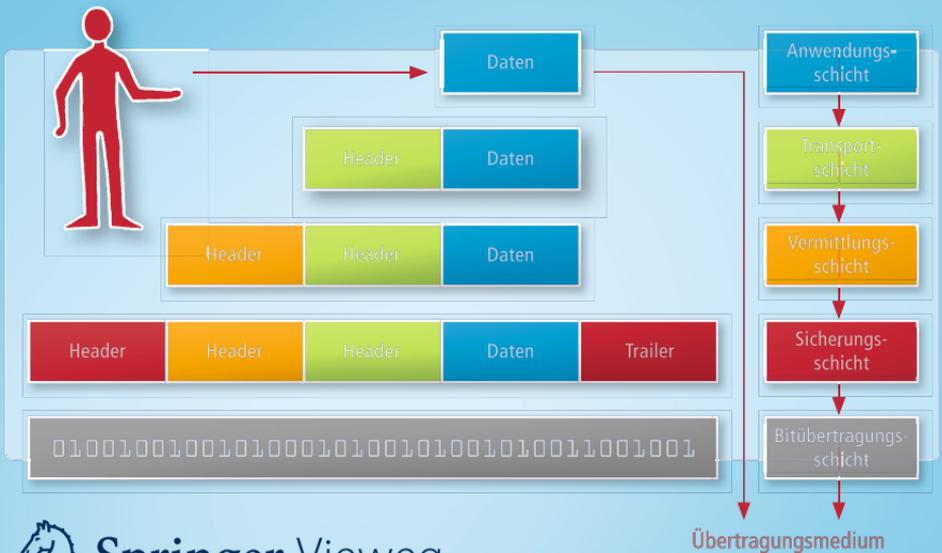


Christian Baun

# Computernetze kompakt

4. Auflage



IT kompakt

Werke der „kompakt-Reihe“ zu wichtigen Konzepten und Technologien der IT-Branche:

- ermöglichen einen raschen Einstieg,
- bieten einen fundierten Überblick,
- sind praxisorientiert, aktuell und immer ihren Preis wert.

Weitere Bände in der Reihe

<http://www.springer.com/series/8297>

Christian Baun

# Computernetze kompakt

4. Auflage



**Springer** Viewweg

Christian Baun  
University of Applied Sciences  
Frankfurt am Main, Deutschland

ISSN 2195-3651

ISSN 2195-366X (electronic)

IT kompakt

ISBN 978-3-662-57468-3

ISBN 978-3-662-57469-0 (eBook)

<https://doi.org/10.1007/978-3-662-57469-0>

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

Springer Vieweg

© Springer-Verlag GmbH Deutschland, ein Teil von Springer Nature 2012, 2013, 2015, 2018

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt 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.

Der Verlag, die Autoren und die Herausgeber gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag noch die Autoren oder die Herausgeber übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen. Der Verlag bleibt im Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutionsadressen neutral.

Gedruckt auf säurefreiem und chlorfrei gebleichtem Papier

Springer Vieweg ist ein Imprint der eingetragenen Gesellschaft Springer-Verlag GmbH, DE und ist ein Teil von Springer Nature.

Die Anschrift der Gesellschaft ist: Heidelberger Platz 3, 14197 Berlin, Germany

---

# Vorwort zur 4. Auflage

Diese Auflage enthält außer neuen Inhalten auch zahlreiche didaktische Verbesserungen. Die Beschreibung der komplementären Signale bei Twisted-Pair-Kabeln in Abschn. 5.2.2 ist nun anschaulicher. Die Auswirkungen von Kreisen auf der Sicherungsschicht werden in Abschn. 6.1.2 anhand eines Beispiels demonstriert. Abschn. 6.4 beschreibt nun ausführlicher, wie die einzelnen Rahmen von Ethernet und WLAN abgegrenzt sind. Umfangreich überarbeitet wurde in Abschn. 6.8.2 die Beschreibung der verschiedenen Medienzugriffsverfahren von WLAN. Abschn. 8.4.2 beschreibt die Verbindungszustände des Transportprotokolls TCP noch anschaulicher.

Neu hinzugekommen sind Abschn. 6.7 zur Fehlerkorrektur und Abschn. 6.9 zur Flusskontrolle in der Sicherungsschicht. Weitere neue Abschnitte sind Abschn. 7.2 (IPv6), Abschn. 7.7 (NAT) und Abschn. 9.4 (HTTP/2).

Ein Aspekt der Computernetze, der aus Platzgründen auch in dieser Auflage quasi ignoriert wird, ist der Themenkomplex Netzwerksicherheit. Als Literatur zu diesem Thema empfehle ich das Standardwerk von Martin Kappes [1].

Wie bei den vorherigen Auflagen auch, wird die Errata-Liste zu dieser Auflage auf meiner Webseite<sup>1</sup> veröffentlicht, sobald Verbesserungen vorliegen.

An dieser Stelle möchte ich Marcus Legendre und ganz besonders Katrin Baun für das Korrekturlesen danken.

Frankfurt am Main  
Februar 2018

Prof. Dr. Christian Baun

---

<sup>1</sup> <http://www.christianbaun.de>

---

# Vorwort zur 3. Auflage

Erneut erscheint nach wenig mehr als einem Jahr eine neue Auflage. Diese enthält im Vergleich zur 2. Auflage zahlreiche Korrekturen. Verbessert wurden unter anderem die Abbildungen zu Broadcast-Domänen und Kollisionsdomänen und die Unterscheidung zwischen Router und Layer-3-Switch.

