

---

# SC6138 Linux 开发环境用户指南

文档版本	0.1.0
发布日期	2015-3-23

版权所有 © 杭州士兰微电子股份有限公司 2014 - 2015

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 概述

本文档主要介绍以下内容：

- Linux 开发软硬件环境的搭建
- bootloader、Linux 内核和根文件系统的烧写
- 应用程序开发要点

## 产品版本

产品名称	版本
SC6138	V100

## 读者对象

本文档主要适用于以下人员：

- 技术支持工程师
- 软件开发工程师

## 修订记录

版本	日期	修订说明
0.1.0	2015-3-23	初次创建

# 目 录

1	开发环境 .....	- 1 -
1.1	连接宿主机与目标机 .....	- 1 -
1.1.1	硬件连接 .....	- 1 -
1.1.2	IP 地址分配 .....	- 1 -
1.2	搭建 Linux 开发环境.....	- 2 -
1.2.1	安装 Linux 主机.....	- 2 -
1.2.2	新建管理组用户 .....	- 2 -
1.2.3	连接配置网络 .....	- 2 -
1.2.4	连接配置串口 .....	- 3 -
1.2.5	安装配置 TFTP 服务器.....	- 4 -
1.2.6	安装配置 NFS 服务器 .....	- 5 -
1.2.7	安装交叉编译工具链 .....	- 5 -
1.2.8	安装 SC6138 SDK.....	- 6 -
2	Bootloader .....	- 6 -
2.1	UBOOT .....	- 6 -
3	Linux 内核.....	- 6 -
3.1	内核镜像 .....	- 7 -
3.2	打印信息 .....	- 7 -
3.2.1	根文件系统 NFS 挂载 .....	- 7 -
4	根文件系统（rootfs） .....	- 7 -
4.1	主机上部署根文件系统 .....	- 7 -
4.2	initramfs 制作.....	- 7 -
5	烧写启动镜像和根文件系统 .....	- 8 -
5.1	烧写 bootloader .....	- 8 -
5.2	烧写 initramfs 根文件系统镜像 .....	- 8 -
6	应用程序开发 .....	- 8 -
6.1	编写代码 .....	- 9 -
6.2	编译代码 .....	- 9 -
6.3	运行应用程序 .....	- 9 -
6.3.1	应用程序添加到根文件系统 .....	- 9 -
6.3.2	应用程序自动启动 .....	- 9 -

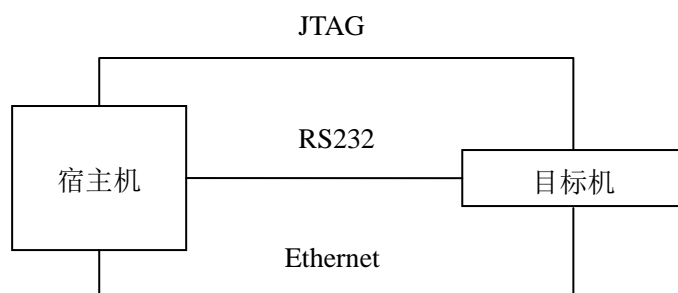
## 图表目录

图表 1-1 宿主机目标机连接示意图 .....	- 1 -
图表 1-2 SC6138 Demo 板硬件连接图 .....	- 1 -
图表 1-3 UART-RS232 转换板与 Demo 板连接示意图 .....	- 4 -
图表 1-4 串口配置 .....	- 4 -

