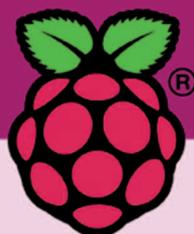




Andrew
Robinson

Mike
Cook



Spannende Projekte mit dem **Raspberry Pi**®

470 Seiten Raspberry-Pi-Praxis



Mit Beiträgen
von

Jonathan
Evans

Sean
McManus

Hinweis des Verlages zum Urheberrecht und Digitalen Rechtemanagement (DRM)

Der Verlag räumt Ihnen mit dem Kauf des ebooks das Recht ein, die Inhalte im Rahmen des geltenden Urheberrechts zu nutzen. Dieses Werk, einschließlich aller seiner Teile, ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und Einspeicherung und Verarbeitung in elektronischen Systemen.

Der Verlag schützt seine ebooks vor Missbrauch des Urheberrechts durch ein digitales Rechtemanagement. Bei Kauf im Webshop des Verlages werden die ebooks mit einem nicht sichtbaren digitalen Wasserzeichen individuell pro Nutzer signiert.

Bei Kauf in anderen ebook-Webshops erfolgt die Signatur durch die Shopbetreiber. Angaben zu diesem DRM finden Sie auf den Seiten der jeweiligen Anbieter.

Dr. Andrew Robinson, Mike Cook, Jonathan Evans, Sean McManus

Spannende Projekte mit dem Raspberry Pi[®]

Übersetzung aus dem Englischen
von Knut Lorenzen



Bibliografische Information der Deutschen Nationalbibliothek

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

ISBN 978-3-8266-9909-2
1. Auflage 2014

www.mitp.de
E-Mail: kundenservice@hjr-verlag.de
Telefon: +49 6221 / 489 -555
Telefax: +49 6221 / 489 -410

Übersetzung der englischen Originalausgabe:
Andrew Robinson, Mike Cook, Jonathan Evans, Sean McManus:
Raspberry Pi® Projects, ISBN: 978-1-118-55543-9.
Original English Edition Copyright © 2014 John Wiley & Sons, Ltd.
Published by John Wiley & Sons Ltd., The Atrium, Southern Gate,
Chichester, West Sussex, PO19 8 SQ United Kingdom.

All rights reserved. Authorised translation from the English language edition published by John Wiley & Sons Limited. Responsibility for the accuracy of the translation rests solely with Verlagsgruppe Hüthig Jehle Rehm GmbH and is not the responsibility of John Wiley & Sons Limited. No part of this book may be reproduced in any form without the written permission of the original copyright holder, John Wiley & Sons Limited.

© 2014 mitp, eine Marke der Verlagsgruppe Hüthig Jehle Rehm GmbH
Heidelberg, München, Landsberg, Frechen, Hamburg

Dieses Werk, einschließlich aller seiner Teile, ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen, 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.

Lektorat: Sabine Schulz
Sprachkorrektur: Petra Heubach-Erdmann
Satz: III-satz, Husby, www.drei-satz.de

Inhaltsverzeichnis

	Über die Autoren	13
	Einleitung	15
Teil I	Erste Schritte mit dem Raspberry Pi	23
1	Den Raspberry Pi zum Laufen bringen	25
1.1	Das Betriebssystem	25
1.1.1	Das Betriebssystem auf eine SD-Karte schreiben	26
1.2	Den Raspberry Pi anschließen	32
1.2.1	Anschluss eines Bildschirms	34
1.2.2	Anschluss ans Netzwerk	37
1.2.3	Start des Betriebssystems	37
1.2.4	Einschalten!	38
1.3	Der Startvorgang	39
1.3.1	Sprache, Ländereinstellungen und Tastaturbelegung anpassen	40
1.4	Die grafische Benutzeroberfläche starten	41
1.5	Terminal unter X starten	42
1.6	Fehlerbehebung	42
1.6.1	Häufige Fehlerquellen	43
1.6.2	Weitere Hilfestellung	44
1.7	Der Spaß geht los!	44
2	Ein erstes Programm: Der Beschimpfungsgenerator	45
2.1	Das erste Python-Programm	46
2.2	Das Programm speichern	48
2.3	Beschimpfung generieren	51
2.3.1	Variablen	51
2.3.2	Strings	52
2.3.3	Listen	52
2.3.4	Funktionen	52

2.4	Beschimpfung mit Anrede	54
2.4.1	Bedingtes Verhalten	56
2.5	Ein ganzer Schwall an Beschimpfungen	56
2.5.1	Eigene Funktionen erstellen	56
2.5.2	Schleifen	58
2.6	Fazit	60

Teil II Software-Projekte 61

3	Tic-Tac-Toe	63
3.1	Fehler	64
3.2	Zum Auftakt das Spielbrett	64
3.2.1	Verbesserung der Spielbrettausgabe	66
3.2.2	Gibt es einen Gewinner?	66
3.2.3	Hilfsfunktionen	68
3.3	Ein Spiel für zwei Teilnehmer	68
3.4	Der Computer als Spieler	72
3.4.1	Der Computer als Fünfjähriger	72
3.4.2	Einen siegreichen Spielzug erkennen	74
3.4.3	Eine Strategie hinzufügen	77
3.4.4	Strategieverfeinerung	79
3.5	Sie sind am Zug!	83
4	Hier sind die Nachrichten	85
4.1	Die ersten Teleprompter	85
4.2	Der Pi-Prompter	86
4.3	Erforderliche Schritte	87
4.4	Ein weiterer Schritt zu einem brauchbaren Programm	92
4.5	Das fertige Programm	97
4.6	Mechanischer Aufbau	102
4.7	Sie sind am Zug!	105
5	Ping	107
5.1	Erste kommerzielle Produkte	107
5.2	Das Spiel	108
5.2.1	Bewegung auf dem Bildschirm	108
5.2.2	Kollisionen feststellen	110
5.2.3	Aufpralltest	111
5.3	Das Spiel verfeinern	113

5.4	Version für Einzelspieler	117
5.5	Version für zwei Spieler	125
5.6	Sie sind am Zug!	132
6	Pie-Man: Ein Pac-Man-Klon	133
6.1	Das Spiel.	134
6.2	Ressourcen zusammentragen	135
	6.2.1 Die Soundeffekte	135
	6.2.2 Die Spielfiguren	136
6.3	Bühne frei!	139
6.4	Das Spielgeschehen.	144
	6.4.1 Pillen verspeisen.	147
	6.4.2 Das Ende des Pie-Mans – oder eines Geistes.	150
	6.4.3 Die jagende Meute	152
6.5	Bildschirmdarstellung.	153
6.6	Die main()-Funktion	156
	6.6.1 Das Spiel beginnt	161
	6.6.2 Geisterjagd	161
	6.6.3 Spielende.	162
6.7	Sie sind am Zug!	162
7	Minecraft Maze Maker	165
7.1	Minecraft installieren	166
7.2	Start des Spiels.	167
7.3	Spielablauf	167
	7.3.1 Umherstreifen	168
	7.3.2 Objekte erzeugen und zerstören	168
7.4	Vorbereitungen für den Einsatz von Python	169
7.5	Verwendung des Minecraft-Moduls	170
	7.5.1 Koordinaten in Minecraft.	171
	7.5.2 Neupositionierung des Spielers	171
	7.5.3 Bausteine hinzufügen	172
	7.5.4 Änderungen durch den Spieler verhindern	174
	7.5.5 Die Parameter des Labyrinths	174
	7.5.6 Das Fundament legen	176
	7.5.7 Aufbau der Wände	177
	7.5.8 Der Labyrinth-Algorithmus	178
	7.5.9 Variablen und Listen vorbereiten	178
	7.5.10 Funktionen erstellen	179

