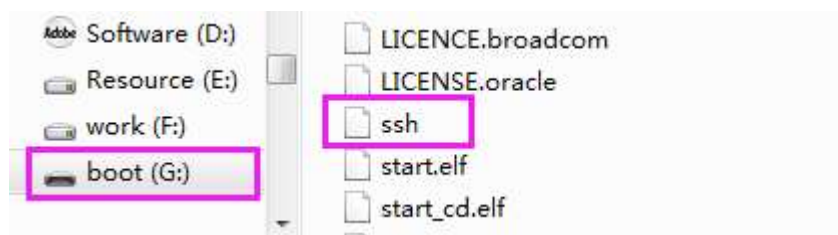


Get Started

Preparation

1. Prepare a MicroSD/TF card of no less than 8GB, a 5V 2A DC power adapter with a MicroUSB port, and a network cable (to connect your router and Raspberry Pi, or plug in the USB Wi-Fi adapter directly if you have one).
2. Download the image for the Raspbian system onto your computer. Refer to instructions through **DOWNLOADS->RASPBIAN** on the official website raspberrypi.org: <https://www.raspberrypi.org/documentation/installation/installing-images/README.md>. Write the image into the microSD/TF card, and then plug the card into the slot on your Raspberry Pi.

Note: For 2016-11-25 release or above, SSH (a protocol securing remote login session and other network service) is Disabled by default. Therefore, when you need to log in remotely, you need to create a file named "ssh" under /boot/ to enable it.



Access to Raspberry Pi's Console

In the subsequent tutorials, the console will be used from time to time. It is platform for interactions in Linux. Therefore, before starting the lessons, you may need to know how to access to Raspberry Pi's console.

Using Console in GUI (**terminal, recommended for beginners**)

Using console in GUI is of great help for the beginners. You can not only compile and run the code in terminal, but also be able to do some simple file operating, code-downloading, etc. cooperating with GUI.

1. Preparations: a screen monitor, an HDMI cable (if your monitor only support VGA, use a VGA-HDMI converter), a USB mouse, a USB keyboard and a network cable or a USB Wi-Fi dongle.
2. Connect the monitor to power. Then connect it with the Raspberry Pi via the converter cable (HDMI cable). Connect the Ethernet cable or the USB Wi-Fi dongle, and the mouse and keyboard to USB ports. At last, connect a 5V 2A DC power to the RPi. Power on the screen if needed. Then you can see the display showing the Raspberry Pi icon as shown below.

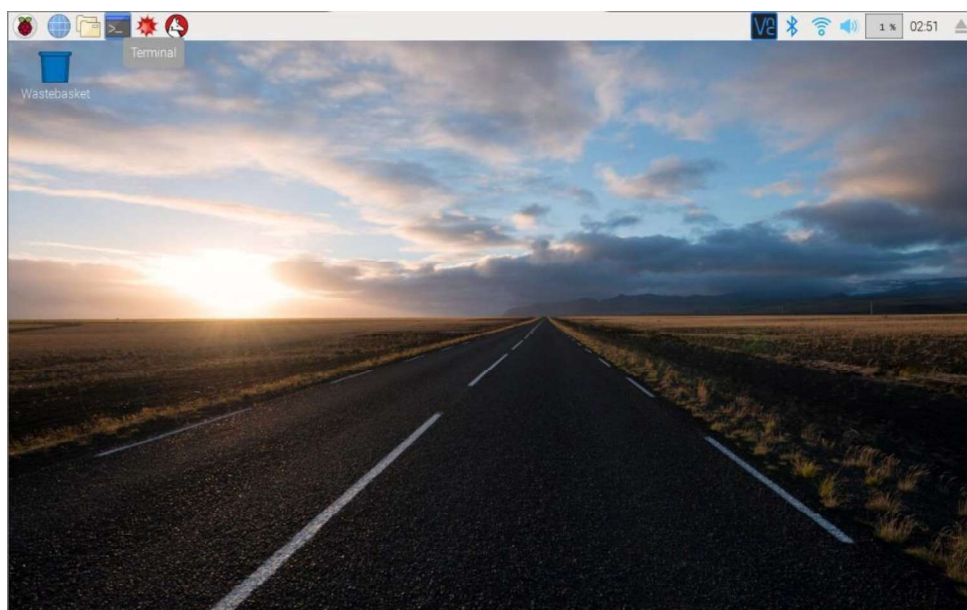
If the monitor displays colored texts with a black background after booting, and that is in console. You can just use this as a terminal (but not recommend for beginners), or change the option for automatically loading a graphic user interface (GUI). To activate GUI, you can type in `startx` with the keyboard and press **Enter**, and to always boot up to GUI, type in `sudo raspi-config` and go through **Boot Options** > **Desktop/Desktop Autologin**, and reboot. Wait for a while and the GUI display will show up as below.



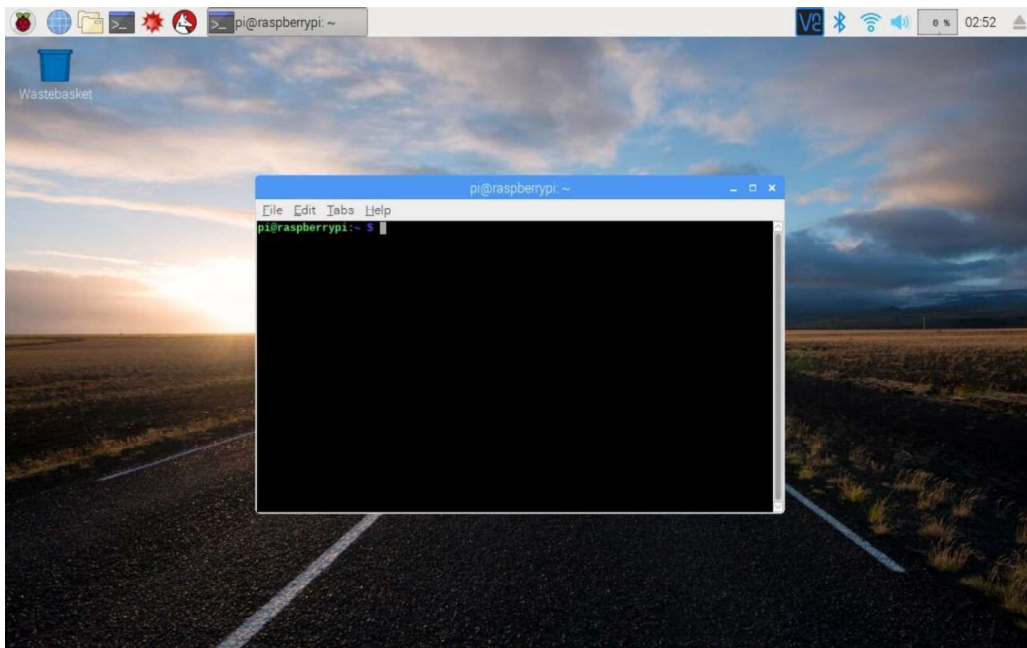
Note:

The screen monitor shown above is the 10.1 inch one we're using, available on our website www.sunfounder.com and our Amazon store. Check out now and use your Raspberry Pi in a most convenient way.

