

X.systems.press



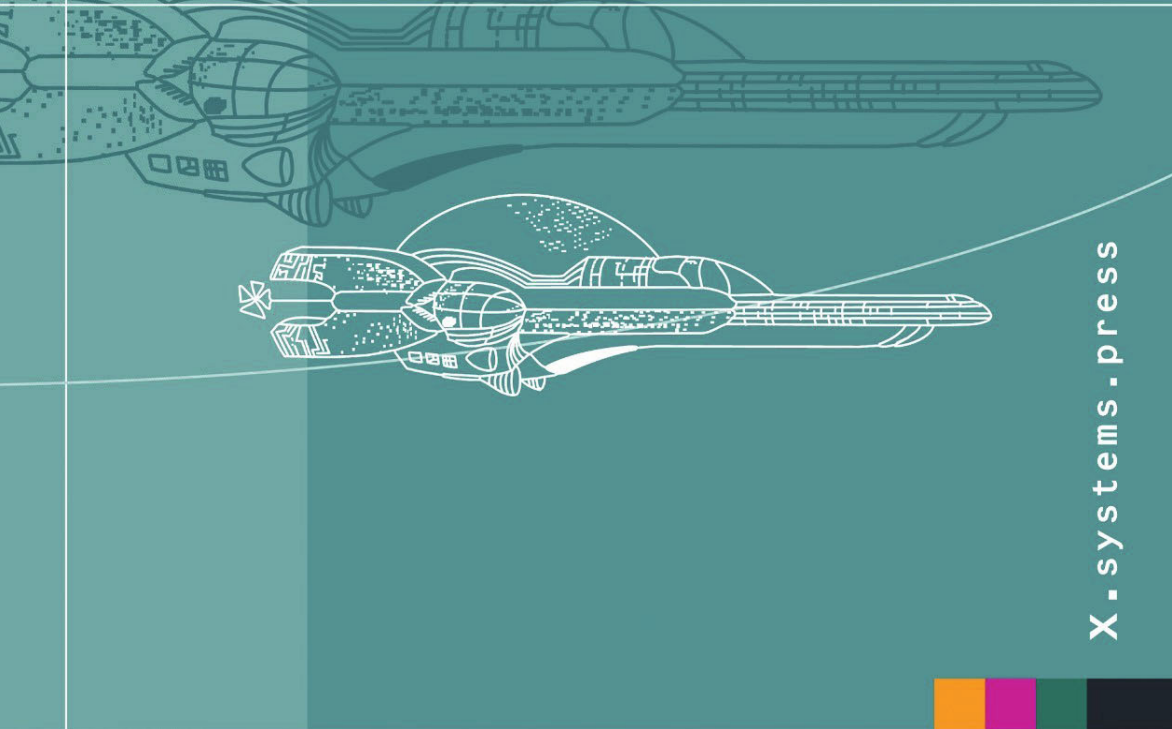
Joachim Schröder  
Tilo Gockel  
Rüdiger Dillmann

# Embedded Linux

Das Praxisbuch

 Springer

0° 10° 20° 30° 40° 50° 60° 70° 80° 90° 100° 110° 120° 130° 140°



X-systems.press

Joachim Schröder  
Tilo Gockel  
Rüdiger Dillmann

# Embedded Linux

Das Praxisbuch

**X . systems . press**

X.systems.press ist eine praxisorientierte  
Reihe zur Entwicklung und Administration von  
Betriebssystemen, Netzwerken und Datenbanken.