Neu hinzugekommen ist Kap. 11, das eine Übersicht über wichtige Kommandozeilenwerkzeuge zur Netzwerkkonfiguration und Analyse von Netzwerkproblemen enthält. Der Aufbau des Kapitels orientiert sich an den Schichten im hybriden Referenzmodell und damit auch am Aufbau dieses Buches.

An dieser Stelle möchte ich Kristian Kraljic, Tobias Kurze, Michael Stapelberg, Torsten Wiens und ganz besonders Katrin Baun für das Korrekturlesen des neuen Kapitels danken.

Frankfurt am Main  
Februar 2015

Prof. Dr. Christian Baun

---

# Vorwort zur 2. Auflage

Aufgrund der positiven Rückmeldungen und Verbesserungsvorschläge haben ich nach nur knapp einem Jahr seit Erscheinen der 1. Auflage einige Teile des Buches überarbeitet und erweitert.

Neu hinzugekommen ist Abschn. 3.10, der eine Beschreibung der Themen Bandbreite und Latenz und des damit verbundenen Bandbreite-Verzögerung-Produkts enthält.

Umfangreich erweitert wurde das Thema Rahmen bei WLAN in Abschn. 6.4. Neu hinzugekommen ist ebenfalls Abschn. 6.5, der die Maximum Transmission Unit (MTU) beschreibt.

Die sehr knappe Beschreibung der verschiedenen Routing-Protokolle ist einer detaillierteren Betrachtung der beiden Routing-Protokolle RIP und OSPF in den Abschn. 7.4 und 7.5 gewichen. Neu hinzugekommen ist eine ausführliche Beschreibung des *Internetworking*, also der Kommunikation über verschiedene physische Netze in Abschn. 7.6.

Die Beschreibung des Transportprotokolls TCP in Abschn. 8.4 wurde unter anderem um die Themen Flusskontrolle und Überlastkontrolle erweitert.

Überarbeitet wurden auch die Abschn. 9.1 und 9.2, in denen die Anwendungsprotokolle DNS und DHCP beschrieben sind.

Komplett neu hinzugekommen ist Kap. 10, das eine Einführung in die Netzwerkvirtualisierung enthält.

Frankfurt am Main  
September 2013

Prof. Dr. Christian Baun

---

# Vorwort zur 1. Auflage

Computernetze sind ein dauerhaft aktuelles Thema der Informatik. Waren in der Vergangenheit die kabelgebundene Vernetzung von PCs und Workstations sowie das globale Internet dominierende Themen, sind es heute zunehmend mobile Systeme und Funknetze.

Dieses kompakte Werk über das breite Thema Computernetze wurde mit dem Ziel geschrieben, dem Leser einen Überblick über die wichtigsten Vernetzungstechnologien, Netzwerkgeräte, Protokolle und Übertragungsmedien zu verschaffen und so das Verständnis dafür zu wecken, wie Computernetze funktionieren.

Für das Korrekturlesen danke ich Maximilian Hoecker, Kristian Kraljic, Michael Stapelberg und Torsten Wiens.

An dieser Stelle möchte ich ganz besonders meiner Familie danken, die mich auch bei diesem Werk so viel unterstützt hat.

Mannheim  
Juli 2012

Dr. Christian Baun

---

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b> .....	<b>1</b>
<b>2</b>	<b>Grundlagen der Informationstechnik</b> .....	<b>3</b>
2.1	Bit .....	3
2.2	Repräsentation von Zahlen .....	4
2.2.1	Dezimalsystem .....	5
2.2.2	Dualsystem .....	5
2.2.3	Oktalsystem .....	6
2.2.4	Hexadezimalsystem .....	7
2.3	Datei- und Speichergrößen .....	7
2.4	Informationsdarstellung .....	9
2.4.1	ASCII-Kodierung .....	10
2.4.2	Unicode .....	12
2.4.3	Darstellung von Zeichenketten .....	13
<b>3</b>	<b>Grundlagen der Computervernetzung</b> .....	<b>15</b>
3.1	Entwicklung der Computernetze .....	15
3.2	Komponenten von Computernetzen .....	17
3.3	Räumliche Ausdehnung von Computernetzen .....	17
3.4	Datenübertragung .....	18
3.4.1	Serielle und parallele Übertragung .....	19
3.4.2	Synchrone und asynchrone Übertragung .....	19
3.4.3	Richtungsabhängigkeit der Übertragung .....	20
3.5	Geräte in Computernetzen .....	21
3.6	Topologien von Computernetzen .....	23
3.6.1	Bus-Topologie .....	24

