

菜鸟进阶系列

嵌入式

Linux



应用开发菜鸟进阶

• 梁旭辉 主编

- 整本书是以一个初学者的心态来进行学习的，同时又以一个刚刚进阶的开发者角度来叙述
- 书中总结了笔者学习嵌入式Linux的点点滴滴
- 其中有笔者对嵌入式Linux的学习见解和曾经做过的项目
- 对于程序的运行结果，本书给出了大量的图片和数据



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

菜鸟进阶系列

嵌入式 Linux 应用开发菜鸟进阶

梁旭辉 主编

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书详细介绍了嵌入式 Linux 系统的方方面面,从嵌入式系统的发展、环境的搭建、相关的理论基础知识,再到深入地讲解嵌入式系统的开发,由易到难详细讲解了嵌入式系统开发的流程。本书分为四部分。第一部分是嵌入式 Linux 系统基础篇,主要介绍了嵌入式系统的发展,搭建嵌入式 Linux 系统环境, Linux 程序开发基础和嵌入式 Linux 设备驱动开发基础等知识。第二部分是嵌入式 Linux 设备驱动核心理论篇,讲解了 Linux 设备驱动的一些基本理论知识,有并发竞态、阻塞非阻塞、驱动延迟操作、内存分配中断处理以及工程开发等相关知识。第三部分是字符设备驱动篇,主要讲解 Linux 字符设备驱动的相关内容,如基于内存的字符设备、按键控制 LED 字符驱动、温度传感器、MP3 播放器和动态图像系统开发。第四部分是块设备驱动篇,内容包括块设备驱动程序的开发、基于 ramdisk 的虚拟块设备驱动以及 SD 卡块设备驱动。

本书不仅有浅显易懂的嵌入式基础理论,还有详尽的软硬件设计开发的相关知识,并辅以多种实例来进行讲解,有较强的可读性与实用性。本书适合广大从事嵌入式系统设计与开发的工程技术人员阅读。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

嵌入式 Linux 应用开发菜鸟进阶 / 梁旭辉主编. —北京:电子工业出版社, 2013.9

(菜鸟进阶系列)

ISBN 978-7-121-21187-4

I. ①嵌… II. ①梁… III. ①Linux 操作系统—程序设计 IV. ①TP316.89

中国版本图书馆 CIP 数据核字 (2013) 第 179182 号



策划编辑:王敬栋

责任编辑:周宏敏

印 刷:三河市鑫金马印装有限公司

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:18.75 字数:480 千字

印 次:2013 年 9 月第 1 次印刷

印 数:4 000 册 定价:48.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前 言

本书从理论到实践，内容非常丰富，在章节编排上采用循序渐进的方法，从基础讲起，读者可根据自身理论水平，合理安排学习进度。

本书是一本面向嵌入式 Linux 初学者的书籍，全书基于一块经典开发板——友善之臂的 mini2440，在最新版本的 Linux 内核代码中都可以找到这块开发板的支持代码，是初学者学习嵌入式 Linux 入门的经典开发板。本书编者也是从这块开发板一步一步学习至今的。整本书是以一个初学者的心态来进行学习的，同时又以一个刚刚进阶的开发者的身份来叙述，书中总结了编者学习嵌入式 Linux 的点点滴滴，其中有编者对嵌入式 Linux 的学习见解和曾经做过的项目，在此愿意一一呈现给读者。

对于程序的运行结果，本书给出了大量的图片和数据。供读者在使用代码时进行比较，修改或改进代码，进而实现自己预定的目标。书中不仅详细讲解了相关的理论知识，更给出了实例，便于读者迅速掌握相关知识。

本书由梁旭辉主编，参加本书编写的还有刘宏伟、王利锋、牛冬芳、程建兵、贾善静、徐凤玉、吕国锋、宋海涛、边有喜、范凤南、谭鹏超、祝意、尚开卫和师慧平。

由于编者水平有限，书中难免存在错误及不妥之处，敬请读者提出宝贵意见。

编 者

反侵权盗版声明

电子工业出版社依法对本作品享有专有出版权。任何未经权利人书面许可，复制、销售或通过信息网络传播本作品的行为；歪曲、篡改、剽窃本作品的行为，均违反《中华人民共和国著作权法》，其行为人应承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。