Joachim Schröder · Tilo Gockel · Rüdiger Dillmann

# Embedded Linux

Das Praxisbuch

 Springer

Dipl.-Ing. Joachim Schröder  
Universität Karlsruhe  
Informatikfakultät  
Lehrstuhl IAIM Prof. Dr.-Ing. R. Dillmann  
Kaiserstraße 12  
76128 Karlsruhe  
schroede@ira.uka.de

Prof. Dr.-Ing. Rüdiger Dillmann  
Universität Karlsruhe  
Informatikfakultät  
Lehrstuhl IAIM Prof. Dr.-Ing. R. Dillmann  
Kaiserstraße 12  
76128 Karlsruhe  
dillmann@ira.uka.de

Dr.-Ing. Tilo Gockel  
Universität Karlsruhe  
Informatikfakultät  
Lehrstuhl IAIM Prof. Dr.-Ing. R. Dillmann  
Kaiserstraße 12  
76128 Karlsruhe  
gockel@ira.uka.de

ISBN 978-3-540-78619-1

ISBN 978-3-540-78620-7 (eBook)

DOI 10.1007/978-3-540-78620-7

Springer Dordrecht Heidelberg London New York

X.systems.press ISSN 1611-8618

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

© Springer-Verlag Berlin Heidelberg 2009

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funk- sendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwider- handlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk be- rechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

*Einbandentwurf:* KünkelLopka, Heidelberg

Gedruckt auf säurefreiem Papier

Springer ist Teil der Fachverlagsgruppe Springer Science+Business Media ([www.springer.de](http://www.springer.de))

---

## Vorwort

Der Lehrstuhl von Herrn Prof. Dillmann am Institut für Anthropomatik der Universität Karlsruhe (TH) beschäftigt sich mit vielfältigen Themen rund um das Forschungsfeld der Robotik. Zu den Schwerpunkten zählen mobile Service-roboter und intelligente Fahrzeuge, maschinelles Lernen und Robotik in der Medizin, um nur einige davon zu nennen.

Aus den laufenden Forschungsarbeiten am Institut ergaben sich in den letzten Jahren zahlreiche Fragestellungen bei der Umsetzung spezieller Aufgaben auf Linux-Systemen. Die Themen betrafen die Auswahl geeigneter Hardware-Plattformen, die Komponentenanbindung, Netzwerkkommunikation, Software-Entwicklung für verschiedene Zielsysteme, Erstellung grafischer Benutzeroberflächen, Multithreading, Echtzeitfähigkeit und Smart-Camera-Technologie.

Die Antworten auf diese Fragen wurden zunächst in losen Blattsammlungen festgehalten, teilweise in elektronischer Form dokumentiert oder auch nur mündlich weitergegeben. Das vorliegende Buch hat zum Ziel, das Wissen zu strukturieren und zu bündeln. Die Inhalte wurden hierzu systematisch aufgearbeitet, um Grundlagenwissen ergänzt und durch viele Beispielapplikationen praxisgerecht veranschaulicht.

*Embedded Linux* ist mittlerweile das vierte Buch der Praxisbuchreihe, die nach dem Start mit dem Buch *Embedded Robotics* im Jahre 2005 mit dem *Praxisbuch Computer Vision*, und dessen englischsprachigem Pendant *Computer Vision – Principles and Practice*, weitergeführt wurde. Die zu diesen Themen vorliegenden Fachbücher sind oftmals als theoretische Lehrwerke konzipiert, entsprechend trocken zu lesen und wenig hilfreich in der Praxis. Die Praxisbücher sollen die Theorie ähnlich tiefgehend vermitteln, darüber hinaus aber die Inhalte um viele Versuche und Implementierungen ergänzen, die nicht nur im Laborumfeld, sondern auch im produktiven Umfeld standhalten.

Das entstandene Buch richtet sich an Studenten der Informatik und der Ingenieurwissenschaften, an Berufsanfänger, Praktiker und generell an alle Interessierten.

Die vorgestellten Bibliotheken und Applikationen werden bei Erscheinen des Buches online frei verfügbar sein. Der Download kann über den Link auf der Website des Springer-Verlages oder über <http://www.praxisbuch.net> erfolgen.

