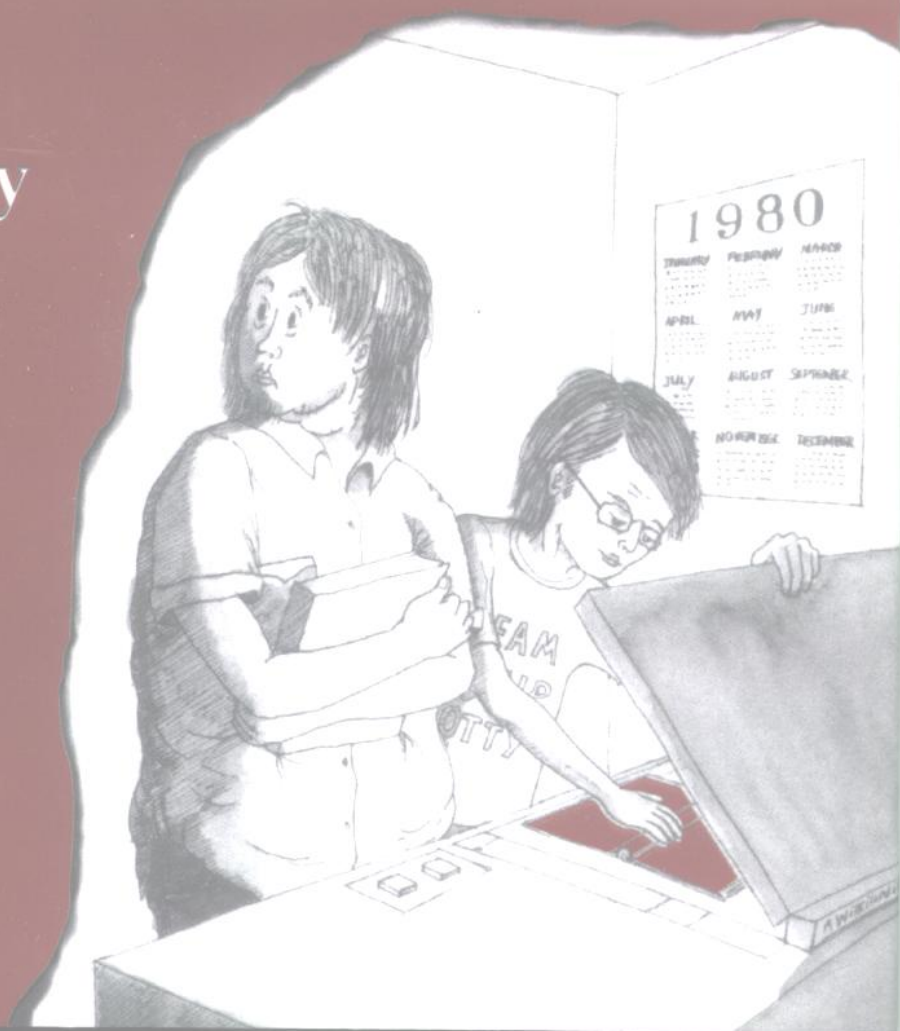


莱昂氏 UNIX 源代码分析

(澳) John Lions 著 尤晋元 译

Lions'
Commentary
on UNIX
6th Edition
with Source
Code



机械工业出版社
China Machine Press



莱昂氏UNIX源代码分析

(澳) John Lions 著

尤晋元 译



机械工业出版社
China Machine Press



Z089111

55106/10
本书由上、下两篇组成。上篇为UNIX版本6的源代码，下篇是莱昂先生对UNIX操作系统版本6源代码的详细分析。本书语言简洁、透彻，曾作为未公开出版物广泛流传了二十多年，是一部杰出经典之作。本书适合UNIX操作系统编程人员、大专院校师生学习参考使用。

John Lions: Lions' Commentary on UNIX 6th Edition with Source Code.

Authorized translation from the English language edition published by Peer-to-Peer Communications, Inc.

Selection and compilation of materials in this volume Copyright © 1996 by Peer-to-Peer Communications, Inc. All rights reserved.

