

---

# Inhalt

<b>Vorwort</b> .....	<b>XV</b>
<b>Glossar</b> .....	<b>XXIII</b>
<b>1 Einführung</b> .....	<b>1</b>
Was ist Bitcoin? .....	1
Geschichte des Bitcoins .....	4
Bitcoin: Anwendungsfälle, Anwender und deren Geschichten .....	5
Erste Schritte .....	6
Wahl einer Bitcoin-Wallet .....	7
Schnelleinstieg .....	9
Ihr erster Bitcoin .....	11
Den aktuellen Bitcoin-Preis ermitteln .....	12
Bitcoin senden und empfangen .....	13
<b>2 Wie Bitcoin funktioniert</b> .....	<b>15</b>
Transaktionen, Blöcke, Mining und die Blockchain .....	15
Bitcoin-Übersicht .....	15
Eine Tasse Kaffee kaufen .....	16
Bitcoin-Transaktionen .....	18
Inputs und Outputs von Transaktionen .....	18
Transaktionsketten .....	19
Wechselgeld .....	20
Gängige Transaktionsformen .....	21
Eine Transaktion konstruieren .....	22
Die richtigen Inputs .....	23
Die Outputs erzeugen .....	24
Die Transaktion zum Kassenbuch hinzufügen .....	25
Bitcoin-Mining .....	26
Transaktionen in Blöcke einfügen .....	28
Die Transaktion einlösen .....	30

Andreas M. Antonopoulos, Bitcoin & Blockchain - Grundlagen und Programmierung, O'Reilly, ISBN 978-3-9

<b>3</b>	<b>Bitcoin Core: die Referenzimplementierung</b> .....	<b>33</b>
	Bitcoin-Entwicklungsumgebung .....	34
	Bitcoin Core aus dem Quellcode kompilieren .....	34
	Wahl einer Bitcoin-Core-Release .....	35
	Den Bitcoin-Core-Build konfigurieren .....	36
	Die Bitcoin-Core-Executables erzeugen .....	38
	Einen Bitcoin-Core-Knoten ausführen .....	39
	Bitcoin Core zum ersten Mal ausführen .....	41
	Den Bitcoin-Core-Knoten konfigurieren .....	41
	Bitcoin Core Application Programming Interface (API) .....	45
	Informationen zum Status des Bitcoin-Core-Clients abrufen .....	46
	Transaktionen untersuchen und decodieren .....	47
	Blöcke untersuchen .....	49
	Die Bitcoin Core API nutzen .....	50
	Alternative Clients, Bibliotheken und Toolkits .....	53
	C/C++ .....	53
	JavaScript .....	54
	Java .....	54
	Python .....	54
	Ruby .....	54
	Go .....	54
	Rust .....	54
	C# .....	55
	Objective-C .....	55
<b>4</b>	<b>Schlüssel und Adressen</b> .....	<b>57</b>
	Einführung .....	57
	Public-Key-Kryptografie und Kryptowährungen .....	58
	Private und öffentliche Schlüssel .....	59
	Private Schlüssel .....	60
	Öffentliche Schlüssel .....	62
	Kryptografie mit elliptischen Kurven .....	63
	Einen öffentlichen Schlüssel generieren .....	65
	Bitcoin-Adressen .....	67
	Base58- und Base58Check-Codierung .....	69
	Schlüsselformate .....	73
	Schlüssel und Adressen in Python implementieren .....	80
	Fortgeschrittene Schlüssel und Adressen .....	83
	Verschlüsselte private Adressen (Encrypted Private Keys, BIP-38) ..	83
	Pay-to-Script-Hash-(P2SH-)Adressen und Multisig-Adressen .....	84