### *Danksagung*

Die Entstehung des vorliegenden Buches ist auch vielen anderen Menschen zu verdanken, die bei den Implementierungen mitgewirkt, die Ergebnisse korrigiert und viel Geduld mit den Autoren bewiesen haben. Es sind dies: Damira Mambetova, Ulla Scheich, Susanne und Detlef Schröder, Steven Wieland, Tobias Gindele, Manfred Kröhnert, Moritz Hassert, Alexander Bierbaum, Pedram Azad, Alexander Kasper, Peter Steinhaus und Andreas Böttinger. Besonders zu nennen sind an dieser Stelle auch Daniel Jagszent, der sein fundiertes Wissen als Gastautor in Kapitel 12 eingebracht hat und Stephan Riedel, der die Quelltext-Beispiele auf den verschiedenen Plattformen getestet hat.

Weiterhin möchten wir Herrn Prof. Rüdiger Dillmann für die Unterstützung und für das Vertrauen, das er in uns wissenschaftliche Mitarbeiter setzt, danken. Durch die hiermit verbundene Freiheit ist die Buchreihe der Praxisbücher erst möglich geworden. Abschließend danken wir Frau Dorothea Glaunsinger und Herrn Hermann Engesser vom Springer-Verlag für die Motivation, Geduld und die reibungslose und professionelle Zusammenarbeit.

Wir wünschen Ihnen viel Freude mit dem vorliegenden Buch und hoffen, Sie für den interessanten und zukunftssträchtigen Bereich der Embedded Systems unter Linux begeistern zu können.

Karlsruhe,  
den 5. Januar 2009

*Das Autorenteam*

### Hinweis

Die Informationen in diesem Buch werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Die erwähnten Soft- und Hardware-Bezeichnungen können auch dann eingetragene Warenzeichen sein, wenn darauf nicht gesondert hingewiesen wird. Sie sind Eigentum der jeweiligen Warenzeicheninhaber und unterliegen gesetzlichen Bestimmungen. Verwendet werden u. a. folgende geschützte Bezeichnungen: iPhone, Texas Instruments, Code Composer, Vision Components, Sony, KEIL,  $\mu$ Vision2.



---

# Inhaltsverzeichnis

---

## Teil I Grundlagen und Plattformen

---

<b>1</b>	<b>Grundlagen</b> . . . . .	17
1.1	Einführung . . . . .	17
1.2	Architekturen, Plattformen und Geschichtliches . . . . .	18
1.3	Eigenschaften eingebetteter Systeme . . . . .	21
1.3.1	Formfaktor . . . . .	21
1.3.2	Mechanik, Kühlung, Robustheit . . . . .	21
1.3.3	Speichermedien . . . . .	22
1.3.4	Schnittstellen . . . . .	23
1.3.5	Stromversorgung . . . . .	24
1.3.6	Chipsätze . . . . .	24
1.3.7	Watchdog . . . . .	24
1.3.8	Echtzeitfähigkeit . . . . .	25
1.4	Betriebssysteme . . . . .	27
1.4.1	Allgemeine Anforderungen . . . . .	27
1.4.2	Prozess-Scheduling . . . . .	29
1.4.3	Systembeispiele . . . . .	32
1.5	Software-Entwicklung . . . . .	33
1.6	Aufbau und Gebrauch des Buches . . . . .	38
<b>2</b>	<b>Hardware-Plattformen</b> . . . . .	41
2.1	Einführung . . . . .	41
2.2	Network-Attached-Storage NSLU2 . . . . .	42
2.3	WLAN-Router WL-500gP . . . . .	44
2.4	MicroClient Jr. und Sr. . . . .	47
2.5	OpenRISC Alekto . . . . .	51
2.6	Mini-ITX-Mainboard D945GCLF2 mit Dual-Core Atom CPU . . . . .	52
2.7	Pegelanpassung für die RS-232-Schnittstelle . . . . .	55