# 1 开发环境

## 1.1 连接宿主机与目标机

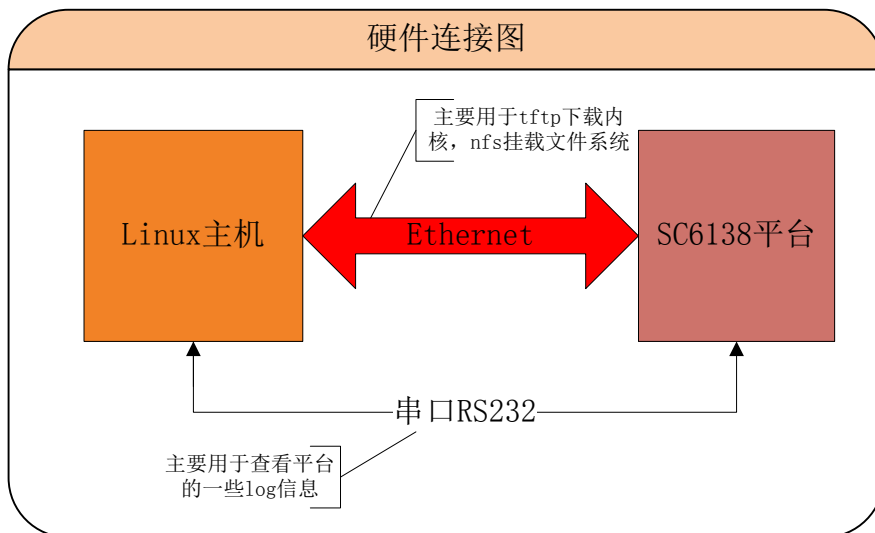
目标机（Demo 板）软件的开发在宿主主机上进行，即采用“宿主机+目标机”的形式。除了开发裸机程序时常用的 JTAG 连接与串口连接，通常增加以太网连接以方便 Linux 程序的开发，如图 1-1 所示。



图表 1-1 宿主机目标机连接示意图

### 1.1.1 硬件连接

对于 Silan Demo 板，bootloader 已经烧写完毕，因此开发 Linux 应用时不再需要 JTAG，连接图如下：



图表 1-2 SC6138 Demo 板硬件连接图

### 1.1.2 IP 地址分配

宿主机（以下简称主机）的默认 IP：192.168.1.2。

目标机（以下另称为 Demo 板）的默认 IP：boot 阶段为 192.168.1.5，Linux 运行阶段为 192.168.1.4。

## 1.2 搭建 Linux 开发环境

### 1.2.1 安装 Linux 主机

建议选择常用的 Linux 发行版，便于获得各类技术资源，比如：

- Ubuntu

**注：本指南后续内容描述以 Ubuntu 11.04 为原型。**

在安装时，请选择 EXT4 作为根文件系统。

### 1.2.2 新建管理组用户

新建一个管理组用户用作开发。步骤如下：

#### （1）新建用户 sc6138

```
$ sudo adduser sc6138
```

然后根据提示输入密码，完成创建。

#### （2）将用户添加到 sudo 组

```
$ sudo usermod -aG admin sc6138
```

#### （3）登录 sc6138

切换到 sc6138 用户：

```
$ su sc6138
```

或者重启，然后以 sc6138 登录。

### 1.2.3 连接配置网络

#### 1.2.3.1 连接网络

为方便调试，主机与 Demo 板需要处于同一局域网内。局域网的搭建有下述两种方案。

##### （1）方案 A：主机与 Demo 板以网线直连

这种办法适用于主机不需要访问外网的场合。当前基本所有网卡都是收发自适应的，所以使用直连线即可。

## (2) 方案 B: 主机与 Demo 连接到同一 HUB 上, HUB 的上行端口与路由器相连

这种办法适用于主机需要访问外网的场合。路由器需使用 NAT 功能将主机在局域网内的静态 IP(192.168.1.2) 转换为外部 IP。若路由器支持多 IP 地址映射, 则不需要 HUB (将 192.168.1.\*网段内的包在局域网内路由, 将 192.168.1.\*网段外的包向外路由)。

### 1.2.3.2 配置主机 IP

#### (1) 打开配置文件/etc/network/interfaces

```
$ vim /etc/network/interfaces
```

#### (2) 在文件最后增加 eth0 配置信息 (静态 IP)

加入如下配置:

```
auto eth0
iface eth0 inet static
address 192.168.1.2
broadcast 192.168.1.255
netmask 255.255.255.0
gateway 192.168.1.1
```

