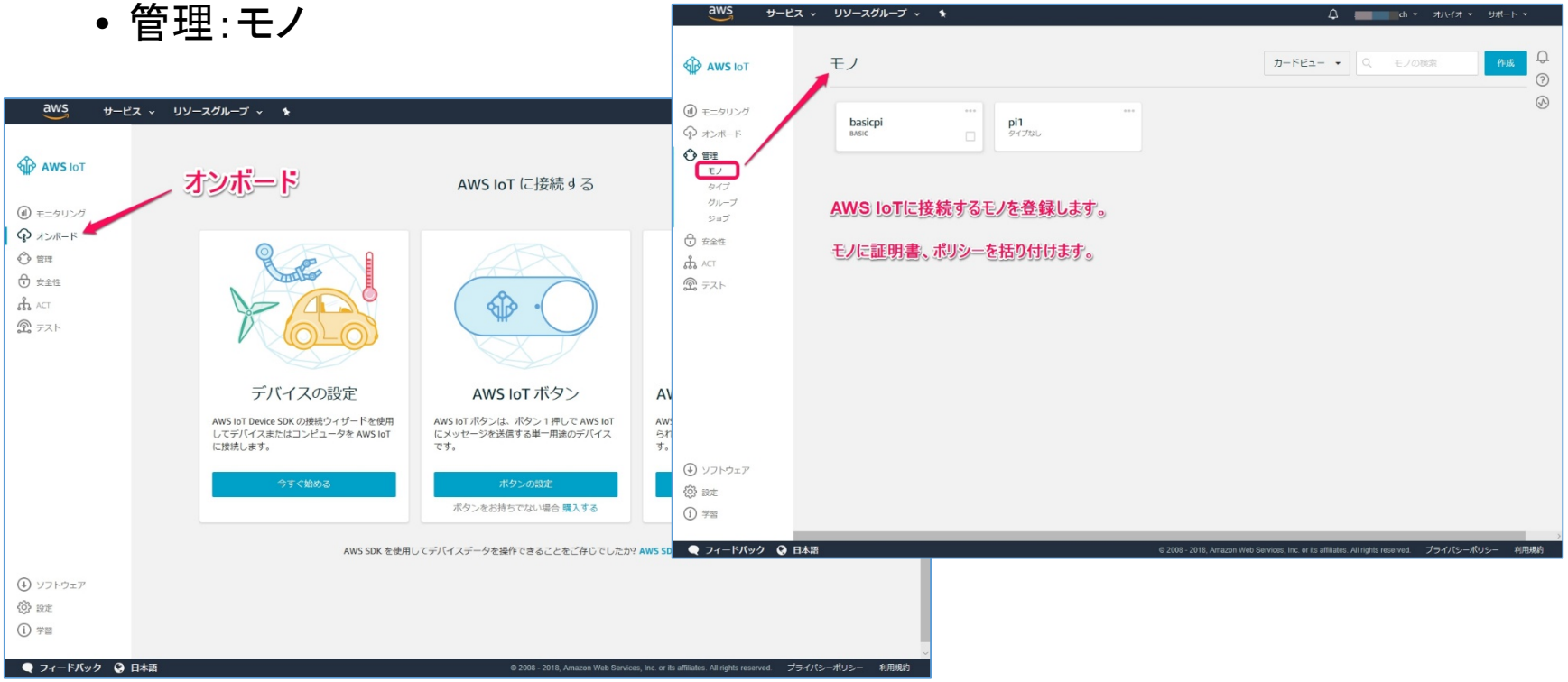


all rights reserved 2021 spectrum technology co.

AWS　IoT開発
③. AWS　IoT設定

AWS IoT Core

- 画面説明2
  - オンボード：特に使用しません
  - 管理：モノ



all rights reserved 2021 spectrum technology co.