3. Now click the icon of **Terminal** on the screen, or press **CTRL+ALT+T** simultaneously.



4. Then a terminal will pop up as follows:



Here's the console we talk about before.

Using Console Only

There are several ways to use the console only and they can be divided into mainly two ways: using directly and remotely.

A. Directly

A screen is needed when you use the console directly.

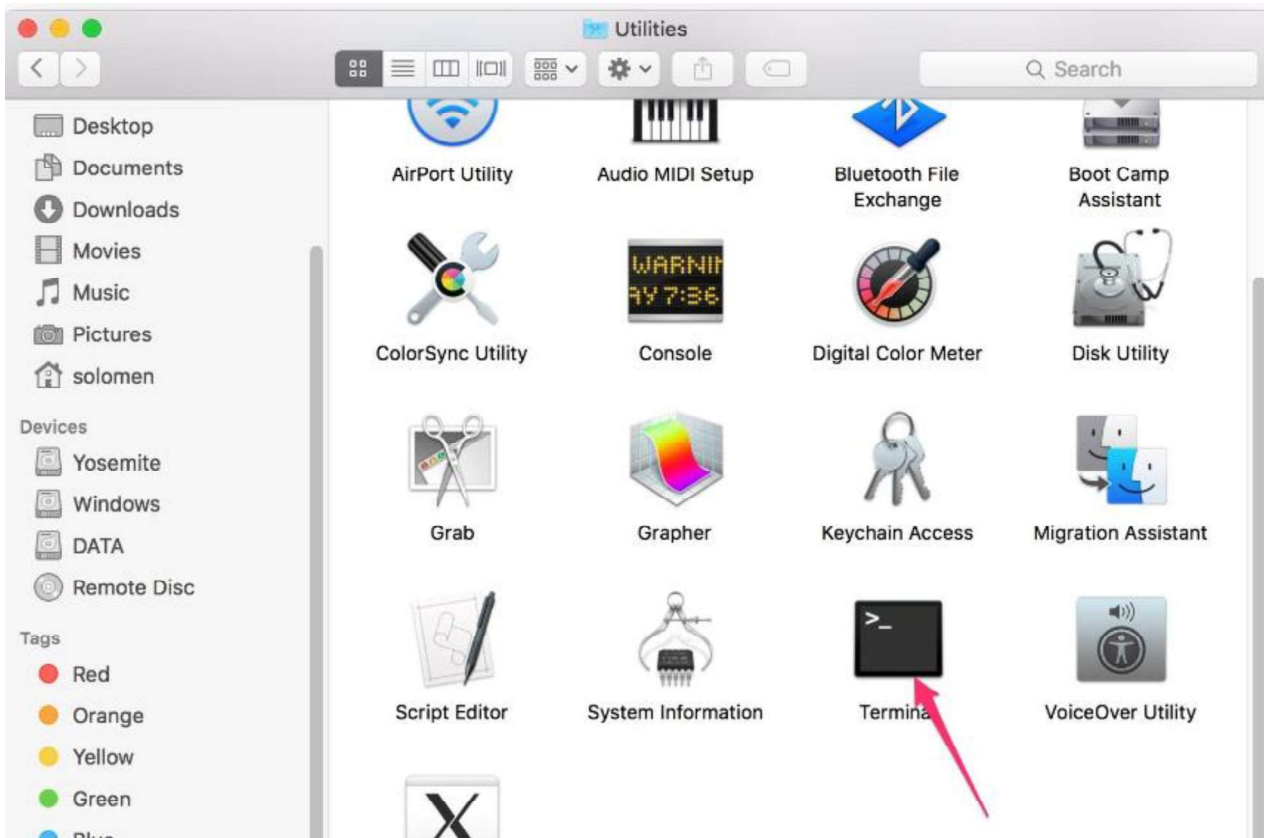
1. Preparations: a screen monitor, an HDMI cable (if your monitor only support VGA, use a VGA-HDMI converter), a USB keyboard and a network cable or a USB Wi-Fi dongle. Yes, you don't need a mouse, or you cannot use mouse.
2. Connect the monitor to power. Then connect it with the Raspberry Pi via the converter cable. Connect the Ethernet cable or the USB Wi-Fi dongle, and keyboard. At last, connect a 5V 2A DC power to the RPi. Power on the screen if needed. Then you can see the console full screen. If you boot into GUI instead, Open a terminal, type in `sudo raspi-config` and press enter, go through **Boot Options** > **Console/Console Autologin**, and reboot.

B. Remotely

For three platforms: **Windows**, **Mac** and **Linux**, it might be a little bit different to do this.

Linux and **Mac** users can easily log into the Raspberry Pi via ssh.

On Linux or Mac, find **Terminal** and open it.



Type in `ssh pi@<ip_address>`

– `ssh` is the tool for remote login; `pi`, the user name, and `<ip_address>` as the name suggests, your RPi's IP address. For example:

```
ssh pi@192.168.0.1
```

Press **Enter** to confirm.

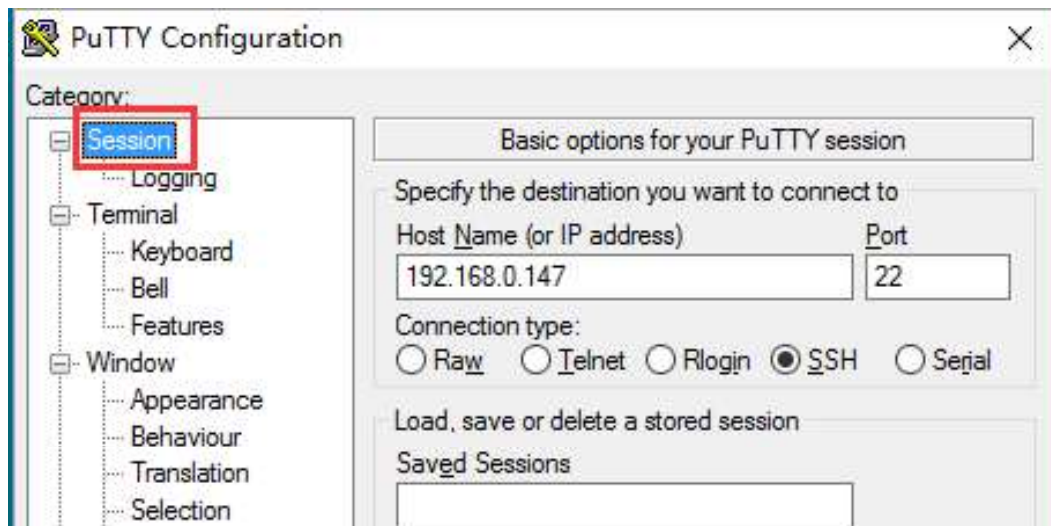
If you get a prompt that no ssh is found, you need to install a ssh tool like Ubuntu and Debian by yourself:

```
sudo apt-get install ssh
```

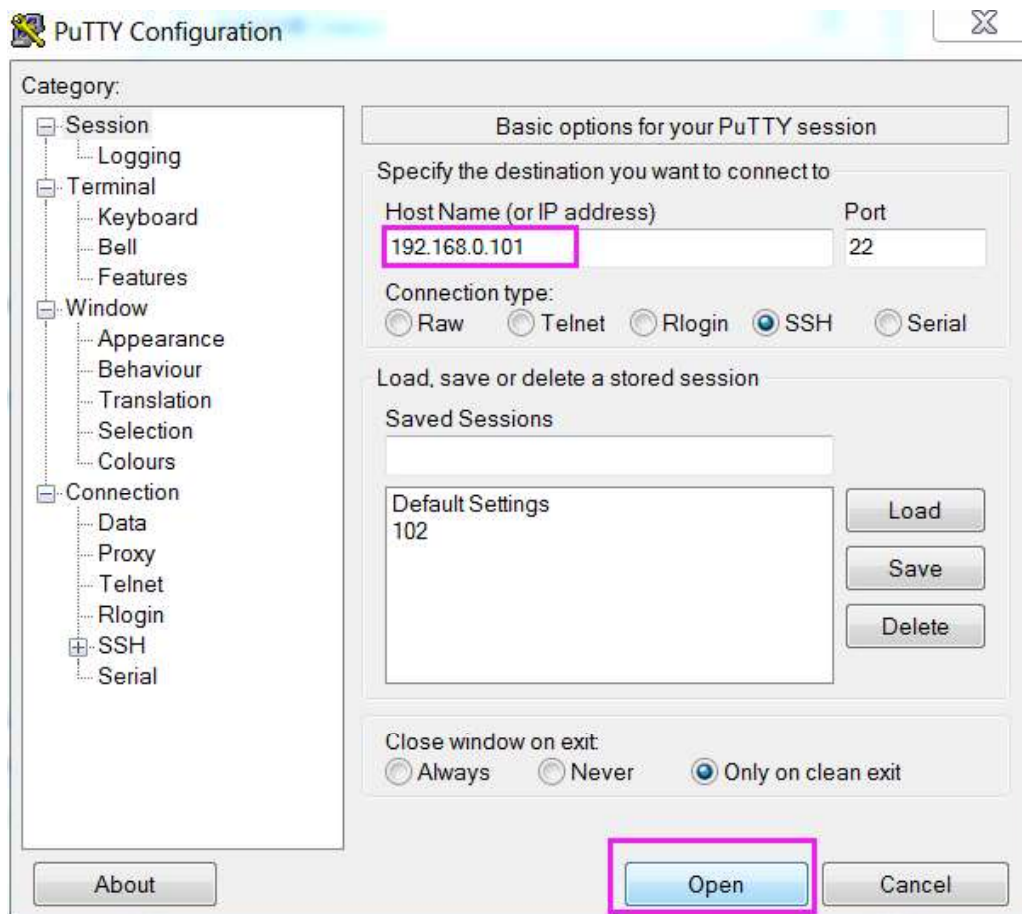
For other platforms, please contact your supplier.

For **Windows users**, you may use a ssh tool to log into Raspberry Pi remotely, like PuTTY.

1. Similarly plug the TF card into the Raspberry Pi (RPi), power the RPi with a 5V 2A DC power and connect the Ethernet cable (Better not a USB Wi-Fi dongle). Now the Raspberry Pi is ready.
2. Then you need to know the IP address of the RPi. You can find it on the settings interface of your router. For more details, please refer to your router provider's page.
3. Open PuTTY and click **Session** on the left tree-alike structure (generally it's collapsed upon PuTTY startup):

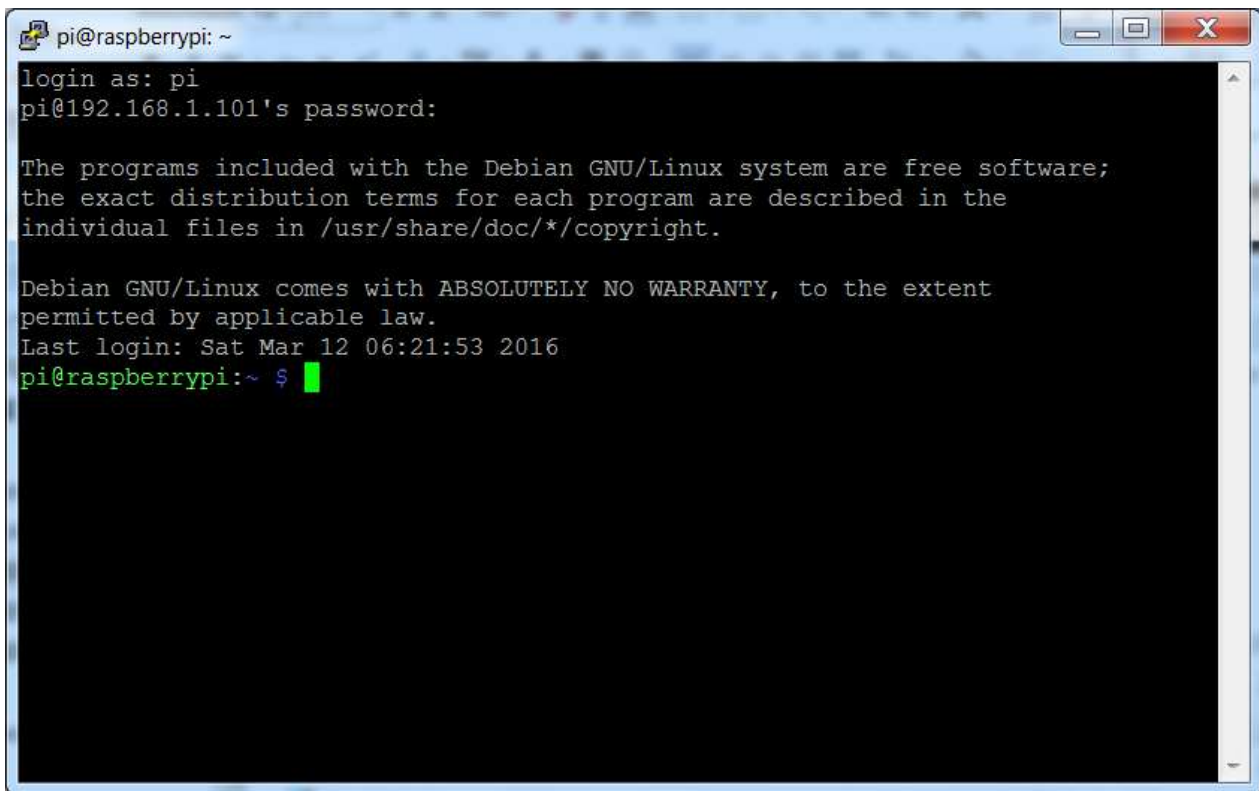


4. Enter the IP address of the RPi you just got in the textbox under Host Name (or IP address) and 22 under Port (by default it is 22)



5. Click **Open**. Note that when you first log in to the Raspberry Pi with the IP address, you'll be prompted with a security reminder. Just click **Yes**. When the PuTTY window prompts login as: type in the user name of the RPi: pi, and password: raspberry (the default one, if you haven't changed it).

Note: when you're typing the password in, the window shows nothing just null, but you're in fact is typing things in. So just focus on typing it right and press **Enter**. After you log in the RPi successfully, the window will display as follows.



```
pi@raspberrypi: ~
login as: pi
pi@192.168.1.101's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Mar 12 06:21:53 2016
pi@raspberrypi:~ $
```

Now, no matter what method you take, you can get in the console of your Raspberry Pi.

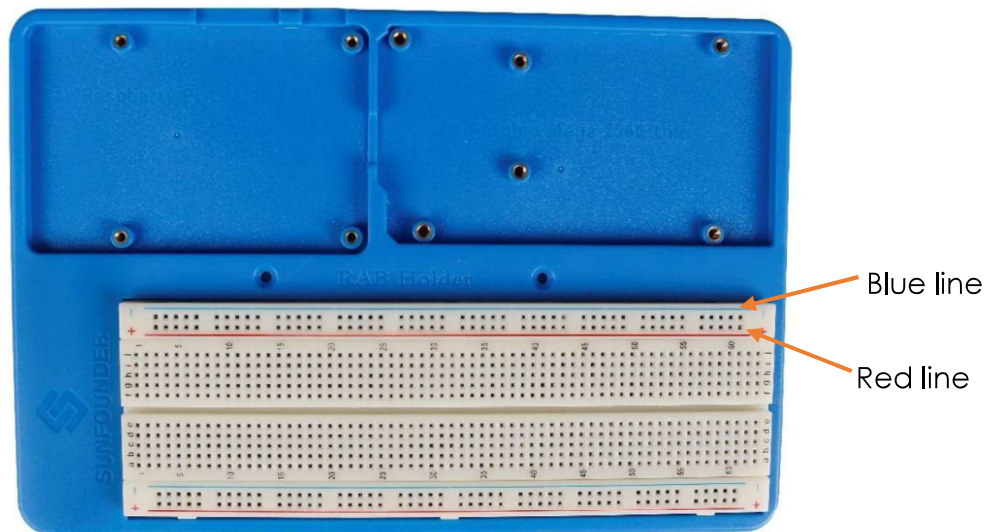
Introduction to Raspberry Pi

RAB Holder

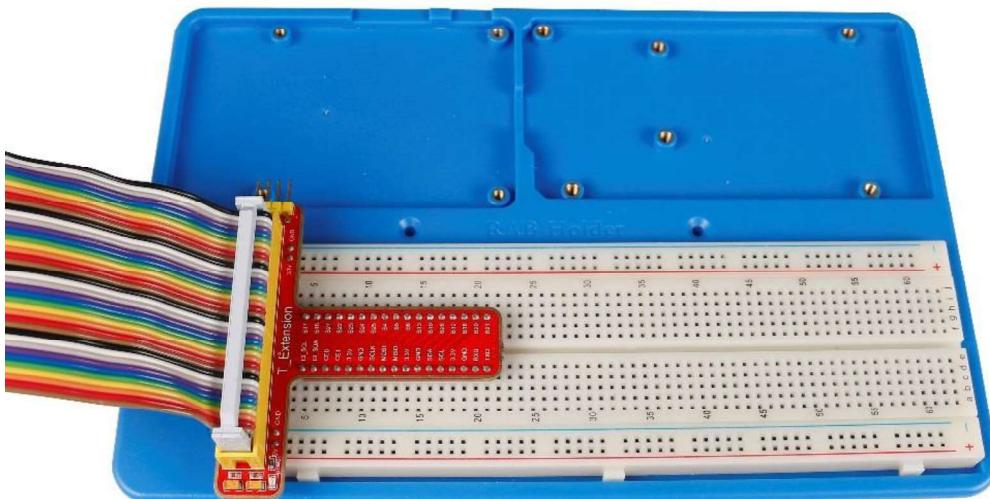
RAB Holder is a basic but indispensable component for your experiment. It makes your experiment easier and can be used for fixing a bread board, an Arduino board like Uno board or Mega2560, or a Raspberry Pi board.

Before starting the experiment, you need to fix the Raspberry Pi and the breadboard on the RAB Holder first.

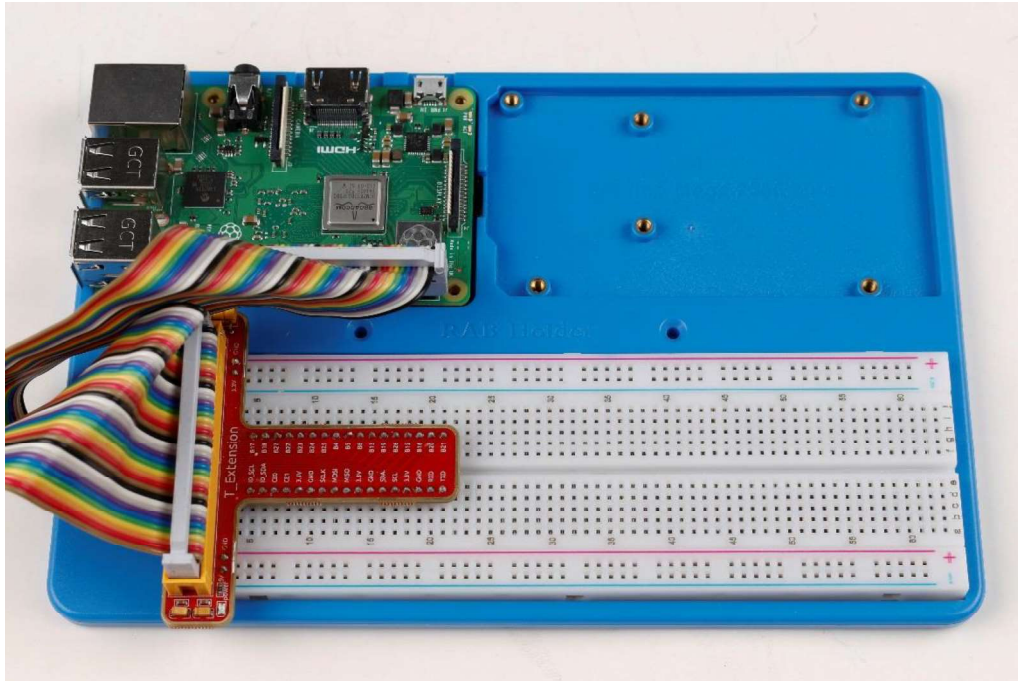
Remove the sticker from the back of the breadboard first, and fix the breadboard on the RAB Holder. Pay attention to place it in such a position as shown in the figure below, so that the 3.3V pin and 5V pin on the T-Extension Board align with the bus strips besides the two red lines when we insert the T-Extension Board later.



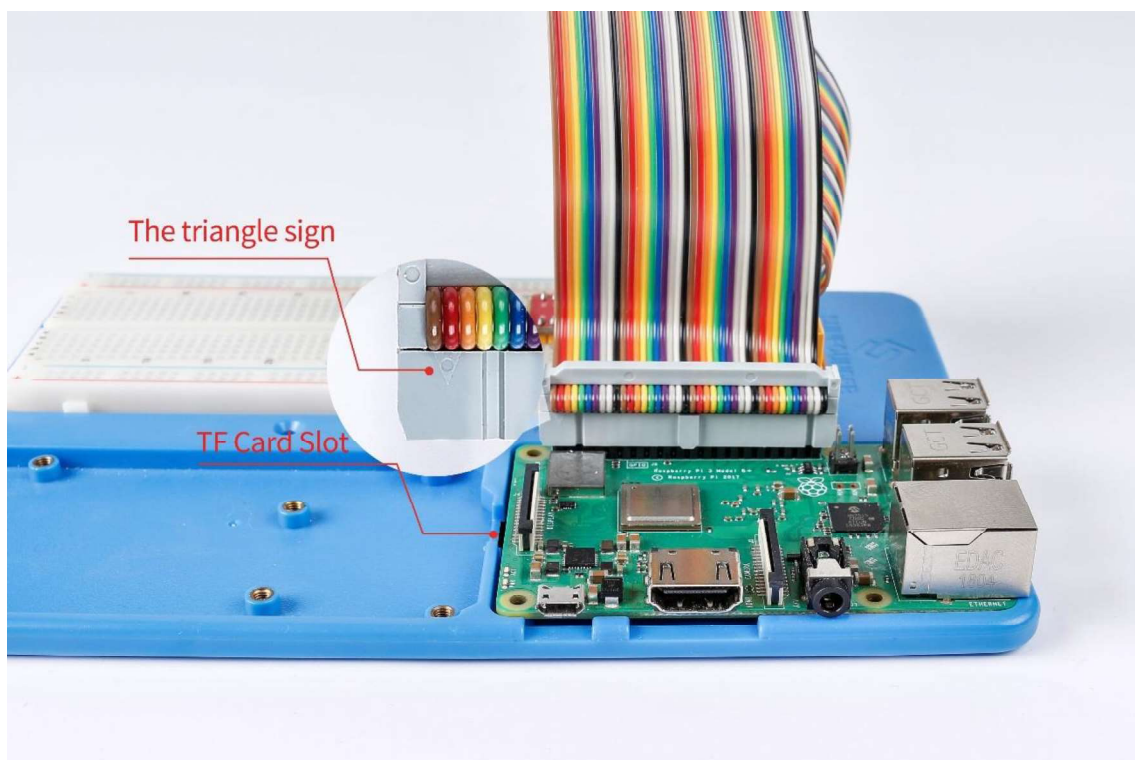
Then insert the T-Extension Board into the Breadboard, and insert the 40-pin GPIO Cable into the board.



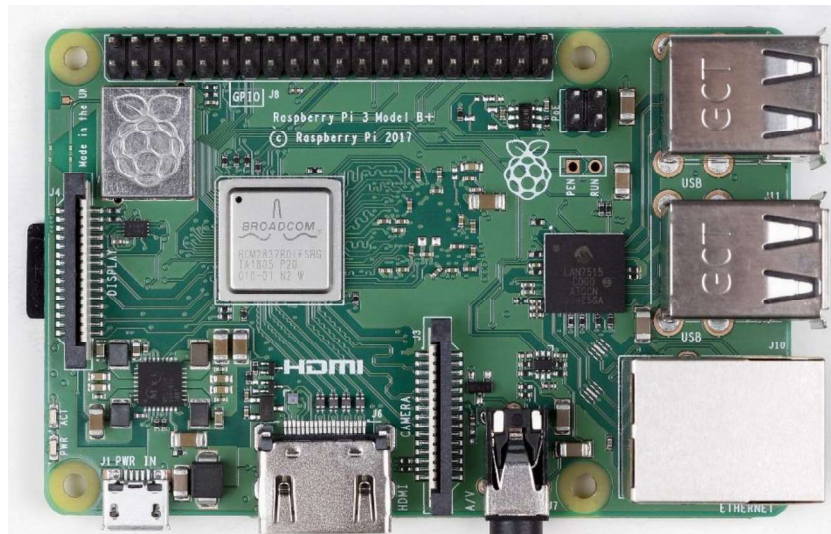
Then place the Raspberry Pi in the holder, fasten it with M2.5x5 screws. Since it may be a little difficult to fasten it, be careful to operate.



Pay attention to the direction when plugging in the 40-pin GPIO Cable into the Raspberry Pi pins. The triangle sign at the edge should be close to the TF card slot. **DO NOT** connect them inversely, or the Raspberry Pi will be short cut!



Raspberry Pi



A Raspberry Pi is an indispensable component for the kit, the control device. But it is not included in the kit and you need to prepare one yourself.

The **Raspberry Pi** is a kind of minicomputer for users like amateurs, teachers, students and small businesses, etc. With a pre-installed Linux system, it is credit card-sized and equipped with an ARM architecture processor, whose operational performance is similar to that of smart phone. As for ports, the Raspberry Pi provides USB ports for mouse and keyboard. In addition, there are also ports for the Fast Ethernet, SD card and HDMI display or TV's connection. Being low cost and low consumption, the Raspberry Pi is very suitable for embedded projects. Many people have been able to apply the Pi to a variety of projects including some simple ones for children and complex ones with more advanced functions. You can apply it like a PC for spreadsheets-making, word-processing and games, or to play HD videos of up to 1080p.

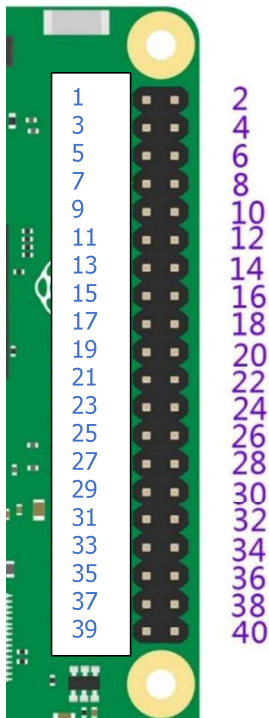
It can't be better describing the Raspberry Pi as "**though small, perfectly formed**". It has the same ability compared with the PC and carries ports for the USB, Ethernet, HDMI, RCA, and 3.5mm stereo jack. Moreover, it can control GPIOs, while the PC cannot. Through this kit, you will learn how to use the GPIOs to make simple experiments and how to program.

The Raspberry Pi evolves through many versions including the latest (so far) Raspberry Pi 3 Model B+ , 3 Model B, 2 model B, 1 Model B+, Zero, and 1 Model A+. Certainly, **the newer, the more powerful**. The 3 Model B+ and 3 model B now even supports Bluetooth and Wi-Fi. You can choose according to actual needs.

Raspberry Pi Pin Name

There are no pins printed on the latest Raspberry Pi, which may bring various troubles to new users. The following is the actual pins definition and form definition of the Raspberry Pi. The actual pins are corresponding to the Physical on the right. There are 40 pins in total from left

to right. The columns near Physical are the V (voltage), Mode (Input or Output), Name (original name), wPi (wiringPi, C language based on), BCM (Python based on) of the pins.



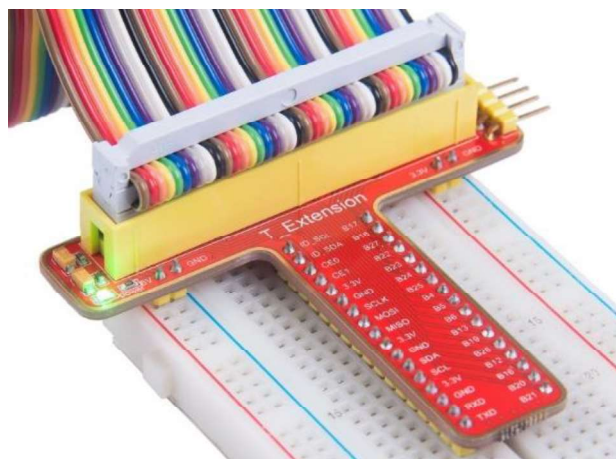
Pi 3											
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	
		3.3v			1	2		5v			
2	8	SDA.1	ALT0	1	3	4		5V			
3	9	SCL.1	ALT0	1	5	6		0v			
4	7	GPIO. 7	IN	0	7	8	1	IN	TxD	15	14
		0v			9	10	1	IN	RxD	16	15
17	0	GPIO. 0	IN	0	11	12	0	IN	GPIO. 1	1	18
27	2	GPIO. 2	IN	0	13	14		0v			
22	3	GPIO. 3	IN	0	15	16	0	IN	GPIO. 4	4	23
		3.3v			17	18	0	IN	GPIO. 5	5	24
10	12	MOSI	ALT0	1	19	20		0v			
9	13	MISO	ALT0	1	21	22	0	IN	GPIO. 6	6	25
11	14	SCLK	ALT0	0	23	24	1	OUT	CE0	10	8
		0v			25	26	1	OUT	CE1	11	7
0	30	SDA.0	IN	1	27	28	1	OUT	SCL.0	31	1
5	21	GPIO. 21	IN	0	29	30		0v			
6	22	GPIO. 22	IN	0	31	32	0	IN	GPIO. 26	26	12
13	23	GPIO. 23	IN	1	33	34		0v			
19	24	GPIO. 24	IN	0	35	36	0	IN	GPIO. 27	27	16
26	25	GPIO. 25	IN	0	37	38	0	IN	GPIO. 28	28	20
		0v			39	40	0	IN	GPIO. 29	29	21
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	

Besides the original name of the pins, there are other three ways of naming including physical, wiringPi and BCM. If you see a pin being defined as 0 in the C language, its original name is GPIO 0. In Python code, it's 17 (BCM) or 11(physical). So you need to know the name, physical, wiringPi and BCM of a pin.

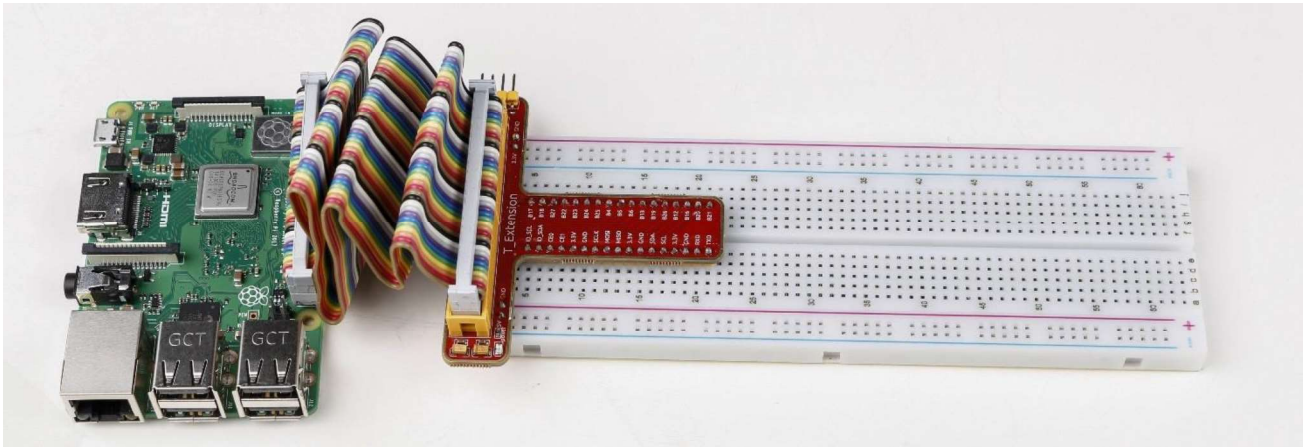
Extension Board

The function of the extension board is to lead out pins of the Raspberry Pi to breadboard by GPIO Extension Board to avoid GPIO damage caused by frequent plugging in or out. For plugging convenience, we designed it in T-shape and name it T-Shape Extension Board.

40-pin GPIO Extension Board



This is our 40-pin GPIO Extension Board and GPIO cable for Raspberry Pi model B+, 2 model B, 3 model B and 3 model B+.

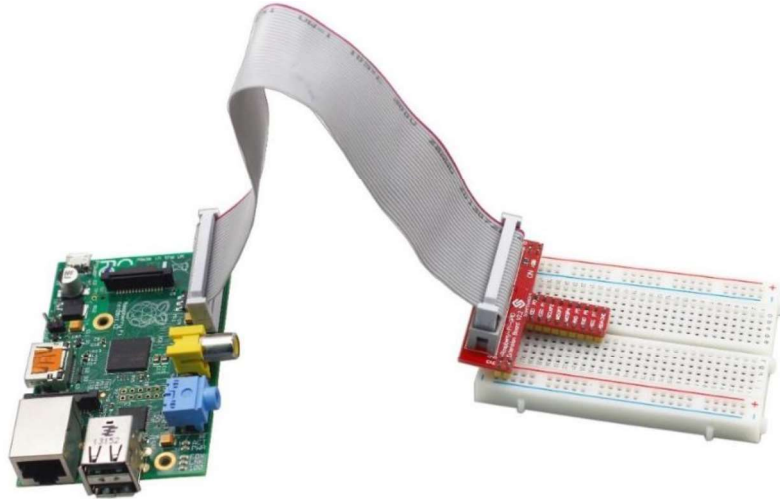


For your better understanding of every pins, we have drawn a table for you to know the Name, BCM and wiring pi of each pin.

Name	wiringPi	BCM	T Extension				
ID_SCL	31	1	ID_SCL	B17	17	0	GPIO0
ID_SDA	30	0	ID_SDA	B18	18	1	GPIO1
CEO	30	8	CEO	B27	27	2	GPIO2
CE1	11	7	CE1	B22	22	3	GPIO3
3.3V	3.3V	3.3V	3.3V	B23	23	4	GPIO4
0V	GND	GND	GND	B24	24	5	GPIO5
SCLK	14	11	SCLK	B25	25	6	GPIO6
MOSI	12	10	MOSI	B4	4	7	GPIO7
MISO	13	9	MISO	B5	5	21	GPIO21
3.3V	3.3V	3.3V	3.3V	B6	6	22	GPIO22
0V	GND	GND	GND	B13	13	23	GPIO23
SDA	8	2	SDA	B19	19	24	GPIO24
SCL	9	3	SCL	B26	26	25	GPIO25
3.3V	3.3V	3.3V	3.3V	B12	12	26	GPIO26
0V	GND	GND	GND	B16	16	27	GPIO27
RXD	16	15	RXD	B20	20	28	GPIO28
TXD	15	14	TXD	B21	21	29	GPIO29

26-pin GPIO Extension Board

26-pin GPIO Extension Board and GPIO cable for Raspberry Pi model B and Raspberry Pi model A.



Also, we have drawn a table of the corresponding BCM, wiringPi and Name of each pins.

Name	wiringPi Pin	BCM GPIO			BCM GPIO	wiringPi Pin	Name
Raspberry-Pi-GPIO Extension Board V2.2							
CE1	11	7	CE1	P0	17	0	GPIO0
CE0	10	8	CE0	P1	18	1	GPIO1
SCLK	14	11	SCLK	P2	21	2	GPIO2
MISO	13	9	MISO	P3	22	3	GPIO3
MOSI	12	10	MOSI	P4	23	4	GPIO4
RXD	16	15	RXD	P5	24	5	GPIO5
TXD	15	14	TXD	P6	25	6	GPIO6
SCL0	9	1	SCL	P7	4	7	GPIO7
SDA0	8	0	SDA	GND	0V	0V	0V

GPIO Libraries

WiringPi

Introduction

WiringPi is a GPIO library for C applied to the Raspberry Pi. It complies with GUN Lv3. The functions in wiringPi are similar to those in the wiring system of Arduino. They enable the users familiar with Arduino to use wiringPi more easily.

