

树莓派

Raspberry Pi

现在启动系统，和树莓派的操作系统见个面，  
熟悉一下常用目录结构和常用命令

- **Linux/Raspbian 目录管理**
- 最常用的树莓派 **Linux** 命令
- 常用的编辑工具
- **Linux/Raspbian 内核的编译**

# Linux/Raspbian 目录管理

Linux 的目录有非常明确的规则

- 在Windows中，典型的路径可能是这样的：

D:\Folder\subfolder\file.txt

- 而在Linux中，路径则是这样的

/Folder/subfolder/file.txt

- 而且Linux系统对文件或者文路径的名称中的大小写字符是敏感的。

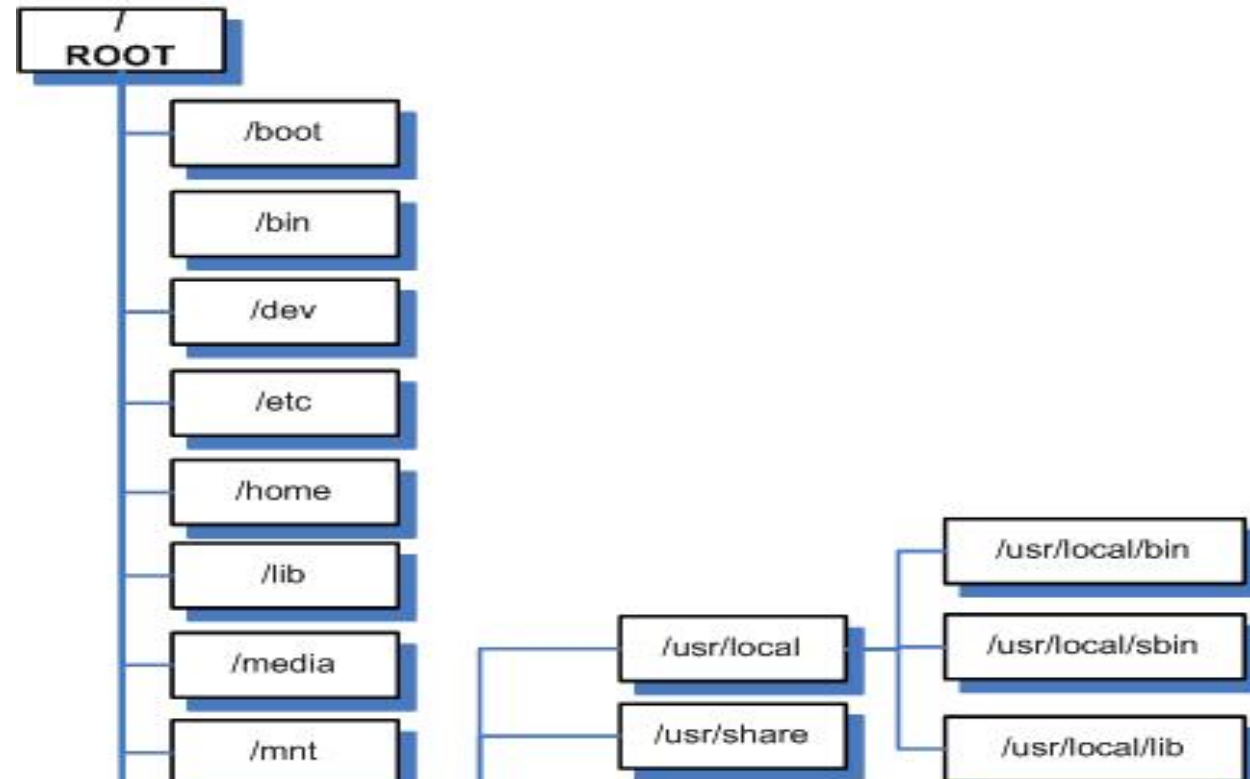
- 比如

/Folder/subfolder/file.txt 与 /folder/subfolder/file.txt

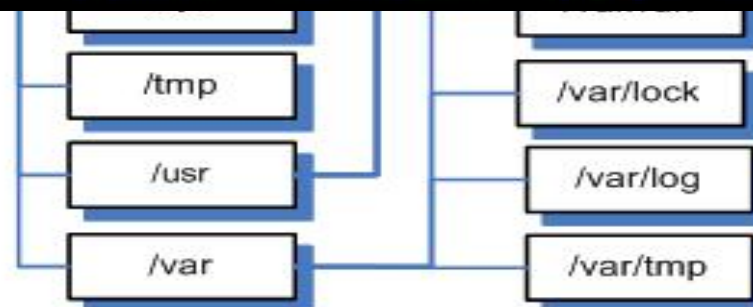
并不是同一个文件。

- 所有的文件、文件夹、设备以及不同的硬盘光驱之类的，也都挂载在了 /

# LINUX系统目录结构图



```
wax@wax:~ $ cd /
wax@wax:/ $ ls
bin    dev    home   lost+found  mnt    proc    run    srv    tmp    var
boot   etc    lib    media      opt    proc    sbin   sys    usr
wax@wax:/ $
```



- `/`: 根目录，所有的目录、文件、设备都在`/`之下，`/`就是Linux文件系统的组织者，也是最上级的领导者。
- `/bin`: **系统相关程序**，bin 就是二进制（binary）英文缩写。包含用户模式下的二进制文件以及linux常用的命令工具，比如cat，ls，cp这些命令。

bash	cat	chmod	cp	date	echo
expr	kill	ln	ls	mkdir	mv
pwd	rm	rmdir	sleep	test	unlink



- **/boot:** Linux的内核及引导系统程序所需要的文件目录，比如 `vmlinuz` `initrd.img` 文件都位于这个目录中。在一般情况下，**GRUB**或**LILO**系统引导管理器也位于这个目录。

- **/root:** Linux是多用户系统，每个用户会有一个用户目录，位于不同的路径下。**/root**是**root**的用户目录。该目录文件的拥有者和拥有组都是**root**。
