

Maik Schmidt

Raspberry Pi

Einstieg • Optimierung • Projekte

dpunkt.verlag



Maik Schmidt arbeitet seit beinahe 20 Jahren als Softwareentwickler für mittelständische und Großunternehmen. Er schreibt seit einigen Jahren Buchkritiken und Artikel für internationale Zeitschriften und hat auch selbst schon vier Bücher verfasst (»Enterprise Recipes with Ruby and Rails«, »Enterprise Integration with Ruby«, »Arduino: A Quick-Start Guide« und »Raspberry Pi: A Quick-Start Guide«).

Gern spricht er auf internationalen Konferenzen und seine besonderen Steckenpferde sind die Programmiersprache Ruby und die Mikroelektronik, speziell der Arduino.

Maik Schmidt

Raspberry Pi

Einstieg · Optimierung · Projekte

Maik Schmidt
contact@maik-schmidt.de

Übersetzung: Volkmar Gronau, Flensburg

Lektorat: Dr. Michael Barabas

Copy-Editing: Ursula Zimpfer, Herrenberg

Satz & Herstellung: Birgit Bäuerlein, Miriam Metsch

Umschlaggestaltung: Helmut Kraus, www.exclam.de

Druck und Bindung: M.P. Media-Print Informationstechnologie GmbH, 33100 Paderborn

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:

Buch 978-3-86490-032-7

PDF 978-3-86491-269-6

ePub 978-3-86491-270-2

1. Auflage 2013

1. aktualisierter Nachdruck

Translation Copyright für die deutschsprachige Ausgabe © 2013 dpunkt.verlag GmbH

Ringstraße 19 B

69115 Heidelberg

Copyright der amerikanischen Originalausgabe © 2012 The Pragmatic Programmers, LLC.

Title of American original: Raspberry Pi: A Quick-Start Guide.

Pragmatic Bookshelf, The Pragmatic Programmers, LLC, Dallas, Texas, Raleigh, North Carolina

<http://pragprog.com>

ISBN: 978-1-937785-04-8

Die vorliegende Publikation ist urheberrechtlich geschützt. Alle Rechte vorbehalten. Die Verwendung der Texte und Abbildungen, auch auszugsweise, ist ohne die schriftliche Zustimmung des Verlags urheberrechtswidrig und daher strafbar. Dies gilt insbesondere für die Vervielfältigung, Übersetzung oder die Verwendung in elektronischen Systemen.

Es wird darauf hingewiesen, dass die im Buch verwendeten Soft- und Hardware-Bezeichnungen sowie Markennamen und Produktbezeichnungen der jeweiligen Firmen im Allgemeinen warenzeichen-, marken- oder patentrechtlichem Schutz unterliegen.

Alle Angaben und Programme in diesem Buch wurden mit größter Sorgfalt kontrolliert. Weder Autor noch Verlag können jedoch für Schäden haftbar gemacht werden, die in Zusammenhang mit der Verwendung dieses Buches stehen.

Danksagungen

Immer wenn ich Menschen erzähle, dass ich Buchautor bin, sehen sie mich für ein paar Sekunden verträumt an. Scheinbar denken noch immer viele, dass Schreiben bedeutet, hinter einem alten Holztisch zu sitzen, an stürmischen Tagen aus dem Fenster zu schauen und ein gutes Glas Rotwein zu trinken. Bei mir ist das eher selten der Fall, obwohl ich meistens viel Spaß habe, wenn ich Bücher schreibe.

Auch beim Schreiben dieses Buchs hatte ich meinen Spaß, hauptsächlich wegen der tollen Unterstützung durch meine Lektorin Jacqueline Carter. Sie munterte mich immer wieder auf und ihre durchdachten Ratschläge lösten viele meiner Probleme. Vielen herzlichen Dank, Jackie!

Wie immer war das gesamte Team bei The Pragmatic Bookshelf unglaublich hilfsbereit und flink. Ohne euch wäre dieses Buch nicht entstanden.

Dieses Buch beschäftigt sich mit Elektronik und alle Schaltungen habe ich mit Fritzing¹ entworfen. Ich bin dem Fritzing-Team zutiefst dankbar, dass sie ein solches Werkzeug kostenfrei anbieten. Auch die Adafruit-Fritzing-Bibliothek² war ungemein hilfreich. Außerdem möchte ich Gordon Henderson für WiringPi³ danken. Es macht das Basteln mit den GPIO-Pins des Raspberry Pi zum Kinderspiel und hat mir unzählige Stunden der Fehlersuche im Low-Level-Code erspart.

Simon Quernhorst gab mir die freundliche Genehmigung, Screenshots seines tollen Spiels A-VCS-tec Challenge in diesem Buch zu verwenden.

Auch meinen Korrekturlesern kann ich nicht genug danken: Daniel Bachfeld, Gordon Haggart, Michael Hunter, Thomas Lockney, Angus Neil,

1) <http://fritzing.org/>

2) <https://github.com/adafruit/Fritzing-Library/>

3) <https://projects.drogon.net/raspberry-pi/wiringpi/>

Stephen Orr, Mike Riley, Sam Rose, Mike Williamitis und Tony Williamitis. Eure Anmerkungen und Vorschläge haben dieses Buch um vieles besser gemacht.

Schließlich danke ich Tanja und Mika für ihr Verständnis und ihre Geduld. Ich bin dankbar, dass es euch gibt.

Vorwort

In den letzten Jahrzehnten sind Computer immer billiger geworden, sodass Sie sie heute nicht nur auf Ihrem Schreibtisch finden, sondern in fast jedem Elektronikgerät, angefangen beim Smartphone bis zum DVD-Player. Trotzdem sind sie noch nicht so günstig, dass Sie beim Einkaufen einfach einen mitnehmen würden. Normalerweise planen Sie den Kauf Ihres nächsten PCs gründlich, weil Sie ihn mehrere Jahre lang verwenden werden.

Computer wie der Raspberry Pi werden diese Situation in naher Zukunft komplett verändern. Der Raspberry Pi, oder kurz Pi, ist ein vollwertiger Desktop-PC, der nur 35 Euro kostet. Sie können ihn direkt ans Internet anschließen und HD-Videos auf ihm abspielen. Er läuft unter Linux, sodass Sie für das Betriebssystem nichts bezahlen müssen. Dadurch wird der Pi zum ersten Wegwerfcomputer in der Geschichte.

Die Raspberry-Stiftung¹ entwickelte den Pi, um Kindern das Programmieren beizubringen. Es überrascht daher nicht, dass der Pi sich hierzu hervorragend eignet. Darüber hinaus kann der Pi für viele andere spannende Dinge verwendet werden. Sie können ihn beispielsweise in ein Multimediacenter verwandeln, ihn als günstigen und leistungsfähigen Webserver einsetzen oder damit Spieleklassiker spielen.

Der Pi ist auch ein tolles Gerät für Elektronikexperimente. Im Gegensatz zu vielen beliebten Mikrocontroller-Boards wie Arduino verfügt der Pi über ein vollständiges Betriebssystem und Sie können eine Vielzahl von Programmiersprachen verwenden, um Ihre Projekte umzusetzen.

1) <http://www.raspberrypi.org/>

Mit günstigen und kleinen Geräten wie dem Raspberry Pi hat eine neue Ära des allgegenwärtigen Computers begonnen, und Sie können Teil davon sein. Mit diesem Buch bekommen Sie einen schnellen Einstieg.

Für wen dieses Buch gedacht ist

Dieses Buch ist für alle gedacht, die mit dem Raspberry Pi schnell loslegen möchten. Selbst wenn Sie etwas Erfahrung mit anderen Computern haben, werden Sie schnell erkennen, dass der Pi in vielerlei Hinsicht anders ist. Mit diesem Buch vermeiden Sie die häufigsten Fallstricke.

Sie können aus einer ganzen Reihe von Betriebssystemen für den Pi wählen. Der Schwerpunkt dieses Buchs liegt auf Debian Linux (Raspbian), weil es für den Einsteiger die bequemste Alternative darstellt. Wenn Sie noch nie mit Linux gearbeitet haben, sollten Sie mit Anhang A »Eine Einführung in Linux« beginnen. Aber auch wenn Sie schon mit Linux etwas vertraut sind, könnten Sie noch Neues erfahren, denn Linux auf dem Pi ist in mancher Hinsicht anders.

Am meisten holen Sie aus diesem Buch heraus, wenn Sie einen Raspberry Pi besitzen und die Beispiele direkt nachvollziehen.

Was Sie in diesem Buch finden

Der Raspberry Pi wird ohne Handbuch geliefert, mit diesem Buch jedoch lernen Sie schrittweise, wie Sie das Maximum aus Ihrem Minicomputer herausholen können. Sie erfahren nicht nur, wie die Pi-Hardware prinzipiell funktioniert, sondern auch, wie Sie verschiedene Betriebssysteme laufen lassen und wie Sie den Pi für spezielle Anwendungen, wie z.B. ein Multimediacenter, einsetzen.

Die folgenden Dinge lernen Sie im Verlauf dieses Buchs:

- Das Buch beginnt mit einer Einführung in die Hardware des Raspberry Pi. Sie lernen die Anschlüsse des Pi kennen und welche Zusatzhardware Sie benötigen, um den Pi das erste Mal zu starten.
- Wenn Sie alle erforderlichen Geräte an den Pi angeschlossen haben, benötigen Sie ein Betriebssystem. Obwohl der Pi ein recht junges Projekt ist, können Sie bereits zwischen mehreren wählen und lernen deren Für und Wider kennen.

- Ein Betriebssystem auf dem Pi zu installieren unterscheidet sich ziemlich von der Installation auf einem regulären PC. Sie werden lernen, wie Sie Debian Linux auf Ihrem Pi zum Laufen bringen.
- Debian Linux funktioniert schon in der Grundkonfiguration gut auf dem Pi. Das Maximum holen Sie jedoch mit ein paar Konfigurationsparametern heraus. Zum Beispiel ist es sinnvoll, das Tastaturlayout richtig einzustellen. Sie lernen weiter, wie Sie neue Software installieren, aktualisieren und wieder entfernen.
- Die Pi-Hardware, insbesondere was die Grafik angeht, ist in vielerlei Hinsicht speziell. Abhängig vom eingesetzten Bildschirm müssen Sie einige Low-Level-Einstellungen in der Pi-Firmware vornehmen. Sie lernen, welche Einstellungen verfügbar sind und wie Sie die häufigsten Firmware-Probleme lösen.
- Um zu sehen, was Sie mit nur minimalem Einsatz erreichen können, verwandeln Sie den Pi in ein Kiosk-System. Er zeigt eine statische Dia- show an oder aktuelle Informationen aus dem Internet.
- Bis hierhin haben Sie den Pi mehr oder weniger isoliert betrieben. Jetzt binden Sie ihn in ein Netzwerk ein. Sie verwenden den Pi für alltägliche Aufgaben wie das Surfen im Netz, greifen per SSH darauf zu und verwandeln ihn dann in einen kompletten Webserver. Sie lernen auch, wie Sie die Oberfläche des Pi vom PC aus bedienen und umgekehrt.
- Mit dem Projekt XBMC verwandeln Sie Ihren Raspberry Pi mit Leichtigkeit in ein Multimediacenter. Sie können damit nicht nur Ihren Freunden Fotos im Wohnzimmer präsentieren, sondern auch Musik in den gängigen Formaten abspielen. Dasselbe gilt für Ihre Lieblingsfilme und Fernsehshows in HD.
- Das Raspberry-Team hat den Pi eigentlich zu Lernzwecken entworfen, Sie können ihn aber leicht in eine Spielkonsole verwandeln. Auch wenn es möglich ist, Ego-Shooter laufen zu lassen, bevorzugen Sie vielleicht eher klassische Spiele, wie Adventures und Point-and-Click-Spiele.
- Einer der größten Vorteile des Pi gegenüber normalen PCs sind seine GPIO-Pins. In den letzten beiden Kapiteln des Buchs lernen Sie, wie Sie damit Ihre eigenen Elektronikprojekte und den Pi verbinden können.
- Der Anhang enthält eine kurze Einführung in Linux. Wenn Sie noch nie mit Linux gearbeitet haben, sollten Sie mit dem Anhang beginnen.

Wo bekomme ich einen Raspberry Pi und Zusatzhardware?

Zur Drucklegung produzierten und verkauften nur zwei Händler den Raspberry Pi in Europa. Um einen Pi zu kaufen, besuchen Sie den Webshop von Farnell² oder RS Components³.

Diese Anbieter verkaufen auch viel Zubehör für den Pi wie Netzteile, Tastaturen, Mäuse usw. Die Ware wird aus Großbritannien versendet.

Adafruit⁴ bietet ebenfalls nützliches Zubehör, versendet allerdings nur aus den USA.

Sie finden eine schnell wachsende Liste mit kompatibler Hardware im Projekt-Wiki: http://elinux.org/RPi_VerifiedPeripherals.

Wenn Sie Zweifel haben, kaufen Sie die Hardware lieber bei einem der hier genannten Anbieter. Insbesondere verkaufen einige Händler noch alte Modelle mit nur 256MB RAM zu recht überzogenen Preisen.

Debian Linux

Das beliebteste Betriebssystem für den Pi ist Linux. Es sind verschiedene Distributionen für den Pi erhältlich; wir haben uns für Debian entschieden. Erst kürzlich hat das Debian-Team die letzte Version namens wheezy freigegeben, und dank der Anstrengungen des Raspbian-Teams⁵ ist es schon für den Pi erhältlich. Raspbian löst Debian »squeeze« ab, das lange Zeit das Referenzbetriebssystem für den Pi war.

Die Raspbian-Distribution hat gegenüber ihren Vorgängern einige Vorteile. Sie ist viel schneller, bietet mehr aktuelle Software und wird bald immer stabiler laufen. Es ist außerdem die bevorzugte Lösung des Raspberry-Teams, weshalb sich dieses Buch auf Raspbian konzentriert.

2) <http://www.farnell.com/>

3) <http://www.rs-online.com/>

4) <http://adafruit.com/category/105>

5) <http://www.raspbian.org/>

Codebeispiele und Konventionen

In diesem Buch finden Sie Codebeispiele in PHP, Python, HTML und in der Programmiersprache der Bash-Shell. Sie alle sind recht kurz und wenn Sie schon programmiert haben, werden Sie sie sicher verstehen. Wenn Sie noch keine Software entwickelt haben, können Sie den Code trotzdem auf den Pi kopieren und zum Laufen bringen.

Onlinematerial

Für dieses Buch gibt es unter <http://www.dpunkt.de/raspi> eine eigene Webseite, auf der Sie den Code aller Beispiele herunterladen können. Auf der Webseite haben Sie auch die Möglichkeit, am Diskussionsforum teilzunehmen und andere Leser oder mich zu treffen. Wenn Sie Fehler in der Software oder im Text dieses Buchs finden oder auf andere Ärgernisse stoßen, lassen Sie es mich bitte über die Errata-Seite dieses Buchs wissen.

Jetzt ist es Zeit, den Raspberry Pi auszupacken und ein bisschen Spaß zu haben!

Stimmen zu Raspberry Pi

Der Raspberry Pi bringt die goldenen Zeiten des Experimentierens mit Heimcomputern zurück und Maiks Buch ist der ideale Einstieg. Die beschriebenen Projekte sind ideal für Anwender des Raspberry Pi, egal welchen Alters oder mit welchen Vorkenntnissen.

Tony Williamitis
Senior Embedded Systems Engineer

Schmidt wirft einen kurzen Blick auf die vielen Dinge, die Sie mit einem Raspberry Pi ohne viel Zubehör anstellen können. Das hat mir sehr dabei geholfen, zu erkennen, wozu ich meinen Pi einsetzen kann, und Ideen für die nächsten Projekte geliefert!

Stephen Orr
Technikenthusiast und Web Developer

Dies ist das Anwenderhandbuch, das sich alle Raspberry Pi-Anwender zulegen sollten, bevor sie loslegen. Es ist klar verständlich und kompakt. Mehr benötigt man nicht.

Thomas Lockney
Professional Geek DorkbotPDX

Eine wunderbar klare, präzise und nützliche Einführung in den Raspberry Pi.

Michael Hunter

Inhaltsverzeichnis

1	Lernen Sie den Raspberry Pi kennen	1
1.1	Die Hardware	1
Was im Pi steckt	1	
Was der Pi nicht bietet	6	
1.2	Was Sie sonst noch benötigen	7
Auswählen eines Netzteils	7	
Auswählen einer SD-Karte	8	
Anschließen von Tastatur und Maus	9	
Auswählen eines Bildschirms	10	
Auswählen des richtigen Netzwerk-Equipments	10	
Hinzufügen eines Gehäuses	10	
1.3	Die nächsten Schritte	11
2	Ein Betriebssystem installieren	13
2.1	Eine Auswahl treffen	13
Arch Linux ARM	15	
RISC OS	16	
2.2	Vorbereiten einer bootfähigen SD-Karte	16
Vorbereiten einer SD-Karte unter Windows	17	
Vorbereiten einer SD-Karte unter Linux	19	
Vorbereiten einer SD-Karte unter Mac OS X	20	
2.3	Die nächsten Schritte	23

3	Raspbian konfigurieren	25
3.1	Den Pi zum ersten Mal starten	25
3.2	Anpassen Ihrer Installation mit Raspi-config	27
Nutzen des gesamten Speichers Ihrer SD-Karte	27	
Konfigurieren des Overscan-Modus	27	
Ändern der Tastaturbelegung und der Ländereinstellung	28	
Einstellen von Zeitzone, Zeit und Datum	30	
Anpassen der Speicherkonfiguration	31	
Ändern Ihres Passworts	31	
Den Pi beschleunigen (Overclocking)	32	
3.3	Den Desktop starten	33
3.4	Verwalten Ihrer Software mit apt-get	35
Installieren neuer Software	36	
Entfernen von Software	38	
Ihre Software auf dem neuesten Stand halten	38	
Finden von Paketen mit apt-file	40	
3.5	Software und Medien installieren mit dem Pi Store	41
3.6	Die nächsten Schritte	42
4	Die Firmware konfigurieren	43
4.1	Aktualisieren von Firmware und Kernel	43
4.2	Konfigurieren der Videoausgabe	45
4.3	Testen und Konfigurieren des Audiosystems	48
4.4	Die nächsten Schritte	49
5	Intermezzo: Mit Pi einen Kiosk aufbauen	51
5.1	Anzeigen von Informationen aus der Twitter-Livesuche	52
5.2	Webseiten automatisch aktualisieren	55
5.3	Die nächsten Schritte	56

6	Netzwerken mit dem Pi	57
6.1	Erledigen Sie alltägliche Webaufgaben	57
6.2	Secure Shell und Pi	60
Mittels Passwort auf den Pi zugreifen	60	
Mit einem Public-Private-Schlüsselpaar auf den Pi zugreifen ..	63	
6.3	Desktops mit Pi gemeinsam nutzen	65
6.4	Machen Sie aus dem Pi einen Webserver	69
6.5	Den Pi um WLAN erweitern	74
6.6	Die nächsten Schritte	80
7	Den Pi in ein Multimediacenter verwandeln	81
7.1	Raspbmc installieren	81
7.2	Raspbmc das erste Mal starten	84
7.3	Hinzufügen von Dateien zu XBMC	86
7.4	Fernbedienung von XBMC	88
7.5	Die nächsten Schritte	91
8	Spiele auf dem Pi spielen	93
8.1	Textadventures	93
8.2	Point-and-Click-Adventures	95
8.3	Emulieren anderer Plattformen	97
8.4	Native Spiele	101
8.5	Die nächsten Schritte	102
9	Mit den GPIO-Pins herumbasteln	103
9.1	Was Sie benötigen	103
9.2	Die GPIO-Pins des Pi kennenlernen	105
9.3	Einen einfachen Schaltkreis erstellen	106
9.4	Mit den GPIO-Pins eine Leuchtdiode ansteuern	108

9.5	Einen Alarm für Speichermangel erstellen	111
9.6	Den GPIO-Status in einem Browser anzeigen	116
9.7	Wenn es nicht klappt	117
9.8	Die nächsten Schritte	118
10	Arbeiten mit Digital- und Analogsensoren	119
10.1	Was Sie benötigen	119
10.2	Bewegungserkennung mit dem Pi	121
	Den PIR-Sensor mit dem Pi verbinden	122
	Einen PIR-Sensor steuern	123
10.3	Mit dem Pi Temperaturen messen	127
	Der MCP3008	127
	SPI auf dem Pi aktivieren	129
	Jetzt spielt alles zusammen	130
	Den MCP3008 steuern	132
10.4	Wenn es nicht klappt	135
10.5	Die nächsten Schritte	135
A	Eine Einführung in Linux	137
A.1	Erste Begegnungen	138
A.2	Durch das Dateisystem navigieren	140
A.3	Textdateien bearbeiten	142
A.4	Benutzer verwalten	144
A.5	Prozesse verwalten	148
A.6	Den Pi herunterfahren und neu starten	150
A.7	Wo Sie Hilfe erhalten	150
Index		151

1 Lernen Sie den Raspberry Pi kennen

Bevor Sie den Raspberry Pi das erste Mal starten, sollten Sie sich mit seinen Anschlüssen und Fähigkeiten vertraut machen. So können Sie leichter entscheiden, für welche Projekte Sie den Pi einsetzen können, und verstehen besser, welche Zusatzhardware Sie benötigen. Sie brauchen z.B. ein Netzteil, eine Tastatur, eine Maus und einen Bildschirm. In diesem Kapitel erfahren Sie, welche Geräte am besten geeignet sind.

1.1 Die Hardware

Es ist spannend, den Pi auszupacken, wenn auch nicht so sehr wie ein neues Apple-Produkt. Normalerweise wird der Pi in einem einfachen Pappkarton zusammen mit einer oder zwei Seiten Sicherheitshinweise für elektronische Produkte und einer Schnellstartanleitung geliefert.

Die erste Version des Pi sieht nur für Bastler ansprechend aus. Es handelt sich um einen Einplatinencomputer in der Größe einer Scheckkarte. Er erinnert irgendwie an die Eingeweide vieler Elektronikgeräte, die Sie vielleicht in Ihrer Kindheit geöffnet haben. Spätere Versionen des Pi könnten ein Gehäuse besitzen, bis dahin konzentrieren wir uns allerdings auf seine inneren Werte – das ist es, was zählt, oder?

Was im Pi steckt

Der Pi ist in zwei Versionen erhältlich: Modell A und Modell B. Modell A ist etwas günstiger und bietet nicht so viele Anschlüsse wie Modell B. Im folgenden Text gehe ich kurz auf die Unterschiede ein. Da Modell A erst seit kurzer Zeit verfügbar ist, behandle ich im Rest des Buchs nur Modell B, das Sie in Abbildung 1–1 sehen.

