

# HARDWARE HACKING

Have Fun While  
Voiding Your Warranty

SYNGRESS®

# HARDWARE HACKING

Have Fun While  
Voiding Your Warranty

**Joe Grand** Author of *Stealing the Network*

**Ryan Russell** Author of *Stealing the Network and  
Hack Proofing Your Network, Second Edition*

And featuring **Kevin D. Mitnick** Technical Reviewer

Foreword by **Andrew “bunnie” Huang**

**Lee Barken** **Marcus R. Brown** **Job de Haas** **Deborah Kaplan**  
**Bobby Kinstle** **Tom Owad** **Albert Yarusso**

Syngress Publishing, Inc., the author(s), and any person or firm involved in the writing, editing, or production (collectively “Makers”) of this book (“the Work”) do not guarantee or warrant the results to be obtained from the Work.

There is no guarantee of any kind, expressed or implied, regarding the Work or its contents. The Work is sold AS IS and WITHOUT WARRANTY. You may have other legal rights, which vary from state to state.

In no event will Makers be liable to you for damages, including any loss of profits, lost savings, or other incidental or consequential damages arising out from the Work or its contents. Because some states do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

You should always use reasonable care, including backup and other appropriate precautions, when working with computers, networks, data, and files.

Syngress Media®, Syngress®, “Career Advancement Through Skill Enhancement®,” “Ask the Author UPDATE®,” and “Hack Proofing®,” are registered trademarks of Syngress Publishing, Inc. “Syngress: The Definition of a Serious Security Library”™, “Mission Critical™,” and “The Only Way to Stop a Hacker is to Think Like One™” are trademarks of Syngress Publishing, Inc. Brands and product names mentioned in this book are trademarks or service marks of their respective companies.

**KEY SERIAL NUMBER**

001	HJIRTCV764
002	PO9873D5FG
003	829KM8NJH2
004	B7NMW3V9KM
005	CVPLQ6WQ23
006	VBP965T5T5
007	HJJJ863WD3
008	2987GVTWMK
009	629MP5SDJT
010	IMWQ295T6T

**PUBLISHED BY**  
Syngress Publishing, Inc.  
800 Hingham Street  
Rockland, MA 02370

**Hardware Hacking: Have Fun While Voiding Your Warranty**

Copyright © 2004 by Syngress Publishing, Inc. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be reproduced for publication.

Printed in the United States of America

1 2 3 4 5 6 7 8 9 0

ISBN: 1-932266-83-6

Technical Editor: Joe Grand  
Technical Reviewer: Kevin D. Mitnick  
Acquisitions Editor: Catherine B. Nolan  
Page Layout and Art: Patricia Lupien

Cover Designer: Michael Kavish  
Copy Editor: Darlene Bordwell  
Indexer: J. Edmund Rush  
Editorial Assistant: Michael Rubin

Distributed by O’Reilly & Associates in the United States and Jaguar Book Group in Canada.



# Acknowledgments

We would like to acknowledge the following people for their kindness and support in making this book possible.

To Jeff Moss and Ping Look of Black Hat for being great friends and supporters of Syngress.

A special thanks to Kevin Mitnick for sharing his invaluable expertise and knowledge, and to Darci Wood for her support of this book and the Syngress publishing program.

Syngress books are now distributed in the United States by O'Reilly & Associates, Inc. The enthusiasm and work ethic at ORA is incredible and we would like to thank everyone there for their time and effort in bringing Syngress books to market: Tim O'Reilly, Laura Baldwin, Mark Brokering, Mike Leonard, Donna Selenko, Bonnie Sheehan, Cindy Davis, Grant Kikkert, Opol Matsutaro, Lynn Schwartz, Steve Hazelwood, Mark Wilson, Rick Brown, Leslie Becker, Jill Lothrop, Tim Hinton, Kyle Hart, Sara Winge, C. J. Rayhill, Peter Pardo, Leslie Crandell, Valerie Dow, Regina Aggio, Pascal Honscher, Preston Paull, Susan Thompson, Bruce Stewart, Laura Schmier, Sue Willing, and Mark Jacobsen.

The incredibly hard working team at Elsevier Science, including Jonathan Bunkell, Duncan Enright, David Burton, Rosanna Ramacciotti, Robert Fairbrother, Miguel Sanchez, Klaus Beran, and Rosie Moss for making certain that our vision remains worldwide in scope.

David Buckland, Wendi Wong, Daniel Loh, Marie Chieng, Lucy Chong, Leslie Lim, Audrey Gan, and Joseph Chan of STP Distributors for the enthusiasm with which they receive our books.

