

みんなのラズパイコンテスト 2018 応募用紙

■応募に関するお願い

「みんなのラズパイコンテスト」(主催:ラズパイマガジン、日経 Linux、日経ソフトウェア)へのご参加、ありがとうございます。特設サイト(<http://nkbp.jp/rpic>)内にある応募フォームより本ファイルを送付ください。本応募用紙は返却しませんのでご了承ください。

特設サイト内で公開している「応募の手引き」をご覧ください。本ファイル内に審査に必要な情報(説明文や写真、イラストなど)をすべて盛り込んでください。提出いただくファイル形式は、Wordまたは LibreOffice Writer、PDF に限ります。ファイルサイズは 10M バイト以下にしてください。

■作品名 ※必須記入 (Web サイトの応募フォームでも本作品名を記載していただきます)

なんちゃって主夫のありあわせ食材でチャチャッとスマートホーム

■使用した無償提供製品 (使用していない場合は空欄で構いません) [詳細は、特設サイト (<http://nkbp.jp/rpic>) の無償提供のお知らせを参照してください]

オムロン環境センサ 2JCIE-BL01

■応募種別 (どちら一方を選んで、当てはまる 印を 印に変更してください)

個人で応募 (個人名で表彰します)

チームで応募 (チーム名、または企業名・部署名で表彰します)

※チーム応募は複数のメンバーが協力して企画・作成していることが条件です

■応募者 (チームで応募する場合は代表者) の属性 ※必須記入

個人情報には本用紙には記入せず、特設サイトの専用応募フォームに記入してください。

※応募者が 18 歳未満の場合は、保護者同意の上ご応募ください

性別	男
年齢	48 才 (2018 年 10 月 1 日時点)
所属 (当てはまる選択肢だけを残し、あとは消去してください)	その他
チーム名または企業名 (所属)	

※個人情報について

日経 BP 社の個人情報保護方針 (<http://corporate.nikkeibp.co.jp/information/privacy/>) と、個人情報取得に関するご説明 (<http://corporate.nikkeibp.co.jp/information/privacy/p8.shtml>) をご覧いただき、同意の上ご応募ください。Web サイトの応募フォームでご記入いただいた個人情報は、応募に関する確認や受賞に関するご連絡、コンテストに関連する情報のご案内など、本コンテストの審査・運営のために利用します。受賞者のお名前はラズパイマガジンや日経 Linux、日経ソフトウェアの誌面等で公表いたします。

■応募内容の概要

ご応募いただいたものは何でしょうか。用途や特徴について教えてください。詳細については後述いただくので、ここでの記入は簡潔に一言二言で構いません。

オムロン環境センサと家に既にあった機器を使って、お金と手間をかけずに、なんちゃってスマートホームを作る。

■応募内容の属性 (どちら一方を選んで、当てはまる□印を■印に変更してください)

・ 応募内容は作品 (実物がある) ですか、それともアイデア (実物は存在しない) ですか？

■作品 □アイデア

・ 応募内容の種類は何ですか？

□電子工作 ■アプリケーション □その他

■応募内容の詳細

ご応募いただいたものについて、詳しくご紹介ください。応募したものが持つ機能・使い方、特徴的な仕組み・技術などを説明してください。

審査員が応募内容を理解しやすいよう、写真やイラスト、画面ショットなどを貼り付けることをお勧めします。作品/アイデアを紹介しているブログや動画サイトなどがあれば、URLも併せて記載してください。

ファイルサイズが全体で10Mバイト以下に収まるようご注意ください。応募内容の書き方は自由です。

前菜(はじめに)

私は、なんちゃって主夫です。家の中で家事などの作業をやり、空き時間には趣味のなんちゃってプログラマなどをしています。家中心の生活なので、どうしても運動不足気味になってしまうので、いつもなんとかしたいと考えています。

今回、みんなのラズパイコンテスト 2018 でオムロン環境センサ 2JCIE-BL01(以下、環境センサ)を無償提供していただいたので、ラズパイと環境センサなどを使って、主夫生活が楽しくなるように、ありあわせ素材でチャチャッと簡単ななんちゃってスマートホームシステムを作ってみました。