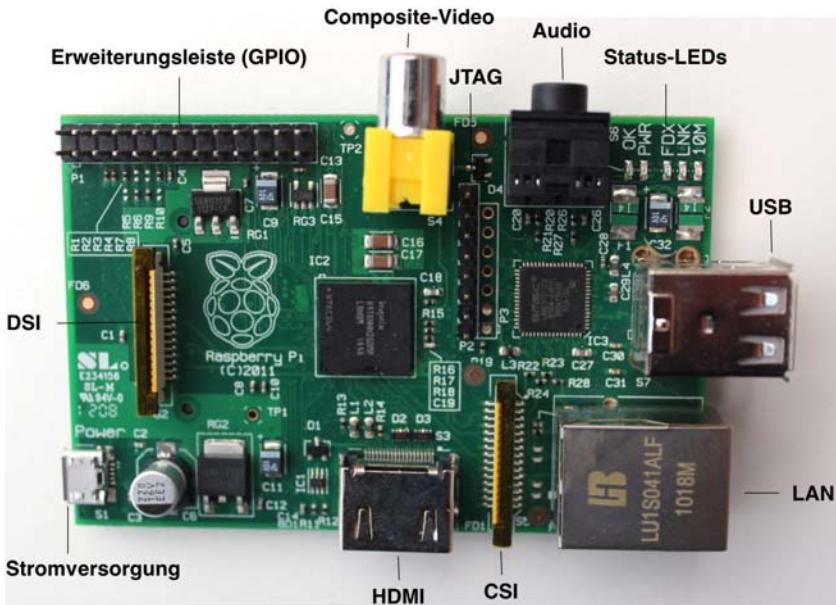


Abb. 1-1 Die Vorderseite von Modell B

Alle Raspberry Pi haben dasselbe Herz und dasselbe Gehirn: Ein Einchip-system (System on a Chip, SoC) namens BCM2835¹, das Sie in vielen Mobiltelefonen finden können. Es ist günstig, leistungsfähig und verbraucht nicht viel Strom. Diese Eigenschaften machten es zur ersten Wahl für das Raspberry-Team.

Im Gegensatz zu einer typischen PC-Architektur sind in einem SoC Prozessor (CPU), Grafikeinheit (GPU) und etwas Speicher in einer einzigen Baugruppe zusammengefasst. Der BCM2835 enthält einen ARM1176JZ-F-Prozessor mit 700 MHz, 512 MB Speicher und eine GPU namens VideoCore IV. Geräte der ersten Generation enthielten nur 256 MB Speicher.

Für Puristen ist die GPU etwas problematisch, da ihr Design und ihre Firmware proprietär sind, ihr Quellcode also nicht öffentlich erhältlich ist. Bei Ihrer täglichen Arbeit mit dem Pi werden Sie das nicht bemerken, es ist jedoch für strenge Verfechter freier Software eine ärgerliche Sache. Immerhin hat Broadcom nach langem Ringen die Software für die Trei-

1) <http://www.broadcom/products/BCM2835>

ber unter der BSD-Lizenz freigegeben. Dies ist ein absolutes Novum und nicht zuletzt das Verdienst der Raspberry-Foundation.

Der Pi hat viele Anschlüsse und die meisten dürften Ihnen vertraut vorkommen. Auf einer Modell-B-Platine finden Sie zwei normal große USB-Anschlüsse, an die Sie z.B. eine Tastatur und eine Maus anschließen können. Des Weiteren gibt es einen Micro-USB-Anschluss, der zur Stromversorgung des Pi dient und an den Sie keine weiteren Geräte anschließen können. Für den Anschluss zusätzlicher Geräte benötigen Sie einen USB-Hub. Die Modell-A-Platine besitzt nur einen einzelnen USB-Anschluss, sodass Sie fast immer einen USB-Hub verwenden müssen.

Sie können das Modell B über seinen Ethernet-Anschluss mit einem Netzwerk verbinden. Modell A hat keine Ethernet-Schnittstelle, Sie können jedoch eine mittels eines USB-zu-Ethernet-Konverters hinzufügen. Interessanterweise verwendet Modell B die interne USB-Hardware auch für das Netzwerk, sodass es keine Leistungsunterschiede zwischen Modell B und einem Modell A mit USB-zu-Ethernet-Konverter gibt.

Um den Pi mit einem Bildschirm oder einem Fernsehgerät zu verbinden, haben Sie zwei Möglichkeiten: Der Pi hat Anschlüsse für HDMI und Composite Video. Der digitale HDMI-Standard ist wesentlich leistungsfähiger als sein älterer analoger Bruder, Composite Video. Mit HDMI können Sie hochauflösende Videosignale in kristallklarer Qualität übertragen, während der Composite-Ausgang auf das beschränkt ist, was ältere Personen aus ihrer Kindheit kennen. Mit Composite Video können Sie keine hochauflösenden Grafiken darstellen und das Bild neigt meist ein wenig zum Flackern. Der größte Vorteil ist, dass Sie noch immer viele Fernsehgeräte finden, die einen Composite-Eingang haben. Allerdings gewinnt HDMI immer mehr an Einfluss. Übrigens hat das Raspberry-Team keinen VGA-Anschluss vorgesehen, weil es der Meinung ist, dass es sich bei VGA um eine auslaufende Technologie handelt. Natürlich können Sie das HDMI-Signal des Pi mit einem Adapter an einen DVI- oder VGA-Bildschirm anschließen.

Mittels HDMI können Sie außerdem Video und Ton gleichzeitig übertragen, während Sie bei Composite einen separaten Tonanschluss verwenden müssen. Dazu dient der Audioanschluss. Sie können ihn an Kopfhörer, Lautsprecher oder Ihren Receiver mit einem 3,5-mm-Klinkenstecker anschließen.

Links vom Composite-Video-Anschluss sehen Sie eine Erweiterungsleiste, die aus zwei Kontaktreihen besteht. Die meisten dieser Kontakte sind universelle Ein-/Ausgabekontakte (GPIOs), mit denen Sie den Pi an andere elektronische Geräte anschließen können. Wie Sie vielleicht aufgrund des Namens bereits vermutet haben, dienen sie keinem bestimmten Zweck; Sie können damit die unterschiedlichsten Dinge anstellen. Zum Beispiel können Sie daran Ihren guten, alten Atari VCS 2600-Joystick anschließen und Ihre 8-Bit-Lieblingsspiele in einem Emulator spielen. In Kapitel 9 »Mit den GPIO-Pins herumbasteln« lernen Sie, die Erweiterungskontakte einzusetzen, und bauen ein kleines Hardwareprojekt.

Auf der Platine befinden sich weitere Anschlüsse. Der CSI-Anschluss² dient zum Anschluss einer Kamera an den Pi (eine solche Kamera ist mittlerweile im Handel erhältlich). Mit dem DSI-Anschluss³ können Sie einen Bildschirm anschließen und die JTAG-Steckplätze⁴ helfen Ihnen, Ihre Hardwareprojekte zu debuggen.

Jedes Board hat einige Status-LEDs. Modell A verfügt über zwei LEDs (ACT bzw. PWR). Modell B besitzt fünf LEDs mit leicht unterschiedlichen Etiketten je nach Version des Boards. Auf Boards der Version 1.0 heißen die LEDs: OK, PWR, FDX, LINK und 10M. Auf Boards der Version 2.0 lauten die Bezeichnungen ACT, PWR, FDX, LNK und 100. Die LEDs haben folgende Bedeutung:

- Die OK/ACT-LED zeigt einen Zugriff auf die SD-Karte an, und bei jedem Zugriff des Pi auf die Karte blinkt sie. Allerdings können Sie die LED auch per Software steuern, das heißt, sie ist nicht sonderlich genau.
- Sobald Sie den Pi mit einem Netzteil verbinden, geht die PWR-LED an.
- Die FDX-LED zeigt an, ob das LAN im Vollduplex-Modus läuft.
- Bei jedem LAN-Zugriff blinkt die LINK/LNK-LED.
- Die 10M/100-LED zeigt an, ob die Ethernet-Schnittstelle des Pi mit 10 Mbit/s oder 100 Mbit/s läuft. Leuchtet die LED, beträgt die Geschwindigkeit 100 Mbit/s.

2) http://en.wikipedia.org/wiki/Camera_interface

3) http://en.wikipedia.org/wiki/Display_Serial_Interface

4) <http://en.wikipedia.org/wiki/Jtag>

In der folgenden Abbildung sehen Sie die Rückseite eines Pi mit dem Steckplatz für die SD-Karte rechts.

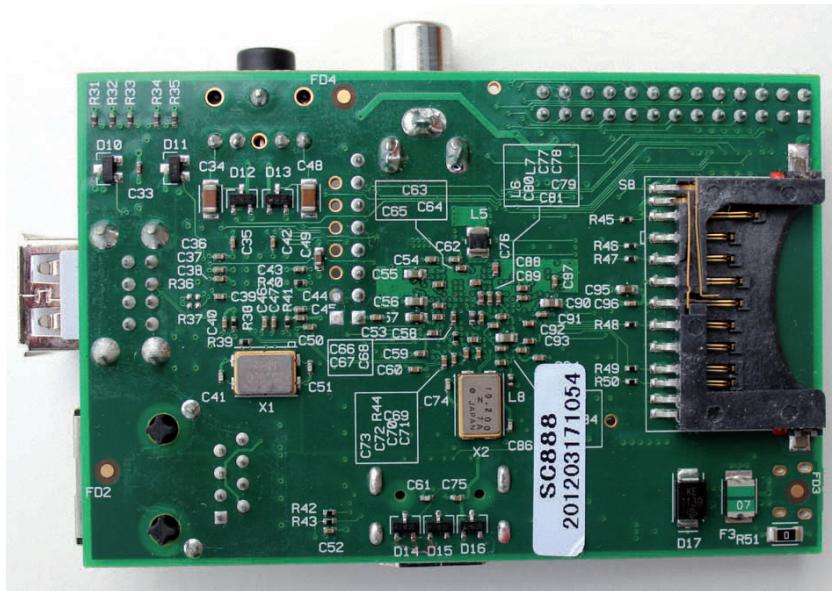


Abb. 1–2 Die Rückseite eines Pi

Der Pi besitzt keinen dauerhaften internen Speicher, sodass Sie ihn von einer SD-Karte hochfahren müssen. Sie kennen SD-Karten möglicherweise schon als beliebtes Speichermedium in Kameras, Mobiltelefonen und tragbaren Spielkonsolen. Es gibt sie in verschiedenen Abmessungen und mit unterschiedlich viel Speicher, normalerweise von 1 GB bis 64 GB (siehe auch Abb. 1–3).



Abb. 1–3 Vorder- und Rückseite einer SD-Karte

Was der Pi nicht bietet

Berücksichtigt man den günstigen Preis, bietet der Pi viele nützliche Dinge, andere fehlen jedoch. Zum Beispiel hat der Pi keine Echtzeituhr (Real-Time Clock; RTC) mit Batteriebackup und er besitzt auch kein grundlegendes Ein- und Ausgabesystem (BIOS⁵). Sie können die fehlende Uhr einfach über einen Zeitserver im Netzwerk ersetzen, und die meisten Betriebssysteme machen das automatisch, aber das fehlende BIOS führt zu ein paar Einschränkungen.

Einfach gesagt ist das BIOS ein im Nur-Lese-Speicher (ROM) abgelegtes Programm, das auf einem PC beim Systemstart ausgeführt wird. Unter anderem dient es dazu, neue Geräte zu konfigurieren und die Startreihenfolge festzulegen. Zum Beispiel können Sie mit dem BIOS festlegen, ob Sie von der Festplatte oder von einer DVD starten. Der Pi besitzt kein BIOS und startet daher immer von einer SD-Karte. Auch wenn Sie ein perfekt installiertes Betriebssystem auf einem USB-Stick oder einer externen Festplatte haben, können Sie es nicht starten. Natürlich können Sie auf externe Speichergeräte zugreifen, nur starten kann der Pi nicht davon.

Der Pi unterstützt von Haus aus kein Bluetooth oder WLAN, aber Sie können beides über USB-Geräte realisieren. Leider sind die meisten Linux-Distributionen noch immer wählerisch, was die Hardware angeht. Daher sollten Sie zuerst prüfen, ob Ihre Linux-Version Ihr Gerät unterstützt. (Lesen Sie auch die Hinweise, wo Sie kompatible Hardware erhalten.) Dies gilt auch für andere Hardware, wie Mikrofone und Webcams. Solange Ihr Betriebssystem und Ihre Anwendungen Ihre Geräte unterstützen, ist alles gut. Wenn nicht, halten Sie lieber nach einer Alternative Ausschau, die vom Betriebssystem unterstützt wird.

Sie wissen jetzt, wozu all die Anschlüsse am Pi dienen. Im nächsten Abschnitt lernen Sie, welche Geräte Sie an den Pi anschließen können.

5) <http://en.wikipedia.org/wiki/BIOS>

1.2 Was Sie sonst noch benötigen

Wenn Sie das erste Mal einen Pi auspacken, erkennen Sie schnell, dass das Raspberry-Team dem Motto »BYOP«⁶ folgt. Die Schachtel enthält nur die Platine. Sie benötigen aber noch ein paar Dinge mehr, um sie richtig einsetzen zu können. Die meisten davon haben Sie vermutlich schon im Hause.

Auswählen eines Netzteils

Zuerst benötigen Sie ein Netzteil mit Micro-USB-Stecker, denn momentan wird mit dem Pi keines mitgeliefert. Entsprechend der Pi-Spezifikation erfordern beide Modelle ein Netzteil mit 5 V Ausgangsspannung. Das Netzteil sollte 300 mA für Modell A und 700 mA für Modell B liefern. Abhängig von den angeschlossenen Geräten kann die benötigte Leistung auch höher liegen.

Viele Ladegeräte für Mobiltelefone entsprechen den Anforderungen des Pi, und das ist kein Zufall. Das Raspberry-Team wollte diese Kompatibilität, da solche Ladegeräte nahezu überall verfügbar sind. Ich habe das Ladegerät des Samsung Galaxy S II ein paar Tage verwendet, und meine ersten Experimente haben damit prima funktioniert. Für den Anschluss weiterer Geräte reichte die Leistung aber nicht mehr und ich habe es durch ein Ladegerät von Belkin ersetzt (siehe Abb. 1–4). Es hat eine Leistung von 1 A und funktioniert besser, ist aber für manche Hardwareprojekte noch immer nicht ausreichend.

Die größte Einschränkung des Pi bei der Stromversorgung besteht darin, dass kein externes USB-Gerät mehr als 100 mA verbrauchen darf. Solange also Ihre Tastatur und Ihre Maus jeweils bis zu 100 mA verbrauchen, ist alles in Ordnung. Normalerweise finden Sie auf der Unterseite der Geräte einen kleinen Aufkleber mit dem Stromverbrauch. Wenn ein externes Gerät mehr als 100 mA benötigt, werden Sie früher oder später seltsame Effekte erleben⁷. Auf der sicheren Seite sind Sie mit einem Netzteil, das für Modell B 1 A bis 1,2 A liefert. Für Modell A sollten es zwischen 500 mA und 700 mA sein.

6) Bring Your Own Peripherals (Bringen Sie eigene Peripheriegeräte mit).

7) http://elinux.org/RPi_Hardware#Power

Sie können den Pi durch einen USB-Hub mit eigener Stromversorgung entlasten, das funktioniert aber nicht mit jedem Hub. Bevor Sie also etwas für Ihren Pi kaufen, sollten Sie einen Blick in das Projekt-Wiki⁸ werfen.



Abb. 1–4 Ein USB-Ladegerät

Auswählen einer SD-Karte

Auch mit der perfekten Stromversorgung tut der Pi nicht viel, wenn Sie ihn starten, da er eine SD-Karte mit einem Betriebssystem benötigt. Sie können dazu vorbereitete SD-Karten kaufen⁹, aber Sie können auch eine leere Karte verwenden und sie selbst vorbereiten (lesen Sie dazu Abschnitt 2.2 »Vorbereiten einer bootfähigen SD-Karte«). Normalerweise ist dies die bessere Wahl, weil Sie so die neueste und beste Software für Ihren Pi bekommen. Als dieses Buch geschrieben wurde, basierten einige fertige SD-Karten noch auf Debian »squeeze«, das schon durch Debian »wheezy« (Raspbian) abgelöst war. Auch Raspbian selbst wird kontinuierlich verbessert und unterstützt zum Beispiel seit kurzer Zeit die neuen Pi-Modelle mit 512 MB RAM.

8) http://elinux.org/RPi_VerifiedPeripherals

9) <http://uk.farnell.com/raspberry-pi-accessories#sdcards>

Einige Anwender berichten über inkompatible SD-Karten. Wenn Sie Zweifel haben, sollten Sie einen Blick auf den Abschnitt »Wo bekomme ich einen Raspberry Pi und Zusatzhardware« werfen. Prinzipiell können Sie Karten jeder Größe einsetzen. Natürlich hängt die Mindestgröße von Ihrem Betriebssystem, den Anwendungen, die Sie installieren wollen, und den Daten ab, die auf dem Pi später erzeugt werden. Wie so oft im Leben ist größer meist besser und Sie sollten für einen bequemen Einsatz des Pi mindestens 4 GB verwenden.

Anschließen von Tastatur und Maus

Sofern Sie nicht planen, den Pi als pures Serversystem¹⁰ einzusetzen, benötigen Sie eine Tastatur und eine Maus. Wahrscheinlich haben Sie zu Hause eine Ersatztastatur und eine zweite Maus, und solange diese einen USB-Anschluss besitzen, funktionieren sie vermutlich auch mit dem Pi. Manchmal verursachen Tastaturen mit integriertem USB-Hub Probleme, da sie dem Pi Strom wegnehmen, der anderweitig benötigt wird. Wenn Sie eigenartige Effekte feststellen, wie eine nicht reagierende Tastatur oder sich unendlich wiederholende Zeichen, probieren Sie eine andere Tastatur aus oder schließen Sie sie über einen USB-Hub mit Stromversorgung an. Im Idealfall verbrauchen Ihre Tastatur und Maus jeweils maximal 100 mA.

Einige drahtlose Tastaturen und Mäuse könnten ebenfalls nicht funktionieren, da Linux nicht alle unterstützt. Zu Anfang sollten Sie einfach konservativ sein und drahtgebundene Geräte verwenden, bis alles richtig funktioniert. Tauschen Sie anschließend die Komponenten nacheinander aus und wenn ein Problem auftritt, prüfen Sie, ob Ihr Betriebssystem die entsprechende Tastatur oder Maus unterstützt.

Häufig werden Sie mehr als zwei USB-Geräte (oder eines beim Modell A) benötigen, weshalb Sie sie mit einem USB-Hub an den Pi anschließen müssen. Achten Sie darauf, dass der Hub alle angeschlossenen Geräte mit genug Energie versorgen kann. In beinahe allen Fällen benötigen Sie einen Hub mit eigener Stromversorgung.

10) http://en.wikipedia.org/wiki/Headless_system

Auswählen eines Bildschirms

Abhängig vom verwendeten Bildschirm benötigen Sie ein HDMI- oder ein Composite-Video-Kabel. Wenn Sie HDMI verwenden und Ihr Bildschirm auch Lautsprecher hat, ist alles geritzt. Wenn nicht, müssen Sie den Audioausgang des Pi durch ein Kabel mit 3,5-mm-Klinkensteckern an das Soundsystem anschließen. Diesen Stecker finden Sie auch in den Kopfhörern Ihres iPod, die Sie natürlich auch verwenden können.

Auswählen des richtigen Netzwerk-Equipments

Wenn Sie ein Modell B an ein Netzwerk anschließen möchten, benötigen Sie nur ein Ethernet-Kabel. Modell A besitzt keine eigene Ethernet-Schnittstelle, sodass Sie einen USB-zu-Ethernet-Konverter einsetzen müssen.

Hinzufügen eines Gehäuses

Zukünftige Versionen des Pi werden vielleicht mit einem Gehäuse geliefert. Bis dahin müssen wir uns jedoch selbst helfen. Wie jedes elektronische Gerät ist auch der Pi empfindlich gegenüber Staub und leitenden Oberflächen, sodass früher oder später ein Gehäuse fällig ist.

Die Pi-Community ist sehr kreativ, und es haben schon viele Personen Gehäuse aus Lego-Steinen gebaut¹¹ und sogar aus Pappe¹². Eines der größeren Probleme mit selbst gebauten Gehäusen ist, dass sie keinen bequemen Zugang zu den Erweiterungsanschlüssen des Pi bieten. Daher besteht die beste Lösung im Kauf eines professionellen Gehäuses, z.B. bei Adafruit¹³ oder bei ModMyPi¹⁴.

Zusätzlich zu den erwähnten Geräten benötigen Sie einen PC, um z.B. ein Datenträgerabbild auf eine SD-Karte zu kopieren oder Anwendungen zu komplizieren. Einen Pi einzurichten ist also nicht ganz so kostengünstig, wie es zu Anfang schien.

11) <http://www.raspberrypi.org/archives/1515>

12) http://squareitround.co.uk/Resources/Punnet_net_Alpha3.pdf

13) <https://www.adafruit.com/products/859>

14) <http://modmypi.com>

Ein typisches Pi-System sieht auf Ihrem Schreibtisch ziemlich unordentlich aus, wenn alle Kabel angeschlossen sind (siehe Abb. 1–5). Trotz des ungewohnten Aussehens ist die Hardware jetzt zu einem ersten Testlauf bereit.



Abb. 1–5 Pi-System mit angeschlossenen Kabeln

1.3 Die nächsten Schritte

In diesem Kapitel haben Sie gelernt, wozu all die Anschlüsse des Pi dienen, welche Zusatzgeräte benötigt werden und wie Sie die richtigen auswählen. Im Prinzip können Sie den Pi jetzt das erste Mal starten. Ohne ein Betriebssystem passiert jedoch nicht viel. Im folgenden Kapitel erfahren Sie, welche Möglichkeiten Sie haben und wie Sie ein vollständiges Linux-System installieren.

2 Ein Betriebssystem installieren

Wie jeder andere Computer auch benötigt der Raspberry Pi ein Betriebssystem, und zwar vorzugsweise Linux. Zum einen deshalb, weil Linux freie Software ist, aber hauptsächlich, weil der ARM-Prozessor des Pi von Linux unterstützt wird, während die meisten anderen Betriebssysteme nur auf Intel-Architekturen laufen. Trotzdem läuft nicht jede Linux-Distribution auf dem Pi, da nicht jede den besonderen ARM-Prozessortyp des Pi unterstützt. Zum Beispiel können Sie auf dem Pi kein Ubuntu Linux installieren. In diesem Kapitel lernen Sie daher zuerst, welche Auswahlmöglichkeiten Sie haben.

Die Auswahl eines Betriebssystems ist nur ein erster Schritt, denn Sie müssen es auch installieren. Der Installationsvorgang ist deutlich anders als das, was Sie vermutlich gewöhnt sind, jedoch nicht sehr schwierig: Sie müssen das Betriebssystem auf einer SD-Karte installieren. In diesem Kapitel werden Sie die neueste Linux-Distribution von Debian installieren, die Vorgehensweise ist jedoch für alle Betriebssysteme gleich. Sie können auch mehrere SD-Karten mit unterschiedlichen Betriebssystemen erstellen, sodass Sie am Ende ein universelles System besitzen, das Sie durch einen einfachen Kartenwechsel in unterschiedlichste Maschinen verwandeln können.

2.1 Eine Auswahl treffen

Linux ist noch immer die beliebteste Wahl für den Pi und hilft Ihnen dabei, das Maximum aus dem Gerät herauszuholen. Außerdem sind viele mit Linux vertraut, während andere vom Pi unterstützte Betriebssysteme schon etwas exotisch sein können.

Auch wenn eine Linux-Distribution auf dem Pi läuft, sieht sie oft anders aus und verhält sich anders als das »reguläre« Desktop-Betriebssystem. Außerdem werden Sie nicht alle gewohnten Anwendungen vorfinden, z.B. nicht alle Webbrowser und Office-Suiten.

