Contents

Introduction xv
About This Guide xv
Audience xv
Document Overview xvi
Related Documentation xvii
Reference Material xvii
Notation and Syntax Conventions xviii

1. Introduction to Device Drivers 1
Driver Overview 2
    Device Types 2
    Levels of Device Drivers 3
System Hardware 4
    Hardware Platforms 5
    Bus Interfaces 15
System Software 18

2. Writing a Device Driver 21
Creating Device Drivers 21
    Creating User-level Device Drivers 21
    Creating Kernel-level Device Drivers 22
Device-special File 22
    Creating Device-special Files 23
    Major and Minor Device Numbers 23
    Device-special File Example 24
Configuration Files 25
Including a Device Driver in the Kernel 26

Driver Entry Points 28
- Missing Driver Entry Points 29
- Character and Block Entry Point Driver Routines 29
- Writing Other Driver Routines 42
- STREAMS Driver Entry Points 51

3. Writing a VME Device Driver 57

VME-bus Interface Overview 58
- VME-bus Adapter 59
- VME-bus Address Space 59
- VME-bus Read-Modify-Write Cycle 60
- VME-bus Adapter Requests 61
- VME-bus Interrupts 61
- Distribution of VME Interrupts on Multiprocessors 62

Choosing a Driver Model 63
- User-level VME-bus Device Driver 63
- Kernel-level VME Device Driver 64
- Kernel-level General Memory-mapping Device Driver 64

Writing User-level VME Device Drivers 64
- Example VME Device Driver 66
- Using mmap 72
- User-level DMA Library (udmalib) 75

Writing Kernel-level VME Device Drivers 78
- Determining VME Device Addresses 78
- Including VME Device Drivers in the Kernel 79
- Writing edtinit() 82
- VME Interrupt Handler 87
- Programmed I/O (PIO) 88
- DMA Operations 91
- VME Devices with Scatter/Gather Capability 93
- Using DMA Maps 95
- DMA on A24 Devices with No DMA Mapping 98
- DMA on A32 Devices with No Scatter/Gather Capability 102

4. **Writing an EISA Device Driver** 105
EISA-bus Interface Overview 106
Initialization 106
EISA-Bus Locked Cycles 107
EISA-Bus Request Arbitration 108
EISA-Bus Interrupts 108
EISA-Bus Data Transfers 108
EISA-Bus Address Space 108
Byte Swapping 110
Choosing a Device Driver Model 110
When to Write a User-level Device Driver 110
When to Write a Kernel-level Device Driver 111
When to Write a Kernel-level Memory-mapping Device Driver 111
Writing a User-level EISA Device Driver 111
User-level EISA Special Files 112
Using the mmap Operating System Function 112
Writing a Kernel-level EISA Device Driver 113
Configuring a Kernel-level EISA Device Driver 113
Determining ISA/EISA Device Addresses 114
Including EISA Drivers in the Kernel 114
Writing edtinit() 122
I/O and Memory Space Initialization 124
Dynamic Resource Allocation 125
EISA Locked Cycles 129
DMA Address Mapping 129
EISA DMA Utility Functions and Structures 130

5. **Writing a SCSI Device Driver** 131
SCSI-bus Interface Overview 132
Choosing a Driver Model 132
User-level SCSI-bus Device Driver 133
Kernel-level SCSI Device Driver 133

User-level SCSI Device Drivers 134
Creating Device-special Files for User-level SCSI Drivers 135
dsreq – User-level Driver Communication Structure 136
dslib Routines Description 143
SCSI Open and Close Driver Routines 144
SCSI Function-Building Routines – Group 1 145
SCSI Function-Building Routines – Group 2 148
Using the dslib Routines 156
Kernel-level SCSI Device Drivers 157
Configuring a Kernel-level SCSI Driver 157
Writing a SCSI init() 159
SCSI Device Interface Overview 159
Using the SCSI Access Routines 168
SCSI Device Driver Example 171

