Raspberry Pi 3 Model BによるIoTの実践(2019/09/18改)

第一回Raspberry Pi 3 Model Bの紹介と連載予定 (2017/01/10)	松井博和,	松崎大起
第二回L <u>inuxOSのインストール</u> (2017/03/16 <u>) 注(2019/09/1</u> 8)	松井博和,	島中美羽
第三回Raspbianの設定(前半) (2017/08/22)	松井博和	
補回 R <u>aspbianの設定(後半)(</u> 2017/09/18) <u>注(2017/11/24</u>)	松井博和	
第四回 <u>画像処理のためのOpenCVの導入</u> (2017/09/19)(2019/09/18 <u>改) 注(2017/11</u> /24)	松井博和	
第五回GPIOを用いるLED点滅 (2017/10/19)(2019/09/18改)(トイサーボ制御割愛) 注(2019/09	<mark>/18)</mark> 松井博	和
第六回 <u>ネットワークの設定</u> (2017/11/20)	松井博和	
第七回 <u>リモートログイン(前回続編) (</u> 2018/11/25)	松井博和	
第八回 <u>アナログ入力の使用</u> (spiデバイスとの通信例) (2018/12/17)	松井博和	
補回 RasPi3Bでの音声の入出力 (未定)	松井博和	
来回 Mathmaticaの使用(未定)	松井博和	
来回 Fpga拡張カード(未定)	松井博和	
来回 音声認識(CMU Sphinx)(未定)	松井博和,	松崎大起

この連載の一部は、公益財団法人三重県産業支援センターが毎年3月にその実施を決定し10月から翌年2月まで無料で実施する 高度製造技術基礎講座の一部として演習付きで実施する.本レポートの内容の詳細は、その講座でも解説する.

> 三重大 機械工 メカトロニクス研 松井博和 mailto:hmatsui{at}robot.mach.mie-u.ac.jp Phone: (059)231-9802

Raspberry Pi 3 Model Bによる IoT の実践(第一回)

三重大学 機械工 松井博和, 松崎大起 http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: 超小型パソコン, IoT, 画像処理, ロボット制御

1. Raspberry Pi 3 Model Bの紹介

Fig. 1 にロゴを示す Raspberry Pi は,英国のラズベ リーパイ財団製の名刺大の超小型コンピュータのシリー ズであり,発売約4年で累計1000万台売れている.その



Fig. 1 Raspberry Pi Logo

中でも Fig. 2 に示す最新の Raspberry Pi 3 Model B(以降 RasPi3B)は,日本で税込み 5000円弱で購入でき,無線 LAN や bluetooth をもつ IoT(Internet of Things)の 基として適するものである.RasPi3B は下記特徴をもつ.



Fig. 2 Raspberry Pi 3 Model B

- (1) 名刺大コンピュータ HDMI ポートにディスプレイ, USB ポートに,あるいは,bluetooth でキーボード, マウスを接続すれば,低スペックではあるが,一般の パソコンと同様に使える.インターネットへの接続と して,有線のみでなく無線LANもあり,IoT(Internet of Things)の機器として適している.
- (2) ハードウェアへの接続性 Fig. 2の本体の奥側に並ぶ2列のピン配列は, GPIO (General-Purpose Input/Output)というデジタル入出力ピンである.このピンを用いることで,センサからの入力, LED や モータへの出力を直接制御できる.
- (3)OS 使用例の多くは Linux 系の Debian であるが, Windows 10 IoT Core も使える. Debian には, さま

ざまなフリーソフトウェアが使え, CやPython など さまざまな言語のプログラミング環境も整っている.

- (4) Mathematica 数値演算シミュレータの Mathematica の製造元の Wolfram が, Raspberry Pi 用に無料 ライセンスを提供している.これにより,一般のパ ソコン上で動かすより,約10倍ほど処理が遅いが制 限なく Mathematica を使える.
- (5) Arduino との比較 Arduino は,アナログ入力端子 をもつマイコンであり,ハードウェア制御には向い ている.しかし,インターネット接続には別にシー ルドと呼ばれる拡張機器が必要であり,ソフト開発 環境は別のパソコンに用意する必要がある.

これら上記の特徴により, RasPi3B の USB ポートに WebCam を接続することで,対象に変化があるときのみ に録画する知的遠隔カメラや, RasPi3B の GPIO ポート にモータ制御指令を出させることによるモータ制御,ま た,これらを組み合わせることによる遠隔操作移動ロボッ トなどをつくることができる.

