会津大学 講習会

ブレッドボードに LED を配線してプログラ ムから LED を点灯

 $\mathbbm{C}$  2018 The University of Aizu

# 目次

1		課	題		1
	1.1		課題	〔説明	1
2		機	材記	论明	<b>2</b>
	2.1		Ras	pberry Pi	<b>2</b>
	2.2		ブレ	ィッドボード	3
	2.3		LE	D	4
	2.4		ジャ	マンパー	4
	2.5		抵打	ւ Լ․․․․․	<b>5</b>
	2.	5.	1	抵抗の見分け方	<b>5</b>
	2.	5.	2	抵抗の値の求め方	6
3		デ	ジタ	マル出力	9
	3.1		デシ	<sup>ジ</sup> タル出力について	9
	3.2		Ras	pberry Pi でのデジタル出力の使い方	9
	3	.2.	1	Python での使い方	9
	3	.2.	1	モジュールの導入と初期化処理	9
	3 3	.2. .2.	1 $2$	<b>モジュールの導入と初期化処理</b> 出力の処理1	9
4	3 3	.2. .2.2 回	1 2 路代	モジュールの導入と初期化処理 出力の処理	9 .0 .1
4	3 3 4.1	.2. .2.: 回	1 2 路代 LE]	<b>モジュールの導入と初期化処理</b> 出力の処理1 F成	9 .0 .1 .1
4	3 3 4.1 4.2	.2. .2.5 回	1 2 <b>路作</b> LEI 抵打	<b>モジュールの導入と初期化処理</b>	9 .0 .1 .1 .2
4	3 3 4.1 4.2 4.3	.2. .2.: 回	1 2 路作 LEI 抵却 Ras	<b>モジュールの導入と初期化処理</b>	9 .0 .1 .1 .2 .2
4	3 3 4.1 4.2 4.3 4.4	.2. .2.: 回	1 2 <b>路作</b> LEI 抵却 Ras Ras	<ul> <li>モジュールの導入と初期化処理</li> <li>出力の処理</li> <li>F成</li> <li>D を差し込む</li> <li>1</li> <li>tを差し込む</li> <li>pberry Pi と接続</li> <li>pberry Pi に電源を入れる</li> </ul>	9 .0 .1 .1 .2 .2 .3
4	3 3 4.1 4.2 4.3 4.4	.2. .2. 回	1 2 LEI 抵却 Ras てつく	<ul> <li>モジュールの導入と初期化処理</li> <li>出力の処理</li> <li>F成</li> <li>D を差し込む</li> <li>1</li> <li>tを差し込む</li> <li>npberry Pi と接続</li> <li>npberry Pi に電源を入れる</li> <li>ゴ</li> <li>ブラム</li> </ul>	9 .0 .1 .1 .2 .2 .3 .4
4	3 3 4.1 4.2 4.3 4.4 5.1	.2. .2.: 回 プ	1 2 路代 LEI 技 Rass アロプロ	モジュールの導入と初期化処理       1         出力の処理       1         F成       1         D を差し込む       1         広を差し込む       1         pberry Pi と接続       1         opberry Pi に電源を入れる       1         ブラム       1         ログラムの流れ       1	9 .0 .1 .1 .2 .2 .3 .4 .4
4	3 3 4.1 4.2 4.3 4.4 5.1 5.2	.2. .2.: 回	1 2 路LEI括Rass クロプ点	モジュールの導入と初期化処理       1         出力の処理       1         F成       1         D を差し込む       1         広を差し込む       1         opberry Pi と接続       1         opberry Pi に電源を入れる       1         ゴグラムの流れ       1         「のプログラム       1	9 .0 .1 .1 .2 .2 .3 .4 .4 .4
4	3 3 4.1 4.2 4.3 4.4 5.1 5.2 5	.2. .2. 回 プ .2.	1 2 路LEI 打 Rass プ 点 1	モジュールの導入と初期化処理       1         出力の処理       1         F成       1         D を差し込む       1         Loberry Pi と接続       1         upberry Pi と接続       1         upberry Pi に電源を入れる       1         uグラムの流れ       1         uのプログラム       1         uのプログラム       1         uのプログラム       1         uのプログラム       1	9 .0 .1 .2 .2 .3 .4 .4 .4 .4
4	3 3 4.1 4.2 4.3 4.4 5.1 5.2 5 5 5	.2. .2.: 回 プ .2.: .2.:	1 2 路L抵RRロプ点1 2 作II 打 ass クロガ	モジュールの導入と初期化処理1出力の処理1F成1D を差し込む1D を差し込む1D を差し込む1Ipberry Pi と接続1Ipberry Pi に電源を入れる1ブラム1ブラムの流れ1「のプログラム1Raspberry Pi ヘプログラムを転送1	9 .0 .1 .2 .2 .3 .4 .4 .4 .4 .4 .5
4	3 4.1 4.2 4.3 4.4 5.1 5.2 5 5.3	.2. .2.: 回 .2.: .2.:	1 2 路EI 抵 R ass クロ パ に り フ に り 2 プ	モジュールの導入と初期化処理1出力の処理1F成1D を差し込む1D を差し込む1D を差し込む1Ipberry Pi と接続1Ipberry Pi に電源を入れる1ブラム1ブラムの流れ1パフログラム1Raspberry Pi ヘプログラムを転送1ログラムの実行1	9 .0 .1 .2 .2 .3 .4 .4 .4 .4 .5 .6

