**1、树莓派是什么？**  
        Raspberry Pi(中文名为“树莓派”,简写为RPi，或者RasPi/RPi)是为学生计算机编程教育而设计，只有信用卡 大小的卡片式电脑，其系统基于Linux。

        树莓派由注册于英国的慈善组织“Raspberry Pi 基金会”开发，Eben·Upton/埃·厄普顿为项目带头人。2012年3月，英国剑桥大学埃本·阿普顿（Eben Epton）正式发售世界上最小的台式机，又称卡片式电脑，外形只有信用卡大小，却具有电脑的所有基本功能，这就是Raspberry Pi电脑板，中文译名"树莓派"！这一基金会以提升学校计算机科学及相关学科的教育，让计算机变得有趣为宗旨。基金会期望这 一款电脑无论是在发展中国家还是在发达国家，会有更多的其它应用不断被开发出来，并应用到更多领域。

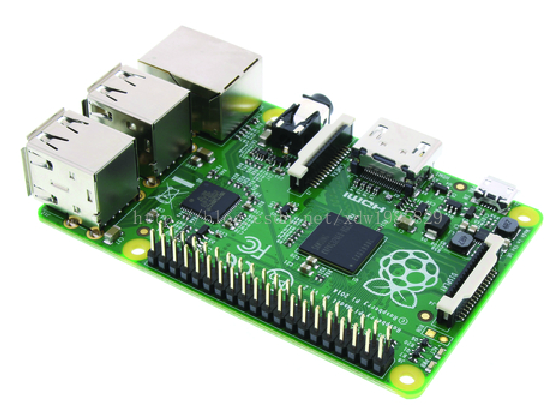
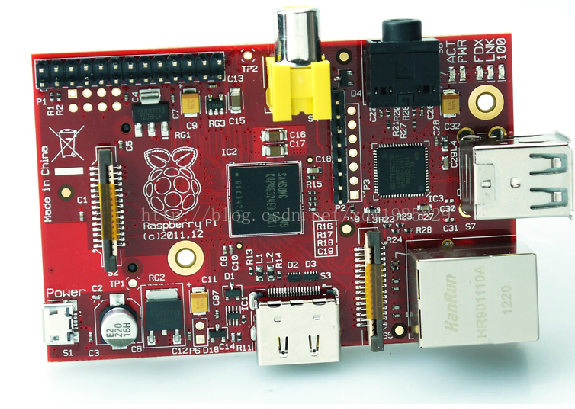
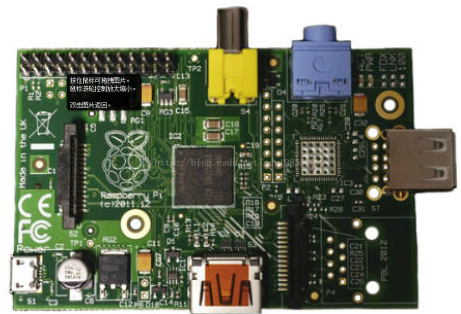
**一句话：树莓派是一个卡片大小的开发板，上面可以运行Linux系统，我们可以用它开发我想要的设备。**  
  
**2、树莓派的种类**

A型：1个USB、无有线网络接口、功率2.5W,500mA、256MB RAM（基本已经见不到了）

B型：2个USB、支持有线网络、功率3.5W，700mA、512MB RAM、26个GPIO(市售还有很多)。

B+型：4个USB口、支持有线网络，功耗1W，512M RAM 40个GPIO（2014新出的，推荐使用）

购买请在淘宝上搜一下，非常多：200到--300元之间（裸板，不含SD卡、电源）



**3、树莓派参数：**

B型：

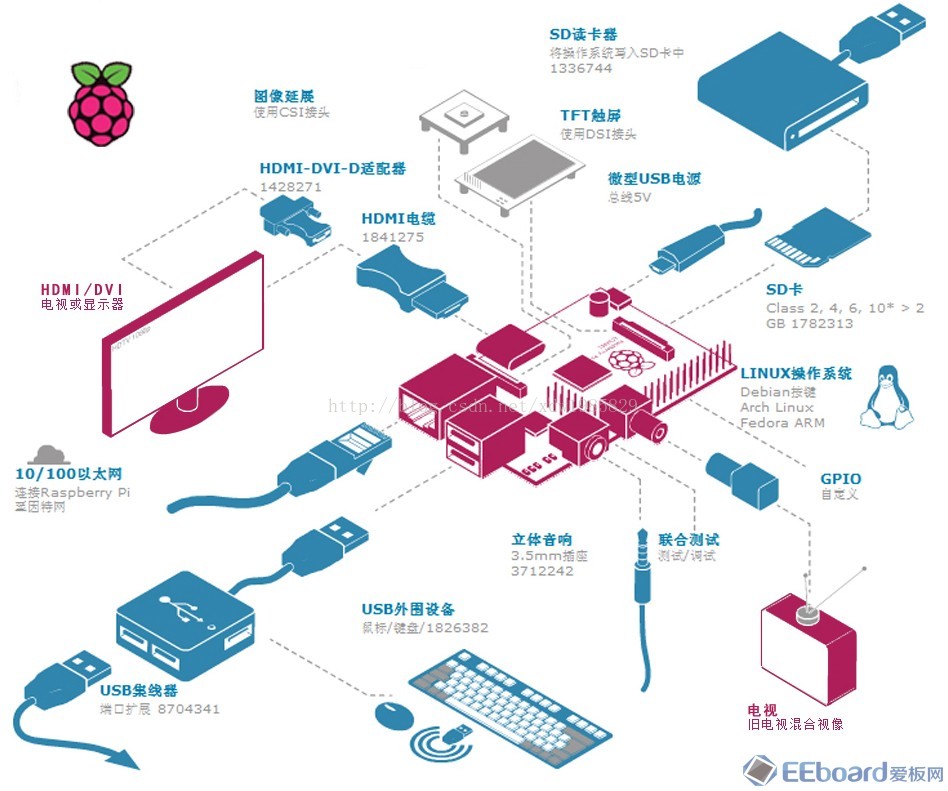
|  |  |
| --- | --- |
| 处理器 | BroadcomBCM2835（CPU，GPU，DSP和SDRAM，USB） |
| CPU | ARM1176JZF-S核心（ARM11系列）700MHz |
| GPU | Broadcom VideoCrore IV，OpenGL ES 2.0,1080p 30 h.264/MPEG-4 AVC高清解码器 |
| 内存 | 512MByte |
| USB 2.0 | 2（支持USB hub扩展） |
| 影像输出 | Composite RCA（PAL & NTSC），HDMI（rev 1.3 & 1.4），raw LCD Panels via DSI 14 HDMI resolution from 640x350 to 1920x1200 plus various PAL andNTSC standards |
| 音源输出 | 3.5mm插孔，HDMI |
| 板载存储 | SD/MMC/SDIO卡插槽 |
| 网络接口 | 10/100以太网接口 |
| 外设 | 8xGPIO、UART、I2C、带两个选择的SPI总线，+3.3V，+5V，ground（负极） |
| 额定功率 | 700mA（3.5W） |
| 电源输入 | 5V / 通过MicroUSB或GPIO头 |
| 总体尺寸 | 85.60 x 53.98 mm（3.370 x 2.125 in） |
| 操作系统 | Debian GNU/linux，Fedora，Arch Linux ARM，RISC OS, XBMC |

**B+改进**

• 更多的GPIO：B+将通用输入输出引脚增加到了40个，而Model B则只有26个；  
• 更多的USB：B+提供了4个USB端口，对热插拔有着更好的兼容性(Model B只有2个)；  
• 支持microSD：旧款的SD卡插槽，已经被换成了更漂亮的推入式microSD卡槽；  
• 更低的功耗：将线性式稳压器换成了开关式，功耗降低了0.5W到1W；  
• 更好的音频：音频电路部分采用了专用的低噪供电；  
• 简洁的外形：USB接口被推到了主板的一边，复合视频移到了3.5mm音频口的位置，此外还增加了4个独立的安装孔。

**4、树莓派分布图**

**B型，B+类似，请自行对比参考**



**1、树莓派启动方式及支持的系统：**

**树莓派开发板没有配置板载FLASH，因为它支持SD卡启动，所有我们需要下载相应镜像，并将其烧写在SD上，启动系统即可**

**(这个镜像里包含了我们通常所说的bootloader、kernel、文件系统)**

树莓派由于其开源特性，支持非常多的系统类型（指的文件系统）：

Raspbian、Arch Linux ARM、Debian Squeeze、Firefox OS、Gentoo Linux  
Google Chrome OS、Raspberry Pi Fedora Remix、Slackware ARM  
QtonPi、Slackware ARM、WebOS、RISC OS、FreeBSD

NetBSD、Android 4.0(Ice Cream Sandwich)

详细介绍请看维基百科：http://zh.wikipedia.org/wiki/%E6%A0%91%E8%8E%93%E6%B4%BE

**2、树莓派镜像下载地址**

http://www.raspberrypi.org/

官网（速度较慢）

http://pan.baidu.com/share/home?uk=671504480#category/type=0

树莓派论坛提供的下载地址 （百度网盘速度较快）

建议下载：wheezy-raspbian

笔者更新时，最新版本为：2014.6.20(B+版建议使用此版本，笔者将旧版烧入B+版中貌似网卡及USB驱动不能识别)

**3、镜像烧写:**

**准备：**

1)一张2G以上的SD卡及读卡器，最好是高速卡，推荐Class4以上的卡，卡的速度直接影响树莓派的运行速度  
   笔者建议最好4G以上，否则后续开发会使用经常不够用

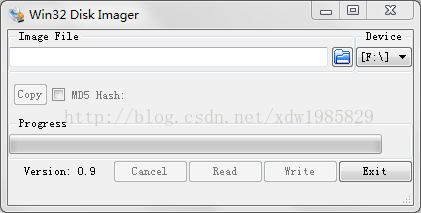
2)winXP和win7下安装镜像的工具：Win32DiskImager.zip  
   同样在上一步的网址下载  
  
3)上一步下载好的镜像

**安装：**

1)解压下载的系统压缩文件，得到img镜像文件

2)将SD使用卡托或者读卡器后，连上电脑

3)解压并运行win32diskimager工具



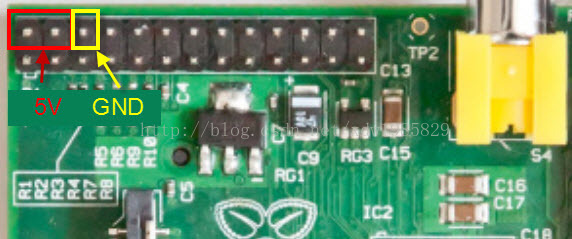
4)在软件中选择img文件，“Device”下选择SD的盘符，然后选择“Write”  
然后就开始安装系统了，根据你的SD速度，安装过程有快有慢。  
  
5)安装结束后会弹出完成对话框，说明安装就完成了，如果不成功，请关闭防火墙一类的软件，重新插入SD进行安装  
     请注意安装完，win系统下看到SD只有74MB了，这是正常现象，因为linux下的分区win下是看不到的！

**4、开发板供电：**

树莓派开发板有两种供电方式：

1)使用micro usb供电：所有android手机的充电线都可以给他供电

2)外接电源直接供电：B型（B+型）GPIO 左上角的2、4管角为电源正极，6管角为地





**5、树莓派访问方式**

**1)初始启动：**

**a、外接HDMI或者AV显示器，USB鼠标、键盘**

    把树莓派当作一个小电脑，可以用外接键盘、鼠标操作树莓派，树莓派启动界面会显示在HDMI或者AV显示器上

笔者不太建议这种启动方式——抱着个显示、鼠标键盘太不方便了，而且带HDMI的显示器不好找

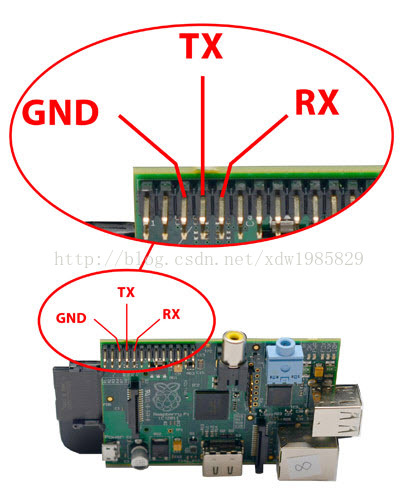
（虽然可以用HDMI转VGA接在电脑显示器上代替）



**b、使用USB转ttl 串口线通过串口访问树莓派**

     树莓派默认支持串口启动，我们只需要一根：usb转ttl线（淘宝上很多，10几块钱一根）连接树莓派与电脑，即可通过串口控制调试树莓派（电脑上使用超级终端或者其它串口调试工具，笔者建议使用Xshell4.0）

     树莓派串口连接口请见下图中所示



在终端中会显示类似启动信息，如下：

中间会提示输出用户名和密码：

**默认用户名：pi**

**默认密码：raspberry**

octopi login: pi  
Password:   
Last login: Fri Jun 20 09:32:38 UTC 2014 on ttyAMA0  
Linux octopi 3.12.22+ #691 PREEMPT Wed Jun 18 18:29:58 BST 2014 armv6l  
  
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.  
  
NOTICE: the software on this Raspberry Pi has not been fully configured. Please run 'sudo raspi-config'

pi@octopi:~$

**2)其它访问方式：**

**SSH：**

由于树莓派只有一个串口，当我们开发中需要占用这个串口或者我们希望进行远程开发（通过一个根网线或无线连接到树莓派）

可以借助Linux系统特有的一种服务SSH来完成（请见后面的文章）

**VNC：**

VNC是linux下面常用的远程桌面，用它可以在windows或者unix主机上方便的通过网络操作远程主机而不需要一个额外的显示器，非常实用（请见后面的文章）

**远程桌面：**

远程桌面是windows自带的一个远程桌面登陆系统，很好用（请见后面的文章）

以上三种方式都是通过网络对树莓派进行控制，可以说只要知道了树莓派的ip地址，一切控制so easy

[树莓派开发系列教程3——树莓派rasp-config配置](http://blog.csdn.net/xdw1985829/article/details/38816375)

**经过前面两步我们的树莓派已经正常的工作起来了，但是在真正用它开发之前还需要进行一些列的配置以及软件的安装，这样开发起来才会得心应手，下面我们介绍一下常用的软件和服务**

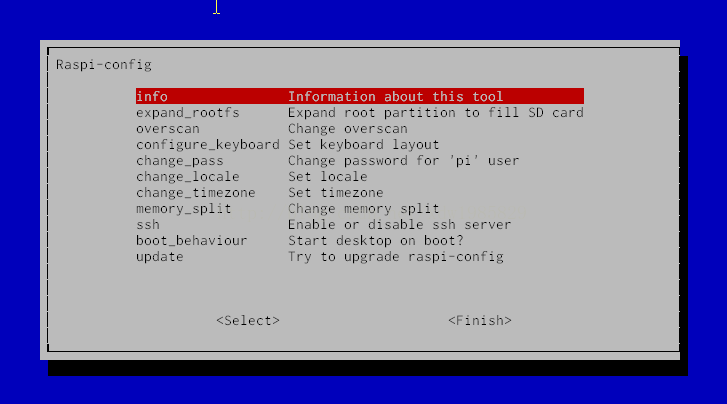
**1、配置选项：**

    树莓派第一次使用的时候需要进行一个简单的配置，在命令行模式下运行以下命令：

    $sudo raspi-config

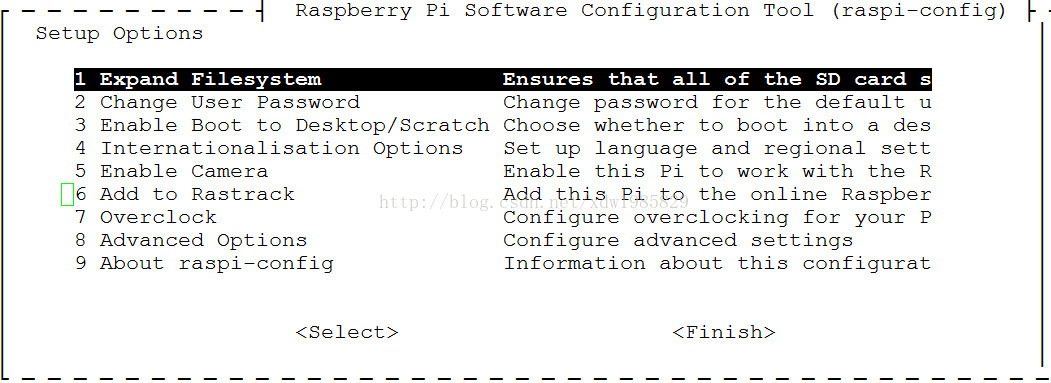
新旧版本的配置界面不太一样，下面列举两种比较常见的：

**1)旧版本**



* **expand\_rootfs** –将根分区扩展到整张SD卡（树莓派默认不使用SD卡的全部空间，有一部分保留，建议选中）
* **overscan**–可以扩充或缩小屏幕（旧版不能自适应屏幕，新版没有这个选项，貌似可以自适应，没仔细研究）
* **configure\_keyboard**  - 键盘配置界面
* **change\_pass** – 默认的用户名是pi，密码是raspberry，用ssh远程连接或串口登录时要用到这个用户名和密码，这里可以更改密码。
* **change\_locale** – 更改语言设置。在Locales to be generated: 中，选择en\_US.UTF-8和zh\_CN.UTF-8。在Default locale for the system environment:中，选择en\_US.UTF-8（等启动完机器，装完中文字体，再改回zh\_CN.UTF-8，否则第一次启动会出现方块）。
* **change\_timezone** –因为树莓派没有内部时钟，是通过网络获取的时间，选择Asia – Shanghai。
* **memory\_split** –配置给桌面显示的显存。
* **ssh** – 是否激活sshd服务。
* **boot\_behaviour** – 设置启动时启动图形界面，正常肯定是Yes。