ただ、主夫なのであまり自由になるお金がないことと、家事以外の空き時間もそれ程多くないため、以下を目標にコース(システム)を作成しました。

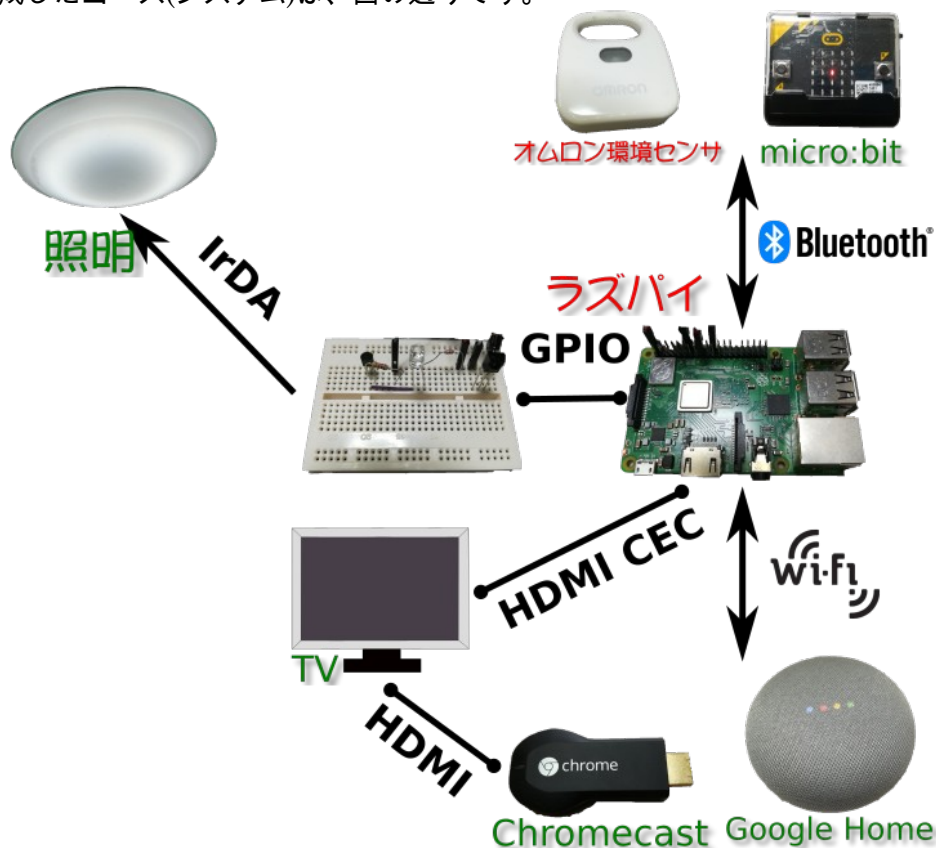
- ・ ありあわせ(節約): できるだけ手元にある食材(ハードウエア)だけを使って出費を抑えます。
- ・ チャチャッと(時短): 色々なところで公開されている、先人のレシピをできる限り使わせてもらい、手早くコース(システム)を作ります。

既にあるレシピ(ソフトウエア)を使うため開発言語などがまちまちになりますが、言語と言語の間は UNIX の shell や cron などをつなぎにして使います。

では、私のコース(システム)をお楽しみください。

コース(システム)

今回作成したコース(システム)は、図の通りです。



以下、はじめに作ったメニュー(機能)について、次に食材(ハードウェア)、レシピ(ソフトウェア)と順番に説明していきます。

作ったメニュー(機能)

