FreeBSD Device Drivers: A Guide For The Intrepid
**Synopsis**

Device drivers make it possible for your software to communicate with your hardware, and because every operating system has specific requirements, driver writing is nontrivial. When developing for FreeBSD, you've probably had to scour the Internet and dig through the kernel sources to figure out how to write the drivers you need. Thankfully, that stops now. In FreeBSD Device Drivers, Joseph Kong will teach you how to master everything from the basics of building and running loadable kernel modules to more complicated topics like thread synchronization. After a crash course in the different FreeBSD driver frameworks, extensive tutorial sections dissect real-world drivers like the parallel port printer driver. You'll learn:

- All about Newbus, the infrastructure used by FreeBSD to manage the hardware devices on your system
- How to work with ISA, PCI, USB, and other buses
- The best ways to control and communicate with the hardware devices from user space
- How to use Direct Memory Access (DMA) for maximum system performance
- The inner workings of the virtual null modem terminal driver, the USB printer driver, the Intel PCI Gigabit Ethernet adapter driver, and other important drivers
- How to use Common Access Method (CAM) to manage host bus adapters (HBAs)

Concise descriptions and extensive annotations walk you through the many code examples. Don't waste time searching man pages or digging through the kernel sources to figure out how to make that arcane bit of hardware work with your system. FreeBSD Device Drivers gives you the framework that you need to write any driver you want, now.

**Book Information**

Paperback: 352 pages
Publisher: No Starch Press; 1 edition (May 10, 2012)
Language: English
ISBN-10: 1593272049
Product Dimensions: 7 x 0.9 x 9.2 inches
Shipping Weight: 1.5 pounds (View shipping rates and policies)
Average Customer Review: 4.9 out of 5 stars See all reviews (9 customer reviews)
Best Sellers Rank: #1,249,600 in Books (See Top 100 in Books) #22 in Books > Computers & Technology > Programming > APIs & Operating Environments > Device Drivers #23 in Books > Computers & Technology > Operating Systems > BSD #426 in Books > Computers & Technology > Operating Systems > Unix

**Customer Reviews**
I already have the other book by the same author, Joseph Kong, “Designing BSD Rootkits: An
Introduction to Kernel Hacking” and liked it very much, so when I got the chance to get an advance
copy of his new book for review, "FreeBSD Device Drivers: A Guide for the Intrepid", I couldn’t say
no. :)To make the review more practical, I decided to write a simple driver myself and posted about
it and the book on the FreeBSD forums, to find that post follow the WWW: link in the pkg-descr of
the comms/uartlirc port (you can also look the port up on freshports.org).About the book:The book
introduces you to almost everything you need to know to write many types of drivers, it does this
mainly by doing code walkthroughs for several example- and real-world drivers. It obviously cannot
cover _everything_ (sound drivers for example are not covered, nor is miibus(4)), but what it covers
I’d say should give you enough information to be able to look at manpages and existing drivers for
missing details. 100% recommended!

Most programmers consider device drivers the darkest of the dark computer arts, but to write a good
device driver what’s needed is a decent template and some good documentation. The FreeBSD
Operating Systems has plenty of templates, in the form of already working drivers, and with the
publication of Joseph Kong’s latest book, FreeBSD Device Drivers, now there is good
documentation as well. The book takes the reader from the simplest types of drivers, such as those
used to do serial communication, up through disk, usb and network drivers, which are far more
complex and require the programmer to have a greater understanding about the operating system
in which they’re working. The introductory chapters give enough of the required background
information for writing a driver, covering areas such as memory allocation, and synchronization
primitives, without preventing the reader from, very quickly, getting down to working on real
code. One of the beauties of this book is that it covers running code in a real world operating system,
making it far more relevant for both students and working programmers. Many books on
programming create neat and easy problems that the authors think will take the reader through the
necessary steps to understanding a concept, but this book doesn’t shy away from the nitty gritty
details of low level code. The book has an easy to read, narrative style which makes reading it an
enjoyable experience, a seeming rarity in technical books. I’d recommend this book to anyone who
wants to truly understand what goes on, under the hood, in an operating system.

Just before this book was released, I recently started a FreeBSD device driver project. I started out
by searching the web for all related documentation and tutorials. I found myself trying to piece
together dozens of different sources, most of which were too simplistic, narrowly focused, or
outdated. Save yourself the time and frustration; start by buying this book. It is a well organized
guide for learning to write device drivers and will get you productive much faster than scouring the
web.

This is the type of book every Unix developer or systems administrator should own. Don't let the title
fool you, while this is definitely a FreeBSD-heavy book, the methodology and examples would be
useful to any aspiring kernel developer or systems administrator. Using real world examples, with
informative, in depth explanations on what is happening "under the hood," Joseph Kong's book will
walk the reader through every aspect of writing a drivers for a variety of hardware. As a systems
administrator of FreeBSD, Linux, and Solaris systems, I have found this book to be an asset and am
really happy I put it on my technical references shelf and feel confident any system administrator or
kernel developer will feel the same way.

This is a really helpful book, I have read and searched forth and back the Freebsd documentation
for clues for starting driver development, well you could read the source code of the drivers as well,
but is it better to read this book first as it will make things clear or you will be more familiar with the
current infrastructure, this book is really is a must if you want to start digging on FBSD driver coding,
I really recommend it.

Download to continue reading...

FreeBSD Linux Device Drivers Developing Windows NT Device Drivers: A Programmer’s Handbook
Essential Linux Device Drivers (Prentice Hall Open Source Software Development Series) Writing
Windows VxDs and Device Drivers Pro Windows Embedded Compact 7: Producing Device Drivers
(Expert’s Voice in Windows) Chromecast: Chromecast Easy Guide: Master Your Chromecast
Device and Enjoy TV Entertainment With Low-Cost Media Streamer (Chromecast, Chromecast
User Guide, Chromecast books, Chromecast Device) How to Add A Device To My Account: How to
Add a Device The Intrepid Art Collector: The Beginner's Guide to Finding, Buying, and Appreciating
Art on a Budget The Intrepid Woman's Guide to Van Dwelling: Practical Information to Customize a
Chic Home on Wheels & Successfully Transition to an Awesome Mobile Lifestyle Big Data
Fundamentals: Concepts, Drivers & Techniques (The Prentice Hall Service Technology Series from
Thomas Erl) Windows 10 Troubleshooting: Windows 10 Manuals, Display Problems, Sound
Problems, Drivers and Software: Windows 10 Troubleshooting: How to Fix Common Problems ...
Tips and Tricks, Optimize Windows 10) Designing Cards and Drivers for the Macintosh Family
Plague: One Scientist’s Intrepid Search for the Truth about Human Retroviruses and Chronic Fatigue Syndrome (ME/CFS), Autism, and Other Diseases

Lesbian Pulp Fiction: The Sexually Intrepid World of Lesbian Paperback Novels 1950-1965

Highway 1, Hell & Horror - Vietnam War Convoys - Ambushes: Truck Drivers - Chasing Ghosts - Outsystem: A Military Science Fiction Space Opera Epic: Aeon 14 (The Intrepid Saga)