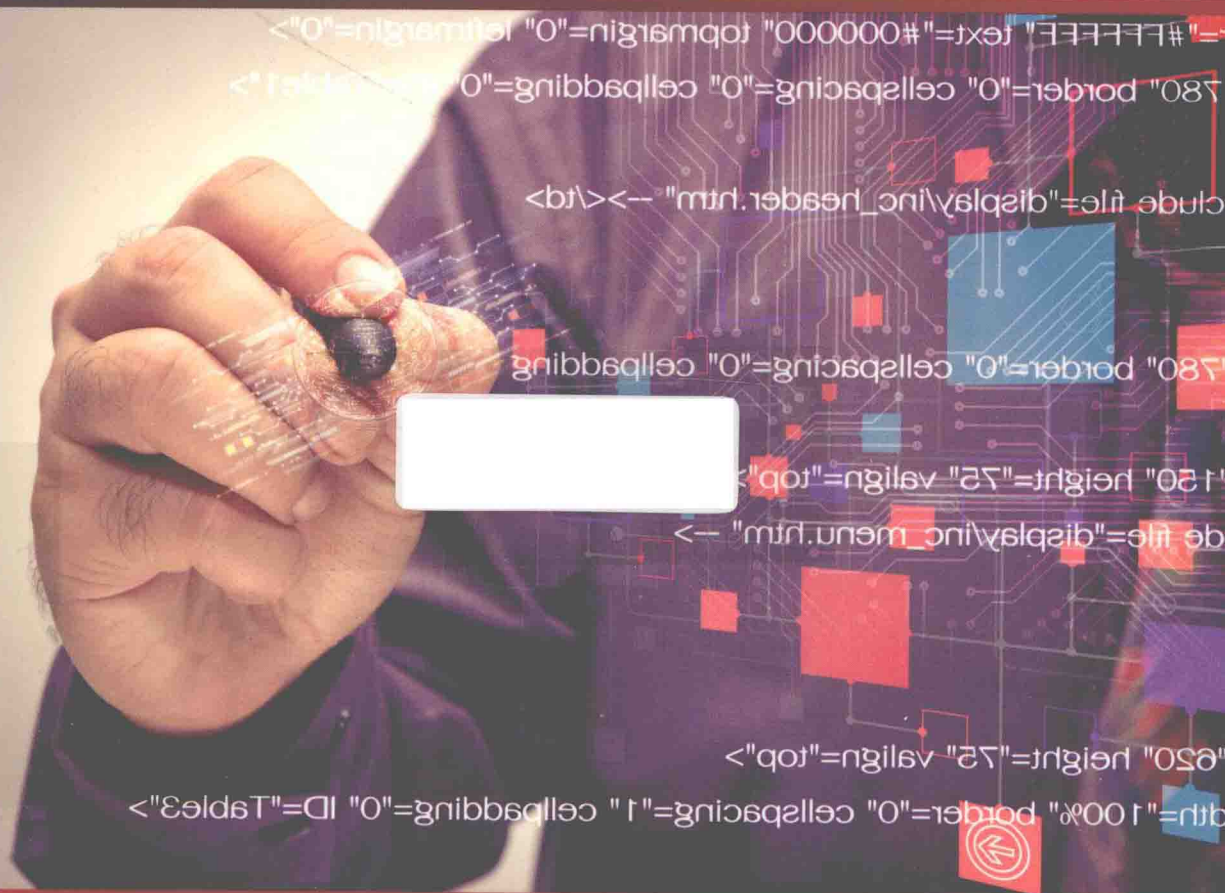


程国钢 编著



清华大学出版社

精通 Linux C 编程

程国钢 编著



清华大学出版社

北 京

内 容 简 介

本书深入浅出、循序渐进地讲解了 Linux 平台下的 C 程序设计,并通过大量的程序实例,以及综合开发案例的演示,帮助读者快速掌握 Linux 平台下 C 语言编程的方法和技巧。