---

3.6.2	Ring-Topologie	25
3.6.3	Stern-Topologie	25
3.6.4	Maschen-Topologie	26
3.6.5	Baum-Topologie	26
3.6.6	Zellen-Topologie	27
3.7	Frequenz und Datensignal	27
3.8	Fourierreihe und Bandbreite	28
3.9	Bitrate und Baudrate	28
3.10	Bandbreite und Latenz	29
3.10.1	Bandbreite-Verzögerung-Produkt	30
3.11	Zugriffsverfahren	31
3.11.1	Deterministisches Zugriffsverfahren	31
3.11.2	Nicht-deterministisches Zugriffsverfahren	32
3.12	Kollisionsdomäne (Kollisionsgemeinschaft)	32
<b>4</b>	<b>Protokolle und Protokollschichten</b>	<b>35</b>
4.1	TCP/IP-Referenzmodell	36
4.2	Hybrides Referenzmodell	37
4.2.1	Bitübertragungsschicht	37
4.2.2	Sicherungsschicht	38
4.2.3	Vermittlungsschicht	39
4.2.4	Transportschicht	39
4.2.5	Anwendungsschicht	40
4.3	Ablauf der Kommunikation	41
4.4	OSI-Referenzmodell	42
4.4.1	Sitzungsschicht	42
4.4.2	Darstellungsschicht	43
4.5	Fazit zu den Referenzmodellen	43
<b>5</b>	<b>Bitübertragungsschicht</b>	<b>45</b>
5.1	Vernetzungstechnologien	45
5.1.1	Ethernet	45
5.1.2	Token Ring	47
5.1.3	Wireless Local Area Network (WLAN)	48
5.1.4	Bluetooth	60
5.2	Übertragungsmedien	62
5.2.1	Koaxialkabel	63

---

5.2.2	Twisted-Pair-Kabel . . . . .	64
5.2.3	Lichtwellenleiter . . . . .	71
5.3	Strukturierte Verkabelung . . . . .	72
5.4	Geräte der Bitübertragungsschicht . . . . .	73
5.4.1	Auswirkungen von Repeatern und Hubs auf die Kollisionsdomäne . . . . .	75
5.5	Kodierung von Daten in Netzwerken . . . . .	75
5.5.1	Non-Return to Zero (NRZ) . . . . .	76
5.5.2	Non-Return to Zero Invert (NRZI) . . . . .	79
5.5.3	Multilevel Transmission Encoding – 3 Levels (MLT-3) . . . . .	79
5.5.4	Return-to-Zero (RZ) . . . . .	80
5.5.5	Unipolares RZ . . . . .	81
5.5.6	Alternate Mark Inversion (AMI) . . . . .	81
5.5.7	Bipolar with 8 Zeros Substitution (B8ZS) . . . . .	82
5.5.8	Manchester . . . . .	82
5.5.9	Manchester II . . . . .	83
5.5.10	Differentielle Manchesterkodierung . . . . .	84
5.6	Nutzdaten mit Blockcodes verbessern . . . . .	84
5.6.1	4B5B . . . . .	85
5.6.2	5B6B . . . . .	87
5.6.3	8B10B-Kodierung . . . . .	88
5.7	Weitere Leitungscodes . . . . .	88
5.7.1	8B6T-Kodierung . . . . .	88
<b>6</b>	<b>Sicherungsschicht . . . . .</b>	<b>91</b>
6.1	Geräte der Sicherungsschicht . . . . .	91
6.1.1	Lernende Bridges . . . . .	93
6.1.2	Kreise auf der Sicherungsschicht . . . . .	94
6.1.3	Spanning Tree Protocol (STP) . . . . .	99
6.1.4	Auswirkungen von Bridges und Layer-2-Switches auf die Kollisionsdomäne . . . . .	103
6.2	Adressierung in der Sicherungsschicht . . . . .	103
6.2.1	Format der MAC-Adressen . . . . .	104
6.2.2	Eindeutigkeit von MAC-Adressen . . . . .	104
6.2.3	Sicherheit von MAC-Adressen . . . . .	105
6.3	Rahmen abgrenzen . . . . .	106

6.3.1	Längenangabe im Header . . . . .	106
6.3.2	Zeichenstopfen . . . . .	107
6.3.3	Bitstopfen . . . . .	108
6.3.4	Verstöße gegen Regeln des Leitungscodes . . . . .	108
6.4	Rahmenformate aktueller Computernetze . . . . .	109
6.4.1	Rahmen bei Ethernet . . . . .	109
6.4.2	Rahmen bei WLAN . . . . .	112
6.4.3	Spezielle Rahmen bei WLAN . . . . .	116
6.5	Maximum Transmission Unit (MTU) . . . . .	117
6.6	Fehlererkennung . . . . .	117
6.6.1	Zweidimensionale Parität . . . . .	117
6.6.2	Zyklische Redundanzprüfung . . . . .	118
6.7	Fehlerkorrektur . . . . .	120
6.8	Medienzugriffsverfahren . . . . .	121
6.8.1	Medienzugriffsverfahren bei Ethernet . . . . .	121
6.8.2	Medienzugriffsverfahren bei WLAN . . . . .	125
6.9	Flusskontrolle . . . . .	131
6.10	Adressauflösung mit dem Address Resolution Protocol . . . . .	132
<b>7</b>	<b>Vermittlungsschicht . . . . .</b>	<b>135</b>
7.1	Geräte der Vermittlungsschicht . . . . .	136
7.1.1	Auswirkungen von Routern auf die Kollisionsdomäne . . . . .	136
7.1.2	Broadcast-Domäne (Rundsendedomäne) . . . . .	137
7.2	Adressierung in der Vermittlungsschicht . . . . .	137
7.2.1	Aufbau von IPv4-Adressen . . . . .	139
7.2.2	Subnetze im IPv4-Adressraum . . . . .	141
7.2.3	Private IPv4-Adressen . . . . .	144
7.2.4	Aufbau von IPv4-Paketen . . . . .	145
7.2.5	Fragmentieren von IPv4-Paketen . . . . .	147
7.2.6	Darstellung und Aufbau von IPv6-Adressen . . . . .	147
7.2.7	Darstellung von Netzen im IPv6-Adressraum . . . . .	149
7.2.8	Einige IPv6-Adressbereiche . . . . .	151
7.2.9	IPv4-Adressen im IPv6-Adressraum einbetten . . . . .	151
7.2.10	Aufbau von IPv6-Paketen . . . . .	152