7.5.II	Die Hauptschleife	181
7.5.I2	Eine Decke hinzufügen	183
7.5.I3	Die Spielfigur positionieren.	183
7.5.I4	Der endgültige Code.	184
7.6	Sie sind am Zug!	187

Teil III Hardware-Projekte 189

8	Schnappen	191
8.1	Das Spiel implementieren.	192
	8.1.1 Die Theorie	192
	8.1.2 Das Gehäuse herstellen	200
8.2	Die Software zum Testen des Geräts	205
8.3	Die Spielsoftware	208
8.4	Sie sind am Zug!	214
9	Reaktionstest	215
9.1	Willkommen in einer anderen Computerwelt!	215
	9.1.1 Bewährte Verfahrensweisen	216
9.2	Beschaffung von Bauteilen	217
	9.2.1 Eine Schnittstellen-Platine	217
	9.2.2 PiFace-Digital	217
9.3	PiFace-Digital einrichten.	218
	9.3.1 Installation der PiFace-Digital-Software	219
9.4	PiFace-Digital anschließen	222
9.5	Den Emulator verwenden	222
9.6	Die Schnittstelle zu Python.	223
	9.6.1 LED einschalten	223
	9.6.2 Schaltzustand erkennen.	225
9.7	Der Reaktionstest.	226
	9.7.1 Die Anschlüsse des PiFace-Digital	227
	9.7.2 Einen Taster und eine LED anschließen.	231
	9.7.3 Mit dem Reaktionstester spielen	236
9.8	Schnittstellen	237
10	Gezwitscher	239
10.1	Das Spielzeug hacken	239
	10.1.1 Verdrahtung	241
10.2	Das Huhn zum Sprechen bringen	244

10.2.1	espeak in Python verwenden	245
10.2.2	Test des espeak-Moduls	246
10.3	Bewegungssteuerung	246
10.3.1	Klassen erstellen	248
10.3.2	Objekte erzeugen	249
10.3.3	Ständiges Testen	250
10.4	Verbindungsaufnahme mit Twitter	250
10.4.1	Das Python-Modul kompilieren und installieren	251
10.4.2	Kommunikation mit Twitter	253
10.5	Fertigstellung	256
10.6	Fazit	258
10.6.1	Weitere Ideen	259
II	Disco-Beleuchtung	261
II.1	Die Reihenfolge festlegen	262
II.2	Den Code erweitern	263
II.3	Ein kleiner theoretischer Exkurs	266
II.4	Entwurf des Sequenzers	267
II.5	Implementierung des Sequenzers	267
II.6	Lichterketten	275
II.7	Längere Lichterketten	278
II.8	Musik als Taktgeber	279
II.9	Schaltungsentwurf	280
II.10	Schaltungsaufbau	282
II.11	Anschluss der Schaltung	283
II.12	Sie sind am Zug!	283
12	Türschloss	285
12.1	Überblick	286
12.2	Sicherheitskritische Systeme	286
12.3	Die Türschloss-Hardware	287
12.4	Software-Simulation	288
12.5	Ausgabe	290
12.5.1	Anschluss der Schaltung	290
12.5.2	Programmierung der Türsteuerung	292
12.6	Eingabe	293
12.6.1	Benutzereingabe abfragen	293
12.7	Authentifizierung	293
12.7.1	Vertrauliche Daten speichern	294

12.8	Berührungsloses Türöffnen	295
12.8.1	Anschluss des RFID-Lesegeräts.....	296
12.8.2	Den RFID-Leser in Python verwenden.....	299
12.8.3	Die Programmteile zusammenfügen	301
12.9	Programmtest und Schlosseinbau	301
12.10	Mehrere Türen vernetzen	303
12.11	Sie sind am Zug!	304
12.12	Die Kunst der Programmierung.....	304
13	Hausautomation	307
13.1	Das Internet der Dinge	307
13.2	Projekt 1: Bewegungsmelder und Türkontaktschalter	308
13.2.1	Schaltungsaufbau	310
13.2.2	Software.....	313
13.3	Projekt 2: Überwachung per Webcam	314
13.3.1	Schaltungsaufbau	314
13.3.2	Software.....	315
13.4	Projekt 3: Temperaturanzeige.....	320
13.4.1	Schaltungsaufbau	320
13.4.2	Software.....	321
13.5	Projekt 4: Warnmeldungen via E-Mail versenden.....	325
13.5.1	Anforderungen	325
13.5.2	Software.....	325
13.6	Projekt 5: E-Mail per Funkfernsteuerung versenden	330
13.6.1	Schaltungsaufbau	331
13.6.2	Software.....	333
13.7	Sie sind am Zug!	337
14	Computergesteuertes Modellbahnrennen	339
14.1	Beschaffung einer Modellrennbahn.....	339
14.2	Umbau der Rennbahn.....	340
14.2.1	Test der Schaltung	341
14.3	Spielereingaben	342
14.3.1	Taster.....	343
14.3.2	Gehäuse.....	347
14.3.3	Joystick-Test	350
14.4	Die Software.....	351
14.5	Das Spiel.....	353
14.6	Sie sind am Zug!	360

15	Drehgeber mit Facebook-Anbindung	361
15.1	Das Konzept	361
15.2	Drehgebertypen	362
15.3	Ausgabe des Drehgebers	362
15.4	Automatisches Hinaufladen	370
15.4.1	Flickr	371
15.5	Das fertige Programm	374
15.6	Symmetrische Muster erzeugen	381
15.7	Sie sind am Zug!	387
16	Das Pi-Pendel – ein Harmonograph	389
16.1	Das Konzept	390
16.2	Der Hall-Effekt	390
16.3	Kurz vorgestellt: Der Arduino	392
16.4	Zusammenbau	393
16.4.1	Sanfte Schwingungen	398
16.4.2	Elektronik	399
16.5	Programmierung des Arduino	403
16.5.1	Der fertige Arduino-Code	409
16.6	Programmierung des Raspberry Pi	417
16.7	Das Pi-Pendel in Aktion	422
16.8	Sie sind am Zug!	424
17	Das Hightech-Vogelhäuschen – Beobachtung der Tierwelt	425
17.1	Bau einer Lichtschranke	427
17.1.1	Erforderliche Bauteile	427
17.1.2	Anschluss des Senders	427
17.1.3	Anschluss des Empfängers	428
17.1.4	Test der Sensoren	430
17.2	Montage der Sensoren	430
17.2.1	Schutz vor Wind und Wetter	432
17.3	Aktivitäten aufzeichnen	433
17.3.1	Verarbeitung in Echtzeit oder im Nachhinein?	433
17.3.2	Das Programm zum Aufzeichnen	435
17.3.3	Das Programm testen	444
17.4	Die Daten verarbeiten	445
17.4.1	Endliche Automaten	446
17.4.2	Ein einfaches Programm zur Auswertung	447
17.5	Störsignale handhaben	451

17.5.1	Störsignale herausfiltern	451
17.5.2	Auswertungsprogramm mit Störsignalfilterung	452
17.6	Ein Diagramm erstellen	457
17.7	Das Vogelhäuschen in Betrieb nehmen.	460
17.8	Sie sind am Zug!	460
17.8.1	Teilen Sie Ihre Daten	461
17.8.2	Weitere Sensoren hinzufügen.	461
17.9	Unbegrenzte Möglichkeiten	461
	Stichwortverzeichnis	463

Den Erfindern des Küchentischs und ihren seit Langem leidenden Familien, die damit leben müssen.

Andrew Robinson

Für Mike Bibby, dem ersten Redakteur, der mir die Möglichkeit gab, regelmäßig über Computer und Hardware zu schreiben. Sein unermüdlicher Enthusiasmus allem gegenüber und sein Unvermögen, etwas als gegeben hinzunehmen, sind uns allen ein Beispiel.