6. Writing Kernel-level GIO Device Drivers 177
GIO-bus Architecture 178
Determining GIO Device Addresses 179
Including GIO Device Drivers in the Kernel 180
Creating a System File 180
Creating a Master File 182
Creating a Boot File 182
The GBD Example 182
Writing edtinit() 183
GIO-specific Functions 187
setgiovector 187
splgio0, splgio1, splgio2 188
setgioconfig 188
GIO Interrupt Handler 189
Programmed I/O (PIO) 190
DMA Operations 192
Memory Parity Workarounds 193
GIO Devices with Hardware-supported Scatter/Gather Capability 195

DMA on GIO Devices Without Scatter/Gather Capability 198
Device Driver Example 200

7. Writing Kernel-level General Memory-mapping Device Drivers 213
Including a Memory-mapping Device Driver in the Kernel 214
Mapping and Unmapping Functions 214
Sharing Kernel Memory with a User Program 218
Example Program 219
Returning Opaque Handle Data 221

8. Writing Multiprocessor Device Drivers 223
Preliminary Considerations 224
Shared Data between Upper-half and Interrupt Routines 225
Protecting Shared Data Among Upper-half Routines 226
Semaphore and Spinlock Calls 227
Multiprocessing STREAMS Drivers 231
STREAMS Monitor 231
STREAMS Example 233

9. Writing Network Device Drivers 235
Preliminary Discussion 236
IRIX Kernel Networking Design 236
ifnet Driver Interfaces 238
Multiprocessor Issues 239
Compilation Flags for MP TCP/IP 240
Input Queueing Example 241
Interrupt Handler Example 242
ifnet Device Driver Example 243

10. **Driver Installation and Testing** 269
Using the Kernel Debugger 270
Making a Debuggable Kernel 270
Making an ASCII Terminal the Console 272
Invoking symmon 273
Displaying and Changing Registers 274
Using symmon Commands 277
Using symmon's Kernel Print Command 296
Multiprocessor Debugging 301
Configuring the System Software 301
Preparing the System Hardware 301
symmon and Multiprocessor Debugging 302
Forcing System Memory Dumps 303

11. **Kernel-level Dynamically Loadable Modules (DLMs)** 305
Module Configuration 306
Using a Dynamically Loadable Kernel Module 307
Load 307
Register 308
Unload 309
Unregister 309
List 309
Master File Configuration 310
Auto Registration 310
Auto Unload 310
Kernel Configuration 311
Module Entry Points 311
Module Initialization 312
Edt Type Drivers 312
Library Modules 312
Loadable Modules and Hardware Inventory 313
Run-time Symbol Table 313
Debugging Loadable Kernel Modules 314
Load/Register Failures 315

Example 1 316

Example 2 317

A. System-specific Issues 319

CPU Types 320

Data Cache Write Back and Invalidation 321

Flushing the Write Buffer 324

Registers and Register Optimization 324

Reliable Multiprocessor Spinlocks 324

VME Slave Addressing 325

VME Master Addressing 326

VME-bus Space Reserved for Customer Drivers 329

POWER Indigo2 and POWER CHALLENGE M Drivers 330

- Sharing Data Between CPU and Peripheral Devices 332
- Using mmap() 332

B. SCSI Controller Error Messages 333

Introduction 334

Sense Key Information 335

SCSI Driver Error Messages 340

SCSI Driver Debugging Messages 342

SCSI States and Phases 344

C. Device Driver Migration Notes 349

Introduction 349

- Intended Audience and Prerequisites 349
- Background 349
- Migration Overview 350

IRIX 4.x to 5.x Migration 352
SVR4 API Changes 352
STREAMS Changes 352
DDI Changes 352
DKI Changes 353
CHALLENGE/Onyx Family Support 354
Kernel Configuration Issues 356
Equipped Device Table (EDT) Changes 356
IRIX 5.2 to 5.3 Migration 359
Standard Device Drivers 359
STREAMS Modules 360
ifnet Drivers 360
Migration to IRIX 6.0 366
Virtual Page Size 366
ioctl Support 367

Glossary 369

Index 393