#### (3) 重启网络服务使网络配置文件生效

```
$sudo /etc/init.d/networking restart
```

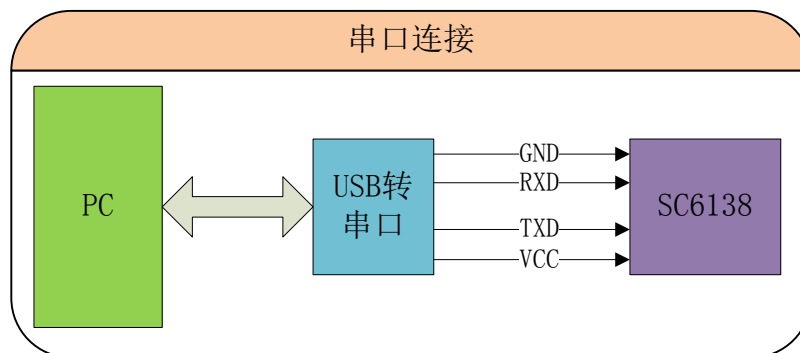
## 1.2.4 连接配置串口

### (1) 连接串口

若主机没有 RS232 接口, 则可以使用 USB-RS232 转换线扩展串口。在主机上, 其设备节点通常为/dev/ttyUSB0。

对于 Silan Demo 板, 为减小 PCB 面积, UART 未转成 RS232 接口, 且只引出了 GND, RxD, TxD, VCC 四个端口。因此, 需要 Silan 提供的 UART- RS232 转接板。所以对于 Silan Demo 板, 串口连接如下:

主机 --> USB-RS232 转换线 --> Silan Demo 板



图表 1-3 UART-RS232 转换板与 Demo 板连接示意图

## (2) 启动主机上串口工具

串口工具可以使用 putty 或 minicom，启动方法如下：

```
$ putty&
$ minicom
```

## (3) 配置串口

串口配置如下表所示：

图表 1-4 串口配置

波特率	数据位	停止位	奇偶校验	流控制
115200	8	1	0	XON/XOFF

## 1.2.5 安装配置 TFTP 服务器

TFTP 主要用于在 boot 阶段将内核镜像从主机下载到 Demo 板。安装配置步骤如下：

### (1) 安装 TFTP 服务器（若尚未安装）

通过网络安装（若主机能上外网）：

```
$ sudo apt-get install xinetd
$ sudo apt-get install tftp tftpd
```

若不能通过网络安装，则请下载安装包手动安装。

### (2) 设置 TFTP 默认路径并修改权限

在 Ubuntu 上，TFTP 配置文件是/etc/xinetd.d/tftp，打开该文件将参数“server\_args”配置如下：

```
server_args = -s /tftpboot
```

以上配置将/tftpboot 设置为默认目录，用户也可以将其它目录设为默认目录。

修改该目录的权限：

```
$ sudo chmod 777 /tftpboot
```

### (3) 启动/重启 TFTP 服务器

启动：

```
$ /etc/init.d/xinetd start
```

重启：



```
$ /etc/init.d/xinetd restart
```

## 1.2.6 安装配置 NFS 服务器

NFS 用于挂载根文件系统。安装配置步骤如下：

### (1) 安装 NFS 服务器（若尚未安装）

```
$ sudo apt-get install nfs-kernel-server
```

### (2) 配置

NFS 允许挂载的目录及权限，在文件/etc/exports 中进行定义。

```
$ sudo vim /etc/exports
```

加入以下内容：

```
/home/sc6138 *(rw,insecure,sync,no_subtree_check,no_root_squash)
```

### (3) 重启服务

```
$ sudo /etc/init.d/portmap restart          <-- 重启 portmap,  
$ sudo /etc/init.d/nfs-kernel-server restart <-- 重启 nfs 服务  
$ showmount -e                             <-- 显示共享出的目录
```

注：关于 tftp 和 nfs 环境的搭建可以参考以下博文。

<http://blog.csdn.net/eastmoon502136/article/details/7905960>

## 1.2.7 安装交叉编译工具链

请按照以下步骤安装工具链：

### (1) 将工具链解压到安装目录

```
$ cp csky-linux-x86_32.tar.gz /home/sc6138/tools  
$ cd /home/sc6138/tools  
$ tar -xf csky-linux-x86_32.tar.gz
```

### (2) 修改当前用户环境变量 PATH 的值

```
$ vim ~/.bashrc
```

在最后一行中添加：

```
export PATH=$PATH:/home/sc6138/tools/csky-linux/bin
```

(3) 运行脚本使配置生效

```
$ source ~/.bashrc
```

## 1.2.8 安装 SC6138 SDK

SC6138 SDK (sllib) 是基于 Silan SDK 框架的应用程序开发包，包含了 Linux 应用程序开发时所使用库及源代码。将 Silan SDK 压缩包解压至工作目录即完成安装：

```
$ cp SC6138_SDKxxx_Release.rar /home/sc6138
$ cd /home/sc6138
$ tar -xf SC6138_SDKxxx_Release.rar
```