本书内容翔实,共分为三大部分。第一部分为基础篇,共有 5 章,主要讲解了 Linux 系统基础、C 语言编程基础、Linux 下的两种常用文本编辑器 vim 和 Emacs、程序编译器 gcc 和调试器 gdb,以及 make 管理工具。第二部分为提高篇,共有 8 章,主要讲解了 Linux 系统下各种操作的系统调用,包括文件操作、进程控制、进程间通信、线程控制、网络编程、GTK+图形界面编程等。第三部分为实战篇,共有 5 章,分别为 5 个不同的 Linux 平台下 C 程序开发的综合案例,向读者详细阐述了 Linux 文件操作、GTK+图形界面编程、Linux 网络编程,以及基于 Linux 平台的嵌入式软件开发的方法和技巧。

本书由作者根据多年来的开发工作经验编著而成,语言通俗易懂,内容丰富,注重实例讲解,知识涵盖面广。非常适合 Linux 平台下 C 语言编程的初学者以及高校本科生、研究生阅读,也适合在 Linux 系统下进行 C 程序开发的工程师查阅和学习。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

精通 Linux C 编程 / 程国钢 编著. —北京:清华大学出版社, 2015
ISBN 978-7-302-39367-2

I. ①精… II. ①程… III. ①Linux 操作系统—程序设计 ②C 语言—程序设计 IV. ①TP316.89 ②TP312

中国版本图书馆 CIP 数据核字(2015)第 031610 号

责任编辑:刘金喜

装帧设计:牛静敏

责任校对:成凤进

责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:清华大学印刷厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:36.25 字 数:905 千字
(附光盘 1 张)

版 次:2015 年 7 月第 1 版 印 次:2015 年 7 月第 1 次印刷

印 数:1~2500

定 价:68.00 元

前言

P R E F A C E

Linux 诞生于 1991 年,由 UNIX 发展而来。几乎每天 Linux 都会以某种方式出现在我们的生活中,我们已经数不清在 Linux 上有多少应用程序,以及有多少机构在使用 Linux。国内外无数大型企业都在使用 Linux 系统作为服务器解决方案,尤其是在嵌入式开发领域,Linux 的应用更是在不断增加。在百度上搜索“Linux 软件工程师”的相关网页约有 1 230 000 篇,由此可见,在 Linux 平台下进行程序开发的需求之大。

作者根据自己多年来在 Linux 平台下进行 C 程序开发所积累的经验,并融合大量的程序实例而著成此书。本书由浅入深,适合各个水平阶段的读者学习。

本书特点

1. 配套视频讲解光盘

为了让读者更加快速、直观地学习本书内容,作者专门为本书录制了全程多媒体视频教学,包括各个技术知识点,以及程序案例的分析。结合视频的讲解,能够帮助读者更高效地掌握 Linux 下 C 语言编程的技巧与方法。

光盘具体内容有:

- Linux 常用工具使用视频
- Linux 入门视频
- 案例源文件
- 本书全程多媒体教学视频
- 实战演练参考答案

2. 循序渐进,由浅入深

本书从 Linux 系统的安装、C 语言编程基础、Linux 下的基本编辑器、程序编译器和调试器、make 工具管理器,到 Linux 系统的各种函数调用,再到 Linux 下具体程序案例的设计开发,内容由浅入深,囊括了 Linux 下 C 程序开发的各个环节。

3. 程序实例丰富,实践性强

在本书中,几乎每个知识点都会伴随一个或多个程序实例,通过实例来加深和巩固读者对知识点的理解和掌握。对于每个程序实例,作者都添加了十分详细的注释,方便读者理解。并且,对于所有的实例,读者都可以在自己的实验环境中完整实现。尤其是第 3 部分的 5 个项目

案例，更是完整地向读者演示了 Linux 环境下项目实例的设计与开发。

4. 技术全面，知识点阐述到位

网络编程和图形界面编程是 C 程序学习中比较深入的知识，也是本书重点讲解的内容。在第 3 部分，我们将这些知识点与实际的项目开发结合，通过逐步设计与实现，深化和巩固读者对它们的理解与掌握。将 Linux 系统开发技术、C 语言开发技术、软件工程思想融会贯通，使得本书成为思想和内容都极其丰富的图书。

本书内容

第 1 章：介绍 Linux 系统的基本概念和安装方法，Linux 下的常用命令，以及 Shell 的使用。这些是使用 Linux 的基础，帮助读者为本书后续的学习打下扎实的基础。