Kwon Sung June at Acorn Publishing for his support.

Jackie Gross, Gayle Voycey, Alexia Penny, Anik Robitaille, Craig Siddall, Darlene Morrow, Iolanda Miller, Jane Mackay, and Marie Skelly at Jackie Gross & Associates for all their help and enthusiasm representing our product in Canada.

Lois Fraser, Connie McMenemy, Shannon Russell, and the rest of the great folks at Jaguar Book Group for their help with distribution of Syngress books in Canada.

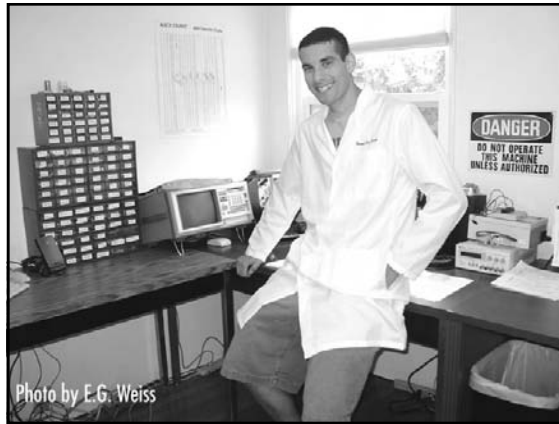
David Scott, Tricia Wilden, Marilla Burgess, Annette Scott, Geoff Ebbs, Hedley Partis, Bec Lowe, and Mark Langley of Woodslane for distributing our books throughout Australia, New Zealand, Papua New Guinea, Fiji Tonga, Solomon Islands, and the Cook Islands.

Winston Lim of Global Publishing for his help and support with distribution of Syngress books in the Philippines.

To all the folks at Malloy who have made things easy for us and especially to Beth Drake and Joe Upton.



# Technical Editor & Contributor



**Joe Grand; Grand Idea Studio, Inc.** Joe Grand is the President and CEO of Grand Idea Studio, a product design and development firm that brings unique inventions to market through intellectual property licensing. Many of his creations, including consumer electronics, medical products, video games and toys, are sold worldwide.

A recognized name in computer security and electrical engineering, Joe's pioneering research on product design and analysis, mobile devices, and digital forensics is published in various industry journals. He is a co-author of *Hack Proofing Your Network, Second Edition* (Syngress Publishing, ISBN 1-928994-70-9) and *Stealing The Network: How to Own the Box* (Syngress, ISBN 1-931836-87-6).

Joe has testified before the United States Senate Governmental Affairs Committee on the state of government and homeland computer security, and is a former member of the legendary hacker think-tank, L0pht Heavy Industries. He has presented his work at numerous academic, industry, and private forums, including the United States Naval Post Graduate School Center for INFOSEC Studies and Research, the United States Air Force Office of Special Investigations, the USENIX Security Symposium, and the IBM Thomas J. Watson Research Center. Joe holds a BSCE from Boston University.

*Joe is the author of Chapter 1 "Tools of the Warranty Voiding Trade," Chapter 2 "Electric Engineering Basics," Chapter 3 "Declawing Your CueCat," and Chapter 13 "Upgrading Memory on Palm Devices."*



# Contributors

**Lee Barken** (CISSP, CCNA, MCP, CPA) is the co-director of the Strategic Technologies and Research (STAR) Center at San Diego State University. He has worked as an IT consultant and network security specialist for Ernst & Young's Information Technology Risk Management (ITRM) practice and KPMG's Risk and Advisory Services (RAS) practice. Lee is the co-founder of the San Diego Wireless Users Group and writes and speaks on the topic of wireless LAN technology and security. He is the technical editor for *Mobile Business Advisor Magazine*, and the author of *How Secure Is Your Wireless Network? Safeguarding Your Wi-Fi LAN* (ISBN: 0-13-140206-4).

*"Let's be grateful for those who give us happiness; they are the charming gardeners who make our soul bloom." —Marcel Proust*

With deepest appreciation for my charming gardeners, a special thank you to my love Stephanie, my mom and dad, Frieda and Israel, my brothers, Derren and Martin, my sister Randi and her husband Scott, my Uncle Harry and my Grandmother Sophie. Thank you for your support and love.

*Lee is the author of Chapter 10 "Wireless 802.11 Hacks."*

**Marcus R. Brown** is a software engineer at Budcat Creations. His work includes writing low-level drivers and system-level programming such as resource management, file loading, and audio streaming. He is currently working on an unannounced title for the PlayStation 2 and Xbox. Marcus lives in Las Vegas, Nevada.