<b>3</b>	<b>OpenWrt auf dem WLAN-Router WL-500g Premium</b> . . . . .	57
3.1	Einführung . . . . .	57
3.2	Einrichtung des OpenWrt-Build-Systems . . . . .	58
3.2.1	Aufspielen des Flash-Images . . . . .	61
3.2.2	Der erste Einlog-Vorgang . . . . .	62
3.3	Schnelleres Einloggen mit SSH-Keys . . . . .	64
3.4	Software-Entwicklung für OpenWrt . . . . .	65
3.5	Erstellung eigener OpenWrt-Module . . . . .	67
3.6	IO-Warrior-Erweiterung und Kernelmodule unter OpenWrt . . . . .	71
<b>4</b>	<b>Debian auf dem NAS-Gerät NSLU2</b> . . . . .	75
4.1	Einführung . . . . .	75
4.2	Debian-Installation . . . . .	76
4.3	Erste Schritte . . . . .	79
4.4	Software-Entwicklung für die NSLU2 . . . . .	80
4.5	NSLU2 als Druckerserver . . . . .	81
4.6	Weiterführende Hinweise . . . . .	84
4.6.1	Erstellung eines vollständigen NSLU2-Backups . . . . .	84
4.6.2	Einstellung der Taster-Funktion . . . . .	84
4.6.3	Probleme beim Booten . . . . .	84
<b>5</b>	<b>Debian auf dem Embedded-PC OpenRISC-Alekto</b> . . . . .	87
5.1	Einführung . . . . .	87
5.2	Angepasste Debian-Installation . . . . .	88
5.3	Erste Schritte . . . . .	89
5.4	Software-Entwicklung . . . . .	91
5.5	Zugriff auf die Alekto-Hardware . . . . .	93
5.5.1	Anwendung der /proc-Erweiterungen in der Konsole . . . . .	93
5.5.2	Zugriff über <code>ioctl()</code> -Befehle . . . . .	94
5.6	Watchdog-Timer . . . . .	96
5.7	Erstellung eines eigenen Alekto-Kernels . . . . .	97
5.8	Vollständige Debian-Installation . . . . .	98
<b>6</b>	<b>Puppy Linux auf dem Embedded-PC MicroClient Jr./Sr.</b> . . . . .	101
6.1	Einführung . . . . .	101
6.2	Puppy-Installation . . . . .	101
6.3	Paket-Management unter Puppy . . . . .	103
6.4	Software-Entwicklung unter Puppy . . . . .	105

---

## Teil II Anwendungen

---

<b>7</b>	<b>Legacy-Schnittstellen und digitale IOs</b> . . . . .	111
7.1	Einführung . . . . .	111
7.2	RS-232 . . . . .	112

7.2.1	Grundlagen der RS-232-Schnittstelle . . . . .	112
7.2.2	Ansteuerung und Programmierung . . . . .	116
7.2.3	Ansteuerung einer seriellen Relaiskarte . . . . .	121
7.3	Centronics und IEEE 1284 . . . . .	124
7.4	General Purpose Input/Output (GPIO) . . . . .	127
7.5	Schnittstellenerweiterung über IO-Warrior . . . . .	129
7.5.1	IO-Warrior-Bausteine . . . . .	129
7.5.2	Installation der IO-Warrior-Treiber unter Debian . . . . .	130
<b>8</b>	<b>Der Inter-IC-Bus . . . . .</b>	<b>133</b>
8.1	Einführung . . . . .	133
8.2	I <sup>2</sup> C-Datenübertragung . . . . .	136
8.2.1	Konzept . . . . .	136
8.2.2	Steuersignale . . . . .	136
8.2.3	Clock Stretching . . . . .	137
8.2.4	Multi-Master-Betrieb . . . . .	138
8.2.5	Adressierung . . . . .	138
8.2.6	I <sup>2</sup> C-Buserweiterungen . . . . .	141
8.3	I <sup>2</sup> C-Anbindung . . . . .	144
8.3.1	I <sup>2</sup> C-Steckverbindung . . . . .	146
8.3.2	Verwendung des I <sup>2</sup> C-Busses bei NSLU2 und Alekto . . . . .	147
8.3.3	I <sup>2</sup> C-Busanbindung über einen IO-Warrior-Baustein . . . . .	149
8.3.4	Die IO-Warrior-I <sup>2</sup> C-Bibliothek . . . . .	150
8.4	Alternative serielle Bussysteme . . . . .	152
8.4.1	Controller Area Network (CAN) . . . . .	153
8.4.2	Local Interconnect Network (LIN) . . . . .	154
8.4.3	1-Wire-Bus . . . . .	155
8.4.4	Serial Peripheral Interface (SPI) . . . . .	156
8.4.5	Universal Serial Bus (USB) . . . . .	156
<b>9</b>	<b>Inter-IC-Bus-Komponenten . . . . .</b>	<b>161</b>
9.1	Einführung . . . . .	161
9.2	Die I <sup>2</sup> C-Bibliothek . . . . .	163
9.2.1	Die Klasse IICBus . . . . .	163
9.2.2	Die Klasse IICBase . . . . .	165
9.3	Tastatur- und LC-Display-Ansteuerung mit PCF8574 . . . . .	167
9.3.1	Philips 8-Bit-I/O-Erweiterung PCF8574 . . . . .	167
9.3.2	I <sup>2</sup> C-Tastaturmodul . . . . .	169
9.3.3	Die Klasse IICKeyboard . . . . .	169
9.3.4	I <sup>2</sup> C-LC-Display . . . . .	170
9.3.5	LC-Display-Treiberbaustein HD44780 . . . . .	171
9.3.6	Die Klasse IICDisplay . . . . .	173
9.3.7	Die Klasse IICIOExpander . . . . .	176
9.4	Temperaturmessung mit DS1631 . . . . .	177
9.4.1	Dallas DS1631 . . . . .	177