- **/home:** **用户登录子目录的位置和服务器数据子目录**，每一个用户在这个目录下，都会单独有一个以其用户名命名的目录，在这里保存着用户的个人设置文件，尤其是以 **profile** 结尾的文件。但是也有例外，**root**用户的数据就不在这个目录中，而是单独在根路径下，保存在单独的**/root**文件夹下。
- 如果建立一个用户，用户名是“**xx**”，那么在**/home**目录下就有一个对应的**/home/xx**路径，**/home**是用来存放用户的主目录。

- **/dev:** 内含必需的系统文件和驱动器。**dev** 是设备 (device) 的英文缩写。这个目录对所有的用户都十分重要。因为在这个目录中包含了所有linux系统中使用的外部设备。但是这里并不是放的外部设备的驱动程序。这一点和常用的windows,dos操作系统不一样。它实际上是一个访问这些外部设备的端口。可以非常方便地去访问这些外部设备，和访问一个文件，一个目录没有任何区别。
  - 每个设备文件对应一个设备，比如存储器和UART接口。通过这些设备文件，设备还可以是没有硬件实物的虚拟设备，比如中断。
  - Linux的设备有主编好和副编号。主标号说明了设备的类型，副标号就是后面跟的“0”，即该类型下编号为0的设备。

- **/etc: 系统和网络服务及应用程序的配置文件**，在这个目录下存放了系统管理时要用到的各种配置文件和子目录。要用到的网络配置文件，文件系统，系统配置文件，设备配置信息，设置用户信息等都在这个目录下。
  - 保存着关键的操作系统配置文件，这些配置文件可以改变操作系统解绑的行为。
  - 操作系统启动时的init进程及init调用的脚步也在/etc下。
  - 保存着用户和用户组的相关信息。
  - 增加和删除用户的操作，时间上就是修改这些文件。
  - 应用程序配置文件。

例如：，比如 /etc/hosts, /etc/resolv.conf, nsswitch.conf 下都是配置文件

- /sbin: **特殊操作的系统程序**，这个目录是用来存放系统管理员的系统管理程序。大多是涉及系统管理的命令的存放，是超级权限用户root的可执行命令存放地，普通用户无权限执行这个目录下的命令，这个目录和/usr/sbin; /usr/X11R6/sbin或/usr/local/sbin目录是相似的，凡是目录sbin中包含的都是root权限才能执行的。
- 例:ifconfig

fdisk	hwclock	ifconfig	reboot	shutdown
-------	---------	----------	--------	----------

