

ECHONET Lite

プロトタイピング・ツールの 紹介

2016年11月17日



ホーム・ネットワークWG 主査
長島 宏明 / コアーズ株式会社

Cores

ホーム・ネットワークWGとは

- 2013年度ソフトウェア開発委託事業
 - 「ECHONET Lite通信ミドルウェアの開発」で開発したTOPPERS/ECNLの一般公開を機に
 - TOPPERS/ECNLの利用促進などの活動のため2015年にワーキング・グループ化し活動中
- ワーキング・グループでは
 - ECHONET Liteを含むホーム・ネットワークを対象としたミドルウェアやサンプルアプリを開発

昨年からの活動概要

- 2016年1月 ENEX2016の展示物にμKadecotが採用
- 2016年2月 ECHONET第4回フォーラムへのμKadecot動態デモ展示
- 2016年2月 OSC2016 Tokyo/Spring動態デモ展示
- 2016年3月 USB充電器電流計サンプルアプリの公開
- 2016年5月 ESEC2016動態デモ展示
- 2016年10月 OSC2016 Tokyo/Fall動態デモ展示
- 2016年11月 TOPPERS/ECNL Ver.1.03

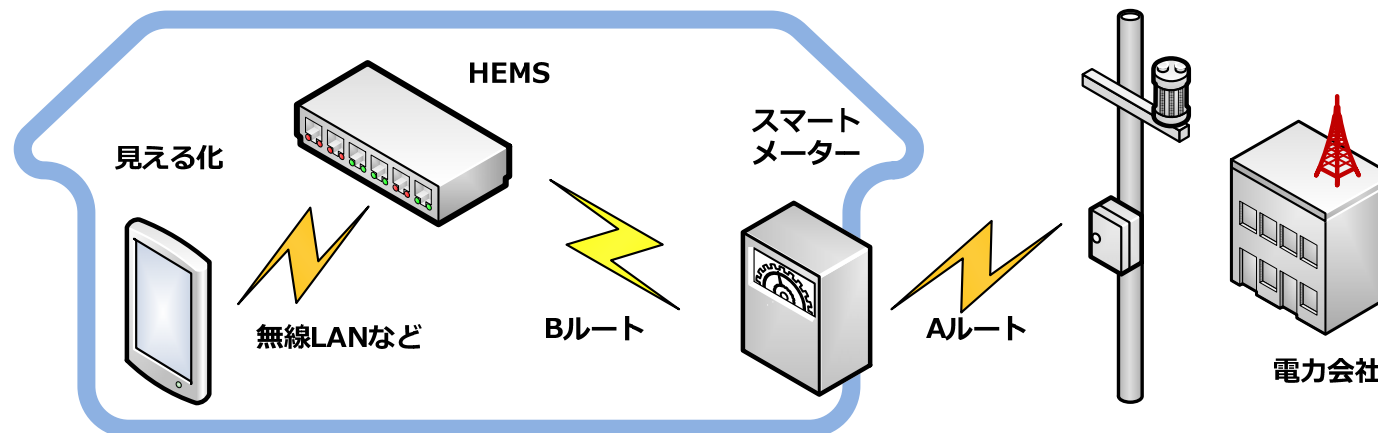
身近なECHONET Lite機器

- スマートメーターは、国の政策としてすべての家庭に導入されつつある
- 2016年4月に開始した電力自由化でも必要な設備
- スマートメーターは、全電力会社が同じ規格で通信する
 - 無線通信の規格「Wi-SUN」
 - 上位レイヤーの通信規格「ECHONET Lite」
- つまり、「ECHONET Lite」は日本の家庭に広まりつつある

電力会社だけが対象？

- スマートメーターは、電力会社だけが利用するものではなく、個人でも30分ごとの電力使用量を見ることが出来る
- これは、「Bルート」と呼ばれる通信経路で、HEMSで利用することを想定した仕様

