

～ RaspberryPi を軽く説明するよ ～

[20150719 版]

1. RaspberryPi とは

RaspberryPi (以下 RasPi または RPi と略記) 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ある。英国ラズベリーパイ財団によって、英国で開発された。教育機関でコンピュータ科学の教育に使われることを目的としている。

それほど高価ではなく、Amazon・RS コンポーネンツ・スイッチサイエンスといった通販会社、電子部品販売店店頭などから簡単に購入できる。価格は販売店によりまちまちである。

発売初期は品薄で入手困難だったが、最近では容易に入手可能である。

2015年2月2日に、クアッドコア ARM Cortex-A7 BCM2836 が搭載された「RaspberryPi 2 Model B」が発表された。これにより、従来の ARM11 系 CPU が搭載された製品は RaspberryPi1 と呼ばれるようになっている。

RaspberryPi では、主に Linux 系 OS をインストールして使用する。HDMI とビデオ出力が搭載されており、モニタを接続可能である。また、OS によっては GUI も使用可能である。

プログラミング言語としては、財団から Python が推奨されているが、Python 以外の言語も使用可能。OS として Raspbian を使用する場合は C コンパイラもインストールされている。

また、2015年4月29日に開催されたマイクロソフトの開発者向けイベント「Build2015」にて、RaspberryPi 2 向けに Windows10 IoT Core for RaspberryPi 2 Insider Preview が発表された。現在、開発者向けリリースとして無料で入手可能である(登録が必要)。

以降では、RaspberryPi 1 を RPi1、RaspberryPi 2 を RPi2 と略記する。

2. 機能概要

CPU は、RPi 1 には Broadcom BCM2835(ARM11 ファミリ)が搭載されている。

RPi 2 では Broadcom BCM2836(ARM Cortex-A7)が搭載されている。

動作周波数は、RPi1 は 700MHz。設定変更によりクロックアップ可能。RPi2 では、900MHz となった。

RPi1、RPi2 とも、USB ポート・HDMI ポート・ビデオ出力・音声出力ポート・GPIO ピンなどを備えている。

モデルが数種類あり、各モデルにより、価格とハードウェア構成が異なる。

3. 主要なモデルの説明

RPi1 にはこれまで発売された Model A/A+/B/B+の計 4 種類が存在する。

RPi2 は、現在 ModelB のみである。

各モデルの主な仕様は以下の通り：

	Model A	Model B	Model A+	Model B+	RaspberryPi 2 Model B
メインメモリ	256MB	512MB	256MB	512MB	1GB
LAN	なし	あり	なし	あり	あり
USB	1	2	1	4	4
GPIO	26	←	40	←	←
ストレージ	SD	←	MicroSD	←	←
最大消費電力	2.5W ^{※1}	3.5W ^{※1}	1.5W 未満 ^{※2}	約 2.0W ^{※3}	約 3W ^{※3}

※1 <http://itpro.nikkeibp.co.jp/atcl/column/14/343137/101600003/>

※2 http://ja.wikipedia.org/wiki/Raspberry_Pi

※3 RS コンポーネンツの解説ページによる

<http://www.rs-online.com/designspark/electronics/jpn/blog/raspberry-pi-2-enters-the-multi-core-realm>

その他、Compute module という SO-DIMM 型のバリエーションが発売されている。

Model A+/B+での変更点として、GPIO ポートのピン数の増加があるが、1~26Pin までは ModelA/B の P1 端子と同じ配置となっている。ただし、基板上での場所が違うため、ModelA/B 用に開発された拡張ボードやケースなどを使用する場合は、注意が必要である。また、ModelB+には、ModelA/B にあった P2,P3,P5 がない。

加えて、ModelA+/B+では、ModelA/B からねじ穴の場所が変更されたため、ケースの流用などには注意が必要である。

現在、RPi1 については、ModelA が販売終了となっている。ModelB/A+/B+については、販売は継続されている。また、RPi2 の ModelA は、現状では 2015 年末までの販売予定はないとのことである。

4. RaspberryPi 2 に関する情報

RPi2 では、CPU がクアッドコアとなり、動作周波数も上がったため、最大消費電力が増えている。

ネジ穴位置は RPi1 ModelA+/B+と同じとなっているが、基板上の部品レイアウトが変わったため、ModelA+/B+のケースを流用する際には、ケースを一部加工する必要があるかもしれない。

また、CPU の位置が変更されているため、通気口の位置が合わない場合がある。また、放熱板を使用する際には、放熱板がケースと干渉する可能性がある。

ハードウェアの不具合として、強い光が基板表側に当たることにより、クラッシュするという問題^{※4}がある。これは、RPi2 で WLCSP タイプの電源 IC が採用されたことによる現象である。カメラのフラッシュやレーザーポインタの光など、かなり強い光でなければこの不具合は発生せず、また、原因となる部品に覆いをすることで対策できる。