**2）新版本（比较新的镜像大部分是这个界面，做了不少改变）**



**1 Expand Filesystem  扩展文件系统（同旧版）。  
2 Change User Password  改变默认pi用户的密码，按回车后输入pi用户的新密码。  
3 Enable Boot to Desktop/Scratch 启动时进入的环境选择**  
  Console Text console, requiring login(default)  
    启动时进入字符控制台，需要进行登录（默认项）。  
  Desktop log in as user 'pi' at the graphical desktop  
    启动时进入LXDE图形界面的桌面。  
  Scratch Start the Scratch programming environment upon boot  
    启动时进入Scratch编程环境。  
**4 Internationalisation Options  国际化选项，可以更改默认语言**  
  I1 Change Locale  
     语言和区域设置，建议不要改，默认英文就好。想改中文，最好选安装了中文字体再进行这步，安装中文字体的方法：  
     sudo apt-get update  
     sudo apt-get install ttf-wqy-zenhei ttf-wqy-microhei  
     移动到屏幕底部，用空格键选中zh-CN GB2312,zh-CN GB18030,zh-CN UTF-8,然后按回车，然后默认语言选中zh-cn然后回车  
  I2 Change Timezone  
     设置时区，如果不进行设置，PI的时间就显示不正常。选择Asia（亚洲）再选择Chongqing（重庆）即可。  
  I3 Change Keyboard Layout  
      改变键盘布局  
**5 Enable Camera**  
      启动PI的摄像头模块，如果想启用，选择Enable，禁用选择Disable就行了  
**6 Add to Rastrack**  
      把你的PI的地理位置添加到一个全世界开启此选项的地图，建议还是不要开了，免得被跟踪。  
**7 Overclock**  
  None 不超频，运行在700Mhz，核心频率250Mhz，内存频率400Mhz，不增加电压  
  Modest 适度超频，运行在800Mhz，核心频率250Mhz，内存频率400Mhz，不增加电压  
  Medium 中度超频，运行在900Mhz，核心频率250Mhz，内存频率450Mhz，增加电压2  
  High 高度超频，运行在950Mhz，核心频率250Mhz，内存频率450Mhz，增加电压6  
  Turbo 终极超频，运行在1000Mhz，核心频率500Mhz，内存频率600Mhz，增加电压6  
**8 Advanced Options  高级设置**  
  A1 Overscan   是否让屏幕内容全屏显示  
  A2 Hostname   在网上邻居或者路由器能看到的主机名称  
  A3 Memory Split 内存分配，选择给GPU多少内存  
  A4 SSH 是否运行SSH登录，建议开户此选项，以后操作PI方便，有网络就行，不用开屏幕了。  
  A5 SPI 是否默认启动SPI内核驱动，新手就不用管了。  
  A6 Audio 选择声音默认输出到模拟口还是HDMI口  
    0 Auto 自动选择  
    1 Force 3.5mm ('headphone') jack强制输出到3.5mm模拟口  
    2 Force HDMI 强制输出到HDMI  
  A7 Update 把raspi-config这个工具自动升级到最新版本

**9 About raspi-config 关于raspi-config的信息。**

**配置完成后确认会重启系统，选项即可生效**

[树莓派开发系列教程4——树莓派网络与更新配置（有线、无线）](http://blog.csdn.net/xdw1985829/article/details/38817195)

**树莓派的使用无法离开网络，如果没有网络，建议大家洗洗睡吧，没有网路就不要玩树莓派了~**

**1、有线网络：**

    1)自动获取IP:树莓派默认有线网卡是使能的，只需将网线插入树莓派网卡，即可自动获得IP(要求在局域网内)

    2)手动设定IP:如果是电脑与树莓派直连，不能自动获得IP，可以使用：ifconfig eth0 192.168.1.123设定ip(下次重启就没了)

    3)设置静态IP：如果担心在同网络情况下ip或者不固定，可以讲电脑设置为静态ip，方法如下：

    在终端中打开一下文件：

sudo vi  /etc/network/interfaces(需要学习VI基本操作，树莓派自带的vi不是很好用，请先参考下面文档更新一下vi）

auto lo  
iface lo inet loopback

**iface eth0 inet dhcp**  
  
讲以上内容改为：  
auto lo  
iface lo inet loopback  
  
**iface eth0 inet static  
address 192.168.1.1  
netmask 255.255.255.0  
gateway 192.168.1.1**

**2、无线网络**

**有线网络需要受到网线的限制，现在无线网络已经非常的普及，通过一个无线网卡与无线路由器链接起来岂不是更加无拘无束，下面介绍一下无线网络的配置。**

**1)无线网卡驱动的确认**

树莓派内置了很多无线网卡的驱动，大家可以在这个网站查找所支持的型号

http://elinux.org/RPi\_VerifiedPeripherals#USB\_Wi-Fi\_Adapters

笔者手里面有一个:8188CUS(网卡芯片)的验证支持

**验证方法：**

将USB无线网卡插入树莓派USB接口（旧版系统会自动重启，新版不会），敲入：

$lsusb：

lsusb  
**Bus 001 Device 002: ID 0424:9514 Standard Microsystems Corp.  
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp.  
Bus 001 Device 004: ID 0bda:8176 Realtek Semiconductor Corp. RTL8188CUS 802.11n WLAN Adapter**

或者敲入：

$ifconfig:

wlan0  Link encap:Ethernet  HWaddr 00:0b:81:87:e5:f9    
          UP BROADCAST MULTICAST  MTU:1500  Metric:1  
          RX packets:0 errors:0 dropped:3 overruns:0 frame:0  
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000   
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

说明树莓派支持你的无线网卡，可进行下一步设置

**2)无线网卡的配置**

使用vi打开以下文件进行修改：

auto lo  
iface lo inet loopback  
iface eth0 inet dhcp

**auto wlan0  
allow-hotplug wlan0  
iface wlan0 inet dhcp  
wpa-ssid "360WiFi"  
wpa-psk "hellworld"  
#wpa-roam /etc/wpa\_supplicant/wpa\_supplicant.conf**

重启即可连入:网络名为360WIFI，密码为helloworld的网络，

如果想要设置为静态ip，类似有线网卡，修改相应代码即可

**3、更新源测试：**

有线或者无线网络连通过，我们后面对软件更新是，需要首先进行更新列表更新，执行以下命令即可：

**sudo apt-get update**

Get:1 http://raspberrypi.collabora.com wheezy Release.gpg [836 B]                
Get:2 http://archive.raspberrypi.org wheezy Release.gpg [490 B]                  
Get:3 http://raspberrypi.collabora.com wheezy Release [7,532 B]                  
Get:4 http://mirrordirector.raspbian.org wheezy Release.gpg [490 B]              
Get:5 http://mirrordirector.raspbian.org wheezy Release [14.4 kB]                
Get:6 http://raspberrypi.collabora.com wheezy/rpi armhf Packages [2,214 B]       
Get:7 http://archive.raspberrypi.org wheezy Release [7,263 B]

....

Ign http://mirrordirector.raspbian.org wheezy/rpi Translation-en\_GB  
Ign http://mirrordirector.raspbian.org wheezy/rpi Translation-en  
Fetched 6,992 kB in 1min 12s (96.6 kB/s)        
Reading package lists... Done

**改变更新源**：

笔者测试这个默认的网址还是挺快的，网络上有人建议改变更新源为国内的（我没发现快多少，大家根据自己网络选择）：

sudo vi /etc/apt/sources.list

将默认的用#号屏蔽，改为

**deb http://mirrors.ustc.edu.cn/raspbian/raspbian/   wheezy main contrib non-free rpi**

**或者**

**deb http://mirror.nus.edu.sg/raspbian/raspbian wheezy main contrib non-free rpi**

**或者用以下地址代替上面的地址栏**

中山大学

Raspbian http://mirror.sysu.edu.cn/raspbian/raspbian/  
  
中国科学技术大学

Raspbian http://mirrors.ustc.edu.cn/raspbian/raspbian/  
  
清华大学

Raspbian http://mirrors.tuna.tsinghua.edu.cn/raspbian/raspbian/  
  
华中科技大学

Raspbian http://mirrors.hustunique.com/raspbian/raspbian/

Arch Linux ARM http://mirrors.hustunique.com/archlinuxarm/

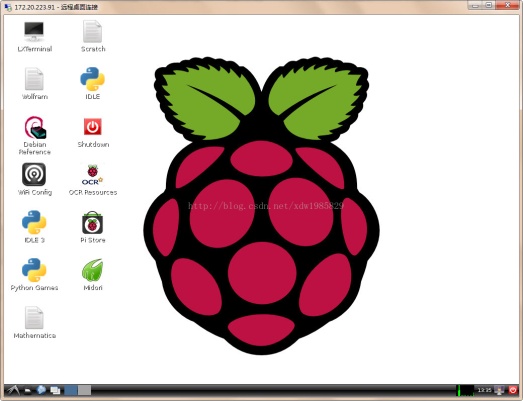
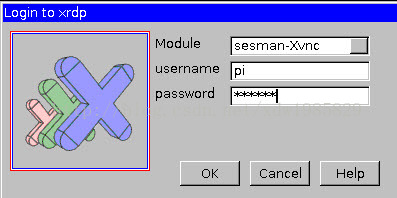
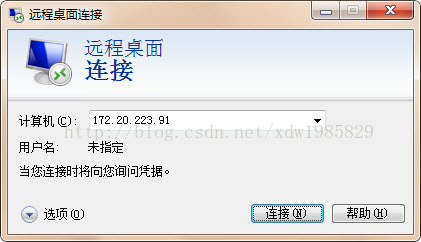
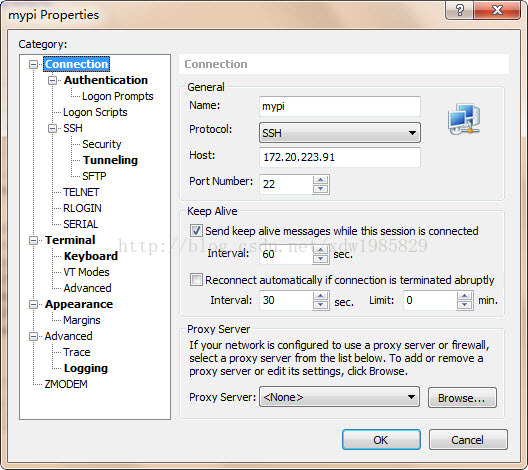
大连东软信息学院源（北方用户）  
Raspbian http://mirrors.neusoft.edu.cn/raspbian/raspbian/

[树莓派开发系列教程5——树莓派常用软件及服务(vi、远程桌面、ssh、samba、u盘)](http://blog.csdn.net/xdw1985829/article/details/38818193)

**1、更新vim**  
树莓派自带的vim不是很好用，比如默认不支持小键盘及方向键等，我们可以更新一下

sudo apt-get install vim

**2、使用远程桌面**  
通过windows自带的远程桌面软件，可以远程访问树莓派桌面应用

sudo apt-get install xrdp  
  
  
  
**3、ssh**ssh服务默认已经安装完毕  
只需要安装一个ssh客户端即可，笔者推荐xshell4.0(Xmanage4.0的一个组件)  
host填入:开发板ip即可  
用户名、密码同串口登陆一致（默认：pi、raspberry）  
  
  
**4、使用samba**  
通过samba服务，可以自由、访问修改树莓派家目录下的文件、代码

**1)首先安装 samba：**

sudo apt-get install samba samba-common-bin

**2)然后修改配置文件：**

sudo vi /etc/samba/smb.conf

下面的配置是让用户可以访问自己的 home 目录。

a)开启用户认证，找到“##### Authentication #####”，将“#  security = user ”的井号去掉。

b)配置用户可以读写自己的 home 目录，在“[homes]”节中，把 “read only = yes” 改为 “read only = no”

c)找到browseable=no改为yes 否则等会访问时此文件夹会隐藏

**3)重启samba服务：**

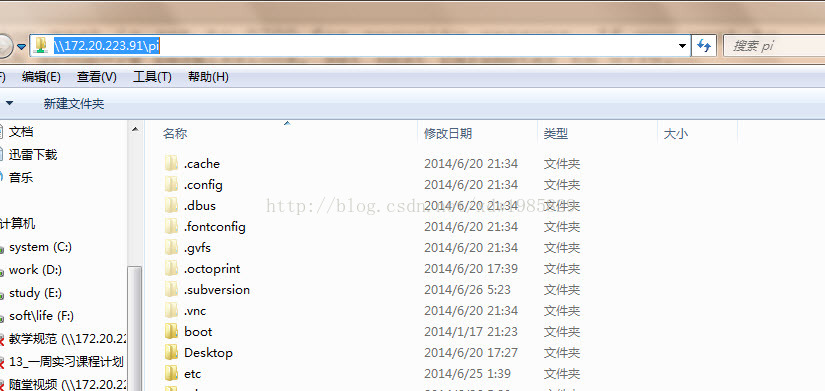
sudo /etc/init.d/samba restart

**4)把系统默认用户pi添加到samba ：**

sudo smbpasswd -a pi

会提示你输入密码：建议跟串口登陆的密码一致，防止密码太多弄混了  
  
**5)windows访问**  
任意打开一个文件浏览器，输入ip地址:\\172.20.223.91\pi

**现在就可以从其它机器上访问 pi 这个用户的 home 目录了。**

  
  
**5、挂载U盘**  
当需要大容量外接存储设备时，可使用U盘挂载

通常我们在 /mnt 或 /media 目录下新建一个目录作为挂载点。比如：

sudo mkdir /mnt/udisk  
  
**1)手动挂挂载：**  
挂载命令：

sudo mount -o uid=pi,gid=pi /dev/sda1 /mnt/udisk

用完之后卸载

sudo umount /mnt/1GB\_USB\_flash

**注意：**

sda1 是取决于你的实际情况，a表示第一个硬盘，1表示第一个分区。

FAT 格式U盘 mount 本身就能支持，但如果你的U盘或移动硬盘使用的是 exFAT 格式，mount 会说不支持。没关系，安装 exfat-fuse 软件之后 mount 就支持了。

sudo apt-get install exfat-fuse

**2)开机挂载：**

如果想开机自动挂载，而不是每次手工执行，可以编辑 /etc/fstab 文件。在末尾添加一行：

/dev/sda1 /mnt/udisk vfat rw,defaults 0 0  
每次开机就会自动挂载  
  
**3)热插挂载  
需要希望想电脑一样，插上自动识别挂载在某一目录下，拔下自动umount，请按一下操作  
sudo vi /etc/udev/rules.d/10-usbstorage.rules(此文件默认没有，需要新建)  
赋值以下内容即可，会自动在/mnt/udisk目录下挂载U盘**

**[ruby]** [view plaincopy](http://blog.csdn.net/xdw1985829/article/details/38818193)

1. KERNEL!="sd\*", GOTO="media\_by\_label\_auto\_mount\_end"
2. SUBSYSTEM!="block",GOTO="media\_by\_label\_auto\_mount\_end"
3. IMPORT{program}="/sbin/blkid -o udev -p %N"
4. ENV{ID\_FS\_TYPE}=="", GOTO="media\_by\_label\_auto\_mount\_end"
5. ENV{ID\_FS\_LABEL}!="", ENV{dir\_name}="%E{ID\_FS\_LABEL}"
6. ENV{ID\_FS\_LABEL}=="", ENV{dir\_name}="Untitled-%k"
7. ACTION=="add", ENV{mount\_options}="relatime,sync"
8. ACTION=="add", ENV{ID\_FS\_TYPE}=="vfat", ENV{mount\_options}="iocharset=utf8,umaskk
9. =000"
10. ACTION=="add", ENV{ID\_FS\_TYPE}=="ntfs", ENV{mount\_options}="iocharset=utf8,umaskk
11. =000"
12. ACTION=="add", RUN+="/bin/mkdir -p /mnt/udisk/", RUN+="/bin/mount -o **$env**{mount\_\_
13. options} /dev/%k /mnt/udisk/"
15. ACTION=="remove", ENV{dir\_name}!="", RUN+="/bin/umount -l /mnt/udisk/}", RUN+="//
16. bin/rmdir /mnt/udisk/"
17. LABEL="media\_by\_label\_auto\_mount\_end"

[树莓派开发系列教程6——树莓派做wifi热点](http://blog.csdn.net/xdw1985829/article/details/38845533)

**原理：Pi使用有线连入网络，然后接USB无线网卡作为热点，提供Wifi接入。**

**1、USB无线网卡驱动**

如果接上USB无线网卡，使用ifconfig命令，能直接看到wlan0，那么恭喜你，可以直接跳过这一步。

如果没有请查询一下树莓派支持的USB无线网卡型号，可参考以下网址：

http://elinux.org/RPi\_VerifiedPeripherals#USB\_Wi-Fi\_Adapters

**2、修改wlan0为静态IP**

**相当于设置路由器lan口IP,即我们访问路由器通常使用的:192.168.1.1**

**sudo vim /etc/network/interfaces**  
把原来关于wlan0的注释掉：（可能跟这个不一样，跟wlan0有关的注释掉即可）

#auto wlan0  
#iface wlan0 inet dhcp  
#wpa-ssid "360WiFi-li"  
#wpa-psk "xiaolizi"