A Commentary on the UNIX Operating System, Copyright © 1997 by John Lions. All rights reserved.

本书中文简体字版由美国Peer-to-Peer公司授权机械工业出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

本书版权登记号：图字：01-2000-0082

图书在版编目(CIP)数据

莱昂氏UNIX源代码分析/(澳)莱昂氏(Lions, J.)著；尤晋元译. -北京：机械工业出版社，2000.7

书名原文：Lions' Commentary on UNIX 6th Edition with Source Code
ISBN 7-111-08018-1

I. 莱… II. ①莱… ②尤… III. UNIX操作系统-源语言-码 IV. TP316.81

中国版本图书馆CIP数据核字(2000)第20647号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑：赵红燕

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2000年7月第1版·2000年9月第2次印刷

787mm×1092mm 1/16·24.75印张

印数：8 001-13 000册

定价：49.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

献 辞

谨以此书献给UNIX软件系统的创建者肯·汤姆森和丹尼斯·里奇，由于他们的工作，才使UNIX源代码及其分析的编写成为可能；献给新南威尔士大学昔日的同事和学生，他们帮助我准备了本书的初稿；同时，也献给我的妻子玛丽安妮、女儿凯瑟琳和伊丽莎白，她们并不了解UNIX，但是对我的写作工作给予了矢志不渝的支持。

莱昂氏著作 本书由上、下两篇组成，上篇是UNIX 版本6 内核部分的完整源代码；下篇是对该源代码的分析，其中讨论了各种算法。从1976年开始，此书在新南威尔士大学内部流通，数年之后成为贝尔实验室以外任何人都可得到的详细分析UNIX 内核的文献。由于Western Electric公司希望保守在UNIX内核方面的商业秘密，本书未能正式出版，并且只允许向UNIX许可证的持有者发放。尽管如此，本书仍然在大量早期UNIX“黑客”中得到传播。

——出自埃里克.S.雷蒙著《新黑客词典》第2版

目录摘要

致谢

序(一)

序(二)

历史注记

UNIX操作系统版本6源代码

莱昂氏UNIX源代码分析

致 谢

对于给予帮助，使本书在历经20年后终于出版的所有人士，谨表衷心的感谢。

我要感谢多恩贝格和Peer-to-Peer出版公司的员工，他们对此次出版给予大力支持。同时要感谢SCO公司，SCO遵循UNIX发展的宗旨，惠允公开这些文本，并作出免除那些繁文缛节的积极安排。

对于肯·汤姆森、彼得·柯林森、麦克·奥德尔、彼得·雷杰斯、格里格·罗斯和麦克尔·泰尔森等人，我要表示特别的感谢，他们以其所作的高度评价而使本书增色。

最后，我对彼得·萨鲁斯和丹尼斯·里奇深表感激之情，他们两人为本书的出版不辞辛劳，并不遗余力地排除法律上的各种障碍。同样感激古德哈特，他以始终如一的支持和宝贵的友情，帮助我实现此书的出版。

这本小书的初版成书之后，在1978、1983和1989年，我曾有幸三次应邀赴贝尔实验室休年假，并以研究小组成员的身份参与那里的工作。在贝尔实验室期间，众多同事们给予我的支持和友谊，终身难以忘怀。在那个时期，在贝尔实验室的同事和我在澳大利亚的同事之间，建立起了牢固的联系，这促成了开发UNIX系统的几位先驱对澳大利亚的工作访问，并在各种会议上发表演说。澳大利亚UNIX用户组对此项活动给予高度评价，也因此而受益匪浅。

约翰·莱昂

序(一)