AWS IoT開発
③. AWS IoT設定

AWS IoT Core

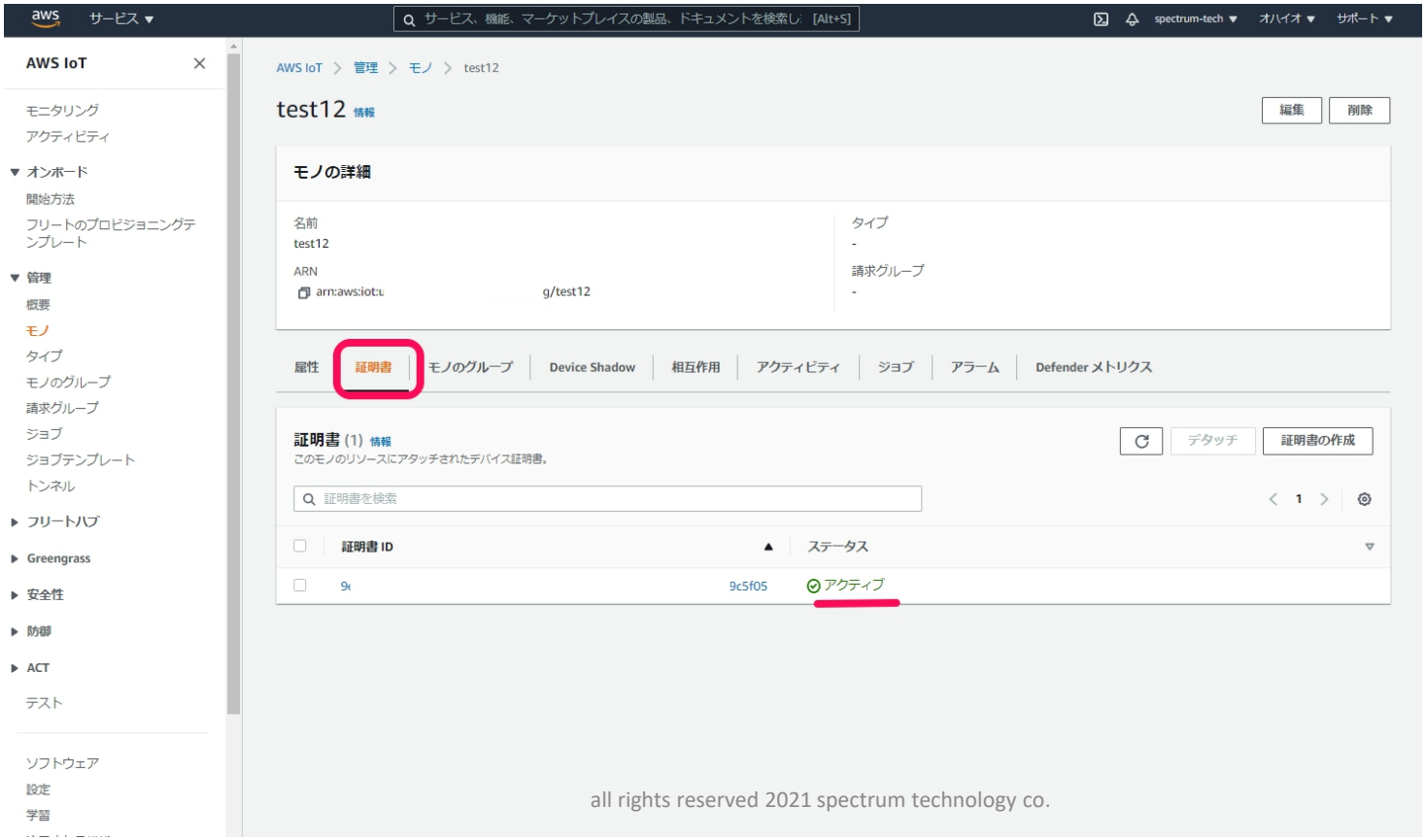
- モノの登録2
  - 単一のモノを作成するを押します。

all rights reserved 2021 spectrum technology co.