2. 連載記事の予定

Raspberry Pi 3 Model B の実際の使用例を Linux 系 OS をベースに 2017 年 2 月から 9 月まで月一の計 8 回 連続連載する予定である.RasPi3B の情報は,インター ネットや本でも多く公開されているが,当連載記事とし ては,より具体的な実践の手順を中心に解説する予定で ある.(1)RasPi3B の紹介と連載予定(今回),(2)LinuxOS のインストール,(3)OpenCV による画像処理の初 歩,(4)GPIO による LED 点灯とモータ制御,(5) ネッ トワークの設定,(6)Mathmatica のインストールと使用, (7) 音声認識(CMU Sphinx),(8)Fpga 拡張カード.ただ し,これらの予定は変更する可能性がある.

関連演習の予定

この連載の一部は、公益財団法人三重県産業支援セン ターが毎年3月にその実施を決定し10月から翌年2月ま で無料で実施する(仮称)高度製造技術基礎講座の一部 として演習付きで実施する予定である.随時更新する上 記 URL でも確認して下さい.

Raspberry Pi 3 Model Bによる IoT の実践(第二回)

三重大学 機械工 松井博和 , 島中美羽 http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: IoT , Linux , Raspbian

1. RasPi3B用OSのインストール

Raspberry Pi 3 Model B(以下 RasPi3B) を起動するた めの Operating System(OS) には,制限付きの MS 系の Windows 10 IoT Core を無料で用いることもできるが, 非力な PC でも快適に動作する Linux 系 OS の Debian を RasPi 用に調整した Raspbian が一般的には用いら,書籍 やネット上での情報も多い.ここでは,2017/02/16 にリ リースされた Raspbian Jessie with Pixel を用いる.Jessie は,Debian の最新ディストリビューションであり,Pixel は,Pi Improved Xwindows Environment, Lightweight の略で,Xwindow という GUI が軽くなるように改良し たものである.インストール手順は,OS のダウンロード と microSD への書き込みである.

2. LinuxOSの用意

OS データは,(参1) にブラウザでアクセスをすると, Fig. 1 の表示が出て,左側のボタンをクリックすると, 1.5GByte ある zip 圧縮データがダウンロードできる(容 量が大きいので注意).それを unzip して,2017-03-02raspbian-jessie という OS データを得る.



Fig. 1 OS のダウンロード画面

3. MS上でのLinuxOSの書き込み

OS データを Windows 上で microSD(8GByte 以 上) へ書き込む.ここでは, Fig. 2の Transcend の 32GのTS32GUSDHC10Eを Baffaloのカードリーダ (BSCR24EU2BK) に差し, Windows10 PC に USB 接 続で接続し, Win32 Disk Imager というソフトを用い て書き込む.Win32 Disk Imager は, (参 2) にアクセ スし, win32diskimager-1.0.0-install.exe (Date: 2017-03-08, Size: 12.3 MB)をダウンロードしてインストールする.



Fig. 2 使用した microSD とカードリーダ

4. RasPi3Bの起動

Fig. 3のように, RasPi3B に先ほどの microSD を差 し, USB キーボードとマウスを, HDMI ケーブルでモニ タを接続すると, 名刺大のディスクトップ PC として使 用できる. **Fig. 4**は, 電源をつないで OS を起動させた





直後の画面である.



Fig. 4 Raspbian の起動後の画面

- 参1 https://www.raspberrypi.org/downloads/raspbian/
- 参2 https://osdn.net/projects/sfnet_win32diskimager/

注2019/09/18現在では,第二回の記事の通りに実行すると,ホームページの表示が下記のものを含めて最新のDebianパッケー ジであるBusterの3つのパッケージサイズのものがダウンロードできるように変更されている.

8	recommended software Image with desktop and recommended software based on Debian Buster		
	Version:	July 2019	
	Release date:	2019-07-10	
	Kernel version:	4.19	
	Size:	1945 MB	
	Release notes	クリック	
NUA 256	Download Tor	rent 🔁 Download ZIP	

記事と同じものをダウンロードするには,下記URLにアクセスし,「2017-07-05-raspbian-jessie.zip」をダウンロードして下 さい.

http://ftp.jaist.ac.jp/pub/raspberrypi/raspbian/images/raspbian-2017-07-05

Raspberry Pi model 3B+を使用するには、第二回の記事のOSより新しいバージョンが必要で、OpenCVを使うことを考えると、Debian Busterよりは古いパッケージであるDebian Stretchの中で最新である「2019-04-08-raspbian-stretch.zip」を下記URLにアクセスし、ダウンロードして下さい. (フォルダ名とファイル名の日付が異なるので注意) http://ftp.jaist.ac.jp/pub/raspberrypi/raspbian/images/raspbian-2019-04-09