Mike Cook

Über die Autoren

Andrew Robinson ist Gründer einer erfolgreichen Beratungsfirma für eingebettete Computersysteme mit Sitz in Manchester. Bildung und Lehre liegen ihm sehr am Herzen und er veranstaltet Seminare und Schulungen für Einsteiger und Fortgeschrittene, angefangen bei Schulkindern bis hin zu Konstruktionsingenieuren. Seine Projekte mit dem Raspberry Pi sind durch die überregionale Presse und das Fernsehen (ITV, Channel 5, BBC) bekannt geworden. Er ist ehrenamtlicher wissenschaftlicher Mitarbeiter der Universität Manchester, an der er mit einer Arbeit über Niedrigenergieprozessoren auch den Dokortitel erwarb.

Seit Andrew im Alter von fünf Jahren einen funktionierenden Modellbauleuchtturm gebastelt hat, ist er von Elektronik und Computern begeistert.

Mike Cook ist ein erfahrener Autor von Technikbüchern und Elektronikbastler aus Großbritannien. Er ist in Manchester aufgewachsen und lebt noch immer dort in der Nähe. Einer breiten Öffentlichkeit ist er durch seine mehr als 300 Artikel bekannt, die von 1983 bis 2000 in den Publikationen *The Micro User*, *Acorn Computing* und *Acorn User* erschienen sind. In diesen Artikeln wurden unter den Überschriften »Body Building Course« und »Run the Risc« der Entwurf und Bau von technischem Spielzeug, Schnittstellen und Peripheriegeräten für den damaligen BBC-Computer und den RISC PC vorgestellt. Er hat für diese Computerzeitschriften außerdem zahlreiche Testberichte und Programmvorstellungen verfasst sowie Leserfragen beantwortet.

Mike begann in den späten 60er Jahren bei einem Elektronikunternehmen in Oldham zu arbeiten und erwarb in Newcastle einen Abschluss in Elektrophysik, wobei er zwischenzeitlich auch ein Jahr an der seebehördlichen Einrichtung für Unterseewaffentechnik in Portland verbrachte. Danach beschäftigte er sich an der Universität Salford mit der Kompression von Audiodaten. Er hat mehr als 20 Jahre lang an der städtischen Universität von Manchester (ehemals Polytechnikum Manchester) Vorlesungen über Physik gehalten und sich dabei auf Computermesstechnik, Astronomie und Bildverarbeitung spezialisiert. Später wendete er sich wieder der Industrie zu, leitete ein Entwicklerteam für die Hardware einer zukunftsweisenden Set-Top-Box für den Empfang digitaler Fernsehprogramme und arbeitete als Entwicklungsleiter für Sicherheitstechnik und RFID-Produkte.

Mittlerweile arbeitet er freiberuflich als Autor und Berater für eingebettete elektronische Systeme. Sein neuestes Buch trägt den Titel *The Raspberry Pi For Dummies* und ist bei Wiley erschienen.

Seit Kurzem beschäftigt er sich mit der Kombination von Hard- und Software im Rahmen des *Physical Computing* (wobei Aktoren, Sensoren, Motoren, LEDs etc. einbezogen werden) und hat bei verschiedenen »Maker Fair«-Veranstaltungen teilgenommen, auf denen Bastler ihre Arbeiten präsentieren. Auf der Maker Faire 2013 in Rom wurde er mit einem »Blauen Band« ausgezeichnet.

Jonathan Evans hat sich schon immer für Computer und Elektronik interessiert. Im zarten Alter von 10 Jahren hat er sich selbst das Programmieren eines Computers beigebracht und schnell bemerkt, dass sich Computer und Elektronik miteinander kombinieren lassen, um seine Geschwister aus seinem Zimmer fernzuhalten. Er ist inzwischen zu einem hervorragenden IT-Experten mit mehr als 20 Jahren Erfahrung gereift. Das Phänomen Raspberry Pi und seine Leidenschaft für Erfindungen und Innovationen passen perfekt zueinander. In seiner Freizeit vergnügt er sich mit Projekten, die den Raspberry Pi für den alltäglichen Gebrauch einsetzen. Unter <http://www.projects.privateeyepi.com> lässt er Sie an seinen Ideen teilhaben und untersucht weiterhin die unerschöpflichen Möglichkeiten dieser Computerplattform.

Sean McManus verfasst anregende Bücher und Artikel zu Computerthemen. Von ihm stammt das Kapitel über Minecraft und er ist Autor der Bücher *Raspberry Pi For Dummies* (gemeinsam mit Mike Cook), *Scratch Programming in Easy Steps*, *iPad for the Older and Wiser*, *Microsoft Office for the Older and Wiser* und *Web Design in Easy Steps*. Besuchen Sie seine Website unter <http://www.sean.co.uk>.

Einleitung

von Dr. Andrew Robinson

Für Kinder ist es ganz selbstverständlich, Neues zu erschaffen. Sie malen Bilder, denken sich neue Spiele aus, erzählen sich Geschichten und basteln aus einer Spülmittelflasche eine Rakete, von der sie überzeugt sind, dass sie bis zum Mond fliegt. Die Kindheit ist zugleich Abenteuer und Entdeckungsreise – ein ständiges Streben nach Neuem.

Wenn man älter wird und das »wahre Leben« beginnt, bereiten solche Dinge zwar weiterhin Freude, es wird aber schwieriger, die Zeit zu finden, um ihnen Raum zu geben. Dennoch sind einige der bedeutendsten Erfindungen und Entdeckungen der Geschichte durch Leute zustande gekommen, die sich nicht scheuten, einfach etwas auszuprobieren – und das oftmals in den eigenen vier Wänden oder im Gartenhäuschen, nicht in technisch bestens ausgestatteten Ingenieurbüros.

Was das alles in einem Buch über Projekte mit dem Raspberry Pi zu suchen hat?

Nun, wenn Sie weiterlesen und versuchen, einige der Projekte nachzuvollziehen, werden Sie vielleicht entdecken, welche Freude es bereitet, mithilfe des Computers Dinge zum Laufen zu bringen. Der Computer erschließt der Abenteuerlust und der Kreativität eine fantastische neue Welt ungeahnter Möglichkeiten. Er ist in so viele Lebensbereiche vorgedrungen (auch Spielkonsolen, Set-Top-Boxen und Smartphones sind Computer!), dass Sie ihn für fast jedes Hobby oder Steckenpferd einsetzen können.

Sie werden sehen, dass Computerkenntnisse jedem nutzen und dass ein kreativer Augenblick am Küchentisch große Auswirkungen nach sich ziehen kann. Und nebenbei erfahren Sie außerdem die Geschichte, die hinter einem bestimmten kreditkartengroßen Computer steckt.

Die Geschichte des Bastelns

So manche weltbewegende Erfindung kam auf unkonventionelle Weise zustande.

Die Brüder Orville und Wilbur Wright waren einfache Leute aus Ohio, die ein Fahrradgeschäft betrieben. Von der Funktionsweise dieser simplen Maschinen fasziniert, gelangten sie zu der Überzeugung, dass sie eine Flugmaschine bauen könnten. Und das taten sie dann auch. Im Jahre 1902 hoben sie erstmals in der Geschichte mit einem Flugzeug vom Boden ab. Fast ein Jahrhundert später, als Afrika von AIDS heimgesucht wurde, kam Trevor Baylis, der früher als Stuntman arbeitete, zu der Erkenntnis, dass er helfen könne. In einem Gartenhäuschen am Stadtrand entwickelte er ein preisgünstiges und robustes aufziehbares Funkgerät, um das Gesundheitswesen betreffende Nachrichten in Afrika zu ver-

breiten. Er hat damit ohne Zweifel viele Leben gerettet. Steve Jobs und Steve Wozniak, die Gründer von Apple, erwarben ihre Computer- und Elektronikkenntnisse durch Experimente in ihren Schlafzimmern und einer Garage. Das sind nur drei Beispiele dafür, wie das heimische Basteln zu weltweiten Auswirkungen für Millionen von Menschen führen kann.