为了维护市场秩序，保护权利人的合法权益，本社将依法查处和打击侵权盗版的单位和个人。欢迎社会各界人士积极举报侵权盗版行为，本社将奖励举报有功人员，并保证举报人的信息不被泄露。

举报电话：(010) 88254396; (010) 88258888

传 真：(010) 88254397

E-mail: dbqq@phei.com.cn

通信地址：北京市海淀区万寿路 173 信箱

电子工业出版社总编办公室

邮 编：100036

目 录

第 1 章 嵌入式 Linux 系统简介	1
1.1 嵌入式与 Linux 系统概述	1
1.1.1 Linux 操作系统及其发展	1
1.1.2 Linux 发行版本	2
1.2 Windows 与 Linux 文件系统的异同	4
1.2.1 Linux 文件系统类型	4
1.2.2 Windows 与 Linux 文件系统的相同点	5
1.2.3 Windows 与 Linux 文件系统的不同点	5
1.3 Linux 文件类型与属性	7
1.3.1 Linux 文件类型	7
1.3.2 Linux 文件属性	7
1.4 嵌入式 Linux 常用命令及工具	8
1.4.1 Linux 常用命令	8
1.4.2 嵌入式 Linux 常用工具	9
小结	10
第 2 章 搭建嵌入式 Linux 学习环境	12
2.1 硬件开发平台	12
2.1.1 mini2440 开发板硬件资源	12
2.1.2 外围设备	14
2.2 软件开发平台搭建	14
2.2.1 tftp 服务器与 nfs 服务器搭建	15
2.2.2 交叉编译工具	16
2.2.3 uboot 的移植	17
2.2.4 Linux 内核移植	18
2.2.5 Linux 根文件系统移植	19
小结	23
第 3 章 Linux 应用程序学习	24
3.1 Linux shell 脚本	24
3.1.1 语法	24
3.1.2 变量	25
3.1.3 shell 脚本语句	26

3.1.4 注释	27
3.2 Linux 下 makefile 的编写	27
3.3 Linux 文件编程	28
3.3.1 文件与目录	29
3.3.2 系统调用	30
3.3.3 标准 I/O 库	32
3.4 线程	34
3.4.1 线程优点	34
3.4.2 线程分类	35
3.4.3 线程实现	35
3.5 网络编程	37
3.5.1 TCP/IP 协议	37
3.5.2 TCP 和 UDP	38
3.5.3 套接字	39
小结	40
第 4 章 嵌入式 Linux 设备驱动开发基础	41
4.1 嵌入式 Linux 设备驱动的概述	41
4.1.1 Linux 设备驱动的作用	41
4.1.2 Linux 设备驱动的分类及特点	42
4.2 处理器与储存器	42
4.2.1 处理器	42
4.2.2 存储器	43
4.3 接口	45
4.3.1 串口	46
4.3.2 I2C	46
4.3.3 USB	47
4.3.4 以太网接口	48
4.4 Linux 内核模块	49
小结	50
第 5 章 Linux 设备驱动之并发与竞态	51
5.1 并发及其管理	51
5.2 自旋锁	52
5.2.1 自旋锁 API	52
5.2.2 自旋锁基本使用模式	53
5.2.3 其他自旋锁函数	54
5.2.4 读者/写者自旋锁	55
5.2.5 Linux 内核自旋锁	55

5.3 信号量与互斥体	57
5.3.1 信号量	57
5.3.2 互斥体	62
5.4 原子操作	64
5.4.1 原子整数操作	64
5.4.2 原子位操作	65
5.5 中断屏蔽	65
小结	66
第 6 章 Linux 设备驱动中的阻塞与非阻塞 I/O	67
6.1 阻塞与非阻塞 I/O	67
6.1.1 休眠	68
6.1.2 等待队列	68
6.1.3 阻塞和非阻塞操作	70
6.2 轮询操作	70
6.2.1 轮询概述	71
6.2.2 Linux 应用程序中的轮询编程——select()	71
6.2.3 Linux 设备驱动中的轮询编程——poll()	73
小结	75
第 7 章 Linux 设备驱动之时间、延迟及延缓操作	76
7.1 Linux 内核时间	76
7.1.1 处理器特定的寄存器	76
7.1.2 延迟操作	77
7.2 内核定时器	78
7.3 tasklet	80
7.4 工作队列	82
小结	84
第 8 章 Linux 设备驱动之内存分配与 I/O 访问	85
8.1 内存空间与 I/O 空间	85
8.2 内存管理	85
8.3 内存分配	88
8.3.1 kmalloc 与 vmalloc	88
8.3.2 get_free_page 和相关函数	89
8.4 设备 I/O 端口和 I/O 内存的访问	90
8.4.1 I/O 寄存器和常规内存	90
8.4.2 操作 I/O 端口（申请，访问，释放）	91
8.4.3 操作 I/O 内存（申请，映射，访问，释放）	92
8.4.4 像 I/O 内存一样访问端口	93