当这本书首次发行时，从阅读这样一本单纯技术性的文献中能够感受到如此大的愉快，曾令我惊讶。约翰·莱昂确实创造了一件杰出的技术作品。UNIX系统的核心代码原本就是精致之作，即使在今天，仍然值得人们去研究。约翰对UNIX的这些源代码一行一行地作了分析，再加上同样堪与媲美的注释。源代码和注释结合天成，相映生辉。我还从未见过可以与此相提并论的任何其他技术成就。整整一代操作系统的开发人员，曾把这一著作当成主要学习工具；而我所奇怪的，还在于它对确立UNIX在计算机工作站、互联网络服务器和关键业务计算环境中的统治地位，何以能够产生如此大的影响。但与此同时，它的广泛流行也受到限制，因为它是一种对特许专用源代码的诠释文本。我极其高兴这一著作如今能够变成大众手中的读物。作为我们曾经走过的技术发展历史的一部分，今天仍然可以从本书中获得教益。

迈克尔·泰尔森

(泰尔森是UniForum Association的总裁和SCO公司的首席信息官。)

1993年之初，我开始准备UNIX操作系统诞生25周年的纪念活动。这个系统是由丹尼斯·里奇、肯·汤姆林、道格·麦克罗伊和鲁德·卡那地于1969年夏天创建的。

我盼望到1994年夏天能够实现的一件事情，就是这卷UNIX文献中的材料正式出版。届时，约翰·莱昂会到波士顿去参加USENIX协会在那里举行的庆祝活动，并接受一项颁发给他的终身成就奖。要是能对他呈献上这本书的话，无疑是一件最适宜不过的礼物。

但遗憾的是，这一愿望终于未能实现。

在丹尼斯的多方奔走之下，AT&T的律师们才表示他们“不反对”

此书的出版。但是为出版此书而同Novell举行的谈判，进展迟缓，Novell是从AT&T那里购得UNIX系统的买主。由于此种原因，拖到1995年底，SCO公司才宣布获准从Novell公司买断UNIX。闻讯后，我和丹尼斯立即写信给迈克尔·泰尔森和道格·米歇尔斯，他们两人都是SCO公司的董事，我们与之有私人交往。那时，迈克尔·泰尔森自己实际上拥有一本莱昂著作的拷贝，并珍藏着它，而且在一段很短的时期，同SCO的律师达成允许使用这份材料的安排。

在20世纪70年代和80年代这一时期，莱昂的几册《源代码分析》，曾是人们争相传看的UNIX系统的“地下出版物”。我们今天来审视这一著作，尽管其中的代码已属过时，但其注释者所作的大部分注释，依然完美如初。要想学习操作系统原理，就必须阅读和理解源代码。莱昂的工作，使我们中间的大多数人都能达到这一步。

在1969年，AT&T是电信垄断企业的一个复合体，包括了贝尔电话实验室和Western Electric公司。前者为我们提出了许多发展研究项目，而UNIX操作系统就是其中的一个。后者则致力于产品制造和产品经销。对于《源代码分析》中的引文，我们保留其1977年时的原有形式。贝尔实验室今天已是朗讯公司的一部分；Western Electric公司就是现时的AT&T技术公司。

我以能为本书的最终付梓效力而感到自豪。至此，我可以停止使用我的经过多道复印的源代码分析拷贝了。

彼得·H.萨鲁斯

(萨鲁斯是USENIX和Sun用户组前执行主任和《重访计算机精英》丛书主编。)

序(二)

正式出版约翰·莱昂编著的《莱昂氏UNIX源代码分析》标志着一个悠长故事的圆满结束。

UNIX的传播总是伴随着变化着的商业世界和研究及学术社团利益之间的不相一致和对立的观点。70年代和80年代期间，在系统源代码的分发方面非常自由的政策鼓励了很多创造性的开发工作，特别是在加州大学伯克利分校所进行的工作对于构造今日的Internet和工作站产业是非常重要的。同时，法人团体的管理人员则一直担心他们的企业和社团会失去一切权利。

莱昂源代码分析的故事也说明了这种紧张的关系。当肯·汤姆森和我两人见到这两本包含注释和源代码的册子时(原书分两册)，都为其及时出现而感到欣慰，同时也为他的深刻的洞察力和在课程安排方面的技巧而留下深刻印象。UNIX支持组与我们有同感。他们重印了这两本书并用于贝尔实验室的训练课程。大约两年之后，莱昂应邀到UNIX支持组工作一年，担任顾问并进行写作。贝尔实验室和UNIX社会对以莱昂为核心的一群澳大利亚人的聪明才智表示赞赏并因之受益，他们集中在莱昂工作的新南威尔士大学，有些则来自悉尼大学和另外一些地方。在澳大利亚以外，也有遍布世界的大量人群通过阅读这本书学习操作系统。