Zusätzlich gibt es einige Einschränkungen bei der Installation. Moderne Betriebssysteme sind relativ umfangreich und werden als DVD oder als ISO-Abbild zum Download angeboten. Diese Datenträgerabbilder und DVDs enthalten eine Komplettinstallation des Betriebssystems. Sie starten ein Programm, das die Hardware Ihres Computers erkennt, und kopieren alle benötigten Dateien auf die Festplatte. Leider können Sie in den Pi keine DVD einlegen und installieren, und da der Pi kein BIOS hat (siehe Abschnitt »Was der Pi nicht bietet« auf Seite 6), kann er auch nicht von einem externen USB-Laufwerk gestartet werden. Sie können auch kein ISO-Image einer DVD auf eine SD-Karte kopieren. Stattdessen benötigen Sie einen Schnappschuss eines bereits installierten Systems, von dem Sie starten können.

Wir müssen also ein Image einer Linux-Distribution finden oder erzeugen, das wir auf eine SD-Karte kopieren können und das mit dem Pi kompatibel ist. Am einfachsten bekommen wir ein solches Image auf der Downloadseite des Raspberry-Projekts¹. Als dieses Buch geschrieben wurde, standen Images für Raspbian (Debian Wheezy), Arch Linux ARM und RISC OS zur Verfügung. Weitere Betriebssysteme werden in Zukunft folgen, zumindest Bodhi Linux² steht schon heute zur Verfügung. Einige clevere Bastler versuchen momentan auch, Google Chrome OS³ auf den Pi zu portieren.

Momentan ist Raspbian (Debian Wheezy) die beste Wahl für die ersten Schritte auf dem Pi. Es unterstützt die Pi-Hardware vollständig, bietet einen kompletten Desktop (siehe Abb. 2–1) und enthält einige nützliche Anwendungen, wie zum Beispiel einen Webbrowser.

1) <http://www.raspberrypi.org/downloads>

2) <http://jeffboogland.blogspot.co.uk/2012/06/bodhi-linux-arm-alpha-release-for.html>

3) <http://www.cnx-software.com/2012/04/19/building-chromium-os-for-raspberry-pi-armv6/>



Abb. 2-1 Desktop in Raspbian (Debian Wheezy)

Zusätzlich bietet es einen leistungsfähigen Paketmanager, der die Installation weiterer Software sehr leicht macht. Wir werden für den Rest dieses Buchs Debian verwenden und im folgenden Abschnitt lernen Sie, wie man es installiert. Beachten Sie, dass wir die Namen Raspbian und Debian gleichberechtigt verwenden und dass die Distribution Soft-float Debian »wheezy« weitestgehend mit Raspbian übereinstimmt. Sie nutzt aber nicht die hardwarebeschleunigten Fließkommaoperationen und wird lediglich aus Kompatibilitätsgründen angeboten.

Die anderen Distributionen sind ebenfalls sehr interessant, wenden sich jedoch an einen anderen Anwenderkreis. Daher beschreibe ich sie nur kurz in den folgenden Abschnitten.

Arch Linux ARM

Arch Linux⁴ ist sehr minimalistisch und setzt größere Vorkenntnisse in Linux voraus. Arch Linux verbraucht nicht viel Systemleistung, verfügt ebenfalls über einen bequemen Paketmanager und ist eine gute Wahl, wenn der Pi als Server dienen soll. Für ein Desktop-System ist Debian jedoch bequemer, schon deshalb, weil Arch Linux ohne Desktop-Umgebung geliefert wird. Diese müssen Sie selbst installieren und konfigurieren.

4) <http://www.archlinux.org/>

RISC OS

Der Pi unterstützt nicht nur Linux, sondern zum Beispiel auch RISC OS⁵. Das ist keine Überraschung, da RISC OS eines der ersten Betriebssysteme für die ARM-Architektur war. Es hat noch immer viele Fans und ist definitiv einen Blick wert.

Darüber hinaus sind Spezialdistributionen für bestimmte Anwendungsfälle in der Linux-Welt sehr verbreitet. In Kapitel 7 »Den Pi in ein Multimediacenter verwandeln« lernen Sie Raspbmc kennen, eine Linux-Distribution, die aus Ihrem Pi ein Multimediacenter macht.

Auch wenn sich die Pi-Hardware nicht so einfach verändern lässt, können Sie den Pi mit nur einem Handgriff in ein gänzlich anderes Gerät verwandeln: Legen Sie einfach eine SD-Karte mit einem anderen Betriebssystem ein. Im nächsten Abschnitt erfahren Sie, wie Sie eine solche SD-Karte vorbereiten.

2.2 Vorbereiten einer bootfähigen SD-Karte

Wie wir in Kapitel 1 »Lernen Sie den Raspberry Pi kennen« gesehen haben, bietet der Pi kein BIOS und keinen internen Permanentenspeicher. Er hat nur einen SD-Kartenslot. Sie müssen das Betriebssystem des Pi auf einem separaten Computer auf einer SD-Karte vorbereiten, mit der Sie den Pi dann starten. Das haben einige Tüftler erfreulicherweise schon für verschiedene Betriebssysteme erledigt und bieten ihre Arbeit kostenfrei im Internet als Download für SD-Karten an. In diesem Kapitel lernen Sie, wie Sie ein SD-Kartenimage auf eine SD-Karte übertragen.

Sie benötigen einen PC mit Kartenleser (was eigentlich keine korrekte Bezeichnung ist, da Sie ihn auch zum Beschreiben der Karte verwenden können), um die SD-Karte zu modifizieren. Manche PCs haben integrierte Leser, es gibt aber auch für wenige Euro welche mit USB-Anschluss. Das verwendete Betriebssystem ist prinzipiell nicht von Bedeutung und wir spielen die Erstellung der SD-Karte auf allen wichtigen Plattformen durch. Wenn Sie einen Windows-PC benutzen können, rate ich Ihnen dringend dazu, weil der Vorgang einfacher und bequemer

5) http://de.wikipedia.org/wiki/RISC_OS

ist als unter Mac OS X oder Linux. Eine SD-Karte unter Mac OS X oder Linux einzurichten ist zwar keine Hexerei, Sie müssen dazu aber einen ziemlich gefährlichen Befehl verwenden und könnten dabei leicht wichtige Dateien löschen. Unter Windows erhalten Sie außerdem mehr Rückmeldungen, wenn Sie das Kartenimage kopieren. Ein besserer Raspbian-Installer ist in Arbeit, bis er jedoch läuft, müssen Sie das Kartenimage selbst kopieren.

Ganz gleich, welches Betriebssystem Sie für den Installationsvorgang verwenden, zuerst müssen Sie das Debian-Image von der offiziellen Downloadseite herunterladen⁶, entweder mittels HTTP oder via Torrent. Wenn der Download abgeschlossen ist, sollte sich eine Datei namens 2012-12-16-wheezy-raspbian.zip auf Ihrer lokalen Festplatte befinden (der Dateiname könnte auch anders lauten, wenn es sich um eine neuere Version handelt).

Der in den folgenden Abschnitten beschriebene Vorgang ist für Images aller kompatiblen Pi-Betriebssysteme gleich. Nur der Name der Image-datei ist unterschiedlich.

Vorbereiten einer SD-Karte unter Windows

Eine SD-Karte unter Windows vorzubereiten ist der komfortabelste Weg, und zwar wegen Win32DiskImager⁷. Diese kleine Anwendung ist frei erhältlich, besitzt eine nette Benutzerschnittstelle und dient einem einzigen Zweck: Images auf SD-Karten zu schreiben. Sie müssen sie nicht einmal installieren. Es ist ausreichend, die Zip-Datei von der Projektwebsite herunterzuladen und sie in ein beliebiges Verzeichnis zu extrahieren. Klicken Sie doppelt auf Win32DiskImager.exe und Sie können loslegen.

6) <http://www.raspberrypi.org/downloads>

7) <http://www.softpedia.com/get/CD-DVD-Tools/Data-CD-DVD-Burning/Win32-Disk-Imager.shtml>

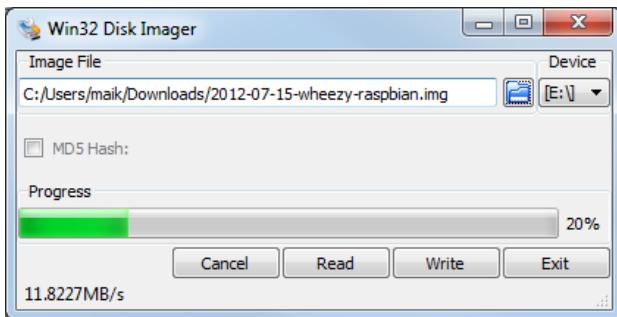


Abb. 2–2 Win32DiskImager bei der Arbeit

Bevor Sie das SD-Kartenimage auf die SD-Karte schreiben, können Sie es optional auf Manipulationen prüfen. Dazu berechnen Sie die SHA1-Prüfsumme der Zip-Datei mithilfe des Befehls `fciv`, den Sie zuvor installieren müssen. Auf der Supportseite von Microsoft finden Sie viele Informationen dazu.⁸

Wenn Sie `fciv` installiert haben, können Sie den Befehl wie folgt einsetzen:

```
C:>fciv 2012-12-16-wheezy-raspbian.zip -sha1
// File Checksum Integrity Verifier version 2.05.
//
514974a5fcbbbea02151d79a715741c2159d4b0a 2012-12-16-wheezy-
raspbian.zip
```

Wenn die lange Hexadezimalzahl derjenigen auf der Downloadseite entspricht, wurde die Zip-Datei nicht manipuliert und Sie können sicher fortfahren. Andernfalls laden Sie die Datei von einem anderen Server herunter.

Nachdem die Anwendung gestartet wurde, wählen Sie das Debian-Image und den Laufwerksbuchstaben Ihres Kartenlesers aus. Seien Sie vorsichtig und vergewissern Sie sich, dass Sie keinen anderen Laufwerksbuchstaben verwenden. Sie riskieren sonst den Verlust wichtiger Daten. Klicken Sie dann auf die Schaltfläche »Write« und Sie sehen etwas wie in Abbildung 2–2. Es dauert ein paar Minuten, bis das Image geschrieben ist, dann haben Sie jedoch eine SD-Karte, mit der Sie den Pi starten können.

8) <http://support.microsoft.com/kb/841290>

Vorbereiten einer SD-Karte unter Linux

Eine SD-Karte für den Pi unter einem modernen Linux zu installieren ist nicht allzu schwierig, Sie sollten bei den folgenden Schritten jedoch vorsichtig sein, damit Sie keine wichtigen Daten löschen. Legen Sie die SD-Karte jetzt noch nicht in den Kartenleser ein. Dies erfolgt in einem späteren Schritt, um den Gerätenamen des Lesers zu ermitteln.

Laden Sie die Zip-Datei mit dem Debian-Image von der offiziellen Downloadseite herunter, öffnen Sie ein Terminalfenster und wechseln Sie in das Verzeichnis, in das Sie die Zip-Datei gerade heruntergeladen haben. Auch wenn es nicht zwingend notwendig ist, schadet es nicht, die Integrität der heruntergeladenen Datei zu prüfen.

```
maik> shasum 2012-12-16-wheezy-raspbian.zip
514974a5fcbbbea02151d79a715741c2159d4b0a 2012-12-16-wheezy-
raspbian.zip
```

Wenn die lange Hexadezimalzahl derjenigen auf der Downloadseite entspricht, wurde die Zip-Datei nicht manipuliert und Sie können sicher fortfahren. Ansonsten laden Sie die Datei von einem anderen Server herunter.

Der folgende Befehl entpackt die Imagedatei in das aktuelle Verzeichnis.

```
maik> unzip 2012-12-16-wheezy-raspbian.zip
Archive: 2012-12-16-wheezy-raspbian.zip
      inflating: 2012-12-16-wheezy-raspbian.img
```

Nun müssen Sie Ihren Kartenleser finden. Führen Sie den folgenden Befehl aus, um eine Liste aller momentan an Ihren Computer angeschlossenen Speichergeräte zu erhalten.

```
maik> df -h
Filesystem  Size  Used   Avail Use% Mounted on
/dev/sda1    63G   15G    46G  24%   /
udev        494M  4.0K   494M  1%   /dev
tmpfs       201M  740K   200M  1%   /run
none         5.0M   0     5.0M  0%   /run/lock
none         501M  124K   501M  1%   /run/shm
```

Legen Sie die SD-Karte ein und führen Sie den Befehl erneut aus.

```
maik> df -h
Filesystem  Size  Used   Avail   Use%  Mounted on
/dev/sda1    63G   15G    46G    24%   /
udev        494M  4.0K   494M    1%   /dev
tmpfs       201M  772K   200M    1%   /run
none         5.0M   0     5.0M    0%   /run/lock
none         501M  124K   501M    1%   /run/shm
/dev/sdc2    1.6G  1.2G   298M   81%   /media/18c27e44-ad29-4264-
                           9506-c93bb7083f47
/dev/sdc1    75M   29M    47M   39%   /media/95F5-0D7A
```

Wie Sie auf meinem System erkennen können, hat die SD-Karte den Namen *sdc* und sie besteht aus zwei Partitionen namens *sdc1* und *sdc2*. Das kann auf Ihrem System abweichen, Sie könnten mehr oder weniger Partitionen haben und die Karte könnte beispielsweise *sdd* heißen. Bevor Sie weitermachen, müssen Sie alle Partitionen abmelden, was in diesem Fall mit den folgenden Befehlen geschieht:

```
maik> umount /dev/sdc1
maik> umount /dev/sdc2
```

Als letzten Schritt müssen Sie das Image auf die SD-Karte kopieren. Den folgenden Befehl müssen Sie mit Root-Berechtigung ausführen und sicher sein, dass Sie für die Option *of* den richtigen Gerätenamen angeben.

```
maik> sudo dd bs=1M if=2012-12-16-wheezy-raspbian.img of=/dev/sdc
[sudo] password for maik:
1850+1 records in
1850+1 records out
1939865600 bytes (1.9 GB) copied, 160.427 s, 12.1 MB/s
```

Der Kopiervorgang dauert einige Minuten, aber wenn alles klappt, erhalten Sie eine startfähige SD-Karte, die Debian auf Ihren Pi bringt.

Vorbereiten einer SD-Karte unter Mac OS X

Eine SD-Karte mit Debian auf dem Mac zu erstellen ähnelt dem Vorgang unter Linux, es gibt jedoch einige wichtige Unterschiede. Sie benötigen nur wenige Befehle, müssen sich dabei aber konzentrieren.

Legen Sie die SD-Karte jetzt noch nicht in den Kartenleser ein. Dies erfolgt in einem späteren Schritt, um den Gerätenamen des Lesers zu ermitteln. Laden Sie die neueste Zip-Datei mit dem Debian-Image von der offiziellen Downloadseite herunter. Öffnen Sie ein Terminalfenster und wechseln Sie zu dem Ordner, in dem Sie die Zip-Datei gespeichert haben. Erzeugen Sie jetzt mit dem folgenden Befehl eine Prüfsumme (dieser Schritt ist optional, wenn Sie Ihrer Downloadquelle vertrauen oder die Zip-Datei aus einer anderen vertrauenswürdigen Quelle stammt):

```
maik> shasum 2012-12-16-wheezy-raspbian.zip  
514974a5fcbbbea02151d79a715741c2159d4b0a  
2012-12-16-wheezy-raspbian.zip
```

Wenn die Hexadezimalzahl nicht mit der auf der Downloadseite übereinstimmt, könnte die Zip-Datei manipuliert worden sein und Sie sollten sie von einem anderen Server herunterladen. Ansonsten machen Sie einfach weiter. Entpacken Sie die Datei in das aktuelle Verzeichnis.

```
maik> unzip 2012-12-16-wheezy-raspbian.zip  
Archive: 2012-12-16-wheezy-raspbian.zip  
  inflating: 2012-12-16-wheezy-raspbian.img
```

Nun müssen Sie Ihren Kartenleser finden. Geben Sie folgenden Befehl ein, um alle momentan auf dem Mac zur Verfügung stehenden Dateisysteme anzeigen zu lassen:

```
maik> df -h  
Filesystem      Size  Used  Avail Capacity Mounted on  
/dev/disk0s2    465Gi 383Gi  81Gi   83%   /  
devfs          189Ki 189Ki   0Bi   100%   /dev  
map -hosts      0Bi   0Bi   0Bi   100%   /net  
map auto_home   0Bi   0Bi   0Bi   100%   /home  
/dev/disk1s1    466Gi 460Gi  6.1Gi  99%   /Volumes/macback
```

Die Ausgabe auf Ihrem System wird davon abweichen, wir benötigen sie aber nur, um die SD-Karte zu finden. Legen Sie die Karte jetzt in den Kartenleser ein und geben Sie den Befehl nach ein paar Sekunden erneut ein.

```
maik> df -h
Filesystem      Size   Used  Avail Capacity Mounted on
/dev/disk0s2    465Gi  383Gi  81Gi   83%   /
devfs          191Ki  191Ki  0Bi    100%   /dev
map -hosts      0Bi   0Bi   0Bi    100%   /net
map auto_home   0Bi   0Bi   0Bi    100%   /home
/dev/disk1s1    466Gi  460Gi  6.1Gi  99%   /Volumes/macback
/dev/disk3s1    15Gi   1.1Mi  15Gi   1%    /Volumes/SD
```

Wie Sie bei meinem Mac sehen, wird die SD-Karte unter `/dev/disk3s1` gefunden. Auf Ihrem Mac kann das auch an einer anderen Stelle sein, und die Karte muss auch nicht unter `/Volumes/SD` stehen. Im folgenden Befehl ersetzen Sie `/dev/disk3s1` durch die Position Ihrer SD-Karte:

```
maik> diskutil unmount /dev/disk3s1
Volume SD on disk3s1 unmounted
```

Dadurch wird das Dateisystem der SD-Karte abgemeldet und wir können schließlich das Debian-Image darauf kopieren. Für diesen Vorgang benötigen Sie den Gerätenamen der SD-Karte. Sie können ihn aus der Position der SD-Karte ableiten, indem Sie `s1` löschen und vor `disk` ein `r` einfügen. Auf meinem System ist das `/dev/rdisk3`.

Warnung: Der folgende Befehl kopiert das Debian-Image auf das Gerät, das Sie mit der Option `of` angeben. Wenn Sie das falsche Gerät angeben, z.B. die interne Festplatte Ihres Mac oder ein externes USB-Laufwerk mit Ihren schönsten Fotos, werden diese Daten gelöscht. Wenn Sie absolut sicher sind, das richtige Ziel angegeben zu haben, führen Sie folgenden Befehl aus:

```
maik> sudo dd if=2012-12-16-wheezy-raspbian.img of=/dev/rdisk3 bs=1m
1850+0 records in
1850+0 records out
1939865600 bytes transferred in 150.830724 secs (12861210 bytes/sec)
```

Der Befehl läuft ohne weitere Statusmeldungen ab. Wie Sie anhand der vorherigen Ausgabe erkennen können, dauerte der Kopiervorgang mehr als zwei Minuten, haben Sie also etwas Geduld.

Wenn Sie eine SD-Karte vorbereiten, müssen Sie auf eine Sache aufpassen: Manche Anwender haben auf aktuellen MacBooks und MacBook Pro mit deren internen Kartenlesern Erfahrungen mit Lese/Schreibfehlern

oder nicht erkennbaren SDHC-Karten gemacht. Ein externer Kartenleser hat aber diese Probleme behoben.

Nun können Sie die Karte auswerfen.

```
maik> diskutil eject /dev/rdisk3
Disk /dev/rdisk3 ejected
```

Das war's! Sie haben jetzt eine startfähige SD-Karte mit Debian auf Ihrem Mac erzeugt.

2.3 Die nächsten Schritte

Egal welches Betriebssystem Sie verwendet haben, Sie haben jetzt eine startfähige SD-Karte mit Debian Linux. Außerdem wissen Sie jetzt, wie Sie jedes beliebige mit dem Pi kompatible Betriebssystem auf eine startfähige SD-Karte übertragen. Im nächsten Kapitel lernen Sie, wie Debian zum ersten Mal auf dem Pi gestartet wird.

3 Raspbian konfigurieren

Kein Betriebssystem und keine Hardware erfüllen alle Bedürfnisse des Anwenders von vornherein. Das gilt insbesondere für die Debian-Version, die auf dem Pi läuft, da sie als Image geliefert wird und wir daher nicht alle Konfigurationsparameter einstellen können, die wir bei einer normalen Installation setzen würden. Zum Beispiel hat das Image fest eingestellte Tastaturbelegungen und Ländercodes. In diesem Kapitel starten wir den Pi erstmalig und sehen uns das Ganze mal an. Sie lernen, wie Sie viele Grundeinstellungen wie Passwort und Zeitzone vornehmen.

3.1 Den Pi zum ersten Mal starten

Es ist wichtig, die Hardware vorzubereiten und ein Betriebssystem zu installieren, viel mehr Spaß macht es jedoch, den Pi zu starten und zu sehen, was er kann. Legen Sie also die im letzten Kapitel vorbereitete SD-Karte ein und schließen Sie das Netzteil an.

Wenn Sie schon vorher mit Linux gearbeitet haben, kennen Sie die meisten Meldungen, die über den Bildschirm laufen. Das ist keine Überraschung, denn auch wenn der Pi ein ungewöhnlicher Computer ist, bleibt Raspbian eine normale Linux-Distribution.

Wenn Sie Raspbian das erste Mal starten, wird ein Konfigurationsprogramm namens Raspi-config ausgeführt. Es hilft Ihnen dabei, die wichtigsten Einstellungen des Linux-Systems vorzunehmen. Sie sehen das Hauptmenü in Abbildung 3–1.

Sie sind vermutlich gewöhnt, die Benutzerschnittstelle mit der Maus zu bedienen, Raspi-config müssen Sie jedoch mit der Tastatur steuern. Verwenden Sie die Pfeil-unten-Taste, um zum nächsten Menüpunkt zu gelangen und die Pfeil-oben-Taste, um zum vorhergehenden zurückzugelan-

gen. Um einen Menüpunkt zu wählen, drücken Sie die Tabulator- oder Pfeil-rechts-Taste. Dadurch wird die *Select*-Schaltfläche unten aktiviert. Drücken Sie jetzt die Leer- oder Returntaste, um den Menüpunkt zu aktivieren.

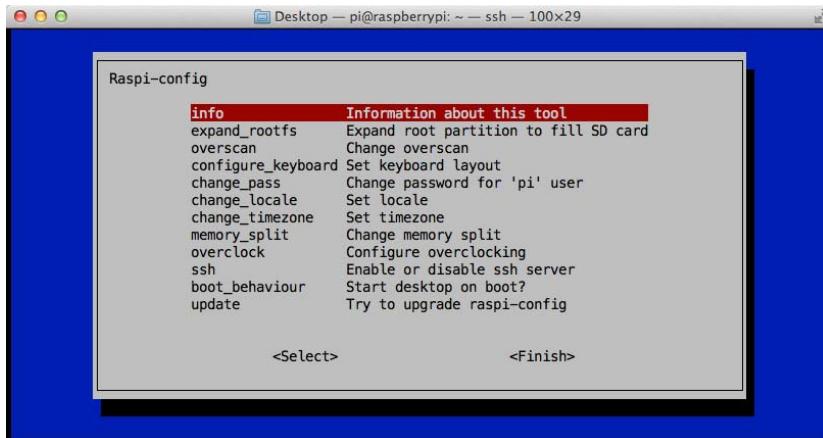


Abb. 3-1 Raspi-config macht die Konfigurationseinstellungen zum Kinderspiel.

Um mit Raspi-config vertraut zu werden, wählen Sie zuerst den Menüpunkt *Info*. Dadurch öffnet sich ein Fenster, das kurz den Zweck von Raspi-config erläutert. Klicken Sie auf die Schaltfläche *Ok* und drücken Sie die Leertaste, um zum Hauptmenü zurückzukehren.