8.4.5 ARM 体系的 I/O 操作接口	94
8.5 I/O 内存的静态映射	94
小结	96
第 9 章 Linux 设备驱动之中断处理	97
9.1 Linux 中断处理程序框架	97
9.1.1 中断处理过程	97
9.1.2 Linux 中断处理程序的上半部与下半部机制	99
9.2 Linux 中断编程	100
9.3 中断共享	103
9.4 中断驱动的 I/O	104
小结	104
第 10 章 Linux 设备驱动工程开发技术	105
10.1 自创建设备文件	105
10.2 混杂设备驱动	107
10.3 Linux 设备模型	109
10.3.1 kobject 和 kset 数据结构	110
10.3.2 sysfs 文件系统	113
10.3.3 总线、设备和驱动程序	114
10.4 设备驱动分层思想	120
小结	121
第 11 章 Linux 字符设备驱动	122
11.1 字符设备驱动结构	122
11.1.1 主、次设备号	122
11.1.2 字符设备驱动的重要数据结构	124
11.2 字符设备的注册和释放	128
11.2.1 字符设备的注册	129
11.2.2 字符设备的释放	131
11.3 字符设备驱动的一些小技巧	131
小结	133
第 12 章 基于内存的简单字符设备	134
12.1 简单字符驱动的数据结构	134
12.1.1 定义字符设备驱动的设备数据结构	134
12.1.2 定义 file_operation 结构和挂接相应的系统调用函数	135
12.2 简单字符驱动设计	136
12.2.1 字符设备驱动的加载与卸载	137
12.2.2 字符设备驱动的打开与关闭	139
12.2.3 字符设备驱动的读写函数	140

12.2.4 字符设备驱动的 llseek 函数	141
12.3 应用程序测试	142
12.3.1 应用程序设计原理	142
12.3.2 程序代码	142
12.4 简单字符设备驱动运行效果	143
12.4.1 makefile 程序的编写	143
12.4.2 字符设备驱动运行效果	144
小结	145
第 13 章 按键控制 LED 字符驱动	146
13.1 按键和 LED 设备硬件接口和寄存器	146
13.2 按键控制 LED 驱动原理	147
13.3 按键控制 LED 驱动程序分析	148
13.3.1 按键与 LED 集成在一个驱动中	148
13.3.2 按键与 LED 驱动的实现	156
13.4 按键控制 LED 驱动测试	171
13.5 驱动程序的运行效果	171
小结	173
第 14 章 基于 DS18B20 传感器的驱动开发	174
14.1 DS18B20 温度传感器简介	174
14.1.1 温度传感器的工作特点	175
14.1.2 DS18B20 的外部结构	175
14.1.3 DS18B20 的内部结构	176
14.1.4 DS18B20 的应用电路结构	178
14.1.5 DS18B20 的工作原理	178
14.2 DS18B20 温度传感器程序设计实现	180
14.2.1 复位程序	180
14.2.2 写 1 字节子程序	182
14.2.3 读 1 字节子程序	183
14.2.4 确定温度传感器的头文件和相应的数据结构	185
14.2.5 确定驱动要实现的功能函数和 file_operation 结构的挂接	186
14.2.6 编写加载和卸载函数	186
14.2.7 编写要实现的功能函数	187
14.3 应用程序设计	195
14.4 驱动运行效果	197
14.4.1 makefile 文件的编写	197
14.4.2 应用程序运行效果图	197
小结	198