但是对立的观点仍然十分明显：本书的重要价值和叙述的生动活泼也引起了关注，由此而来的是UNIX 版本7的许可证注明禁止使用其源代码用作教学材料。尽管如此，本书、版本7源代码及为VAX体系结构开发的后继者的源代码(贝尔实验室开发的32位版本及各种BSD系统)仍然是处处可见并在80年代早期的重要开发中得到应用。

不同的观点基本围绕着下面的问题：在公开出版软件及对软件的注释，以及鼓励有关人员学习它们的同时又如何仍能保持商业和技术方面的控制，这一问题至今仍未解决，并且无疑并不存在解决方案。

自由软件基金会希望所有软件的源代码都是可以使用的。学术界的大多数人基本上同意此种观点，但是其中也有相当一部分人希望保持权利并从中获利。工商业界则希望捍卫权利并谋取尽可能多的利润。在过去的很多年中，UNIX在此方面激起了波涛。

令人感到欣慰的是，SCO——当前提供UNIX操作系统和应用程序的领导者，认为出版莱昂的分析及相关源代码具有历史和教育方面的价值，并同意其公开出版。我为此向该公司表示感谢。

本书中的材料确定是有点过时了，其内容不包括图形、网络以及1975年后出现的新事物。即使在1979年的编译中也不再使用的线性搜索、基本数据结构和C代码还保留在本书的源代码中，而该代码面向的计算机非常简单，只比存储器稍稍多一些。从中你会发现有很多粗糙之处。但是在代码中你也可以观察到其基本结构，该结构沿用了很长时间，而且能够包容在计算环境中发生的巨大变化。在莱昂的分析中，你可以觉察到新鲜的、经常提出问题的立场，其中的很多词语和思想都很适宜于教育和学习。莱昂非常清楚地赞赏他阅读的UNIX源代码，但又立即指出其不足之处。他帮助他的学生理解蕴含在源代码中的很多思想和主题，也坦率地说明他对该代码某些部分感到的困惑之处。

此处重印的文献已经教育了一代人，它是计算机界中复印数量最多的一本书稿。将此公开记录在案是件大好事。

丹尼斯.M.里奇

最后我还想提及的是：现在，得到最广泛传播的一份地下计算机科学文献已经可以自由地使用了。我对1977年中的那一天仍然是记忆犹新，那时我接到邮寄来的此书的第一份草稿，开始时我对此书并无很多期望，但是最后却是逐字逐句进行了仔细阅读。

20年之后，此书依旧是对一个实际操作系统工作的最好分析说明。

肯·汤姆森

(汤姆森和里奇因为开发和实现UNIX操作系统而共同获得1983年ACM美国计算机协会图灵奖)。

历史注记

1977年，约翰·莱昂写完了他的《UNIX操作系统源代码分析》，成为他重新整理的UNIX版本6源代码的姐妹篇。这两本篇幅不大的书，也许算得上计算机业界未获出版的最重要著作。我请求作者评论《源代码》和《分析》这两本书。他说道：

“我是讲授操作系统的教师；曾经同一位讲授编译原理的同事竞赛，进行代码阅读练习看来是一种好办法；我们的UNIX许可证正好也不明确限制这样的活动。那么何以选择用UNIX呢？因为别无更多的选择余地。如你所知，这是最富竞争能力的活动，而且比竞赛更有意义（我们也已获得了布林奇·汉森的SOLO系统）。”