第二回の記事よりも新しいパッケージを用いると,起動直後にsetupのために2,3の質問を答えてと英語で尋ねてくるものがある.ここでは,「Cancel」と「Next」の内,「Cancel」を選択する. 三重大 機械工 メカトロニクス研 松井博和

mailto:hmatsui{at}robot.mach.mie-u.ac.jp Phone: (059)231-9802

Raspberry Pi 3 Model Bによる IoT の実践(第三回)

三重大学 機械工 松井博和

http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: 日本語化, wi-fi 設定

1. Raspbianの日本語化

Raspbian Jessie with Pixel のインストール後の調整と して,日本語化,無線LAN接続を取り上げる.ここでは, 日本語化として,(A)表示メッセージの日本語化,(B)日 本時刻,(C)日本語キーボードの対応,(D)wi-fiの地域設 定,(E)日本語入力を設定する.(A,B,C,D)の作業は,Off Line(インターネット接続なし)でも,実行できるが,(E) は,On Line(インターネットに接続あり)でないと実行で きない.(A,B,C,D)の設定のために,Fig.1の起動画面



Fig. 1 メニュー

の左上のメニューバーの中,一番左の RasPi のロゴをク リック(1)する.そこから出てくるメニューの中の"Preferences"から、さらにサブメニューの"Raspberry Pi Configuration"をクリック (2) する. 出てきた" Raspberry Pi Configuration" ウィンドウの" Localisation" のタグをクリ ック(3) すと, Fig. 2の左上図のようになる. "Set Locale", "Set Timezone", "Set Keyboard", "Set WiFi Country"をクリック (4a)(4b)(4c)(4d) して, それぞれ 4 つの ウィンドウを出し設定する.具体的には,(4a)では,右 上図のように, "ja(Japanese)", "JP(Japan)", "UTF-8"を 選択し, "OK"をクリックする.(4b)では, 左中図のよう に、"Asia","Tokyo"を選択し、"OK"をクリックする.(4c) では,右下図のように,"Japan","Japanese(PC98xx...)" を選択し, "OK"をクリックする.(4d)では, 左下図のよ うに, "JP Japan"を選択し, "OK"をクリックする.こ れらを設定した後、クリック(5)をする、英語で再起動 するか聞かれるので,"OK"をクリックし,日本語化を有 効にする.



Fig. 2 日本語設定

2. 無線 LAN 接続

無線 LAN 接続について, Fig. 3 の左図に示すように, 起動画面の右上のメニューバーの2つの×印の記号をク リック(6)する.すると,現在接続可能な無線 LAN のア クセスポイント(AP)を表示する.図の例では,2つしか AP を表示していないが,場所によっては20個近く表示 される.APの1つである"MieC"をクリック(7)すると, 右下図が表示され,そのAPの接続キー(SSID)を聞いて くる.空白に SSID を入力し,"OK"をクリックすると, 右上図のように,2つの×印の記号が,通信可能であるこ とを示す表示に変更される.この AP への接続は,再起



Fig. 3 無線 LAN の接続

動しても,有効のままである.

3. 日本語入力

ibus-anthy をインストールし,前章と同様に"Preferences"から設定可能であるが,補回で説明する.

Raspberry Pi 3 Model Bによる IoT の実践(補回)

三重大学 機械工 松井博和

http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: 日本語入力,パッケージインストール, OS アップデート

1. 日本語入力設定

第三回の (E) 日本語入力を可能にするには,(1)LAN の 接続をした上で,(2) 日本語入力パッケージのインストー ルをし,(3) 日本語入力設定をする.この補回では,(1) の説明は第三回でしたので,(2)(3) の説明をする.

2. パッケージインストールについて

パッケージは,ターミナルを用いてインストールする. ターミナルは,画面左上にあるメニューバーの左から4 番目のアイコンをクリックすると出てくる.Fig.1にア イコンをクリックしてターミナルが出てきた状態を示す.



Fig. 1 ターミナル

このターミナルに「sudo apt-get install ibus-anthy」と 打ち「Enter キー」を何回か打てば,日本語入力用パッケー ジ ibus-anthy と関連パッケージがインストールされる.

ただし、どんなパッケージもインストールする前に、OS のアップデートをする方が良い.具体的には、ターミナ ルに「, sudo apt-get update」と打ち「Enter キー」を打 ち「, sudo apt-get upgrade」と打ち「Enter キー」を何回 か打つ.updateは、パッケージのデータベースのアップ デートであり、upgradeは、パッケージのデータベース と本機にインストールされているバージョンを比較して、 新しいものがあれば自動入れ替えする.入れ替え数が多 い場合など、入れ替えの是非を確認してくる.