---

7.3	Weiterleitung und Wegbestimmung	153
7.4	Routing Information Protocol (RIP)	154
7.4.1	Count-to-Infinity	155
7.4.2	Split Horizon	156
7.4.3	Fazit zu RIP	158
7.5	Open Shortest Path First (OSPF)	161
7.5.1	Routing-Hierarchie mit OSPF	161
7.5.2	Arbeitsweise von OSPF	162
7.5.3	Aufbau von OSPF-Nachrichten	164
7.5.4	Fazit zu OSPF	165
7.6	Netzübergreifende Kommunikation	167
7.7	Network Address Translation	169
7.8	Diagnose und Fehlermeldungen mit ICMP	172
<b>8</b>	<b>Transportschicht</b>	<b>175</b>
8.1	Eigenschaften von Transportprotokollen	175
8.2	Adressierung in der Transportschicht	176
8.3	User Datagram Protocol (UDP)	178
8.3.1	Aufbau von UDP-Segmenten	178
8.4	Transmission Control Protocol (TCP)	179
8.4.1	Aufbau von TCP-Segmenten	180
8.4.2	Arbeitsweise von TCP	182
8.4.3	Flusskontrolle	187
8.4.4	Überlastkontrolle	191
<b>9</b>	<b>Anwendungsschicht</b>	<b>199</b>
9.1	Domain Name System (DNS)	199
9.1.1	Arbeitsweise des DNS	200
9.1.2	Auflösung eines Domainnamens	201
9.2	Dynamic Host Configuration Protocol (DHCP)	203
9.2.1	Arbeitsweise von DHCP	203
9.2.2	Aufbau von DHCP-Nachrichten	205
9.2.3	DHCP-Relay	206
9.3	Telecommunication Network (Telnet)	207
9.4	Hypertext Transfer Protocol (HTTP)	208
9.5	Simple Mail Transfer Protocol (SMTP)	211
9.6	Post Office Protocol Version 3 (POP3)	213

---

9.7	File Transfer Protocol (FTP)	213
<b>10</b>	<b>Netzwerkvirtualisierung</b>	217
10.1	Virtual Private Networks (VPN)	217
10.1.1	Technische Arten von VPNs	218
10.2	Virtual Local Area Networks (VLAN)	219
<b>11</b>	<b>Kommandozeilenwerkzeuge</b>	223
11.1	ethtool	226
11.2	mii-tool	226
11.3	iwconfig	227
11.4	iwlist	228
11.5	arp	229
11.6	ifconfig	229
11.7	ip	230
11.8	iftop	231
11.9	tcpdump	232
11.10	route	232
11.11	ping	233
11.12	traceroute	233
11.13	netstat	234
11.14	nmap	235
11.15	dhclient	235
11.16	dig	236
11.17	ftp	236
11.18	nc	236
11.19	netperf	237
11.20	nslookup	238
11.21	ssh	238
11.22	telnet	239
<b>Glossar</b>		241
<b>Literatur</b>		247
<b>Sachverzeichnis</b>		249

Dieses Buch will einen Überblick über Computernetze und deren Komponenten schaffen, ohne dabei den Anspruch auf Vollständigkeit zu erheben. Das Ziel ist es, den Leserinnen und Lesern ein grundlegendes Wissen über die Funktionsweise moderner Computernetze und deren Komponenten zu vermitteln. Technische Vorkenntnisse sind dabei nicht erforderlich.

In den Kap. 2 und 3 findet eine Einführung in die Grundlagen der Informationstechnik (IT) und der Computervernetzung statt. Dies ist nötig, um die Thematik Computernetze und den Inhalt dieses Buchs verstehen zu können.

Kap. 4 beschäftigt sich mit den Grundlagen von Protokollen und deren Einordnung in etablierte Referenzmodelle. Das Kapitel beschreibt auch den Ablauf der Kommunikation und den Weg der Nutzdaten durch die einzelnen Protokollschichten.

In den Kap. 5 bis 9 werden die einzelnen Protokollschichten von der untersten Schicht, der Bitübertragungsschicht, bis zur obersten Schicht, der Anwendungsschicht, behandelt.

In Kap. 10 findet eine Einführung in die Varianten der Netzwerkvirtualisierung statt.

Kap. 11 enthält eine Übersicht über wichtige Kommandozeilenwerkzeuge zur Netzwerkkonfiguration und Analyse von Netzwerkproblemen.

Um die Funktionsweise der Computernetze zu verstehen, ist ein grundlegendes Verständnis der Informationstechnik (IT) nötig. Bei diesen Grundlagen handelt es sich um die Möglichkeiten der Informationsdarstellung und Repräsentation von Zahlen, Größenordnungen und die Art und Weise, wie Informationen (speziell Texte) in Rechnern dargestellt werden.