- **/lib:** **系统库文件**，lib是库（library）英文缩写。这个目录是用来存放系统动态连接共享库的。几乎所有的应用程序都会用到这个目录下的共享库。因此，千万不要轻易对这个目录进行什么操作，一旦发生问题，系统就不能工作了。
- **/lost+found:** 在ext2或ext3文件系统中，当系统意外崩溃或机器意外关机，而产生一些文件碎片放在这里。当系统启动的过程中fsck工具会检查这里，并修复已经损坏的文件系统。有时系统发生问题，有很多的文件被移到这个目录中，可能会用手工的方式来修复，或移到文件到原来的位置上。

- **/mnt**: 这个目录一般是用于存放挂载储存设备的挂载目录的，比如网络硬盘、光驱和额外的硬盘。存储通常需要手动挂载或者在挂载文件里增加对应的条目。
- **/media**: 有些linux的发行版使用这个目录来挂载那些usb接口的移动硬盘（包括U盘）、CD/DVD驱动器等等。
- **/opt**: 这里主要存放那些可选的程序。
- **/proc**: 可以在这个目录下获取系统信息。这些信息是在内存中，由系统自己产生的。
  - 内核直接管理的硬件信息可以在**/proc**下查询。 **/proc**是一个伪文件系统，直接对应了内存上的内核空间。通过**/proc**，内核给用户提供了一个查询内核信息的简易窗口。 **/proc/cupinfo**中保存着CPU信息， **/proc/meminfo**中保存着内存使用信息。因为内核直接管理的设备对于计算机运行至关重要，所以**/proc**下的文件大多是只读的，不允许用户直接进行写入操作。内核还保存进程的信息，这些原版在内核空间的信息也以文件的形式呈现在**/proc**目录下。

- `/srv` 服务启动后，所需访问的数据目录，举个例子来说，`www`服务启动读取的网页数据就可以放在`/srv/www`中
- `/tmp`: 临时文件目录，用来存放不同程序执行时产生的临时文件。有时用户运行程序的时候，会产生临时文件。`/tmp`就用来存放临时文件的。`/var/tmp`目录和这个目录相似。`Raspbian`会在开机后清空`/temp`文件夹。
- `/usr` 这是linux系统中占用硬盘空间最大的目录。用户的很多应用程序和文件都存放在这个目录下。在这个目录下，可以找到那些不适合放在`/bin`或`/etc`目录下的额外的工具
- `/usr/local`: 这里主要存放那些手动安装的软件，即不是通过“新立得”或`apt-get`安装的软件。它和`/usr`目录具有相类似的目录结构。让软件包管理器来管理`/usr`目录，而把自定义的脚本（`scripts`)放到`/usr/local`目录下面。



- `/usr/share` : 系统共用的东西存放地, 比如 `/usr/share/fonts` 是字体目录, `/usr/share/doc`和`/usr/share/man`帮助文件。
- `/var`: 这个目录的内容是经常变动的, 看名字就知道, 可以理解为vary的缩写, `/var`下有`/var/log` 这是用来存放系统日志的目录。
  - `/var/www`目录是定义Apache服务器站点存放目录; `/var/lib` 用来存放一些库文件, 比如MySQL的, 以及MySQL数据库的的存放地。
  - 邮件程序的可执行文件是一个大小不变的静态文件, 单电子邮件的相关文字和图片会随着用户使用快速增长。因此电子邮件归档保存在`/var`下。
  - 缓存数据战绩的空间经常浮动变化, 因此也保存在`/var`下。

**/usr** 与用户有关的各种操作的程序文件，用户数据目录，包含了属于用户的实用程序和应用程序。这里有很多重要的，但并非关键的文件系统挂载这个路径下面。在这里，你会重新找到一个 bin、sbin 和 lib 目录，其中包含非关键用户和系统二进制文件以及相关的库和共享目录，以及一些库文件。

- **/usr/sbin** 包含系统中非必备和并不是特别重要的系统二进制文件以及网络应用工具。
- **/usr/bin** 包含用户的非必备和并不是特别重要的二进制文件。
- **/usr/lib** 保存着 **/usr/sbin** 以及 **/usr/bin** 中二进制文件所需要的库文件。
- **/usr/share** “平台无关”的共享数据目录。
- **/usr/local** 是 **/usr** 下的二级目录，这里主要保存着包含系统二进制文件以及运行库在内的本地系统数据。
- **/proc** 这个目录是驻留在系统内存中的虚拟（psuedo，伪）文件系统，其中保存的都是文本格式的系统内核和进程信息。