Viele Erfinder haben zwar eine gewisse Vorstellung davon, was sie erschaffen möchten, wissen jedoch nicht, wie sie es praktisch umsetzen können. Angespornt durch die Freude an der Kreativität, bringen sie sich selbst die erforderlichen Fähigkeiten bei, um ihr Vorhaben zu realisieren. Wozniak und Jobs haben das getan, indem sie vorhandene Geräte auseinandernahmen, herausfanden, wie sie funktionierten, und dann wieder zusammenbauten. In einigen Fällen führte dieses Herumbasteln tatsächlich zur Verbesserung eines Geräts – gelegentlich funktionierte es auch überhaupt nicht mehr. Das hielt sie aber nicht vom Weitermachen ab. Manchmal geht es eben nur darum, herauszufinden, wie etwas funktioniert, oder technische Widrigkeiten zu überwinden, und nicht darum, gleich ein gelacktes Produkt anzufertigen.

Computer für den Endverbraucher

Dass ausgerechnet das Herumstochern im Innenleben von technischen Geräten zur Geburtsstunde von Apple-Computern führte, ist wohl die Ironie des Schicksals. Heutzutage werden Computer in schnittigen, eleganten Aluminiumgehäusen verkauft, die es verhindern, herumzuxperimentieren und herauszufinden, wie sie eigentlich funktionieren. Im fortwährenden Bestreben, ihre Produkte wertvoller erscheinen zu lassen, locken die Hersteller ihre Kunden mit möglichst einfach einsetzbaren Produkten und einer mühelosen Benutzerschnittstelle, die ihnen jeden Wunsch von den Lippen abliest.

Dieser Wandel vollzieht sich leider etwas unsanft. Moderne Computersysteme tun nur selten genau das, was der Benutzer wünscht. Wie oft sitzt man frustriert vor einem unfähigen Computer und wird mit einem Hinweis des Herstellers vertröstet, der dann etwa »Wird mit der nächsten Aktualisierung behoben« oder »Um das zu tun, müssen Sie die neueste Version erwerben« lautet? Für technisch Versierte sind solche Hinweise fast schon ein Schlachtruf, der dazu auffordert, so lange am Computer herumzuxperimentieren, bis er ihrem Willen gehorcht. Doch sind dieser Tage die wenigsten Benutzer mutig oder erfahren genug, die Ärmel hochzukrempeln und sich selbst am Computer zu schaffen zu machen.

Computerkenntnisse für alle

Computer sind wirklich allgegenwärtig und durchdringen sämtliche Lebensbereiche. Denken Sie dabei neben Laptop, Desktop und Smartphone auch an die Computer, die lebenserhaltende medizinische Systeme, Online-Banking, Einkaufen im Internet, Produktionsprozesse und die Versorgungskette für Lebensmittel ermöglichen. Computer sind zur Kommunikation, zum Betrieb von digitalem Funk und Fernsehen, für Mobiltelefonnetze und das Internet unverzichtbar. Angesichts der Bedeutung, die Computer für das reibungslose Funktionieren von Medien, Wirtschaft und Verwaltung besitzen, ist es erstaunlich, wie viele Menschen keine Ahnung von der Arbeitsweise eines Computers haben.

Wenn man bedenkt, wie sehr man sich auf korrekt arbeitende Computer verlässt, sollte ein besseres Verständnis von großem Nutzen sein. Wirtschaftsführer und Politiker könnten bei Projekten, die sich um Computer drehen, angemessene Entscheidungen treffen, und Otto Normalverbraucher würde Online-Betrügereien oder verlogenen Werbebehauptungen nicht mehr zum Opfer fallen. Jedermann wäre in der Lage, sich dank Computerhilfe das Leben zu erleichtern.

Ich kann eine gewisse Ähnlichkeit zwischen der Computerei und dem Kochen entdecken. Beim Kochen gibt es Rezepte, denen schrittweise zu folgen ist. Es geht darum, aus mehreren Zutaten ein Gericht zuzubereiten. Sie müssen, um etwa einen Apfelkuchen zu backen, diese Aufgabe in handhabbare Schritte unterteilen (Teig anrühren, Äpfel schälen, Backzeit beachten), die schließlich zu einem (hoffentlich wohlschmeckenden) fertigen Apfelkuchen führen. Hierbei handelt es sich um ein einfaches Beispiel einer *Abstrahierung*, die der Schlüssel zur Lösung von Computerproblemen ist. Eine bei Computerarbeiten entwickelte Vorgehensweise wie diese Abstraktion, also durch logisches Denken Problemlösungen zu finden, ist auch in anderen Lebensbereichen hilfreich.

Wir lehren unseren Kindern das Kochen nicht, damit sie später einmal professionelle Köche werden, sondern weil es eine entscheidende Fähigkeit darstellt, ohne die man sein Leben lang dazu verdammt wäre, teure und unzufriedenstellende Fertiggerichte aufzuwärmen. Bei vielen Menschen führt das Erlernen dieser grundlegenden Fertigkeiten zu einer dauerhaften Vorliebe fürs Kochen. Sie können beim Kochen ihre Kreativität zeigen – vielleicht indem sie anfangs ein bekanntes Rezept übernehmen und dieses aufpeppen, um es sich zu eigen zu machen. Kochen ist eine soziale Beschäftigung, die es ermöglicht, am lebhaft bevölkerten Esstisch Erfolge vorzuzeigen sowie Verfahrensweisen und Schwierigkeiten zu diskutieren.

Ich möchte behaupten, dass das Erlernen des Umgangs mit Computern einige Parallelen mit dem Kochenlernen aufweist. Die grundlegenden Fertigkeiten benötigen alle. Manche Leute bauen diese Fähigkeiten aus und werden professionelle Programmierer, ich hoffe aber, dass die meisten Leute es als Chance begreifen, nicht nur ihre Kreativität auszuleben, sondern auch besser mit den heutigen modernen Computerumgebungen zurechtzukommen.

Angesichts der Tatsache, dass immer mehr Leute immer mehr über die Funktionsweise und die Verlässlichkeit von Computern wissen müssen, ist es ebenfalls paradox, dass der Einstieg in die Arbeit mit modernen Computern schwieriger geworden ist. Das heißt, bis ein gewisser kreditkartengroßer Computer die Bühne betrat ...

Der Raspberry Pi

Für die meisten Leute ist es eine beängstigende Vorstellung, mit einem 1.000 Euro teuren Laptop Experimente anzustellen, die womöglich auch noch wertvolle Daten in Gefahr bringen. Ich jedenfalls würde es mir zweimal überlegen, ob ich alle meine Digitalfotos, meine Musiksammlung und meine Online-Banking-Daten riskiere! Außerdem halten Spielkonsolen und manche Smartphones viele Leute aktiv davon ab, Spiele und Apps selbst zu entwickeln, womöglich um die Umsätze zu schützen und die Konsumenten zum Kauf industrieller Produkte zu bewegen.

Aus dem Wunsch heraus, den Computerspaß mit anderen zu teilen, und aufgrund des Bedarfs an computerverständigen Leuten entwarf Eben Upton am Küchentisch einen einfachen kleinen Computer. Dank der Hilfe von Dr. Rob Mullins, Professor Alan Mycroft und Jack Lang von der Universität Cambridge, dem Hardware-Experten Pete Lomas und David Braben wurde daraufhin die *Raspberry Pi Foundation* gegründet. Diese Einrichtung hat es sich zum Ziel gesetzt, die Lehre von Informatik und verwandten Themen insbesondere an Schulen zu fördern und dass die Beschäftigung mit Computern wieder Spaß macht.