# 1 課題

### 1.1 課題説明

ブレッドボードに LED を差し込み,Raspberry Pi と接続してプログラムから LED を点灯さ せます。



# 2 機材説明

この課題は以下の機材を使用します。手元にあることを確認してください。

機材	個数
Raspberry Pi	1
ブレッドボード	1
LED	1
抵抗 100Ω	1
ジャンパー(オス-メス)	2

# 2.1 Raspberry Pi

Raspberry Pi は小型のコンピュータです。ハードウェアにつなげやすく電子工作に使われます。



今回使用する	Raspberry Pi	lt[Raspberry	pi B+]です。	スペック	は以下になります。
--------	--------------	--------------	-----------	------	-----------

	スペック
CPU	700 MHz/ ARM1176JZF-S ⊐7
メモリ	512  MB
USB ポート	4
ネットワーク	10/100Mbps イーサネット
ビデオ出力	HDMI, DSI
音声出力	3.5mm ジャック
低レベル出力	GPIO×40(GPIO, UART, I2C, SPI, 3.3V, 5V, GND など)
必要電源	5V, Micro USB Micro-B
サイズ	$86 \times 56 \times 18$ mm (約 45g)

Raspberry Pi には GPIO というインターフェースがあります。この GPIO を使用して電子回路を制御します。



 3.3V, 5V:電圧を常に供給するピン(ON/OFFは出来ない) 電池でいうプラスに当たる。
 GND:電圧が0になるピン。電池でいうマイナスに当たる。
 GPIO:デジタル入出力が出来るピン。
 I2C: I2C通信に関わるピン。SDAとSCLが当たる。
 SPI:SPI通信に関わるピン。
 MOSI, MISO, SCLK, CEが当たる

### 2.2 ブレッドボード

電子回路を作成するためのボード。穴がありその中に LED やジャンパーを差し込み、回路を 作成します。

10 ---12 --E 12 ------25 29 00 2 9 18 6 20 22 33 24 26 27 28 0 2 3 4 -5 10 (0) 8 0 -D 0 0) 0 O 0 0 0 5 5 2 3 4 9 2 8 9 20 21 22 23 24 25 25 26 5 83 63 08 +