9.4.2	Die Klasse IICTempSensor .....	178
9.5	A/D- und D/A-Wandler .....	179
9.5.1	Philips PCF8591 .....	179
9.5.2	Die Klasse IICADCConverter .....	180
9.6	TMC222-Schrittmotorsteuerung .....	184
9.6.1	Trinamic TMC222 .....	184
9.6.2	Conrad C-Control I <sup>2</sup> C-Bus-Stepper-Driver .....	185
9.6.3	Die Klasse IICStepper .....	187
9.6.4	Programmierung des TMC222-OTP-Speichers .....	189
9.7	Chipkarten-Ansteuerung .....	190
9.7.1	EEPROM-Chipkarte AT24Cxx .....	191
9.7.2	Die Klasse IICChipcard .....	192
9.7.3	AES-Verschlüsselung .....	194
9.7.4	Die Klasse AES .....	197
9.8	I <sup>2</sup> C-Bus-Erweiterung über Multiplexer .....	199
9.8.1	Philips PCA9548 I <sup>2</sup> C-Multiplexer .....	199
9.8.2	Die Klasse IICMultiplexer .....	200
<b>10</b>	<b>USB-Komponenten .....</b>	<b>203</b>
10.1	Einführung .....	203
10.2	USB-Audioanbindung: MP3-Player und Sprachausgabe .....	204
10.3	USB-WLAN-Adapter .....	206
10.3.1	Grundlagen .....	206
10.3.2	Netgear MA111 unter Puppy .....	207
10.3.3	Alternative: WLAN-Anbindung über Access Point .....	209
10.4	USB-Bluetooth-Erweiterung .....	210
10.4.1	Grundlagen .....	210
10.4.2	Die Werkzeuge <i>bluez-Utils</i> .....	211
10.4.3	Datentransfer mit ObexFTP .....	216
10.4.4	Serielle Bluetooth-Kommunikation und AT-Befehle .....	217
10.4.5	Das Mobiltelefon als Fernbedienung .....	219
10.5	USB-GPS-Module .....	222
10.5.1	Der GPS-Daemon GPSD .....	223
10.5.2	GPS in der Anwendung .....	224
10.5.3	Die Klasse GPSReceiver .....	225
10.6	USB-Speichererweiterung .....	226
10.6.1	Partitionierung und Einbindung eines USB-Sticks .....	226
10.6.2	Auslagerung des Home-Verzeichnisses auf einen USB-Stick .....	228
<b>11</b>	<b>Gerätetreiber und Kernelmodule .....</b>	<b>231</b>
11.1	Einführung .....	231
11.2	Grundlagen .....	232
11.2.1	Systemarchitektur .....	232
11.2.2	Der Kernel .....	234