*Marcus is the author of Chapter 9 "Hacking the PlayStation 2."*

**Job de Haas** is Managing Director of ITSX BV, a Dutch company located in Amsterdam. ITSX BV provides security testing services in the broadest sense. Job is involved in testing, researching, and breaking security aspects of the latest technologies for corporate clients. In assignments for telecommunication operators and mobile phone manufacturers, Job gained experience with the internal operations of modern phones.

Job holds a master's degree in electrical engineering from Delft Technical University. He previously held positions at the Dutch Aerospace Agency (NLR) as a robotics researcher and at Digicash BV as a developer of cryptographic applications. He lives in Amsterdam, The Netherlands.

*Job is the author of Chapter 12 “Can You Hear Me Now? Nokia 6210 Mobile Phone Modifications.”*

**Deborah Kaplan** (PCP) is an independent consultant focusing on revision control systems, system administration tools, release engineering, and open-source software. Deborah has developed enterprise-wide technology infrastructure, integrating telecommunications with heterogeneous Windows and UNIX environments. She specializes in building tools that automate repetitive tasks and monitor systems for performance tuning.

Deborah holds a bachelor's degree from Haverford College and a master's degree from Simmons.

*Deborah is the author of Chapter 14 “Operating Systems Overview” and Chapter 15 “Coding 101.”*

**Bobby Kinstle** works in the Reliability Engineering department at Apple Computer, Inc. where he performs destructive simulations of extreme use and abuse of the products. His specialties are performing voltage and frequency margin analysis as well as detailed thermal performance studies. He also performs environmental testing, mechanical shock and vibration, and repetitive stress testing. Bobby also designed and built the lab's test network of over 600-switched Ethernet ports with 4-gigabit fiber optic backbones and NetBoot servers as well as the department data center. When projects are slow Bobby teaches Mac OS X Server training classes within the company.

*Bobby is the author of Chapter 4 “Terabyte FireWire Hard Drive Case Mod” and a co-author of Chapter 5 “Macintosh Hacks.”*

**Tom Owad** is the owner and Web master of Applefritter, [www.applefritter.com](http://www.applefritter.com), a community where the artist and the engineer meet. Applefritter provides its members with discussion boards for the exchange of ideas and hosts countless member-contributed hardware hacks and other projects. Tom is pursuing a Bachelor’s Degree in Computer Science and International Affairs from Lafayette College, Pennsylvania.

*Tom is a co-author of Chapter 5 “Macintosh Hacks.”*

**Ryan Russell** has worked in the IT field for over 13 years, focusing on information security for the last seven. He was the primary author of *Hack Proofing Your Network, Second Edition* (Syngress Publishing, ISBN 1-928994-70-9) and *Stealing the Network: How to Own the Box*, Syngress Publishing (ISBN: 1-931836-87-6, and is a frequent technical editor for the Hack Proofing series of books. He is also a technical advisor to Syngress Publishing’s *Snort 2.0 Intrusion Detection* (ISBN: 1-931836-74-4). Ryan founded the vuln-dev mailing list, and moderated it for three years under the alias “Blue Boar.” He is a frequent lecturer at security conferences, and can often be found participating in security mailing lists and website discussions. Ryan is the Director of Software Engineering for AnchorIS.com, where he’s developing the anti-worm product, Enforcer. One of Ryan’s favorite activities is disassembling worms.

*Ryan is the author of Chapter 6 “Home Theater PCs.”*

**Albert Yarusso** is a principle of Austin Systems ([www.austinsystems.com](http://www.austinsystems.com)), an Austin, Texas-based firm that specializes in web design programming and hosting services. Albert's background consists of a wide range of projects as a software developer, with his most recent experience focused in the game industry. Albert previously worked for Looking Glass Technologies and more recently for Ion Storm Austin, where he helped create the highly acclaimed PC game "Deus Ex."

Albert co-founded AtariAge ([www.atariage.com](http://www.atariage.com)) in 2001, a comprehensive web-site devoted to preserving the history of Atari's rich legacy of video game consoles and computers, which has become one of the busiest destinations on the web for classic gaming fans. In 2003, Albert helped bring the first annual Austin Gaming Expo ([www.austingamingexpo.com](http://www.austingamingexpo.com)) to Austin, an extremely successful event that drew over 2,000 visitors in its first year.

