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

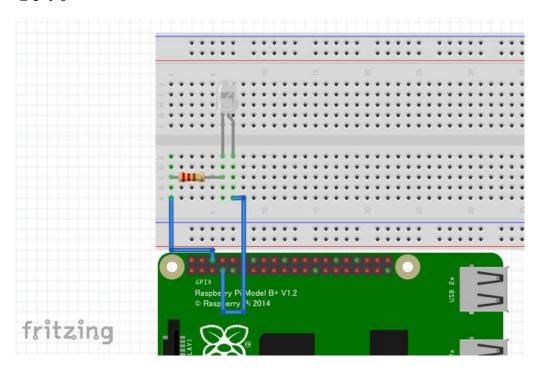
# 目次

1		課題	<u> </u>	1
	1.1	誹	果題説明	1
2		機材	才説明	2
	2.1	R	aspberry Pi	2
	2.2	ブ	ブレッドボード	3
	2.3	L	ED	4
	2.4	ジ	<sup>ジ</sup> ャンパー	4
	2.5	担	氐抗	5
	2.	. 5. 1	抵抗の見分け方	5
	2.	. 5. 2	2 抵抗の値の求め方	6
3		デシ	ジタル出力	9
	3.1	テ	デジタル出力について	9
	3.2	R	aspberry 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.	.2.2	出力の処理	10
4		回路	8作成	11
	4.1	L	ED を差し込む	11
	4.2	担	<b>氐抗を差し込む</b>	12
	4.3		Caspberry Pi と接続	
	4.4		Caspberry Pi に電源を入れる	
5		プロ	ュグラム	14
	5.1		プログラムの流れ	
	5.2	点	京灯のプログラム	
	5	.2.1	• •	
	-	.2.2	, , , , , , , , , , , , , , , , , , ,	
	5.3		プログラムの実行	
	5.4	$\mathbf{R}$	aspberry Pi シャットダウン	16

# 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 は [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 コア	
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	86×56×18mm (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。

GPIO2 / SDA1 +5V 0 GPIO3 / SCL1 GND GPIO4 TXD0 / GPIO 14 RXD0 / GPIO 15 GND GPIO 18 GPIO27 GND GPIO22 GPIO 23 +3V3 GPIO 24 GPIO10 / MOSI GND 0 0 GPIO9 / MISO CE0# / GPIO8 GPIO11 / SCLK GND CE1# / GPIO7 GPIO0 / ID\_SD ID\_SC / GPIO1 GPI05 GND GPIO13 GND GPIO19 / MISO CE2# / GPIO16 GPIO26 MOSI / GPIO20 GND O O SCLK / GPI021 Pin 39 Pin 40

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

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。

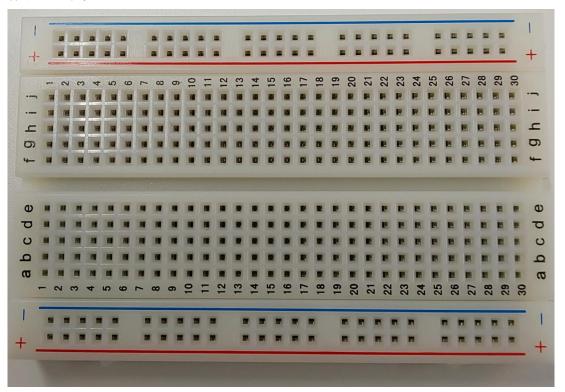
I2C: I2C 通信に関わるピン。SDA と SCL が当たる。

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。



#### 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.marutsu.co.jp/pc/i/36030/

#### 特徴 (p.1: Characters 参照)

- · 赤色の高輝度 LED
- 明るさは55mcd

## 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:20mA

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: Outline Dimensions 参照)

端子名	説明
ANODE	LED の長い方。プラス極のこと
	を指す。
CATHODE	LED の短い方。マイナス極のこ
	とを指す。

※接続時、端子の向きに注意!!逆向きにすると電流が流れません

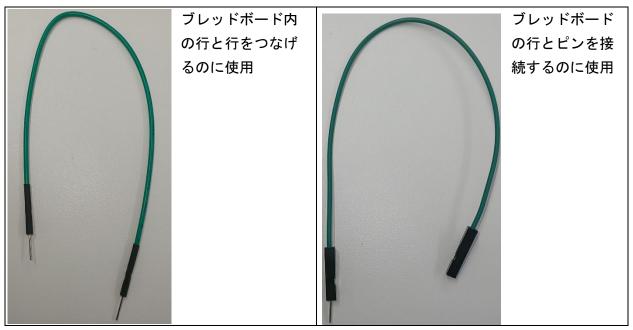


ANODE

## 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。

ジャンパー (オス-オス) ジャンパー (オス-メス)