第 2 章：详细讲述了 C 语言的编程基础，它们是熟悉和掌握 C 语言的必备知识，同时也方便了读者在阅读本书时查阅 C 语言中的相关知识点。

第 3 章：讲述 Linux 下最常用的两种文本编辑器 vi 和 Emacs，并通过实例讲解让读者一步步地学会如何使用这些编辑器。

第 4 章：讲解 Linux 系统下的程序编译器 gcc 和程序调试器 gdb，两者是在 Linux 下进行 C 程序开发所必备的工具。

第 5 章：讲述 Linux 下的工程管理器 make，以及 Makefile 的书写规则。make 工具大大提高了实际项目的开发效率，几乎所有 Linux 下的项目编程都会涉及它。

第 6 章：讲解基于文件描述符的文件 I/O 操作，以及 Linux 中文件系统的概念。文件操作是 Linux 系统中最常见的操作之一，在 Linux 中，所有的内容都被看成文件，所有的操作都可以归结为对文件的操作。

第 7 章：讲述基于流的文件 I/O 操作。基于流的 I/O 操作是由标准 C 函数库提供的，与基于文件描述符的 I/O 操作相比，基于流的 I/O 更简单、方便。在大多数情况下，程序员更愿意使用基于流的输入输出方法。

第 8 章：详细阐述了 Linux 下进程控制的原理。进程是操作系统中一个非常重要的概念，熟悉使用进程的操作和进程控制的相关系统调用，会使用户在使用 Linux 系统完成各种工作时更加得心应手。

第 9 章：讲解 Linux 信号的基本概念，以及 Linux 下信号处理的机制。信号的使用对于灵活使用 C 语言在 Linux 环境下进行程序开发是非常有益的，在编写大型的程序时，经常会需要处理多个进程之间的异步事件，所以是离不开信号的使用的。

第 10 章：详细讲述了 Linux 进程间通信的概念，以及 Linux 下 5 种最常用的进程间通信方式，它们包括管道、命名管道、消息队列、共享内存、信号量等。

第 11 章：介绍了 Linux 的线程相关基础知识，如何对其进行操作以及线程间同步的方法。随着多核心处理器的普及，合理使用多线程可以大大提高程序代码的执行效率。

第 12 章：详细讲述了 Linux 网络编程的原理与方法，通过大量的程序实例演示了 Socket

编程中常用 API 的使用方法。

第 13 章：讲述 GTK+图形界面编程。GTK+是 Linux 下基于 C 的图形界面开发库，本章通过程序实例向读者演示了使用 GTK+库创建各种界面元件的方法。

第 14 章：通过讲述一个计算器软件的设计案例，进一步向读者演示图形界面编程的技巧，以及 GTK+信号与回调函数的原理。

第 15 章：通过设计一个类似于 QQ 的聊天软件，向读者演示了小型项目工程软件的模块划分方法，以及 Linux 下的 C 程序开发的步骤。使读者能够更深层次地掌握 GTK+图形界面编程，以及 Linux 网络编程的原理方法。

第 16 章：讲述 Linux 下一种基于 C/S 模式实现的远程管理工具的设计。使读者对 Linux 下的文件 I/O 操作、相关系统服务的管理有更深层次的了解和认识，并进一步掌握 GTK+图形界面编程，套接字 Socket 网络编程的使用方法。

第 17 章：讲述了在 Linux-2.4.20-8 内核下利用 Netfilter 数据控制过滤机制完成简易防火墙软件的设计，实现了对固定端口、网页访问及不同协议类型的数据报文的管理和控制。

第 18 章：讲述基于 Linux 的嵌入式家庭网关远程交互操作平台的设计。本章向读者阐述了嵌入式系统，以及家庭网关的概念。案例采用 B/S 结构的开发模式，嵌入式 Web 服务器选取 Boa，并结合 CGI 技术，实现了动态的具体智能设备的访问和控制。

本书适合的读者：

- Linux 及 C 程序设计初学者。
- 大中专院校计算机及相关专业的学生。
- 高校计算机及相关专业本科生、研究生。
- Linux 软件开发从业人员。
- 开源软件开发爱好者。
- 社会相关培训学员。

最后，我要感谢我的家人及好友陈曦在我最困难的时候带给我快乐与动力，支持我一直坚持下来，并最终完成这部著作。

本书主要由程国钢主持编写，参加本书编写工作的还有许小荣、张泽、刘荣、张璐、王统、周艳丽、刘波、苏静、王冬、王龙、陈作聪、王松年、卿前华、蔡娜、肖岳平、聂阳、沈毅、张华杰等，在此，编者对以上人员致以诚挚的谢意！