*Albert is the author of Chapter 7 "Hack Your Atari 2600 and 7800," Chapter 8 "Hack Your Atari 5200 and 8-Bit Computer," and Chapter 11 "Hacking the iPod."*



## Foreword Contributor

**Andrew “bunnie” Huang** (PhD) is a staff engineer with Luxtera, and a part-time research staff with the California Institute of Technology. He also heads up a private consultancy firm, Xenatera LLC. bunnie is the author of *Hacking the Xbox*. bunnie has a broad background in electronics and firmware that comes in handy for various hardware hacking and reverse engineering projects. bunnie holds a PhD, M.Eng, and SB from the Massachusetts Institute of Technology, and is a member of the IEEE. He lives in San Diego, CA, with his fiancée, Nicole Justis.



## Technical Reviewer

**Kevin D. Mitnick** is a security consultant to corporations worldwide and a cofounder of Defensive Thinking, a Las Vegas-based consulting firm ([www.defensivethinking.com](http://www.defensivethinking.com)). He has testified before the Senate Committee on Governmental Affairs on the need for legislation to ensure the security of the government’s information systems. His articles have appeared in major new magazines and trade journals, and he has appeared on Court TV, *Good Morning America*, *60 Minutes*, CNN’s *Burden of Proof* and *Headline News*, and has been a keynote speaker at numerous industry events. He has also hosted a weekly radio show on KFI AM 640, Los Angeles. Kevin is also author of the best-selling book, *The Art of Deception: Controlling the Human Element of Security*.

# Contents

<b>Foreword</b>	<b>xxvii</b>
<b>Introduction</b>	<b>xxxv</b>
<b>Part I Introduction to Hardware Hacking</b>	<b>1</b>
<b>Chapter 1 Tools of the Warranty Voiding Trade</b>	<b>3</b>
Introduction	4
The Essential Tools	4
Taking it to the Next Level	6
Hardcore Hardware Hackers Only	8
Where to Obtain the Tools	10
<b>Chapter 2 Electrical Engineering Basics</b>	<b>13</b>
Introduction	14
Fundamentals	14
Bits, Bytes, and Nibbles	14
Reading Schematics	18
Voltage, Current, and Resistance	20
Direct Current and Alternating Current	21
Resistance	22
Ohm's Law	22
Basic Device Theory	23
Resistors	23
Capacitors	25
Diodes	28
Transistors	30
Integrated Circuits	32
Soldering Techniques	34

Hands-On Example: Soldering a Resistor to a Circuit Board	34
Desoldering Tips	36
Hands-On Example: SMD Removal Using ChipQuik	37
Common Engineering Mistakes	40
Web Links and Other Resources	41
General Electrical Engineering Books	41
Electrical Engineering Web Sites	42
Data Sheets and Component Information	43
Major Electronic Component and Parts Distributors	43
Obsolete and Hard-to-Find Component Distributors	43
<b>Part II Hardware Hacks</b>	<b>45</b>
<b>Chapter 3 Declawing Your CueCat</b>	<b>47</b>
Introduction	48
Model Variations	49
Opening the CueCat	51
Preparing for the Hack	51
Opening the Four-Screw PS/2 CueCat	51
Opening the Two-Screw PS/2 CueCat	54
Opening the USB CueCat	55
Removing the Unique Identifier	56
Preparing for the Hack	57
Removing the UID: Four-Screw PS/2CueCat	57
Removing the UID: Two-Screw PS/2CueCat	60
Removing the UID: USB CueCat	62
Under the Hood: How the Hack Works	64
Removing the Proprietary Barcode Encoding	68
Preparing for the Hack	68
Removing the Encoding from the Four-Screw PS/2 CueCat	69
Removing the Encoding from the Two-Screw PS/2 CueCat	71
Removing the Encoding from the USB CueCat	73
Under the Hood: How the Hack Works	74

Technical Information	76
The CueCat Encoding Scheme	76
More Physical Model Variations	78
More History of Political and Legal Issues	80
CueCat Litter Box: Web Links and Other Resources	82
Open-Source CueCat Software and Drivers	83
DigitalConvergence Patents for CueCat Technologies	83
<b>Chapter 4 Case Modification: Building a Custom Terabyte FireWire Hard Drive</b>	<b>83</b>
Introduction	84
Case Mod Primer	84
Creating a 1.2TB FireWire RAID	85
Preparing for the Hack	85
Performing the Hack	86
Under the Hood: How the Hack Works	92
Custom Case Modification for the FireWire RAID	94
Preparing for the Hack	94
Performing the Hack	95
Under the Hood: How the Hack Works	105
Additional Resources	108
Case Modifications	109
<b>Chapter 5 Macintosh</b>	<b>111</b>
Compubrick SE	112
Preparing for the Hack	113
Performing the Hack	114
Taking Apart the Mac	114
Encasing the Speaker	120
Covering the Mouse and the Keyboard	121
Encasing the Disk Drive	123
Encasing the Hard Drive	125
Encasing the Motherboard	127
Encasing the CRT	129
How the Hack Works	131
Building a UFO Mouse	132