Die Raspberry Pi Foundation möchte durch eine taschengeldtaugliche Hardware die Welt des Computers für jedermann zugänglich machen und so dafür sorgen, dass niemand Angst zu haben braucht, Experimente damit anzustellen. Das Gerät wird ohne Gehäuse ausgeliefert, um das Herumbasteln damit zu erleichtern.

2011 wurde nach fünf Jahren Entwicklungsarbeit am Küchentisch der erste Prototyp des Raspberry Pi hergestellt. Nach einem Bericht über den Raspberry Pi im Blog des Technologiejournalisten Rory Cellan-Jones verbreitete sich die Nachricht wie ein Lauffeuer und die Foundation fragte sich, ob sie gerade etwas in Bewegung gesetzt hatte, das viel größere Ausmaße annahm als vermutet.

Dank einer geschickten technischen Planung war es möglich, den Raspberry Pi so günstig zu produzieren, dass er für ganze 25 Dollar verkauft werden konnte. Am 29. Februar 2012 um 6 Uhr morgens standen die ersten 10.000 Stück zum Verkauf. Sie waren in wenigen Minuten ausverkauft. Knapp zwanzig Monate später sind weltweit mehr als 2 Millionen Raspberry Pis über den Ladentisch gegangen.

Über dieses Buch

Während der Entwicklung des Raspberry Pi habe ich an der Universität Manchester an öffentlichen Projekten mitgewirkt, die weitere Leute zur Beschäftigung mit dem Computer ermutigen sollten.

Ich habe das Vorankommen des Raspberry Pi von einer sehr frühen Phase an verfolgt, weil ich der Ansicht war, dass er enormes Potenzial besitzt. Wie Tausende anderer Ingenieure auch war ich davon fasziniert, mit welchen Technologien die winzige Platine vollgepackt war. Andererseits war mir aber auch bewusst, dass viele weniger computerbegeisterte Leute ebendiese Platine nicht besonders spannend finden würden und sie eher als eine etwas unheimliche Ansammlung von Draht, Bauteilen und Metall betrachten. Ganz im Sinne der Foundation war es mein Ziel, das Staunen und die Freude weiterzugeben, die bei der Beschäftigung mit Computern möglich sind.

Der große Vorteil des Raspberry Pi ist es, dass Sie ihn an Orten anbringen können, an denen für einen normalen PC einfach kein Platz vorhanden ist. Ich wünschte mir, dass der Raspberry Pi für die Dinge, die den Leuten wichtig sind, eine Bedeutung bekommt. Um es zu vereinfachen, etwas an den Raspberry Pi anzuschließen, begann ich damit, in meiner Freizeit und an Wochenenden am heimischen Küchentisch eine digitale Schnittstelle namens *PiFace* zu entwickeln. Ich bin noch immer erstaunt, wenn ich sehe, was überall auf der Welt Leute mit dem Raspberry Pi und dem PiFace zustande bringen. Dazu zählen Kinder, die Roboter bauen, Türöffnungssysteme für Senioren, Spiele und industrielle Anwendungen in Banken und Bahnhöfen.

Verwendung dieses Buches

Das Buch möchte die Frage »Ich habe einen Raspberry Pi – und was jetzt?« beantworten und stellt haufenweise Projekte für den Raspberry Pi vor, die Sie inspirieren sollen.

Es ist in drei Teile gegliedert, die stellenweise aufeinander aufbauen, aber wenn Sie Ihren Raspberry Pi erst einmal zum Laufen gebracht haben, sollten Sie die Lektüre eigentlich problemlos an einer beliebigen Stelle des Buchs fortsetzen können. Folgen Sie einfach den schrittweisen Anleitungen, um schnell zu Ergebnissen zu gelangen. Scheuen Sie sich aber nicht, dabei herumzuexperimentieren, denn hier beginnt der Spaß ja erst! Falls Sie die Projekte ausweiten möchten, stehen dazu Hintergrundinformationen bereit, die Ihnen helfen sollen, die notwendigen Fähigkeiten und Kenntnisse zu erwerben.

Am Ende eines jeden Kapitels finden Sie Vorschläge zum Ausbau der Projekte, aber vermutlich werden Sie schon eigene Ideen haben. Wir sind sehr daran interessiert zu erfahren, zu welchen Ergebnissen Sie kommen, also teilen Sie Ihre Resultate unter dem Stichwort »RaspberryPiProjects« über Social Media wie Facebook, Twitter oder YouTube.

Falls Sie irgendwo nicht weiterkommen, stehen einige der Programmlistings unter <http://www.mitp.de/9699> oder <http://www.wiley.com/go/raspberrypiprojects> zum Herunterladen bereit. Wenn Sie das Programmieren erlernen, gehört es jedoch einfach dazu, Programmcode einzugeben, daher ist nicht sämtlicher Code verfügbar!

Ein Großteil der Hintergrundinformationen ist für den Schulunterricht geeignet, und das Buch kann unterrichtsbegleitend verwendet werden. Falls Sie selbst Lehrer sind, sollten Sie nach weiteren Informationen Ausschau halten, die den Schülern beim Lernen durch Projekte mit dem Raspberry Pi helfen können.

Teil 1: Erste Schritte mit dem Raspberry Pi

In diesem Teil erfahren Sie, wie Sie den Raspberry Pi zusammenstöpseln, die Software installieren und ihn zum Laufen bringen. Außerdem erhalten Sie eine Einführung in die Programmiersprache Python.

- Kapitel 1, *Den Raspberry Pi zum Laufen bringen*, erläutert die ersten grundlegenden Schritte, um den Raspberry Pi betriebsbereit zu machen.
- Kapitel 2, *Ein erstes Programm: Der Beschimpfungsgenerator* führt Sie in die Programmiersprache Python ein.

Teil 2: Software-Projekte

Hier finden Sie eine Reihe interessanter Programme:

- In Kapitel 3, *Tic-Tac-Toe*, werden Sie dieses Spiel unter besonderer Berücksichtigung von Listen und künstlicher Intelligenz programmieren.
- Kapitel 4, *Hier sind die Nachrichten*, führt Ihnen vor, wie sich ein Teleprompter realisieren lässt.
- Kapitel 5, *Ping*, beschreibt die Programmierung eines Ping-Pong-Spiels und zeigt Ihnen, wie der Computer Ballbewegungen, Kollisionen und die Reflexion an einer Oberfläche simuliert.

- In Kapitel 6, *Pie-Man: Ein Pac-Man-Klon*, wird mittels animierter Sprites, Ebenen und transparenter Pixel ein Pac-Man-Klon programmiert.
- In Kapitel 7, *Minecraft Maze Maker*, wird ein Python-Programm entwickelt, das Minecraft-Labyrinth erzeugt.