## 2.1 Bit

Ein *Bit* ist die kleinstmögliche Einheit der Information und jede Information ist an einen Informationsträger gebunden [5]. Ein Informationsträger, der sich in genau einem von zwei Zuständen befinden kann, kann die Datenmenge 1 Bit darstellen. Den Wert eines oder mehrerer Bits nennt man *Zustand*. Ein Bit stellt zwei Zustände dar. Verschiedene Sachverhalte können die Datenmenge 1 Bit darstellen. Beispiele sind:

- Die Stellung eines Schalters mit zwei Zuständen
- Der Schaltzustand eines Transistors
- Das Vorhandensein einer elektrischen Spannung oder Ladung
- Das Vorhandensein einer Magnetisierung
- Der Wert einer Variable mit den logischen Wahrheitswerten.

**Tab. 2.1** Die Anzahl der darstellbaren Zustände verdoppelt sich mit jedem zusätzlichen Bit

Bits	Zustände	Bits	Zustände	Bits	Zustände
1	$2^1 = 2$	9	$2^9 = 512$	17	$2^{17} = 131.072$
2	$2^2 = 4$	10	$2^{10} = 1.024$	18	$2^{18} = 262.144$
3	$2^3 = 8$	11	$2^{11} = 2.048$	19	$2^{19} = 524.288$
4	$2^4 = 16$	12	$2^{12} = 4.096$	20	$2^{20} = 1.048.576$
5	$2^5 = 32$	13	$2^{13} = 8.192$	21	$2^{21} = 2.097.152$
6	$2^6 = 64$	14	$2^{14} = 16.384$	22	$2^{22} = 4.194.304$
7	$2^7 = 128$	15	$2^{15} = 32.768$	23	$2^{23} = 8.388.608$
8	$2^8 = 256$	16	$2^{16} = 65.536$	24	$2^{24} = 16.777.216$

Benötigt man zur Speicherung einer Information mehr als zwei Zustände, sind Folgen von Bits (*Bitfolgen*) nötig. Mit  $n$  Bits kann man  $2^n$  verschiedene Zustände darstellen (siehe Tab. 2.1). Also kann man mit 2 Bits  $2^2 = 4$  verschiedene Zustände repräsentieren, nämlich 00, 01, 10 und 11. Mit 3 Bits kann man schon  $2^3 = 8$  verschiedene Zustände (000, 001, 010, 011, 100, 101, 110 und 111) repräsentieren. Jedes zusätzliche Bit verdoppelt die Anzahl der möglichen darstellbaren Zustände (Bitfolgen) [5].

## 2.2 Repräsentation von Zahlen

Zahlen kann man auf unterschiedliche Arten darstellen. Eine Aufgabe der IT ist es, Zahlen aus der *realen Welt* im Computer abzubilden. Wichtig ist dabei die Unterscheidung zwischen *Wert* und *Darstellung*.

In der Mathematik unterscheidet man Zahlen als Elemente verschiedener Wertemengen (natürliche Zahlen, ganze Zahlen, reelle Zahlen, komplexe Zahlen, usw.). Den Wert einer Zahl nennt man auch *abstrakte Zahl* und der Wert ist unabhängig von der Darstellung (zum Beispiel  $0,5 = 1/2$ ).

Operationen eines Rechners werden aber nicht auf Werten, sondern auf Bitfolgen ausgeführt. Darum ist für die IT besonders die Darstellung der Zahlen interessant. Die Darstellung wird vom verwendeten Stellenwertsystem (Positionssystem) bestimmt. Die für die IT wichtigen Stel-

lenwertssysteme sind das Dezimalsystem, das Dualsystem, das Oktalsystem und das Hexadezimalsystem.

### 2.2.1 Dezimalsystem

Das Dezimalsystem verwendet als Basis die Zahl 10. Jede Ziffer  $D$  an der Stelle  $i$  hat den Wert  $D \times 10^i$ . Ein Beispiel ist:

$$2.013 = 2 \times 10^3 + 0 \times 10^2 + 1 \times 10^1 + 3 \times 10^0$$

Computer-Systeme unterscheiden prinzipiell zwischen zwei elektrischen Zuständen. Darum ist aus Sicht der IT als Basis die Zahl 2 und damit das Dualsystem optimal geeignet.

### 2.2.2 Dualsystem

Das Dualsystem verwendet als Basis die Zahl 2. Zahlen werden nur mit den Ziffern der Werte Null und Eins dargestellt. Zahldarstellungen im Dualsystem heißen Dualzahlen oder Binärzahlen. Alle positiven natürlichen Zahlen inklusive der Null können durch Folgen von Symbolen aus der Menge  $\{1, 0\}$  repräsentiert werden. Das niederwertigste Bit ( $x_0$ ) heißt *Least Significant Bit* (LSB) und das höchstwertigste Bit ( $x_{n-1}$ ) ist das *Most Significant Bit* (MSB), wobei  $n$  der Anzahl der Bits entspricht.

Da lange Reihen von Nullen und Einsen für Menschen schnell unübersichtlich werden, verwendet man zur Darstellung von Bitfolgen häufig das Oktalsystem oder das Hexadezimalsystem.