※4 この件についての参考文献

<http://akkiesoft.hatenablog.jp/entry/20150209/1423491754>

<https://www.raspberrypi.org/forums/viewtopic.php?f=82&t=99466>

5. RasPi で使用可能な OS について

OS としては、Linux ベースのものが多く配布されている。

以下に、主要な OS ・面白そうな OS を挙げる。

(1) Linux ベース

(i) Linux ディストリビューション

- Raspbian

Debian ベース。新しい NOOBS では、デフォルトで含まれるのは Raspbian のみである。OS のサイズは大きい。

現在の最新リリースは 2015-05-05 版。

カーネルのバージョンは 3.18。

- Pidra

Fedora ベース。OS サイズは Raspbian より小さいが、そこそこ大きい。

以下、最新リリースのリリースノートから抜粋：

Pidora 2014 (Raspberry Pi Fedora Remix) will be the fifth version of the Remix released.

◇ Release date: (July 24, 2014)

◇ Fedora package collection version: 20

◇ Kernel version: (3.12.23)

◇ Architecture: armv6hl

- Ubuntu Mate

PC 用のディストリビューション Ubuntu が移植されたもの。

RPi2 専用。

最新リリースは Ubuntu MATE 15.04。

- ArchLinux

OS のサイズが小さい。パッケージ管理ツールは Debian 系でも Fedora 系でもないのので、覚えなおす必要がある。

現在、RaspberryPi.org の公式ダウンロードページでは案内がなくなっている。

(ii) XBMC Media Center 使用

XBMC は無料のオープンソース・メディアプレイヤーアプリケーションで、もともとは初代 XBox 用のメディアセンタとして開発されたアプリある。

非営利組織である XBMC Foundation にて開発されており、Linux、Windows、MacOS などに移植されている。

XBMC が使用可能になっている OS には、以下のものがある：

- OpenELEC
- Raspbmc

なお、OpenELEC では、RPi1 用のバイナリと RPi2 用のバイナリが別に用意されている。

(iii) OSMC

XBMC ベースではない、メディアセンターアプリケーション。

(iv) RISC OS

Acorn Computer が開発し、現在も開発が継続されている、マルチユーザー・疑似マルチタスク（ノンプリエンプティブ）OS。

ライセンスはプロプライエタリである。

(2) Linux 系でない UNIX 系 OS など

以下に、各 OS ユーザコミュニティや個人などの有志で移植されているものを挙げる。

(i) BSD 系

- NetBSD

コミュニティの有志により移植された。日本語の情報もある。

<ftp://ftp.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/README-RPI.pdf>

- FreeBSD on RaspberryPi

<https://wiki.freebsd.org/FreeBSD/arm/Raspberry%20Pi>

(ii) Plan9

<http://bendyworks.com/getting-plan-9-running-on-the-raspberry-pi/>

(3) OS を使わない

上記の OS を使う方法以外にも、普通のマイコンボードのように、特に OS を使わずに RaspberryPi を利用することも可能である。参考文献を以下に挙げる。

(i) ベアメタルで遊ぶ Raspberry Pi (西永俊文 著、秋田純一・出村成和 監修)

<http://tatsu-zine.com/books/raspi-bm>

(ii) mruby を Raspberry Pi(bare metal)で動かす: Windows 編

<http://qiita.com/yamanekko/items/3b1eae1934c4ead618cf>

(4) Windows

RPi 2 向けの Windows10 として、「Windows10 IoT Core for RaspberryPi 2 Insider Preview」が発表され、現在、開発者向けリリースとして無料で入手可能である。これまでダウンロードには登録が必要であったが、現在は登録不要になっている。

OS イメージの SD カードへの書き込みには、Windows10 Insider Preview Build10074 か、Windows8.1 が起動する PC が必要である。

以前は DISM というコマンドを使用して書き込みを行っていた。現在は、GUI を持った書き込みプログラムを使用するようになっており、非常に簡単に書き込みが可能である。

PC で使用する Windows とは違い、インストール直後は特にできることはなく、アプリケーションを作成・インストールして利用する。

Windows の VisualStudio2015 RC を用いてアプリケーション開発を行うことができ、リモートデバッグ・リモート実行などが可能である。

プログラミング例として、Github にて、サンプルプログラムが公開されている。

6. OS イメージの書き込み

SD からのブートが可能なイメージが用意されている場合は、Windows、Mac、Linux などの OS で、各 OS イメージを SD カードへ書き込むツールを使用し、ブート SD を作成する。