3. 日本語入力設定について

インストールした日本語を入力できるようにするため には,左上のメニューバーから,Fig.2のように,入力 メソッドを選択する.前回の日本語化により,メニュー が日本語されているのが分かる.1つ目の「インプット



Fig. 2 入力設定

メソッド設定」というタイトルのウィンドウが出るので, 「OK(O)」を選択し,2つ目でも「OK(O)」を選択する. 3つ目の Fig. 3に示すウィンドウでは,「ibus」を選択し てから「OK(O)」を選択する.4つ目で「OK(O)」を選 択すると日本語入力ができるようになる.



Fig.3 入力メソッド選択

最後に Fig. 4 のように,右上のメニューバーの「US」 をクリック(1)し,日本語をクリック(2)すれば,「A ち」 となり日本語入力できる.日本語と英語の入力の入れ替 えは,Ctrl キーと j キーを同時に押すことできる.



Fig. 4 言語モード変換

注補回の記事に記述不足があります. 最後の段落の「US」のクリックの前に再起動(システム)が必要です. 三重大 機械工 メカトロニクス研 松井博和 <u>mailto:hmatsui{at}robot.mach.mie-u.ac</u>.jp Phone: (059)231-9802

Raspberry Pi 3 Model Bによる IoT の実践(第四回)

三重大学 機械工 松井博和

http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: **画像処理**, C 言語, OpenCV

1. 画像処理

カメラを用いる画像処理は,工場ラインから自動運転 まで,さまざまなところで用いられる古くて新しい技術 である.カメラセンサーは,他のセンサーに比べて,価 格が安く,センサー密度が高い特徴がある.本報告では, Raspberry Pi 3 Model B に,専用カメラモジュールで なく,Fig. 1(a)のように,一般的な UVC(Usb Video Class)の USB カメラを取り付け画像を取り込み画像表示 をする.画像処理ライブラリとして OpenCV を用い,プ ログラミング言語には C 言語を用いる.

2. OpenCVとは

OpenCVは,フリーライセンスの画像処理のプログラ ムセット(ライブラリ)で,c/c++言語で使えるだけでな く,Java や最近流行りの python で使用することができ る.ここでは,RasPi3B上で動く RaspbianOS上でプロ グラミングをし使用するが,OpenCVは,Linux系OS上 のみならず MS Windows や Mac OS X などのさまざまな OS に対して無料配布されている.そのため,Raspbian上 で作成したプログラムコードを改変することなしに,MS Windows上で使用することもできる.ただし,c/c++の 場合は,再コンパイルは必要である.

3. OpenCVのインストール

補回で記述した方法で OpenCV 用パッケージ libopencv-dev をインストールする.ターミナル上で, OS のアップグレードしてからインストールする.具体 的には、「sudo apt-get update」+「Enter キー」、「sudo apt-get upgrade」+「Enter キー」数回,「sudo apt-get install libopencv-dev」+「Enter キー」数回する.これ で100以上の関連パッケージもインストールされる.



(a)カメラ付きラズパイ

(b)実行時の画面表示

```
Fig. 1 RasPi3B の外観とその実行画面
```

4. OpenCVのサンプルコード

カメラから画像を取り込み画面に表示する c 言語プロ グラムを sample.c という名前で作成する.具体的には, ターミナル上で「leafpad sample.c」+「Enter キー」と 入力し,下記プログラムを入力し保存する.ここでは,プ ログラムの説明を行間に次の行のコメント文として挿入 する.画像処理する場合,「Image に対する画像処理プロ グラム」の箇所に画像処理プログラムを挿入する. #include <cv.h> #include <highgui.h>

main()

ł

- unsigned char key; //キー入力の文字格納用変数 CvCapture *Cam; //カメラ情報用変数 IplImage *Image; //画像情報用変数
- //0 番目のカメラ情報をメモリに展開しポインタを Cam に格納 Cam = cvCreateCameraCapture(0);

//Video という名前のウィンドウを自動サイズで用意 cvNamedWindow("Video", CV_WINDOW_AUTOSIZE);

```
do{ // do-while文
```

- //Cam カメラから画像情報を 1 フレームとる
- Image = cvQueryFrame(Cam); //「Image に対する画像処理プログラム」
- //
- //10msecの間のキー入力を key に代入
 key = (unsigned char)cvWaitKey(10);
 }while(key != 'q'); // 'q' が押されるまで実行