Teil 3: Hardware-Projekte

- Kapitel 8, *Schnappen*, stellt ein einführendes Hardware-Projekt vor, das anhand verschiedenfarbiger LEDs eine Variante des Spiels »Schnappen« realisiert und Ihnen zeigt, wie Sie LEDs gefahrlos mit Spannung versorgen und SMD-Bauteile verwenden.
- In Kapitel 9, *Reaktionstest*, werden Sie einen einfachen computergesteuerten Schaltkreis aufbauen.
- Kapitel 10, *Gezwitscher*, erörtert, wie Ihr Code mit Twitter kommunizieren kann und wie Sie einige Haushaltsgegenstände zweckentfremden können.
- In Kapitel 11, *Disco-Beleuchtung*, werden LED-Lichterketten im Takt der Musik angesteuert.
- In Kapitel 12, *Türschloss*, erfahren Sie, wie der Computer RFID-gesteuert ein Türschloss öffnet und wie die Authentifizierung funktioniert.
- Kapitel 13, *Hausautomation*, führt vor, wie Sie Hausautomation dazu verwenden können, ihre häusliche Umgebung durch Mikroschalter, Bewegungsmelder, eine Webcam und E-Mail-Alarme intelligenter zu gestalten.
- In Kapitel 14, *Computergesteuertes Modellbahnrennen*, werden Sie die Steuerung einer Modellrennbahn neu verdrahten und sie dazu verwenden, den Punktstand in einem Quiz für zwei Spieler aufzuzeichnen.
- Kapitel 15, *Drehgeber mit Facebook-Anbindung*, zeigt Ihnen den Einsatz von Drehgebern zum Zeichnen von Bildern, die automatisch bei Flickr oder Facebook eingestellt werden.
- In Kapitel 16, *Das Pi-Pendel – ein Harmonograph*, wird eine Maschine zum Zeichnen komplizierter Muster gebaut, wobei ein Arduino dem Raspberry Pi beim Sammeln von Daten in Echtzeit hilft.
- Kapitel 17, *Das Hightech-Vogelhäuschen – Beobachtung der Tierwelt*, wird der Umbau eines Vogelhäuschens vorgestellt, um damit die Vögel in Ihrem Garten zu beobachten.

Ausblick

Es ist absehbar, dass Computer eine weiter zunehmende Bedeutung erlangen und in weitere Lebensbereiche vordringen. Die Systeme werden immer komplexer und sind immer stärker miteinander vernetzt. Es ist nicht unwahrscheinlich, dass Ihre Waschmaschine oder Ihr Kühlschrank in naher Zukunft mit Ihrem Smartphone kommuniziert. Wir alle benötigen bessere Computerkenntnisse, um damit umgehen zu können. Aber neue Arten der Computernutzung eröffnen auch neue Perspektiven und bieten Möglichkeiten, das Leben der Menschen zum Besseren zu ändern und viele der zweifelsohne vorhandenen Probleme auf dieser Welt zu lösen.

Bevor Sie sich in das Abenteuer Raspberry Pi stürzen, muss ich eine Warnung aussprechen: Wenn Sie einmal damit anfangen, werden Sie möglicherweise nie wieder aufhören

wollen! Elektronik und Programmierung besitzen Suchtpotenzial – und wer könnte schon vorhersagen, wie Sie Ihre durch die Lektüre dieses Buches neu erworbenen Kenntnisse einsetzen?

Das Anfertigen oder Zusammenbauen eigener Geräte kann eine sehr erfüllende und zufriedenstellende Erfahrung sein. Wir wünschen uns, dass mehr Leute auf dieser Welt, statt Technologie nur zu konsumieren, selbst zu Produzenten von Technologie werden. Die Projekte in diesem Buch sind dafür ein guter Ausgangspunkt, aber der wahre Lohn Ihrer Anstrengungen offenbart sich erst, wenn Sie Ihre eigenen Ideen Realität werden lassen.

Willkommen in der Welt des Bastelns mit digitaler Elektronik. Sind Sie bereit dazu, die Zukunft mitzuerfinden?

Teil I

Erste Schritte mit dem Raspberry Pi

In diesem Teil:

- **Kapitel 1**
Den Raspberry Pi zum Laufen bringen 25
- **Kapitel 2**
Ein erstes Programm: Der Beschimpfungsgenerator . . . 45

Den Raspberry Pi zum Laufen bringen

von Dr. Andrew Robinson

Dieses Kapitel behandelt folgende Themen

- Die Aufgaben des Betriebssystems
- Installation des Betriebssystems auf einer SD-Karte
- Anschließen des Raspberry Pi
- Der Startvorgang
- Fehlerbehebung: Falls Ihr Raspberry Pi nicht startet

Dieses Kapitel ist ein Leitfaden für Einsteiger, der Sie durch die ersten Schritte bei der Einrichtung eines Raspberry Pi führt. Es beginnt mit dem Auspacken des Geräts und endet mit der gewünschten Anzeige auf dem Bildschirm. Auch wenn Sie Ihren Raspberry Pi bereits erfolgreich eingerichtet haben, sollten Sie das Kapitel zumindest überfliegen, denn Sie erfahren hier nicht nur, wie ein 21-jähriger Student die Welt verändert hat, sondern lernen auch die Funktionsweise des Betriebssystems Ihres Raspberry Pi kennen. Nach diesem Kapitel geht der Spaß beim Erstellen neuer Projekte erst richtig los!

1.1 Das Betriebssystem

Der Raspberry Pi verwendet als Betriebssystem weder Microsoft Windows noch Apples OS X, sondern Linux. Das *Betriebssystem* ist ein Programm, das es dem Benutzer erleichtert, die zugrunde liegende Hardware zu verwenden. Obwohl der Prozessor (der Chip in der Mitte des Raspberry Pi) eigentlich immer nur eine einzige Aufgabe erledigen kann, erweckt das Betriebssystem den Eindruck, dass viele Dinge gleichzeitig geschehen, indem es blitzschnell zwischen den verschiedenen Aufgaben hin- und herschaltet. Darüber hinaus steuert das Betriebssystem die Hardware und sorgt dafür, dass der Raspberry Pi mit dem Netzwerk oder einer SD-Karte kommunizieren kann.

Linux

Einen Teil seines Erfolges verdankt der Raspberry Pi der begeisterten Nutzergemeinschaft, die ihn unterstützt. Linux ist ein Beleg dafür, was sich durch die weltweite Unterstützung Freiwilliger erreichen lässt.

Im Jahre 1991 begann Linus Torvalds als 21-jähriger Student an der Universität Helsinki einfach nur aus Spaß an der Freude mit der Arbeit an einem Betriebssystem. Ein Jahr später war sein Hobby-Betriebssystem für 80386-PCs unter der Bezeichnung *Linux* online verfügbar. Dadurch war es Freiwilligen weltweit möglich, Beiträge zur Weiterentwicklung zu leisten, Bugs zu finden und zu beseitigen und die Arbeit anderer Freiwilliger wiederzuverwenden oder an eigene Projekte anzupassen. Wenn Sie die Projekte dieses Buches nachvollzogen und Ihre Computerkenntnisse erweitert haben, warum sollte dann nicht eines Ihrer Projekte zu einem ähnlichen Erfolg werden wie dasjenige von Linus Torvalds?

Linux erfreute sich zunehmender Beliebtheit, und neben der Anwendung als normales PC-Betriebssystem wird es heute auch bei einem Großteil der Webserver, für Android-Geräte und in den meisten Supercomputern eingesetzt. Für Sie ist es an dieser Stelle von Bedeutung, dass es auch auf dem Raspberry Pi Anwendung findet.

Linux-Distributionen

Da Linux öffentlich verfügbar ist, haben verschiedene Unternehmen und Einrichtungen gewisse Änderungen daran vorgenommen und geben diese modifizierten Versionen ihrerseits weiter. Das hat zu einer Reihe unterschiedlicher Linux-Distributionen wie Red Hat, Fedora, Debian, Arch, Ubuntu und openSUSE geführt. Einige Unternehmen verkaufen ihre Versionen und/oder bieten kostenpflichtigen Support dafür an, während andere Distributionen völlig kostenlos zu haben sind. *Raspbian* beruht auf einer Debian-Distribution mit einigen speziellen Anpassungen für den Raspberry Pi und ist Gegenstand dieses Buches.