Die Umrechnung der Stellenwertsysteme ist einfach möglich. Zur Verdeutlichung ist das Stellenwertsystem der jeweiligen Zahl in den folgenden Beispielen tiefgestellt beigefügt.

Bei der Umwandlung von Dualzahlen in Dezimalzahlen werden die Ziffern mit ihren Stellenwertigkeiten ausmultipliziert und die Ergebnisse addiert.

$$10100100_2 = 2^7 + 2^5 + 2^2 = 164_{10}$$

**Tab. 2.2** Die Dezimalzahl  $164_{10}$  in die Dualzahl  $10100100_2$  umwandeln

	<i>Quotient</i>	<i>Rest</i>
<i>k</i>	<i>k DIV 2</i>	<i>k MODULO 2</i>
164	82	$0 = x_1$
82	41	$0 = x_2$
41	20	$1 = x_3$
20	10	$0 = x_4$
10	5	$0 = x_5$
5	2	$1 = x_6$
2	1	$0 = x_7$
1	0	$1 = x_8$

Die Umwandlung von Dezimalzahlen in Dualzahlen ist unter anderem mit Hilfe des Restwertalgorithmus (siehe Tab. 2.2) möglich. Dabei wird die Dezimalzahl durch die Basis 2 dividiert und das Ergebnis und der Rest (Wert Null oder Eins) werden notiert. Das Ergebnis der Division wird in der nächsten Runde (Zeile der Tabelle) erneut durch die Basis dividiert und erneut werden das Ergebnis und der Rest notiert. Der Algorithmus wird so lange weitergeführt, bis das Ergebnis der Division Null ist.

### 2.2.3 Oktalsystem

Das Oktalsystem verwendet als Basis die Zahl 8 und kann Gruppen von 3 Bits (Oktade) mit einem Zeichen darstellen.

Bei der Umwandlung von Dualzahlen in Oktalzahlen wird die Bitkette vom niederwertigsten Bit beginnend in Oktaden unterteilt. Jede Oktade entspricht einer Stelle der Oktalzahl.

$$164_{10} = 10|100|100_2 = 244_8$$

Die Umwandlung von Oktalzahlen in Dualzahlen erfolgt analog. Eine Stelle im Oktalsystem entspricht drei Stellen im Dualsystem.

## 2.2.4 Hexadezimalsystem

Das Hexadezimalsystem verwendet als Basis die Zahl 16. Die Darstellung positiver natürlicher Zahlen erfolgt mit den 16 Ziffernsymbolen aus der Menge  $\{0, 1, \dots, 8, 9, A, B, C, D, E, F\}$ . Ein Zeichen kann eine Gruppe von 4 Bits (Tetrade oder Nibble) darstellen.

Bei der Umwandlung von Dualzahlen in Hexadezimalzahlen wird die Bitkette vom niederwertigsten Bit beginnend in Tetraden unterteilt. Jede Tetrade entspricht einer Stelle der Hexadezimalzahl.

$$164_{10} = 1010|0100_2 = A4_{16}$$

Die Umwandlung von Hexadezimalzahlen in Dualzahlen geschieht analog. Eine Stelle im Hexadezimalsystem entspricht vier Stellen im Dualsystem.

Tab. 2.3 enthält eine Übersicht der verschiedenen Darstellungen der ersten 16 positiven natürlichen Zahlen im Dezimalsystem, Dualsystem, Oktalsystem und Hexadezimalsystem.

---

## 2.3 Datei- und Speichergrößen

Rechner lesen und schreiben aus Geschwindigkeitsgründen meist nicht einzelne Bits, sondern arbeiten mit Bitfolgen, deren Längen Vielfache von Acht sind. Eine Gruppe von 8 Bits nennt man *Byte*. Den Wert eines Bytes kann man entweder durch 8 Bits oder zwei Hexadezimalziffern darstellen.

Eine Datei ist eine beliebig lange Folge von Bytes und enthält inhaltlich zusammengehörende Daten. Alle Informationen (Zahlen, Texte, Musik, Programme, usw.), mit denen ein Rechner arbeiten soll, müssen als Folge von Bytes repräsentiert werden können und als Datei gespeichert werden [5].

Da sich die Größenordnungen der meisten Dateien im Bereich mehrerer Tausend oder Millionen Bytes befinden, gibt es verschiedene Größeneinheiten zur verkürzten Zahlendarstellung. Für Datenspeicher mit binärer Adressierung ergeben sich Speicherkapazitäten von  $2^n$  Byte, also Zweierpotenzen (siehe Tab. 2.4).