# 最常用的树莓派 Linux 命令

- Linux系统可以在两种环境下操作：
- 一种是在X Window的桌面环境下操作，它与Microsoft Windows环境下的操作十分相似；
- 另一种是在控制台的字符屏幕下操作，它通过键盘输入命令来实现对系统的操作。

- 命令在哪里输入？

进入树莓派的桌面运行终端程序，或者通过 **SSH** 客户端登录你的树莓派，接下来就可以输入命令了。

# Linux常用命令

- shell是一种命令行解释程序（Command-Language Interpreter），负责用户和操作系统的沟通。
- 在终端方式的shell 提示窗口如图所示。

数据包：已发送 = 4，已接收 = 4，  
往返行程的估计时间(以毫秒为单位)：  
最短 = 1ms，最长 = 2ms，平均 = 1ms

```
Processing triggers for initramfs-tools (0.140ubuntu2)
Processing triggers for libpam-modules (1.3.0-5ubuntu1)
Processing triggers for libpam-modules:amd64 (1.3.0-5ubuntu1)
Processing triggers for libc-bin (2.25-1ubuntu2)
wax@wax:~$ cd /
wax@wax:/ $ ls
bin      dev      home     lost+found
boot     etc      lib      media
wax@wax:/ $
```

C:\Users\Administrator>

图 shell提示窗口  
左： windows 右： linux

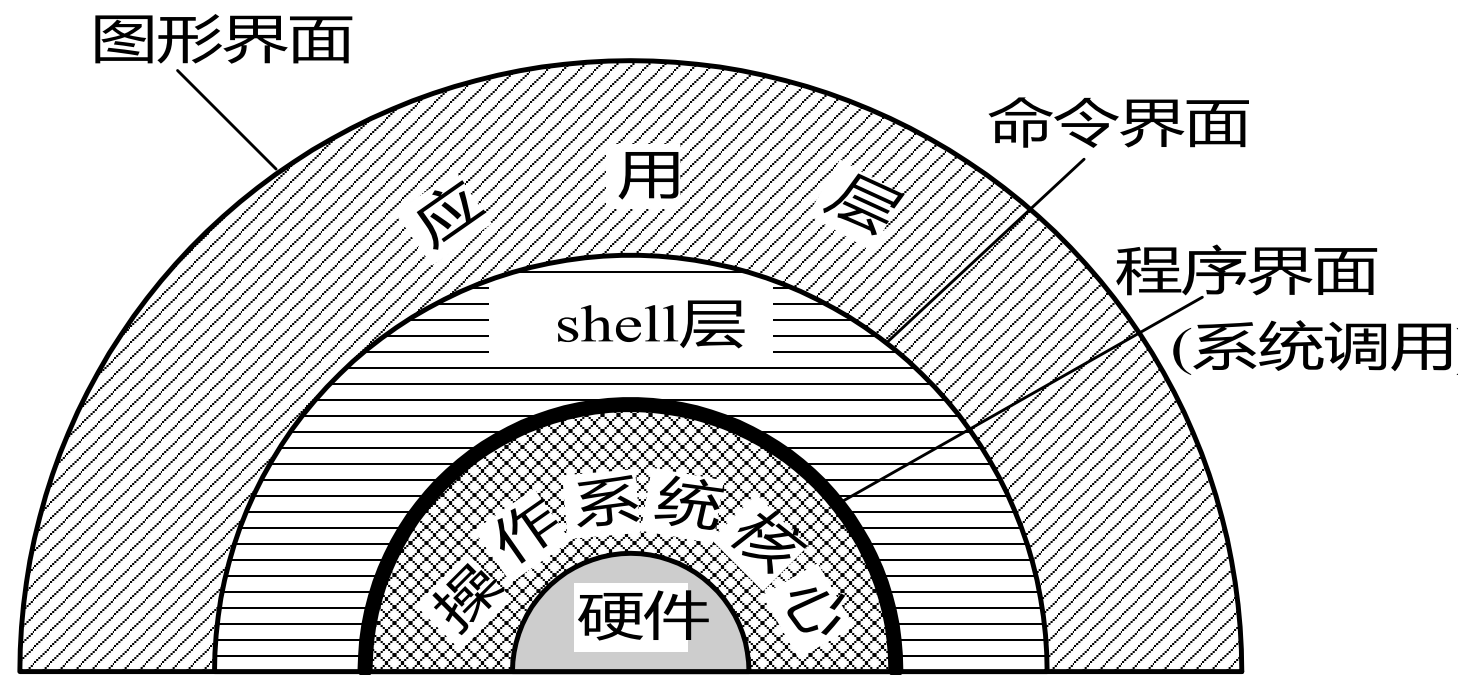


图 操作系统的三种界面



- `pi@raspberrypi:~$`
- `pi`是用户名，`raspberrypi`是计算机的名字，`$`是命令提示符。
- `Shell`：运行在终端中的文本互动程序。
- `Shell`分析文本输入，把文本转换成相应的计算机动作。

什么是 sudo 命令？

在一行命令之前加上 sudo，表示以系统管理员身份执行这条命令。如果使用不当可能会造成事故，所以仅在特别需要使用管理员权限运行的时候添加。

- `sudo apt-get update` :更新软件列表。
  - `sudo apt-get upgrade` #升级软件包。需要用update更新完才能upgrade
- `sudo apt-get install XXXX` 网络安装指定数据包