SDK 的使用和详细说明，请参考文档《SC6138\_SDK 用户手册》。

# 2 Bootloader

## 2.1 UBOOT

UBOOT 是对 CPU 的引导程序。

u-boot.bin 的主要职责是完成通用寄存器、时钟和 DDR 控制器初始化，拷贝引导程序至内存，以及 C 运行环境的初始化，并完成 Linux 内核的装载和引导。此外，还包含 SPI FLASH 烧写、ping、TFTP、内存测试等一系列工具，以及对 FAT 和 EXT4 文件系统的支持。

(1) 烧写地址与大小

u-boot.bin 在 SPI FLASH 中的烧写地址：0x00000；文件大小不超过 128KB。

# 3 Linux 内核

发布已经编译好的内核镜像 vmlinux.bin，其中包含了 TCP/IP 协议栈、EXT4/NFS 文件系统等功能块，以及 DRM、DSP 等硬件模块驱动程序。

## 3.1 内核镜像

### (1) 加载地址

Linux 内核的内存加载地址：0x80000000。

### (2) 不同功用的二进制镜像

Image：通过 NFS 挂载主机上的根文件系统。该镜像一般用于调试。

## 3.2 打印信息

### 3.2.1 根文件系统 NFS 挂载

若内核成功启动并挂载主机上的根文件系统，

则可以再串口终端输入你需要的命令，运行你需要运行的程序

## 4 根文件系统（rootfs）

Silan 发布已经制作好的根文件系统，其中除了存放 Linux 系统使用的库、配置文件、应用程序及其它需要用到的服务，还存放了从 Silan SDK 生成的库文件（存放在/usr/lib 目录下）和例子程序（存放在/usr/bin 目录下）。

### 4.1 主机上部署根文件系统

在主机上部署根文件系统的主要目的是为了更方便调试，即 Demo 板根文件系统的任何更新都可以在主机上完成（将文件拷贝到 Demo 板挂载目录下相应的子目录即可）。部署流程如下：

#### (1) 将根文件系统压缩包拷贝到主机工作目录下

```
$ cd /home/sc6138/release (sdk 中已经有文件系统的压缩包)
```

#### (2) 将压缩包解压

```
$ cd /home/sc6138/release  
$ tar -xf rootfs.tar.gz
```

在内核镜像 Image 中修改.config 中的 CMDLINE，主机上的挂载目录即为/home/sc6138/release/rootfs。

### 4.2 initramfs 制作

制作 initramfs，可以烧录到 flash 中。

## （1）内核配置和编译

Kernel 目录下执行

```
$ make sc6138_flash_defconfig
```

再配置文件系统路径

```
$ make menuconfig
```

General setup--->

**[\*] Initial RAM filesystem and RAM disk (initramfs/initrd) support**

**(/rootfs\_dir) Initramfs source file(s)**

其中 rootfs\_dir 就是我们需要压缩的 rootfs 的路径。

## （2）最后 make 编译镜像：

```
make uImage
```

之后在 arch/csky/boot/下就会生成 uImage，把 uImage 拷贝到 SD 卡上就可以升级系统。

# 5 烧写启动镜像和根文件系统

## 5.1 烧写 bootloader

uboot.bin 的烧写到 Flash 从地址 0 开始烧写 128Kbyte 大小即可。

## 5.2 烧写 initramfs 根文件系统镜像

uImage 是制作好的 initramfs 镜像，烧录流程如下：

- （1）将 uImage 拷贝到 SD 第一个分区根目录下，并且要求该分区文件系统格式为 FAT。
- （2）上电复位自动识别是否有 uImage 文件，有的话就直接进入 SD 卡升级根文件系统模式。
- （3）升级完成后自动加载并启动新的根文件系统。

# 6 应用程序开发

代码在编写与编译在主机上进行，生成的二进制镜像在 Demo 板上运行。

## 6.1 编写代码

用户可根据个人习惯选择平台和代码编写工具。在 Linux 环境下可以使用 vim+ctags+cscope 或 EMACS；在 Windows 环境下通常使用 Source Insight，但若需要在 Linux 下编译，建议首先将源文件转化为 UNIX 格式（使用 dos2unix）。