作者力图使本书案例功能翔实，并尽量使用关键编程技术进行程序设计和简化程序代码，但由于水平有限，书中难免有错误、纰漏之处，欢迎广大读者、同仁批评斧正。

服务邮箱：wkservice@vip.163.com

编者

2014 年 4 月

目录

CONTENTS

第 I 部分 基础篇

第 1 章 Linux 系统概述.....3

1.1 什么是 Linux 4

1.2 Linux 系统特点及主要功能..... 5

1.2.1 Linux 系统特点..... 5

1.2.2 Linux 系统的主要功能..... 6

1.3 Linux 的内核版本和发行版本..... 7

1.4 系统的安装..... 8

1.4.1 系统安装前的准备工作..... 8

1.4.2 安装 Linux..... 9

1.4.3 使用虚拟机..... 13

1.5 Shell 的使用..... 16

1.5.1 Shell 简介..... 16

1.5.2 常见 Shell 的种类..... 17

1.5.3 Shell 的简单使用..... 18

1.5.4 通配符..... 19

1.5.5 引号..... 20

1.5.6 注释符..... 21

1.6 Linux 常用命令..... 22

1.6.1 与目录相关的命令..... 22

1.6.2 与文件相关的命令..... 22

1.6.3 与网络服务相关的命令..... 23

1.7 本章小结..... 24

实战演练..... 24

第 2 章 C 语言编程基础.....25

2.1 C 语言的历史背景..... 26

2.2 C 语言的特点..... 26

2.3 C 语言的基本数据类型..... 27

2.3.1 整型.....27

2.3.2 实型.....28

2.3.3 字符型.....29

2.4 运算符与表达式..... 31

2.4.1 算术运算符与算术表达式.....31

2.4.2 赋值运算符与赋值表达式.....32

2.4.3 逗号运算符与逗号表达式.....33

2.5 C 程序的 3 种基本结构..... 33

2.5.1 顺序结构.....34

2.5.2 选择结构.....35

2.5.3 循环结构.....38

2.6 C 语言中的数据输入与输出..... 40

2.6.1 字符输出函数 putchar.....41

2.6.2 字符输入函数 getchar.....41

2.6.3 格式输出函数 printf.....41

2.6.4 格式输入函数 scanf.....43

2.7 函数..... 44

2.7.1 函数的定义.....44

2.7.2 函数的调用.....44

2.7.3 变量的存储类别.....46

2.8 数组..... 49

2.8.1 一维数组的定义和使用.....50

2.8.2 二维数组的定义和使用.....51

2.8.3 字符数组和字符串.....52

2.8.4 常用字符串处理函数.....53

2.9 指针..... 56

2.9.1 地址和指针.....57

2.9.2 指针的定义和使用.....57

2.9.3 数组与指针	58	3.3.1 启动与退出 Emacs	87
2.9.4 字符串与指针	59	3.3.2 Emacs 下的基本操作	88
2.9.5 指向函数的指针	60	3.4 Emacs 使用实例	93
2.10 结构体和共用体	60	3.5 本章小结	94
2.10.1 定义和引用结构体	60	实战演练	94
2.10.2 结构体数组	61	第 4 章 gcc 编译器与 gdb 调试器	97
2.10.3 指向结构体的指针	62	4.1 gcc 编译器简介	98
2.10.4 共用体	63	4.2 如何使用 gcc	99
2.10.5 使用 typedef 定义类型	64	4.2.1 安装和配置 gcc	99
2.11 链表	65	4.2.2 gcc 编译初步	100
2.11.1 链表概述	65	4.2.3 警告提示功能	102
2.11.2 建立动态单向链表	66	4.2.4 优化 gcc	104
2.11.3 单向链表的输出	67	4.2.5 链接库	107
2.11.4 对单向链表的删除操作	68	4.2.6 同时编译多个源程序	108
2.11.5 对单向链表的插入操作	69	4.2.7 管道	108
2.11.6 循环链表	69	4.2.8 调试选项	109
2.11.7 双向链表	70	4.3 gdb 调试器	110
2.12 位运算符和位运算	71	4.3.1 gdb 简介	110
2.12.1 “按位与”运算符(&)	71	4.3.2 gdb 常用命令	111
2.12.2 “按位或”运算符()	71	4.3.3 gdb 调试初步	112
2.12.3 “取反”运算符(~)	72	4.4 gdb 的使用详解	114
2.12.4 “异或”运算符(^)	72	4.4.1 调用 gdb	115
2.12.5 移位运算符(<<和>>)	72	4.4.2 使用断点	115
2.12.6 位域	72	4.4.3 查看运行时数据	117
2.13 C 语言预处理命令	73	4.4.4 查看源程序	122
2.13.1 宏定义	73	4.4.5 改变程序的执行	124
2.13.2 文件包含	75	4.5 xxgdb 调试器简介	127
2.13.3 条件编译	75	4.6 本章小结	128
2.13.4 #error 等其他常用预 处理命令	76	实战演练	128
2.14 本章小结	77	第 5 章 make 的使用和 Makefile 的编写	131
实战演练	77	5.1 什么是 make	132
第 3 章 vim 与 Emacs 编辑器	79	5.1.1 make 机制概述	132
3.1 vim 的使用	80	5.1.2 make 与 Makefile 的关系	134
3.2 vim 使用实例	85	5.2 Makefile 的书写规则	137
3.3 Emacs 的使用	87		