11.3	Programmierung von Kernelmodulen . . . . .	237
11.3.1	Aufbau von Kernelmodulen . . . . .	237
11.3.2	Übersetzung von Kernelmodulen . . . . .	239
11.3.3	Test und Debugging . . . . .	239
11.3.4	Übergabe von Kommandozeilenparametern . . . . .	242
11.4	Zeichenorientierte Gerätetreiber . . . . .	243
11.4.1	Major-, Minor- und Geräteummern . . . . .	243
11.4.2	Modul-Registrierung . . . . .	245
11.4.3	Gerätetreiber-Registrierung nach alter Schule . . . . .	248
11.5	Implementierung von Dateioperationen . . . . .	249
11.5.1	Die Struktur <code>file_operations</code> . . . . .	249
11.5.2	Kopieren von Daten zwischen Kernel- und User-Space . . . . .	250
11.5.3	Die <code>ioctl()</code> -Schnittstelle . . . . .	254
11.5.4	Verwendung von Gerätetreibern in der Anwendung . . . . .	255
11.6	Hardware-Zugriff . . . . .	257
11.6.1	Zugriff über IO-Ports und IO-Speicher . . . . .	257
11.6.2	Zugriff über das Dateisystem . . . . .	260
<b>12</b>	<b>Multithreading</b> . . . . .	<b>263</b>
12.1	Einführung . . . . .	263
12.2	Grundlagen . . . . .	264
12.3	Posix-Schnittstelle . . . . .	269
12.3.1	Thread-Funktionen . . . . .	270
12.3.2	Mutex-Funktionen . . . . .	271
12.3.3	Funktionen für Zustandsvariablen . . . . .	272
12.3.4	Beispiel . . . . .	273
12.4	C++-Schnittstelle . . . . .	275
12.4.1	Die Klasse Thread . . . . .	275
12.4.2	Die Klasse Mutex . . . . .	277
12.4.3	Die Klasse WaitCondition . . . . .	279
12.4.4	Die Klasse PeriodicThread . . . . .	282
12.5	Anwendungsbeispiel: Servo-Ansteuerung . . . . .	284
12.5.1	Servo-Anbindung an einen PC . . . . .	285
12.5.2	Software-Entwurf zum Beispiel . . . . .	286
12.5.3	Linux und Echtzeitfähigkeit . . . . .	288
12.5.4	Zeitmessung . . . . .	290
<b>13</b>	<b>Netzwerkkommunikation</b> . . . . .	<b>295</b>
13.1	Einführung . . . . .	295
13.2	Datenübertragung via UDP . . . . .	297
13.2.1	Grundlagen zu Sockets . . . . .	297
13.2.2	Berkeley Sockets . . . . .	300
13.2.3	Verwendung der Berkeley Socket API . . . . .	307
13.2.4	Socket-Debugging mit NetCat . . . . .	310
13.2.5	Host Byte Order und Network Byte Order . . . . .	310

13.2.6	Practical Sockets .....	312
13.2.7	Definition eigener Protokolle auf Anwendungsschicht ...	313
13.2.8	Verwendung der Practical Sockets .....	318
13.3	Kommunikation mit einer Qt-Anwendung .....	320
13.3.1	Client-Server-Kommunikation mit Qt4 .....	321
13.3.2	Remote-Schrittmotorsteuerung mit grafischer Benutzeroberfläche .....	327
13.4	Interaktion mit einem Webserver via CGI .....	333
13.4.1	Messdatenanzeige .....	336
13.4.2	Gezielte Anfragen mit JavaScript .....	338
<b>14</b>	<b>Video for Linux</b> .....	341
14.1	Einführung .....	341
14.2	Treiberinstallation und Inbetriebnahme .....	341
14.3	Bildeinzug unter Linux per V4L .....	345
14.4	Treiberkapselung für die IVT-Bibliothek .....	352
<b>15</b>	<b>Intelligente Kamera</b> .....	355
15.1	Einführung .....	355
15.2	Sicherheitssystem mit Bewegungserkennung .....	355
15.3	Weiterführende Informationen .....	358
15.3.1	Kommentare zum Hardware-Aufbau .....	358
15.3.2	Triggerung und IEEE 1394-Übertragung .....	360
15.3.3	Weitere Anwendungen .....	362
<b>16</b>	<b>Ausblick</b> .....	365
16.1	Communities, Projekte, Trends .....	365
16.2	Schlusswort und Kontaktdaten .....	369