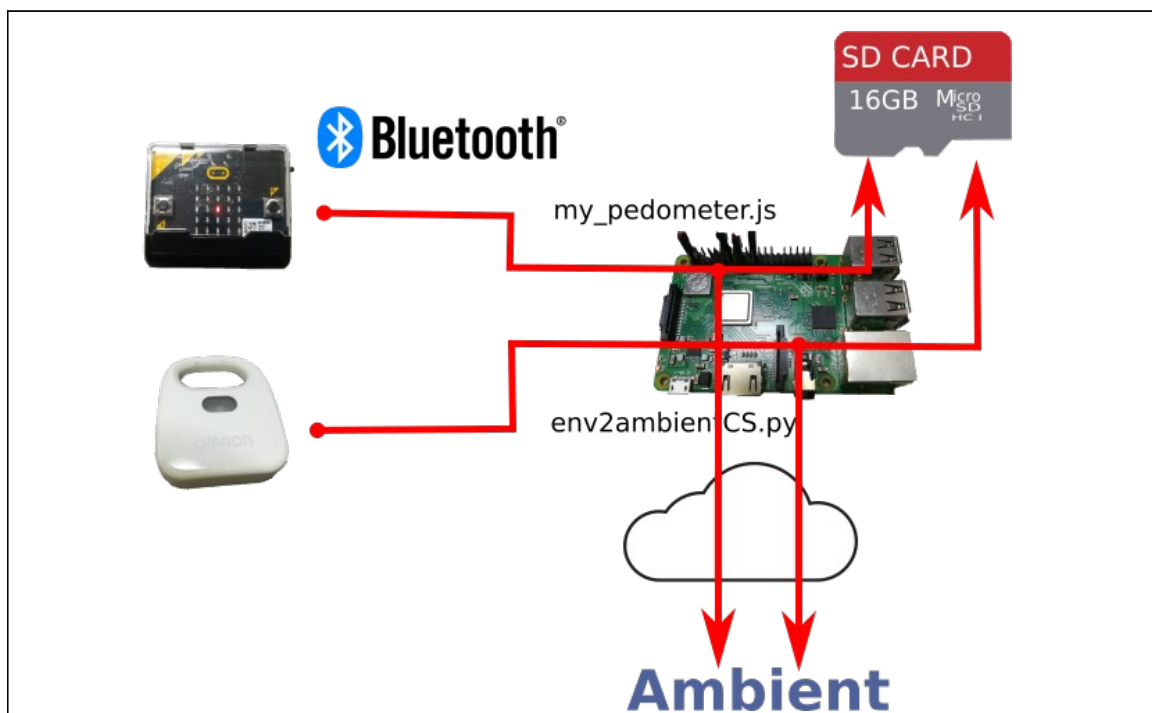
今回作ったメニュー(機能)は、以下のとおりです。

- 環境センサや micro:bit から、センサデータの収集をします(Bluetooth の近海データ集め:データ収集部)。
- 熱中症度が高い時に、音声で警告をします(あんじょうしいや:警告部)。
- 歩数が少ない時に、音声で警告し、エクササイズ動画を流します(あんじょうしいや:警告部)。
- 音声で照明を制御します(もっと光を:照明部)。
- 自動で照明を制御します(もっと光を:照明部)。

Bluetooth の近海データ集め:データ収集部

データ収集部は、環境センサと micro:bit を Bluetooth 経由でラズパイにつなぎ、ローカルの CSV と、クラウドの Ambient に記録するようになっています。