この課題ではジャンパー(オス-メス)を使用

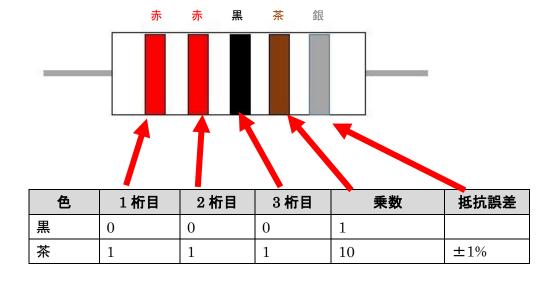
## 2.5 抵抗

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

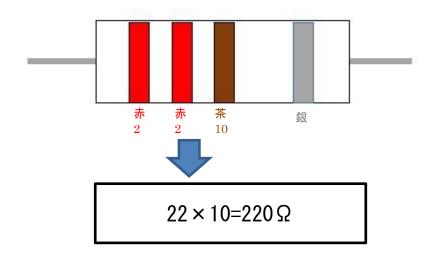
抵抗の値は印刷された色で判断します。抵抗には4本又は5本の色が印刷されており、その色から抵抗の値や誤差の範囲が分かるようになっています。



デュアルウェア講習会課題1

赤	2	2	2	100	±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	10000000	$\pm 0.25\%$
紫	7	7	7	100000000	±0.1%
灰	8	8	8	1000000000	
白	9	9	9	10000000000	
金				.1	±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

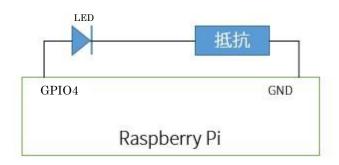
LED を電子回路に組み込むとき抵抗が必要になりますが、どれでもいいわけではなく LED の順電圧・順電流や電源の電圧によって必要な抵抗の値が変わります。

抵抗の値は[オームの法則]で求めることができます。

 $E(V) = R(\Omega) \times I(A)$ 

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。

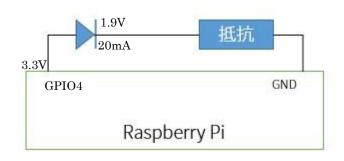


LED を上図の様に直列に接続します。

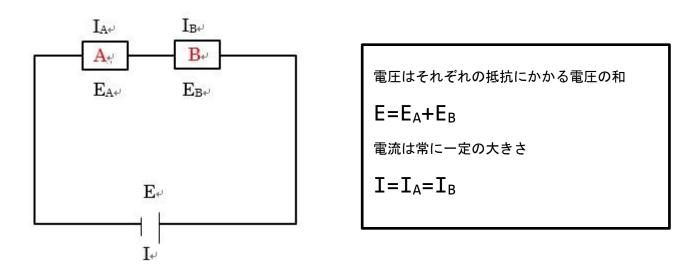
この時の LED の電気特性は以下のようになります

順電圧: 1.9V 順電流: 20mA

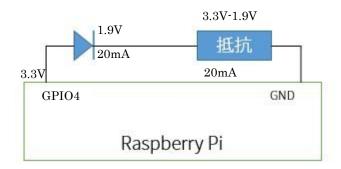
GPIO ピン出力電圧: [3.3V]



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  $1.4 \rm V$ 。電流は  $20 \rm mA$  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。

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  $70\Omega$  以上になります。それ以下だと電流が 20 mA 以上になるので LED が破損する可能性があります。今回は  $100\Omega$  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。

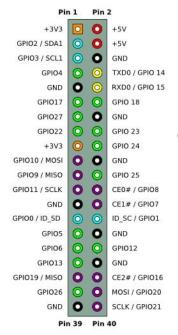
この初期化の処理でピン番号の設定の仕方とどのピンをどのモードで使用するか決めます。

GPIO.setmode(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



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。

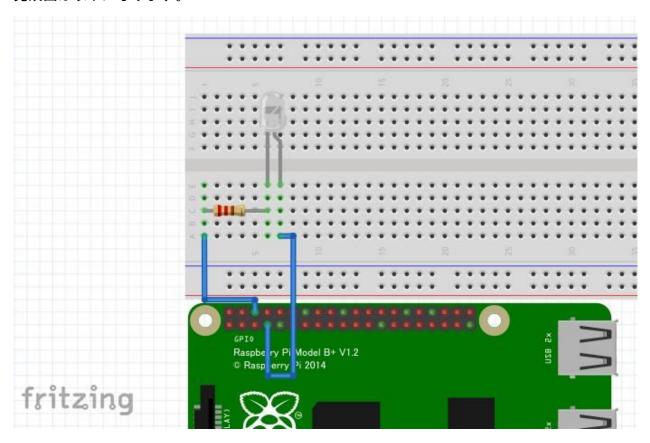
GPIO.output(4, True)#GPIO4の出力電圧を3.3Vに変更

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 ØV に変更

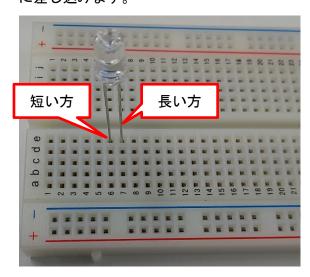
# 4 回路作成

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。

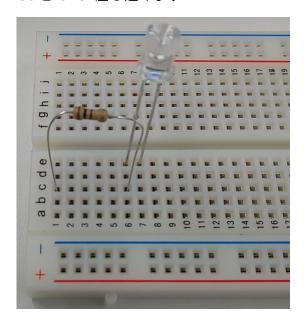


## 4.1 LED を差し込む

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。

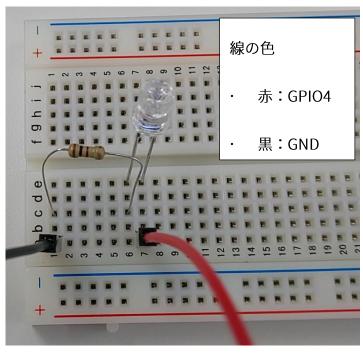


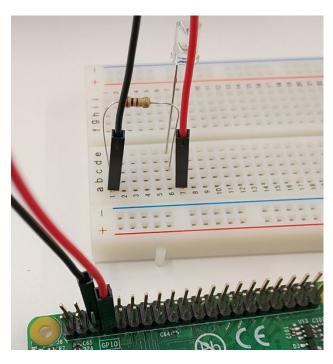
## 4.2 抵抗を差し込む



## 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
- 3. 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

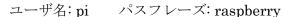
プログラム作成後、Raspberry Pi に転送します。今回は「Tera Term」を利用します。 ※Tera Term をダブルクリックして起動してください。

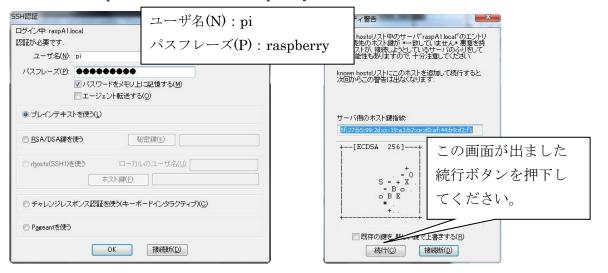
· Tera Term



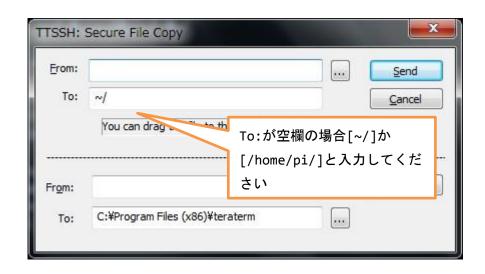
TeraTerm を起動すると接続ダイアログが現れるので、接続する Raspberry Pi のホスト名 +.local 、または、Raspberry Pi の IP アドレスを「ホスト」のテキストボックスに入力し OK を押します。今回の講習会ではホスト名は箱の付箋に記載されています。

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。





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。



From に[01\_LED.py]選択し、Send 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。