//Cam のメモリ上のデータを開放
 cvReleaseCapture(&Cam);
//Video ウィンドウの崩壊

- cvDestroyWindow("Video");

3

5. コンパイルと実行

ここでは、先ほど作成した sample.c の RaspbianOS 上 でのコンパイルの仕方を示す.OpenCV のコンパイルは、 C 言語で記述しても、c++コンパイラを用いる必要があ る、ターミナル上で、下記のコマンドを改行なしで入力 すると、sample.c がコンパイルされて、sample という実 行ファイルができる.

g++ 'pkg-config --cflags opencv' 'pkg-config --libs opencv' -o sample sample.c

上記中の記号 'は, バッククオートでシングルクオート' ではない. OpenCV のソースコードは, 他の OS で共通

に使えても,コンパイルの仕方は必ずしも共通ではない. 実行ファイル sample をターミナル上で下記のように実 行すると,Fig.1(b)の図中右側のウィンドウのように カメラからの画像を画面に表示する.

./sample

注第四回の記事に記述の分かりにくい部分があります.g++のコンパイルオプションにおいて使用されている記号「`」は, バックオートという記号です.シングルクオート「'」とは異なります. 三重大 機械工 メカトロニクス研 松井博和 <u>mailto:hmatsui{at}robot.mach.mie-u.ac.jp</u> Phone: (059)231-9802

Raspberry Pi 3 Model Bによる IoT の実践(第五回)

三重大学 機械工 松井博和

http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: デジタル入出力, LED 点滅

1. RasPi3BのGPIOへの出力

RasPi3B の特徴の一つは,カード上に 2 列に並 ぶ 40 本のピンヘッダの中に,多くの汎用目的入出力 (GPIO:General Purpose Input/Output) ピンがあること である.これらの GPIO ピンは,それぞれ独立に,1 あ るいは0をH:3.3[V]とL:0.0[V]の2種類の電圧として出 力でき,逆に,ピンの電圧が3.3V付近のHか,0.0付近 のLかを,1か0として入力できる.Fig.1に,40本



Fig. 1 RasPi3B のピンヘッダの配置

のピンヘッダのピン配置を示す.ピン番号は,一番左下 のピンが1番ピンであり,その上が2番ピン,その右下 が3番ピンとジグザグに40番までつけられている.図 中太丸で囲まれる12個のピンは,電源端子を表し,2ピ ンと4ピンにそれぞれある"5V"の記号が5V電源端子で あることを表し,1ピンと17ピンにそれぞれある"3.3" が3.3V電源端子を,図中8個ある"G"が,Ground,す なわち0V電源端子を表す.図中残りの細丸で囲まれる 28個のピンは,GPIO端子を表し,本報告で用いる11ピ ンのGPIO17(と12ピンのGPIO18)だけマークして表し ている.

2. GPIO ライブラリのインストール

ここでは, RasPi3BのGPIOをC言語を用いて制御する.そのために, RasPi3BのRaspbianにwiringPiライ ブラリ(http://wiringpi.com/参照)をインストールする.具体的には,前報告の通りインターネットを接続し, ターミナルを開いて下記の手順を実行する.

git clone git://git.drogon.net/wiringPi
cd wiringPi
git pull origin
./build

上記のコマンドの実行は,次章のプログラムのコンパイ ルと異なり,一度だけすれば良い.

3. 制御プログラム(C言語)

ここでは, Raspberry PI 3 model Bの GPIO17 ピンの 出力をどのように, C 言語を用いて制御するかを例を用い 説明する.実際に, GPIO をセットアップし, GPIO17の ピンを出力ピン設定にして, GPIO17の出力電圧を High 一秒, Low 一秒, High 一秒, Low にして終了するプログ ラムを下記する.

```
/******* GPIO control for Raspberry PI ******/
#include <stdio.h> //標準入出力用ヘッダ
#include <stdlib.h> //標準リブ用ヘッダ
#include <unistd.h> //UNIX 標準ヘッダ
#include <wiringPi.h> //GPIO 制御用ヘッダ
#define GPI017 17
//#define GPI018 18
```

```
void main()
```

{
 if(wiringPiSetupGpio() == -1){
 fprintf(stderr,"GPIO Setup error\n"); exit(1);
 }

pinMode(GPI017, OUTPUT); //GPI017を出力ピンに digitalWrite(GPI017, 1);//GPI017をHighに usleep(1000000); // 1000000 µ sec休憩 digitalWrite(GPI017, 0);//GPI017をLowに usleep(1000000); digitalWrite(GPI017, 1); usleep(1000000); // 1000000 µ sec 休憩 digitalWrite(GPI017, 0); //pinMode(GPI018, INPUT); //GPI018を入力ピンに //printf("GPI018=%d\n", digitalRead(GPI018)); } 上記のプログラムをターミナル上で「leafpad gpio.c」を 実行してファイル gpio.c を作成する.ファイル gpio.c を 下記のコマンドにより , コンパイルしコマンド gpio を作 成する.その後,コマンドgpioを./gpioで実行する. cc -o gpio gpio.c -lwiringPi ./gpio