添加下面的：

**iface wlan0 inet static  
address 192.168.0.1  
netmask 255.255.255.0**

**gateway 192.168.0.1**

**完成之后需要重启**

**3、安装hostapd**

**官方的hostapd不支持8188CUS，后面需要重新卸载安装新的**

**笔者测试貌似这里必须先装旧的，然后后面卸了装新的，否则也不能用**

sudo apt-get install hostapd

**1）编辑hostapd 默认配置文件：**

sudo vim /etc/default/hostapd

找到#DAEMON\_CONF= ""，修改为：

**DAEMON\_CONF="/etc/hostapd/hostapd.conf"**

**2）然后编辑：sudo vim /etc/hostapd/hostapd.conf**

**增加以下代码：**

**[ruby]** [view plaincopy](http://blog.csdn.net/xdw1985829/article/details/38845533)

1. # Basic configuration
3. interface=wlan0
4. ssid=raspberrywifi
5. channel=1
6. #bridge=br0
8. # WPA and WPA2 configuration
10. macaddr\_acl=0
11. auth\_algs=1
12. ignore\_broadcast\_ssid=0
13. wpa=3
14. wpa\_passphrase=12345678
15. wpa\_key\_mgmt=WPA-PSK
16. wpa\_pairwise=TKIP
17. rsn\_pairwise=CCMP
19. # Hardware configuration
21. driver=rtl871xdrv
22. ieee80211n=1
23. hw\_mode=g
24. device\_name=RTL8192CU
25. manufacturer=Realtek

修改wifi名和密码

**ssid=raspberrywifi**

**wpa\_passphrase=12345678**

**3）保存退出，然后重启服务：**

**sudo service hostapd restart**

或者执行以下命令生效

**sudo hostapd -dd /etc/hostapd/hostapd.conf**

**4）如果你使用的网卡提示一下信息**

Configuration file: /etc/hostapd/hostapd.conf  
nl80211: 'nl80211' generic netlink not found  
Failed to initialize driver 'nl80211'  
rmdir[ctrl\_interface]: No such file or directory

那么，还是要使用第三方的hostapd。

**4、安装新的hostapd**

**1)删除原来的hostapd（笔者测试，不卸载貌似也行）**

**sudo apt-get autoremove hostapd**

**2)下载第三方驱动并安装**

**wget https://github.com/jenssegers/RTL8188-hostapd/archive/v1.1.tar.gz**

**tar -zxvf v1.1.tar.gz**

**3)编译：**

**cd RTL8188-hostapd-1.1/hostapd**

**sudo make  
sudo make install**

**4)然后再重启服务，应该提示成功：**

**$ sudo service hostapd restart**

**[ ok ] Stopping advanced IEEE 802.11 management: hostapd.**

**[ ok ] Starting advanced IEEE 802.11 management: hostapd.**

**5)将hostapd加入开机自启动**

**sudo service hostapd start**

**sudo update-rc.d hostapd enable**

**笔者这里提示的还是失败，但是重启后网络确实建立成功，用手机可以搜到这个网络**

**5、安装DHCP服务**

**以上步骤建立起了wifi热点，但是无法自动获取ip，需要以下步骤**

**sudo apt-get install udhcpd**

**1)编辑配置文件：**  
**sudo vim /etc/udhcpd.conf //修改以下信息，start和end是重点，注意跟第一步的静态ip在一个网段**

**start 192.168.0.20  
end 192.168.0.200  
interface wlan0**

**2)接下来编辑/etc/default/udhcpd并且将下面这行注释掉，以使DHCP Server正常工作：**

**#DHCPD\_ENABLED="no"**

**3)启动dhcp服务器**

**sudo service udhcpd start**

**sudo update-rc.d udhcpd enable**

**经过此步手机已经可以接入wifi网络，并且自动获取ip**

**6、配置路由转发**

**理论上是经过这一步，手机可以通过共享树莓派的无线网络上网了，但是笔者一直没有成功**

**1)设置路由映射规则**

sudo iptables -F  
sudo iptables -X

sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

sudo iptables -A FORWARD -i eth0 -o wlan0 -m state --state RELATED,ESTABLISHED -j ACCEPT

sudo iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT

sudo bash -c iptables-save > /etc/iptables.up.rules

**2)编辑：sudo vim /etc/network/if-pre-up.d/iptables**

添加下面两行代码：

**#!/bin/bash  
/sbin/iptables-restore < /etc/iptables.up.rules**

保存退出，然后修改iptables权限：

**sudo chmod 755 /etc/network/if-pre-up.d/iptables**

**4)开起内核转发：**

**sudo vim /etc/sysctl.conf**

  找到下面两行：

**#Uncomment the next line to enable packet forwarding for IPv4  
#net.ipv4.ip\_forward=1  
把net.ipv4.ip\_forward 前面的#去掉，保存退出。**

  然后**:sudo sysctl -p**

**7、其它问题**

最近经常发现无线网卡配置的DHCP不能发挥作用，经过排查发现给无线网卡指定的静态IP失败了，也就是说无线网卡没有IP导致DHCP无法工作，将/etc/default/ifplugd的内容修改配置如下：

**INTERFACES="eth0"**

**HOTPLUG\_INTERFACES="eth0"**

**ARGS="-q -f -u0 -d10 -w -I"**

**SUSPEND\_ACTION="stop"**

[树莓派开发系列教程7——树莓派做web服务器(nginx、Apache)](http://blog.csdn.net/xdw1985829/article/details/38919495)

**一想到Linux Web服务器，我们首先想到的是：**

**Apache + MySql + Php．**

**Apache:**是世界使用排名第一的Web服务器软件。

可以运行在几乎所有广泛使用的计算机平台上，由于其跨平台和安全性被广泛使用，是最流行的Web服务器端软件.

**MySQL:**是一个关系型数据库管理系统，由瑞典MySQL AB公司开发。是最流行的关系型数据库管理系统，在WEB应用方面MySQL是最好的RDBMS(关系数据库管理系统)应用软件之一。

**PHP:**（外文名: Hypertext Preprocessor，中文名：“超文本预处理器”）是一种通用开源脚本语言。语法吸收了C语言、Java和Perl的特点，易于学习，使用广泛，主要适用于Web开发领域。

**树莓派可以安装这个LAMP系列，但Apache 和 MySql对于树莓派这个小小的机器，太重了，主要是消耗内存多＼速度慢＼占用磁盘大(约200M吧)，所可以选择安装一个轻量级的Web服务器：**

**nginx + php + sqlite**

**nginx:**是个轻量级的Web服务器，是一款轻量级的Web 服务器/反向代理服务器及电子邮件（IMAP/POP3）代理服务器,上nginx的并发能力确实在同类型的网页服务器中表现较好。

**SQLite:**是一款轻型的数据库，是遵守ACID的关系型数据库管理系统，它的设计目标是嵌入式的，而且目前已经在很多嵌入式产品中使用了它，它占用资源非常的低，在嵌入式设备中，可能只需要几百K的内存就够了。

**Apache + MySql + Php．**

**1、安装Apache**

Apache可以用下面的命令来安装  
sudo apt-get install apache2

Apache默认路径是/var/www/

其配置文件路径为： /etc/apache2/

可以通过：sudo vi /etc/apache2/ports.conf修改监听端口号

重启服务生效：sudo service apache2 restart

**2、安装mysql**  
sudo apt-get install mysql-server  
安装过程中，会出现一个提示符让你输入一个密码。  
这个密码是mysql root用户的密码。

**3、安装PHP**  
输入下面的命令，就可以安装PHP 5,以及PHP访问mysql数据库所需要的库。

sudo apt-get install php5

sudo apt-get install php5-mysql

**4、测试**

安装完成后，可以在浏览器中输入你路由器的IP或域名，就可以访问你的网站了。

你应该能看到一个页面显示“It works”，但是没有其它内容。

创建一个/var/www/index.php

**[php]** [view plaincopy](http://blog.csdn.net/xdw1985829/article/details/38919495)

1. <?php
2. print <<< EOT
3. <!doctype html>
4. <html lang="en">
5. <head>
6. <meta charset="UTF-8">
7. <title>Test successful</title>
8. </head>
9. <body>
10. <h1>Test successful</h1>
11. <p>Congratulations.</p>
12. <p>Your webserver **and** PHP are working.</p>
13. </body>
14. </html>
15. EOT;
17. ?>

**二、nginx + php + sqlite**

**1、安装nginx  web服务器 (约6MB)**  
sudo apt-get install nginx

**2、启动nginx**

sudo /etc/init.d/nginx start

nginx的www根目录默认在 /usr/share/nginx/www中

**3、修改nginx的配置文件**  
sudo vi /etc/nginx/sites-available/default

**1)以下几个选项注意一下：**

listen   8080;                                             ## listen for ipv4; this line is default and implied

//监听的端口号，如果与其它软件冲突，可以在这里更改

root /usr/share/nginx/www;

//nginx 默认路径html所在路径

index index.html index.htm **index.php**;

//nginx默认寻找的网页类型，我们可以增加一个index.php

**2)PHP脚本支持（不设这几项PHP脚本无法识别）**

找到php的定义段，将这些行的注释去掉 ，修改后内容如下

location ~ .php$ {  
　fastcgi\_pass unix:/var/run/php5-fpm.sock;  
　fastcgi\_index index.php;  
　include fastcgi\_params;  
}  
  
php段中有一些其它定义，不要去动它，比如：  
#      fastcgi\_split\_path\_info ...  
#      fastcgi\_pass 127.0.0.1:9000

**4)安装php和sqlite(约3MB)**  
sudo apt-get install php5-fpm php5-sqlite

**5)重新加载nginx的配置**  
sudo /etc/init.d/nginx reload

**6)测试html**

通过主机的IE访问树莓派，可以看到主页(表示Web服务器已正常启动)