Die meisten Menüs in Raspi-config haben eine Schaltfläche zum Abbrechen. Um den aktuellen Vorgang abzubrechen, drücken Sie die Tabulator-taste, bis *Cancel* hervorgehoben wird, und dann die Leer- oder Return-taste.

Das Hauptmenü bietet die Schaltfläche *Finish*, mit der Sie Raspi-config verlassen. Ein Großteil der Änderungen in Raspi-config erfordert einen Neustart des Pi. Wenn Sie also die Schaltfläche *Finish* in Raspi-config aktivieren, werden Sie gegebenenfalls gefragt, ob Sie neu starten möchten.

Raspi-config wird beim nächsten Start des Pi nicht automatisch ausgeführt. Machen Sie sich darüber aber keine Sorgen. Sie können es im Terminal immer wie folgt aufrufen:

```
pi@raspberry:~$ sudo raspi-config
```

Im nächsten Abschnitt erfahren Sie, welchem Zweck die meisten Optionen in Raspi-config dienen.

3.2 Anpassen Ihrer Installation mit Raspi-config

Bevor Sie irgendetwas mit dem Pi machen, sollten Sie die wichtigsten Einstellungen Ihrer Installation mit Raspi-config anpassen. Zum Beispiel sollten Sie den freien Speicherplatz auf Ihrer SD-Karte vergrößern und die richtige Ländereinstellung vornehmen.

In diesem Abschnitt lernen Sie die wichtigsten Menüpunkte von Raspi-config kennen, die restlichen Menüpunkte folgen später.

Nutzen des gesamten Speichers Ihrer SD-Karte

Das Raspbian-Image beschränkt Ihr Root-Dateisystem auf 2 GB. Mit anderen Worten, egal wie groß die SD-Karte ist, sie wird auf 2 GB beschränkt. Sie können das Image auf eine SD-Karte mit 16 GB kopieren, davon aber nur 2 GB nutzen.

Mit dem Menüpunkt *expand_rootfs* in Raspi-config können Sie das schnell abstellen. Wählen Sie den Menüpunkt aus, und nach dem nächsten Neustart kann der Pi auf den gesamten Speicher Ihrer SD-Karte zugreifen. Abhängig von Geschwindigkeit und Größe Ihrer SD-Karte kann das eine Weile dauern.

Vergessen Sie nicht, dass Raspi-config nicht automatisch neu gestartet wird. Sie müssen sich mit dem Benutzernamen *pi* und dem Passwort *raspberry* anmelden. Um Raspi-config erneut auszuführen, geben Sie folgenden Befehl ein:

```
pi@raspberry:~$ sudo raspi-config
```

Konfigurieren des Overscan-Modus

Das Raspberry-Team wollte, dass der Pi mit so vielen unterschiedlichen Bildschirmen wie möglich funktioniert. Deshalb mussten Overscan und Underscan berücksichtigt werden. Im Falle von Underscan nutzt die Videoausgabe nicht die gesamte Bildschirmgröße, sodass Sie einen schwarzen Rahmen um das eigentliche Bild sehen. Bei Overscan passiert das Gegen-

teil: In manchen Fällen sehen Sie nicht das gesamte Bild, weil es an den Rändern abgeschnitten wird. Mit dem Menüpunkt *Overscan* in Raspi-config können Sie den Overscan-Modus ein- oder ausschalten. In Abschnitt 4.3 lernen Sie, wie Sie noch feinere Einstellungen vornehmen können.

Ändern der Tastaturbelegung und der Ländereinstellung

In der Voreinstellung nimmt Debian an, dass Sie eine englische Tastatur verwenden, was verwirrend sein kann, wenn das nicht der Fall ist. Sie können die Tastaturbelegung im Menüpunkt *configure_keyboard* in Raspi-Config ändern. Dadurch wird ein Konfigurationsprogramm gestartet, das Sie zuerst nach der Art Ihrer Tastatur fragt (siehe Abb. 3–2).

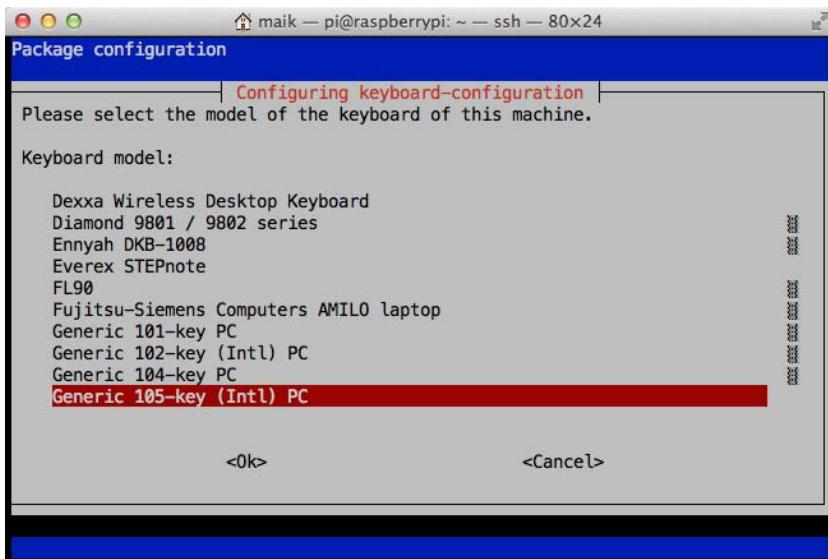


Abb. 3–2 Wählen Sie Ihre Tastatur.

Nun müssen Sie Ihre Sprache einstellen und danach das Verhalten einiger Sondertasten konfigurieren.

Um das neue Tastaturlayout zu aktivieren, müssen Sie Raspi-config über die *Finish*-Schaltfläche verlassen und den Pi neu starten. Vorher sollten Sie jedoch noch die Ländereinstellung anpassen. Eine Ländereinstellung besteht aus mehr als nur dem Tastaturlayout. Sie legt beispielsweise fest,

wie Text und Daten sortiert und dargestellt werden. Sie beeinflusst auch die Sprache, in der das System Informationen wie Menütexte und Anwendungen anzeigt. In Abbildung 3–3 sehen Sie z.B. einen deutschen LXDE-Desktop. Sie können die Ländereinstellung im Menüpunkt *change_locale* von Raspi-config einstellen.

Dadurch starten Sie ein Konfigurationsprogramm, das Sie mit dem Menü in Abbildung 3–4 begrüßt.

Hier können Sie die Ländereinstellungen vornehmen, die Raspbian verwenden soll. Sie können mehrere Einstellungen auswählen und bei Bedarf zwischen ihnen umschalten. Verwenden Sie die Pfeiltasten, um sich durch die Liste zu bewegen, und die Leertaste, um eine Ländereinstellung ander abzuwählen. Mit der Tabulatortaste können Sie zwischen der Länderauswahl und den *Ok*- und *Cancel*-Schaltflächen umschalten. Drücken Sie Return, um eine Schaltfläche auszuwählen.

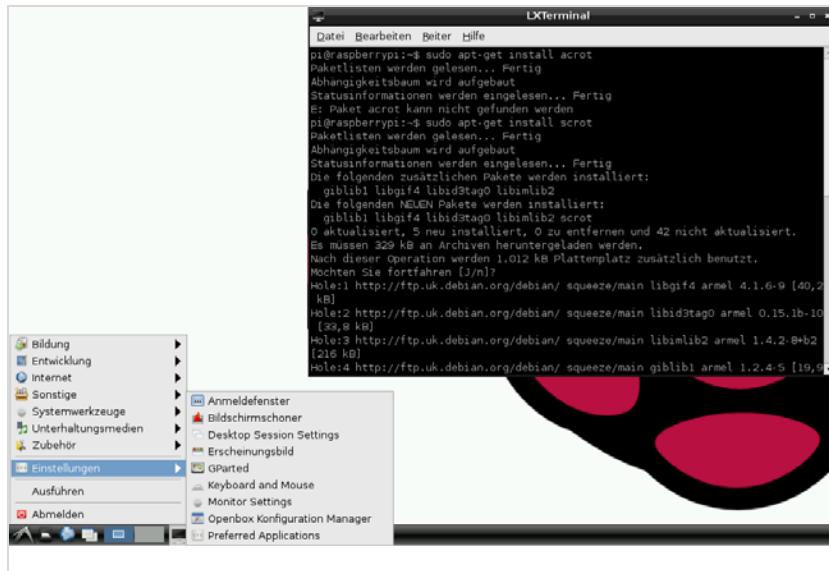


Abb. 3–3 Eine deutsche Version von LXDE

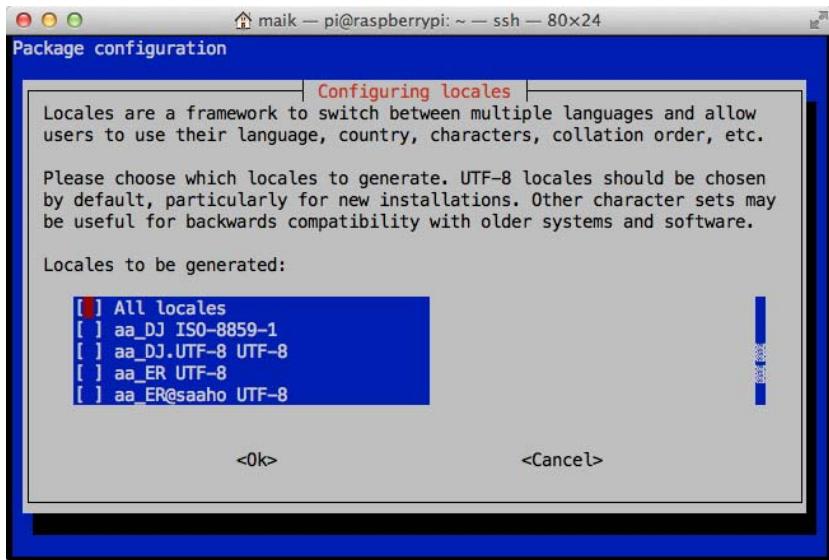


Abb. 3-4 Ihre Ländereinstellung anpassen

Wenn Sie eine Liste mit Ländereinstellungen ausgewählt und auf *Ok* gedrückt haben, können Sie die gewünschte Voreinstellung wählen. Drücken Sie erneut auf *Ok*, und Sie sind fertig.

Einstellen von Zeitzone, Zeit und Datum

Um Kosten zu sparen, besitzt der Pi keine Echtzeituhr, das heißt, er speichert die aktuelle Uhrzeit und das aktuelle Datum nicht intern. Die richtige Uhrzeit und das Datum einzustellen ist nicht nur eine nette Funktion, sondern auch wichtig für kryptografische Funktionen, wie das Überprüfen von Zertifikaten. Sie benötigen die aktuelle Uhrzeit und das Datum also für viele Zwecke. Raspbian verbindet sich beim Systemstart mit einem Zeitserver im Internet und stellt Uhrzeit und Datum automatisch ein.

Intern kennt der Pi also die richtige Uhrzeit in der UTC-Darstellung, jedoch nicht Ihre Zeitzone. Dazu dient der Menüpunkt *change_timezone* in Raspi-config. Wählen Sie ihn aus und beantworten Sie einige Fragen, die exakt bestimmen, wo Sie leben. Dann werden die Informationen über die Zeitzone in Ihrem Profil abgelegt, sodass der Pi beim nächsten Start weiß, in welcher Zeitzone Sie leben.

Wenn Sie Ihren Pi nicht mit dem Internet verbunden haben, können Sie Datum und Uhrzeit wie folgt manuell einstellen:

```
pi@raspberry:~$ sudo date --set="2013-02-23 13:24:42"
```

Diese Lösung hat ein paar Nachteile: Sie ist nicht so genau wie eine automatische Einstellung und Sie müssen die Einstellung bei jedem Neustart des Pi vornehmen, was leicht vergessen wird.

Anpassen der Speicherkonfiguration

Wie Sie bereits wissen, hat der Pi entweder 256 MB oder 512 MB RAM. Lassen Sie uns das auf einem Gerät mit 256 MB einmal überprüfen:

```
pi@raspberry:~$ free -m
              total    used    free   shared    buffers    cached
Mem:        186      37     149      0        5       19
-/+ buffers/cache:    12     174
Swap:       127      0     127
```

Nanu! Scheinbar hat der Pi viel weniger als 256 MB. Wie kann das sein? Machen Sie sich darüber keine Sorgen. Mit Ihrer Hardware ist alles in Ordnung, und Ihr Pi hat tatsächlich 256 MB RAM. Er teilt den Speicher nur zwischen der CPU und der GPU auf (das Gerät, das für die Grafikverarbeitung zuständig ist). In der Voreinstellung erhält die CPU 192 MB Arbeitsspeicher und die GPU 64 MB. In den meisten Anwendungsfällen ist das sinnvoll, manchmal ist jedoch eine andere Einteilung notwendig. Wenn Sie den Pi z.B. als Server einsetzen, benötigen Sie keine besondere Grafikleistung, sondern mehr RAM für die CPU.

Sie können die Speicherkonfiguration mit Menüpunkt *memory_split* in Raspi-config ändern. Hier können Sie angeben, wie viel Speicher dem Grafikchip zugeteilt werden soll. Wählen Sie die passende Menge für Ihren Anwendungsfall aus und starten Sie den Pi dann neu.

Ändern Ihres Passworts

Als dieses Buch geschrieben wurde, mussten Sie den Benutzernamen *pi* und das Passwort *raspberry* verwenden, um sich am Pi anzumelden. Wenn Sie zu den Glücklichen gehören, die eine der ersten Platinen erwischt haben, bekamen Sie noch ein Beiblatt mit falschen Anmelde-

informationen. In früheren Versionen war das Passwort *suse*. Um ganz sicher zu gehen, sollten Sie die aktuellen Angaben auf der Downloadseite¹ nachschlagen.

Verwenden Sie den Menüpunkt *change_pass* in Raspi-config, um das Passwort zu ändern. Raspi-config fragt Sie nach dem neuen Passwort und fordert noch eine Bestätigung des Passworts an. Aus Sicherheitsgründen können Sie keine einfachen Passwörter wie 123 oder aaaa verwenden. Wenn Sie mehr über Benutzer und Passwörter erfahren möchten, lesen Sie Abschnitt A.4.

raspberry ist übrigens ein sehr schlechtes Passwort, nicht nur, weil man es leicht erraten kann, sondern auch, weil es den Buchstaben »y« enthält. Jeder, der keine englische oder amerikanische Tastatur besitzt, wird bei der Anmeldung Probleme haben, weil die Debian-Voreinstellung von einem QWERTY-Tastaturlayout ausgeht. In Deutschland verwendet man gewöhnlich ein QWERTZ-Layout. Wenn Sie also absolut sicher sind, das Passwort schon zehnmal richtig eingegeben zu haben, probieren Sie einmal »raspberrz«.

Den Pi beschleunigen (Overclocking)

Normalerweise arbeitet der Pi intern mit einem Takt von 700 MHz. Das ist schon recht flott, schneidet aber im Vergleich zu modernen PCs eher bescheiden ab. Wie viele PCs können Sie den Pi aber auch übertakten, und zwar mit dem Menüpunkt *overclocking* in Raspi-config. Hier können Sie den Systemtakt auf bis zu 1 GHz setzen. Das beschleunigt zwar den Pi deutlich, erhöht aber auch den Stromverbrauch und die Betriebstemperatur.

Je nach Leistung und Qualität Ihres Netzteils kann das Übertakten deshalb auch zu Stabilitätsproblemen führen. In diesem Fall halten Sie beim Booten des Pi die Shift-Taste gedrückt. Dies schaltet die Übertaktung ab und Sie können anschließend mittels Raspi-config eine geringere Taktrate wählen.

1) <http://www.raspberrypi.org/downloads>

3.3 Den Desktop starten

Im Gegensatz zu anderen Betriebssystemen ist eine Desktop-Umgebung bei Linux optional. Es ist daher nicht ungewöhnlich, sie manuell starten zu müssen. Wenn Sie die Kommandozeile nicht oft verwenden, können Sie sie automatisch starten lassen, wenn der Pi hochfährt. Wählen Sie dazu den Menüpunkt *boot behaviour* in Raspi-config, um das Startverhalten zu ändern.



Abb. 3-5 Anmeldeauflorderung

Ansonsten sehen Sie nach einer erfolgreichen Anmeldung immer nur eine langweilige Eingabeauflorderung. Starten Sie den Desktop mit dem folgenden Befehl, um etwas mehr Farbe zu bekommen. (Es erinnert an die guten, alten MS-DOS-Zeiten, als Sie noch *win* starten mussten, um loszulegen, oder?)

```
pi@raspberry:~$ startx
```

Nach ein paar Sekunden zeigt der Raspberry Pi einen hübschen Desktop mit einer farbenfrohen Himbeere als Hintergrund an:

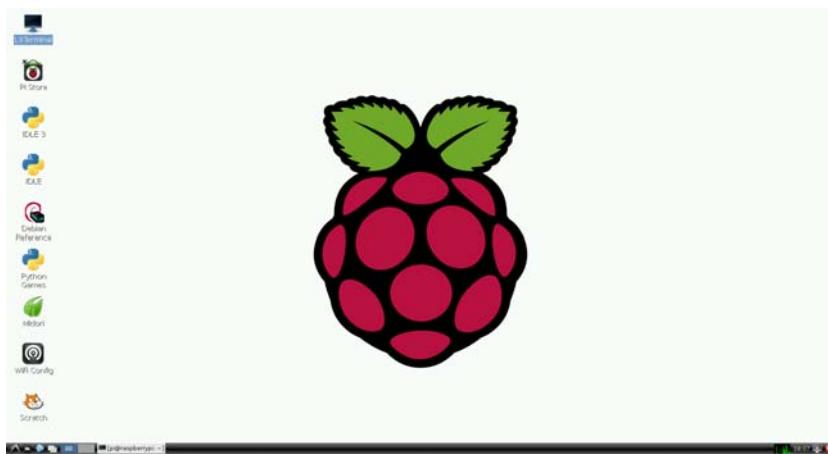


Abb. 3–6 Desktop des Raspberry Pi

Die Desktop-Umgebung, die Sie gerade gestartet haben, heißt LXDE², und obwohl sie nicht sehr viel Systemleistung benötigt, hält sie einige nette Funktionen bereit. Zum Beispiel bietet sie virtuelle Bildschirme, die Sie mit den Schaltflächen in der Werkzeugeiste unten verwalten.

Das Starten von Anwendungen funktioniert genauso wie auf Windows-Systemen vor Windows 8. Klicken Sie auf das kleine LXDE-Logo links von der Werkzeugeiste unten, und Sie sehen, welche Anwendungen zur Verfügung stehen. Bewegen Sie sich mit der Maus durch das Pop-up-Menü und starten Sie eine Anwendung mit einem Mausklick auf deren Namen. In Abbildung 3–3 sehen Sie das Pop-up-Menü in Aktion.

Sie können viele weitere Einstellungen vornehmen, wie das »Look and Feel« aller Elemente, die Auflösung usw. Die meisten dieser Einstellungen ändern Sie in den Menüs für die Systemeinstellungen. Abbildung 3–7 zeigt einige davon.

Um LXDE zu verlassen, verwenden Sie den kleinen Schalter unten rechts im Bildschirm. Wenn Sie Raspi-config so eingestellt haben, dass der

2) <http://lxde.org/>

Desktop immer gestartet wird, fährt der Pi herunter, wenn Sie sich aus LXDE ausloggen. Wenn nicht, gelangen Sie zurück zum Terminalfenster des Pi. Um ihn von dort herunterzufahren, geben Sie folgenden Befehl ein:

```
pi@raspberry:~$ sudo shutdown -h now
```

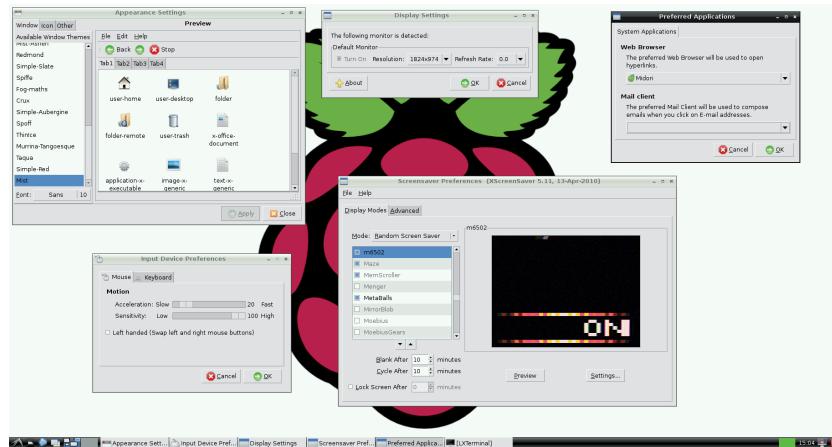


Abb. 3-7 Sie können in LXDE viele Einstellungen ändern.

3.4 Verwalten Ihrer Software mit apt-get

Nachdem Sie Debian eingerichtet haben, wollen Sie vermutlich weitere Software installieren. In der guten, alten Zeit war es sehr lästig, auf Linux-Systemen neue Software zu installieren. Normalerweise mussten Sie den Quellcode eines Programms herunterladen und es selbst komplizieren und installieren. Basierte das Programm auf anderen Projekten oder Bibliotheken, erfuhren Sie das, wenn der Compiler oder Linker Fehlermeldungen ausspuckte, und Sie mussten die Probleme selbst lösen, also weitere Programme herunterladen, komplizieren und installieren usw.

Diese Zeiten sind glücklicherweise vorbei. Alle modernen Linux-Distributionen besitzen einen Paketmanager, der das Herunterladen und Installieren von Software automatisiert. Nicht nur die Abhängigkeiten werden vom Paketmanager automatisch aufgelöst, er spart auch viel Zeit, indem

er binäre Pakete herunterlädt, statt sie lokal zu kompilieren. Und er hilft dabei, Software loszuwerden, die Sie nicht mehr brauchen.

Auch Debian wird mit einem Paketmanager geliefert, und sein Name ist apt-get (apt steht für Advanced Packaging Tool, also fortschrittliches Paketwerkzeug). In diesem Abschnitt lernen Sie, wie Sie Software hinzufügen, aktualisieren und entfernen.

Installieren neuer Software

Die Debian-Distribution des Pi wird mit einer kleinen Auswahl von Anwendungen geliefert. Das ist sinnvoll, da der Pi nicht über eine Festplatte verfügt. Um das Maximum aus dem Gerät herauszuholen, müssen Sie also einige Programme installieren. Die gute Nachricht lautet, dass sich die Installation von Software auf dem Pi nicht von der Softwareinstallation auf einem normalen PC mit Debian unterscheidet. Ihre Software erhalten Sie meist aus derselben Quelle und es gibt dort eine riesige Auswahl an Programmen. Leider sind nicht alle Pakete für die ARM-Architektur des Pi erhältlich, und manche Anwendungen laufen einfach nicht, weil sie mehr Ressourcen benötigen, als der Pi bietet. Trotzdem gibt es zahlreiche sinnvolle Programme.

In diesem Abschnitt installieren wir einen PDF-Reader auf Ihrem Pi. Wenn Sie bislang nur mit Microsoft Windows oder Apple Mac OS X gearbeitet haben, haben Sie sich vermutlich nicht um einen PDF-Reader gekümmert. Auf manchen Linux-Systemen und vor allem auf dem kleinen Pi können Sie ein solches Programm aber nicht voraussetzen.

Erfreulicherweise können wir aus einer ganzen Reihe verschiedener Werkzeuge auswählen und es gibt sogar eine spezielle Seite mit kostenfreien PDF-Readern.³ Zwei dieser Reader sehen besonders interessant aus: xpdf⁴ und evince⁵. Wir werden beide installieren, ausprobieren und dann entfernen Sie denjenigen, der Ihnen nicht gefällt.