- 環境センサでは、取得可能な全てのデータを集め、記録します。
- micro:bit では、加速度計データを取得し、歩数に変換したのち記録します。

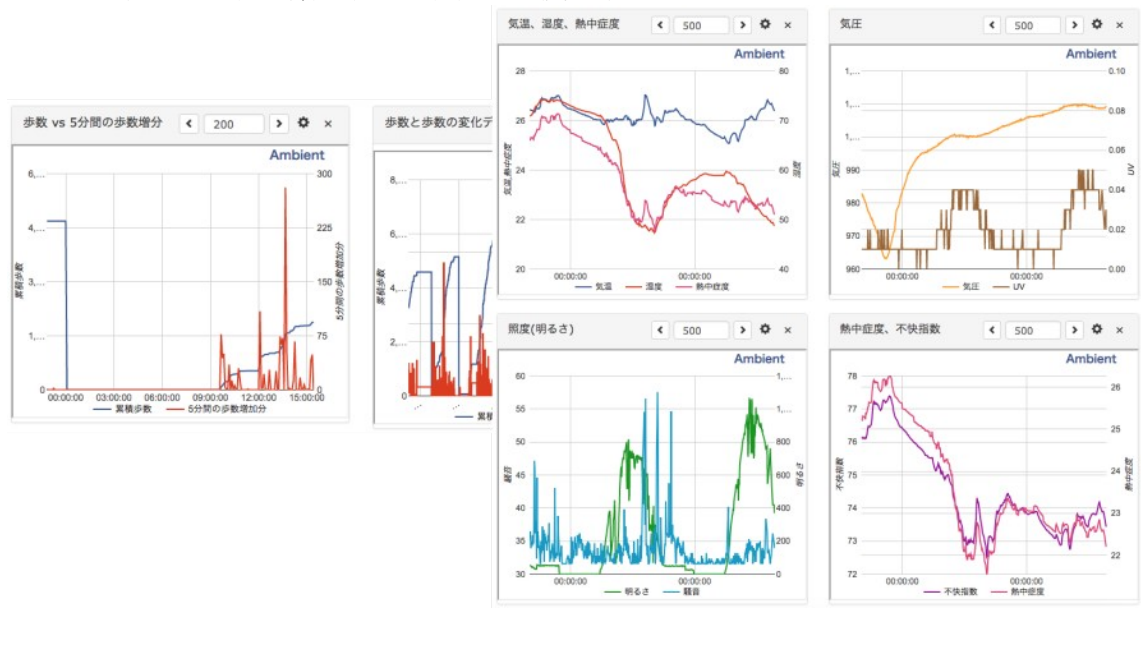


センサからのデータ取得には、Bluetoothでの接続性が必要なため、センサはラズパイから近い場所、つまり、家の中で利用することになります。

以下に、Ambientに記録したデータをグラフ化したものを示します。左が micro:bit 歩数計による歩数データで、右が環境センサで測定したデータです。

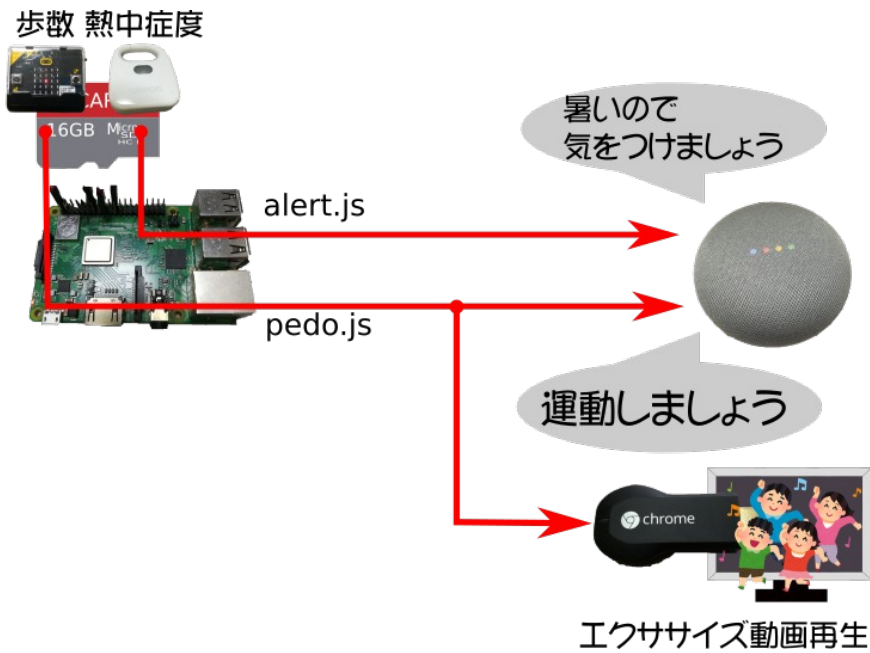
- 歩数データ <https://ambidata.io/ch/channel.html?id=6173>
- 環境センサ <https://ambidata.io/ch/channel.html?id=5624>

データの収集のみの目的以外で、スマートホームのメニュー(機能)を実現するために利用しているデータは、歩数と明るさ、熱中症度です。



あんじょうしいや:警告部

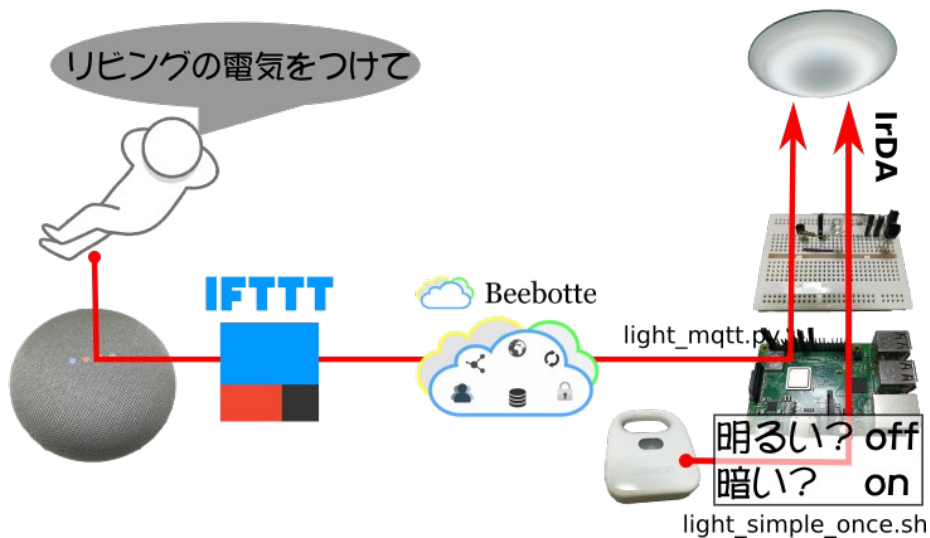
警告部では、歩数と熱中症度によって、Google Homeを使った音声警告を定期的に行います。歩数が目標より少ない時には、ランダムなエクササイズ映像が、Chromecast経由でTVに自動再生されます。