4. 実行

Fig. 2 に , ./gpio を実行しているときの様子を示す . 発 光している緑の LED は , 内部に適切な抵抗がある LED で , 5V の電圧を LED の両足にそのまま加えても焼けな い . そのため , LED の両足を RasPi3B の GPIO の 2 列 の図中手前側の Pin9 のグランドと Pin11 の GPIO17 に それぞれケーブルのみで直接つなげる . LED には向きが あり , 全く点灯しないときは LED の両足を入れ替える .



Fig. 2 LED の点滅制御

注第五回の記事において, wiringPiは, gitを用いてインストールするように書かれていますが, 2019年9月18日現在ではでき ません. その代わり, wiringPiは, 下記のようにaptを用いてインストールできます. apt-get install wiringpi

三重大 機械工 メカトロニクス研 松井博和 mailto:hmatsui{at}robot.mach.mie-u.ac.jp Phone: (059)231-9802

Raspberry Pi 3 Model Bによる IoT の実践(第六回)

三重大学 機械工 松井博和

http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: インターネットアドレス (IP アドレス), 固定 IP アドレス, DHCP

1. 固定 IP と動的 IP

RaspberryPiを離れたところに設置し、リモートで制 御しようとするとき、無線、有線を問わずに、IPアドレ スを基に通信するのが一般的である、端末の IPアドレス は、一般的には、固定 IPでなく、DHCP(Dynamic Host Configuration Protocol)を用いる動的 IPが用いられる、 動的 IPは、接続する都度、そのネットワーク適した IP と設定をサーバからもらうため、ユーザは IP設定を気に しなくても良い利点がある、その一方、端末でなく接続 先のサーバとして用いる場合、都度変動する動的 IPでは なく、固定 IPを用いる。

2. IP アドレス

IP アドレス (Internet Protocol address) は,イン ターネットアドレスとも呼ばれるインターネットにつ ながるコンピュータがもつ実名である.マシン名 (例: Lily.robot.mach.mie-u.ac.jp)は,通称であり,通信する ときには必ず IP アドレスに変換する.この変換をドメイ ン (例: robot.mach.mie-u.ac.jp) ごとに管理し, それをつ ないで,インターネット全体のマシン名と IP を両方向で 変換するのがドメインネームサーバ:DNS である. IP ア ドレスは基本的にはユニーク(世界に同じ IP アドレスを もつものがない状態)に割り当てられる.多く用いられ る IP アドレスのバージョンには, v4 と v6 の 2 つがある. ここでは,使用例が特に多い IPv4 を説明する. IPv4 ア ドレスは, 32bits で表現されるインターネット通信用ア ドレスであり,通常 8bits ごとの4つに分けられ,それ ぞれの 8bits を 10 進数で表現する. すなわち, IPv4 は, 2³² つまり約 40 億個の PC に番号を割り当てることがで きる.しかし,割り当ての不均衡により現在 IPv4 のアド レスは枯渇してきている.そこで,下記のグローバル IP アドレスには割り当てられていない範囲の IP アドレスを ユニークでなくても良いプライベート(ローカル)アドレ スとして用いる.この IP アドレスでローカルネットワー クを組み,その内の一台だけをグローバルネットワーク と接続することでグローバル IP の数を節約する.このた め, グローバル IP アドレスは世界に1つしかないが,同 じローカル IP アドレスのマシンは世界中に複数ある.

10.0.0.0	_	10.255.255.255	(10.0.0/8)
172.16.0.0	_	172.31.255.255	(172.16.0.0/12)
192.168.0.0	—	192.168.255.255	(192.168.0.0/16)