3) <http://pdfreaders.org/>

4) <http://www.foolabs.com/xpdf/>

5) <http://projects.gnome.org/evince/>

Neue Pakete installieren Sie mit dem Befehl `install`. Um `xpdf` und `evince` zu installieren, führen Sie die folgenden Befehle aus (und stellen Sie sicher, dass Sie mit dem Internet verbunden sind):

```
pi@raspberry:~$ sudo apt-get install xpdf  
pi@raspberry:~$ sudo apt-get install evince
```

Alternativ können Sie auch mehr als ein Paket gleichzeitig installieren:

```
pi@raspberry:~$ sudo apt-get install xpdf evince
```

Jetzt haben Sie beide PDF-Reader als unabhängige Pakete installiert und können sie starten und testen, um herauszufinden, welcher Ihnen besser gefällt. Sie finden für beide Anwendungen Einträge im Abschnitt *Graphics* des LXDE-Startmenüs. Sie können sie auch in einem Terminal starten, indem Sie einen der beiden folgenden Befehle ausführen (wenn die Desktop-Umgebung vorher gestartet wurde).

```
pi@raspberry:~$ evince
```

oder:

```
pi@raspberry:~$ xpdf
```

In Abbildung 3–8 sehen Sie beide Programme, die das gleiche Dokument anzeigen.

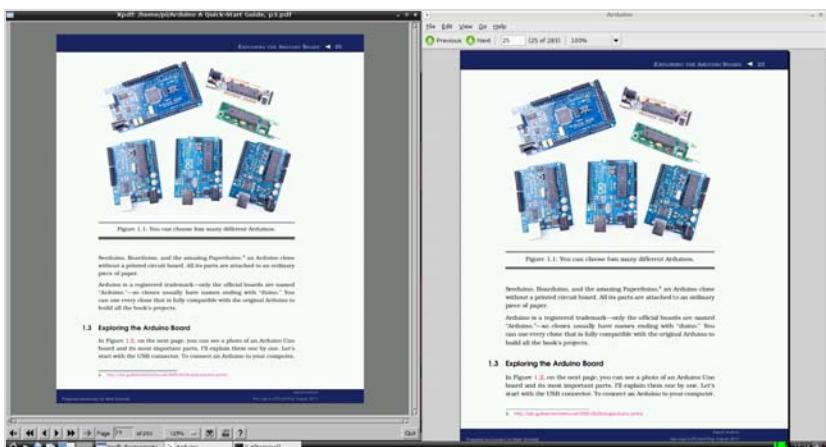


Abb. 3–8 Zwei PDF-Reader mit dem gleichen Dokument

Entfernen von Software

Probieren Sie beide Anwendungen aus und entscheiden Sie, welche Ihnen mehr zusagt. Lassen Sie uns annehmen, Sie bevorzugen evince. Dann können Sie xpdf mit dem folgenden Befehl entfernen:

```
pi@raspberry:~$ sudo apt-get purge xpdf
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer
required:
  lesstif2 poppler-data ghostscript poppler-utils libxp6
Use 'apt-get autoremove' to remove them.
The following packages will be REMOVED:
  xpdf
0 upgraded, 0 newly installed, 1 to remove and 42 not upgraded.
After this operation, 487 kB disk space will be freed.
Do you want to continue [Y/n]?
(Reading database ... 49208 files and directories currently installed.)
Removing xpdf ...
Purging configuration files for xpdf ...
Processing triggers for menu ...
Processing triggers for man-db ...
```

Die Anwendung xpdf ist jetzt spurlos verschwunden. Wenn Sie nur die Anwendung aber nicht die dazugehörigen Konfigurationsdateien löschen wollen, verwenden Sie remove statt purge.

Ihre Software auf dem neuesten Stand halten

Damit die Softwareinstallation so einfach und unbeschwerlich wie möglich verläuft, enthält apt-get eine Datenbank, also eine Liste mit allen erhältlichen Paketen und deren Abhängigkeiten. Diese Datenbank besteht aus nur wenigen Dateien. Normalerweise werden Sie nie direkt mit ihr arbeiten. Mit dem folgenden Befehl sollten Sie sie jedoch von Zeit zu Zeit aktualisieren:

```
pi@raspberry:~$ sudo apt-get update
```

Dadurch wird die aktuelle Paketliste von einem zentralen Server heruntergeladen und die lokale Liste von apt-get aktualisiert. Wenn Sie diesen Befehl also vor der Installation neuer Pakete ausführen, können Sie sicher

sein, dass Sie die neueste verfügbare Version erhalten. In einigen Fällen müssen Sie diesen Befehl zweimal ausführen, apt-get ist dann so freundlich, Ihnen das mitzuteilen.

Wenn Sie auf Ihrem Pi schon Software mit apt-get installiert haben, möchten Sie sie von Zeit zu Zeit vielleicht aktualisieren. Mit dem folgenden Befehl aktualisieren Sie alle momentan auf dem Pi installierten Programme:

```
pi@raspberry:~$ sudo apt-get upgrade
```

Die Ausführung dieses Befehls kann einige Zeit dauern. Wenn er jedoch abgeschlossen ist, haben Sie alle Anwendungen und Bibliotheken des Pi auf dem neuesten Stand. Dazu muss apt-get eine Menge Dateien herunterladen, die Sie nach Beendigung des Aktualisierungsvorgangs nicht mehr benötigen. Sie können die überflüssigen Dateien einfach löschen.

```
pi@raspberry:~$ sudo apt-get autoclean
Reading package lists... Done
Building dependency tree...
Reading state information... Done
```

Manchmal ändern sich die Abhängigkeiten zwischen den Paketen, weshalb es vorkommen kann, dass Sie installierte Pakete nicht mehr benötigen. Sie entfernen sie mit folgendem Befehl:

```
pi@raspberry:~$ sudo apt-get autoremove
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
ghostscript lesstif2 libxp6 poppler-data poppler-utils
0 upgraded, 0 newly installed, 5 to remove and 0 not upgraded.
After this operation, 15.3 MB disk space will be freed.
Do you want to continue [Y/n]?
(Reading database ... 49185 files and directories currently installed.)
Removing ghostscript ...
Removing lesstif2 ...
Removing libxp6 ...
Removing poppler-data ...
Removing poppler-utils ...
Processing triggers for man-db ...
```

Im Prinzip wissen Sie jetzt alles, um die Software auf einem Debian-System zu verwalten. Es gibt aber noch ein weiteres sinnvolles Werkzeug, das Sie kennen sollten und das ich Ihnen im nächsten Abschnitt vorstelle.

Finden von Paketen mit apt-file

Wenn Sie den genauen Namen eines Pakets kennen, das Sie installieren möchten, reicht apt-get aus. Manchmal wissen Sie den Namen jedoch nicht. Zum Beispiel müssen Sie dann und wann immer noch Programme vom Quellcode aus installieren und kompilieren. Wenn diese Software auf einer bestimmten Bibliothek basiert, die Sie nicht installiert haben, hält der Compiler oder Linker mit einer Fehlermeldung an. Normalerweise enthält die Fehlermeldung den Namen der fehlenden Datei, sodass es prima wäre ein Werkzeug zu haben, das alle Pakete findet, die diese Datei enthalten. apt-file ist ein solches Werkzeug und Sie können es wie folgt installieren:

```
pi@raspberry:~$ sudo apt-get install apt-file
```

Wie apt-get basiert auch apt-file auf einer lokalen Datenbank mit einer Liste aller Pakete und deren Abhängigkeiten. Um diese Datenbank zu aktualisieren, sollten Sie folgenden Befehl ausführen:

```
pi@raspberry:~$ sudo apt-file update
```

Jetzt können Sie mit apt-file nach Paketen suchen, die eine bestimmte Datei enthalten. Nehmen wir an, Sie haben von diesem tollen PDF-Reader für den Pi namens evince gehört, wissen aber nicht, welches Paket Sie installieren müssen, um ihn zu verwenden. Dieser Befehl erledigt alles Notwendige für Sie:

```
pi@raspberry:~$ apt-file -l search evince
evince
evince-common
evince-dbg
evince-gtk
gir1.0-evince-2.30
libevince-dev
libevince2
python-evince
```

Hierdurch wird eine Liste aller Pakete ausgegeben, die mit evince zu tun haben, und Sie können entscheiden, welches Sie installieren wollen.

Sie können apt-file auch dazu verwenden, den Inhalt eines Pakets auszugeben, selbst wenn Sie es nicht installiert haben.

```
pi@raspberry:~$ apt-file list evince
```

Paketmanager sind eine hilfreiche Sache, und jede moderne Linux-Distribution verfügt über einen. Bei Debian ist es apt-get, bei Fedora ist es *yum* und bei Arch Linux ist es *pacman*. Obwohl sie sich in ihrer Syntax geringfügig unterscheiden, bieten alle dieselben Operationen und Möglichkeiten.

3.5 Software und Medien installieren mit dem Pi Store

Ende 2012 startete die Raspberry-Stiftung zusammen mit IndieCity⁶ und Velocix⁷ den Pi Store⁸. Dort finden Sie sowohl kostenlose als auch kommerzielle Anwendungen, Spiele, Entwicklungs-Tools und Medien wie Zeitschriften oder Videos. Es gibt zwei Arten von kommerziellen Anwendungen: für einige muss man eine Gebühr im Voraus zahlen, während Sie für andere eine Spende machen können, wenn Sie möchten.

Der Pi Store ist als Website zugänglich, viel bequemer ist es aber, ihn als native Client-Anwendung zu installieren (siehe Abbildung 3–9). Im aktuellsten Raspbian Image ist der Pi Store Client bereits enthalten und man kann ihn durch einen Doppelklick auf das entsprechende Symbol auf dem Desktop starten.

Wenn Sie nicht bereits über den Pi Store Client verfügen, können Sie ihn mit den folgenden Befehlen installieren:

```
pi@raspberry:~$ sudo apt-get update  
pi@raspberry:~$ sudo apt-get install pistore
```

6) <http://store.raspberrypi.com/>

7) <http://www.indiecity.com/>

8) <http://www.velocix.com/>

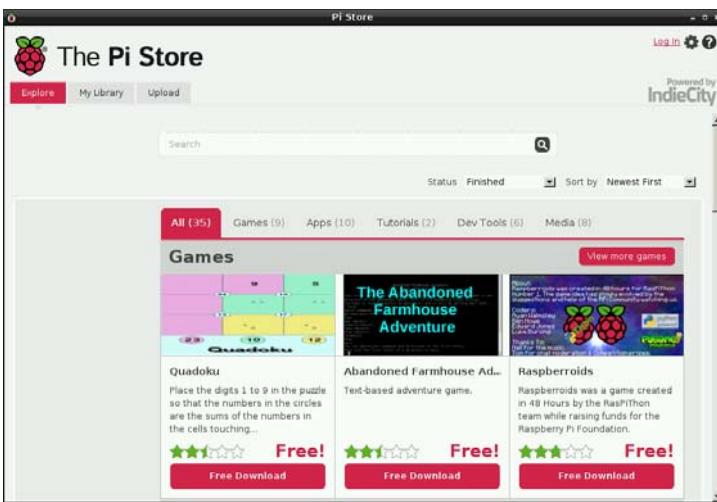


Abb. 3–9 Der Pi Store besitzt einen nativen Client.

Die Anwendung ist ziemlich selbsterklärend. Nachdem Sie eingeloggt sind, können Sie Software mit einem einzigen Mausklick installieren. Zur Zeit enthält der Pi Store nur einige wenige Artikel, aber wenn man die Beliebtheit des Pi berücksichtigt, könnte sich dies bald ändern.

Ein Titel ist besonders interessant: *Iridium Rising*⁹ ist ein exklusives Spiel für den Pi. Es ist kostenlos und es sieht sehr professionell aus.

3.6 Die nächsten Schritte

In diesem Kapitel haben Sie den Pi zum ersten Mal gestartet und viele Einstellungen entsprechend Ihren persönlichen Präferenzen vorgenommen. Sie wissen jetzt auch, wie Sie die Software auf Ihrem Pi verwalten – wie Sie sie installieren, aktualisieren und entfernen.

Die Installation und Konfiguration des Pi-Betriebssystems sind wichtige Schritte, im Gegensatz zu normalen PCs benötigt der Pi jedoch weitere Anpassungen. Im nächsten Kapitel lernen Sie die Pi-Firmware kennen und erfahren, wie Sie sie an Ihre Bedürfnisse anpassen.

9) <http://iridiumrising.co.uk/>

4 Die Firmware konfigurieren

Der Pi benötigt nicht nur ein Betriebssystem, sondern auch eine Firmware, die die Hardware grundlegend steuert. Die Firmware steuert und konfiguriert z.B. die GPU, den Kartenleser und in einigen Belangen sogar die CPU. Sie ist ein wesentlicher Bestandteil des Raspberry Pi und kann viele Probleme lösen, wie z.B. die Qualität der Videoausgabe, indem sie die richtigen Parameter einstellt. In diesem Kapitel lernen Sie, wie die Pi-Firmware konfiguriert und aktualisiert wird.

Neben der Firmware sollten Sie außerdem den Linux-Kernel von Zeit zu Zeit aktualisieren. Der Kernel ist das Herz eines Linux-Systems, da er alle Prozesse und die Hardware verwaltet. Alle Anwendungen sind abhängig vom Kernel und in diesem Kapitel erfahren Sie, wie Sie ihn auf dem aktuellen Stand halten.

4.1 Aktualisieren von Firmware und Kernel

Das Debian-Image wird schon mit einer Firmware für den Pi geliefert, aber die Entwickler des Linux-Kernels und der Pi-Firmware bringen häufig Aktualisierungen heraus. Neue Versionen enthalten Fehlerbehebungen und Verbesserungen, sodass es sinnvoll ist, den Kernel und die Firmware von Zeit zu Zeit zu aktualisieren. Um zu prüfen, welche Versionen von Kernel und Firmware auf Ihrem Pi installiert sind, führen Sie die folgenden Befehle aus:

```
pi@raspberrypi ~ $ uname -a
Linux raspberrypi 3.6.11+ #348 PREEMPT Tue Jan 1 16:33:22 GMT 2013
armv6l GNU/Linux
pi@raspberrypi ~ $ /opt/vc/bin/vcgencmd version
Dec 28 2012 11:22:54
Copyright (c) 2012 Broadcom
version 359904 (release)
```

Sie finden die aktuelle Fassung dieser Dateien immer auf GitHub¹ und Sie müssen sie nur auf die SD-Karte herunterladen. Um einen neuen Kernel und eine neue Firmware zu installieren, müssen Sie nur einige Dateien im Startverzeichnis /boot des Pi ersetzen. Das Verzeichnis /boot befindet sich in der Startpartition der SD-Karte, die mit dem FAT-Dateisystem formatiert ist. Sie können sie daher nicht nur auf dem Pi, sondern auf fast jedem anderen Computer auf der Welt lesen und beschreiben. Im folgenden Screenshot sehen Sie ihren Inhalt:

Name	Date Modified	Size	Kind
bootcode.bin	10/28/12 5:40 PM	18 KB	MacBinary archive
cmdline.txt	10/28/12 10:11 PM	142 bytes	Plain Text
config.txt	10/28/12 10:11 PM	1 KB	Plain Text
fixup_cd.dat	10/28/12 5:40 PM	2 KB	Mac Wavelets WBIN document
fixup.dat	10/28/12 5:40 PM	5 KB	Mac Wavelets WBIN document
issue.txt	10/28/12 11:01 PM	137 bytes	Plain Text
kernel_cutdown.img	10/28/12 5:40 PM	2.1 MB	NDIF Disk Image
kernel_emergency.img	10/28/12 5:40 PM	9.5 MB	NDIF Disk Image
kernel.img	10/28/12 5:40 PM	2.7 MB	NDIF Disk Image
start_cd.elf	10/28/12 5:40 PM	523 KB	Unix Executable File
start.elf	10/28/12 5:40 PM	2.3 MB	Unix Executable File

11 items, 41.4 MB available

Abb. 4-1 Startpartition der SD-Karte

Die Datei start.elf enthält die Firmware und der Kernel befindet sich in kernel.img.

Sie können den neuen Kernel und die Firmware von einem normalen PC herunterladen und sie mittels Kartenleser auf die SD-Karte kopieren. Das ist jedoch unnötig kompliziert und fehleranfällig. Erfreulicherweise erleichtert rpi-update² den gesamten Prozess.

Wenn Sie es auf dem Pi installiert haben, prüft es, ob eine neue Firmware-Version zur Verfügung steht und lädt sie wenn nötig herunter. Bevor Sie rpi-update jedoch auf dem Pi installieren können, müssen Sie ein paar weitere Pakete installieren.

```
pi@raspberry:~$ sudo apt-get install ca-certificates git-core
```

1) <https://github.com/raspberrypi/firmware>

2) <https://github.com/Hexxeh/rpi-update>

Jetzt können Sie rpi-update herunterladen und ausführbar machen.

```
pi@raspberry:~$ sudo wget http://goo.gl/1B0fJ -O /usr/bin/rpi-update  
pi@raspberry:~$ sudo chmod +x /usr/bin/rpi-update
```

Führen Sie anschließend rpi-update aus.

```
pi@raspberry:~$ sudo rpi-update  
raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS  
Performing self-update  
ARM/GPU split is now defined in /boot/config.txt using the gpu_mem  
option!  
We're running for the first time  
Setting up firmware (this will take a few minutes)  
Using HardFP libraries  
If no errors appeared, your firmware was successfully setup  
A reboot is needed to activate the new firmware
```

Wie Sie sehen, hat der Pi jetzt eine neue Firmware, die Sie durch einen Neustart aktivieren können. Beachten Sie, dass rpi-update sogar versucht, Ihre Speicherkonfiguration zu erkennen und beizubehalten. In Abschnitt 3.2 »Anpassen der Speicherkonfiguration« erfahren Sie, was eine Speicherkonfiguration ist und wie Sie sie konfigurieren.

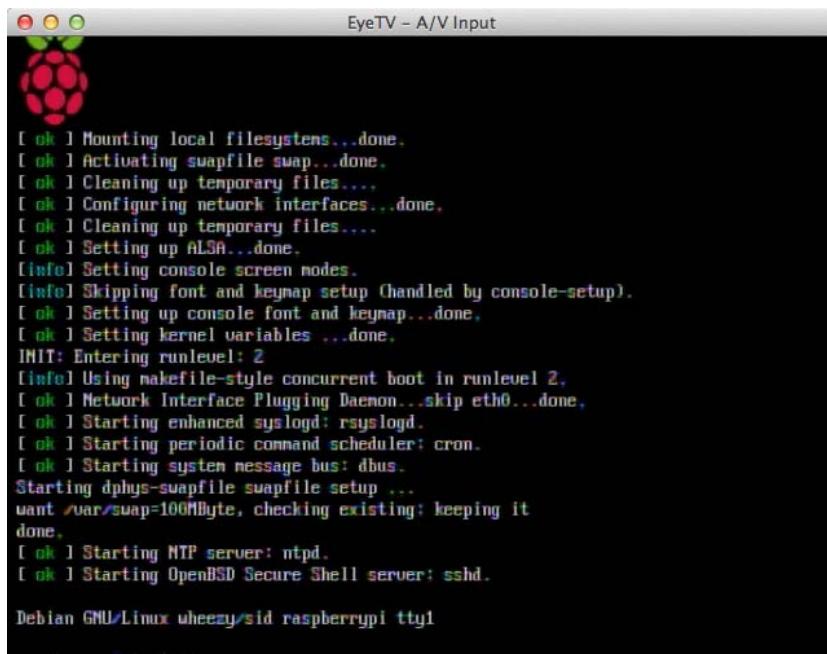
4.2 Konfigurieren der Videoausgabe

Sie können viele Aspekte der Firmware mit der Datei /boot/config.txt steuern. Sie enthält alle Konfigurationsparameter der Pi-Firmware und es ist sinnvoll, ihre Beschreibungen³ mit einem Lesezeichen im Webbrower zu sichern. Früher oder später werden Sie einige Dinge verändern wollen. Mit der Konfigurationsdatei können Sie die Video- und Audioausgabe einstellen und sogar die Taktfrequenz der CPU.

Die meisten Voreinstellungen sind auf allen Systemen sinnvoll, die Videoausgabe funktioniert aber nicht immer korrekt. Die Hauptprobleme bestehen in Overscan und Underscan, besonders bei der Ausgabe von Composite Video. Im Falle von Underscan nutzt die Videoausgabe nicht die gesamte Bildschirmgröße, sodass Sie einen schwarzen Rahmen um das eigentliche Bild sehen. Bei Overscan passiert das Gegenteil: In manchen Fällen sehen Sie nicht das gesamte Bild, weil es an den Rändern

3) http://elinux.org/RPi_config.txt

abgeschnitten wird. In Abbildung 4–2 ist die letzte Zeile z.B. nicht vollständig sichtbar. Solche Probleme lösen Sie mit wenigen Konfigurationseinstellungen.



The screenshot shows a terminal window titled "EyeTV - A/V Input". The window contains a truncated boot log from a Raspberry Pi. The log shows the system booting up, mounting filesystems, activating swapfile, cleaning temporary files, configuring network interfaces, and setting up ALSA and console screen nodes. It also handles font and keymap setup, sets up console font and keymap, and initializes kernel variables. The system enters runlevel 2, uses makefile-style concurrent boot, and starts various daemons like rsyslogd, cron, and sshd. It also starts an NTP server (ntpd) and an OpenBSD Secure Shell server (sshd). The final line visible is "Debian GNU/Linux wheezy~\$id raspberrypi tty1".

```
[ ok ] Mounting local filesystems...done.  
[ ok ] Activating swapfile swap...done.  
[ ok ] Cleaning up temporary files....  
[ ok ] Configuring network interfaces...done.  
[ ok ] Cleaning up temporary files....  
[ ok ] Setting up ALSA...done.  
[info] Setting console screen modes.  
[info] Skipping font and keymap setup (handled by console-setup).  
[ ok ] Setting up console font and keymap...done.  
[ ok ] Setting kernel variables ...done.  
INIT: Entering runlevel: 2  
[info] Using makefile-style concurrent boot in runlevel 2.  
[ ok ] Network Interface Plugging Daemon...skip eth0...done,  
[ ok ] Starting enhanced syslogd: rsyslogd.  
[ ok ] Starting periodic command scheduler: cron.  
[ ok ] Starting system message bus: dbus.  
Starting dphys-swapfile swapfile setup ...  
want /var/swap=100Mbyte, checking existing: keeping it  
done.  
[ ok ] Starting NTP server: ntpd.  
[ ok ] Starting OpenBSD Secure Shell server: sshd.  
  
Debian GNU/Linux wheezy~$id raspberrypi tty1
```

Abb. 4–2 Overscan führt zu einer abgeschnittenen Ausgabe.

Wie Sie in Abschnitt 4.1 gelernt haben, können Sie alle Dateien im Verzeichnis `/boot` auch auf Ihrem PC bearbeiten. Wenn Sie `/boot/config.txt` lieber auf dem Pi bearbeiten, öffnen Sie die Datei mit dem Texteditor `nano` oder einem anderen Editor Ihrer Wahl.

```
pi@raspberry:~$ sudo nano /boot/config.txt
```

Um den Overscan des Bildschirms anzupassen, fügen Sie der Konfigurationsdatei die folgenden Zeilen hinzu:

```
# Adjust overscan.  
overscan_left=10  
overscan_right=20  
overscan_top=0  
overscan_bottom=10
```

Konfigurationsparameter haben ein einfaches Format: Sie beginnen mit einem Namen, gefolgt von einem Gleichheitszeichen und einem Wert. Sie können auch Kommentare in die Konfigurationsdatei einfügen. Diese beginnen mit einem #-Zeichen. Das vorhergehende Beispiel stellt die Option overscan_bottom auf 10 Pixel ein, sodass der Pi nach dem nächsten Neustart die unteren 10 Pixel auf dem Bildschirm freilässt. Abbildung 4–3 zeigt den Effekt.



```
EyeTV - A/V Input

[info] Setting console screen modes.
[info] Skipping font and keymap setup (handled by console-setup).
[ 26.065969] smsc95xx 1-1.1:1.0: eth0: link up, 100Mbps, full-duplex, lpa 0xC5E1
[ ok ] Setting up console font and keymap...done,
[ ok ] Setting kernel variables ...done,
INIT: Entering runlevel: 2
[info] Using makefile-style concurrent boot in runlevel 2.
[ ok ] Network Interface Plugging Daemon...skip eth0...done.
[ ok ] Starting enhanced syslogd: rsyslogd.
[ ok ] Starting periodic command scheduler: cron.
[ ok ] Starting system message bus: dbus.
Starting dphys-swapfile swapfile setup ...
want /var/swap=100MByte, checking existing; keeping it
done.
[ ok ] Starting NTP server: ntpd.
[ ok ] Starting OpenBSD Secure Shell server: sshd.
My IP address is 192.168.2.109
Debian GNU/Linux wheezy$ id raspberrypi tty1
raspberrypi login:
```

Abb. 4–3 Overscan-Probleme lassen sich leicht lösen.

Sie können dieselben Optionen auch nutzen, um Underscan-Probleme zu lösen, den Bildbereich also zu vergrößern und den schwarzen Rahmen verschwinden zu lassen. Um das zu erreichen, verwenden Sie die Option mit negativen Werten.

```
overscan_left=-20
overscan_right=-10
```

Sie müssen den Pi nach jeder Änderung in /boot/config.txt neu starten, weshalb es etwas dauern kann, bis Sie die korrekten Einstellungen gefunden und aktiviert haben.

Es ist sinnvoll, sich zumindest kurz mit den Konfigurationsparametern zu beschäftigen, damit Sie sich im Falle eines Problems an sie erinnern. Mit den meisten Einstellungen können Sie herumspielen, aber seien Sie gewarnt: Einige Optionen, z.B. die Einstellungen zum Übertakten, können zum Garantieverlust Ihres Pi führen. Wenn Ihr Pi nicht hochfährt oder der Bildschirm nicht mehr lesbar ist, bearbeiten Sie /boot/config.txt auf einem separaten PC und machen Sie Ihre letzten Änderungen rückgängig. Wenn überhaupt nichts mehr geht, löschen Sie einfach /boot/config.txt und der Pi startet mit seinen internen Vorgaben. Natürlich können Sie auch eine frische Version des Raspbian-Image auf die SD-Karte kopieren.

4.3 Testen und Konfigurieren des Audiosystems

Die Audioausgabe ist immer noch etwas problematisch unter Linux, Raspbian aktiviert Audio jedoch standardmäßig. Für einen ersten Test verwenden Sie die folgenden Befehle:

```
pi@raspberry:~$ cd /opt/vc/src/hello_pi/libs/i1client  
pi@raspberry:~$ make  
pi@raspberry:~$ cd ../../hello_audio  
pi@raspberry:~$ make  
pi@raspberry:~$ ./hello_audio.bin
```

Dadurch wird ein kleines Testprogramm kompiliert, das einen Sirenenton abspielt. Die Ausgabe erfolgt über den analogen Audioanschluss, schließen Sie also Kopfhörer oder Lautsprecher an, um die Sirene zu hören. Alternativ können Sie den Ton über HDMI ausgeben.

```
pi@raspberry:~$ ./hello_audio.bin 1
```

Der Pi verwendet automatisch die bestmögliche Schnittstelle für Ton. Er wählt HDMI, wenn verfügbar, und ansonsten den Analogausgang. Dieses Verhalten können Sie mit *amixer* ändern. Es handelt sich um ein kleines Werkzeug, mit dem Sie die Audiohardware konfigurieren können. Führen Sie es aus, um zu sehen, welche Optionen Sie ändern können:

```
pi@raspberry:~$ amixer controls  
numid=3,iface=MIXER,name='PCM Playback Route'  
numid=2,iface=MIXER,name='PCM Playback Switch'  
numid=1,iface=MIXER,name='PCM Playback Volume'
```

Sie können nur drei Optionen ändern und müssen sie über ihre *numid* ansprechen. Es wäre viel angenehmer, wenn die Optionen einen richtigen Namen hätten, die Entwickler wollten jedoch numerische IDs verwenden. Die Option für den Ausgabekanal hat den Wert 3. Setzen Sie ihn wie folgt:

```
pi@raspberry:~$ sudo amixer cset numid=3 1
```

Dadurch wird der Wiedergabekanal auf 1 gesetzt (analog). Sie können den Wert auch auf 0 (automatisch) oder 2 (HDMI) setzen. Wenn alles wie gewünscht funktioniert, können Sie den amixer-Befehl in die Datei /etc/rc.local aufnehmen, sodass er automatisch beim Systemstart ausgeführt wird. Öffnen Sie einen Texteditor wie *nano* und fügen Sie folgende Zeile in /etc/rc.local ein:

```
amixer cset numid=3 1
```

Fügen Sie sie am Ende der Datei ein, aber nicht als letzte Zeile. Schreiben Sie sie vor die Anweisung *exit 0*.

Einige Bildschirme erkennen das HDMI-Kabel nicht ordnungsgemäß, sodass Sie zwar Videosignale über HDMI bekommen, aber keinen Ton. Dies ändern Sie, indem Sie den Firmware-Konfigurationsparameter *hdmi_drive* in /boot/config.txt auf 2 setzen. Das geschieht, wie in Abschnitt 4.2 beschrieben.

4.4 Die nächsten Schritte

In diesem Kapitel haben Sie gelernt, wie die Pi-Firmware konfiguriert wird. Sie können jetzt auch Probleme bei der Bildschirmdarstellung lösen und wissen, wie Sie Systemparameter an Ihre Bedürfnisse anpassen. Im nächsten Kapitel machen wir eine kurze Pause und verwandeln den Pi in ein Kiosk-System.

5 Intermezzo: Mit Pi einen Kiosk aufbauen

Wenn Sie in letzter Zeit in einem Wartezimmer gewesen sind, kann es gut sein, dass Sie ein Kiosk-System¹ gesehen haben. Es besteht normalerweise aus einem alten Fernsehgerät und einem DVD-Player und manchmal sogar einem Videorecorder. Im Wartezimmer einer Arztpraxis sehen und hören Sie eine Menge über neue und kostspielige Behandlungen, während Sie in einer Reparaturwerkstatt oft mit Werbung für unnütze Zubehörteile bombardiert werden. Häufig flackert das Videosignal stark, die ganze Vorführung strotzt von Rechtschreibfehlern und insgesamt kommt der Eindruck auf, dass sich hier besser ein Profi ans Werk gemacht hätte.

Andererseits finden sich auch wirklich gute Kiosk-Systeme, z.B. beim Warten auf die U-Bahn. Hier sehen Sie die neuesten Nachrichten, Wettervorhersagen und Zeichentrickfilme auf Großbildschirmen. Der große Unterschied zwischen guten und schlechten Kiosk-Systemen besteht darin, dass die guten meist nicht dasselbe Material unendlich oft wiederholen. Sie aktualisieren ihren Inhalt über das Internet.

Der Raspberry Pi ist die perfekte Plattform, um ein günstiges und doch leistungsfähiges Kiosk-System aufzubauen. In diesem Kapitel lernen Sie, wie Sie mit dem Pi einen Kiosk basteln, der Informationen aus der Twitter-Livesuche anzeigt.

1) http://en.wikipedia.org/wiki/Kiosk_software

5.1 Anzeigen von Informationen aus der Twitter-Livesuche

Die einzige Software, die Sie für die meisten Kiosk-Anwendungen benötigen, ist ein Webbrowser. Webbrowser sind gut zur Darstellung von Multimediainhalten geeignet, weshalb Sie einfach die gewünschten Informationen als HTML-Seite aufbereiten müssen.

Sie müssen außerdem die Menüzeile des Browsers deaktivieren und sicherstellen, dass der Browser den Inhalt regelmäßig aktualisiert. Die meisten modernen Browser verfügen bereits über einen Kiosk-Modus und machen all das automatisch.

In Kapitel 6 erfahren Sie mehr über Midori, den Webbrowser, der mit Debian geliefert wird. Dieser Browser ist sehr leistungsfähig und wir können damit ein tolles Kiosk-System aufbauen.

Unser System zeigt eine Livesuche in Twitter anhand einer Wortliste an. Für dieses Buch versteht es sich von selbst, dass wir nach den Begriffen *maik_schmidt*, *pragprog* und *raspberrypi* suchen. Im folgenden Screenshot sehen Sie das Ergebnis.

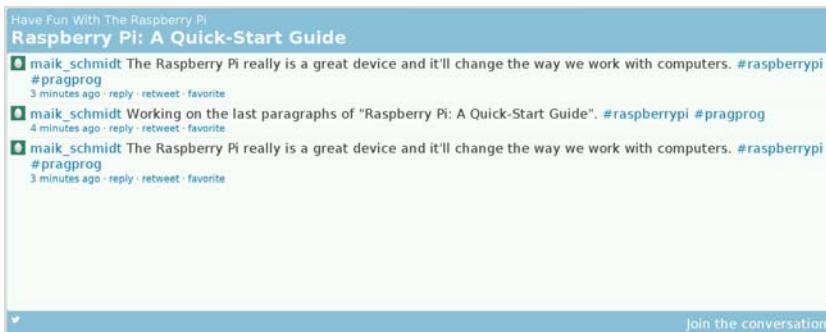


Abb. 5-1 Ergebnis der Livesuche

Der Inhalt wird alle dreißig Sekunden aktualisiert, wobei die letzte Meldung langsam aus dem Fenster verschwindet, während der Inhalt nach oben scrollt. Auf einem 46-Zoll-Fernseher sieht das System wirklich eindrucksvoll aus und wäre eine nette Ergänzung für den Eingangsbereich Ihrer Firma.

Vielelleicht erinnert Sie der Screenshot an ein modifiziertes Twitter-Widget? Das ist kein Zufall, da es sich wirklich um eines handelt. Normalerweise integrieren Sie ein Twitter-Such-Widget in Ihre Webseiten, es funktioniert aber auch mit einem Kiosk-System. Der Trick besteht darin, das Widget nicht als Widget, sondern als vollständige Kiosk-Anwendung einzusetzen. Wir müssen es nur vergrößern und auf die Größe des Webbrowsers bringen.

Das Tolle bei Widgets ist, dass Sie sie normalerweise nicht selbst programmieren müssen. Auf der Twitter-Website können Sie beispielsweise das Twitter-Suche-Widget anpassen und herunterladen². Sie können Titel, Breite, Höhe, einige Farben und den Suchbegriff frei auswählen. Dann erstellen wir das JavaScript, das notwendig ist, um das Widget in unsere eigene Seite zu integrieren. Das Endergebnis sieht aus wie hier gezeigt:

kiosk/widget.html

```
Zeile 1 <html>
-   <head>
-     <style>
-       .twtr-widget {
-         text-align: center;
-       }
-       .twtr-doc {
-         text-align: left;
-         margin: auto;
-       }
-     </style>
-
-     <script charset="utf-8" src="http://widgets.twimg.com/j/2/
-                               widget.js">
-   </script>
15
-   <script>
-     new TWTR.Widget({
-       version: 2,
-       type: 'search',
-       search: 'maik_schmidt pragprog raspberrypi',
-       interval: 10000,
-       title: 'Have Fun With The Raspberry Pi',
```

2) <https://twitter.com/settings/widgets/new/search>

```

-
-         subject: 'Raspberry Pi: A Quick-Start Guide',
-
25      width: 1700,
-         height: 800,
-         theme: {
-             shell: {
-
-                 background: '#8ec1da',
-                 color: '#ffffff'
30             },
-             tweets: {
-
-                 background: '#ffffff',
-                 color: '#444444',
-                 links: '#1985b5'
-
35             }
-         },
-         features: {
-             scrollbar: false,
-             loop: true,
40             live: true,
-             behavior: 'default'
-
-         }
-     }).render().start();
-     </script>
45   </head>
-
-
-   <body>
-   </body>
- </html>

```

Die Twitter-Website generiert den kompletten JavaScript-Code für Sie. Sie müssen nur etwas Basis-HTML hinzufügen. Die Zeilen 4 bis 10 enthalten außerdem Stilvorlagen, um das Widget zu zentrieren. Wenn Sie das Widget später neu konfigurieren, müssen Sie es nicht erneut generieren. Die wichtigsten Konfigurationsparameter befinden sich in den Zeilen 20 bis 25. Hier können Sie den Widget-Titel, die Suchbegriffe, seine Abmessungen und das Aktualisierungsintervall angeben. Das Aktualisierungsintervall müssen Sie in Millisekunden festlegen. Beachten Sie, dass Twitter das Intervall nicht garantiert. Das Widget könnte also auch weniger häufig aktualisiert werden, als Sie es gern möchten.

Um den Widget-Code auf den Pi zu bekommen, öffnen Sie einen Texteditor wie *nano* und geben Sie ihn ein. Einfacher ist es jedoch, das Zip-Archiv, das den Quellcode dieses Buchs enthält, von der Begleitwebsite³

3) <http://www.dpunkt.de/leseproben/4192/msraspi-code.zip>

herunterzuladen. In der Voreinstellung lädt Midori alle Dateien in das /tmp-Verzeichnis herunter. Öffnen Sie also, nachdem Midori die Datei heruntergeladen hat, ein Terminalfenster und geben Sie die folgenden Befehle ein:

```
pi@raspberrypi ~ $ cd /tmp  
pi@raspberrypi ~ $ unzip msraspi-code.zip
```

Jetzt finden Sie den Widget-Code unter /tmp/code/kiosk/widget.html. Bevor Sie ihn in Midori ausführen, öffnen Sie ihn mit einem Texteditor wie *nano* und passen Sie den Suchbegriff, die Breite und die Höhe wie gewünscht an.

```
pi@raspberrypi ~ $ nano /tmp/code/kiosk/widget.html
```

Wenn Sie mit Ihren Anpassungen fertig sind, drücken Sie in *nano* Strg-X, bestätigen zunächst, dass Sie Ihre Änderungen speichern möchten, und dann bestätigen Sie noch den Dateinamen.

Um das Kiosk-Widget auszuführen, starten Sie den LXDE-Desktop auf Ihrem Pi und danach den Midori-Browser. Anschließend wählen Sie in Midori das Menü *Öffnen* und die Datei mit dem Widget-Code aus. Das Widget wird sofort gestartet. Um einen Gesamteindruck zu bekommen, sollten Sie jedoch mit F11 den Vollbildmodus von Midori aktivieren.

Das Widget bedeckt jetzt den gesamten Bildschirm und aktualisiert sich automatisch. Mit Strg-+ können Sie die Schriftgröße vergrößern. Ohne eine einzige Zeile Code zu schreiben, haben wir den Pi in ein Kiosk-System verwandelt.

5.2 Webseiten automatisch aktualisieren

Das Twitter-Widget aus dem vorherigen Abschnitt hat eine nette Funktion: Es aktualisiert den Inhalt automatisch mittels JavaScript. Das bieten uns leider nicht alle Webseiten, sodass wir einige von ihnen selbst aktualisieren müssen. Eine Lösung besteht darin, ein Meta-Tag im Head-Abschnitt der HTML-Seite einzufügen.

```
<meta http-equiv="refresh" content="120"></meta>
```

Das gezeigte Element lädt die Seite alle 120 Sekunden neu. Das ist eine einfache Lösung, die aber nur funktioniert, wenn Sie die Seite verändern dürfen und Zugriff auf den betreffenden Server haben. Die beste Lösung wäre, dem Browser mitzuteilen, dass er die Seite in einem bestimmten Intervall neu laden soll. Die meisten Browser bieten solch eine Funktion und sie wird üblicherweise Kiosk-Modus genannt. Auch Midori hat einen Kiosk-Modus. Um ihn zu verwenden, müssen Sie Midori von der Eingabeaufforderung aus starten. Mit der Option `-a` geben Sie die Adresse an, die Midori anzeigen soll. Um das Aktualisierungsintervall zu setzen, verwenden wir die Option `-i`, und den Vollbildmodus schalten wir mir `-e` ein. Die Befehlszeile sieht dann wie folgt aus:

```
pi@raspberrypi ~ $ midori -i 30 -e Fullscreen \
-a "http://twitter.com/search?q=pragprog"
```

Dadurch werden die Ergebnisse einer Twitter-Suche nach dem Begriff *pragprog* angezeigt und die Seite alle dreißig Sekunden aktualisiert. Vergrößern Sie die Schrift, indem Sie ein paar Mal Strg+- eingeben, und Sie sind fertig. Midori unterstützt viele weitere Optionen, um sich von außen steuern zu lassen. Sie können sie mit folgendem Befehl auflisten:

```
pi@raspberrypi ~ $ midori --help-execute
```

5.3 Die nächsten Schritte

In diesem Kapitel haben Sie gelernt, wie Sie aus dem Pi in nur wenigen Schritten ein Kiosk-System machen. Suchen Sie in Ihrem Unternehmen nach möglichen Einsatzgebieten. Zum Beispiel können Sie den Status Ihrer wichtigsten Systeme auf einem großen Bildschirm anzeigen lassen. Sie könnten auch die aktuelle Anzahl von Kunden oder Aufträgen ausgeben. Natürlich können Sie auch konservativ sein und einfach eine Diashow laufen lassen, die zeigt, wie wunderbar und toll Ihre Firma ist.

6 Netzwerken mit dem Pi

Wie jeder andere Computer auch wird der Pi bedeutend spannender, sobald Sie ihn mit einem Netzwerk verbinden. Plötzlich können Sie den Pi für alltägliche Aufgaben wie das Surfen im Netz und zum Twittern verwenden. Sie können mittels SSH auf Ihren Pi zugreifen und sicher und bequem auch von einem anderen Computer damit arbeiten. Sie können sogar mit Ihrem PC über das Netzwerk auf den Desktop des Pi zugreifen und umgekehrt.

Zusätzlich können Sie den Pi als günstigen und leistungsfähigen Webserver einsetzen, der nicht nur statischen Inhalt liefert, sondern auch z.B. in PHP geschriebene Anwendungen ausführen kann.

6.1 Erledigen Sie alltägliche Webaufgaben

Vermutlich ist es für Sie normal, viele Aufgaben nur mit Ihrem Webbrowsert zu erledigen, so wie das Lesen von E-Mails und RSS-Feeds, Anschauen von Videos, Twittern usw. Das ist möglich, weil die meisten modernen Browser HTML5, JavaScript, Flash und Java unterstützen. Ohne diese Technologien wäre das Web immer noch auf dem Stand von 1995.

All diese Annehmlichkeiten funktionieren nur mit modernen Browsern wie Google Chrome oder Mozilla Firefox. Auch wenn Sie Chromium auf Raspbian¹ installieren können, wird es noch eine Weile dauern, bis es hinreichend schnell auf dem Pi läuft. Außerdem werden noch nicht alle Funktionen (z.B. Video) unterstützt. Die größte Einschränkung besteht aber im geringen Arbeitsspeicher des Pi, von dem moderne Browser eine Menge verschlingen. Der mit Debian für den Pi ausgelieferte Browser ist

1) <http://hexxeh.net/?p=328117859>

Midori², ein sehr guter Browser, der nur wenig Speicher benötigt. Leider steckt er noch in den Kinderschuhen und weist einige Einschränkungen auf. Flash und Java werden z.B. noch nicht auf dem Pi unterstützt. Wenn Sie also auf eine Website kommen, die Flash oder ein Java-Applet enthält, funktioniert sie noch in keinem Browser auf dem Pi.

Ich will aber Chrome

Zur Drucklegung dieses Buchs ist Chromium noch neu und weit entfernt von Stabilität. Außerdem ist es recht langsam. Wenn Sie Chromium dennoch auf dem Pi installieren möchten, verwenden Sie folgenden Befehl:

```
pi@raspberrypi ~ $ sudo bash <(curl -sL http://goo.gl/5vuJI)
```

Dann starten Sie Chromium von einem Terminal mit folgendem Befehl:

```
pi@raspberrypi ~ $ chrome --disable-ipv6
```

Damit Chromium so schnell läuft wie möglich, verwenden Sie maximal 64 MB für den Grafikspeicher (siehe Abschnitt »Anpassen der Speicherkonfiguration« in Kap. 3). Auch empfiehlt es sich, den Pi zu übertakten.^a

a. Siehe »*Den Pi beschleunigen (Overclocking)*«, S. 32 ff.

Obwohl Midori HTML5, CSS3 und JavaScript versteht, kann es nicht alle modernen Webseiten korrekt interpretieren. Google Mail funktioniert z.B. von Haus aus nicht korrekt, da Midori nicht den kompletten JavaScript-Code richtig versteht. Außerdem benötigt Midori etwas Zeit, um die Standardansicht von Google Mail zu berechnen. Sie können diesen Zustand verbessern, indem Sie JavaScript über das Midori-Menü *Prefences > Behaviour* deaktivieren und die HTML-Ansicht von Google Mail aktivieren. Selbst dann können Sie noch nicht korrekt mit allen Eingabefeldern arbeiten, z.B. mit den Feldern, um E-Mail-Empfänger anzugeben. Es ist am besten, Adressen nur aus dem Adressbuch auszuwählen oder sie aus einem anderen Textfeld zu kopieren und dann in die Felder *An* und *CC* einzufügen.

2) <http://twotoasts.de/index.php/midori/>

JavaScript zu deaktivieren kann die Benutzbarkeit auch anderer Webseiten verbessern. Wenn Sie JavaScript in Ihrem Browser deaktivieren, liefern viele Websites eine einfache HTML-Version. Diese HTML-Version verfügt nicht über all den Schnickschnack der ursprünglichen Seite, dafür können Sie sie aber verwenden.

Für Twitter gilt Ähnliches. Midori kann zwar den Inhalt darstellen, ist dabei aber langsam. Eine gute Alternative ist die Twitter-Seite für Mobilgeräte.³ Sie bietet nicht alle Funktionen der Originalseite, aber Twitter lässt sich darüber bedienen und das Ganze läuft auf Midori.

Ein anderer Trick besteht darin, den User-Agent von Midori zu verändern. Alle Webbrowers senden mit jeder Anfrage einen Identifikator, der dem Webserver genau mitteilt, um welche Art Browser es sich handelt. Dieser Identifikator wird User-Agent genannt und manche Webseiten passen ihre Daten daran an. Zum Beispiel generieren manche Webseiten eine Fehlermeldung, wenn sie den User-Agent nicht erkennen. In Midori verstehen Sie den User-Agent im Menü *Preferences > Network*. Hier können Sie angeben, dass Midori vorgibt, Mozilla Firefox zu sein oder Safari oder ein iPhone.

Manche Seiten werden aber trotzdem nicht funktionieren, egal, was Sie versuchen. YouTube⁴ zum Beispiel benötigt Video-Unterstützung im Browser, sei es HTML5 oder Flash. Zurzeit unterstützt Midori auf dem Pi keines von beiden, sodass YouTube nicht korrekt angezeigt wird. Sie haben ebenfalls kein Glück, wenn eine Website auf JavaScript-Code basiert, den Midori nicht versteht, oder wenn sie ein Java-Applet benötigt. Glücklicherweise sind Java-Applets heute nicht besonders beliebt, aber einige Onlinebanking-Seiten verwenden sie.

Zusätzlich ist der Ressourcenverbrauch von Midori hoch, vor allem bei der CPU. Es geschieht oft, dass Midori beinahe die gesamte Rechenleistung der CPU benötigt, um eine Website darzustellen. Manchmal müssen Sie mehrere Minuten warten, bis die Seite berechnet ist.

3) <http://mobile.twitter.com>

4) <http://youtube.com>

Wenn Sie im Hinterkopf behalten, dass Sie JavaScript deaktivieren können und dass es von den populären Websites meist eine mobile Version gibt, reicht Midori aber in der Mehrzahl der Fälle aus. Und natürlich könnte die Beliebtheit des Pi in naher Zukunft zu Verbesserungen führen.

6.2 Secure Shell und Pi

Es ist sehr wahrscheinlich, dass Sie Ihren Pi mit einem Netzwerk verbinden, um von anderen Computern darauf zuzugreifen und umgekehrt. Eines der besten Verfahren, über das zwei Computer miteinander kommunizieren, ist Secure Shell (SSH), ein Netzwerkprotokoll für den sicheren Austausch von Daten. Debian bringt alles mit, was für SSH notwendig ist; Sie müssen nur einige Einstellungen konfigurieren.

Mittels Passwort auf den Pi zugreifen

Wenn Sie mit dem Pi nur auf andere Computer zugreifen wollen, müssen Sie nichts konfigurieren. Zum Beispiel können Sie sich als Administrator am Host `maik-schmidt.de` anmelden, indem Sie SSH auf dem Pi mit dem Namen und Passwort des anderen Computers starten.

```
pi@raspberrypi ~ $ ssh admin@maik-schmidt.de
admin@maik-schmidt.de's password:
Last login: Wed Jan 9 09:41:34 2013 from 94.221.82.250
admin@maik-schmidt.de:~$ exit
logout
Connection to maik-schmidt.de closed.
```

Wenn Sie jedoch mit SSH auf den Pi zugreifen wollen, müssen Sie auf dem Pi zuerst über Raspi-config den SSH-Server aktivieren.

```
pi@raspberrypi ~ $ sudo raspi-config
```

Wählen Sie den Menüpunkt `ssh` und aktivieren Sie den SSH-Server. Klicken Sie dann auf die Schaltfläche *Finish* und verlassen Sie Raspi-config. Nach dem Neustart enthält das Startprotokoll des Pi eine neue Meldung.

```
Starting OpenBSD Secure Shell server: sshd
My IP address is 192.168.2.109
```

Das bedeutet, dass Sie auf den Pi jetzt mittels SSH zugreifen können und dass die IP-Adresse des Pi 192.168.2.109 ist. In Ihrem Fall wird die Adresse vermutlich eine andere sein. Wenn Sie später die IP-Adresse Ihres Pi herausfinden müssen, geben Sie folgenden Befehl ein:

```
pi@raspberrypi ~ $ ip addr | grep 'inet .* eth0'  
inet 192.168.2.109/24 brd 192.168.2.255 scope global eth0
```

Die erste IP-Adresse im Text ist die Adresse Ihres Pi. Mit dieser Adresse können Sie Ihren Pi von allen Computern Ihres Netzwerks erreichen.

Auf einem Mac oder unter Linux starten Sie SSH von der Eingabeaufforderung aus und übergeben die IP-Adresse und das Passwort des Pi-Anwenders.

```
maik> ssh pi@192.168.2.109  
pi@192.168.2.109's password:  
Linux raspberrypi 3.6.11+ #348 PREEMPT Tue Jan 1 16:33:22 GMT 2013  
armv6l  
  
The programs included with the Debian GNU/Linux system are free  
software; the exact distribution terms for each program are described  
in the individual files in /usr/share/doc/*copyright.
```

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.
```

```
Type 'startx' to launch a graphical session
```

```
Last login: Sat Jan 12 19:48:06 2013  
pi@raspberrypi ~ $ exit  
logout  
Connection to 192.168.2.109 closed.
```

Um von Windows aus auf den Pi zuzugreifen, benötigen Sie einen SSH-Client. Einer der besten ist PuTTY⁵. Es ist ein sehr kleines Programm, das Sie nicht einmal installieren müssen. Laden Sie die ausführbare Datei herunter, starten Sie sie und Sie sehen den folgenden Konfigurationsbildschirm:

5) <http://www.chiark.greened.org.uk/~sgtatham/putty/download.html>

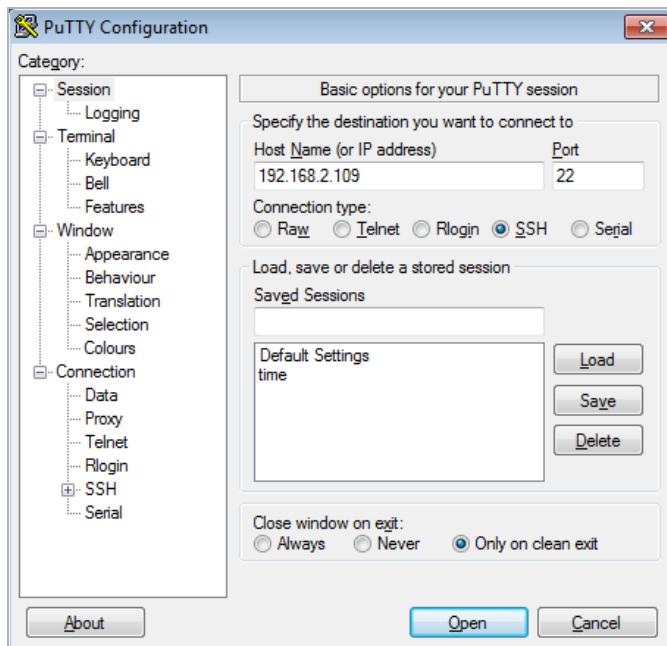


Abb. 6–1 Konfigurationsbildschirm von PuTTY

Sie können in PuTTY eine Menge konfigurieren und für jede Art von Verbindung lässt sich die Konfiguration auch speichern. Um sich auf dem Pi einzuloggen, müssen Sie nur seine IP-Adresse eingeben und auf die Schaltfläche *Open* klicken. Dann erhalten Sie den normalen Login-Prompt des Pi, so wie in Abbildung 6–2 dargestellt.

A screenshot of a PuTTY terminal window titled '192.168.2.109 - PuTTY'. The window displays a Linux login prompt: 'login as: pi'. The user 'pi' enters their password. Below the password entry, the terminal shows the system information: 'Linux raspberrypi 3.1.9+ #90 Wed Apr 18 18:23:05 BST 2012 armv6l'. It then displays the standard Debian GNU/Linux copyright notice. At the bottom, it shows the user has logged in from '192.168.2.100' and is currently at the command prompt 'pi@raspberrypi:~\$'. A small green cursor is visible at the end of the command line.

Abb. 6–2 Der Zugriff auf den Pi von Windows ist leicht.

Mit einem Public-Private-Schlüsselpaar auf den Pi zugreifen

Wenn Sie oft mit SSH auf Ihren Pi zugreifen müssen, kann es lästig werden, beim Einloggen immer das Passwort eingeben zu müssen. Viel bequemer ist die Public-Key-Verschlüsselung. Dazu müssen Sie auf Ihrem PC einen Schlüssel generieren. Der Schlüssel besteht aus zwei Teilen, einem geheimen und einem öffentlichen Schlüssel, wobei Sie den öffentlichen auf den Pi kopieren. Wenn Sie sich das nächste Mal von Ihrem PC aus am Pi anmelden, kann SSH Ihre Identität dadurch überprüfen, dass die beiden Schlüsselkomponenten zusammengehören. Wenn Sie von mehreren Computern auf den Pi zugreifen wollen, müssen Sie die folgenden Schritte auf jedem von ihnen ausführen.

Bevor Sie ein neues Schlüsselpaar generieren, sollten Sie prüfen, ob Sie nicht schon eines haben. Unter Linux oder Mac OS X öffnen Sie ein Terminalfenster und geben folgenden Befehl ein:

```
maik> ls ~/.ssh/id_rsa.pub  
/Users/maik/.ssh/id_rsa.pub
```

Die Datei `id_rsa.pub` enthält den öffentlichen Schlüssel und der vorher gezeigte Befehl versucht, die Datei anzuzeigen. Wenn die Ausgabe so wie die vorherige aussieht, haben Sie bereits einen Schlüssel. Sie können dann die Generierung auslassen und den öffentlichen Schlüssel auf den Pi kopieren, wie im Folgenden beschrieben. Wenn Sie stattdessen eine Meldung wie »No such file oder directory« erhalten, müssen Sie einen Schlüssel wie folgt generieren:

```
maik> ssh-keygen -t rsa -C "your_email@youremail.com"  
Generating public-private rsa key pair.  
Enter file in which to save the key (/Users/maik/.ssh/id_rsa):  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in  
/Users/mschmidt/.ssh/id_rsa.  
Your public key has been saved in  
/Users/mschmidt/.ssh/id_rsa.pub.  
The key fingerprint is:  
f0:09:09:49:42:46:42:6f:42:3b:42:44:42:09:6a:e8  
your_email@youremail.com  
The key's randomart image is:
```

```
+--[ RSA 2048]----+
| ..0...
|+ ..o + .
|o.o + B o
|.+ o o B .
| E   = S .
| .   . o
|
| .
|
+-----+
```

Dadurch wird in Ihrem Stammverzeichnis ein Schlüsselpaar erzeugt. Sie finden den öffentlichen Schlüssel in einer Datei namens `id_rsa.pub`. Jetzt müssen Sie diese Datei auf den Pi übertragen, auf dem SSH eine Liste der autorisierten Schlüssel in einer Datei namens `.ssh/authorized_keys` im Stammverzeichnis des pi-Anwenders unterhält. Die folgenden Befehle fügen die Datei `id_rsa.pub` in die pi-Liste autorisierter Schlüssel ein.

```
maik> scp ~/.ssh/id_rsa.pub pi@192.168.2.109:/tmp
maik> ssh pi@192.168.2.109 "cat /tmp/id_rsa.pub >> ~/.ssh/
authorized_keys"
```

Der erste Befehl kopiert `id_rsa.pub` in das `/tmp`-Verzeichnis auf dem Pi und der zweite hängt den Inhalt der Datei an die Datei `~/.ssh/authorized_keys` an. Wenn Sie nicht vorhaben, mehrere Schlüssel in der Datei `authorized_keys` zu speichern, können Sie die Datei `id_rsa.pub` natürlich auch direkt kopieren.

```
maik> scp ~/.ssh/id_rsa.pub pi@192.168.2.109:/home/pi/.ssh/
authorized_keys
```

Unter Windows benötigen Sie einige Zusatzwerkzeuge von der PuTTY-Downloadseite, um die Schlüssel zu erzeugen und auf den Pi zu kopieren. In Abbildung 6–3 sehen Sie, wie die Anwendung PuTTYgen Schlüssel erzeugt.

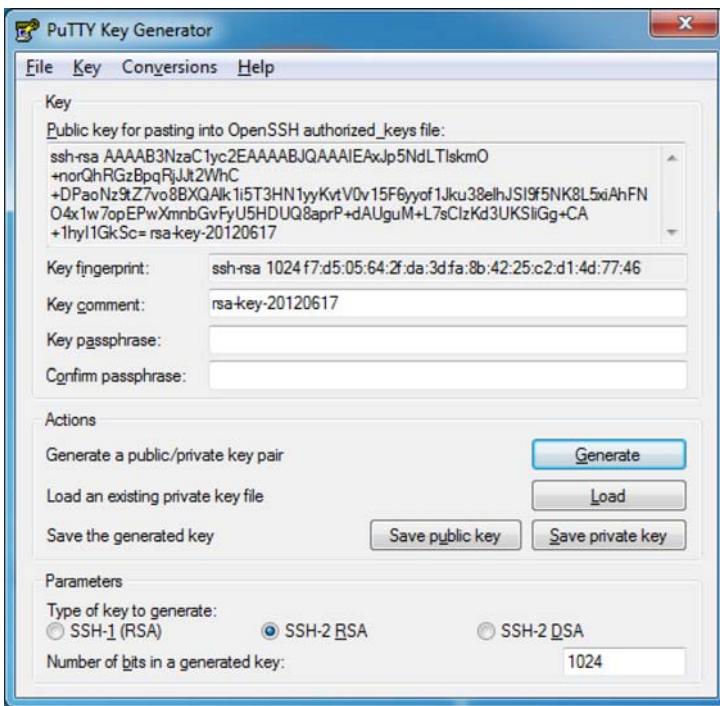


Abb. 6–3 PuTTYgen erzeugt Schlüssel unter Windows.

Um die generierte öffentliche Schlüsseldatei zu kopieren, verwenden sie PSPC. Es funktioniert genau wie scp. Von einer Eingabeaufforderung starten Sie also folgenden Befehl:

```
C:\> pscp id_rsa.pub pi@192.168.2.109:/home/pi/.ssh/authorized_keys
```

Jetzt ist Ihr Pi ein vollwertiges Mitglied Ihres Netzwerks.

6.3 Desktops mit Pi gemeinsam nutzen

Das Einloggen auf dem Pi mittels SSH ist bequem und bietet Ihnen viele neue Möglichkeiten. Zum Beispiel können Sie auf das Pi-Dateisystem zugreifen, Prozesse starten und anhalten sowie überwachen, was auf dem Pi gerade vorgeht. Der größte Nachteil der SSH-Lösung besteht darin, dass sie bislang nur in einem Textterminal funktioniert.

Sie können diese Einschränkung leicht umgehen und Desktop, Tastatur und Maus des Pi von einem anderen Computer aus steuern. Die Lösung heißt Virtual Network Computing (VNC)⁶, eine Technologie, die den gesamten Bildschirminhalt und alle Maus- und Tastatureingaben von einem Computer auf einen anderen überträgt.

Um VNC zu nutzen, benötigen Sie einen VNC-Client und einen VNC-Server. Der Server läuft auf der Maschine, die Sie steuern möchten, und der Client auf dem Steuercomputer. Wenn Sie also den Pi mit Ihrem PC steuern möchten, müssen Sie auf Ihrem Pi einen VNC-Server installieren. Sie können unter verschiedenen auswählen, einer der besten ist jedoch TightVNC⁷. Er ist kostenlos, für alle wichtigen Plattformen erhältlich und Sie können ihn mit apt-get installieren.

```
pi@raspberrypi ~ $ sudo apt-get install tightvncserver  
pi@raspberrypi ~ $ tightvncserver
```

```
You will require a password to access your desktops.
```

```
Password:
```

```
Verify:
```

```
Would you like to enter a view-only password (y/n)? n
```

```
New 'X' desktop is raspberrypi:1
```

```
Creating default startup script /home/pi/.vnc/xstartup  
Starting applications specified in /home/pi/.vnc/xstartup  
Log file is /home/pi/.vnc/raspberrypi:1.log
```

Wenn Sie tightvncserver das erste Mal starten, werden Sie nach einem Passwort gefragt. Sie geben dieses Passwort später in den VNC-Client ein, um zu verhindern, dass fremde Personen Zugriff auf Ihren Pi erhalten. Zusätzlich ermöglicht Ihnen TightVNC, optional ein Passwort nur zum Zuschauen einzurichten. Dieses Passwort erlaubt es Dritten, zu sehen, was auf dem Bildschirm erscheint, sie bekommen aber keinen Zugriff auf Tastatur oder Maus. Das ist beispielsweise für Präsentationen sinnvoll.

6) <http://en.wikipedia.org/wiki/Vnc>

7) <http://www.tightvnc.com/>

Nachdem Sie die Passwörter definiert haben, erzeugt TightVNC einen neuen virtuellen Bildschirm, auf den Sie von Ihrem PC oder Mac zugreifen können. Das Tolle an VNC ist, dass Sie so viele virtuelle Bildschirme einrichten können, wie Sie wollen. Diese Bildschirme müssen nicht notwendigerweise den physischen Bildschirmen entsprechen. Sie sind rein virtuell, sodass viele Anwender Zugriff auf Ihren Pi haben können und alle erhalten ihre eigene Desktop-Umgebung.

Um auf einen virtuellen Bildschirm zuzugreifen, benötigen Sie zwei Dinge: die IP-Adresse des Pi und den Port des Bildschirms. Der VNC-Basisport ist 5900. Für einen Zugriff auf Bildschirm 1 müssen Sie Port 5901 verwenden. Um auf den mit den vorherigen Befehlen erzeugten Bildschirm zuzugreifen, müssen Sie die Netzwerkadresse 192.168.2.109:5901 verwenden. Vergessen Sie nicht, dass Ihre IP-Adresse vermutlich anders lautet.

Nachdem Sie jetzt die Adresse des VNC-Servers des Pi kennen, können Sie darauf vom Mac oder PC mittels eines VNC-Client zugreifen. Auf dem Mac ist das sehr leicht, da er schon mit einem VNC-Client geliefert wird. Sie können mit Safari direkt auf einen VNC-Server zugreifen. Geben Sie einfach die Webadresse `vnc://192.168.2.109:5901` ein und Safari öffnet den virtuellen Bildschirm für Sie. Anschließend geben Sie das vorher definierte Passwort ein und das war's. Das Ergebnis sehen Sie in Abbildung 6–4.

Unter Windows und Linux ist der Vorgang recht ähnlich, aber Sie müssen zuerst einen VNC-Client installieren, was einfach ist, da TightVNC für Windows und Linux erhältlich ist und auch einen Client enthält.

Den Desktop Ihres PC oder Mac vom Pi aus zu steuern ist ebenso einfach. Zuerst müssen Sie auf Ihrem PC einen VNC-Server installieren und auch hier ist TightVNC die erste Wahl für Windows und Linux. Auf dem Mac ist es noch einfacher, da Mac OS X einen integrierten VNC-Server bietet, den Sie einfach aktivieren müssen. In den Systemeinstellungen wählen Sie *Sharing* und aktivieren *Screen Sharing*. Klicken Sie auf die Schaltfläche *Computer Settings*, um ein Passwort festzulegen (Sie sehen die Voreinstellungen in Abb. 6–5).

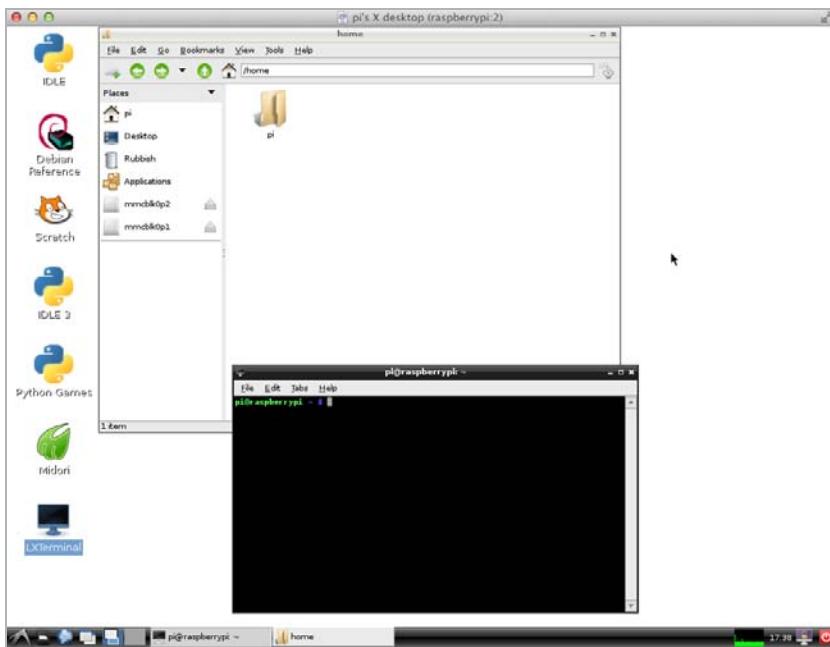


Abb. 6–4 Den Pi vom Mac aus steuern

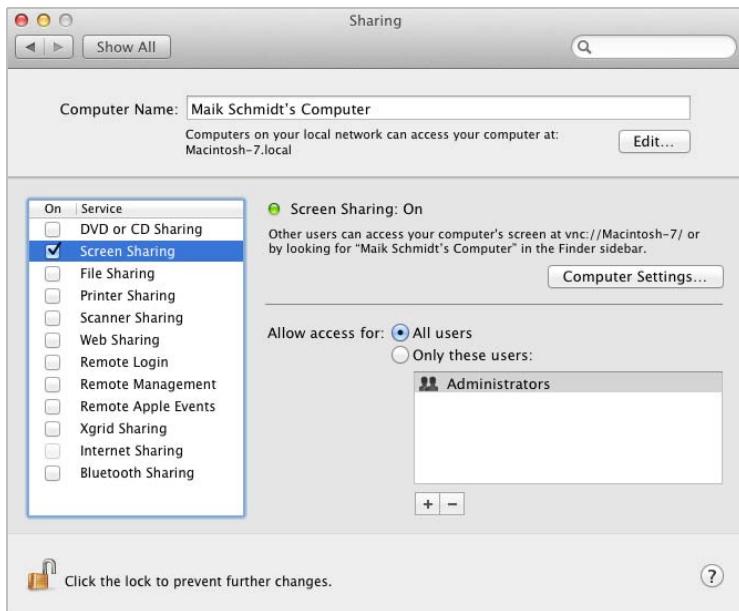


Abb. 6–5 Den Mac zu steuern ist einfach.

Jetzt benötigen Sie einen VNC-Client auf dem Pi. Hier bietet sich xtightvncviewer an. Installieren Sie ihn mit apt-get.

```
pi@raspberrypi ~ $ sudo apt-get install xtightvncviewer
```

Öffnen Sie dann ein Terminalfenster auf dem Pi-Desktop, starten Sie den Client und übergeben Sie ihm die IP-Adresse und den VNC-Port.

```
pi@raspberrypi ~ $ xtightvncviewer 192.168.2.100:5900
```

In Abbildung 6–6 sehen Sie den Mac-Desktop innerhalb eines Fensters auf dem Pi-Desktop. Wenn das nicht funktioniert, prüfen Sie, ob IP-Adresse und Portnummer stimmen. Normalerweise ist sie 5900, kann aber bei verschiedenen VNC-Servern variieren.



Abb. 6–6 Den Mac vom Pi aus steuern

6.4 Machen Sie aus dem Pi einen Webserver

Obwohl der Pi im Vergleich zu modernen Webservern wie ein Spielzeug aussieht, ist er doch leistungsfähig genug, um in Ihrem lokalen Netzwerk interessante Informationen bereitzustellen. Er kann nicht nur statische Webseiten liefern, sondern auch mittels Datenbanken und Webanwendungen dynamische Inhalte generieren. Zusätzlich bietet er mittels Webtechnologien sogar Zugriff auf seine GPIO-Ports.

Der erste Schritt, um den Pi in einen Webserver zu verwandeln, ist ein HTTP-Server, ein Netzwerkdienst, der das Hypertext-Transfer-Protokoll (HTTP) versteht.

Sie können unter mehreren hervorragenden Produkten auswählen, z.B. den Apache HTTP-Server⁸ oder Nginx⁹, aber auch Lighttpd¹⁰ ist eine gute Wahl für den Pi, weil er nur wenig Speicher benötigt.

Lighttpd zu installieren und auszuführen ist kinderleicht.