<http://www.meti.go.jp/press/2013/05/20130515004/20130515004-5.pdf>



電力使用量の活用

- Bルートから、電力使用量を見ることのできる装置は既に販売されている
 - 「スマートメーターBルートブログ」
<http://route-b.iij.ad.jp/archives/30>
- 個人が作ったプログラムでもデータを見ることができる
 - 「法人無料！スマートメーターとIIJのHEMSコントローラを導入して家の電力ログにアクセス」
<http://qiita.com/sgrowd/items/d4aa644121b08ef11256>
 - 「スマートメーターの情報を最安ハードウェアで引っこ抜く」
<http://qiita.com/rukihena/items/82266ed3a43e4b652adb>
 - 「mrubyでスマートメーター」
<http://qiita.com/yamori813/items/f0dfedf31b2b51e658f9>

自前でIoTを活用

- スマートメーターのような、実用性の高いIoT機器が、普及しつつある
 - ECHONET Lite搭載エアコンも増えている
 - IoT活用のプロトタイプなら自前で作ることができる
- たとえばRaspberry Pi
 - 入手しやすく、Linuxの知識があれば開発が容易
 - Webに記事が豊富にある
 - Node.jsやPythonなどプログラムしやすい言語を使用できる
 - 実用的なプロトタイプを作成することができる

TOPPERSでプロトタイプ

- Linuxはソフトウェアも豊富だが、その分複雑もう少しシンプルな環境はないか？



- TOPPERSでは「LEGO MINDSTORMS EV3」向けのプラットフォーム環境を提供し、Ruby言語も使用できる環境も提供している



- RubyでECHONET Liteを扱えるようにしたい

組み込み向けRuby実行環境

- mrubyは組み込み向けのRuby実行環境



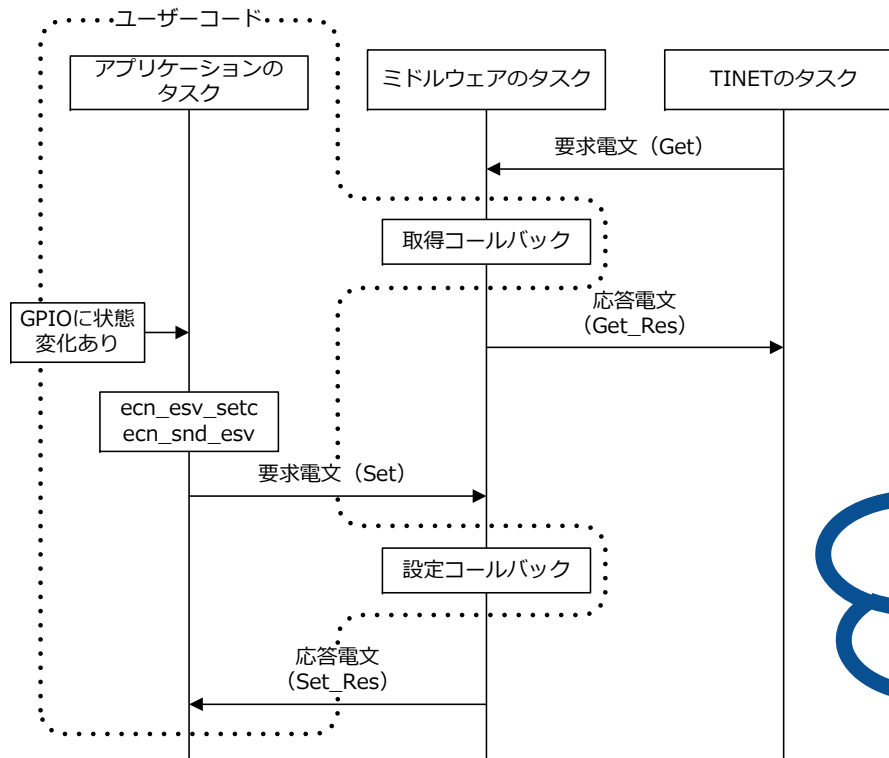
- Rubyが動作する基本機能のみ
- mrbgemと呼ぶ拡張機能で、通信機能や文字列操作などの機能を追加することができる