具体的なネットワークの例として,三重大学機械工メカ トロニクス研究室 AI グループのコンピュータネットワー

ク Fig. 1 を示す. 図中 IP アドレスが並んで示されている のは, 一つのハードウェアが複数の IP アドレスをもつこ とを示している.並んで示される IP 間の矢印は通信がそ の方向へ流れることを示す.グループ内の192.168.1.xxx の IP アドレスをもつ PC は, ディスクトップ PC であ る. これらの PC は, ネットマスクを 255.255.255.0 と してあり, 192.168.1.yyyの IP をもつ PC とは直接通信 する.しかし, 192.168.1.xxxの IP アドレスをもつ PC には, 192.168.1.yyy 以外の IP と通信する場合は, IP ア ドレス 192.168.1.4 の「Pine」をゲートウエイとするよ うに記述してある.具体的には「Lilv」が「Klein」と通 信するとき「Klein」の IP アドレスは 192.168.11.110 で あり 192.168.1.yyy でないので, Pine に通信を転送する. 「Pine」には, 192.168.11.zzzへの通信は, 192.168.1.45の 無線 LAN ルータへ転送することが記述してあり,無線 LAN ルータである ApMieCは, その通信を 192.168.11.1 経由で 192.168.11.110 の「Klein」へ転送する.



Fig. 1 AI グループのコンピュータネットワーク

3. IP アドレスの固定の仕方

ここでは, RasPi3B 用の OS「Raspbian Jessie with Pixel」において固定 IP 化の設定を記述する.そのために, /etc/dhcpcd.conf を編集する.具体的には,前報告と同 様にターミナルを開いて sudo leafpad /etc/dhcpcd.conf として,以下のものをファイルの最後に追加する.

interface wlan0
static ip_address=192.168.11.81/24
static routers=192.168.11.1
static domain_name_servers=192.168.11.1

上記では, IP アドレスを 192.168.11.81 と設定し, ネットマスクを上位 24bits すなわち, 255.255.255.0 と した.また, routers で, ゲートウェイを無線ルータの 192.168.11.1 と設定した.ドメインネームサーバ (DNS) を, 192.168.11.1 と設定した.一般に, DNS は無線 LAN ルータが中継している.DNS の設定がなくても,マシン 名でなく,直接 IP アドレスを用いて通信できる.

Raspberry Pi 3 Model Bによる IoT の実践(第七回)

三重大学 機械工 松井博和

http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: リモートログイン, IP アドレス

1. リモートログイン

リモートログインとは,現在使用しているマシンからイ ンターネットなどの通信経路を用いて別のマシンにログ インすることである.RasPiで多く用いられる Raspbian などの Unix 系 OS では,一般的に暗号化に対応した ssh コマンドを用いる.このとき,ログインする側を ssh ク ライアントと呼び,ログインされる側を ssh サーバと呼 ぶ.リモートログインをするときに,Raspbian では,初 期設定(デフォルト)において,クライアント側は,そ のままで良いが,サーバ側は設定変更が必要である.本 稿は,第六回(本タイトル下の URL 参照)の続編である.

2. ssh サーバ設定

ログインされる側である ssh サーバ設定のためには, 左上のメニューバーから, Fig. 1 のように,「クリック (1)(2)(3)」の場所を順番にクリックし, Fig. 2 の「Raspberry Piの設定」のウィンドウを開かせる.図中の四つあ



Fig. 1 「Raspberry Piの設定」の開き方

るタグの中の「インターフェイス」をクリック(4)し,SSH の「有効」側のラジオボタンをクリック(5)し,OK」を クリック(6)する.以上で,sshサーバの設定が完了する. 補足:Fig.2の中にカメラの項目があり「無効」であ るが,これは RasPi専用のカメラの使用を有効にするも のであり,USB カメラの使用には無関係である.また, RasPi用の A/D,D/A 変換などのガジェットを接続するシ リアル通信の一種である SPIの設定もここで有効にする.

	Raspbe	rry Pi の設定	- •
システム	インターフェイス	パフォーマンス	ス ローカライゼーション
カメラ:	†	◎ 有効	○無効
SSH:	クリック(4)	● 有効	◎ 無効
VNC:		◎ 有効	⊙ 無効
SPI:	クリック(5)	◎ 有効	⊙ 無効
I2C:		◎ 有効	⊙ 無効
シリアル:		◎ 有効	⊙ 無効
1-Wire:		◎ 有効	◎ 無効 クリック(6)
リモートGPIO:		○ 有効	⊙ 無効
			≠ャンセル(C) OK(0)

Fig. 2 インターフェイスの設定

3. ssh コマンドの実行

Fig. 1 に示す「クリック(A)」の場所をクリックしター ミナルを開く.IP:192.168.11.171のマシンにリモートロ グインするときには、ターミナル上で下記を実行する. ssh -X 192.168.11.171

すると,初めて192.168.11.171のマシンにリモートログ インするときには英語で接続を続けるかを (yes/no)で聞 かれ,yes で答えると,サーバのもつ fingerprint をクラ イアント側に登録する.二回目以降は聞いてこない.パ スワードを入力すると,そのターミナルでは,それ以降 192.168.11.171のマシンでの実行となる.そこでの LED 点滅やカメラ画像表示などの実行はリモートになる.