Preparing for the Hack	133
Performing the Hack	134
Opening the Mouse	134
Drilling the Hole	136
Soldering the LED	137
Reassembling the Mouse	138
How the Hack Works	140
Adding Colored Skins to the Power Macintosh G4 Cube	140
Preparing for the Hack	141
Performing the Hack	142
Under the Hood: How the Hack Works	145
Other Hacks and Resources	145
Desktop Hacks	145
Laptop Hacks	146
Electrical and Optical Hacks	146
Case Mods	146
Software	147
Discussion	147
<b>Chapter 6 Home Theater PCs</b>	<b>149</b>
Introduction	150
Before You Begin: Research and Plan	151
How Much Could It Cost?	152
Did Someone Already Build It?	153
The Components of an HTPC Project	154
The Display	155
What Are Your Options for Higher-Quality Video Display?	157
The Video Card	160
The Case	160
The Hard Drives	161
Speed Considerations	163
Sshhhh... Quiet Operations	164
Optical Drives	164
The CPU	165
The Sound Card	166

The Controller	167
The Software	167
Building a Windows HTPC	171
Preparing for the Hack	171
Performing the Hack: Software	175
Eazylook	177
Using the Launcher	178
Using Guide Plus+	178
CDex	180
FairUse	180
Windows Summary	185
Building a Linux HTPC	185
Preparing for the Hack	185
Performing the Hack: Hardware	185
Performing the Hack: Software	192
Installing the Video Capture Drivers	192
Install MPlayer and CODECs	194
Installing MythTV	194
Linux Summary	197
Further Hacking and Advanced Topics	198
<b>Chapter 7 Hack Your Atari 2600 and 7800</b>	<b>199</b>
Introduction	200
The Atari 7800 ProSystem	201
Hacks in This Chapter	202
Atari 2600 Left-Handed Joystick Modification	202
Preparing for the Hack	203
Performing the Hack	204
Use an NES Control Pad with Your 2600	207
Preparing for the Hack	207
Performing the Hack	209
Atari 2600 Stereo Audio Output	214
Preparing for the Hack	216
Performing the Hack	216
Under the Hood: How the Hack Works	223
Atari 7800 Blue LED Modification	223

Preparing for the Hack	223
Performing the Hack	224
Under the Hood: How the Hack Works	227
Atari 7800 Game Compatibility Hack to Play Certain	
2600 Games	228
Preparing for the Hack	229
Performing the Hack	230
Under the Hood: How the Hack Works	232
Atari 7800 Voltage Regulator Replacement	232
Preparing for the Hack	233
Performing the Hack	233
Under the Hood: How the Hack Works	236
Atari 7800 Power Supply Plug Retrofit	237
Preparing for the Hack	238
Performing the Hack	239
Other Hacks	242
2600 Composite/S-Video Modifications	242
Atari 7800 Composite and S-Video Output	243
Sega Genesis to Atari 7800 Controller Modification	243
NES Control Pad to Atari 7800 Controller Modification	243
Atari 7800 DevOS Modification and Cable Creation	243
Atari Resources on the Web	244
<b>Chapter 8 Hack Your Atari 5200 and 8-Bit Computer</b>	<b>247</b>
Introduction	248
The Atari 5200 SuperSystem	249
Hacks in This Chapter	250
Atari 5200 Blue LED Modification	250
Preparing for the Hack	251
Performing the Hack	251
Under the Hood: How the Hack Works	256
Creating an Atari 5200 Paddle	256
Preparing for the Hack	257
Performing the Hack: Disassembling the Paddle	
Controller	258

Performing the Hack: Building the 5200 Paddle Controller	260
Performing the (Optional) Hack: Weighted Dial	266
Under the Hood: How the Hack Works	267
Free Yourself from the 5200 Four-Port Switchbox	268
Preparing for the Hack	269
Performing the Hack	271
Under the Hood: How the Hack Works	279
Build Atari 8-Bit S-Video and Composite Cables	280
Preparing for the Hack	281
Performing the Hack	282
Cable Hack Alternatives	288
Under the Hood: How the Hack Works	289
Technical Information	289
Other Hacks	290
Atari 5200 Four-Port VCS Cartridge Adapter Fix	290
Atari 5200 Composite/S-Video Modification	290
Atari 8-Bit SIO2PC Cable	291
Atari Resources on the Web	291
<b>Chapter 9 Hacking the PlayStation 2</b>	<b>293</b>
Introduction	294
Commercial Hardware Hacking: Modchips	294
Getting Inside the PS2	296
Mainboard Revisions	296
Identifying Your Mainboard	297
Opening the PS2	298
Installing a Serial Port	302
Preparing for the Hack	303
Performing the Hack	304
Testing	309
Under the Hood: How the Hack Works	310
Bootting Code from the Memory Card	310
Preparing for the Hack	310
Performing the Hack: Preparing Title.DB	311
Choosing BOOT.ELF	313