---

## Teil III Anhang

---

<b>A</b>	<b>Kurzreferenzen</b> .....	373
A.1	Einführung .....	373
A.2	Die Linux-Konsole .....	373
A.2.1	Basisbefehlsschatz .....	373
A.2.2	Editoren .....	377
A.3	Netzwerkeinstellungen und SSH .....	380
A.3.1	Netzwerkeinstellungen .....	380
A.3.2	Secure Shell .....	382
A.4	Weitere Werkzeuge und Dienste .....	384
A.4.1	Paketverwaltung APT .....	384
A.4.2	Umgebungsvariablen .....	386
A.4.3	Erstellung von Gerätedateien mit <code>mknod</code> .....	387
A.4.4	Zugriffsrechte .....	388

A.4.5	Root-Rechte mit <code>sudo</code> .....	390
A.4.6	Cronjob-Verwaltung mit <code>crontab</code> .....	391
A.5	Diagnose- und Failsafe-Modi .....	393
A.5.1	Asus WL500g Premium .....	393
A.5.2	Linksys WRT54G .....	393
A.5.3	Linksys NSLU2 .....	394
<b>B</b>	<b>Alternative Hardware-Plattformen</b> .....	395
B.1	Einführung .....	395
B.2	Router .....	395
B.3	Network Attached Storage .....	395
B.4	Industrielle Kompaktsysteme .....	396
B.5	Einplatinencomputer .....	396
B.6	Sonderlösungen .....	396
<b>C</b>	<b>Die IVT-Bibliothek</b> .....	399
C.1	Einführung .....	399
C.2	Architektur .....	400
C.2.1	Die Klasse <code>CByteImage</code> .....	400
C.2.2	Anbindung von grafischen Benutzeroberflächen .....	401
C.2.3	Anbindung von Bildquellen .....	402
C.2.4	Anbindung der <code>OpenCV</code> .....	403
C.2.5	Anbindung von <code>OpenGL</code> über <code>Qt</code> .....	404
C.3	Beispielapplikationen.....	405
C.3.1	Verwendung der Basisfunktionalität .....	405
C.3.2	Verwendung einer grafischen Benutzeroberfläche .....	405
C.3.3	Verwendung eines Kameramoduls .....	405
C.3.4	Verwendung der <code>OpenCV</code> .....	406
C.3.5	Verwendung der <code>OpenGL</code> -Schnittstelle .....	406
C.4	Übersicht zu weiterer Funktionalität der IVT .....	407
C.5	Installation .....	408
C.5.1	<code>OpenCV</code> .....	409
C.5.2	<code>Qt</code> .....	409
C.5.3	Firewire und <code>libdc1394/libraw1394</code> .....	410
C.5.4	IVT .....	411
<b>D</b>	<b>Die Qt-Bibliothek</b> .....	417
D.1	Einführung .....	417
D.1.1	Installation und Grundlagen .....	417
D.1.2	Signals und Slots .....	420
D.1.3	Ein universelles <code>Qt-Makefile</code> .....	424
D.2	Oberflächenerstellung mit <code>Qt Designer</code> .....	425
D.2.1	Installation und Grundlagen .....	425
D.2.2	Verwendung der <code>Qt Designer Plugins</code> .....	428
D.2.3	Erstellung der <code>Qt Designer Plugins</code> .....	430