利用する OS によって書き込み方法が違う場合があるので、各 OS のインストール説明を参照すること。

6.1. Linux 系 OS

(1) Windows での書き込み

Win32DiskImager を使うことで書き込みが可能。

ダウンロードは以下のページから：

<http://sourceforge.net/projects/win32diskimager/>

(2) MacOS X での書き込み

書き込みツールを利用するか、OS 付属の dd コマンドを使用して書き込みが可能。

書き込みツールとしては、以下のものがある：

- ApplePi Baker
<http://www.tweaking4all.com/hardware/raspberry-pi/macosx-apple-pi-baker/>
使いやすいためオススメ。
- RPi-sd card builder v1.2
<http://n.pentest.ninja/?p=31101>

(3) Linux での書き込み

dd コマンドを使用して書き込む。

(4) NOOBS を使用してのインストール

起動時に OS を選択して起動できるツールである。

オフライン+ネットワークインストール版の場合と、ネットワークインストール専用版とで動作が少々異なる。

- オフライン+ネットワークインストール版
Raspbian を使用する場合はネットワーク接続は不要。その他の OS を選択する場合は、OS イメージを自動で配布サーバからダウンロードしてくるため、インターネットへ接続できる環境が必要である。
- ネットワークインストール専用版
どの OS を選択した場合でも、OS イメージを配布サーバからダウンロードしてくるため、インターネット接続環境が必要。

NOOBS 自体は、イメージで配布されているため、上記(1)~(3)のうちどれかの方法にて SD カードに書き込む必要がある。

6.2. Windows 10 IoT Core

Windows10 IoT Core のインストールには、8 GB 以上の microSD カード(Class10 以降を推奨)が必要である。また、インストールツールを動作させる OS として、Windows8.1 または Windows10 が必要である。

Windows10 を用いる場合、現時点では Windows10 は Insider Preview を用いる。
インストールには、<http://ms-iot.github.io/content/en-US/win10/SetupRPI.htm> から、
専用の書き込みツールと MicroSD に書き込むイメージが含まれる ISO イメージをダウンロードし、
書き込みツールをインストール後、MicroSD のイメージを MicroSD に書き込む。
詳しくは <http://ms-iot.github.io/content/en-US/win10/SetupRPI.htm> に説明が書いてある。

7. GPIO プログラミングについて

以下、Linux ベース OS での GPIO プログラミングについて簡単に説明する。

GPIO へのアクセスには、root 権限が必要である。

これは、GPIO プログラミングはシステムリソースへの直接アクセスそのものであり、また、Linux はマルチユーザ・マルチプロセスの OS なので、システムリソースへのアクセスは厳しく制限されるためである。

このため、GPIO を利用したプログラミングでは、実行ファイルの実行時に、sudo コマンドを用いて、実行ファイルに対し一時的に root 権限を付与して実行する。

以下に、比較的よく使われていると思われる、GPIO を利用するプログラミング環境を挙げる。

(1) WiringPi

C 言語と組み合わせて使うライブラリである。

WiringPi 使用時は、ピン番号は WiringPi が独自に割り当てられているものを使用する。

このピン番号は、物理的なピン配置とは異なるため、注意が必要である。

インストールのためには、開発者の GitHub リポジトリからソースファイル一式をダウンロードして、自分でビルドする必要があるが、ビルドスクリプトが整っており、インストールの難易度は低い。

本家 Web ページ：

<http://wiringpi.com/>

(2) C library for Broadcom BCM 2835 as used in Raspberry Pi

別の C 言語ライブラリ実装。作者によりサンプルプログラムなども公開されている。

※筆者は利用していませんので未検証です

本家 Web ページ：

<http://www.airspayce.com/mikem/bcm2835/>

参考文献：

http://netlog.jpn.org/r271-635/2013/01/raspberry_pi_gpio_test.html

(3) Node.JS

Javascript エンジンから GPIO へアクセスする。

WiringPi をラップしているため、別途 WiringPi のインストールが必要である。

※モクモクラボさんによる使用実績があります

(4) Python

WiringPi を利用して GPIO にアクセスするモジュールが、WiringPi 本家ページで紹介されている。

また、Python のライブラリのみで GPIO にアクセスする「RPi.GPIO モジュール」がある。RPi.GPIO モジュールは、現時点では、I2C/SPI/ハードウェア PWM/Serial をサポートしていない。

※筆者は利用していませんので未検証です

WiringPi Version 2 With Extra Python 本家 Web ページ：

<http://pi.gadgetoid.com/article/wiringpi-version-2-with-extra-python>

RPi.GPIO 本家 Web ページ：

<https://pypi.python.org/pypi/RPi.GPIO>

RaspberryPi ハードウェアレイヤへのプログラミング言語からのアクセスに関する参考文献として、http://elinux.org/RPi_Low-level_peripherals に各環境の情報が集められている。

8. 本資料の変更履歴

● 20150523 版

Windows10 について修正

● 20150523 版

RaspberryPi2 について加筆

Windows10 IoT Core for RaspberryPi2 について加筆

その他修正など