第 15 章 MP3 播放器	199
15.1 MP3 播发器简介	199
15.2 按键播放 MP3 的原理	199
15.2.1 MP3 播放器系统框架	199
15.2.2 MP3 音频驱动的移植	201
15.3 MP3 程序设计	203
15.3.1 按键驱动程序设计	203
15.3.2 MP3 应用程序设计	208
15.3.3 最终的程序代码	215
15.4 MP3 播放器的测试	223
小结	224
第 16 章 基于 motion 的动态图像检测系统开发	225
16.1 动态图像检测系统简介	225
16.2 系统框图和流程图	225
16.2.1 系统框图	226
16.2.2 系统设计技术	227
16.3 动态监测系统程序分析	228
16.3.1 motion 动态图像采集系统移植	228
16.3.2 温度采集系统	231
16.3.3 动态图像控制系统	231
16.3.4 远程监控服务系统	239
16.3.5 SD 卡存储系统	248
16.4 系统测试运行效果图	248
小结	250
第 17 章 块设备驱动程序	251
17.1 字符设备驱动与块设备驱动	251
17.2 块设备驱动基本框架	252
17.3 块设备在 Linux 系统中各层的调用关系	252
17.4 块设备驱动的重要数据结构	253
17.4.1 block_device_operations 结构体	254
17.4.2 gendisk 结构体	255
17.4.3 request 与 request_queue 结构体	255
17.4.4 块 I/O	256
17.4.5 块设备中各结构体之间的关系	258
17.5 块设备驱动程序分析	259
小结	261
第 18 章 基于 ramdisk 的虚拟磁盘块设备驱动	262

18.1 块设备驱动程序设计	262
18.2 基于 I/O 调度器的块设备驱动程序设计	262
18.3 没有 I/O 调度器的块设备驱动程序设计	265
小结	267
第 19 章 SD 卡块设备驱动	268
19.1 SD 卡概述	268
19.1.1 SD 卡简介	268
19.1.2 SD 卡的引脚	269
19.1.3 SD 卡的内存组织	270
19.2 SD 卡寄存器	271
19.2.1 CID 寄存器	271
19.2.2 SD 总线协议	272
19.2.3 SD 卡指令和命令	272
19.2.4 块读写操作	274
19.3 SD 卡设备驱动程序分析	275
19.3.1 SD 卡驱动程序的重要数据结构	276
19.3.2 SD 设备驱动分析	278
19.3.3 SD 卡驱动在 mini2440 开发板的开发与移植	281
19.4 测试 SD 卡驱动程序	283
19.4.1 系统设计思路与步骤	283
19.4.2 系统测试结果	285
小结	286

第①章

嵌入式 Linux 系统简介

嵌入式 Linux 技术入门的要求比较高,所以坊间所说的零基础学习嵌入式 Linux 不太现实。在本书的第 1 章,笔者将带大家了解一下什么叫做 Linux 和嵌入式 Linux 的基本概念,以及 Linux 系统与我们所用的 Windows 操作系统的异同。最后教大家如何使用 Linux 操作系统以及一些常用的工具和命令。

当然,学好嵌入式 Linux 不是一两天的事情,在本章最后,作者会有针对性地给出一些学好嵌入式 Linux 的书籍以及相关学习要点等内容,目的是引导读者尽快入门。

在这一章中,我们将学习到如下知识:

1. 嵌入式 Linux 的发展概述。
2. Linux 文件的概念。
3. Linux 文件系统的概念。
4. Linux 系统中常用的命令。
5. Linux 系统中常用的开发工具。



1.1 嵌入式与 Linux 系统概述

嵌入式系统是嵌入到对象体系中的专用计算机系统。电气工程师学会从应用对象的角度将嵌入式系统定义为:用于控制、监视机器、装置、工厂等大规模系统的设备。国内普遍认同的嵌入式系统定义为:嵌入式系统是以应用为中心,以计算机技术为基础,并且软硬件可裁剪,适用于应用系统对功能、可靠性、成本、体积功耗有严格要求的专用计算机系统。如今,嵌入式系统的使用已经非常普遍。MP3、PDA 和手机属于手持的嵌入式产品,DVD 和机顶盒也是嵌入式产品。

嵌入式 Linux 学习的第一步也是最重要的一步就是 Linux 操作系统的学习,所以本章重点是引导读者如何快速学习 Linux 操作系统,而不是全篇介绍基础知识,很多基础知识可以在一些相关的书籍中找到,本书只是引导读者如何进入嵌入式 Linux 世界。