5.2.1	Makefile 的基本语法规则	138
5.2.2	在规则中使用通配符	139
5.2.3	伪目标	140
5.2.4	多目标	141
5.2.5	自动生成依赖性	141
5.3	Makefile 的命令	143
5.4	变量	144
5.4.1	变量的基础	144
5.4.2	赋值变量	145
5.4.3	define 关键字	146
5.4.4	override 指示符	147
5.4.5	目标变量和模式变量	147
5.5	常用函数调用	149
5.5.1	字符串处理函数	149
5.5.2	文件名操作函数	153
5.5.3	循环函数	155
5.5.4	条件判断函数	156
5.5.5	其他常用函数	157
5.6	隐式规则	159
5.6.1	隐式规则举例	159
5.6.2	隐式规则中的变量	160
5.6.3	使用模式规则	162
5.7	本章小结	164
	实战演练	164

第 II 部分 提高篇

第 6 章	文件 I/O 操作	169
6.1	软件编程体系简介	170
6.1.1	Linux 的文件系统结构	170
6.1.2	文件类型	171
6.1.3	文件访问权限	174
6.2	基于文件描述符的 I/O 操作	174
6.2.1	文件描述符	175
6.2.2	标准输入、标准输出和 标准出错	175
6.2.3	文件重定向	175

6.2.4	文件的创建、打开与关闭	177
6.2.5	文件的定位	181
6.2.6	文件的读写	183
6.3	文件的属性操作	187
6.3.1	改变文件访问权限	187
6.3.2	改变文件所有者	189
6.3.3	重命名	189
6.3.4	修改文件长度	190
6.4	文件的其他操作	190
6.4.1	stat、fstatt 和 lstat 函数	190
6.4.2	dup 和 dup2 函数	192
6.4.3	fcntl 函数	192
6.4.4	sync 和 fsync 函数	193
6.5	特殊文件的操作	194
6.5.1	目录文件的操作	194
6.5.2	链接文件的操作	197
6.5.3	管道文件的操作	200
6.5.4	设备文件	200
6.6	本章小结	200
	实战演练	201

第 7 章	基于流的 I/O 操作	203
7.1	流与缓存	204
7.1.1	流和 FILE 对象	204
7.1.2	标准输入、标准输出和 标准出错	204
7.1.3	缓存	204
7.1.4	对缓存的操作	207
7.2	流的打开与关闭	209
7.2.1	流的打开	210
7.2.2	流的关闭	211
7.2.3	流关闭前的工作	213
7.3	流的读写	214
7.3.1	基于字符的 I/O	214
7.3.2	基于行的 I/O	217
7.3.3	直接 I/O	219
7.3.4	格式化 I/O	222