Linux ist das auf dem Raspberry Pi am häufigsten eingesetzte Betriebssystem. Der weitreichende Einsatz von Linux (denken Sie nur an die mehr als 2 Millionen Raspberry Pis, Android-Telefone und Webserver) zeigt in beeindruckender Weise, was aus einer wirklich guten Idee werden kann. Wer weiß, ob Ihnen bei der Beschäftigung mit dem Raspberry Pi nicht eine ähnlich gute Idee kommt, deren Auswirkungen diejenigen der Ideen von Linus Torvalds oder der Raspberry Pi Foundation erreichen oder sogar übertreffen? Also nichts wie ran!

1.1.1 Das Betriebssystem auf eine SD-Karte schreiben

Ohne Betriebssystem weiß der Raspberry Pi überhaupt nichts mit seiner Hardware anzufangen. Nach dem Einschalten sucht er daher auf einer eingesteckten SD-Karte nach einem Betriebssystem. Sie müssen Ihrem Raspberry Pi also eine SD-Karte mit Betriebssystem bereitstellen.

Sie können entweder eine SD-Karte erwerben, auf der bereits ein Betriebssystem installiert ist, oder mit Ihrem Rechner ein Betriebssystem auf Ihre eigene SD-Karte schreiben. Diese vorbereiteten SD-Karten sind zwar besonders einfach einsetzbar, aber meist auch etwas teurer als normale SD-Karten. Eine solche SD-Karte selbst zu erstellen ist nicht weiter schwierig und erfordert nur minimal mehr Aufwand als ein reiner Kopiervorgang.

Vorbereitete SD-Karten

Startfähige SD-Karten sind Bestandteil von Einsteiger-Kits oder separat bei Elektronikhändlern wie Reichelt, Conrad oder Völkner erhältlich. Für den Anfang ist eine für rund 5 Euro

erhältliche 4-GB-Karte ausreichend, aber sparen Sie nicht an der falschen Stelle. 8-GB-Karten sind auch schon für weniger als 10 Euro erhältlich.

Dateisysteme

Massenspeicher für Computer wie SD-Karten, USB-Sticks oder SSD-Festplatten bestehen im Wesentlichen aus Millionen einzelner Speichersegmente, in denen kleinere Datenmengen in einer Art Gitter abgelegt werden. Die einzelnen Speicherelemente, die man als *Blöcke* bezeichnet, werden mittels eines Koordinatensystems angesprochen. Stellen Sie sich ein Blatt kariertes Papier in der Größe eines Fußballfelds vor, das in verschiedene Bereiche (die einzelnen Speicherblöcke) aufgeteilt ist. Das Betriebssystem verwendet diese Blöcke, um ein *Dateisystem* bereitzustellen. Es kümmert sich darum, wie die Daten in diesem riesigen Speicher abgelegt werden. Wenn der Benutzer einen Dateinamen eingibt, sorgt das Betriebssystem dafür, dass alle Datenblöcke dieser Datei in der richtigen Reihenfolge eingelesen und zusammengesetzt werden. Da es verschiedene Möglichkeiten gibt, diese Speicherblöcke zu organisieren, speichern unterschiedliche Dateisysteme ein und dieselbe Datei daher auch in jeweils eigenen Formaten ab.

Normalerweise verwendet Microsoft Windows das Dateisystem FAT (*File Allocation Table*, Dateibelegungstabelle) oder NTFS (*New Technology File System*, Dateisystem neuer Technologie), OS X nutzt HFS+ (*Hierarchical File System Plus*, Hierarchisches Dateisystem Plus) und Linux setzt ext (*Extended File System*, Erweitertes Dateisystem) ein. Die meisten leeren SD-Karten sind ab Werk mit dem Dateisystem FAT formatiert. Da auf dem Raspberry Pi Linux läuft, wird als Dateisystem ebenfalls ext verwendet, das dementsprechend eingerichtet und mit Dateien befüllt werden muss.

Images

Beim Herunterladen des Betriebssystems für den Raspberry Pi ist oft von einem *Image* die Rede, was eine etwas irreführende Bezeichnung ist. Gemeint ist ein Abbild des zugrunde liegenden Speichermediums. (Stellen Sie sich statt der ganzen einzelnen Dateien eine Luftaufnahme des gesamten Fußballfelds vor, auf der sogar die leeren Speicherblöcke sichtbar sind! Wenn Sie die auf der Luftaufnahme erkennbaren Speicherblöcke auf einem anderen Fußballplatz exakt nachstellen, ergibt sich damit eine exakte Kopie sämtlicher Dateien des Originals.)

Nun lässt sich ein solches Image (zum Herunterladen) leicht als eine einzelne Datei innerhalb eines anderen Dateisystems speichern, die in dieser Form aber nicht zum Ausführen auf dem Raspberry Pi geeignet ist. Wenn Sie einfach nur eine Imagedatei auf eine FAT-formatierte SD-Karte kopieren, wird der Raspberry Pi nicht funktionieren. Sie müssen vielmehr Ihr Betriebssystem anweisen, die Daten blockweise auf die SD-Karte zu übertragen, damit die einzelnen Blöcke Ihrer Karte genau den Blöcken auf der Karte des Originals entsprechen. Auf diese Weise »sieht« Linux auf der Karte ein Dateisystem, das exakt mit dem Dateisystem auf der Karte des Originals übereinstimmt.

Kurz: Images bieten eine einfache Möglichkeit, ein vollständiges Dateisystem, inklusive aller Dateien, Zugriffsrechte, Eigenschaften und so weiter, zu klonen.

Vorbereiten einer SD-Karte

Es gibt zwei Methoden, um eine startfähige SD-Karte für den Raspberry Pi einzurichten: Verwenden Sie NOOBS oder übertragen Sie selbst ein Image auf die Karte.

NOOBS verwenden

NOOBS steht für *New Out Of Box Software* (etwa: Neue Software, die sofort einsatzbereit ist; außerdem ein Wortspiel mit dem englischen *noobs*, was »Anfänger« oder »Neulinge« bedeutet) und dient zur Automatisierung des Übertragens von SD-Kartenimages für den Raspberry Pi. NOOBS startet Ihren Raspberry Pi von einer FAT-formatierten SD-Karte, partitioniert sie neu und kloniert ein startfähiges Dateisystem darauf. Mehr als das Herunterladen (<http://www.raspberrypi.org/downloads>) und Entpacken von NOOBS sowie das Formatieren einer SD-Karte auf Ihrem Desktop-PC ist hierfür nicht erforderlich. Einige Betriebssysteme formatieren die Karten allerdings nicht ganz korrekt, daher kann es nötig sein, ein weiteres Programm zur Formatierung der Karte herunterzuladen. NOOBS ist wirklich einfach, funktioniert jedoch nicht in allen Fällen und ist unter manchen Umständen langsamer. Wie dem auch sei, die nachstehend beschriebene selbst gemachte Lösung ist ohnehin zufriedenstellender.

Image übertragen

Sie benötigen eine SD-Karte mit mehr als 2 GB Speicherplatz. Eine 4-GB-Karte ist also bestens geeignet.

Besuchen Sie <http://www.raspberrypi.org/downloads> und folgen Sie dem Link zum Herunterladen der neuesten Raspbian-Version. Halten Sie nach einem Dateinamen Ausschau, der das Wort *raspbian* sowie ein Datum enthält und auf *.zip* endet. Notieren Sie sich außerdem die SHA-1-Prüfsumme. Aufgrund der ständigen Weiterentwicklung werden regelmäßig neue Versionen veröffentlicht, daher wird die tatsächliche Bezeichnung von der in der nachfolgenden Anleitung verwendeten abweichen. Möglicherweise stimmt auch der Pfad zum Speicherort, an den Sie die Datei herunterladen, nicht exakt überein. Verwenden Sie also bitte die entsprechenden eigenen Werte. Auf der Seite zum Herunterladen finden Sie Links zu anderen Distributionen und Betriebssystemen, die Sie sich später ansehen können. Fürs Erste ist es das Beste, bei Raspbian zu bleiben, denn es ist zuverlässig, bringt eine für Einsteiger gut geeignete Sammlung an Programmen mit und die Beispiele im Buch beziehen sich ebenfalls darauf.