Now the Raspbian system has wiringPi pre-installed, you can use it directly.

Test whether wiringPi is installed or not.

WiringPi includes lots of GPIO commands which enable you to control all kinds of interfaces on Raspberry Pi. You can test whether the wiringPi library is installed successfully or not by the following instructions.

gpio -v

```
pi@raspberrypi:~ $ gpio -v
gpio version: 2.32
Copyright (c) 2012-2015 Gordon Henderson
This is free software with ABSOLUTELY NO WARRANTY.
For details type: gpio -warranty
```

If the message above appears, the wiringPi is installed successfully.

Use the command below to see the GPIO layout

gpio readall

```

pi@raspberrypi:~$ gpio readall
-Pi 3-
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi |   Name   | Mode | V | Physical | V | Mode |   Name   | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|  2  |  8  |   3.3v   |      |   |    1    | 2 |      |   5v     |     |     | |
|  3  |  9  |   SDA.1  | ALT0 | 1 |    3    | 4 |      |   5V     |     |     |
|  4  |  7  |   SCL.1  | ALT0 | 1 |    5    | 6 |      |   0v     |     |     |
|  4  |  7  | GPIO. 7  | IN   | 0 |    7    | 8 |  1  | IN   | TxD  | 15  | 14  |
|      |      |   0v     |      |   |    9    |10 |  1  | IN   | RxD  | 16  | 15  |
| 17  |  0  | GPIO. 0  | IN   | 0 |   11   |12 |  0  | IN   | GPIO. 1 | 1  | 18  |
| 27  |  2  | GPIO. 2  | IN   | 0 |   13   |14 |      |   0v     |     |     |
| 22  |  3  | GPIO. 3  | IN   | 0 |   15   |16 |  0  | IN   | GPIO. 4 | 4  | 23  |
|      |      |   3.3v   |      |   |   17   |18 |  0  | IN   | GPIO. 5 | 5  | 24  |
| 10  | 12  |   MOSI   | ALT0 | 1 |   19   |20 |      |   0v     |     |     |
|  9  | 13  |   MISO   | ALT0 | 1 |   21   |22 |  0  | IN   | GPIO. 6 | 6  | 25  |
| 11  | 14  |   SCLK   | ALT0 | 0 |   23   |24 |  1  | OUT  | CE0    |10  |  8  |
|      |      |   0v     |      |   |   25   |26 |  1  | OUT  | CE1    |11  |  7  |
|  0  | 30  |   SDA.0  | IN   | 1 |   27   |28 |  1  | OUT  | SCL.0  |31  |  1  |
|  5  | 21  | GPIO.21  | IN   | 0 |   29   |30 |      |   0v     |     |     |
|  6  | 22  | GPIO.22  | IN   | 0 |   31   |32 |  0  | IN   | GPIO.26 |26  | 12  |
| 13  | 23  | GPIO.23  | IN   | 1 |   33   |34 |      |   0v     |     |     |
| 19  | 24  | GPIO.24  | IN   | 0 |   35   |36 |  0  | IN   | GPIO.27 |27  | 16  |
| 26  | 25  | GPIO.25  | IN   | 0 |   37   |38 |  0  | IN   | GPIO.28 |28  | 20  |
|      |      |   0v     |      |   |   39   |40 |  0  | IN   | GPIO.29 |29  | 21  |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi |   Name   | Mode | V | Physical | V | Mode |   Name   | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

RPi.GPIO

If you are a Python user, you can program GPIOs with API provided by RPi.GPIO and use BCM numbering method to control the GPIOs of Raspberry Pi. Please note that it differs from the way that using wiringPi numbering method to control the GPIO on a Raspberry Pi in C language.

Introduction

RPi.GPIO is a module to control Raspberry Pi GPIO channels. This package provides a class to control the GPIO on a Raspberry Pi. For examples and documents, visit <http://sourceforge.net/p/raspberry-gpio-python/wiki/Home/>.

Now the Raspbian Jessie 2016-05-27 has RPi.GPIO pre-installed, you can use it directly, too.

Test whether RPi.GPIO is installed or not:

Type in `python` to python CLI:

```
pi@raspberrypi:~ $ python
Python 2.7.9 (default, Mar  8 2015, 00:52:26)
[GCC 4.9.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

In Python CLI, Type in:

```
import RPi.GPIOimport RPi.GPIO
```

If no error prompts, it means RPi.GPIO is installed.

```
pi@raspberrypi:~ $ python
Python 2.7.9 (default, Mar  8 2015, 00:52:26)
[GCC 4.9.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import RPi.GPIO
>>> 
```

Then, type in

```
RPi.GPIO.VERSION
```

```
>>> RPi.GPIO.VERSION
'0.6.2'
>>> 
```

If it shows its version like above, your Pi is ready to go!

So in this kit, please note that the example code is ONLY test on Raspbian.

Download the Code

We provide two methods for download:

Method 1: Use git clone (Recommended)

Log into Raspberry Pi's console, just as previously shown.

Change directory to /home/pi

```
cd /home/pi/
```

Note: `cd` to change to the intended directory from the current path. Informally, here is to go to the path /home/pi/.

Clone the repository from GitHub

```
git clone https://github.com/sunfounder/SunFounder_Super_Kit_V3.0_for_Raspberry_Pi.git
```

The advantage of this method is that, you can update the latest code any time you want, using `git pull` under the folder.

Method 2: Download the code.

Download the source code from github:
https://github.com/sunfounder/SunFounder_Super_Kit_V3.0_for_Raspberry_Pi

SunFounder Supet Kit V3.0 for Raspberry Pi

21 commits 1 branch 0 releases 1 contributor GPL-2.0

Branch: master New pull request Find file Clone or download

sunfounder modify the code

C	modify the code
Python	Update 04_breathLed.py
.gitignore	update
LICENSE	create repository
README.md	create repository
show	create repository

Clone with HTTPS ⓘ
Use Git or checkout with SVN using the web URL.
https://github.com/sunfounder/SunFounder_S
Open in Desktop Download ZIP