《UNIX新闻》在1977年3月的第3卷第3期上宣布，持许可证者可以索取该书，同时附有一则费伦兹的评论：“肯·汤姆森已拜读了该书第1版，并说是一本优秀的著作。”书的售价含航寄邮资在内为17.70澳元（当时不到20美元）。英国UKUUG通信宣布了可以求购代码和分析，但是，在其下一期则声明：日后订单需提交贝尔实验室，并且两本书的发行期限是1978年底。在整个计算机科学界，它们无疑是复印最为频繁的两本书。它们载有相应的版权通告和仅限许可证持有者使用的字样，但是再次显示，Western Electric公司无法阻止如此重要著作的某种形式的流行。我承认同时拥有经过多次复印的拷贝和约翰·莱昂赠送给我的桔黄色的和红色的拷贝。

摘自彼得·萨鲁斯的《UNIX25年记》一书。

目 录

献辞	
致谢	
序(一)	
序(二)	
历史注记	

上篇 UNIX操作系统版本6源代码

UNIX操作系统过程分类索引	3
UNIX操作系统文件及过程	5
UNIX操作系统定义的符号列表	7
UNIX操作系统源代码交叉引用列表	9
第一部分 初始化、进程初始化	25
第二部分 陷入、中断、系统调用和 进程管理	75
第三部分 程序交换、基本输入/输出、 块设备	109
第四部分 文件和目录、文件系统、管道	133
第五部分 面向字符的特殊文件	181

下篇 莱昂氏UNIX源代码分析

前言	207
第1章 绪论	209
1.1 UNIX操作系统	209
1.2 公用程序	209
1.3 其他文档	210
1.4 UNIX程序员手册	210
1.5 UNIX文档	211
1.6 UNIX操作系统源代码	211
1.7 源代码中各部分	212
1.8 源代码文件	212
1.9 分析的使用	212
1.10 对程序设计水平的一条注释	212

第2章 基础知识	214
2.1 处理机	214
2.2 处理机状态字	214
2.3 通用寄存器	214
2.4 指令集	215
2.5 寻址方式	216
2.5.1 寄存器方式	217
2.5.2 寄存器延迟方式	217
2.5.3 自动增1方式	217
2.5.4 自动减1方式	217
2.5.5 变址方式	217
2.5.6 立即方式	218
2.5.7 相对方式	218
2.6 UNIX汇编程序	219
2.7 存储管理	219
2.8 段寄存器	220
2.9 页说明寄存器	220
2.10 存储分配	220
2.11 状态寄存器	221
2.12 “i”和“d”空间	221
2.13 启动条件	221
2.14 专用设备寄存器	221
第3章 阅读“C”程序	222
3.1 某些选出的例子	222
3.2 例1	222
3.3 例2	223
3.4 例3	223
3.5 例4	225
3.6 例5	225
3.7 例6	227
3.8 例7	227
3.9 例8	228

3.10 例9	228	6.3 main(1550)	251
3.11 例10	229	6.4 进程	252
3.12 例11	229	6.5 proc〔0〕的初始化	252
3.13 例12	230	6.6 sched(1940)	253
3.14 例13	230	6.7 sleep(2066)	253
3.15 例14	231	6.8 swtch(2178)	253
3.16 例15	231	6.9 再回到main	254
3.17 例16	232	第7章 进程	256
3.18 例17	233	7.1 进程映像	256
第4章 概述	235	7.2 proc结构(0358)	257
4.1 变量分配	235	7.3 user结构(0413)	257
4.2 全局变量	235	7.4 每个进程数据区	258
4.3 “C”预处理程序	235	7.5 段	258
4.4 第一部分	236	7.6 映像的执行	258
4.4.1 第1组“.h”文件	236	7.7 核心态执行	259
4.4.2 汇编语言文件	237	7.8 用户态执行	259
4.4.3 在第一部分中的其他文件	237	7.9 一个实例	259
4.5 第二部分	237	7.10 设置段寄存器	260
4.6 第三部分	238	7.11 estabur(1650)	260
4.7 第四部分	238	7.12 sureg(1739)	261
4.8 第五部分	239	7.13 newproc(1826)	261
第一部分 初始化、进程初始化		第8章 进程管理	263
第5章 两个文件	241	8.1 进程切换	263
5.1 文件malloc.c	241	8.2 中断	263
5.1.1 列表维护规则	241	8.3 程序交换	263
5.1.2 malloc(2528)	242	8.4 作业	264
5.1.3 mfree(2556)	243	8.5 汇编语言过程	264
5.1.4 结论	244	8.6 savu(0725)	264
5.2 文件prf.c	244	8.7 retu(0740)	264
5.2.1 printf(2340)	244	8.8 aretu(0734)	264
5.2.2 printn(2369)	245	8.9 swtch(2178)	265
5.2.3 putchar(2386)	246	8.10 setpri(2156)	265
5.2.4 panic(2419)	247	8.11 sleep(2066)	266
5.2.5 prdev(2433)、deverror(2447)	247	8.12 wakeup(2133)	266
5.3 包含的文件	247	8.13 setrun(2134)	266
第6章 系统初启	249	8.14 expand(2268)	267
6.1 操作员的动作	249	8.15 再回到swtch	267
6.2 start(0612)	249		