AWS IoT開発
③. AWS IoT設定

AWS IoT Core

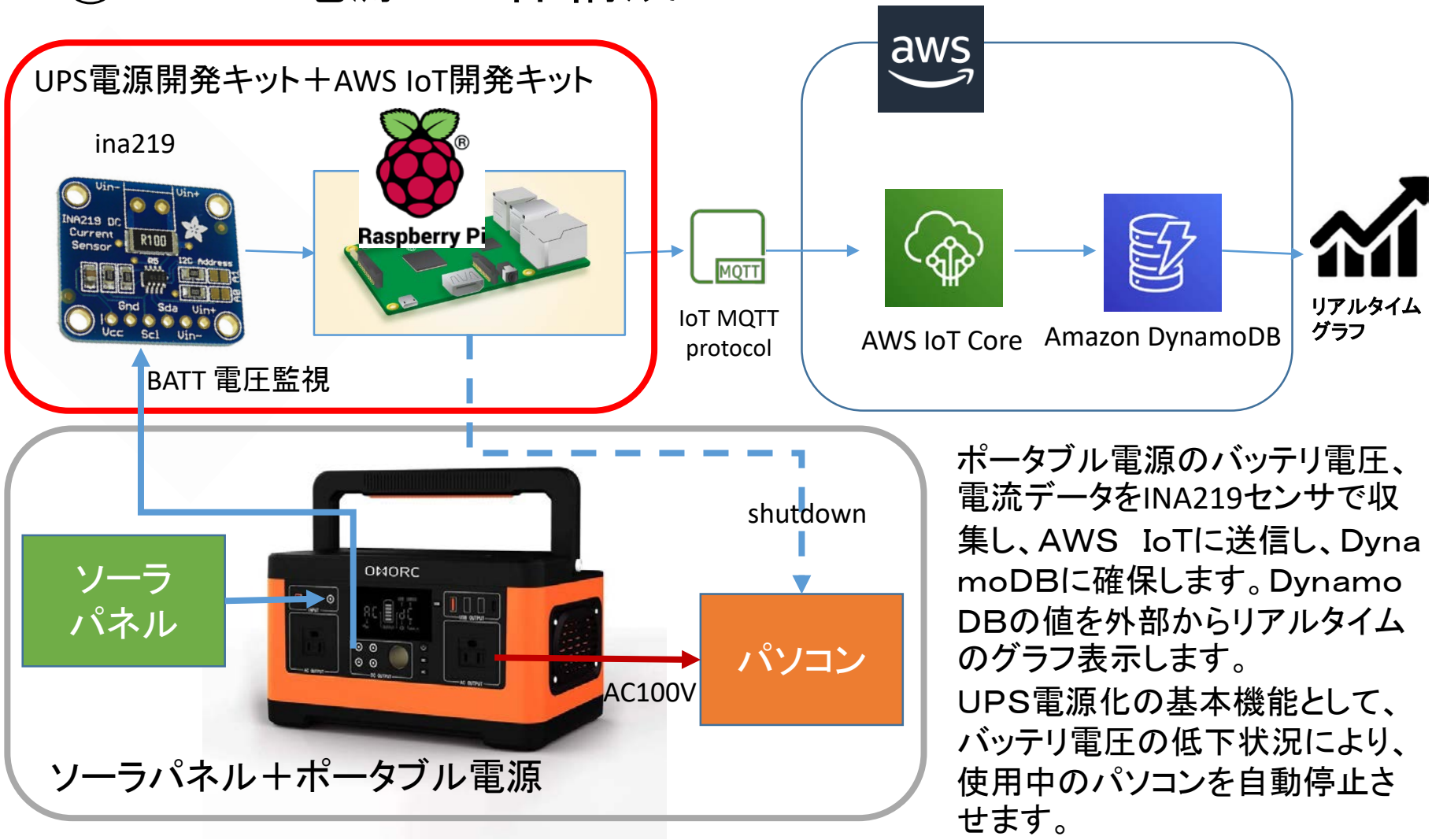
- 証明書確認
  - モノ＞作成したモノ＞
  - 証明書タグでアクティブになっていることを確認



all rights reserved 2021 spectrum technology co.

Spectrum Technology

AWS IoT開発
④ UPS電源 全体構成



ポータブル電源のバッテリ電圧、電流データをINA219センサで収集し、AWS IoTに送信し、DynamoDBに確保します。DynamoDBの値を外部からリアルタイムのグラフ表示します。
UPS電源化の基本機能として、バッテリ電圧の低下状況により、使用中のパソコンを自動停止させます。

all rights reserved 2021 spectrum technology co.