Vanity-Adressen . . . . .	86
Paper-Wallets . . . . .	91
<b>5 Wallets . . . . .</b>	<b>95</b>
Wallet-Technologie in der Übersicht. . . . .	95
Nichtdeterministische (zufallsbasierte) Wallets . . . . .	96
Deterministische (Seed-basierte) Wallets . . . . .	97
HD-Wallets (BIP-32/BIP-44) . . . . .	98
Seeds und mnemonische Codes (BIP-39) . . . . .	99
Die Wallet-Best-Practices . . . . .	99
Eine Bitcoin-Wallet verwenden . . . . .	100
Details der Wallet-Technologie. . . . .	101
Mnemonische Codewörter (BIP-39). . . . .	102
Eine HD-Wallet aus dem Seed-Wert erzeugen. . . . .	108
Einen erweiterten öffentlichen Schlüssel in einem Webshop nutzen . . . . .	113
<b>6 Transaktionen . . . . .</b>	<b>119</b>
Einführung. . . . .	119
Transaktionen im Detail . . . . .	119
Transaktionen – hinter den Kulissen . . . . .	120
Transaktions-Outputs und -Inputs . . . . .	121
Transaktions-Outputs. . . . .	123
Transaktions-Inputs . . . . .	125
Transaktionsgebühren (Fees) . . . . .	128
Gebühren in Transaktionen einfügen . . . . .	131
Transaktionsskripte und Skriptsprache. . . . .	132
Turing-Unvollständigkeit . . . . .	133
Zustandslose Verifikation . . . . .	134
Konstruktion von Skripten (Lock + Unlock) . . . . .	134
Pay-to-Public-Key-Hash (P2PKH). . . . .	138
Digitale Signaturen (ECDSA) . . . . .	140
Wie digitale Signaturen funktionieren . . . . .	141
Die Signatur verifizieren . . . . .	143
Arten von Signatur-Hashes (SIGHASH) . . . . .	143
Die Mathematik hinter ECDSA . . . . .	145
Die Bedeutung der Zufälligkeit für Signaturen. . . . .	147
Bitcoin-Adressen, Guthaben und andere Abstraktionen . . . . .	147
<b>7 Transaktionen und Skripting für Fortgeschrittene . . . . .</b>	<b>151</b>
Einführung. . . . .	151
Multisignatur . . . . .	151

Pay-to-Script-Hash (P2SH) . . . . .	153
P2SH-Adressen. . . . .	155
Vorteile von P2SH . . . . .	156
Redeem-Skript und Validierung. . . . .	156
Data Recording Output (RETURN) . . . . .	157
Timelocks . . . . .	159
Transaktions-Locktime (nLocktime) . . . . .	159
Check Lock Time Verify (CLTV) . . . . .	160
Relative Timelocks. . . . .	162
Relative Timelocks mit nSequence. . . . .	163
Relative Timelocks mit CSV. . . . .	164
Median-Time-Past . . . . .	165
Timelock-Schutz gegen Fee-Sniping . . . . .	166
Skripte mit Ablaufsteuerung (Bedingungsklauseln) . . . . .	166
Bedingungsklauseln mit VERIFY-Opcodes . . . . .	167
Die Ablaufsteuerung in Skripten nutzen . . . . .	168
Komplexes Skriptbeispiel . . . . .	170
<b>8 Das Bitcoin-Netzwerk . . . . .</b>	<b>173</b>
Peer-to-Peer-Netzwerkarchitektur . . . . .	173
Arten und Rollen von Nodes . . . . .	174
Das erweiterte Bitcoin-Netzwerk . . . . .	175
Bitcoin-Relay-Netzwerke . . . . .	178
Netzwerkerkundung. . . . .	178
Full Nodes. . . . .	182
»Inventar« austauschen. . . . .	183
SPV-Nodes (Simplified Payment Verification) . . . . .	184
Bloomfilter . . . . .	187
Wie Bloomfilter funktionieren. . . . .	188
Wie SPV-Nodes Bloomfilter nutzen . . . . .	192
SPV-Nodes und Privatsphäre . . . . .	193
Verschlüsselte und authentifizierte Verbindungen . . . . .	193
Tor-Transport . . . . .	193
Peer-to-Peer-Authentifizierung und -Verschlüsselung . . . . .	194
Transaktionspools . . . . .	195
<b>9 Die Blockchain. . . . .</b>	<b>197</b>
Einführung . . . . .	197
Struktur eines Blocks . . . . .	198
Block-Header . . . . .	199
Blockkennungen: Block-Header und Blockhöhe . . . . .	199