`sudo reboot`

重启树莓派，需要管理员权限才可以执行，因此添加 sudo。

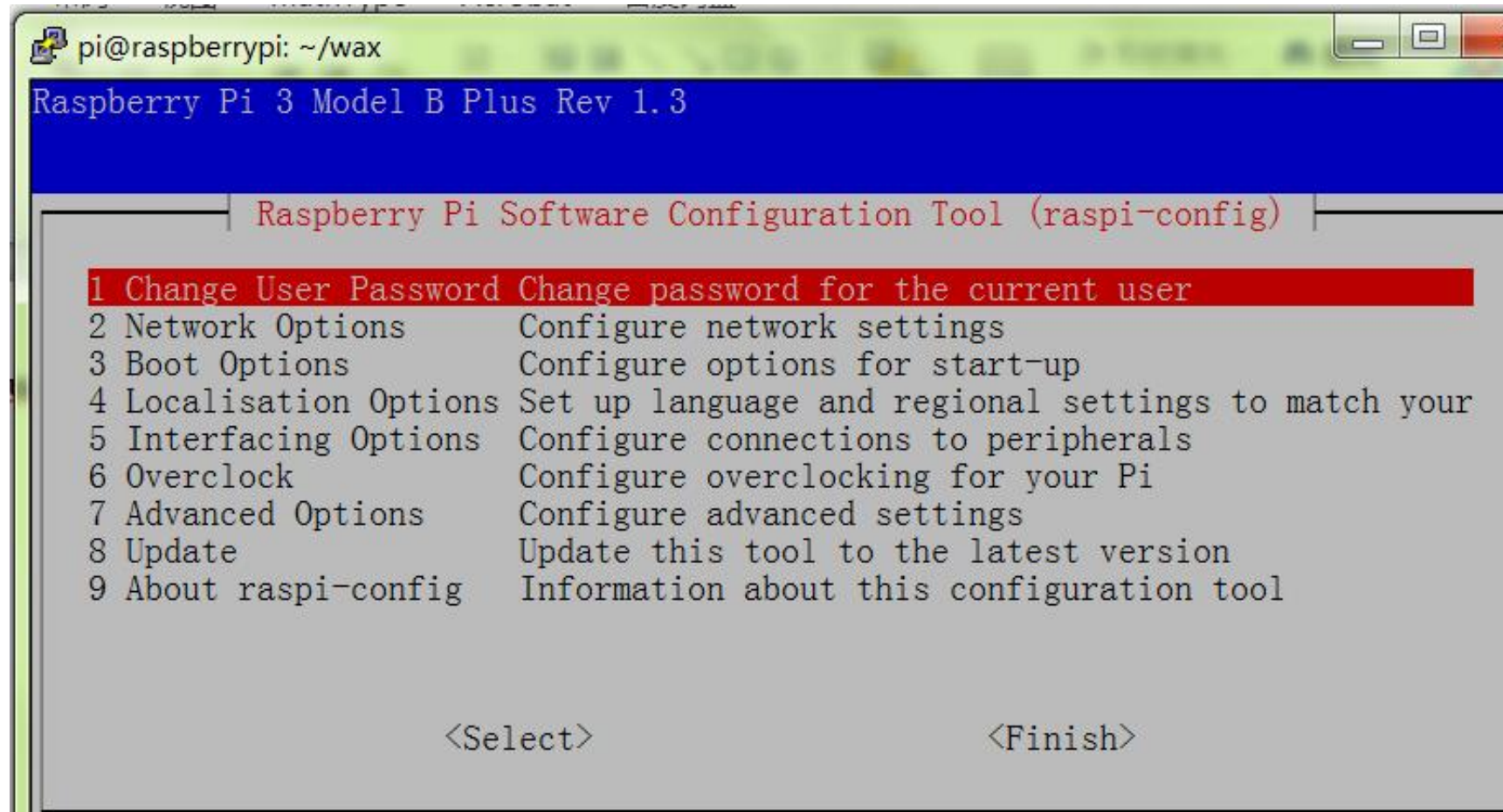
关机，需要管理员权限才可以执行。