- ECHONET Lite通信ミドルウェアECNLのmrbgemを作って、Rubyでソフトウェアを書けるようにすれば、プロトタイプも作りやすくなるのでは？

GR-SAKURAでmruby

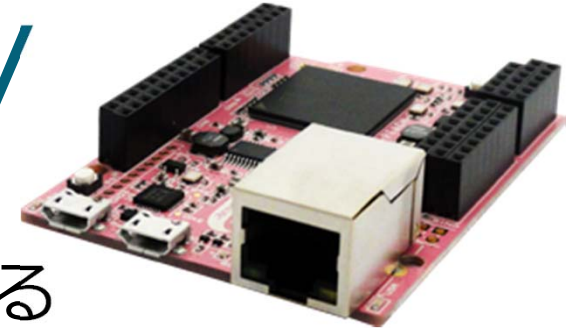
- 使用実績のあるGR-SAKURAで実行環境を作成
- mrbgem版ECNLを作成



ECNLのアプリケーションは3つのタスクで動作していたが、
mrubyで実行するに当たり、
シングルタスク化

メモリ不足でmrubyに
対策が必要
動作はしたものの
余裕がない

GR-PEACHでmruby



- 最新のTOPPERS/ASP3では、GR-PEACHのターゲット依存部がある
- RAMが10MByteありメモリに余裕がある
- mrubyを実行する処理能力がある
- ECHONET機器のプロトタイプを作成に当たって、GPIOなどの操作がしたい
 - GR-PEACHには、mbedやArduinoライブラリもある
 - mrubyのArduino APIの実装がGithubに公開されているので、それを使う

<https://github.com/wakayamarb/wrbb-v2lib-firm>

mrbgem版ECNLの課題

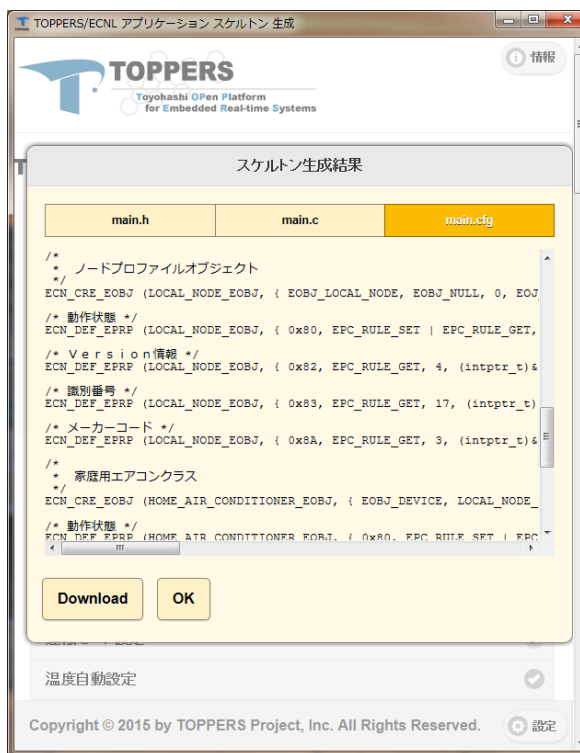
- ASP版ECNLではCFGにより生成しているコードを、ユーザーがRubyで実装しなければならない
- Rubyで記述できる範囲を広くしたため、ユーザーが見るコードが多い
- ユーザーが作成したRubyのプログラムは、コンパイルして書き込む必要がある