Saving TITLE.DB to the Memory Card	314
Independence!	314
Under the Hood: How the Hack Works	314
Other Hacks: Independent Hard Drives	316
PS2 System Overview	316
Understanding the Emotion Engine	317
The Serial I/O Port	318
The I/O Processor	321
The Sub-CPU Interface	321
Additional Web Resources	321
<b>Chapter 10 Wireless 802.11 Hacks</b>	<b>323</b>
Introduction	324
Wireless NIC/PCMCIA Card Modifications:	
Adding an External Antenna Connector	325
Preparing for the Hack	326
Performing the Hack	327
Removing the Cover	327
Moving the Capacitor	329
Attaching the New Connector	331
Under the Hood: How the Hack Works	332
OpenAP (Instant802): Reprogramming Your Access Point	
with Linux	332
Preparing for the Hack	333
Performing the Hack	334
Installing the SRAM Card	335
Power Me Up, Scotty!	338
Under the Hood: How the Hack Works	338
Having Fun with the Dell 1184 Access Point	338
Preparing for the Hack	339
Performing the Hack	340
Under the Hood: How the Hack Works	345
Summary	345
Additional Resources and Other Hacks	345
User Groups	345
Research and Articles	346

Products and Tools	346
<b>Chapter 11 Hacking the iPod</b>	<b>349</b>
Introduction	350
Opening Your iPod	353
Preparing for the Hack	354
First Generation iPods	355
Second and Third-Generation iPods	356
Replacing the iPod Battery	359
Preparing for the Hack	360
Battery Replacement: First- and Second-Generation iPods	361
Battery Replacement: Third-Generation iPods	365
Upgrading a 5GB iPod's Hard Drive	371
Preparing for the Hack	372
Performing the Hack	372
From Mac to Windows and Back Again	381
Preparing for the Hack	381
Going from Windows to Macintosh	381
Going from Macintosh to Windows	383
iPod Diagnostic Mode	384
The Diagnostic Menu	384
Disk Check	387
Additional iPod Hacks	388
Installing Linux on an iPod	388
Repairing the FireWire Port	388
Scroll Wheel Fix	389
iPod Resources on the Web	390
<b>Chapter 12 Can You Hear Me Now? Nokia 6210     Mobile Phone Modifications</b>	<b>391</b>
Introduction	392
Nokia 6210 LED Modification	393
Preparing for the Hack	393
Performing the Hack	395
Opening the Nokia 6210	395
Removing the Old LEDs	400

Inserting the New LEDs	401
Increasing the LED Power	402
Putting the Phone Back Together	403
Under the Hood: How the Hack Works	404
Data Cabling Hacks	406
Data Cables	407
Flashing Cables	410
Net Monitor	411
Other Hacks and Resources	415
<b>Chapter 13 Upgrading Memory on Palm Devices</b>	<b>417</b>
Introduction	418
Model Variations	419
Hacking the Pilot 1000 and Pilot 5000	420
Preparing for the Hack	420
Removing the Memory Card	422
Adding New Memory	423
Under the Hood: How the Hack Works	427
Hacking the PalmPilot Professional and PalmPilot Personal	429
Preparing for the Hack	429
Removing the Memory Card	429
Adding New Memory	430
Under the Hood: How the Hack Works	433
Hacking the Palm m505	436
Preparing for the Hack	436
Opening the Palm	437
Removing the Main Circuit Board	439
Removing the Memory	441
Adding New Memory	442
Under the Hood: How the Hack Works	445
Technical Information	447
Hardware	447
File System	448
Memory Map	448