`sudo raspi-config` :打开树莓派功能配置界面。

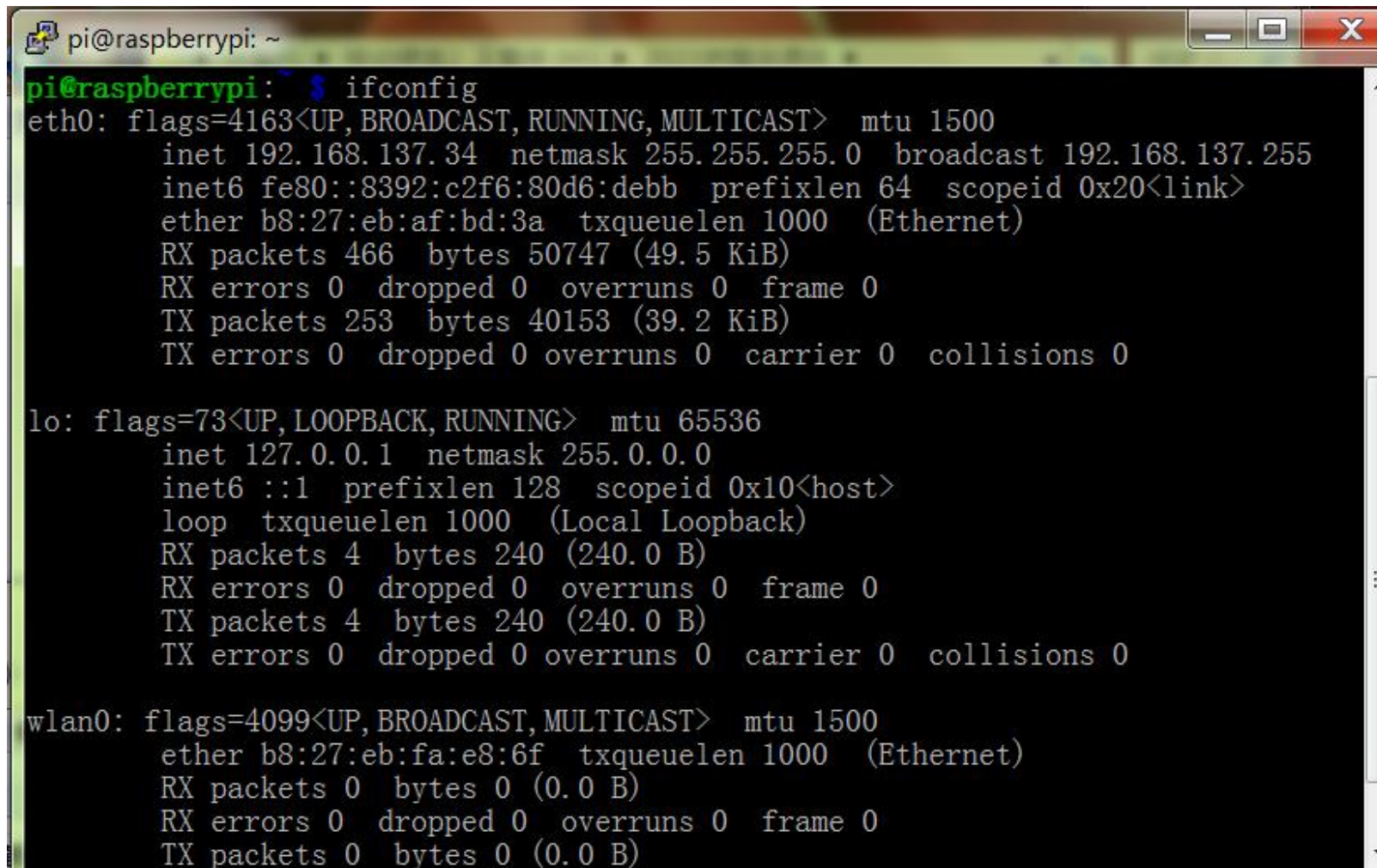
`ifconfig`:列出树莓派的网络配置信息，英文全称是network interfaces configuring。