8.16 临界区	268	13.1 设置期望动作	288
第二部分 陷入、中断、系统调用 和进程管理		13.2 对进程造成中断	288
第9章 硬件中断和陷入	269	13.3 作用	289
9.1 硬件中断	269	13.4 跟踪	289
9.2 中断矢量	270	13.5 过程	289
9.3 中断处理程序	270	13.5.1 期望动作的设置	289
9.4 优先级	270	13.5.2 造成软件中断	289
9.5 中断优先级	271	13.5.3 作用	289
9.6 中断处理程序的规则	271	13.5.4 跟踪	290
9.7 陷入	272	13.6 ssig(3614)	290
9.8 汇编语言trap	272	13.7 kill(3630)	290
9.9 返回	273	13.8 signal(3949)	290
第10章 汇编语言“trap”例程	274	13.9 psignal(3963)	291
10.1 陷入和中断源	274	13.10 issig(3991)	291
10.2 fuibyte(0814)与fuiword(0844)	274	13.11 psig(4043)	291
10.3 中断	275	13.12 core(4094)	292
10.4 call(0776)	275	13.13 grow(4136)	292
10.5 用户程序陷入	276	13.14 exit(3219)	292
10.6 核心态栈	277	13.15 rexit(3205)	293
第11章 时钟中断	279	13.16 wait(3270)	293
11.1 clock(3725)	279	13.17 跟踪	293
11.2 timeout(3845)	281	13.18 stop(4016)	294
第12章 陷入与系统调用	282	13.19 wait(3270)(继续)	294
12.1 trap(2693)	282	13.20 ptrace(4164)	295
12.2 核心态陷入	282	13.21 procxmt(4204)	295
12.3 用户态陷入	283	第三部分 程序交换、基本输入 /输出、块设备	
12.4 系统调用	284	第14章 程序交换	297
12.5 系统调用处理程序	285	14.1 正文段	297
12.6 文件sys1.c	285	14.2 sched(1940)	298
12.6.1 exec(3020)	285	14.3 xswap(4368)	299
12.6.2 fork(3322)	286	14.4 xalloc(4433)	299
12.6.3 sbreak(3354)	286	14.5 xfree(4398)	300
12.7 文件sys2.c和sys3.c	287	第15章 基本输入/输出介绍	301
12.8 文件sys4.c	287	15.1 buf.h文件	301
第13章 软件中断	288	15.2 devtab(4551)	301