Database Structure	449
Palm Links on the Web	450
Technical Information	450
Palm Hacks	450
More Memory Upgrades	450
<b>Part III Hardware Hacking Technical Reference</b>	<b>451</b>
<b>Chapter 14 Operating Systems Overview</b>	<b>453</b>
Introduction	454
OS Basics	454
Memory	455
Physical Memory	455
Virtual Memory	457
File Systems	458
Cache	459
Input/Output	460
Processes	460
System Calls	461
Shells, User Interfaces, and GUIs	461
Device Drivers	462
Block and Character Devices	464
Properties of Embedded Operating Systems	466
Linux	467
Open Source	467
History	468
Embedded Linux (uClinux)	469
Product Examples: Linux on Embedded Systems	470
VxWorks	470
Product Examples: VxWorks on Embedded Systems	470
Windows CE	471
Concepts	471
Product Examples: Windows CE on Embedded Systems	472
Summary	473
Additional References and Further Reading	473

<b>Chapter 15 Coding 101</b>	<b>475</b>
Introduction	476
Programming Concepts	476
Assignment	477
Control Structures	478
Looping	479
Conditional Branching	480
Unconditional Branching	481
Storage Structures	482
Structures	483
Arrays	484
Hash Tables	485
Linked Lists	486
Readability	488
Comments	488
Function and Variable Names	488
Code Readability: Pretty Printing	489
Introduction to C	490
History and Basics of C	490
Printing to the Screen	490
Data Types in C	493
Mathematical Functions	493
Control Structures	496
<i>For</i> Loops	496
<i>While</i> Loops	496
<i>If/Else</i>	498
<i>Switch</i>	500
Storage Structures	501
Arrays, Pointers, and Character Strings	501
Structures	506
Function Calls and Variable Passing	507
System Calls and Hardware Access	508
Summary	509
Debugging	509
Debugging Tools	509

The <i>printf</i> Method	510
Introduction to Assembly Language	512
Components of an Assembly Language Statement	513
Labels	513
Operations	515
Operands	515
Sample Program	516
Summary	518
Additional Reading	518
<b>Index</b>	<b>519</b>



# Foreword

Hacking—and in particular, hardware hacking—has experienced a bit of a renaissance recently. I am personally quite pleased about the increased interest in hacking. Your interest in this book, *Hardware Hacking: Have Fun While Voiding Your Warranty*, is a testament to the increased demand for knowledge about hardware hacking. I'd like to take a few pages and a few minutes of your time to share with you why your interest in the topic makes me happy as a fellow hardware hacker.

First allow me to pontificate on the meaning of the word *hack*. The term has evolved quite dramatically over the years. Hacking has shaped technology perhaps as much as technology has shaped our perception of the hacker. According to *The New Hacker's Dictionary* (a public-domain lexicon of jargon created by hackers, [www.jargon.8hz.com](http://www.jargon.8hz.com)):

hack: 1. /n./ Originally, a quick job that produces what is needed, but not well. 2. /n./ An incredibly good, and perhaps very time-consuming, piece of work that produces exactly what is needed. <sup>1</sup>

The second sense of the word is perhaps the closest to the definition I associate with the word *hack*. Thus, it follows that a hacker is one who labors to create good, typically innovative solutions to targeted problems. This book you are about to read was edited by a true hacker, Joe Grand, and it speaks mostly to the class of hacks that address the need to adapt and improve on existing consumer solutions.

As you can see, my view of hacking is a rather romantic and idealized one. I eschew the Hollywood stereotype of a hacker as a slovenly, socially maladept person with a bent for vengeance, data theft, or per-

haps a penchant to blithely play a game of deploy-the-nuke inside NORAD's computers. Although there are certainly such elements in today's hacker culture, I prefer to focus on promoting the more socially redeeming aspects of hacking. I believe that hacking is rooted in a desire to play with and understand technology, a modern manifestation of the values of exploration, passion, and hard work that date back to the first explorers and settlers of this country. Furthermore, hacking is a kind of grass-roots technology movement, in contrast to the kinds of technology movements that are forwarded by corporations and governments. As a result, hackers tend to play the part of proxy for the masses when it comes to sorting out the interplay of technology, society, and business. As technology continues to infuse our daily lives, it is becoming more important for society to bring its representatives to the technology direction table.

It is interesting and perhaps informative to see how hardware hacking has evolved over the years. In the early days of electronics, common hobbyists—hackers of sorts, but the term wasn't coined back then—could cobble together unique, useful, and sometimes outright impressive pieces of hardware that could match commercially available products in both performance and quality. In fact, some of the projects that hackers labored over in their garages went on to form the roots of today's technology.

Roll the calendar back to 1938: A young Bill Hewlett and Dave Packard get together and invent, in their garage, a high-quality piece of audio test equipment, the HP200A resistance-capacitance audio oscillator. Hewlett and Packard continued on to found the company we know today, and its rich history of engineer-friendly products helped forge the technology base we now enjoy. Most people are familiar with HP as a manufacturer of computers and printers, but HP's richest contributions to technology have been through enabling technologies, such as the tools engineers require to do their jobs. I myself use an HP48GX calculator, and I have an HP1650B logic analyzer on my desk, on top of my old HP8410C network analyzer.