**7)测试php**

在树莓派中生成一php文件

sudo vi /usr/share/nginx/www/index.php

在文件中输入以下内容

**[php]** [view plaincopy](http://blog.csdn.net/xdw1985829/article/details/38919495)[在CODE上查看代码片](https://code.csdn.net/snippets/460304)

1. <?php
2. print <<< EOT
3. <!doctype html>
4. <html lang="en">
5. <head>
6. <meta charset="UTF-8">
7. <title>Test successful</title>
8. </head>
9. <body>
10. <h1>Test successful</h1>
11. <p>Congratulations.</p>
12. <p>Your webserver **and** PHP are working.</p>
13. </body>
14. </html>
15. EOT;
17. ?>

存盘退出  
  
IE访问一下这一页，说明php也是OK的

[树莓派开发系列教程8——树莓派内核编译与固件升级](http://blog.csdn.net/xdw1985829/article/details/39077611)

**前面我们已经使用官方给的镜像搭建好了树莓派的开发环境，可以说已经可以上手应用了，但是如果官方提供的内核有些功能不能满足我们的需要或者需要对内核进行部分裁剪，那么就需要对树莓派的内核进行裁剪，这里我们教一下大家如何更新树莓派内核**

**1、获取升级所需源码**

**1)下载地址：**

**官方网址：https://github.com/raspberrypi**

上面列出了树莓派所有的开源软件：

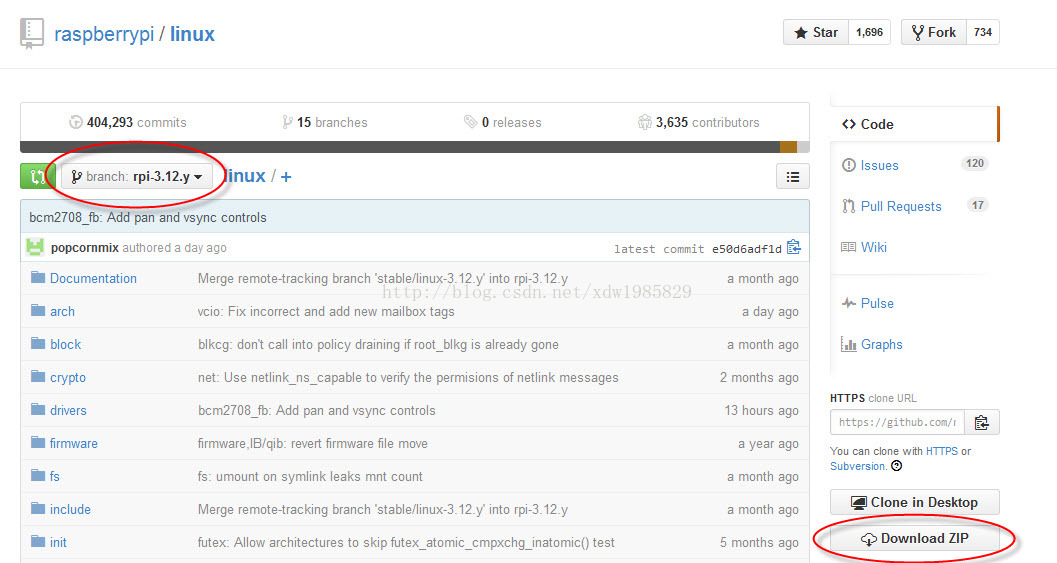
1. **firmware:树莓派的交叉编译好的二进制内核、模块、库、bootloader**
2. **linux:内核源码**
3. **tools:编译内核和其他源码所需的工具——交叉编译器等**

**我们只需要以上三个文件即可，下面的工程可以了解一下**

1. documentation:树莓派离线帮助文档，教你如何使用、部署树莓派（树莓派官方使用教程）
2. userland：arm端用户空间的一些应用库的源码——vc视频硬浮点、EGL、mmal、openVG等
3. hats：Hardware Attached on Top，树莓派 B+型板子的扩展板资料
4. maynard：一个gtk写成的桌面环境
5. scratch：一个简易、可视化编程环境
6. noobs:一个树莓派镜像管理工具，他可以让你在一个树莓派上部署多个镜像
7. weston：一个应用程序
8. target\_fs：树莓派最小文件系统，使用busybox制作
9. quake3：雷神之锤3有线开发源码firmwareb

**2)下载方法：**

**a、网页直接下载：**



**点到所需要下载的工程，左上角选版本，右方有一个download ZIP按钮可直接下载（笔者下载完成后，在linux中解压提示出错，windows又非常慢切内核建议不要在windows环境解压，所以笔者不建议使用这种办法）**

**b、使用git下载**

$ mkdir raspeberrypi\_src  
$ cd raspberrypi\_src  
$ git clone git://github.com/raspberrypi/firmware.git  
$ git clone git://github.com/raspberrypi/linux.git  
$ git clone git://github.com/raspberrypi/tools.git

**会得到三个文件夹：**

**firmware linux tools**

**2、编译、提取内核及其模块**

**1)获得内核配置文件**

在运行的树莓派中运行：

**$ls /proc/**

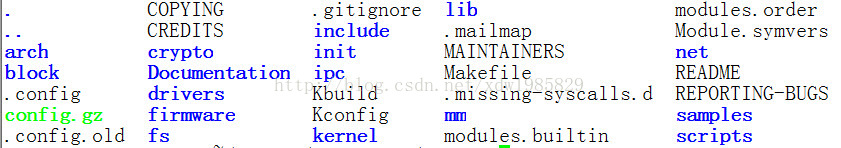
可看到一个叫config.gz的文件，他是当前的树莓派配置选项记录文件，我们将他拷出，放入我们的内核源码目录树下

**$cp /proc/config /home/pi**

我们这里使用前面交过的samba拷出并拷入内核源码目录下，不熟悉的人可参考前面文章

在linux内核源码下执行：

**$zcat config.gz > .config**

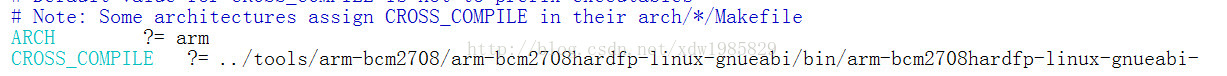


**2)配置、编译内核**

**a、修改内核源码makefile ARCH类型和编译器路径**

**$vi Makefile +195**

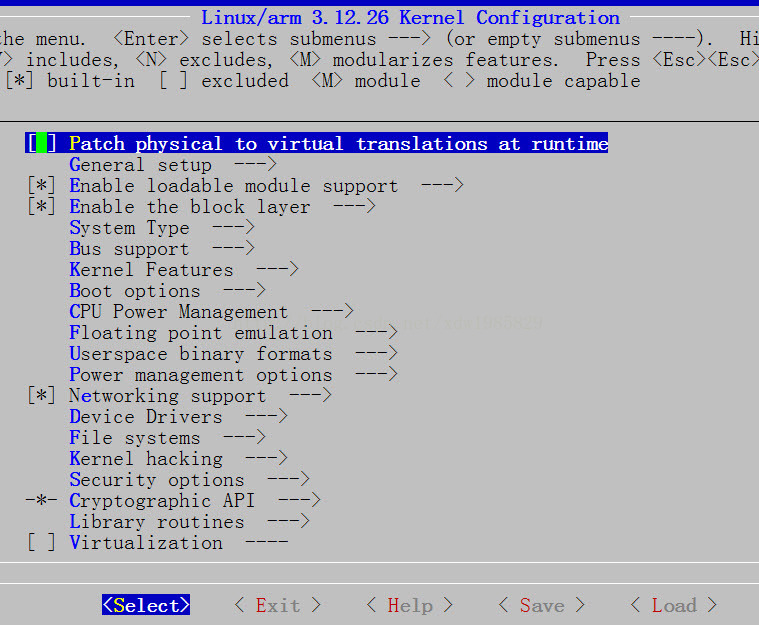
找到以上类似代码，改为如图所示



**b、查看、修改配置选项**

$make menuconfig

可出现以下界面



**如果不做修改，直接选中exit即可（注意使用键盘操作）**

**c、编译内核镜像**

**$make**

在arch/arm/boot目录下可以看到一个叫zImage的文件，就是我们新的内核

但是树莓派需要另外一种格式的镜像，需要进行处理一下，执行以下命令

**$cd tools/mkimage**

**$./imagetool-uncompressed.py ../../linux/arch/arm/boot/zImage**

即可在当前文件夹下看到一个叫：kernel.img的文件，就是我们需要的新内核了

**d、提取modules**

上一步其实不但编译出来了内核的源码，一些模块文件也编译出来了，这里我们提取一下

$cd raspberrypi\_src

$mkdir modules

$cd linux

$ make modules\_install INSTALL\_MOD\_PATH=../modules

即可在modules得到我们需要的模块文件

**2、升级RPi的kernel、Firmware、lib**

将SD卡拔下插在电脑上（可使用读卡器）

**1)升级内核**

将新编好的内核拷入SD卡，改名为：kernel\_new.img

打开boot目录下

找到config.txt文件，加入：kernel=kernel\_new.img这一行

**2）升级boot**

将firmware/boot/目录下 以下文件拷入SD卡boot目录：fbootcode.bin fixup.dat fixup\_cd.dat start.elf