若在 Silan SDK 基础上进行开发，则可以使用 SDK 提供的 API。详细信息请参考《SC6138\_SDK 用户手册》。

## 6.2 编译代码

在 Linux 下编译代码，我们通常需要使用 GNU make 并编写 Makefile。若在 Silan SDK 基础上进行开发，则可以使用 SDK 提供 Makefile 及配置文件模板。详细信息请参考《SC6138\_SDK 用户手册》。

## 6.3 运行应用程序

### 6.3.1 应用程序添加到根文件系统

要运行编译好的应用程序，首先需要将其添加到根文件系统中。

若根文件系统采用 NFS 挂载，则在主机上直接将相关文件拷贝到 Demo 板挂载目录下相应的子目录即可。

通常将应用程序放到/usr/bin 目录下，库文件放到/usr/lib 目录下（若是系统库文件，则放到/lib 下），配置文件则放到/etc 目录下。

### 6.3.2 应用程序自动启动

如果新添加的应用程序需要系统启动后自动运行，可以编辑/etc/init.d/rcS 文件，添加需要启动的应用程序路径。

## 附录:

### 1、在 Windows 主机上搭建 Linux 开发环境

写在环境搭建的最前面:

开发环境是开发人员在开发过程当中, 所需的软硬件。开发环境并不是一个固定的样式, 在这里, 我们详细讲解一个嵌入式 Linux 开发环境搭建的方法。您已经对嵌入式开发非常了解的话, 可以按照自己的需求来搭建环境。如果和本手册环境不一样而产生报错, 您可以从国内一些大 Linux 论坛和网站搜索相关的信息来解决。

Ubuntu 是一个以桌面应用为主的 Linux 操作系统。Ubuntu 拥有很多优点。相对于其他版本的 Linux, Ubuntu 也有着自己的优势。首先, 安装系统非常简单, 只需要非常少的设置即可, 完全可以和 Windows 桌面系统想媲美; 其次, 图形界面很人性化, 模仿了在 xp 下常用的快捷键; 还有, 安装和升级程序时, 可以通过网络, 由系统自行安装依赖的文件包, 从此不必再为 Linux 系统的依赖关系大伤脑筋。综合考虑大家的使用习惯和学习的需要, 我们选用 Ubuntu Linux。

Linux 桌面系统版本众多, 目前所有实验和源码在 Ubuntu10.10 版本测试可以通过。使用其他版本 Linux 桌面系统, 可能会出现 gcc 编译器和库文件相关的问题。碰到类似问题, 可以在 Linux 系统发行商的官方论坛上咨询和查询。

Ubuntu 官网: <http://www.ubuntu.org.cn>

可下载各个版本的 Ubuntu。当然, 官网能找到各种关于 Ubuntu 的信息。

Ubuntu 官方论坛: <http://forum.ubuntu.org.cn/>

可以找到大量的实用的 Ubuntu 资源, 中文论坛。Ubuntu 也有官方的英文论坛。

### 2、在 Windows 系统中使用虚拟机搭建开发环境虚拟机

随着 PC 机主频速度的不断进步以及内存价格的持续降低, 在一台 PC 上运行一个或多个虚拟机已变得越来越轻松。几年前, 开发或学习 Linux 系统应用需要额外安装一台独立的主机, 但目前已经不需要那样做了, 我们可以在 Windows 系统上直接运行一个虚拟 Linux 机, 就可以很方便的使用 Linux 系统。这样做的好处是显而易见的, 可以在 Windows 和 Linux 之间快速切换, 也可以在两者之间共享文件, 为学习和开发 Linux 系统带来了很大的便利。

VMware Workstation 是一款功能强大的桌面虚拟计算机软件, 使得用户可在单一的桌面上同时运行不同的操作系统, 是进行开发、测试、部署新的应用程序的最佳解决方案。VMware Workstation 可在一部实体机器上模拟完整的网络环境, 以及可便于携带的虚拟机器, 其更好的灵活性与先进的技术胜过了市面上其他的虚拟计算机软件。对于企业的 IT 开发人员和系统管理员而言, VMware 在虚拟网路, 实时快照, 拖曳共享文件夹, 支持 PXE 等方面的特点使它成为必不可少的工具。