1.1.1 Linux 操作系统及其发展

Linux 是一种自由和开放源码的类 Unix 操作系统。目前存在着许多不同的 Linux,但它们都使用了 Linux 内核。Linux 可安装在各种计算机硬件设备中,从手机、平板电脑、路由器和视频游戏控制台,到台式计算机、大型机和超级计算机。Linux 是一个领先的操作系统,世界上运算最快的 10 台超级计算机运行的都是 Linux 操作系统。

严格来讲, Linux 这个词本身只表示 Linux 内核, 但实际上人们已经习惯了用 Linux 来形容整个基于 Linux 内核并且使用 GNU 工程的各种工具和数据库的操作系统。

Linux 操作系统诞生于 1991 年 10 月 5 日(这是第一次正式向外公布的时间), 得名于计算机业余爱好者 Linus Torvalds。借助于 Internet 网络, 并经过全世界各地计算机爱好者的共同努力, 现已成为当今世界上使用最多的一种类 Unix 操作系统, 并且使用人数还在迅猛增长。

Linux 操作系统从 1992 年问世至今的 20 年间, 已经发展成为一个功能强大、设计完善的操作系统。目前 Linux 已经可以与各种传统的商业操作系统分庭抗礼, 占据了市场的相当一部分份额。Linux 强大的功能、优良的特性、易于定制、具有大量适用的应用软件等特点使之成为嵌入式操作系统的一个比较理想的选择。同时, Linux 是一个与生俱来的网络操作系统, 成熟而且稳定。

Linux 是源代码开放软件, 不存在黑箱技术, 任何人都可以修改它, 或者用它开发自己的产品。Linux 系统的定制特性可以让系统内核做得很小。一个带有中文系统及图形化界面的核心程序也可以做到不足 1MB, 而且同样稳定。Linux 作为一种可裁剪的软件平台系统, 是发展未来嵌入设备产品的绝佳资源, 遍布全球的众多 Linux 爱好者又能给予 Linux 开发者强大的技术支持。因此, Linux 作为嵌入式系统新的选择, 具有非常大的发展前途。

Linux 操作系统的诞生、发展和成长过程始终依赖以下 5 个重要支柱: Unix 操作系统、MINIX 操作系统、GNU 计划、POSIX 标准和 Internet 网络。

1.1.2 Linux 发行版本

很多人都认为 Linux 是一个操作系统, 其实这种说法并不妥当, Linux 只是一个操作系统中的内核, 而一个完整的操作系统不仅仅是内核。所以, 许多个人、组织和企业开发了基于 GNU/Linux 的 Linux 发行版, 其中最著名的便是 Red Hat 公司的 Red Hat 系列以及社区组织的 Debian 系列。

Linux 发行版本可以大体分为两类, 一类是商业公司维护的发行版本, 另一类是社区组织维护的发行版本。前者以著名的 Red Hat (RHEL) 为代表, 后者以 Debian 为代表。在本书中会用到 RHEL5 作为我们的 Linux 开发系统。

下面简单介绍几个具有代表性的 Linux 发行版本, 即 Red Hat、Debian、SUSE、Ubuntu 等。

1. Red Hat 版本

国内, 乃至全世界的 Linux 用户最熟悉、最耳熟能详的发行版想必就是 Red Hat 了。Red Hat 最早由 Bob Young 和 Marc Ewing 在 1995 年创建。目前 Red Hat 分为两个系列: 由 Red Hat 公司提供收费技术支持和更新的 Red Hat Enterprise Linux, 以及由社区开发的免费的 Fedora Core。

适用于服务器的版本是 Red Hat Enterprise Linux。由于这是个收费的操作系统, 于是国内外许多企业或中间商选择 CentOS。CentOS 可以算是 RHEL 的克隆版, 但它最大的好处是免费。本书将使用 RHEL5 作为我们的开发环境系统, 这是因为 Red Hat 的培训及认

证被认为是Linux 认证的标准。从这方面来讲对我们以后的考证以及企业管理等都有很多好处。

2. Debian GNU/Linux 版本

在很多社区, Debian Linux 是现在讨论得相当热烈的 Linux 发行版。它至今坚持由开源社区的黑客们按照 GNU 的思想以更完善、更开放、更自由的原则独立发布, 不具任何商业性质。它的发布版名称很有意思, 例如最新的稳定版本为 6.0.0, 但是通常不是这样称呼的, Debian 的世界里以电影《玩具总动员》的角色名称来命名自己的产品, 包括 Woody、Sarge 和 Sid, 最新版本的 Debian 叫做“三只眼睛的绿色大嘴玩具外星人”。