AWS IoT開発

④ UPS電源

AWS IoT Core

```
$ cd /home/pi/Documents/solar
$ python3 ina219_simpletest.py
```

ina219単体試験

- Piコンソールでina219_simpletest.pyを動作させます。
- 電圧電流が出力されます。
- 出力されない場合は、I2Cの設定を確認。0x40に値が表示されれば正常。

$ i2cdetect -y 1

pi@raspberrypi: ~

ファイル(F) 編集(E) タブ(T) ヘルプ(H)

```
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: 40 -- -- -- -- -- -- -- 48 -- -- -- -- -- -- --
50: [redacted] -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- 76 --
pi@raspberrypi:~ $
```

pi@raspberrypi: ~/Documents/solar

ファイル(F) 編集(E) タブ(T) ヘルプ(H)

```
0eb1b65f7f-certificate.pem.crt   ina219_mqtt_publish.py       myssh_win.sh
0eb1b65f7f-private.pem.key       ina219_mqtt_publish_ups.py   root-CA.crt
0eb1b65f7f-public.pem.key        ina219_simpletest.py
ct_mqtt_publish_ups.py           mqtt_subscribe_json.py
pi@raspberrypi:~/Documents/solar $ python3 ina219_simpletest.py
ina219 test
Config register:
  bus_voltage_range:    0x1
  gain:                 0x3
  bus_adc_resolution:   0x3
  shunt_adc_resolution: 0x3
  mode:                 0x7

Voltage (VIN+) :  9.333   V
Voltage (VIN-) :  9.332   V
Shunt Voltage  :  0.00070 V
Shunt Current  :  0.0066  A
Power Calc.    :  0.06159 W
Power Register :  0.066   W

Voltage (VIN+) :  9.317   V
Voltage (VIN-) :  9.316   V
Shunt Voltage  :  0.00065 V
Shunt Current  :  0.0065  A
```

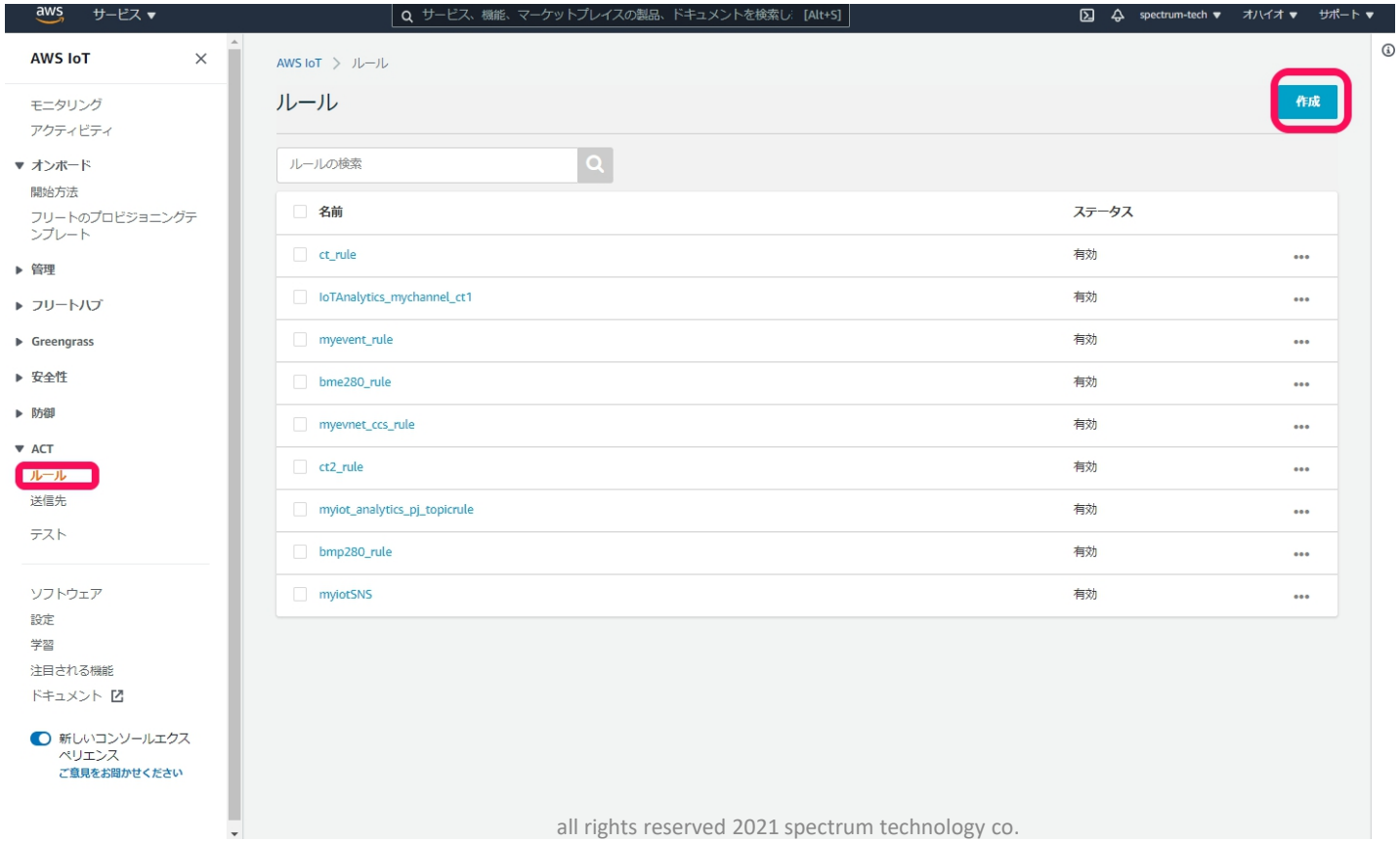
all rights reserved 2021 spectrum technology co.