**3)更新vc库及内核modules**

将第3步d步中编译出来的modules/lib/modules拷入树莓派文件系统/lib下

[树莓派开发系列教程9——树莓派GPIO控制](http://blog.csdn.net/xdw1985829/article/details/39580401)

**一、常用开源工程简介**

    树莓派内核中已经编译自带了gpio的驱动，我们常通过一些第三方写好的库函数来完成具体的操作，比较常见的操作库函数有：

**1、python GPIO**

    【开发语言】——python

    【简单介绍】——树莓派官方资料中推荐且容易上手。python GPIO是一个小型的python库，可以帮助用户完成raspberry相关IO口操作，但是python GPIO库还没有支持SPI、I2C或者1-wire等总线接口。

    【官方网站】—— <https://code.google.com/p/raspberry-gpio-python/>

**2、wiringPi**

    【开发语言】——C语言

    【简单介绍】——wiringPi适合那些具有C语言基础，在接触树莓派之前已经接触过单片机或者嵌入式开发的人群。wiringPi的API函数和arduino非常相似，这也使得它广受欢迎。作者给出了大量的说明和示例代码，这些示例代码也包括UART设备，I2C设备和SPI设备等。

    【官方网站】—— <http://wiringpi.com/>

**3、BCM2835 C Library**

    【开发语言】——C语言

    【简单介绍】BCM2835 C Library可以理解为使用C语言实现的相关底层驱动，BCM2835 C Library的驱动库包括GPIO、SPI和UART等，可以通过学习BCM2835 C Library熟悉BCM2835相关的寄存器操作。如果有机会开发树莓派上的linux驱动，或自主开发python或PHP扩展驱动，可以从BCM2835 C Library找到不少的“灵感”。

    【官方网站】—— <http://www.airspayce.com/mikem/bcm2835/>

**二、树莓派GPIO编号方式**

**1、功能物理引脚：**

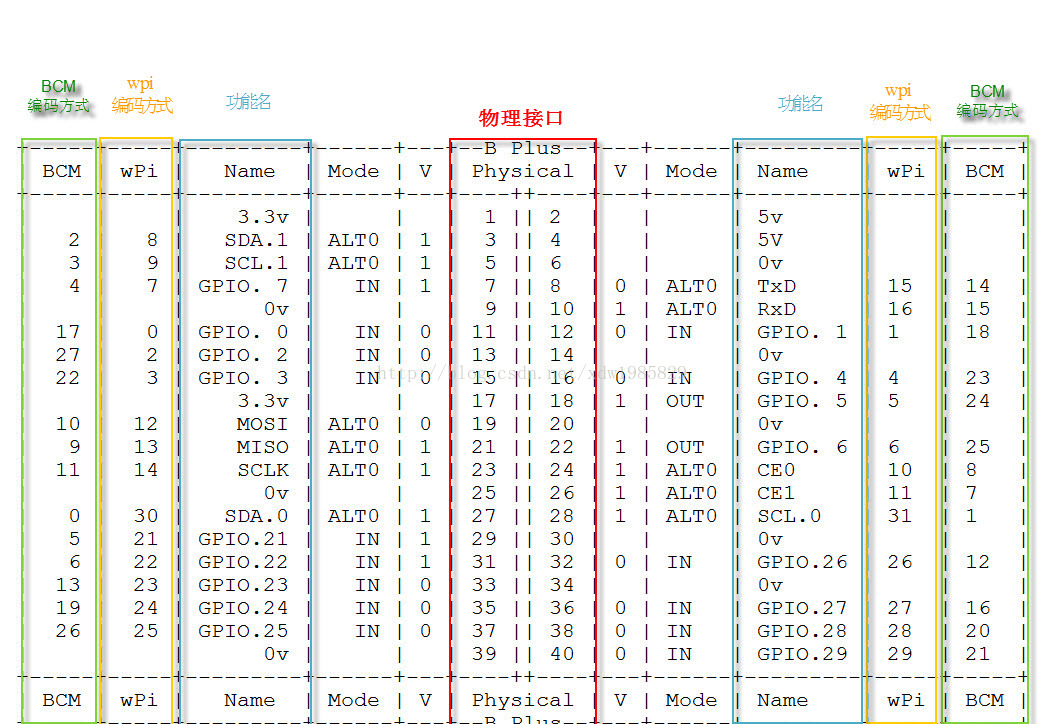
从左到右，从上到下：左边基数，右边偶数：1-40

**2、BCM:**

编号侧重CPU寄存器，根据BCM2835的GPIO寄存器编号。

**3、wpi：**

  编号侧重实现逻辑，把扩展GPIO端口从0开始编号，这种编号方便编程。正如图3 WiringPi一栏。

****

**三、python GPIO**

**1、先安装python-dev，输入以下指令。**

**sudo apt-get install python-dev**

**2、安装RPi.GPIO，依次输入以下指令。**

**1)下载：$ wget http://raspberry-gpio-python.googlecode.com/files/RPi.GPIO-0.5.3a.tar.gz**

**2)解压缩：$ tar xvzf RPi.GPIO-0.5.3a.tar.gz**

**3)进入解压之后的目录 ：$ cd RPi.GPIO-0.5.3a  
4)启动安装 ：$ sudo python setup.py install**  
  
**3、例子：**

**[python]** [view plaincopy](http://blog.csdn.net/xdw1985829/article/details/39580401)[在CODE上查看代码片](https://code.csdn.net/snippets/475804)

1. # -\*- coding: utf-8 -\*-
2. **import** RPi.GPIO as GPIO
3. **import** time
4. # BOARD编号方式，基于插座引脚编号
5. GPIO.setmode(GPIO.BOARD)
6. # 输出模式
7. GPIO.setup(11, GPIO.OUT)
9. **while** True:
10. GPIO.output(11, GPIO.HIGH)
11. time.sleep(1)
12. GPIO.output(11, GPIO.LOW)
13. time.sleep(1)

**4、执行：**  
sudo python led.py  
  
**5、说明：**  
1)GPIO.setmode(GPIO.BOARD)，采用插座引脚编号方式。  
2)由于采用插座引脚编号方式，此处的11脚相当于BCM2835寄存器编号方式的引脚11。  
  
**四、python GPIO  
1、说明：**    WiringPi是应用于树莓派平台的GPIO控制库函数，WiringPi遵守GUN Lv3。wiringPi使用C或者C++开发并且可以被其他语言包转，例如python、ruby或者PHP等。  
    wiringPi包括一套gpio控制命令，使用gpio命令可以控制树莓派GPIO管脚。用户可以利用gpio命令通过shell脚本控制或查询GPIO管脚。wiringPi是可以扩展的，可以利用wiringPi的内部模块扩展模拟量输入芯片，可以使用MCP23x17/MCP23x08（I2C 或者SPI）扩展GPIO接口。另外可通过树莓派上的串口和Atmega（例如arduino等）扩展更多的GPIO功能。另外，用户可以自己编写扩展模块并把自定义的扩展模块集成到wiringPi中。WiringPi支持模拟量的读取和设置功能，不过在树莓派上并没有模拟量设备。但是使用WiringPi中的软件模块却可以轻松地应用AD或DA芯片。

**2.wiringPi安装**

**1)方案A——使用GIT工具**

通过GIT获得wiringPi的源代码

git clone git://git.drogon.net/wiringPi  
cd wiringPi

./build

build脚本会帮助你编译和安装wiringPi  
  
**2)方案B——直接下载**  
我们可以在https://git.drogon.net/?p=wiringPi;a=summary网站上直接下载最新版本编译使用

tar xfz wiringPi-xx.tar.gz

cd wiringPi-xx

./build  
  
**3、测试：**

wiringPi包括一套gpio命令，使用gpio命令可以控制树莓派上的各种接口，通过以下指令可以测试wiringPi是否安装成功。  
**$gpio -v  
$gpio readall**  
即可出现上面的io图 **4、例子：**

**[cpp]** [view plaincopy](http://blog.csdn.net/xdw1985829/article/details/39580401)[在CODE上查看代码片](https://code.csdn.net/snippets/475804)

1. #include <wiringPi.h>
2. **int** main(**void**)
3. {
4. wiringPiSetup() ;
5. pinMode (0, OUTPUT) ;
6. **for**(;;)
7. {
8. digitalWrite(0, HIGH) ; delay (500) ;
9. digitalWrite(0,  LOW) ; delay (500) ;
10. }
11. }

