

SECOND EDITION

---

# Building Embedded Linux Systems

*Karim Yaghmour, Jon Masters, Gilad Ben-Yossef, and  
Philippe Gerum*

**O'REILLY®**

Beijing • Cambridge • Farnham • Köln • Sebastopol • Taipei • Tokyo

---

# Table of Contents

<b>Preface .....</b>	<b>ix</b>
<b>1. Introduction .....</b>	<b>1</b>
Definitions	2
Real Life and Embedded Linux Systems	5
Design and Implementation Methodology	27
<b>2. Basic Concepts .....</b>	<b>33</b>
Types of Hosts	33
Types of Host/Target Development Setups	39
Types of Host/Target Debug Setups	41
Generic Architecture of an Embedded Linux System	43
System Startup	47
Types of Boot Configurations	48
System Memory Layout	51
<b>3. Hardware Support .....</b>	<b>55</b>
Processor Architectures	56
Buses and Interfaces	64
	72
Storage	79
General-Purpose Networking	81
Industrial-Grade Networking	83
System Monitoring	85
<b>4. Development Tools .....</b>	<b>87</b>
A Practical Project Workspace	89
GNU Cross-Platform Development Toolchain	91
C Library Alternatives	115
Java	129
Perl	131
Python	134

---

Other Programming Languages	135
Eclipse: An Integrated Development Environment	135
Terminal Emulators	147
<b>5. Kernel Considerations .....</b>	<b>155</b>
Selecting a Kernel	156
Configuring the Kernel	161
Compiling the Kernel	165
Installing the Kernel	167
In the Field	169
<b>6. Root Filesystem Content .....</b>	<b>173</b>
Basic Root Filesystem Structure	173
Libraries	177
Kernel Modules	183
Kernel Images	183
Device Files	184
Main System Applications	193
Custom Applications	201
System Initialization	201
<b>7. Storage Device Manipulation .....</b>	<b>209</b>
MTD-Supported Devices	209
Disk Devices	231
To Swap or Not To Swap	234
<b>8. Root Filesystem Setup .....</b>	<b>235</b>
Filesystem Types for Embedded Devices	235
Writing a Filesystem Image to Flash Using an NFS-Mounted Root Filesystem	254
Placing a Disk Filesystem on a RAM Disk	254
Rootfs and Iniramfs	255
Choosing a Filesystem's Type and Layout	258
Handling Software Upgrades	261
<b>9. Setting Up the Bootloader .....</b>	<b>273</b>
Embedded Bootloaders	274
Server Setup for Network Boot	278
Using the U-Boot Bootloader	285
<b>10. Setting Up Networking Services .....</b>	<b>301</b>
Network Settings	302

Busybox	303
Dynamic Configuration Through DHCP	303
The Internet Super-Server	305
Remote Administration with SNMP	309
Network Login Through Telnet	312
Secure Communication with SSH	314
Serving Web Content Through HTTP	317
Provisioning	321
<b>11. Debugging Tools .....</b>	<b>325</b>
Eclipse	326
Debugging Applications with gdb	328
Tracing	333
Performance Analysis	336
Memory Debugging	344
A Word on Hardware Tools	348
<b>12. Introduction to Real Time Linux .....</b>	<b>351</b>
What Is Real-Time Processing?	351
Should Your Linux Be Real-Time?	352
Common Real-Time Kernel Requirements	356
Some Typical Users of Real-Time Computing Technology	358
The Linux Paths to Real-Time	360
<b>13. The Xenomai Real Time System .....</b>	<b>365</b>
Porting Traditional RTOS Applications to Linux	366
The Xenomai Architecture	368
How Xenomai Works	375
The Real-Time Driver Model	379
Xenomai, Chameleon by Design	385
<b>14. The RT Patch .....</b>	<b>387</b>
Interrupts As Threads	388
Priority Inheritance	398
Configuring the Kernel with the RT Patch	401
High-Resolution Timers	407
The Latency Tracer	410
Conclusion	417
<b>Index .....</b>	<b>419</b>