AWS　IoT開発
④　UPS電源

AWS IoT Core

AWS IoTのルール設定

- AWS　IoTのコンソールからACT＞ルール
- ルール作成を押します。



all rights reserved 2021 spectrum technology co.

AWS IoT開発
④ UPS電源

AWS IoT Core

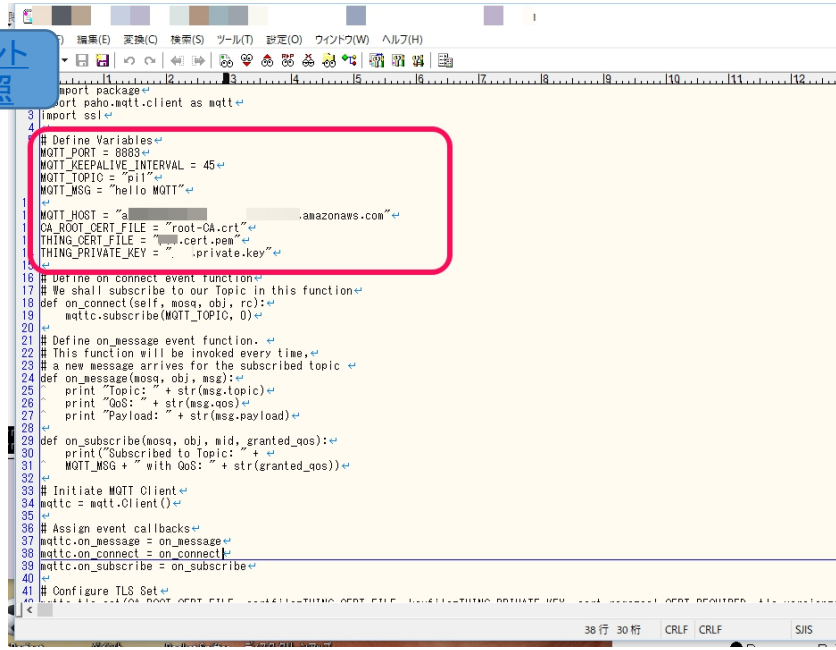
mqtt_subscribe_json.pyのプログラム設定

- Windows PCにさくらエディタなどをインストール
- Windowsネットワークからrasberry piをクリックしてpi>documents>solar
  - mqtt_subscribe_json.pyをwindowsのローカルにコピペします。
  - エディタでローカルのmqtt_subscribe_json.pyを開きます。
  - 電子証明書などの情報を設定します。5-14行

MQTT_TOPIC = “pi3” aws iotルールのトピック

MQTT_HOST = “xxx.amazonaws.com"エンドポイントの値を入れます
CA_ROOT_CERT_FILE = "root-CA.crt"
THING_CERT_FILE = “xxx-certificate.pem.crt"電子証明書番号
THING_PRIVATE_KEY = “xxx-private.pem.key”電子証明書番号

エンドポイントページ参照



- 証明書をフォルダにアップロード
  - 最初にダウンロードした証明書のうち
  - root-CA.crt, xxx-certification.pem.crt,
  - Xxx-private.pem.keyの３個を
  - pi>documents>solarの
  フォルダにアップロード
- 書き換えたmqtt_subscribe_json.pyをアップロード

秘密鍵のペアである公開鍵は使用しません

all rights reserved 2021 spectrum technology co.

Spectrum Technology

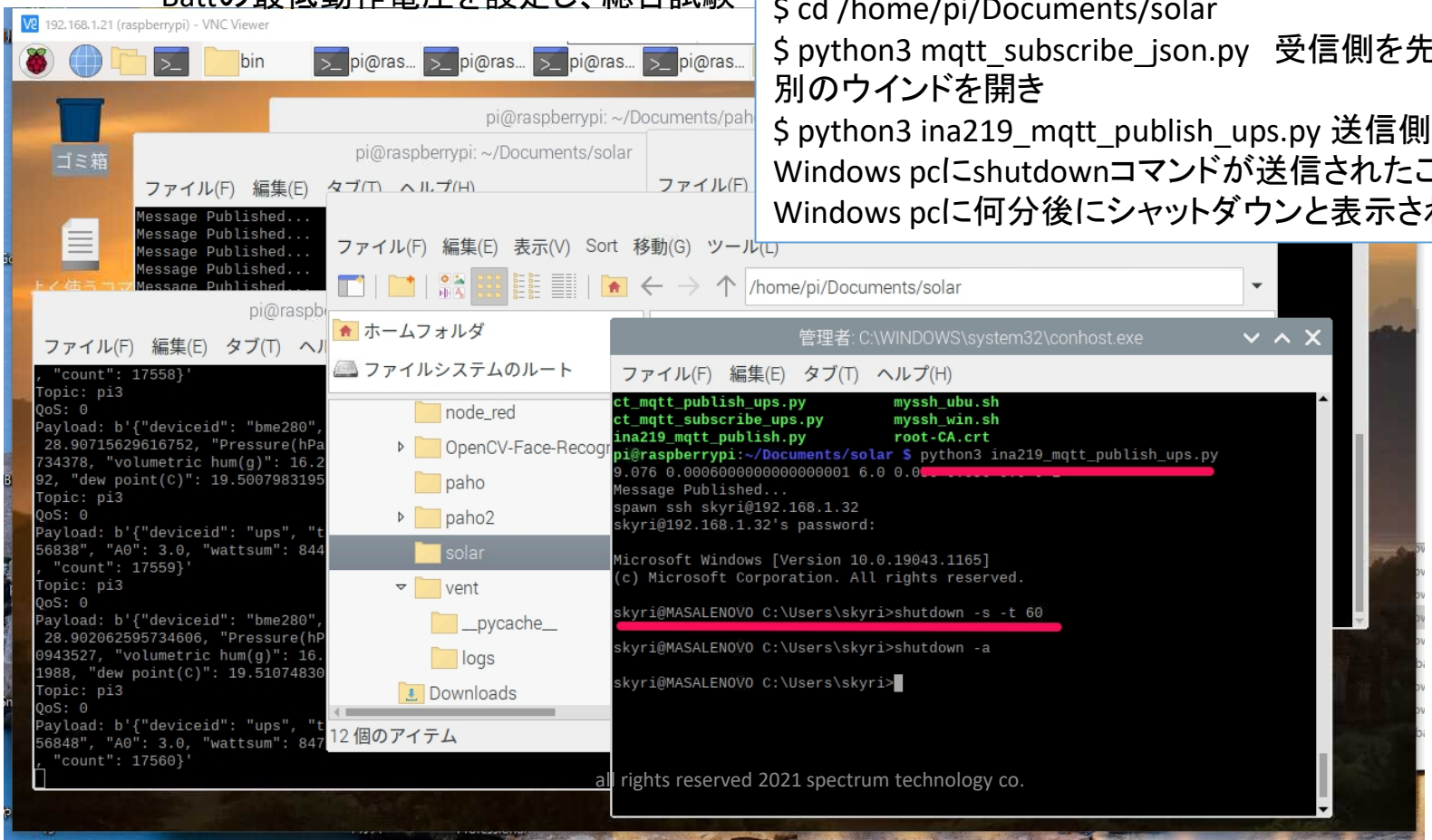
AWS IoT開発
④ UPS電源

AWS IoT Core

AWS IoT総合試験

- Piのコマンド画面からsubscriber, publisherのやりとりを実施
- subscribe側に電圧、電流のデータが受信できていればOk
- Battの最低動作電圧を設定し、総合試験

$ cd /home/pi/Documents/solar
$ python3 mqtt_subscribe_json.py　受信側を先に立ち上げ
別のウインドを開き
$ python3 ina219_mqtt_publish_ups.py 送信側立ち上げ
Windows pcにshutdownコマンドが送信されたことを確認。
Windows pcに何分後にシャットダウンと表示されます



all rights reserved 2021 spectrum technology co.

Spectrum Technology

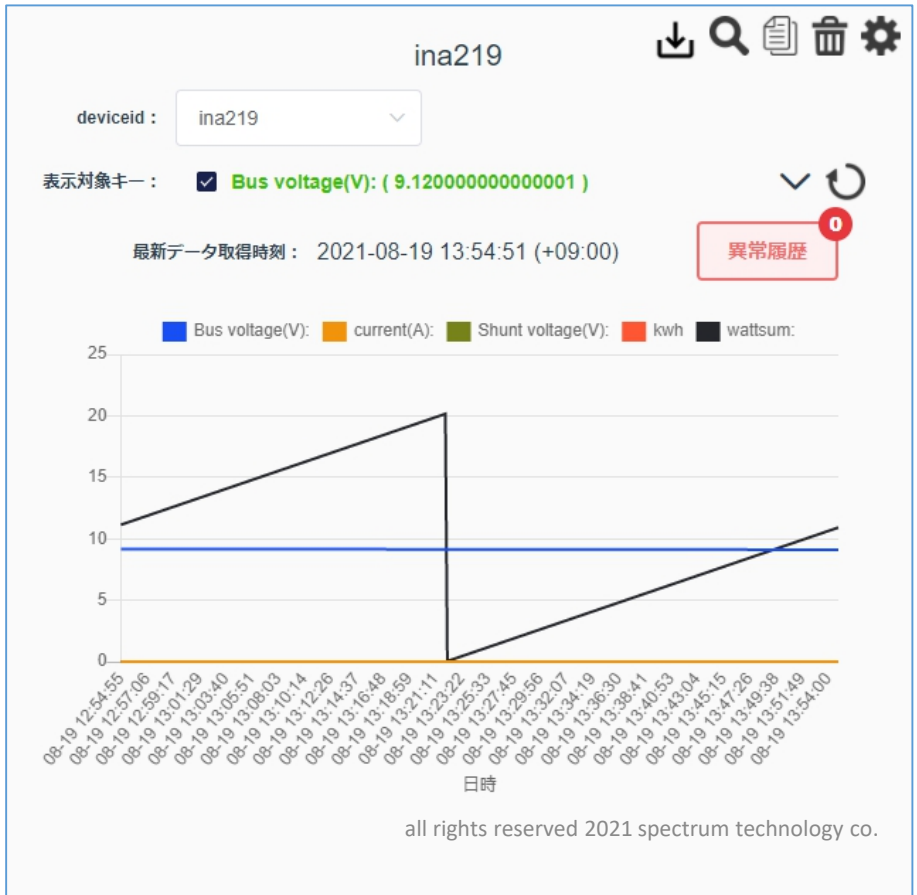
AWS　IoT開発

④　UPS電源

AWS IoT Core

リアルタイム表示

- IoT．Kyoto様の利用＞グラフ
- Deviceid：ina219（python3で書き込み済）
- 表示対象をチェックすると自動で表示されます。



all rights reserved 2021 spectrum technology co.