在 Windows XP 中安装 VMware Workstation

提供了一个 VMware-workstation-full-9.0.2 版本的安装程序, 将 VMware7.rar 拷贝到硬盘上解压, 然后双击 'VMware-workstation-full-9.0.2-1031769.exe' 来安装。安装过程比较简单, 这里将相关步骤从略。虚拟机安装好以后, 就可以在虚拟机中安装相应的系统软件了, 比如 Linux、Windows 等等。这样就能在同一台机器上使用多个系统了。

### 3、Linux 常用命令详解

Linux 是一个真正的多用户操作系统，它可以同时接受多个用户登录。Linux 还允许一个用户进行多次登录，这是因为 Linux 和 UNIX 一样，提供了虚拟控制台的访问方式，允许用户在同一时间从控制台进行多次登录。虚拟控制台的选择可以通过按下 Alt 键和一个功能键来实现，通常使用 F1 - F6 例如，用户登录后，按一下 Alt - F2 键，用户又可以看到"login:"提示符，说明用户看到了第二个虚拟控制台。然后只需按 Alt - F1 键，就可以回到第一个虚拟控制台。一个新安装的 Linux 系统默认允许用户使用 Alt - F1 到 Alt - F6 键来访问前六个虚拟控制台。虚拟控制台可使用户同时在多个控制台上工作，真正体现 Linux 系统多用户的特性。用户可以在某一虚拟控制台上进行的工作尚未结束时，切换到另一虚拟控制台开始另一项工作。当然我们也可以在 KDE 环境下使用终端方式输入命令。常见的命令如下：

文件列表 - ls

ls #以默认方式显示当前目录文件列表

ls -a #显示所有文件包括隐藏文件

ls -l #显示文件属性，包括大小，日期，符号连接，是否可读写及是否可执行

ls -color=never \*.so > obj #不显示文字颜色,将所有 so 文件记录到 obj 文件中

目录切换 - cd

cd dir #切换到当前目录下的 dir 目录

cd / #切换到根目录

cd .. #切换到上一级目录

cd ../../ #切换到上二级目录

cd ~ #切换到用户目录，比如是 root 用户，则切换到/root 下

删除 - rm

rm file #删除某一个文件

rm -fr dir #删除当前目录下叫 dir 的整个目录

复制 - cp

cp source target #将文件 source 复制为 target

cp /root/source . #将/root 下的文件 source 复制到当前目录

cp -av soure\_dir target\_dir #将整个目录复制，两目录完全一样

`cp -fr source_dir target_dir` #将整个目录复制，并且是以非链接方式复制，当 `source` 目录带有符号链接时，两个目录不相同

移动 - `mv`

`mv source target` #将文件 `source` 更名为 `target`

比较 - `diff`

`diff dir1 dir2` #比较目录 1 与目录 2 的文件列表是否相同，但不比较文件的实际内容，不同则列出

`diff file1 file2` #比较文件 1 与文件 2 的内容是否相同，如果是文本格式的文件，则将不相同的内容显示，如果是二进制代码则只表示两个文件是不同的

`comm file1 file2` #比较文件，显示两个文件不相同的内容

回显 - `echo`

`echo message` #显示一串字符

`echo "message message2"` #显示不连续的字符串

文件内容查看 - `cat`

`cat file` #显示文件的内容，和 DOS 的 `type` 相同

`cat file | more` #显示文件的内容并传输到 `more` 程序实现分页显示，使用命令 `less file` 可实现相同的功能

`more` #分页命令，一般通过管道将内容传给它，如 `ls | more`

设置环境变量 - `export`

`export LC_ALL=zh_CN.GB2312` #将环境变量 `LC_ALL` 的值设为 `zh_CN.GB2312`

`export DISPLAY=0:0` #通过该设置，当前字符终端下运行的图形程序可直接运行于 Xserver

时间日期 - `date`

`date` #显示当前日期时间

`date -s 20:30:30` #设置系统时间为 20:30:30

版权所有 © 杭州士兰微电子股份有限公司



date -s 2002-3-5 #设置系统时期为 2003-3-5

clock -r #对系统 Bios 中读取时间参数

clock -w #将系统时间(如由 date 设置的时间)写入 Bios

容量查看 - du

du #计算当前目录的容量

du -sm /root #计算/root 目录的容量并以 M 为单位

查找 - find

find -name /path file #在/path 目录下查找看是否有文件 file

搜索 - grep

grep -ir "chars" #在当前目录的所有文件查找字符串 chars, 并忽略大小写, -i 为大写, -r 为下一级目录

编辑 - vi

vi file #编辑文件 file

vi 原基本使用及命令:

输入命令的方式为先按 ctrl+c, 然后输入:x(退出),:x!(退出并保存):w(写入文件),:w!(不询问方式写入文件), :r file(读文件 file), :%s/oldchars/newchars/g(将所有字符串 oldchars 换成 newchars)这一类的命令进行操作

读取 - man

man ls #读取关于 ls 命令的帮助

man ls | grep color #读取关于 ls 命令的帮助并通过 grep 程序在其中查找 color 字符串

重启 - reboot

reboot #重新启动计算机

关机 - halt

halt #关闭计算机

压缩与解压 - tar

tar xfv file.tgz #将文件 file.tgz 解压

tar xfv file.tgz -C target\_path #将文件 file.tgz 解压到 target\_path 目录下

tar cvf file.tgz source\_path #将文件 source\_path 压缩为 file.tgz

tar c directory > directory.tar #将目录 directory 打包成不压缩的 directory.tar

gzip directory.tar #将覆盖原文件生成压缩的 directory.tar.gz

gunzip directory.tar.gz #覆盖原文件解压生成不压缩的 directory.tar。

tar xf directory.tar #可将不压缩的文件解包

权限设置 - chmod

chmod a+x file #将 file 文件设置为可执行，脚本类文件一定要这样设置一个，否则得用 bash file 才能执行

chmod 666 file #将文件 file 设置为可读写

chown user /dir #将/dir 目录设置为 user 所有

网卡配置 - ifconfig

ifconfig eth0 192.168.1.1 netmask 255.255.255.0 #设置网卡 1 的地址 192.168.1.1，

掩码为 255.255.255.0，不写 netmask 参数则默认为 255.255.255.0

ifconfig eth0:1 192.168.1.2 #捆绑网卡 1 的第二个地址为 192.168.1.2

ifconfig eth0:x 192.168.1.x #捆绑网卡 1 的第二个地址为 192.168.1.x

ifconfig down eth1 #关闭第二块网卡，使其停止工作

创建设备 - mknod

mknod /dev/hda1 b 3 1 #创建块设备 hda1，主设备号为 3，从设备号为 1，即 master 硬盘的的第一个分区

mknod /dev/tty1 c 4 1 #创建字符设备 tty1,主设备号为 4，众设备号为 1，即第一个 tty 终端

装载模块 - insmod

insmod rtl8139.o #装载驱动程序 rtl8139.o

版权所有 © 杭州士兰微电子股份有限公司

insmod sb.o io=0x280 irq=7 dma=3 dma16=7 mpu\_io=330 #装载驱动程序并设置相关的  
irq,dma 参数

删除模块 - rmmod

rmmod rtl8139 #删除名为 rtl8139 的驱动模块

挂接 - mount

mount -t ext2 /dev/hda1 /mnt #把/dev/hda1 装载到 /mnt 目录

mount -t iso9660 /dev/cdrom /mnt/cdrom #将光驱加载到/mnt/cdrom 目录

mount -t smb //192.168.1.5/sharedir /mnt -o username=fangtan,password=fangtan

#将 Windows 的共享目录加载到/mnt/smb 目录，用户名及密码均为 fangtan

mount -t nfs 192.168.1.1:/sharedir /mnt #将 nfs 服务的共享目录 sharedir 加载到  
/mnt/nfs 目录

卸载 - umount

umount /mnt #将/mnt 目录卸载，/mnt 目录必须处于空闲状态

umount /dev/hda1 #将/dev/hda1 设备卸载，设备必须处于空闲状态

进程查看 - ps

ps #显示当前系统进程信息

ps -ef #显示系统所有进程信息

杀死进程 - kill

kill -9 500 #将进程编号为 500 的程序杀死