7.4 本章小结	224	10.2.1 管道的概念	293
实战演练	224	10.2.2 管道的创建与关闭	294
第 8 章 进程控制	227	10.2.3 管道的读写	295
8.1 进程的基本概念	228	10.3 命名管道	300
8.1.1 Linux 进程简介	228	10.3.1 命名管道的概念	300
8.1.2 进程与作业	229	10.3.2 命名管道的创建	301
8.1.3 进程标识	229	10.3.3 命名管道的读写	302
8.2 进程控制的相关函数	231	10.4 消息队列	306
8.2.1 fork 和 vfork 函数	231	10.4.1 消息队列的概念	306
8.2.2 exec 函数	236	10.4.2 消息队列的创建与打开	309
8.2.3 exit 和 _exit 函数	242	10.4.3 消息队列的读写	309
8.2.4 wait 和 waitpid 函数	245	10.4.4 获得或设置消息队列属性	311
8.2.5 进程的一生	251	10.5 共享内存	316
8.2.6 用户 ID 和组 ID	251	10.5.1 共享内存的概念	316
8.2.7 system 函数	253	10.5.2 共享内存的相关操作	317
8.3 多个进程间的关系	255	10.6 信号量	322
8.3.1 进程组	255	10.6.1 信号量的概念	322
8.3.2 会话期	256	10.6.2 信号量集的相关操作	323
8.3.3 控制终端	258	10.7 本章小结	329
8.4 本章小结	259	实战演练	330
实战演练	259	第 11 章 线程控制	331
第 9 章 信号	261	11.1 线程的基本概念	332
9.1 Linux 信号简介	262	11.1.1 Linux 线程简介	332
9.1.1 信号的基本概念	262	11.1.2 线程的标识符	333
9.1.2 信号处理机制	265	11.1.3 用户态和核心态线程	333
9.2 信号操作的相关函数	268	11.1.4 线程的属性	334
9.2.1 信号的处理	268	11.2 线程控制的相关函数	334
9.2.2 信号的发送	276	11.2.1 pthread_create 函数	334
9.2.3 信号的阻塞	284	11.2.2 pthread_exit 函数	336
9.2.4 计时器与信号	287	11.2.3 pthread_join 函数	336
9.3 本章小结	288	11.2.4 pthread_cancel 函数	338
实战演练	289	11.2.5 pthread_cleanup_push 和 pthread_cleanup_pop 函数	338
第 10 章 进程间通信	291	11.2.6 pthread_detach 函数	340
10.1 进程间通信简介	292	11.2.7 线程和进程操作函数对比	342
10.2 管道	293	11.3 线程之间的通信和同步	342

11.3.1 互斥锁	343
11.3.2 条件变量	346
11.4 本章小结	351
实战演练	351
第 12 章 网络编程	353
12.1 网络编程的基础知识	354
12.1.1 计算机网络体系结构	354
12.1.2 传输控制协议 TCP	358
12.1.3 用户数据报协议 UDP	361
12.1.4 客户机/服务器模式	361
12.2 套接口编程基础	362
12.2.1 什么是套接口	362
12.2.2 端口号的概念	363
12.2.3 套接口的数据结构	364
12.2.4 基本函数	365
12.3 TCP 套接口编程	368
12.3.1 TCP 套接口通信工作流程	369
12.3.2 TCP 套接口 Client/Server 程序实例	382
12.4 UDP 套接口编程	387
12.4.1 UDP 套接口通信工作流程	387
12.4.2 UDP 套接口 Client/Server 程序实例	388
12.5 原始套接口编程	391
12.5.1 原始套接口的创建	392
12.5.2 原始套接口程序实例	392
12.6 本章小结	403
实战演练	403
第 13 章 Linux 图形界面编程	405
13.1 Linux 下的图形界面编程简介	406
13.1.1 Qt 简介	406
13.1.2 GTK+简介	406
13.2 界面基本元件	408
13.2.1 一个简单的例子	409
13.2.2 窗口	410

13.2.3 标签	413
13.2.4 按钮	414
13.2.5 文本框	415
13.3 界面布局元件	418
13.3.1 表格	418
13.3.2 框	421
13.3.3 窗格	424
13.4 其他常用元件	426
13.4.1 进度条、微调按钮、 组合框	426
13.4.2 单选按钮、复选按钮	430
13.4.3 下拉菜单	432
13.5 信号与回调函数	435
13.6 本章小结	437
实战演练	438