3. SUSE 版本

这是著名的 Novell 公司旗下的 Linux 发行版, 发行量在欧洲占第一位。它使用的是 YaST 的软件包管理方式, 拥有最华丽的 Linux 界面。商业公司的插手也意味着它的用户会获得更多的支持, 故 SUSE 是目前国内服务器领域用户群比较广泛的 Linux 系统。

SUSE 的社区主页是http://www.opensuse.org/Welcome_to_openSUSE.org。

SUSE 在 Novell 公司的主页是<http://www.novell.com/Linux/suse/>。

4. Ubuntu 版本

Ubuntu 是基于 Debian 体制的新一代 Linux 操作系统, 继承了 Debian 的一切优点, 并提供了更易用、更人性化的使用方式。由于桌面操作友好, 一般作为 Linux 初学者入门学习的操作系统。再者, Ubuntu 有着非常系统的发布时间表, 每半年发布一次, 所以有 11.04、11.10、12.04 等非常规律的名称。由于 Ubuntu 具有安装简单, 操作方便, 启动速度快, 绚丽的桌面, 以及更人性化的特点, 现已成为目前最为引人注目的一个操作系统发行版本。

Ubuntu 是非洲方言, 指的是一种博爱的、乐于分享的和谐境界。该产品更偏于面向广大的桌面用户, 其中文社区拥有相当详细适用的新手文档。非洲人的 Ubuntu 哲学提到: 不同于欧洲“我思固我在”的哲学思想, 非洲人传统的思想是“我之所以成为一个人, 是因为我属于, 我参与, 我分享”, 正因为是群体中的一员, 与他人分享一切, 所以才得到尊重, 从而成为了一个人。和谐、友善、集体的观念融入非洲人的生活, 祖先的智慧告诉他们: 每个人都应该对愤怒、仇恨、报复避而远之。

本书作者最初学习 Linux 也是从 Ubuntu 9.04 版开始的, 对 Ubuntu 有着非常深的感情, 虽然接触过其他的 Linux 发行版本, 但依然觉得 Ubuntu 是初学者最容易接受和学习 Linux 操作系统的发行版本, 特别是其丰富的软件库和简单的软件安装以及绚丽的 3D 效果等, 至今都让人无法忘怀。当然, 现在很多智能操作系统都采用了 Linux 内核, 因此智能设备的很多开发平台也是基于 Ubuntu 来进行开发的。比如, Android 智能手机操作系统, 现在很多开发 Android 的平台都是基于 Ubuntu 的, 包括笔者现在工作的公司都是采用 Ubuntu 作为 Android 手机的开发平台。由此可见, Ubuntu 在未来一段时间内肯定会非常流行, 因此希望读者能够掌握这个 Linux 操作系统的发行版本, 并且能够熟练使用这个 Linux 操作系统。也许在以后的某一天你从事了这个行业, 会用到该 Linux 操作系统发行版本。



1.2 Windows 与 Linux 文件系统的异同

很多人可能只接触过 Windows 操作系统,一下子从 Windows 转换到 Linux 可能会对 Linux 目录中的一大堆文件和目录无所适从。要学好 Linux 操作系统,首先要理解 Linux 的基本设计思想。Linux 的基本思想有两点:第一,一切都是文件;第二,每个软件都有确定的用途。其中第一条详细来讲就是系统中的所有内容都归结为一个文件,包括命令、硬件和软件设备、操作系统、进程等,它们对于操作系统内核而言,都被视为拥有各自特性或类型的文件。至于说 Linux 是基于 Unix 的,很大程度上也是因为这两者的基本思想十分相近。

1.2.1 Linux 文件系统类型

很多人第一次接触并具有感性认识的操作系统就是 Windows,由于先入为主,所以对操作系统的理解也就是对 Windows 的理解。想要学好 Linux,首先就是要区分 Windows 与 Linux 的不同;第二就是要理解 Linux 文件以及文件系统的概念,在此基础之上再理解 Linux 各个功能模块;第三就是要经常使用 Linux 操作系统。要做好这三步才能更好地认识 Linux 操作系统以及开发属于我们的嵌入式 Linux 操作系统。下面给 Linux 文件系统分一下类型。