**5、编译运行：**  
在树莓派上:  
gcc -Wall -o test test.c -lwiringPi   
sudo ./test  
  
在虚拟机中：  
am-linux-gcc -Wall -o test test.c -lwiringPi   
sudo ./test  
  
**6、注意事项：**  
1）IO的编号方式略有不同，采用wiring编码方式。  
2）-lwiringPi表示动态加载wiringPi共享库。  
  
**五、BCM2835 C Library**  
  
**1、下载**:               $ wget http://www.airspayce.com/mikem/bcm2835/bcm2835-1.35.tar.gz

**2、解压缩**:             $tar xvzf bcm2835-1.35.tar.gz  
**3、进入压缩之后的目录**:$cd bcm2835-1.35  
**4、 配置**:              $./configure  
**5、从源代码生成安装包**:$make  
**6、执行检查**:           $sudo make check  
**7、安装 bcm2835库**:    $sudo make install

**8、例子**

**[cpp]** [view plaincopy](http://blog.csdn.net/xdw1985829/article/details/39580401)[在CODE上查看代码片](https://code.csdn.net/snippets/475804)

1. #include <bcm2835.h>
3. // P1插座第11脚
4. #define PIN RPI\_GPIO\_P1\_11
6. **int** main(**int** argc, **char** \*\*argv)
7. {
8. **if** (!bcm2835\_init())
9. **return** 1;
11. // 输出方式
12. bcm2835\_gpio\_fsel(PIN, BCM2835\_GPIO\_FSEL\_OUTP);
14. **while** (1)
15. {
16. bcm2835\_gpio\_write(PIN, HIGH);
17. bcm2835\_delay(100);
19. bcm2835\_gpio\_write(PIN, LOW);
20. bcm2835\_delay(100);
21. }
22. bcm2835\_close();
23. **return** 0;
24. }

**9、注意事项：**  
1）IO的编号方式略有不同，采用wiring编码方式。  
2）-lwiringPi表示动态加载wiringPi共享库。 **六、文章参考以下链接  
http://elinux.org/RPi\_Low-level\_peripherals  
http://blog.csdn.net/xukai871105/article/details/23115627**

[树莓派开发系列教程10——树莓派spi液晶屏支持(fbtft)](http://blog.csdn.net/xdw1985829/article/details/39583239) **树莓派官方支持av及HDMI输出，板子上预留了一个csi接口的液晶显示屏，但是一直没有相应的模组出现。在很多应用场合我们需要一些小型的液晶屏显示一些基本的信息，所以小屏驱动很是必要。  
  
    在github上有一个开源工程：notro/fbtft，完整的实现了framebuffer驱动，让树莓派完美支持tft液晶，下面对移植过程进行一个简单说明  
  
一、官网地址  
  
工程首页：https://github.com/notro  
  
fbtft源码：https://github.com/notro/fbtft  
  
编译好的固件（基于3.12.25+）:https://github.com/notro/rpi-firmware  
  
使用说明(wiki)：https://github.com/notro/fbtft/wiki**  
  
**二、使用编译好的固件(3.12.25+)**  
  
环境：树莓派  
  
https://github.com/notro/rpi-firmware

**1、打开SPI**

树莓派默认spi是关掉的，我们需要打开

sudo vi /etc/modprobe.d/raspi-blacklist.conf

把下面这句话前面的#号删掉

blacklist spi-bcm2708  
  
**2、下载：**  
  
1)以模块的形式编译进内核（需要手动或脚本加载模块）3.12.25+（**试验成功**）  
sudo REPO\_URI=https://github.com/notro/rpi-firmware rpi-update

2)直接编译进内核（**笔者没有试验**）  
sudo REPO\_URI=https://github.com/notro/rpi-firmware BRANCH=builtin rpi-update  
  
3)以模块的形式编译进内核（**需要手动或脚本加载模块，最新版本，笔者试过启动不起来，不知道哪出问题**）  
  
sudo REPO\_URI=https://github.com/notro/rpi-firmware BRANCH=latest rpi-update  
  
4)直接下载压缩包，手动安装（**适合树莓派不能联网的时候**）  
  
http://tronnes.org/downloads/2014-06-20-wheezy-raspbian-2014-07-25-fbtft-master-firmware.zip

**3、配置**

**1)手动加载模块：**

sudo modprobe fbtft\_device name=adafruit22

    name后面的名字，要跟相应的液晶驱动芯片移植

    笔者使用的液晶芯片为：**fb\_ra8875，所以这里写的是：er\_tftm050\_2**

    其它芯片请查阅：https://github.com/notro/fbtft/blob/master/fbtft\_device.c 文件  
  
正常会提示以下信息  
  
fbtft\_device:  SPI devices registered:  
   fbtft\_device:      spidev spi0.0 500kHz 8 bits mode=0x00  
   fbtft\_device:      spidev spi0.1 500kHz 8 bits mode=0x00  
   fbtft\_device:  'fb' Platform devices registered:  
   fbtft\_device:      bcm2708\_fb id=-1 pdata? no  
   fbtft\_device: Deleting spi0.0  
   fbtft\_device:  GPIOS used by 'adafruit22':  
   fbtft\_device:    'reset' = GPIO25  
   fbtft\_device:    'led' = GPIO23  
   fbtft\_device:  SPI devices registered:  
   fbtft\_device:      spidev spi0.1 500kHz 8 bits mode=0x00  
   fbtft\_device:      fb\_hx8340bn spi0.0 32000kHz 8 bits mode=0x00

   graphics fb1: fb\_hx8340bn frame buffer, 176x220, 75 KiB video memory, 16 KiB buffer memory, fps=20, spi0.0 at 32 MHz

**在/dev/目录下出现: /dev/fb1设备**

**2）自动加载模块**

sudo vi  /etc/modules

加入以下语句，既可以在启动时自动加载模块

spi-bcm2708

fbtft\_device name=er\_tftm050\_2  **speed=28000000 fps=25** verbose=0

红色部分根据实际情况调整，可能出现花屏现象

**4、使用（官方给出的方法，笔者测试不成功）**  
  
**1)手动启动x11和控制台到新的液晶屏**  
X Windows显示在fb1上：  
$FRAMEBUFFER=/dev/fb1 startx  
  
Console显示在fb1上：    
$con2fbmap 1 1  
  
**2)自动登陆x11**  
  
**sudo vi /etc/inittab**  
    #1:2345:respawn:/sbin/getty --noclear 38400 tty1  
    1:2345:respawn:/bin/login -f pi tty1 </dev/tty1 >/dev/tty1 2>&1

**sudo vi /etc/rc.local**

    su -l pi -c "env FRAMEBUFFER=/dev/fb1 startx &"  
 **5、使用（笔者使用这个测试通过）**  
  
**1)将fb0上的内容直接拷贝到fb1上,fb0和fb1同步**

**https://github.com/notro/fbtft/wiki/Framebuffer-use#framebuffer-mirroring**

$git clone https://github.com/tasanakorn/rpi-fbcp

$cd rpi-fbcp/  
$mkdir build  
$cd build/  
$cmake ..  
$make  
$sudo install fbcp /usr/local/bin/fbcp

**启动**：fbcp &

**关闭fbcp**：killall fbcp  
  
**2)启动时使用fb1**  
  
$sudo apt-get install xserver-xorg-video-fbdev  
  
$sudo vi /usr/share/X11/xorg.conf.d/99-fbdev.conf

加入以下语句：

Section "Device"    
  Identifier "myfb"  
  Driver "fbdev"  
  Option "fbdev" "/dev/fb1"  
EndSection

**启动：**startx

**测试：**

apt-get -y install fbi

fbi -d /dev/fb1 -T 1 -noverbose -a test.jpg

**三、由内核及源码编译**

**1、下载、编译内核源码：**

**请见《**[**树莓派开发系列教程8——树莓派内核编译与固件升级**](http://blog.csdn.net/xdw1985829/article/details/39077611)**》**

**2、下载、编译fbtft源码**

**$cd linux(进入下载好的内核源码目录)**

**$cd drivers/video**

**$git clone https://github.com/notro/fbtft.git（下载fbtft源码，也可以在别的地方下载好，拷贝过来）**

**修改内核源码的Kconfig及Makefine**

**Add to drivers/video/Kconfig:   source "drivers/video/fbtft/Kconfig"**

**Add to drivers/video/Makefile:  obj-y += fbtft/**

**$make menuconfig(在配置界面加入所选用液晶的驱动支持)**

**[plain]** [view plaincopy](http://blog.csdn.net/xdw1985829/article/details/39583239)

1. Device Drivers  --->
2. Graphics support  --->
3. <M> Support for small TFT LCD display modules  --->
5. <M>   FB driver for the HX8353D LCD Controller
6. <M>   FB driver for the ILI9320 LCD Controller
7. <M>   FB driver for the ILI9325 LCD Controller
8. <M>   FB driver for the ILI9340 LCD Controller
9. <M>   FB driver for the ILI9341 LCD Controller
10. < >     FB driver for the ILI9481 LCD Controller
11. <M>   FB driver for the ILI9486 LCD Controller
12. <M>   FB driver for the PCD8544 LCD Controller
13. <M>   FB driver for the RA8875 LCD Controller

$make

**请见《**[**树莓派开发系列教程8——树莓派内核编译与固件升级**](http://blog.csdn.net/xdw1985829/article/details/39077611)**》**