Table of Contents

Preface xi

1. An Introduction to Device Drivers 1
The Role of the Device Driver 2
Splitting the Kernel 4
Classes of Devices and Modules 5
Security Issues 8
Version Numbering 10
License Terms 11
Joining the Kernel Development Community 12
Overview of the Book 12

2. Building and Running Modules 15
Setting Up Your Test System 15
The Hello World Module 16
Kernel Modules Versus Applications 18
Compiling and Loading 22
The Kernel Symbol Table 28
Preliminaries 30
Initialization and Shutdown 31
Module Parameters 35
Doing It in User Space 37
Quick Reference 39

3. Char Drivers 42
The Design of scull 42
Major and Minor Numbers 43
Some Important Data Structures 49

Char Device Registration 55
open and release 58
scull's Memory Usage 60
read and write 63
Playing with the New Devices 70
Quick Reference 70

4. Debugging Techniques 73
Debugging Support in the Kernel 73
Debugging by Printing 75
Debugging by Querying 82
Debugging by Watching 91
Debugging System Faults 93
Debuggers and Related Tools 99

5. Concurrency and Race Conditions 106
Pitfalls in scull 107
Concurrency and Its Management 107
Semaphores and Mutexes 109
Completions 114
Spinlocks 116
Locking Traps 121
Alternatives to Locking 123
Quick Reference 130

6. Advanced Char Driver Operations 135
ioctl 135
Blocking I/O 147
poll and select 163
Asynchronous Notification 169
Seeking a Device 171
Access Control on a Device File 173
Quick Reference 179

7. Time, Delays, and Deferred Work 183
Measuring Time Lapses 183
Knowing the Current Time 188
Delaying Execution 190
Kernel Timers 196
Tasklets 202

Workqueues 205
Quick Reference 208

8. Allocating Memory 213
The Real Story of kmalloc 213
Lookaside Caches 217
get_free_page and Friends 221
vmalloc and Friends 224
Per-CPU Variables 228
Obtaining Large Buffers 230
Quick Reference 231

9. Communicating with Hardware 235
I/O Ports and I/O Memory 235
Using I/O Ports 239
An I/O Port Example 245
Using I/O Memory 248
Quick Reference 255

10. Interrupt Handling 258
Preparing the Parallel Port 259
Installing an Interrupt Handler 259
Implementing a Handler 269
Top and Bottom Halves 275
Interrupt Sharing 278
Interrupt-Driven I/O 281
Quick Reference 286

11. Data Types in the Kernel 288
Use of Standard C Types 288
Assigning an Explicit Size to Data Items 290
Interface-Specific Types 291
Other Portability Issues 292
Linked Lists 295
Quick Reference 299

12. PCI Drivers 302
The PCI Interface 302
A Look Back: ISA 319
PC/104 and PC/104+ 322

Other PC Buses 322
SBus 323
NuBus 324
External Buses 325
Quick Reference 325

13. USB Drivers 327
USB Device Basics 328
USB and Sysfs 333
USB Urbs 335
Writing a USB Driver 346
USB Transfers Without Urbs 356
Quick Reference 360

14. The Linux Device Model 362
Kobjects, Ksets, and Subsystems 364
Low-Level Sysfs Operations 371
Hotplug Event Generation 375
Buses, Devices, and Drivers 377
Classes 387
Putting It All Together 391
Hotplug 397
Dealing with Firmware 405
Quick Reference 407

15. Memory Mapping and DMA 412
Memory Management in Linux 412
The mmap Device Operation 422
Performing Direct I/O 435
Direct Memory Access 440
Quick Reference 459

16. Block Drivers 464
Registration 465
The Block Device Operations 471
Request Processing 474
Some Other Details 491
Quick Reference 494

17. Network Drivers 497
How snull Is Designed 498
Connecting to the Kernel 502
The net_device Structure in Detail 506
Opening and Closing 515
Packet Transmission 516
Packet Reception 521
The Interrupt Handler 523
Receive Interrupt Mitigation 525
Changes in Link State 528
The Socket Buffers 528
MAC Address Resolution 532
Custom ioctl Commands 535
Statistical Information 536
Multicast 537
A Few Other Details 540
Quick Reference 542

18. TTY Drivers 546
A Small TTY Driver 548
tty_driver Function Pointers 553
TTY Line Settings 560
ioctls 564
proc and sysfs Handling of TTY Devices 566
The tty_driver Structure in Detail 567
The tty_operations Structure in Detail 569
The tty_struct Structure in Detail 571
Quick Reference 573

Bibliography 575

Index 579