- ❑ ext2、ext3 和 ext4: 现在 Linux 主流版本的文件系统一般都是 ext4,从 ext3 到 ext4 有了比较大的变化,从可用性、数据完整性、速度以及易于转化等方面都有了较大提高。从内核 2.6.28 开始就正式开始支持 ext4 文件系统了。
- ❑ swap 文件系统: 该系统在 Linux 中作为交换分区使用。在安装 Linux 操作系统时,交换分区是必须要建立的,手动建立或者默认建立都可以,特别是 Ubuntu,而且一般都是内存的两倍。
- ❑ vfat 文件系统: Linux 中把 DOS 里采用的 FAT 文件系统(包括 FAT12、FAT16 和 FAT32)都称为 vfat 文件系统。
- ❑ NFS 文件系统: 网络文件系统,允许一个系统在网络上与其他人共享目录和文件。用户和程序可以像访问本地文件一样访问远端系统。
- ❑ ISO9660 文件系统: 标准的光盘(CDROM)使用的文件系统,是通用的 Rock Ridge 增强系统,允许长文件名。
- ❑ SMB: SMB 是一种支持 Windows for Workgroups、Windows NT 和 Lan Manager 的基于 SMB 协议的网络操作系统。
- ❑ Proc: 在 Linux 系统中作为一种伪文件系统出现的、用来作为连接内核数据结构的界面。

在学习 Linux 文件系统前,要理解 Windows 与 Linux 的异同,这样才能更快地进入 Linux 操作系统的学习。Linux 与 Windows 相比,相同点有用户和组、文件系统、端口和设备、网络、服务等;不同的有 Linux 应用目标、可选的 GUI、文件名拓展、重新引导、命令区分大小写等。

1.2.2 Windows 与 Linux 文件系统的相同点

1. 用户和组

Linux 是多用户多任务操作系统，而 Windows 是单用户多任务操作系统。二者都可以由许多不同的用户来使用，并为每个用户提供单独的环境和资源。二者都是基于用户身份来控制安全性，都可以以组成员的方式来控制资源的访问权限，因此在用户数目较大时可以不必要为每一个账号设置权限。用户和组可以集中管理，使多个服务器共享相同的用户和身份验证数据。

2. 文件系统

Linux 和 Windows 都支持多种文件系统。文件资源可以通过 NetBIOS、FTP 或者其他协议与其他客户机共享。可以很灵活地对各个独立的文件系统进行组织，由管理员来决定它们在何处可以且以何种方式被访问。

3. 端口和设备

两种操作系统都支持各种物理设备端口，比如并口、串口和 USB 接口。支持各种控制器，比如 IDE 和 SCSI 控制器。Linux 还支持很多刚刚上市的标准硬件。

4. 网络

Linux 和 Windows 都支持多种网络协议，比如 TCP/IP、NetBIOS 和 IPX；都支持多种类型的网络适配器；都具备通过网络共享资源的能力，比如共享文件和打印；都可以提供网络服务能力，比如 DHCP 和 DNS。

5. 服务

Linux 和 Windows 都提供服务。所谓服务，指的是那些在后台运行的应用程序，可以为系统和远程调用该服务的计算机提供一些功能。在系统引导时可以单独控制并自动启动这些程序。（注意：Linux 中沿用了 Unix 的习惯，称这种应用程序为 daemon。）

1.2.3 Windows 与 Linux 文件系统的不同点

1. Linux 的应用目标是网络而不是打印

Windows 最初出现的时候，这个世界还是一个纸张的世界。Windows 的伟大成就之一在于您的工作成果可以被方便地看到并打印出来。这样一个开端影响了 Windows 的后期发展。

同样，Linux 也受到其起源的影响。Linux 的设计定位于网络操作系统。它的设计灵感来自于 Unix 操作系统，因此它的命令设计比较简单，或者说比较简洁。由于纯文本可以非常好地跨网络工作，所以 Linux 配置文件和数据都以文本为基础。

对那些熟悉图形环境的人来说，Linux 服务器初看上去可能比较原始。但是 Linux 开发者关注更多的是其内在功能而不是表面上的东西。即使是在纯文本的环境中，Linux 同样拥有非