https://elinux.org/File:Pi-GPIO-header.png

#### 2.3 LED

電流を流すと光る電子部品。照明やイルミネーションに使用されています。 今回使用する LED の特徴は以下になります。

データシート		
https://www.m	arutsu.co.jp/pc/i/36030/	
<b>特</b> 徵(p.1:Ch	aracters 参照)	
・赤色の高輝	译度 LED	
<ul> <li>明るさは5</li> </ul>	5mcd	
電圧・電流 (p.	.1 : Electrical-Optical Characteristics 参照)	
・ 順電圧:+	- 1.7V ~+ 2.4V (標準電圧 1.9V)	
値の範囲を	·超えると LED が壊れるという値です。	
・ 順電流:20	0mA	
順電流はこ	の値以下までに電流を抑えるという意味です。	
端子(p.2:Ou	tline Dimensions 参照)	
		20DB
端子名	説明	~
ANODE	LED の長い方。プラス極のこと	a ter st
	を指す。	e e
CATHODE	LED の短い方。マイナス極のこ	
	とを指す。	
※接続時、端子	その向きに注意!!逆向きにすると電流が流れません	<b>к</b>
		CATHO ANODE

# 2.4 ジャンパー

ブレッドボード上や電子部品を接続させるための線。この講習会では2種類使用する。

(T, T, T
---



この課題ではジャンパー(オス・メス)を使用

## 2.5 抵抗

LED を点灯させる際に必要以上の電流が流れると LED が破損します。必要以上の電流が流れ ないように電流の量を調節するためのものを抵抗といいます。



#### 2.5.1 抵抗の見分け方

抵抗の値は印刷された色で判断します。抵抗には4本又は5本の色が印刷されており、その色 から抵抗の値や誤差の範囲が分かるようになっています。



赤	2	2	2	100	$\pm 2\%$
橙	3	3	3	1000	±3%
黄	4	4	4	100000	±4%
緑	5	5	5	1000000	$\pm 0.5\%$
青	6	6	6	1000000	$\pm 0.25\%$
紫	7	7	7	10000000	$\pm 0.1\%$
灰	8	8	8	100000000	
白	9	9	9	1000000000	
金				.1	$\pm 5\%$
銀				.01	±10%

例) 下図の4本線の場合は以下のように計算します。



### 2.5.2 抵抗の値の求め方

LED を電子回路に組み込むとき抵抗が必要になりますが、どれでもいいわけではなく LED の 順電圧・順電流や電源の電圧によって必要な抵抗の値が変わります。 抵抗の値は[オームの法則]で求めることができます。

E(V) =	R(Ω)	×	I(A)
電圧	抵抗		電流

Raspberry Piと今回使用する LED を例に考えてみましょう。



LED を上図の様に直列に接続します。 この時の LED の電気特性は以下のようになります 順電圧: 1.9V 順電流: 20mA GPIO ピン出力電圧: [3.3V]



直列回路の場合、電圧・電流は以下の関係になります。



直列回路の場合、各抵抗の電圧の和が出力電圧になること、電流は常に一定の大きさであることを考えると、下図のようになります。



抵抗にかかる電圧は出力電圧から LED の順電圧を引いた値 1.4V。電流は 20mA になります。 これをオームの法則に当てはめると式は以下になります。

(3.3V-1.9V)=抵抗×0.02A

抵抗=1.4÷0.02

抵抗=70Ω

抵抗は 70Ω になります。

従って使用する抵抗の値は 70Ω 以上になります。それ以下だと電流が 20mA 以上になるので LED が破損する可能性があります。今回は 100Ωの抵抗を使います。

# 3 デジタル出力

# 3.1 デジタル出力について

値を ON(1)か OFF(0)で出力する方式

Raspberry Pi だと電圧を「3.3V 出力」「0V 出力」の2種類を出力する

## 3.2 Raspberry Pi でのデジタル出力の使い方

#### 3.2.1 Python での使い方

Python でデジタル出力をするには RPi.GPIO を使用します。これは GPIO を使用するための モジュールで Raspbian に標準でインストールされている Python ライブラリです。