**Tab. 2.3** Verschiedene Darstellungen positiver natürlicher Zahlen

Dezimale Darstellung	Binäre Darstellung	Oktale Darstellung	Hexadezimale Darstellung
00	0000	00	0
01	0001	01	1
02	0010	02	2
03	0011	03	3
04	0100	04	4
05	0101	05	5
06	0110	06	6
07	0111	07	7
08	1000	10	8
09	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Die Maßeinheiten in Tab. 2.4 haben sich für die Größenangabe von Hauptspeicher und Speichermedien in Betriebssystemen eingebürgert. Die Hersteller von Festplatten, CD/DVDs und USB-Speichermedien bevorzugen zur Berechnung der Kapazität und zur Angabe auf der Ver-

**Tab. 2.4** Datei- und Speichergrößen

Name	Symbol	Bedeutung
Kilobyte	kB	$2^{10} = 1.024$ Bytes
Megabyte	MB	$2^{20} = 1.048.576$ Bytes
Gigabyte	GB	$2^{30} = 1.073.741.824$ Bytes
Terabyte	TB	$2^{40} = 1.099.511.627.776$ Bytes
Petabyte	PB	$2^{50} = 1.125.899.906.842.624$ Bytes
Exabyte	EB	$2^{60} = 1.152.921.504.606.846.976$ Bytes
Zettabyte	ZB	$2^{70} = 1.180.591.620.717.411.303.424$ Bytes
Yottabyte	YB	$2^{80} = 1.208.925.819.614.629.174.706.176$ Bytes

packung aber lieber Dezimal-Präfixe, also zum Beispiel den Faktor  $10^9$  anstatt  $2^{30}$  für GB und  $10^{12}$  anstatt  $2^{40}$  für TB. Aus diesem Grund wird zum Beispiel bei einem DVD-Rohling mit einer angegebenen Kapazität von 4,7 GB in vielen Anwendungen korrekterweise nur die Kapazität 4,38 GB angezeigt.

$$10^9 = 1.000.000.000, \quad 2^{30} = 1.073.741.824$$

Der Kapazitätsunterschied zwischen Zweierpotenz und Zehnerpotenz ist in diesem Fall ca. 7,37 %.

Bei größeren Speichern ist der Unterschied noch größer. So können von einer Festplatte mit angeblich 1 TB Speicherkapazität tatsächlich nur etwa 930 GB verwendet werden.

$$10^{12} = 1.000.000.000.000, \quad 2^{40} = 1.099.511.627.776$$

Der Kapazitätsunterschied zwischen Zweierpotenz und Zehnerpotenz ist in diesem Fall schon ca. 9,95 % und mit jeder weiteren Maßeinheit (PB, EB, ZB, usw.) wächst der Kapazitätsunterschied zwischen Zweierpotenzen und Zehnerpotenzen weiter.

Die International Electrotechnical Commission (IEC) schlug 1996 vor, die populären Größenfaktoren, die auf den Zweierpotenzen basieren, mit einem kleinen „i“ zu kennzeichnen und die etablierten Bezeichnungen der Maßeinheiten für die Zehnerpotenzen zu reservieren. Dieser Vorschlag konnte sich bislang nicht durchsetzen und die daraus resultierenden alternativen Bezeichnungen Kibibyte (KiB), Mebibyte (MiB), Gibibyte (GiB), Tebibyte (TiB), Pebibyte (PiB), Exbibyte (EiB) und Zebibyte (ZiB) sind außerhalb des akademischen Bereichs nicht besonders populär.

---

## 2.4 Informationsdarstellung

Daten sind Folgen von Nullen und Einsen, die beliebige Informationen repräsentieren. Um Texte und Zahlen durch Daten darzustellen, kodiert man die Zeichen des Alphabets (Groß- und Kleinschreibung), Satzzeichen wie Punkt, Komma und Semikolon, sowie einige Spezialzeichen

wie zum Beispiel +, %, & und \$ in Bitfolgen. Zudem sind Sonderzeichen wie Leerzeichen (SP), Wagenrücklauf (CR) und Tabulator (TAB) nötig. Die am besten etablierte Kodierung ist der *American Standard Code for Information Interchange (ASCII)*.

## 2.4.1 ASCII-Kodierung

Die ASCII-Kodierung, häufig auch US-ASCII genannt, ist eine 7-Bit-Zeichenkodierung. Das heißt, dass jedem Zeichen eine Bitfolge aus 7 Bits zugeordnet ist. Es existieren also  $2^7 = 128$  verschiedene Bitfolgen und exakt so viele Zeichen definiert die Zeichenkodierung (siehe Tab. 2.5). Von den 128 Zeichen sind 95 Zeichen druckbar und 33 Zeichen nicht druckbar. Die druckbaren Zeichen sind (beginnend mit dem Leerzeichen):

```
!"#$%&'()*+,-./0123456789:;<=>?
@ABCDEFGHIJKLMNOQRSTUVWXYZ[\]^_
`abcdefghijklmnopqrstuvwxyz{|}~
```

Die nicht druckbaren Zeichen  $00_{16}$  bis  $20_{16}$  und  $7F_{16}$ , zum Beispiel Backspace (BS) und Carriage Return (CR), sind Steuerzeichen, die ursprünglich zur Steuerung eines Fernschreibers verwendet wurden. ASCII ist also nicht nur ein Standard zur Datenablage, sondern auch zur Datenübertragung geeignet. Den Anfang und das Ende einer Datenübertragung markiert man mit Start of Text (STX) bzw. End of Text (ETX). Die Steuerung der Übertragung ist mit nicht druckbaren Zeichen wie Acknowledge (ACK) und negative Acknowledge (NAK) möglich. Mit Bell (BEL) kann ein Sender, zum Beispiel bei einem Fehler, ein Alarmsignal an den Empfänger senden.

Das 8. Bit kann als Paritätsbit zur Fehlererkennung verwendet werden. In diesem Fall hat es den Wert 0, wenn die Anzahl der Einsen an den übrigen sieben Bitpositionen gerade ist und ansonsten den Wert 1.