電気の勝手に声かけ風味:照明部

照明部は、二つのメニュー(機能)を組み合わせたものになっています。

- Google Homeの音声コマンドで、ラズパイに接続された赤外線リモコン経由で、照明の点灯・消灯の操作ができます。
- ラズパイが環境センサの明るさを使って、自動で照明の点灯・消灯の操作を行います。



食材(ハードウェア)

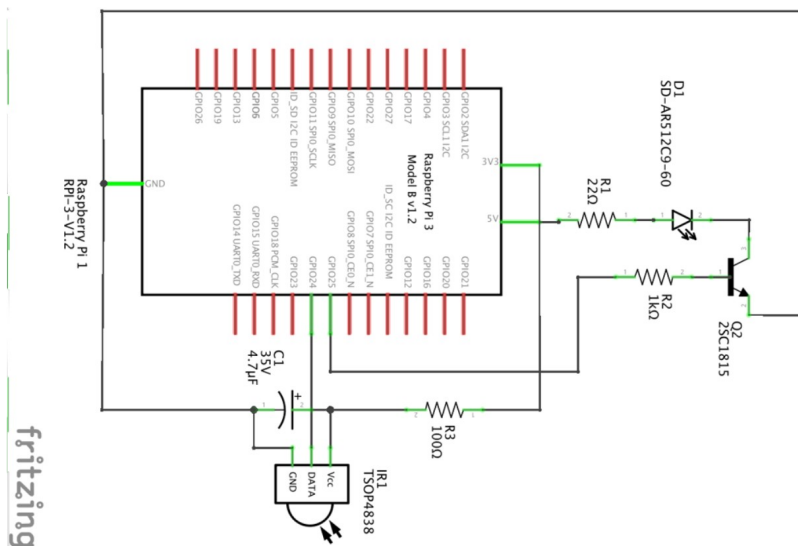
今回、コース(システム)を作るために利用した食材(ハードウェア)は、以下の通りです。いずれの食材も、環境センサの無償提供が決まった時点で既に持っていたありあわせのもので、追加の出費はありませんでした。

食材(ハードウェア)	追加費用	費用(参考)	利用目的	備考
Raspberry Pi 3 B+	0	5670 (switch science)	データ収集など	*BSDで遊ぶために買っていた
オムロン環境センサ2JCIE-BL01	0	10800 (オムロン)	環境データの収集	みんなのラズパイコンテストから無償提供
micro:bit	0	2160 (switch science)	歩数データを得るための加速度センサ	小学生の子供と遊ぶために買っていた
Google Home mini	0	3000 (キャンペーン価格)	音声警告と照明操作	半額キャンペーンの時に買っていた
Chromecast	0	4980 (google)	エクササイズ動画再生	既に持っていた
赤外線リモコン関連部品	0	数百円 (aliexpressなど)	赤外線リモコン	以前からスマートリモコン的なものを作ろうと思って買っていた(部品詳細は後述)

赤外線リモコン部

ラズパイに GPIO 接続した赤外線リモコン部の回路図は以下のとおりです。手元にあったありあわせ素材(部品)で作りました。利用した部品一覧は、表の通りです。

ラベル	部品種類	型番など
C1	コンデンサ	4.7 μ F, 35V
D1	Irレシーバ	SD-AR512C9-60
IR1	Ir-LED	TSOP4838
Q2	NPNトランジスタ	2SC1815
R1	22 Ω	
R2	1k Ω	
R3	100 Ω	



盛り合わせ(機器配置)

盛り合わせ(機器配置)は、写真のようにしました。

- ラズパイ: 赤外線リモコンによる照明の操作と、HDMI CEC を使った TV 電源の操作のために、TV 横に設置しました。
- 環境センサ: 照明の明るさをはっきりと取れるように、照明に近い掛け時計の振り子部

分に設置しました。

- micro:bit: 加速度計を使って歩数が取れるように、電池をつけて、ズボンのポケットに入れました。



レシピ(ソフトウェア)

色々な所で公開されているオリジナルレシピ(ソフトウェアライブラリや記事)を利用して、コース(システム)を作りました。「レシピ」で利用できる部分は再利用して、システム構築に必要な部分だけ「もう一味」としてプログラミングを行ないました。