#### 3.2.1 モジュールの導入と初期化処理

Python プログラムで GPIO モジュールを使用するには import 文で以下のモジュールを読み込 みます。

import RPi.GPIO as GPIO # GPIOを使用するためのモジュール

モジュールの読み込みが出来ましたら GPIO ピンの初期化を行います。

この初期化の処理でピン番号の設定の仕方とどのピンをどのモードで使用するか決めます。

<pre>GPI0.setmode(GPI0.BCM)</pre>		(GPIO.BCM)	# GPIO ピンの設定の仕方			
GPIO.setup(4,GPIO.OUT)			# GPI04 を出力として設定			
Pin:           +3V3           GPI02 / SDA1           GPI03 / SCL1           GPI03 / SCL1           GPI04           GPI07           GPI07           GPI07           GPI07           GPI07           GPI07           GPI07           GPI07           GPI09 / MISO           GPI01 / MISO           GPI01 / SCLK           GPI00 / ID_SD           GPI00 / ID_SD           GPI013           GPI019 / MISO           GPI019 / MISO           GPI019 / MISO           GPI010 / D_SD           GPI010 / MISO           GPI010 / D_SD           GPI010 / MISO           GPI010 / MISO		in 2 +5V +5V GND TXD0 / GPI0 14 RXD0 / GPI0 15 GPI0 18 GND GPI0 23 GPI0 23 GPI0 23 GPI0 24 GND GPI0 25 CE0# / GPI08 CE1# / GPI07 ID_SC / GPI01 GND GPI012 GND CE2# / GPI016 MOSI / GPI020 SCLK / GPI021 in 40	ピンの設定の仕方は以下の種類があります。 ・GPIO.BCM(役割ピン番号):GPIO の数字で指定 ・GPIO.BOARD(物理ピン番号):左上から順の番号で指定 ピン番号 7 番目の GPIO4 を使用するとき GPIO.BCM は 4 GPIO.BOARD は 7 になります。			

3.2.2 出力の処理

GPIO ピンの出力電圧は通常 0V になっています。以下のプログラムから GPIO ピンの出力電 圧を 3.3V に変更できます。

GPIO.output(4, True)#GPIO4の出力電圧を3.3Vに変更

出力電圧を 0V に変更するには True から False に変更します。

GPIO.output( 4, False )# GPIO4の出力電圧を 0V に変更

# 4 回路作成

ブレッドボードにLEDを差し回路を作成します。 完成図は以下になります。



# 4.1 LED を差し込む

LED をブレッドボードに差し込みます。LED の短い線をブレッドボードの E6 に長い方を E7 に差し込みます。



# 4.2 抵抗を差し込む

抵抗を差し込みます。抵抗にはプラスマイナスがないので向きは気にする必要はありません。 C6 と C1 に差し込みます



# 4.3 Raspberry Piと接続

Raspberry Pi とブレッドボードの回路を接続します。ジャンパー(オス - メス)のオスをブレッドボートに、メスを Raspberry Pi の GPIO ピンに差します。A7 と GPIO4 を接続し、A1 と GND を接続します。



回路側	Raspberry Pi
LED のプラス	GPI04 (PIN 番号 7)
抵抗	GND (PIN 番号 6)

# 4.4 Raspberry Pi に電源を入れる

Raspberry Pi にモバイルバッテリを接続し、Raspberry Pi に電源を供給します。



# 5 プログラム

LED が2秒間点灯して消灯するプログラムを作成します。

# 5.1 プログラムの流れ

LED を2秒間点灯して消灯するプログラムは以下のような処理となっています。

 GPIO モジュールの導入 Raspberry Pi の GPIO を使用するためのライブラリの導入
 初期化処理

どの GPIO をどのモードで使用するかの設定

点灯, 消灯処理
 電流を2秒流して点灯後, 電流を流すのをやめて, LED を消灯させる

### 5.2 点灯のプログラム

#### 5.2.1 プログラム

以下のプログラムで LED を点灯させられます。 一部「TODO:」とコメントアウトしていますので、その部分の処理を作成してください。

# coding: utf-8 import RPi.GPIO as GPIO #GPOI を使用するためのモジュール import time #sleep を使用するためのモジュール # Init LEDPIN = # TODO:使用する GPIO の数字# GPIO.setmode( GPIO.BCM )#GPIO ピンの設定の仕方 (GPIO の数字で指定) GPIO.setup( LEDPIN, # TODO:出力設定# ) # 出力として使用 # Program Start GPIO.output( LEDPIN, # TODO:ON にする場合の値# )#LED を ON にする time.sleep( 2.0 )#2 秒間停止する GPIO.output( LEDPIN, # TODO:OFF にする場合の値# )#LED を OFF にする