第III部分 实战篇

第 14 章 设计 Linux 下的计算器	443
14.1 软件功能分析	444
14.2 程序模块的划分	445
14.3 软件的具体实现	447
14.3.1 头文件	447
14.3.2 十六进制界面显示函数	448
14.3.3 十进制界面显示函数	449
14.3.4 八进制界面显示函数	450
14.3.5 二进制界面显示函数	452
14.3.6 进制间转换函数	453
14.3.7 信号处理模块	456
14.3.8 主函数	465
14.4 软件使用效果演示	472
14.5 本章小结	473
第 15 章 Linux 平台下聊天软件的设计	475
15.1 软件功能概述	476
15.1.1 服务器端功能需求	476
15.1.2 客户端功能需求	477

15.1.3	错误处理需求	478	16.3.1	连接界面	509
15.2	Glade 集成开发工具简介	478	16.3.2	主界面	511
15.3	软件功能模块划分	479	16.4	本章小结	513
15.3.1	服务器功能模块划分	479	第 17 章	Linux 下简易防火墙软件的设计	515
15.3.2	客户端功能模块划分	480	17.1	Netfilter 基础	516
15.3.3	消息标识的定义	480	17.1.1	什么是 Netfilter	516
15.3.4	消息结构体的设计	481	17.1.2	Netfilter 的 HOOK 机制	517
15.4	服务器程序的具体实现	482	17.1.3	HOOK 的调用	520
15.4.1	服务器消息处理流程	482	17.1.4	HOOK 的实现	521
15.4.2	服务器主要函数和变量	483	17.1.5	IPTables 简介	523
15.4.3	服务器消息处理模块的设计 与实现	484	17.1.6	Netfilter 可以实现的 控制功能	524
15.4.4	服务器数据存储的方法	485	17.2	软件设计概述	525
15.4.5	用户注册流程	485	17.2.1	软件整体框架	525
15.5	客户端程序的具体实现	485	17.2.2	管理端的设计	527
15.5.1	客户端操作流程	486	17.2.3	控制端的设计	528
15.5.2	客户端发送和接收消息 流程	486	17.3	用 Netfilter 设计控制端 功能模块	530
15.5.3	客户端主要函数和变量	487	17.3.1	ICMP 管理控制模块	530
15.5.4	客户端功能模块的设计 与实现	488	17.3.2	FTP 管理控制模块	532
15.6	聊天软件使用效果演示	489	17.3.3	HTTP 管理控制模块	533
15.7	本章小结	493	17.3.4	模块的编译、加载与卸载	534
第 16 章	Linux 远程管理工具的设计	495	17.4	软件功能测试	536
16.1	软件功能概述	496	17.5	本章小结	538
16.1.1	Webmin 简介	496	第 18 章	基于 Linux 的嵌入式家庭网关 远程交互操作平台的设计	539
16.1.2	软件总体设计	496	18.1	嵌入式技术简介	540
16.2	服务器端程序设计	497	18.1.1	嵌入式系统的概念	540
16.2.1	服务器端工作流程	498	18.1.2	嵌入式操作系统	541
16.2.2	系统用户管理操作	498	18.1.3	嵌入式处理器	542
16.2.3	系统用户组的操作	500	18.2	家庭网关的概念及其网络 体系结构	543
16.2.4	系统服务启动管理	503	18.2.1	智能家庭网络的概念	544
16.2.5	DNS 管理操作	504	18.2.2	家庭网关的远程交互 操作技术简介	544
16.2.6	Apache 服务管理操作	505			
16.2.7	FTP 服务管理操作	508			
16.3	客户端程序	509			

18.2.3 嵌入式家庭网关的网络 体系结构	545	18.4.3 通用网关接口 CGI	553
18.3 嵌入式家庭网关的开发平台	546	18.5 Linux 下软件模块的具体实现	554
18.3.1 S3C2410 微处理器简介	546	18.5.1 登录验证模块	555
18.3.2 交叉编译环境的建立	548	18.5.2 串口通信模块	555
18.4 远程交互平台的设计	549	18.5.3 中央空调控制模块	556
18.4.1 应用软件的开发模式	549	18.5.4 智能水表数据采集模块	561
18.4.2 嵌入式 Web 服务器	550	18.5.5 试验结果	562
		18.6 本章小结	562

第 I 部分

基础篇

- 第 1 章 Linux 系统概述
- 第 2 章 C 语言编程基础
- 第 3 章 vim 与 Emacs 编辑器
- 第 4 章 gcc 编译器与 gdb 调试器
- 第 5 章 make 的使用和 Makefile 的编写