15.3 conf.h文件301

15.4 conf.c文件302

15.5 系统生成302

15.6 swap(5196)302

15.7 竞态条件303

15.8 可重入304

15.9 继续分析 “u.u_ssav”304

第16章 RK磁盘驱动器305

16.1 控制状态寄存器RKCS306

16.2 字计数寄存器RKWC306

16.3 磁盘地址寄存器RKDA306

16.4 rk.c文件306

16.5 rkstrategy(5389)306

16.6 rkaddr(5420)307

16.7 devstart(5096)307

16.8 rkintr(5451)307

16.9 iodone(5018)308

第17章 缓存处理309

17.1 标志309

17.2 一个类超高速缓存存储309

17.3 clrbuf(5038)309

17.4 incore(4899)310

17.5 getblk(4921)310

17.6 brelse(4869)310

17.7 binit(5055)311

17.8 bread(4754)312

17.9 breada(4773)312

17.10 bwrite(4809)312

17.11 bawrite(4856)313

17.12 bdwrite(4836)313

17.13 bflush(5229)313

17.14 physio(5259)313

**第四部分 文件和目录、文件
系统、管道**

第18章 文件存取和控制315

18.1 源代码第四部分315

18.2 文件特征315

18.3 系统调用316

18.4 控制表316

18.4.1 file(5507)316

18.4.2 inode(5659)316

18.5 要求专用的资源317

18.6 打开一个文件317

18.7 creat(5781)317

18.8 open1(5804)317

18.9 open(5763)318

18.10 再回到open1318

18.11 close(5846)318

18.12 closef(6643)319

18.13 iput(7344)319

18.14 删除文件319

18.15 读和写文件319

18.16 rdwr(5731)320

18.17 readi(6221)321

18.18 writei(6276)322

18.19 iomove(6364)322

18.20 bmap(6415)322

18.21 剩余部分322

第19章 文件目录和目录文件323

19.1 文件名323

19.2 目录数据结构323

19.3 目录文件323

19.4 namei(7518)324

19.5 一些注释325

19.6 link(5909)326

19.7 wdir(7477)327

19.8 maknode(7455)327

19.9 unlink(3510)327

19.10 mknod(5952)327

19.11 access(6746)328

第20章 文件系统329

20.1 超级块(5561)329

20.2 mount表(0272)329

20.3	iinit(6922)	330	第23章 字符处理	345	
20.4	安装	330	23.1	cinit(8234)	346
20.5	smount(6086)	330	23.2	getc(0930)	346
20.6	注释	331	23.3	putc(0967)	347
20.7	iget(7276)	331	23.4	字符集	347
20.8	getfs(7167)	332	23.5	图形字符	348
20.9	update(7201)	332	23.6	UNIX惯例	349
20.10	sumount(6144)	333	23.7	maptab(8117)	349
20.11	资源分配	333	23.8	partab(7947)	349
20.12	alloc(6956)	334	第24章 交互式终端	351	
20.13	itrunc(7414)	334	24.1	接口	351
20.14	free(7000)	335	24.2	tty结构(7926)	351
20.15	iput(7344)	335	24.3	注释	352
20.16	ifree(7134)	335	24.4	初始化	352
20.17	iupdat(7374)	335	24.5	stty(8183)	352
第21章 管道		337	24.6	sgtty(8201)	353
21.1	pipe(7723)	337	24.7	klsgtty(8090)	353
21.2	readp(7758)	337	24.8	ttystty(8577)	353
21.3	writep(7805)	338	24.9	DL11/KL11终端设备处理程序	353
21.4	plock(7862)	338	24.10	设备寄存器	354
21.5	prele(7882)	338	24.11	接收器状态寄存器	354
第五部分 面向字符的特殊文件			24.12	接收器数据缓存寄存器	354
第22章 面向字符的特殊文件		339	24.13	发送器状态寄存器	354
22.1	LP11行式打印机驱动程序	339	24.14	发送器数据缓存寄存器	354
22.2	lpopen(8850)	340	24.15	单总线地址	354
22.3	注释	340	24.16	软件方面的考虑	355
22.4	lpoutput(8986)	340	24.17	中断矢量地址	355
22.5	lpstart(8967)	341	24.18	源代码	355
22.6	lpint(8976)	341	24.19	klopen(8023)	355
22.7	lpwrite(8870)	342	24.20	klclose(8055)	356
22.8	lpclose(8863)	342	24.21	klxint(8070)	356
22.9	讨论	342	24.22	klrint(8078)	356
22.10	lpcanon(8879)	342	第25章 tty.c文件	357	
22.11	对读者的建议	343	25.1	flushatty(8252)	357
22.12	PC11纸带阅读机/穿孔机驱动		25.2	wflushatty(8217)	357
程序		344	25.3	字符输入	358
			25.3.1	ttread(8535)	358