また、うまく組み合わせ切れず問題が生じた場合は、サービスの再起動などの力技で解決したところもあります。

Bluetooth の近海データ集め:データ収集部

環境センサの環境データと、歩数計のための micro:bit からの加速度計のデータは、Bluetooth 経由でラズパイに集められます。データは、IoT データ可視化サービスの Ambient と、ローカルに CSV として保存します。

オムロン環境センサのデータ収集のためには、以下のようなレシピを使いました。

- オリジナルレシピ: オムロン環境センサーから BLE 経由でデータをクラウドに送る <https://ambidata.io/samples/temphumid/ble-gw-omron/>
 - ソース: <https://github.com/AmbientDataInc/EnvSensorBleGw>
 - 環境センサからのデータを 5 分毎に取得し、Ambient に記録します (env2ambientCS.py)。
 - もう一味: データを CSV としてローカルにも記録します。

micro:bit 歩数計データの収集のためには、以下のような二つのレシピを使いました。

- オリジナルレシピ 1(micro:bit 操作系): <https://github.com/sandeepmistry/node-bbc-microbit> /examples/{accelerometer-listener.js,button-listener.js,led-text.js}
 - レシピでは、それぞれ、加速度計のデータを取得し、ボタンが押された時に処理を行い、LED に文字列を表示します。
- オリジナルレシピ 2(加速度から歩数を求める): 加速度センサを使って犬用 Wifi 歩数計を作ってみた <https://qiita.com/shinsuki/items/76e2bf2d8a703e88fb10>
 - 加速度センサの値から、歩数に変換します。
- もう一味
 - A ボタンが押された時に、歩数を micro:bit の LED に表示します。

- micro:bit の加速度データを 80ms 毎に取得し、歩数を計算、出力は 5 分毎に CSV に記録します(my_pedometer.js)。
- 5 分毎に、別シェルスクリプト(pedo_ambient.js)で Ambient に記録します。

二つの Bluetooth データ収集系(env2ambientCS.py, my_pedometer.js)は別スクリプトとして実装されており、まれに停止する時があるため、5 分毎に監視し、10 分以上停止していた場合 Bluetooth 関連を再起動するようにしています(reboot_bluetooth.sh)。

あんじょうしいや:警告部

警告部では、ラズパイから記録されたデータの値にもとづいて、Google Home で音声による警告を行います。

- 環境センサの熱中症度を使って、値が大きい時に Google Home から警告を送ります。
- micro:bit から集めた歩数データが小さい時には、運動をするように警告し、Chromecast を使って自動でエクササイズビデオを TV に流し、ビデオが終わった頃に HDMI CEC を使って TV の電源を切ります。

Google Home から警告音声を出すために、以下のレシピを使いました。

- オリジナルレシピ: Google Home 開発入門 / google-home-notifier 解説 <https://qiita.com/SatoTakumi/items/c9de7ff27e5b70508066>
 - ソース: <https://github.com/noelportugal/google-home-notifier>
 - ラズパイなどから Google Home をしゃべらせませす。
 - もう一味: 熱中症度や歩数に応じて、音声による警告を行います。
 - 音声による熱中症度(環境センサ)警告(alert.js)
 - 音声による運動不足(歩数計)警告(pedo.js)

Chromecast で YouTube の動画を再生するためには、以下の三つのレシピを使いました。

- オリジナルレシピ 1& ソース: https://github.com/balloob/pychromecast/examples/youtube_example.py
 - Chromecast を使って、Youtube などの動画を再生します。
 - もう一味: 歩数に応じて、ランダムに選んだエクササイズ動画を再生します。
- オリジナルレシピ 2: Python と Youtube Data API(v3) で再生数を取得する <https://qiita.com/funsuke/items/160684329ef29f2fbf65>
 - YouTube Data API を使って、ビデオの再生回数を取得します。
 - もう一味: TV の電源を動画再生後に切るために、ビデオの再生時間を取得するようにしました。
- オリジナルレシピ 3: Raspberry Pi とテレビを HDMI 接続してテレビの ON/OFF を確認する <https://qiita.com/mascii/items/132c064c30e321dea911>
 - HDMI CEC(cec-client)を使って TV の電源を操作します。
 - もう一味
 - TV を観ている時(もともと電源が入っている時)には、エクササイズ動画は再生しません。
 - エクササイズ動画が終わったあとに、TV の電源を切ります。

電気の勝手に声かけ風味:照明部

照明の操作は、音声での操作と、環境センサの明るさによる自動操作の、二つの方法で行います。照明は赤外線リモコン付属のものを利用し、ラズパイの GPIO につないだ赤外線 LED から操作します。

Google Home からの音声命令は、IFTTT で受け取り、Web Request で Beebotte を経由して、MQTT でラズパイに送られ、最終的にラズパイで赤外線リモコンを irsend で操作して、照明をオン/オフします。

ラズパイに繋いだ赤外線 LED を使った赤外線リモコン操作のためには、以下のレシピを使いました。

- オリジナルレシピ: Raspbian Stretch で LIRC 機能を使った学習リモコン、赤外線リモコンを動かす方法 http://www.neko.ne.jp/~freewing/raspberry_pi/raspberry_pi_stretch_lirc_ir_remote_control_2017/
 - ソース: <http://www.lirc.org/>
 - もう一味: 環境センサの明るさに応じて自動で照明を操作する (light_simple_once.sh)

Google Home への音声コマンドから、ラズパイで任意の命令を実行するためには、以下のレシピを使いました。

- オリジナルレシピ: Google Assistant と Raspberry Pi で自宅の家電を操作する <https://qiita.com/104ki/items/9dcfe03246099d03d4dd>
 - ソース: IFTTT <https://ifttt.com/> , Beebotte <https://beebotte.com/> , paho-mqtt <https://www.eclipse.org/paho/>
 - 音声による照明の点灯、消灯(light_mqtt.py)を行います。
 - もう一味: 汎用的な赤外線リモコン用ライブラリ lirc を使って赤外線リモコンで操作するようにしました。

定期実行部

cron を利用した定期実行では、以下のようなスクリプトを動作させています。

家族がいる土日や朝夕夜には、自動照明調整や音声警告やエクササイズビデオ上映が行われないように調整しています。

- 自動照明スイッチ: 現在の明るさによって、しきい値判定して照明をオン/オフ
*/5 6-20 * * * pi /home/pi/bin/light_simple_once.sh
- Ambient へのデータ記録: ローカル CSV データを Ambient に記録
*/5 * * * * pi /home/pi/ambient/pedo_ambient.js
*/5 * * * * pi /home/pi/bin/raspi_stat.js
- Google Home への警告メッセージとエクササイズビデオ上映
30 8-15 * * 1-5 pi /home/pi/myhome/alert.js
0 12-15 * * 1-5 pi /home/pi/myhome/pedo.js
- Bluetooth データ収集部の再起動: Bluetooth 部不調の時に再起動
*/5 * * * * root /home/pi/bin/reboot_bluetooth.sh
*/5 * * * * root /home/pi/bin/reboot_pedo.sh

お味はいかが?(まとめ)

今回、オムロン環境センサを無償提供していただいたことで、思いもよらずみんなのラズパイコンテストに応募することになりました。

無償提供が決まってから作ったシステムなので、つぎはぎだらけのものになりましたが、楽しんで作ることができました。

以下、このコース(システム)についてのいくつかのエピソードを書きたいと思います。

手間がかかったところ

- 照明の点灯状態がラズパイ側から取得できないため、自動点灯機能の実装に苦労したが、単純な閾値を使うだけでそれなりのものができました。
- 2つのBluetooth(環境センサ、歩数計)の干渉かなにかで、Bluetooth経由のデータ収集がたまに止まることがあるので、再起動するようにしました。

美味しくないとこ(問題点)

- 環境センサと歩数計は、ラズパイから BLE が繋がる範囲でしか利用できないので、現実的には家の中でしか使えません。
- 赤外線リモコンが届くところで、TVにHDMI接続できる必要があるので、ラズパイの設置場所が限られます。

新たなレシピに向けて(今後の作業)

今後、以下のような機能をシステムに追加していきたいです。

- 環境センサやmicro:bit歩数計をラズパイとBluetooth接続できていない時でも利用できるように改良したいです。場合によっては、ラズパイ0Wなどの口ガーを使う構成も考えてみたいです。
- 赤外線リモコンで操作できる家電に、クーラーやTVなども追加したいです。

おわりに

自分の知識などが足りなく、あまり面白いものができませんでしたが、意外と便利で、これからも改良して使っていこうかなと思っています。

欲を言えば、外でのデータ収集のために、以下のような腕時計型の「オムロン人間センサ」が欲しいなと思いました。

- 基本的な生体関係データ: 心電計, 脈拍計, 体温計, 酸素飽和度など
- その他のデータ: 歩数計, GPSログ
- オムロン環境センサのデータ: できれば全てのデータが欲しいが、明るさ, UV, 気温, 熱中症度などは色々使えそう

■アピールポイントなど、審査員へのメッセージ

今回、コンテストに応募しようと思ったきっかけや、企画・作成中のエピソード(苦労話や笑い話など)、応募内容に込めた思いなど、審査員に伝えたいことがあればご自由にご記入ください。

応募のきっかけは、とにかく、オムロン環境センサを使ってみたくて、無償提供してもらえるなら頑張ってみようと思いました。どうせなら、既に家にあったスマートスピーカーなどの素材(機材)をなんとかお金をかけずに連携させて、スマートホームを作ってみたくなりました。

このコース(システム)は、お金も時間も無かったので、既にあるレシピ(ソフトウェア)や素材(機材)を組み合わせて、足と頭を酷使して作ってみました。ありものレシピを使ったので、利用言語が Python, Node.js, シェルスクリプト(sh, bash)と色々になったので、頭の中でごっちゃになって戸惑うこともありました。つぎはぎだらけのコースでも、作ってみれば愛着も湧くし、それなりに便利なので良かったです。

一番困ったのは、Bluetooth 接続のセンサ入力が不定期に不安定になることで、動作を監視して、データ収集で問題が起こらない程度の間隔で再起動するという力技で解決しました。食材(機器)配置には、環境センサは照明の明るさが取りやすい場所が必要で、ラズパイ本体は赤外線リモコンが届いてさらに HDMI CEC が使える場所が必要と制約が多く、赤外線リモコンの出力調整で苦勞をしました。

ご記入ありがとうございました。応募方法や審査日程などについては、特設サイト (<http://nkbp.jp/rpic>) をご覧ください。

■問い合わせ先

応募方法などについてご不明な点がある場合は、個人情報取得に関するご説明 (下記参照) にご同意の上、以下の Web サイトにある問い合わせフォームよりご連絡ください。

https://bpcgi.nikkeibp.co.jp/store/req/QA/form/qa_5_2f.shtml

フォームの「雑誌名」欄では「日経 Linux」を選んでください。「発行号」欄や「記事名」欄、「掲載ページ」欄は空白のまま構いません。「お問い合わせ事項」欄に、ご質問内容を記載ください。その際、「みんなのラズパイコンテストについて」の質問であることを明記してください。

※個人情報取得に関するご説明

日経 BP 社は、個人情報保護に関する法令およびその他の規範を遵守し、下記の通り個人情報を取得いたします。

1. 事業者の名称

日経 BP 社

2. 個人情報の管理者

日経 BP 社 個人情報管理責任者

3. 利用目的

- (1) ご購入・ご登録いただいた商品・サービスを提供するため
- (2) DM やアンケートなどをお届けするため
- (3) 事務連絡・お問い合わせ対応のため

4. 第三者提供

個人情報を第三者提供する場合には、提供先の企業/団体名、提供する個人情報の項目を示して、ご本人の同意を得たうえで提供します。また、第三者提供に当たり、提供した年月日、第三者の氏名等、本人の氏名等、個人データの項目などを記録し、その記録を法定期間保存します。

5. 個人情報取り扱いの委託

個人情報に関する機密保持契約を締結している業務委託企業に対して、必要な範囲で個人情報の取り扱いを委託することがあります。

6. 個人情報をご記入いただけない場合について

個人情報をご記入いただかないとサービスを提供できない場合があります。

7. 本人が容易に認識できない方法による個人情報の取得

日経BP社のWebサイトや各種ネットサービスでは、IPアドレス、クッキー、Webビーコンなどを手がかりにして、利用者の皆様のアクセス情報を、自動的に取得することがあります。

8. 匿名加工情報について

匿名加工情報を作成する際には、個人情報保護委員会規則で定める基準に従い当該個人情報を加工し、作成時および第三者提供時にはインターネット等を通じ、当該匿名加工情報に含まれる個人に関する情報の項目を公表します。

9. お問い合わせ窓口

ご登録情報の開示、内容の訂正、利用の停止等を希望される場合は、

日経BP社読者サービスセンター お客さま相談窓口にご連絡ください。

住所：〒134-8729 日本郵便 葛西郵便局 私書箱20号

URL：<https://bpcgi.nikkeibp.co.jp/toiawase.html>

日経BP社の個人情報保護方針 URL：

<http://corporate.nikkeibp.co.jp/information/privacy/>