- スケルトン生成ツールをRuby用に更新し、開発環境を作成、コンパイルと書き込みを簡単にしたい

スケルトン生成ツールとは

- ASP版ECNLのアプリケーション作成の手間を減らすために作ったWebアプリベースのツールで公開中

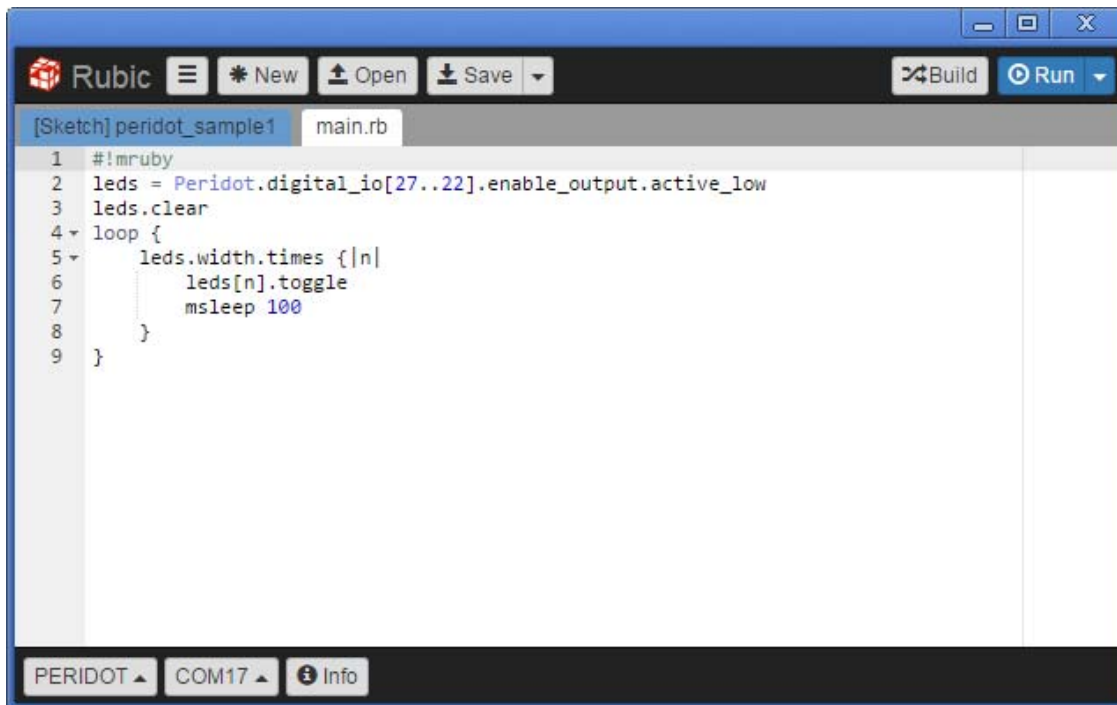


機器オブジェクト
やプロパティを選
択して、H/C/CFG
ファイルのスケル
トンコードを生成

Rubyコードが出力
できるよう変更

mruby開発環境のRubic

- WebブラウザのGoogle Chromeのアプリ



```
1 #!mruby
2 leds = Peridot.digital_io[27..22].enable_output.active_low
3 leds.clear
4 loop {
5   leds.width.times {|n|
6     leds[n].toggle
7     msleep 100
8   }
9 }
```

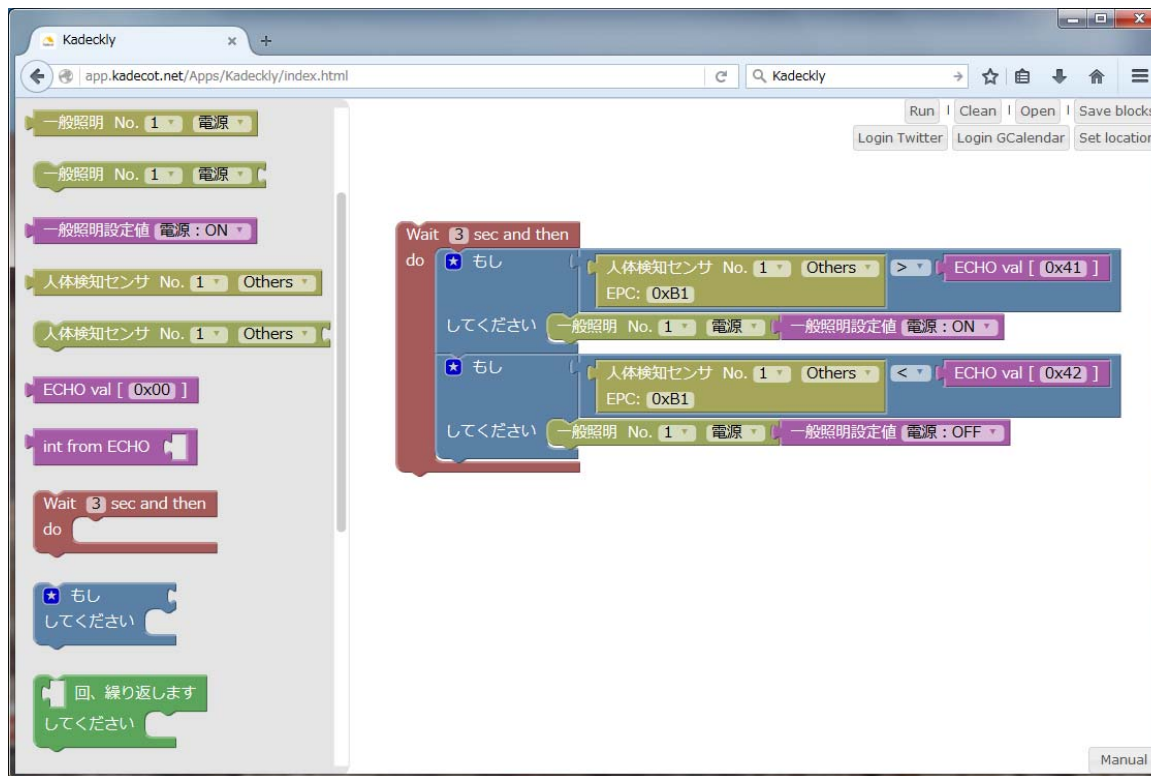
Rubyのコードエディタを持ち、実行ファイルへのコンパイルし、機器へシリアルでの書き込みが出来る開発環境が既にありオープンソースで提供されている