Another well-recognized example of a company and technology with roots in the hacker community is Apple Computer. Roll back to 1976: Steve Wozniak debuts the Apple I at the Homebrew Computer Club in Palo Alto, California. The Apple I was designed over a period of years as a hobby machine, a true product of the hacking culture. Wozniak joined

forces with Steve Jobs, and the two went on to found the Apple Computer that brought us the Apple II and the now ubiquitous Macintosh computer.

The gritty grass-roots hacking culture in the early days of electronics technology served as a kind of incubator for innovation that has resulted in many of the products we enjoy today. Hewlett and Packard, Jobs and Wozniak are just two examples of the influence of the hacker spirit on our society. The basic values of hacking—creating a good thing that is exactly what is needed at a particular time—are a good match with innovation. Furthermore, hackers' independently motivated nature means that thousands of ideas are tested and built by hackers in the absence of venture capital or the risk constraints of investors. Hackers play an important part in the growth of technology, so I am always pleased to see a greater interest and awareness of hacking in the general public.

Recently, hacking has taken on more of a software-oriented bent. This is due in part to the steady pace of hardware improvement guaranteed by Moore's Law. Hardware hacking is a time-consuming labor of love, and it is discouraging to know that almost any hack you can think of to double a computer's performance will be obsolete within 12 months. It is much more rewarding to work in the instant-gratification world of software and let the performance of your programs ride the Moore's Law wave.

Another factor working against hardware hackers is the barrier of entry that was created by the higher levels of integration that naturally followed as a result of Moore's Law. The hackability of the desktop PC met a turning point in the evolution of the IBM PC-XT to the IBM PC-AT. The IBM PC-XT motherboard was chiefly composed of chips that were essentially naked logic gates. This was very hacker-friendly, since most of the core functionality was exposed at a human-friendly scale. The IBM PC-AT, on the other hand, was one of the first desktop computers to use VLSI chips for the processor support logic. I remember my first look at the PC-AT motherboard: I was hoping to be able to read the board like a book, with all the logic gates' part numbers gleaming in their fresh white silkscreen against the matte epoxy bodies of chips. What I saw instead was a closed book; there were perhaps three or four curious, high pin-count chips with part numbers and a manufacturer's logo I had never seen before. These chips were proprietary, and any hope of a deeper level of understanding or hardware exploration seemed to be dashed.

I think perhaps a lot of prospective hardware hackers felt the same way around then, because since then hacking has taken on a distinct software-oriented slant. Some of the most famous hackers today are renowned for their software contributions. Richard Stallman and Linus Torvalds are perhaps household names among the technological elite due to their fantastic contributions to free software through GNU and Linux. The best part about software hacking is its very low barrier of entry. Any willing youth with access to a computer and an Internet connection can plug into any of the various free software efforts and make a contribution to the technology collective. All the tools required to generate high-quality code are virtually free, and aside from the time investment, it costs nothing to use them. On the other hand, hardware hacking has a very real entry cost associated with the activity; there is a bare minimum set of tools that are needed on a daily basis, and an unfortunately large and diverse assortment of expensive, specialized tools is required to accomplish specific jobs. Furthermore, producing a hardware hack typically requires real materials in addition to time and energy, thereby placing creative and/or bold (read: risky) hardware-hacking projects beyond the financial horizon of most young folk. Given that human nature is to follow the path of least resistance, it is no surprise that hacking today is primarily a software affair.

In a twist of fate, recent macro-economic and social trends have worked to reverse the trend and bring more people into hardware hacking. The detritus of the dot-com bubble created fertile soil for sprouting hardware hackers. An overall reduction in demand for components, design, and manufacturing services has resulted from the economic slowdown. High-quality, used test equipment is trickling down into the ranks of hackers, either snatched off the shelf of dead companies or snapped up for pennies on the dollar at auction. Scrap components are also finding their way into distribution, driving down component prices. Combined with an overall soft demand situation, individual hackers are able to command the same level of service and component choice as large corporations. Furthermore, fabrication and assembly services have been forced to drive their prices down, to the point where hardware hackers could purchase high-tech, custom-built multilayer boards for under \$50 per board.

Hardware design tool vendors also experienced a corresponding price adjustment due to the economic slowdown. Perhaps the most significant recent technological change for hardware hackers is the introduction of pro-