GPIO.cleanup() #GPIOの使用を終了

テキストにコピーして「01\_LED.py」のファイル名で保存してください。 コピーが出来ない方は以下 URL からダウンロードしてください。 https://rtc-fukushima.jp/wp/wp-content/uploads/2019/01/01\_LED.zip

### 5.2.2 Raspberry Pi ヘプログラムを転送

プログラム作成後、Raspberry Pi に転送します。今回は「Tera Term」を利用します。 ※Tera Term をダブルクリックして起動してください。

• Tera Term

Tera Term: 新しい接続	ホスト(T):ホスト名.local 例)
• TCP/IP ホスト(T): myhost.local	I
▼ヒストリ(O) サービス: O Telnet TC © SSH_SSHバー	:Pポート#(P): 22 -ジョン(V): SSH2 ▼
っその他 ブロ	H⊐JL(O): UNSPEC▼
○ シリアル ボート(R): COM1: 通信ボート	F (COM1)
OK キャンセル	へルプ(H)

TeraTerm を起動すると接続ダイアログが現れるので、接続する Raspberry Piのホスト名 +.local 、または、Raspberry Piの IP アドレスを「ホスト」のテキストボックスに入力し OK を押します。今回の講習会ではホスト名は箱の付箋に記載されています。

下図の画面が出ましたらユーザ名とパスフレーズを入力してログインします。

ユーザ名: pi パスフレーズ: raspberry

SSH認証 ログイン中: raspA1J 認証が必要です. ユーザ名( <u>N</u> ):	local pi	ユーザ名(N) : pi パスフレーズ(P) :	raspberry	イ警告 hostsリスト中のサーバでra 続のホスト線が、キー安して ストが、持後によっとしている 能性もありますので、十分〉	spA1.local"のエントリ しません* 悪意を持 サーバのふりをして 注意してください.	
バスフレーズ( <u>P</u> ):	: ▼ パスワードをメモリ上に記憶する(M) ■ エージェント転送する(Q)			known hostsリストにこのホストを追加して続行す 次回からこの警告は出なくなります。		
◎ プレインテキス	.トを使う( <u>L</u> )			サーバ側のホスト鍵指紋: 5f:27:b5:99:2d:cc:19:a3:b2:ce:d	0:af:44:b9:d2:f1	
<ul> <li><u>B</u>SA/DSA鍵を</li> <li>rhosts(SSH1)?</li> </ul>	(使う レーカル を使う ローカル ホスト鍵(E):	密鍵(S): のユーザ名(U):		+[ECDSA 256]+ + S = + X S = + X = B o	この画面だ 続行ボタン	が出ました ンを押下し
◎ チャレンジレス	ボンス認証を使う(キーボ	ードインタラクティブ)( <u>C</u> )		o B E *. +	てください	, <b>)</b> <sub>0</sub>
○ Pageantを使う	ОК			<ul> <li>既存の鍵を新いて吸で</li> <li>続行(C)</li> </ul>	E書きする( <u>R</u> ) 8続断( <u>D</u> )	

今回は Tera Term の「SSH SCP …」を使用してファイルをコピーします。Tera Term の 「ファイルメニュー」→「SSH SCP …」を選択します。

Erom:				Send	
To:	~/			<u>C</u> ancel	
	You can drag	h To:が空欄の 「/home/pi/	)場合[~, 1と入力	/]か してくだ	
Fr <u>o</u> m:		さい	_		
To:	C:¥Program Files (x86)¥teraterm				

<u>From</u> に[01\_LED.py]選択し、<u>S</u>end ボタンをクリックします。

# 5.3 プログラムの実行

\$ python 01\_LED.py

LED が赤く2秒間光れば成功です。

# 5.4 Raspberry Pi シャットダウン

Raspberry Pi をシャットダウンは以下のコマンドを行います。

\$ sudo halt

シャットダウン後十数秒後 Raspberry Pi からモバイルバッテリを抜いてください。