Durch verbesserte Protokolle zur Datenübertragung benötigt man das 8. Bit bei der Datenübertragung von ASCII-kodierten Texten nicht mehr zur Fehlererkennung. Darum wurde, um zusätzliche Zeichen kodieren zu können, US-ASCII mit zahlreichen Erweiterungen zu einer 8-Bit-Zeichenkodierung erweitert. Wird jedem Zeichen eine Bitfolge aus 8 Bits

**Tab. 2.5** Tabelle der ASCII-Zeichenkodierung (US-ASCII)

Dez	Hex	Zeichen									
000	00	NUL	032	20	Space	064	40	@	096	60	'
001	01	SOH	033	21	!	065	41	A	097	61	a
002	02	STX	034	22	"	066	42	B	098	62	b
003	03	ETX	035	23	#	067	43	C	099	63	c
004	04	EOT	036	24	\$	068	44	D	100	64	d
005	05	ENQ	037	25	%	069	45	E	101	65	e
006	06	ACK	038	26	&	070	46	F	102	66	f
007	07	BEL	039	27	'	071	47	G	103	67	g
008	08	BS	040	28	(	072	48	H	104	68	h
009	09	TAB	041	29	)	073	49	I	105	69	i
010	0A	LF	042	2A	*	074	4A	J	106	6A	j
011	0B	VT	043	2B	+	075	4B	K	107	6B	k
012	0C	FF	044	2C	,	076	4C	L	108	6C	l
013	0D	CR	045	2D	-	077	4D	M	109	6D	m
014	0E	SO	046	2E	.	078	4E	N	110	6E	n
015	0F	SI	047	2F	/	079	4F	O	111	6F	o
016	10	DLE	048	30	0	080	50	P	112	70	p
017	11	DC1	049	31	1	081	51	Q	113	71	q
018	12	DC2	050	32	2	082	52	R	114	72	r
019	13	DC3	051	33	3	083	53	S	115	73	s
020	14	DC4	052	34	4	084	54	T	116	74	t
021	15	NAK	053	35	5	085	55	U	117	75	u
022	16	SYN	054	36	6	086	56	V	118	76	v
023	17	ETB	055	37	7	087	57	W	119	77	w
024	18	CAN	056	38	8	088	58	X	120	78	x
025	19	EM	057	39	9	089	59	Y	121	79	y
026	1A	SUB	058	3A	:	090	5A	Z	122	7A	z
027	1B	ESC	059	3B	;	091	5B	[	123	7B	{
028	1C	FS	060	3C	<	092	5C	\	124	7C	
029	1D	GS	061	3D	=	093	5D	]	125	7D	}
030	1E	RS	062	3E	>	094	5E	^	126	7E	~
031	1F	US	063	3F	?	095	5F	_	127	7F	DEL

zugeordnet, sind  $2^8 = 256$  verschiedene Bitfolgen verfügbar. Es sind also 128 Zeichen mehr verfügbar, als bei US-ASCII. Da diese 128 zusätzlichen Zeichen nicht ausreichen, um alle international benötigten Sonderzeichen zu kodieren, existieren verschiedene ASCII-Erweiterungen für die verschiedenen Sprachen und Regionen.

Die Erweiterungen sind mit dem ursprünglichen US-ASCII kompatibel. Alle im US-ASCII definierten Zeichen werden in den verschiedenen Erweiterungen durch die gleichen Bitfolgen kodiert. Die ersten 128 Zeichen einer ASCII-Erweiterung sind also mit der ursprünglichen ASCII-Tabelle identisch. Die Erweiterungen wie zum Beispiel ISO Latin 9 (ISO 8859-15) enthalten sprachspezifische Zeichen (zum Beispiel Umlaute) und Sonderzeichen (zum Beispiel das Euro-Symbol €), die nicht im lateinischen Grundalphabet enthalten sind.

Ein Nachteil der ASCII-Erweiterungen ist, dass nicht in allen Betriebssystemen alle Erweiterungen verfügbar sind. Wenn zwei Kommunikationspartner nicht die identische Erweiterung verwenden, werden unter anderem die Sonderzeichen im Text falsch angezeigt.

## 2.4.2 Unicode

Um die Probleme durch unterschiedliche Zeichenkodierungen in Zukunft zu vermeiden, wurde die Mehrbyte-Kodierung Unicode (ISO 10646) entwickelt. Diese wird laufend erweitert und soll in Zukunft alle bekannten Schriftzeichen enthalten.

Es existieren verschiedene Unicode-Standards. Am populärsten ist UTF-8. Die ersten 128 Zeichen werden mit einem Byte codiert und sind mit US-ASCII identisch. Die Kodierungen aller anderen Zeichen verwenden zwischen 2 und 6 Bytes. Aktuell enthält UTF-8 über 100.000 Zeichen.

Bei UTF-8 entspricht jedes mit 0 beginnende Byte einem 7-Bit US-ASCII-Zeichen. Jedes mit 1 beginnende Byte gehört zu einem aus mehreren Bytes kodierten Zeichen. Besteht ein mit UTF-8 kodierte Zeichen aus  $n \geq 2$  Bytes, beginnt das erste Byte mit  $n$  Einsen und jedes der  $n - 1$  folgenden Bytes mit der Bitfolge 10 (siehe Tab. 2.6).