你来试一下

sudo raspi-config : [树莓派配置工具raspi-config](#)



ifconfig:列出树莓派的网络配置信息，英文全称是network interfaces configuring。

A screenshot of a terminal window titled 'pi@raspberrypi: ~'. The terminal shows the command 'ifconfig' being executed. The output displays the configuration for three network interfaces: eth0 (Ethernet), lo (Loopback), and wlan0 (Wireless LAN). Each interface's configuration includes its flags, MTU, IP address, netmask, broadcast address, and various statistics like RX/TX packets and errors.

```
pi@raspberrypi: ~  
pi@raspberrypi: $ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.137.34 netmask 255.255.255.0 broadcast 192.168.137.255  
    inet6 fe80::8392:c2f6:80d6:debb prefixlen 64 scopeid 0x20<link>  
    ether b8:27:eb:af:bd:3a txqueuelen 1000 (Ethernet)  
    RX packets 466 bytes 50747 (49.5 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 253 bytes 40153 (39.2 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 4 bytes 240 (240.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 4 bytes 240 (240.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500  
    ether b8:27:eb:fa:e8:6f txqueuelen 1000 (Ethernet)  
    RX packets 0 bytes 0 (0.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 0 bytes 0 (0.0 B)
```

**clear** :清除终端上的文字。

**cd /folder1/folder2** :进入到目录 */folder1/folder2*。

**uname -a** :查询系统版本：两种树莓派系统架构：armv7l、arch64 //

**lsb\_release -a**

stretch、buster、bullseye 国内源地址,选择其中一个适合当前系统的源

**ls** :列出当前位置的文件和目录，显示全部信息。

**python -v** :查看树莓派上安装的Python版本,也可以：

*python* 命令行查看opencv版本：

*python*

*import cv2*

*print(cv2.\_\_version\_\_)*

```
1 pi@raspberrypi:~ $ python3
2 Python 3.9.2 (default, Mar 12 2021, 04:06:34)
3 [GCC 10.2.1 20210110] on linux
4 Type "help", "copyright", "credits" or "license" for more information.
```

```
1 >>> print(cv2.__version__)
2 4.5.1
```

**sudo find** -name file.txt

查找文件名为 *file.txt* 的文件。

**sudo find** / -name file.\*

查找文件名为 *file.\** 的文件，\*号为通配符。

**sudo mv** ~/file /folder1/folder2/

将 ~/file 移动到 /folder1/folder2/。

**man command**

查看命令的使用手册。*command* 替换成你想了解的任何命令。

**ping** 192.168.1.1

*Ping* 某个 *IP*。查看树莓派和这个 *IP* 的设备的连接状况

# 基本命令

pwd: 显示当前目录路径。

CP: 拷贝文件 ; 用法: CP 源文件 目标文件

mv: 移动文件或目录 mv [-fi] source\_file target\_file

rm: 删除文件或目录 rm [-firR]file...

cd 目录名: 进入目录

mkdir 目录名: 创建目录

rmdir 目录名: 删除空目录

uname -a: 查看内核的版本

echo: 写内容到标准输出。

chmod +x: 设置谁拥有执行这个文件的权限



# 文件的编辑工具-Vi

也可以 vim **nano** **gedit** (sudo apt install gedit)

# 文件的编辑——Vi

- vi，是一款由加州大学伯克利分校，Bill Joy研究开发的文本编辑器. 是所有Unix及Linux系统下标准的编辑器，它的强大不逊色于任何最新的文本编辑器, 对Unix及Linux系统的任何版本，vi编辑器是完全相同的. Vi也是Linux中最基本的文本编辑器，学会它后，您将在Linux的世界里畅行无阻。

