

ECHONET Lite & ラズパイ/3GPI

おうちハック体験 + もくもく会

スケジュール

- 11:00 ~ 11:30 挨拶 + 下準備
- 11:30 ~ 12:30 第一部 npm echonet-lite から家電をいじって遊ぶ
- 12:30 ~ 13:00 休憩
- 13:00 ~ 14:00 第二部 ラズパイ上の LED を ECHONET Lite 照明にしてみる
- 14:00 ~ 15:00 第三部 ラズパイ上の LED を 3G/MQTT 経由で遠隔操作してみる
- 15:00 ~ 15:15 Knowledge Connector 説明
- 15:15 ~ 16:30 第四部 もくもく会
- 16:30 ~ 17:00 発表会・解散

下準備

apt や npm のインストール・アップデート（実行に時間がかかるため、数日前に実施済です）

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install npm
```

自分に割り当てられたラズパイの IP アドレスを確認し、ssh でログインします。

コマンドラインは通常 `ssh pi@[IP Address]` で、パスワードは `raspberrypi` です。

ssh がない場合ご自分に合った ssh クライアントを選んでください（例えば Windows なら TTSSH など：<http://www.sakurachan.org/soft/teraterm-j/ttssh/>）

コマンドラインで使えるテキストエディタとしては、初めての方には nano がおすすめです。

第一部 npm echonet-lite から家電をいじって遊ぶ

- ・まず、実験室内の ECHONET Lite 機器を操作してみましよう。それには、Kadecot とい

う Android アプリをういます。Kadecot はネットワーク内に一つだけ立ち上がってればいいので、その IP アドレスを当日お教えします。このアドレスを使って、以下のアドレスにアクセスしてください：

[http://app.kadecot.net/Apps/APITool/?kip=\[Kadecot IP\]](http://app.kadecot.net/Apps/APITool/?kip=[Kadecot IP])

すると、数秒ののちに次のようなページが開きます。



下のパスワードのフィールドに「a」と入力し、送信ボタンを押してください。

すると、上の画面が出る前のページに遷移し、**Recognized Devices** のところにいくつか機器が現れると思います。ここから機器の機能を開き、希望のプロパティの **J-SET** ボタンを押し、出てくるダイアログで任意のプロパティを設定すると、機器を動かすことができます。**J-GET** を用いれば、機器の状態を取得することもできます。

次に、以下の URL をブラウザで開いてください。

[http://\[Kadecot IP\]:31413/jsonp/v1/devices](http://[Kadecot IP]:31413/jsonp/v1/devices)

すると、表示される JSON オブジェクトを見ることで IP アドレスとペアで機器の種類がわかると思います。このページを開いたままにしておいてください。あとで自分で ECHONET Lite 機器を作るので、自分が作った機器が現れてくるかどうか、ご確認いただきます。

・次は、ラズパイ内でプログラムを格納するフォルダを作ってそこに移動します。(二人以上でひとつのラズパイをシェアしている場合は、`projects0` と `projects1` という二つのフォルダを作ってそれぞれお使いください)

```
mkdir projects
```

```
cd projects
```

・サンプルプログラムをコピーしてきます。(ファイルは3つあります)

```
cp ../samples/* .
```

・`node.js` のライブラリ `echonet-lite` をインストールします。

```
npm install echonet-lite
```

・`controller.js` を実行します。(複数人が同時に実行することはできません)

```
node controller.js
```

・`controller.js` の中身解説

・次に `airconditioner.js` を実行します。(複数人が同時に実行することはできません)

```
node airconditioner.js
```

ここで、先ほどの Kadecot の JSONP をリロードしてみましょう。自分が操作しているラズパイの IP アドレス上に、エアコンオブジェクトが出現しましたか？

第二部 ラズパイ上の LED を ECHONET Lite 照明にしてみる

・まず、ラズパイ上の LED を手作業で ON/OFF してみる (複数人が同時に実行アクセスした場合、誰が動かしたのかわからなくなることがありますが、実行自体は可能です)

```
led_trigger none #通常はSDカードの読み書きにより点滅するので、それをdisableする
led_brightness 1 #LEDが点灯する
led_brightness 0 #LEDが消灯する
led_trigger mmc0 #SDカードの読み書きの表示に戻す
```

・node.js内からの呼び出しからは次の通り：(複数人が同時に実行アクセスした場合、誰が動かしたのかわからなくなることがありますが、実行自体は可能です)

```
var exec = require('child_process').exec;
exec('led_brightness 0'); // ← コマンドラインをそのまま文字列で与える
```

・airconditioner.jsをコピーしてlight.jsをつくる。

```
cp airconditioner.js light.js
```

・light.jsを改造してECHONET Lite照明オブジェクトを作ってみる(複数人が同時に実行することはできません)

・ECHONET Liteから電源のON/OFFコマンドを受け取った時に、LEDの点灯・消灯を行う(複数人が同時に実行することはできません)

ここも先ほどのKadecot API Toolを使用し、照明オブジェクトが出現したかどうか、そして、その照明のON/OFFができるかどうかを試す。直接API URLで操作してもよい。

```
http://[KadecotIP]:31413/jsonp/v1/devices/[DeviceID]?procedure=set&params={"propertyName":"OperationStatus","propertyValue":[48]}
```

48を49にすると電源OFFコマンドを送ることになる。

DeviceIDは、

```
http://[KadecotIP]:31413/jsonp/v1/devices/
```

にアクセスすることにより、IPアドレスとの対応を取ることができる。

第三部 LED を 3G/MQTT 経由で遠隔操作してみる

- ・まず、MQTT の説明

<https://sango.shiguredo.jp/mqtt>

- ・次に、MQTT のテスト。まず、MQTT ライブラリのインストール

```
npm install mqtt
```

- ・サンプルプログラム `mq.js` を開き、TOPIC を自分固有の文字列に変更する。

```
nano mq.js
```

- ・ `mq.js` 実行

```
node mq.js
```

コンソールに

```
>Connected to broker
```

```
>Hello MQTT!
```

と表示されれば成功

- ・ブラウザから MQTT ブローカーにつなぎ、メッセージを送ってみる(ポートは 9001 番で)

- ・課題 : MQTT から受け取ったメッセージに基づいて LED を ON/OFF するように、`mq.js` を変更。

```
client.on('message',function(topic,message){... のあとに、message の内容によって  
LED を明滅させるコードを足せばよい。
```

- ・ 3G シールドをつなぎ、リブート

```
sudo reboot
```

- ・ 3G ネットワークを優先にする (`netstat -r` でデフォルトのネットワークがわかる)

```
sudo route del default dev eth0
```

```
sudo route add default dev ppp0
```

- mq.js をデーモンとして走らせる

```
forever start mq.js
```

- Ethernet を抜く。それでもブラウザからの ON/OFF で LED の ON/OFF ができるはず。

第四部 Knowledge Connector 説明

残った時間があれば、黙々と好きなことを実装するもくもく会とさせていただきますが、その前に、本日の成果物をまとめて頂きたいサイトがあるので、その説明をします。

Knowledge Connector : <http://idea.linkdata.org/>

APPENDIX

- MQTT 設定資料

<http://tomowatanabe.hatenablog.com/entry/2016/01/21/095007>

主催

おうちハック同好会 <https://www.facebook.com/groups/ouch.hack/>

協力

神奈川工科大学 <http://www.kait.jp/index2.php>

LOD チャレンジ実行委員会 <http://lodc.jp/>

メカトラックス株式会社 <http://www.mechatrax.com/>

株式会社ソラコム <https://soracom.jp/>