第 1 章

Linux 系统概述

Linux 是一套免费使用和自由传播的类 UNIX 操作系统，已发展成为现今世界上最流行的操作系统之一。几乎每天 Linux 都以某种方式出现在媒体上，我们已经数不清在 Linux 上有多少应用程序，以及有多少机构在使用 Linux。本章将简单介绍 Linux 操作系统的基础知识，以帮助读者对所要学习的知识建立起清晰的认识，为以后的学习打下坚实的基础。



本章内容：

- ◎ 什么是 Linux。
- ◎ Linux 系统特点及主要功能。
- ◎ Linux 的内核版本和发行版本。
- ◎ Linux 系统的安装。
- ◎ Shell 的使用。
- ◎ Linux 下的常用命令。

1.1 什么是 Linux

Linux 是一种免费的，提供源代码的，能适用于 PC 机的类似于 UNIX 的网络操作系统，它主要用于基于 Intel x86 系列 CPU 的计算机上。这个系统是由世界各地成千上万的程序员设计和实现的，其目的是建立不受任何商品化软件的版权制约的、全世界都能自由使用的 UNIX 兼容产品。

Linux 操作系统是由 UNIX 发展而来，1969 年由 Ken Thompson 和 Dennis Ritchie 在美国贝尔实验室开发的一种操作系统。由于其良好而稳定的性能迅速在计算机中得到广泛的应用，在随后几十年中也有不断的改进。

UNIX 操作系统正式发布于 1974 年美国计算机学会的杂志 ACM 上，到 1975 年引入了多项技术，从而使它成为一个真正的多用户分时操作系统。此后短短两年时间，又出现了 Xenix、SUNOS 等 UNIX 操作系统的不同版本。1985 年美国麻省理工学院在已有的基础上开发出了 UNIX 操作系统的图形化界面 X Window 系统，它已经成为工作站图形界面的标准。

在 20 世纪 80 年代，Andrew S.Tanenbaum 为了满足教学的需要编写了一个与 UNIX 类似的 Minix 系统。1990 年，芬兰人 Linus Torvalds 接触了 Minix 系统后，开始着手研究编写一个开放的与 Minix 系统兼容的操作系统。1991 年 10 月 5 日，Linus Torvalds 在赫尔辛基技术大学的一台 FTP 服务器上发布了一个消息，这也标志着 Linux 系统的诞生。

1984 年，自由软件的积极提倡者 Richard Stallman 组织开发了一个完全基于自由软件的软件体系 GNU，并拟定了一份通用公共许可证(General Public License, GPL)。GPL 的内容主要是保持软件的免费使用和传播，要求必须以源代码的形式发布软件，并且任何使用者都可以以源代码的形式复制或传播软件给任何人。Linus Torvalds 在 1993 年将 Linux 系统转向 GPL，并加入了 GNU。从而最终使自由软件有了发展根基，即基于 Linux 系统的 GNU。这一版权除了规定有自由软件的各项许可权外，还允许用户出售自己的程序拷贝。

2005 年，此时的 Linux 已经发展到了 2.4 版本，该版本的 Linux 内核提供了对 USB、PC 卡、ISA、蓝牙、RAID 和 EXT3 文件系统等的支持；2.6 版本的 Linux 内核则进一步提供了对 PAE、64 位处理器、16TB 大容量存储器以及 EXT4 文件系统等的支持。

截止到 2014 年，在这段时间内 Linux 的发行版呈现了爆炸式的增强，桌面环境 KDE 发布了 KDE 4.2 版，而桌面环境 GNOME 则发布了 GNOME 3 版本，一个全新的桌面环境 Unity 在 Ubuntu 的 11.04 发行版上出现，而 Linux 内核也发布到了 3.13.3 版(2014 年 2 月 13 日)。在这几年中的另外一个突破则是 Linux 被大量地移植到基于 ARM 等处理器的嵌入式系统中，而基于 Linux 内核的移动端商用操作系统 Android(安卓)也在 2009 年 9 月发布。

绝大多数基于 Linux 内核的操作系统使用了大量的 GNU 软件，包括 shell 程序、工具、程序库、编译器及工具，还有许多其他程序，例如 Emacs。正因为如此，GNU 计划的开创者 Richard Stallman 博士提议将 Linux 操作系统改名为 GNU/Linux。但有些人只把操作系统叫作“Linux”。