Prüfsummen

Eine *Prüfsumme* stellt eine einfache Möglichkeit zur Überprüfung der Datenintegrität dar. Sie können damit feststellen, ob Daten beschädigt oder verändert worden sind. Dabei handelt es sich um eine durch ein spezielles mathematisches Verfahren ermittelte Summe, die der Datenanbieter bereitstellt. Wenn Sie die Daten erhalten, können Sie dasselbe Verfahren zur Ermittlung dieser Summe anwenden. Stimmen die Resultate überein, können Sie sich so gut wie sicher sein, dass die Daten intakt sind, ohne diese Bit für Bit vergleichen zu müssen. Prüfsummen sind im Computerwesen weitverbreitet.

Beispiele hierfür sind die Kommunikation im Netzwerk, die Verarbeitung von Kreditkartendaten oder auch Barcodes. Unfehlbar sind Prüfsummen nicht, aber Sie dürfen sich schon ziemlich sicher sein, dass die Daten in Ordnung sind, wenn die Prüfsumme stimmt.

Vorbereiten einer SD-Karte unter Windows

Unter Windows ist es nicht ganz einfach, Prüfsummen zu testen, daher wird im Folgenden vorausgesetzt, dass die heruntergeladene Imagefile unversehrt ist. Folgen Sie nach dem Herunterladen der Datei den nachstehenden Schritten, um die Datei zu entpacken und auf die SD-Karte zu übertragen.

1. Dekomprimieren Sie die heruntergeladene Datei `2013-07-26-wheezy-raspbian.zip`.
2. Stecken Sie eine SD-Karte ein und notieren Sie sich den zugehörigen Laufwerksbuchstaben (wie zum Beispiel E:). Vergewissern Sie sich, dass die Karte keine noch benötigten Daten enthält, denn sie wird vollständig überschrieben.
3. Besuchen Sie die Website <https://launchpad.net/win32-image-writer> und laden Sie sich im Downloadbereich (auf der rechten Seite) die ausführbare Version des Programms Win32DiskImager herunter und entpacken Sie es.
4. Starten Sie `Win32DiskImager.exe` als Administrator. Je nach Konfiguration Ihres Systems müssen Sie auf das Programmsymbol doppelklicken oder es mit gedrückter Umschalttaste mit der rechten Maustaste anklicken und STARTEN ALS auswählen.
5. Wählen Sie im WIN32 DISK IMAGER-Fenster die Datei `2013-07-26-wheezy-raspbian.img` aus.
6. Wählen Sie im Einblendmenü der Geräte auf der rechten Seite den Laufwerksbuchstaben aus, den Sie in Schritt 2 notiert haben (siehe Abbildung 1.1).
7. Klicken Sie auf WRITE (Schreiben) und warten Sie, bis der Vorgang abgeschlossen ist. (Das kann 15 bis 30 Minuten dauern, haben Sie also ein wenig Geduld.)
8. Beenden Sie Win32DiskImager und werfen Sie die SD-Karte aus, die nun das Betriebssystem enthalten sollte.

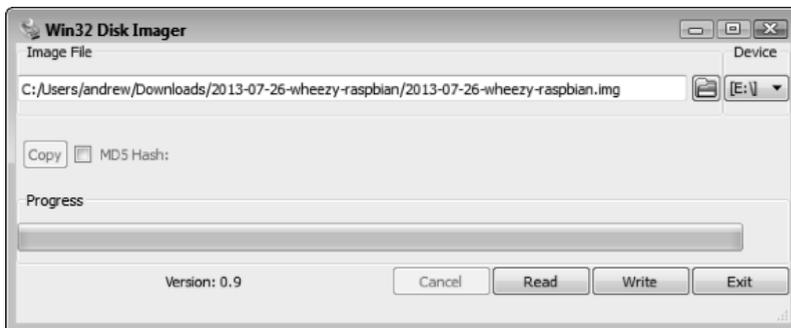


Abb. 1.1: Das Fenster des Win32DiskImagers

Vorbereiten einer SD-Karte mit Linux

Am einfachsten ist das Schreiben einer SD-Karte unter Linux auf der Kommandozeile, wie Sie den folgenden Schritten entnehmen können.

Linux-Zugriffsrechte und sudo

Linux beschränkt bestimmte Aktionen, die Daten anderer Benutzer gefährden könnten. Daher funktionieren gewisse Befehle nur, wenn Sie die erforderlichen Zugriffsrechte besitzen. Bei einigen Distributionen müssen Sie sich als Benutzer *root* (so lautet die Bezeichnung des Administratorbenutzerkontos) anmelden, um Kommandos mit erweiterten Zugriffsrechten auszuführen. In anderen Distributionen ist es ausgewählten Benutzern gestattet, einem solchen Kommando den Befehl *sudo* voranzustellen. In der Anleitung wird vorausgesetzt, dass mit Ihrem Benutzerkonto die Verwendung von *sudo* erlaubt ist. Geben Sie anderenfalls im Terminal das Kommando *su* ein, um zum *root*-Benutzer zu werden.

1. Öffnen Sie ein Terminalfenster und wechseln Sie mittels *cd* in das Verzeichnis, das die heruntergeladene Datei enthält (z. B. *cd Downloads*).
2. Entpacken Sie die heruntergeladene Datei durch Eingabe von *unzip*, gefolgt von einem Leerzeichen und dem Dateinamen (beispielsweise *unzip 2013-07-26-wheezy-raspbian*).
3. Geben Sie *ls *.img* ein, um sich eine Liste der Imagedateien im aktuellen Arbeitsverzeichnis anzeigen zu lassen, und vergewissern Sie sich, dass die soeben entpackte Datei dabei ist.
4. Berechnen Sie nun die Prüfsumme der heruntergeladenen Datei, um sicherzugehen, dass sie nicht beschädigt ist oder manipuliert wurde:

```
sha1sum 2013-07-26-wheezy-raspbian.zip
```

Vergewissern Sie sich, dass die Prüfsumme mit der auf <http://www.raspberrypi.org/downloads> angegebenen übereinstimmt. Es ist zwar unwahrscheinlich, dass sie voneinander abweichen, aber falls doch, laden Sie die Datei erneut herunter und entpacken Sie sie ein zweites Mal.
5. Stecken Sie eine SD-Karte ein. Vergewissern Sie sich, dass sie keine wichtigen Daten enthält, denn sie wird vollständig überschrieben.
6. Geben Sie *dmesg* ein und suchen Sie nach dem Gerätenamen, den Linux für die gerade eingesteckte Karte verwendet. Normalerweise lautet die Bezeichnung *sdd*, *sde*, *sdf* oder ähnlich. Möglicherweise benutzt Linux stattdessen auch einen Namen der Form *mmcblk0*. Verwenden Sie in den weiteren Schritten stets diese Bezeichnung, wenn *sdX* vorkommt.
7. Falls Linux die SD-Karte automatisch ins Dateisystem eingehängt hat, müssen Sie sie durch Eingabe von *sudo umount /dev/sdX* wieder aushängen.
8. Überprüfen Sie, ob Sie wirklich den korrekten Gerätenamen gefunden haben, indem Sie *sudo fdisk -l /dev/sdX* eingeben und vergleichen Sie die angezeigte Größe mit derjenigen der eingesteckten Karte.
9. Wenn Sie absolut sicher sind, dass die Gerätebezeichnung stimmt, müssen Sie in der folgenden Befehlszeile, die das Image auf die Karte schreibt, *sdX* durch den in Schritt 6 ermittelten Namen ersetzen: