

Raspberry Pi2 をウェブ・カメラにする

田中 二郎^{1,a)}

概要：Raspberry Pi2 に赤外線カメラを搭載し、夜間でも撮影可能なウェブ・カメラを作成した。撮影した画像はビデオに変換され、1日分を5分弱で見ることができる。使用言語はPython,Perl,sh-script および PostScript である。

キーワード：プログラミング・シンポジウム, 夏, 報告, Raspberry Pi2

1. はじめに

Raspberry Pi2 は,ARM プロセッサを搭載したシングルボードコンピュータで,イギリスのラズベリーパイ財団によって開発されている。CPUは900MHz動作のARM Cortex-A7(4コア)で,メモリは1GB,ストレージはマイクロSDカードを使用し,USBやイーサネットなどの他に,GPIOと呼ばれるデジタル入出力ピンを備えている。

今回は,専用のカメラを接続し,ウェブ・カメラとして使えるように設定した。

2. Lチカとシャットダウンスイッチ

キーボードやディスプレイを接続しないので,動作確認用にLEDを点滅させた。これを通称「Lチカ」という。また,電源断の前にシャットダウン動作をさせる必要があるので,プッシュスイッチを接続し,これを押すことでシャットダウン動作をさせることとした。このため,GPIOの18番にスイッチを,17番にLEDを(抵抗付きで)接続した。

起動時にスイッチが押されていると,設定を初期化して再起動する。起動すると,LEDが点滅する。

動作中にスイッチが押されると,LEDを点灯状態にする。OSのシャットダウンが終わる頃に,LEDは消灯する。LEDが消えて数秒待てば,電源を切ることができる。

GPIOの制御にはPythonのライブラリ(RPi.GPIO)が便利である。

3. カメラ

専用のカメラを専用のコネクタに接続するだけで,最大2592x1944の静止画を撮影できる。撮影用のソフト(raspistill)もあり,簡単に画像をファイルに書き出すことができる。

また,USBにカメラを接続することもできる。この場合もソフト(fswebcam)が準備されているので,手軽にファイル出力することができる。

今回は10秒毎に撮影することとし,シェル・スクリプトで記述した。撮影した画像にはimagemagickを使い,日時をテロップとして入れている。

4. タイムラプス

撮影した静止画像は,1日で8640枚になる。これをつなげてビデオに変換(mencoderを使用)すると30fpsで288秒,5分弱の動画となる。静止画1枚あたりおよそ400KBのデータであり,1日で

¹ 開智国際大学

^{a)} jiro@kaichi.ac.jp

3GB にもなるが、ビデオに変換することにより 1 日 300MB 以下に圧縮される。ビデオに変換した後は、静止画のデータを消去する。

5. データ表示

その日の撮影画像や過去のビデオを外部から参照するには http を使う。Apache2 を使い、シェル・スクリプトで書いた CGI により 1 時間ごとの画像を並べて表示するなど、見やすいインターフェースを心掛けた。

6. ステータス

Raspberry Pi には I2C インターフェースもあり、センサなどの外部機器を簡単に接続できる。今回はポッシュ社の BME280 を接続し、気温・湿度・気圧を測定することとした。センサーは、本体の発熱をさけるため外部に突き出すようにして数 cm 離しているが、それでも本体の発熱の影響をうけている。できれば本体との間にシールドとなるものを追加したい。

また、Raspberry Pi には CPU の温度を読み取る機能もついている。

さらに、CPU の負荷状況や、ネットワークの状態を確認するための ping 値などをまとめて、im-magick によりグラフ表示することとした。

7. ファイル管理

静止画をビデオに圧縮して保存するとはいえ SD カードの容量は小さい。そのため数日分のビデオを保存すると、ストレージがいっぱいになってしまう。そこで、ディスクの使用量が 80%以上になると、古いビデオから消去するようなシェル・スクリプトを用意した。

8. 設定

各種の設定を変更するために、キーボードやディスプレイを接続したり、ssh でログインするのは面倒である。そこで、カメラより QR コードを読み込ませることで各種設定を行えるようにした。

QR コードの読み込みは Python の qrcode を使用し、設定をおこなうプログラムは Perl で記述

した。

9. 音声出力

QR コードなどで設定した値を確認するために、設定値などを音声で出力するようにした。

音声合成には open-jtalk を使用したが、一部の固定メッセージは、別途生成した音声の wav ファイルを使用している。

10. Watchdog

長期間メンテナンスできない状態で運用することを考え、Raspberry Pi にある watchdog 機能を有効にした。万が一 OS がハングした場合には、自動的に再起動される。

11. 応用

今回はカメラに赤外線対応のものを使用し、夜間でも撮影できるようにした。そのため、市販の赤外線投光器と、それ用の昇圧電源もつけている。

12. ケース

一番苦労したのが、ケースである。Raspberry Pi2 用のケースも市販されているが、カメラまで対応しているものはない。そこで、厚紙を利用してケースを自作した。

なお、型紙の印刷のために PostScript を使用した。

13. まとめ

Raspberry Pi は、ハードウェアをいろいろ接続することができ、ソフトウェアもそろっているので、簡単なスクリプトを書くだけで、高性能なシステムをつくることができた。

スクリプト言語にも得手不得手があり、sh では小数点を含む数が扱いにくい。文字列操作をおこなうなら Perl が一番である。Python はライブラリが充実している。今回は、言語にとらわれず、書きやすいことを念頭に、使用言語を選定した。

表に、各プログラムの行数を示す。音声出力 (sound) の行数が多いのは、open-jtalk の引数が多く、数行にわたって記述しているからである。設定

(config.prl) とともに、場合分けが多いのも行数が増える原因となっている。ステータス (status.prl) も、6 種類の情報をグラフ化するために、行数が増えている。

逆に,qrcode.py は、単純にライブラリを呼び出すだけのラップなので、行数が少ない。

今回はパッケージとして配布されているもの以外には、センサ用のプログラム *1, 音声合成用の音声データ *2 を使用している。先人達の業績に感謝したい。

なお、本システムはホームページ *3 で公開している。

用途	ファイル名	行数	言語
シャットダウン	start.py	32	python
専用カメラ	webcam.sh	34	sh
USB カメラ	webcam2.sh	32	sh
タイムラプス	wncode.sh	30	sh
データ表示	log	33	sh
ステータス	stat.prl	75	perl
ファイル管理	clear.sh	14	sh
設定	config.prl	64	perl
QR コード	qrcode.py	9	python
音声出力	sound	150	sh
インストール	setup.sh	66	sh
ケースの型紙	paper.eps	240	postscript

表 1 プログラム一覧

質疑・応答

- カメラに QR コードを見せて設定するアイデアは、よいと思う。

田中 設定値などを音声でフィードバックするのは、他にないのではないかと思います。

- プライバシーは？

田中 ローカルなネット内からのアクセスを想定しているので、外部からの接続の時点でアクセス制御することになる。

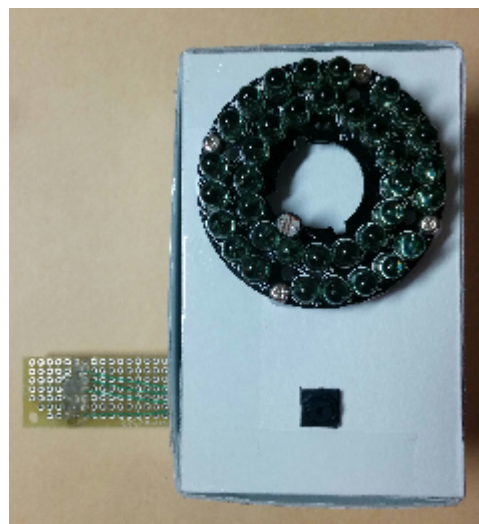


図 1 外観
W60xH90xD55

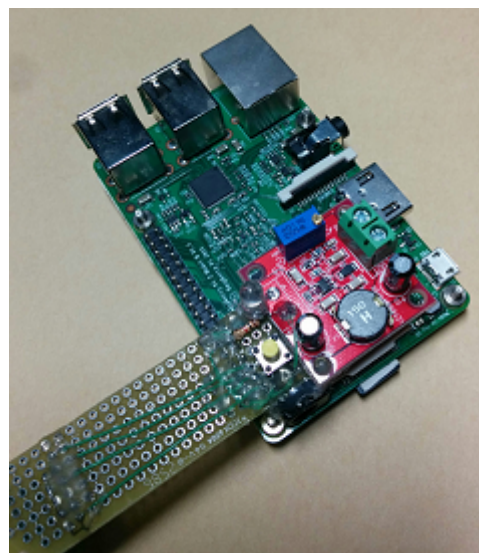


図 2 内部基板
昇圧基板,LED, スイッチ,
センサー (左部裏側)

*1 <https://github.com/SWITCHSCIENCE/BME280>

*2 <http://www.mmdagent.jp/>

*3 <http://gakkan.net/jiro/whoami/pi/>