Der Genesis-Block . . . . .	200
Blöcke in der Blockchain verlinken . . . . .	202
Merkle Trees (Hashbäume) . . . . .	202
Merkle Trees und Simplified Payment Verification (SPV) . . . . .	208
Bitcoins Test-Blockchains . . . . .	209
Testnet – Bitcoins Testspielwiese . . . . .	209
Segnet – das Segregated-Witness-Testnet . . . . .	211
Regtest – die lokale Blockchain . . . . .	211
Test-Blockchains für die Entwicklung nutzen . . . . .	212
<b>10 Mining und Konsens . . . . .</b>	<b>215</b>
Einführung . . . . .	215
Bitcoin-Ökonomie und Währungsgenerierung . . . . .	217
Dezentralisierter Konsens . . . . .	219
Unabhängige Verifikation von Transaktionen . . . . .	220
Mining-Nodes . . . . .	222
Transaktionen in Blöcken zusammenfassen . . . . .	222
Die Coinbase-Transaktion . . . . .	224
Coinbase-Belohnungen und Gebühren . . . . .	225
Struktur der Coinbase-Transaktion . . . . .	226
Coinbase-Daten . . . . .	227
Die Block-Header aufbauen . . . . .	229
Mining des Blocks . . . . .	230
Proof-of-Work-Algorithmus . . . . .	231
Target-Darstellung . . . . .	237
Retargeting zur Anpassung der Difficulty . . . . .	238
Den Block erfolgreich schürfen . . . . .	240
Einen neuen Block validieren . . . . .	240
Ketten von Blöcken zusammensetzen und auswählen . . . . .	241
Blockchain-Forks . . . . .	243
Mining und der Hashing-Wettlauf . . . . .	250
Die Lösung mit der Extra-Nonce . . . . .	252
Mining-Pools . . . . .	253
Konsensangriffe . . . . .	256
Die Konsensregeln ändern . . . . .	260
Hard Forks . . . . .	260
Hard Forks: Software, Netzwerk, Mining und die Chain . . . . .	261
Divergierende Miner und Difficulty . . . . .	263
Umstrittene Hard Forks . . . . .	264
Soft Forks . . . . .	264
Kritik an Soft Forks . . . . .	266

Soft-Fork-Signalisierung mittels Blockversion . . . . .	266
BIP-34-Signalisierung und -Aktivierung. . . . .	267
BIP-9-Signalisierung und -Aktivierung. . . . .	268
Entwicklung von Konsenssoftware. . . . .	270
<b>11 Bitcoins und Sicherheit . . . . .</b>	<b>273</b>
Sicherheitsgrundsätze . . . . .	273
Bitcoin-Systeme sicher entwickeln. . . . .	274
Die Wurzel des Vertrauens . . . . .	275
Best Practices für den Nutzer . . . . .	276
Physische Speicherung von Bitcoins . . . . .	277
Hardware-Wallets . . . . .	277
Risiken abwägen . . . . .	278
Risiken verteilen. . . . .	278
Multisignaturen und Kontrolle . . . . .	278
Überlebensfähigkeit . . . . .	278
Fazit . . . . .	279
<b>12 Blockchain-Anwendungen . . . . .</b>	<b>281</b>
Einführung . . . . .	281
Grundbausteine (Primitive) . . . . .	282
Anwendungen aus Grundbausteinen . . . . .	284
Colored Coins . . . . .	284
Colored Coins nutzen . . . . .	285
Colored Coins ausstellen . . . . .	286
Colored-Coins-Transaktionen . . . . .	286
Counterparty. . . . .	289
Zahlungs- und Zustandskanäle. . . . .	290
Zustandskanäle – grundlegende Konzepte und Terminologie. . . . .	291
Einfaches Zahlungskanalbeispiel . . . . .	293
Vertrauensfreie Kanäle aufbauen . . . . .	296
Asymmetrisch widerrufliche Commitments . . . . .	299
Hash Time Lock Contracts (HTLC) . . . . .	303
Geroutete Zahlungskanäle (Lightning Network) . . . . .	304
Einfaches Lightning-Network-Beispiel . . . . .	305
Lightning Network – Transport und Routing . . . . .	308
Vorteile des Lightning Network. . . . .	310
Fazit . . . . .	311