# 文件的编辑——Vi

1. 启动: `vi [filename]`

2. 退出:

`:w` 保存;

`:w filename` 另存为filename;

`:wq` 保存退出;

`:q!` 不保存退出;

# 文件的编辑——Vi

## 3. vi 的三种模式:

Command（命令）模式，用于输入命令:控制屏幕光标的移动，字符、字或行的删除。

输入i进入插入模式;

Insert（插入）模式，用于插入文本:可以做文字输入，删除等操作。按「ESC」键可回到命令行模式;

最后行模式：退出和保存退出。输入“:”进入最后行模式

# 文件的编辑——Vi

- 1) **vi [filename]** 进入vi之后，是处于「命令行模式（command mode）」
- 2) 在「命令行模式（command mode）」下按一下字母「i」就可以进入「插入模式（Insert mode）」，这时候就可以开始输入文字了
- 3) 处于「插入模式（Insert mode）」就只能一直输入文字，如果发现输错了字！想用光标键往回移动，将该字删除，就要先按一下「ESC」键转到「命令行模式（command mode）」再删除文字。

## 5) 删除文字

「x」：每按一次，删除光标所在位置的"后面"一个字符。

「X」：大写的X，每按一次，删除光标所在位置的"前面"一个字符。

「dd」：删除光标所在行。



# 文件的编辑——Vi

例如建立一个最简单的例子——example6:

启动Vi : Vi example6.c

输入文本:

```
#include <stdio.h>
```

```
int main(void)
```

```
printf(" Good Luck to you ! \n");
```

按ESC转化为命令模式。输入:

: wq退出。

则example6.c建立。

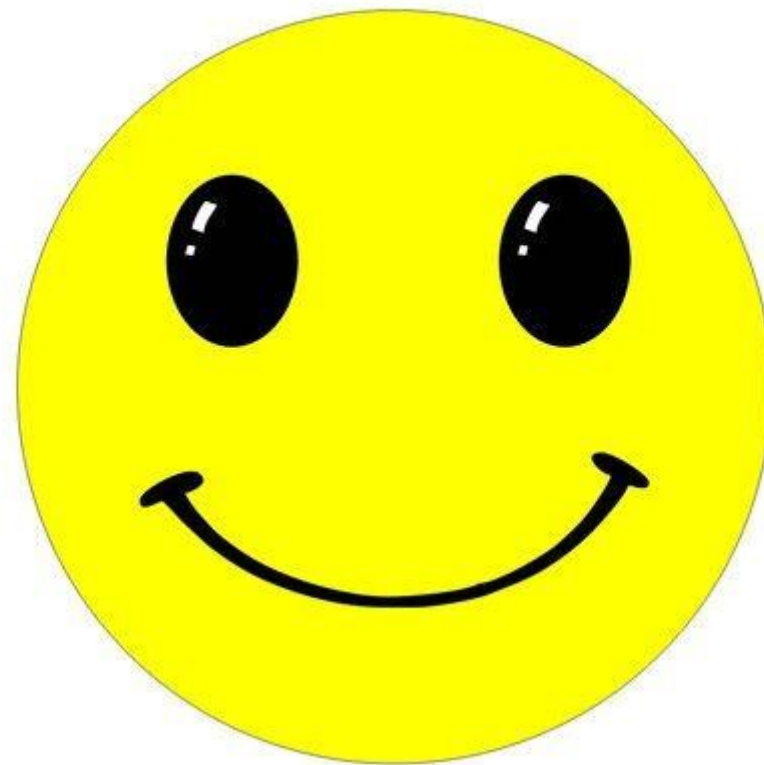
# 实验

- 1、使用编辑器编写简单计算  $\pi$  值的程序;
- 2、运行程序.

```
from random import random as rd
dots = 1000*1000
hits = 0
for i in range(0,dots+1):
    x,y = rd(),rd()
    if pow(x**2+y**2,0.5) <=1:
        hits = hits + 1
pi = (hits/dots)*4
print(pi)
```



- 暂时结束



# Linux/Raspbian 内核的编译

## 为什么要重新编译

重新编译内核的理由：

- 用户需要运行一个程序，但是操作系统一直报告说内核不支持该程序需要的功能。
- 
- 用户希望对内核进行优化，这样它可以在用户的机器上运行得最好，并且还可以少占用RAM系统内存。