Fig. 3 2 つの RasPi3B を用いるカメラ動画表示の例

Fig. 3(a) は,サーバ側が USB カメラと電源ケーブル の接続,クライアント側がモニタ,キーボード,マウス と電源ケーブルの接続である.クライアント側のターミ ナルから無線 LAN を介して ssh でサーバにログインし, カメラ動画表示を実行するだけで,Fig. 3(b)のように, クライアント側にサーバ側の動画表示ができる.

4. IP アドレスの調べ方

Linux マシンの IP アドレスは,固定 IP でも動的 IP で も,インターネットに接続していれば,ターミナル上で 「hostname -I」を入力し,調べられる.

Raspberry Pi 3 Model Bによる IoT の実践(第八回)

三重大学 機械工 松井博和

http://www.robot.mach.mie-u.ac.jp/~hmatsui/RasPi3B/

Keywords: シリアル通信 (SPI), A/D 変換器 (MCP3002)

1. A/D 変換のための SPI 通信

SPI 通信は, i2c 通信とともによく使われるシリアル 通信の規格の一つである.送信するバイト数と同バイト 数受信する.Raspberry Pi Model 3B では,40 ピンの GPIO(General-Purpose Input/Output)の一部で SPI 通 信ができる.RasPiで SPI 通信をするには,本報告前の第 七回の「ssh サーバ設定」と同様に spi を設定し,第五回 の「GPIO ライブラリのインストール」をする.RasPi3B の GPIO と A/D 変換チップの MCP3002 との接続図を Fig. 1,写真を Fig. 2 に載せる.



Fig. 1 RasPi3B と MCP3002 との接続図



Fig. 2 RasPi3B と MCP3002 との接続写真

ここでは, A/D 変換チップの CH0 に与える入力電圧 として,ファンクションジェネレータから最低 0V から最 高 5V で 0.5 秒上昇, 0.5 秒下降の1 秒周期のランプ入力 を用いる.

2. SPI通信する C プログラム

本プログラムは, SPI 通信で 2bytes データを RasPi3B から送り, 2bytes データをガジェットから受け取るプロ グラムである.A/D 変換チップの MCP3002 は, 0x68 と 0x00 の 2 バイトを送ると, CH0 の電圧を 10bits で量子 化し,それを 2bits と 8bits の 2 バイトデータで返す.

```
#include <wiringPiSPI.h>
#include <wiringPi.h>
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#define SsPort 8 //GPI08
#define SpiChannel 0 //SPI チャンネル
#define BuffSize 2
#define SpiSpeed 10000000 //通信速度(100kHz)
void main()
 int i;
 unsigned char spiBuff[BuffSize];//送受信用バッファ
 //SPI チャンネル初期化
 if((wiringPiSPISetup (SpiChannel, SpiSpeed)) < 0){</pre>
   fprintf(stderr, "SPISetup error\n");exit(1);
 }
 if(wiringPiSetupGpio() == -1){//GPIO 初期化
   fprintf(stderr, "SetupGpio error\n");exit(1);
 }
 pinMode(SsPort, OUTPUT);
                        //SSpin を出力に設定
 digitalWrite(SsPort, 1);
                        //SS 信号初期化
 fprintf(stderr, "Start SPI\n");
 for(i=0; i<200; i++ ){//2 秒間計 200 点のサンプリング
   spiBuff[0] = 0x68; spiBuff[1] = 0x00; //CH0 用
   //SPI 通信実行
   digitalWrite(SsPort, 0);
                         //通信開始
   //BuffSize のデータを送受信する.
   wiringPiSPIDataRW(SpiChannel, spiBuff, BuffSize);
   digitalWrite(SsPort, 1); //通信終了
   printf("%4.2f %d\n",0.01*i,spiBuff[0]*256+spiBuff[1] );
   usleep(10*1000);// 10msec インターバル
 fprintf(stderr,"End SPI...\n");
7
上記プログラムをターミナル上の下記コマンドにてコン
パイルし,実行ファイル spiを作り,実行する.
$ cc -o spi spi.c -lwiringPi
```

\$ cc -o spi spi.c -1wiringPi \$./spi Start SPI 0.00 234 0.01 213 197 行の省略 1.99 211 End SPI...

上記のデータをプロットするとファンジェネの出力波形 とほぼ同型の Fig. 3 のグラフを得 , A/D 変換がうまく 動作していることが分かる .