http://gadge.renesas.com/ja/product/citrus_sp1.html

<https://chrome.google.com/webstore/detail/rubic/mqbcgagfggopcpbbfgididdbnhnhjp>

KAIT-4S~Canvas~/Kadeckly

- ビジュアル言語を使用すれば、より簡単に



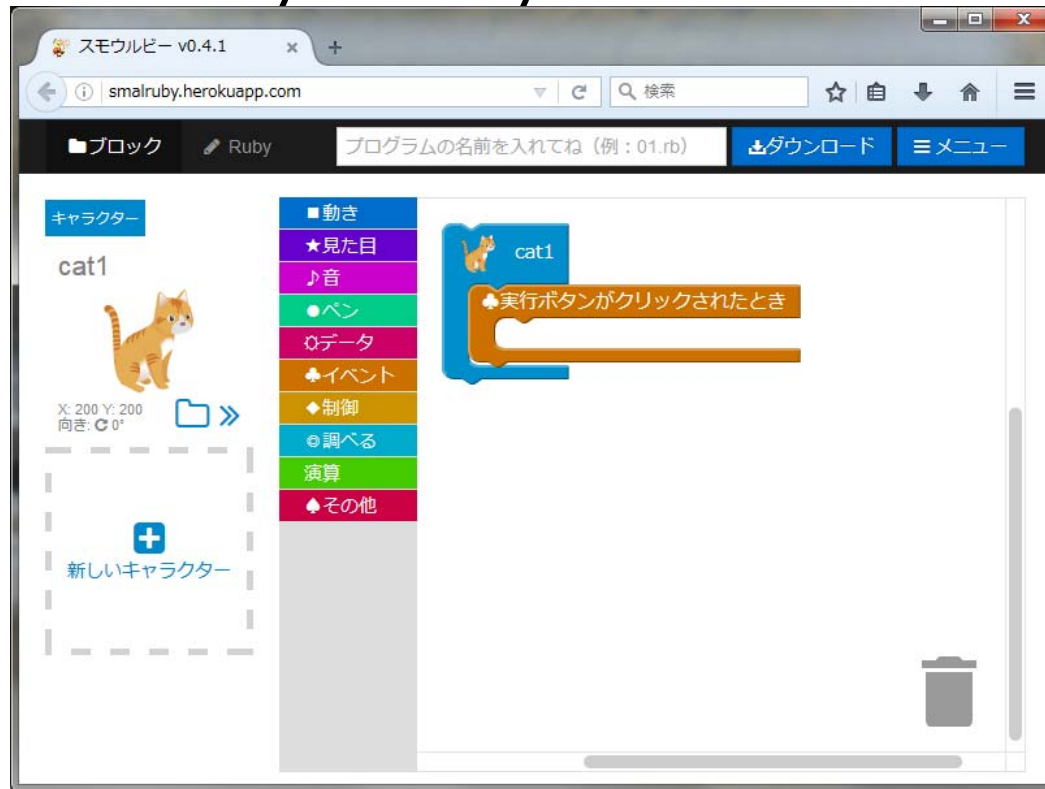
Google Blocklyを使ったECHONET Liteの実行環境も既にある

ただし、Rubyコードを出力することは出来ない

<https://smarthouse-center.org/sdk/download/form/17>

Ruby版Scratch Smalruby

- BlocklyでRubyを出力



ビジュアル言語で記述した内容を、Rubyで出力するScratch風のプログラム環境が既にあり、オープンソースで公開されている

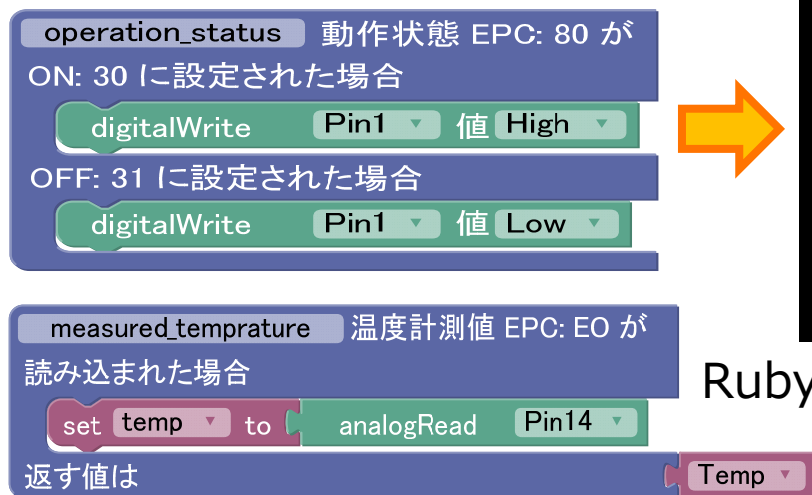
<http://smalruby.jp/>

<https://github.com/smalruby/smalruby-editor>

これらを組み合わせる

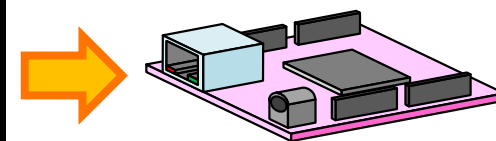
- プロトタイピング・ツールを開発
 - Webアプリによるビジュアル言語の設計
 - mruby実行環境でECHONET機器を実装

Blocklyでプログラム



```
# 動作状態設定関数
def operation_set(prop, src)
  case src.getbyte(0)
  when 48
    digitalWrite(0,1)
  when 49
    digitalWrite(0,0)
  end
  return prop.sz
end
```

Rubyで出力



mrubyで実行

コンパイルと
書き込み

プロトタイピング・ツール

- mrubyの実行環境をもつ機器側
 - GR-PEACH + TOPPERS/ASP3 + TINETで実行
 - ECHONET Lite通信処理をRubyで書ける
 - Ruby版Arduino APIでECHONET機器の実装ができる
- 内蔵Webアプリケーション
 - GR-PEACH上のWebサーバーで提供
 - ビジュアル言語でECHONET機器を実装
 - Ruby言語で出力し、Webアプリでコンパイル
 - 実行ファイルをGR-PEACHのWebサーバー経由で書き込み

完成に向けて作業中

- mrbgem版ECNL
 - mrubyのビルド時に使用できるようにGithubで公開
<https://github.com/h7ga40/mruby-ecnl>
 - Webアプリに合わせて変更予定
- Webアプリ
 - Githubに動作デモあり
<https://h7ga40.github.io/mruby-ecnl-prototyping-tool/>
 - ただし、今後大きく変更する予定
- ソースコードはTOPPERSコントリビューテッドソフトウェアから公開予定



TOPPERS

ありがとうございました