```
pi@raspberrypi ~ $ sudo apt-get install lighttpd
```

Nach der Installation läuft Lighttpd bereits und Sie können mit dem Webbrower Ihres PCs direkt auf die IP-Adresse des Pi gehen. Abbildung 6–7 zeigt, wie Sie vom Server begrüßt werden.

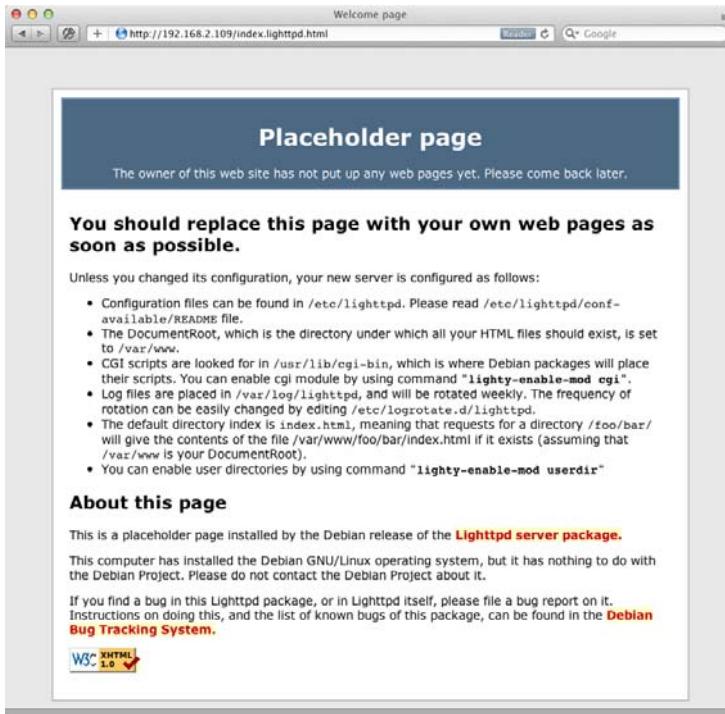


Abb. 6–7 Der Willkommensbildschirm von Lighttpd

-
- 8) <http://httpd.apache.org/>
 - 9) <http://nginx.org/>
 - 10) <http://www.lighttpd.net/>

Um Ihre eigenen Webseiten zu erstellen, müssen Sie sie ins Dokumentverzeichnis von Lighttpd legen, in dem sich alle Dateien befinden, die zur Website gehören. Das Standardverzeichnis von Lighttpd ist /var/www. Sie sollten sicherstellen, dass nur Mitglieder der Betriebssystemgruppe *www-data* eine Schreibberechtigung dafür besitzen. Die folgenden Befehle fügen den Pi-Anwender der Gruppe *www-data* hinzu und setzen den Zugriff auf das Verzeichnis /var/www entsprechend:

```
pi@raspberrypi ~ $ sudo adduser pi www-data
pi@raspberrypi ~ $ sudo chown -R www-data:www-data /var/www
pi@raspberrypi ~ $ sudo chmod -R 775 /var/www
```

Beim nächsten Login kann der Pi-Anwender neue Webseiten erstellen. Sie können das mit jedem Texteditor wie beispielsweise *nano* machen. Der folgende Befehl erzeugt eine neue Datei namens *index.html*, die die Startseite Ihrer ersten Website darstellt.

```
pi@raspberrypi ~ $ nano /var/www/index.html
```

Geben Sie den folgenden Text ein:

Networking/index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hello, world!</title>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>
```

Nachdem Sie das getan haben, drücken Sie Strg-X, um *nano* zu verlassen. Drücken Sie »Y«, um zu bestätigen, dass Sie die Datei speichern wollen, und drücken Sie dann Return, um den Dateinamen zu bestätigen. Anschließend gehen Sie mit Ihrem Browser auf die neue Website und sehen ein Ergebnis wie in Abbildung 6–8:



Abb. 6–8 HTML-Seite: Hello World

Wenn es Ihnen lieber ist, können Sie die Datei `index.html` auch auf Ihrem PC bearbeiten und sie anschließend auf den Pi kopieren (vergessen Sie nicht, die IP-Adresse durch die Ihres Pi zu ersetzen).

```
maik> scp index.html pi@192.168.2.109:/var/www
```

Mit nur einer Handvoll Befehlen haben Sie den Pi in einen vollwertigen Webserver verwandelt, der statische Inhalte wie HTML-Seiten liefern kann. Das ist nützlich und nett, aber manchmal benötigen Sie auch dynamische Inhalte. So wollen Sie vielleicht Daten aus einer Datenbank in Ihre Seiten einbetten oder vielleicht sogar Umweltdaten, die mit an den Pi angeschlossenen Sensoren ermittelt werden.

Um dynamischen Inhalt zu erzeugen, benötigen wir eine Programmiersprache. Sie haben die Wahl aus einem breiten Angebot. Für den Pi ist PHP¹¹ eine gute Wahl, da es nicht viele Ressourcen verbraucht und Sie es einfach installieren können.

```
pi@raspberrypi ~ $ sudo apt-get update  
pi@raspberrypi ~ $ sudo apt-get install php5-cgi  
pi@raspberrypi ~ $ sudo lighty-enable-mod fastcgi  
pi@raspberrypi ~ $ sudo /etc/init.d/lighttpd force-reload
```

Mit diesen Befehlen installieren Sie einen PHP-Interpreter und aktivieren das Modul FastCGI im Lighttpd-Server. FastCGI¹² beschleunigt dynamische Webseiten außerordentlich. Es ist also eine gute Idee, es zu aktivie-

11) <http://www.php.net/>

12) <http://www.fastcgi.com/>

ren. Um die Installation abzuschließen, müssen Sie noch die Konfigurationsdatei von Lighttpd editieren.

```
pi@raspberrypi ~ $ sudo nano /etc/lighttpd/lighttpd.conf
```

Fügen Sie am Ende der Datei die folgenden Zeilen hinzu, um PHP und FastCGI zu aktivieren:

```
fastcgi.server = (".php" => ((  
    "bin-path" => "/usr/bin/php-cgi",  
    "socket" => "/tmp/php.socket"  
)))
```

Wenn Sie die Änderungen an der Konfigurationsdatei gespeichert haben, starten Sie den Webserver neu.

```
pi@raspberrypi ~ $ sudo service lighttpd restart
```

Um zu testen, ob alles wie gewünscht läuft, erzeugen Sie eine Datei namens /var/www/index.php mit dem folgenden Inhalt:

Networking/index.php

```
<?php  
    phpinfo();  
?>
```

Gehen Sie nun mit Ihrem Webbrower auf die neu erzeugte Datei und Sie erhalten eine ähnliche Ausgabe wie in Abbildung 6–9.

Es handelt sich um die PHP-Infoseite, die viele Informationen über das System enthält, auf dem es momentan läuft. PHP generiert diese Seite dynamisch. Sie sehen, dass alles prima funktioniert. Jetzt können Sie anfangen, Ihre eigenen Webanwendungen mit dem Pi zu erstellen. In Kapitel 9 erzeugen Sie eine Webanwendung, die externe, an den Pi angeschlossene Hardware steuert.

The screenshot shows a web browser window titled "phpinfo()" with the URL "192.168.2.105/index.php". The page content is a PHPinfo dump. At the top, it says "PHP Version 5.4.4-9" and features a large "php" logo. Below this is a table with numerous rows of configuration parameters. At the bottom of the page, there is a note about Zend Engine usage and a "Powered By Zend Engine 2" logo.

System	Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l
Build Date	Nov 9 2012 06:13:16
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (.php.ini) Path	/etc/php5/cgi
Loaded Configuration File	/etc/php5/cgi/php.ini
Scan this dir for additional .ini files	/etc/php5/cgi/conf.d
Additional .ini files parsed	/etc/php5/cgi/conf.d/10-pdo.ini
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend Extension Build	API20100525,NTS
PHP Extension Build	API20100525,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.4.0, Copyright (c) 1998-2012 Zend Technologies

Powered By

Abb. 6–9 Dynamische Webseiten mit dem Pi

6.5 Den Pi um WLAN erweitern

Drahtlose Netzwerke sind allgegenwärtig. Cafés, Flughäfen und Hotels bieten ihren Kunden heutzutage kostenfreien WLAN-Zugang. Vermutlich haben Sie zu Hause bereits ein WLAN, sodass Sie per Smartphone bequem auf die wichtigsten Geräte Ihres Hauses zugreifen können, während Sie mit Ihrer Familie im Garten grillen. Unter Windows oder Mac OS X müssen Sie keine großen Anstrengungen unternehmen, um einem WLAN beizutreten, da es fast automatisch geschieht.

Mit dem Pi ist das anders und es hängt maßgeblich von Ihrem WLAN-Adapter ab, ob die Verbindung zu einem drahtlosen Netzwerk automatisch geschieht oder manuell erfolgen muss. Sie benötigen zuerst also einen WLAN-Adapter für den USB-Port, da die Pi-Hardware WLAN von Haus aus nicht unterstützt.

Stecken Sie den WLAN-Adapter in einen der USB-Ports des Pi und starten Sie den Pi. Nach dem Anmelden starten Sie den Desktop LXDE mittels `startx`:

```
pi@raspberrypi ~ $ startx
```

Auf dem Desktop finden Sie ein Programm mit dem Namen *WiFi Config*. Starten Sie es per Doppelklick. Abbildung 6–10 zeigt die Anwendung.



Abb. 6–10 WLAN-Konfiguration mit WiFi Config

Als Adapter sollte `wlan0` ausgewählt sein. Klicken Sie dann auf die Schaltfläche `Scan`, um nach verfügbaren WLAN-Netzen zu suchen. Wenn die Liste das passende Netz enthält, können Sie es per Doppelklick konfigurieren. Den Konfigurationsdialog sehen Sie in Abbildung 6–11.



Abb. 6-11 Konfiguration der WLAN-Parameter

In der Regel können Sie alle Parameter so belassen, wie sie sind. Lediglich das Passwort müssen Sie im Feld PSK eingeben und dann auf *Save* klicken. Im Hauptmenü klicken Sie dann noch auf *Connect*, um sich mit dem WLAN-Netz zu verbinden.

Solange Ihr WLAN-Adapter vom Betriebssystem und von WiFi Config erkannt und unterstützt wird, ist dies der bequemste Weg, um WLAN auf dem Pi zu konfigurieren. Falls Sie aber ohne Desktop arbeiten oder der WLAN-Adapter nicht erkannt wurde, können Sie die Konfiguration auch per Hand durchführen.

Stecken Sie dazu den WLAN-Adapter in einen der USB-Ports des Pi und starten ihn. Rufen Sie nach der Anmeldung in einem Terminal das Kommando `lsusb` auf, um zu sehen, ob der Pi ihn richtig erkennt:

```
pi@raspberrypi ~ $ lsusb
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 002: ID 0424:9512 Standard Microsystems Corp.
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp.
Bus 001 Device 004: ID 050d:0237 Belkin Components F5U237
                                         USB 2.0 7-Port Hub
Bus 001 Device 005: ID 04e8:2018 Samsung Electronics Co.,
                                         Ltd WIS09ABGN LinkStick
                                         Wireless LAN Adapter
Bus 001 Device 006: ID 046d:c312 Logitech, Inc. DeLuxe 250 Keyboard
Bus 001 Device 007: ID 046d:c05a Logitech, Inc. Optical Mouse M90
```

In diesem Fall ist Gerät 005 ein WLAN-Adapter von Samsung. Sie können mit dem Befehl `dmesg` einen genaueren Blick auf die Startmeldung des Pi werfen und prüfen, ob der WLAN-Adapter richtig initialisiert wurde:

```
pi@raspberrypi ~ $ dmesg | less
...
usb 1-1.3.6: new high speed USB device number 5 using dwc_otg
usb 1-1.3.6: New USB device found, idVendor=04e8, idProduct=2018
usb 1-1.3.6: New USB device strings: Mfr=1, Product=2, SerialNumber=3
usb 1-1.3.6: Product: 802.11 n WLAN
usb 1-1.3.6: Manufacturer: Ralink
usb 1-1.3.6: SerialNumber: 1.0
...
...
```

Drücken Sie die Leertaste, um eine Seite nach unten zu gelangen, und »b«, um eine Seite nach oben zu gehen. Drücken Sie »q«, um zurück zum Shell-Prompt zu kommen. Wie Sie im aktuellen Fall erkennen können, verwendet der Samsung-Adapter einen WLAN-Chipsatz der Firma (manufacturer) Ralink. Dieser Chipsatz ist recht beliebt, weshalb ihn Debian sofort erkannt hat. Wenn die Ausgabe `dmesg` direkt nach der Initialisierung Ihres WLAN-Adapters Fehlermeldungen enthält, schauen Sie im Pi-Wiki¹³ nach. Oftmals müssen Sie die Firmware für Ihren WLAN-Adapter manuell herunterladen und den Linux-Kernel neu konfigurieren.

Treten keine Fehler auf, hat Debian Linux Ihren WLAN-Adapter korrekt erkannt. Sie können mit folgendem Befehl den aktuellen Status der drahtlosen Netzwerkschnittstellen Ihres Pi abfragen:

13) http://elinux.org/RPi_VerifiedPeripherals#USB_WiFi_Adapters

```
pi@raspberrypi ~ $ iwconfig
lo      no wireless extensions.

eth0    no wireless extensions.

wlan0   IEEE 802.11abgn ESSID:off/any
        Mode:Managed Access Point: Not-Associated Tx-Power=20 dBm
        Retry long limit:7 RTS thr:off Fragment thr:off
        Power Management:on
```

Momentan ist der Pi nicht mit einem drahtlosen Netzwerk verbunden, aber die Schnittstelle wlan0 ist hochgefahren und läuft. Der folgende Befehl sucht nach drahtlosen Netzwerken:

```
pi@raspberrypi ~ $ sudo iwlist scan | grep ESSID
ESSID:"darknet"
ESSID:"valhalla"
```

In diesem Fall sind zwei drahtlose Netzwerke namens *darknet* und *valhalla* in Reichweite. Um sich mit einem von ihnen zu verbinden, müssen Sie die Konfigurationsdatei /etc/network/interfaces mit einem Texteditor wie z.B. *nano* bearbeiten. Für eine Verbindung mit *darknet* fügen Sie die folgenden Zeilen in die Datei ein:

```
auto wlan0
iface wlan0 inet dhcp
wpa-ssid darknet
wpa-psk t0p$ecret
```

Diese Zeilen aktivieren die Schnittstelle wlan0 automatisch, wenn Sie den Pi das nächste Mal starten. Sie veranlassen den Pi auch, sich per DHCP eine IP-Adresse geben zu lassen. Der Pi wird versuchen, dem Netzwerk namens *darknet* beizutreten und das Passwort »t0p\$ecret« zu verwenden. Natürlich müssen Sie den Namen des Netzwerks und das Passwort entsprechend anpassen.

Ungeduldige müssen den Pi nicht neu starten. Führen Sie folgenden Befehl aus, damit der Pi sich mit dem drahtlosen Netzwerk verbindet:

```
pi@raspberrypi ~ $ sudo ifup wlan0
Internet Systems Consortium DHCP Client 4.2.2
Copyright 2004-2011 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
```

```
Listening on LPF/wlan0/00:12:fb:28:a9:51
Sending on LPF/wlan0/00:12:fb:28:a9:51
Sending on Socket/fallback
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 8
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 14
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 14
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCOFFER from 192.168.1.1
DHCPACK from 192.168.1.1
bound to 192.168.1.101 -- renewal in 2983 seconds.
```

Der Pi hat jetzt die IP-Adresse 192.168.1.101 und ist drahtlos mit Ihrem Netzwerk verbunden (wobei Ihre IP-Adresse vermutlich abweicht). Mit dem Befehl ping können wir prüfen, ob wir auf eine Website wie z.B. Google zugreifen können:

```
pi@raspberrypi ~ $ ping -c 3 google.com
PING google.com (173.194.69.100) 56(84) bytes of data.
64 bytes from google.com (173.194.69.100): icmp_req=1 ttl=45 time=26.7 ms
64 bytes from google.com (173.194.69.100): icmp_req=2 ttl=45 time=32.3 ms
64 bytes from google.com (173.194.69.100): icmp_req=3 ttl=45 time=34.8 ms
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 26.752/31.338/34.863/3.395 ms
```

Wie Sie sehen, ist der Pi jetzt drahtlos mit dem Internet verbunden. Führen Sie folgenden Befehl aus, um Informationen wie die Signalstärke usw. zu erhalten:

```
pi@raspberrypi ~ $ iwconfig
lo      no wireless extensions.

eth0    no wireless extensions.

wlan0   IEEE 802.11abgn ESSID:"darknet"
        Mode:Managed Frequency:2.442 GHz Access Point:
                  54:E6:FC:CF:77:8A
        Bit Rate=135 Mb/s Tx-Power=20 dBm
        Retry long limit:7 RTS thr:off Fragment thr:off
        Power Management:on
        Link Quality=40/70 Signal level=-70 dBm
        Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
        Tx excessive retries:1 Invalid misc:6 Missed beacon:0
```

Vergessen Sie nicht, dass ein Computer wie der Pi mehr als eine IP-Adresse haben kann. Wenn Sie ihn mittels Ethernet und WLAN anschließen, kann Ihre Startmeldung wie folgt aussehen:

```
My IP address is 192.168.2.109 192.168.1.101
```

Das bedeutet, dass Ihr Pi mit zwei Netzwerkschnittstellen ausgerüstet ist und er für jede eine eigene IP-Adresse verwendet.

6.6 Die nächsten Schritte

In diesem Kapitel haben Sie gelernt, wie der Pi in ein Netzwerk eingebunden wird. Sie können jetzt bequem mit SSH auf den Pi zugreifen und ihn sogar als Webserver einsetzen. Im nächsten Kapitel beschäftigen wir uns mit etwas ganz Neuem: Sie verwandeln den Pi in ein Multimediacenter.

7 Den Pi in ein Multimediacenter verwandeln

Die kleine Baugröße des Pi, sein geringer Stromverbrauch und seine Grafikfähigkeiten machen ihn zum perfekten Kandidaten für ein Multimediacenter wie eine PlayStation oder ein AppleTV. Um den Pi in ein solches Multimediacenter zu verwandeln, benötigen Sie eine spezielle Software namens XBMC¹.

XBMC ist ein getunter Media-Player, der beinahe jeden PC in ein Unterhaltungsgerät für digitale Medien verwandelt. Dies gilt auch für den Pi. In diesem Kapitel lernen Sie, wie Sie XBMC auf dem Pi laufen lassen.

7.1 Raspbmc installieren

XBMC ist ein umfangreiches Softwareprojekt und die Installation und Konfiguration kann etwas trickreich werden. Glücklicherweise müssen Sie diese Arbeit nicht selbst für den Pi erledigen, sondern profitieren von den glorreichen Anstrengungen des Raspbmc-Teams.² Raspbmc ist eine Linux-Distribution für den Pi, auf der nur XBMC läuft. Sie können wie gewohnt ein Image dieser Distribution auf einer SD-Karte erstellen und den Pi mit dieser SD-Karte starten. Statt eines Terminals oder Desktops startet die Raspbmc-Distribution dann XBMC automatisch.

Im Gegensatz zu anderen Linux-Distributionen für den Pi bietet das Raspbmc-Team nicht nur vollständige Image-Dateien für die SD-Karte zum Download an. Zusätzlich gibt es für alle wichtigen Plattformen einen komfortablen Installer. Dieser Installer lädt die aktuelle Version von Raspbmc aus dem Web herunter und kopiert sie automatisch auf Ihre SD-Karte.

1) <http://xbmc.org/>

2) <http://www.raspbmc.com/>

Wenn Sie die Raspbmc-Karte von einem Windows-PC erstellen möchten, laden Sie den Installer³ herunter, extrahieren Sie ihn auf Ihre Festplatte und starten Sie das Programm namens `installer.exe`. Sie sehen dann ein Fenster wie das folgende:

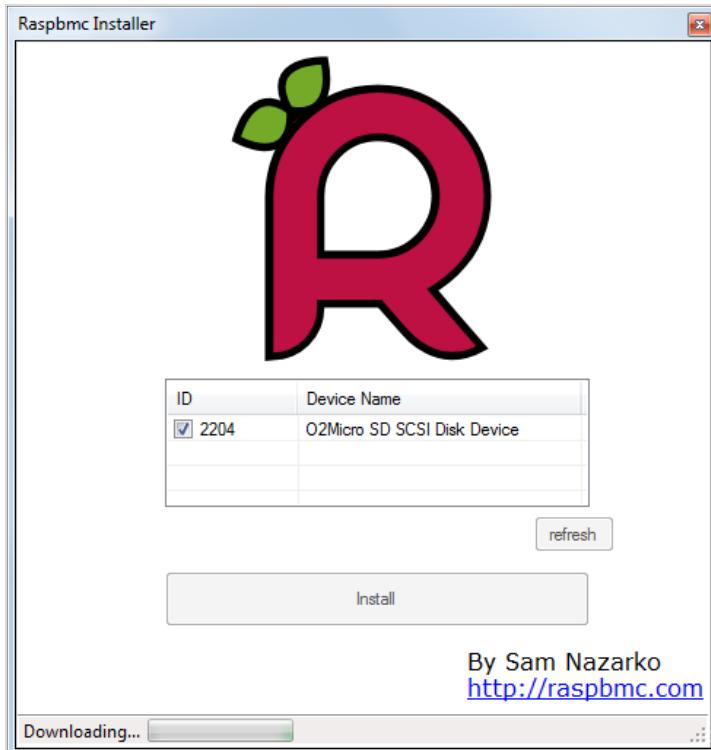


Abb. 7-1 Raspbmc-Installer für Windows

Legen Sie eine SD-Karte ein, wählen Sie Ihren Kartenleser aus und klicken Sie auf die Schaltfläche *Install*. Der Installer lädt dann die aktuelle Version von Raspbmc herunter und kopiert sie auf die SD-Karte. Beachten Sie, dass der Installer alle auf der SD-Karte enthaltenen Daten löscht!

Der Raspbmc-Installer für Linux und Mac OS besitzt keine schöne Benutzeroberfläche, ist aber trotzdem einfach zu verwenden. Es handelt sich um ein Python-Programm, und nachdem Sie es heruntergeladen

3) <http://www.raspbmc.com/wiki/user/windows-installation/>

haben⁴, können Sie es wie im folgenden Beispiel in einem Terminalfenster ausführen:

```
maik> sudo python install.py
< Raspbmc installer for Linux and Mac OS X
http://raspbmc.com
-----
Please ensure you've inserted your SD card, and press Enter to
continue.

Enter the 'IDENTIFIER' of the device you would like imaged, from the
following list:
#:
      TYPE NAME      SIZE      IDENTIFIER
1:          EFI      209.7 MB  disk0s1
2:  Apple_HFS Macintosh HD  499.2 GB  disk0s2
3:  Apple_Boot Recovery HD  650.0 MB  disk0s3
#:
      TYPE NAME      SIZE      IDENTIFIER
1:  Windows_FAT_32      67.1 MB  disk1s1
2:    Linux_Swap      129.0 MB  disk1s2
3:      Linux        16.0 GB  disk1s3

→ Enter your choice here (e.g. 'disk1s1'): disk1s1
< It is your own responsibility to ensure there is no data loss!
Please backup your system before imaging
→ Are you sure you want to install Raspbmc to '/dev/disk1s1'? [y/N] y
< Downloading, please be patient...
Downloaded 49.52 of 49.52 MiB (100.00%)

Unmounting the drive in preparation for writing...
Unmount of all volumes on disk1 was successful
Please wait while Raspbmc is installed to your SD card...
(This may take some time and no progress will be reported until it has
finished.)

0+3022 records in
0+3022 records out
198000640 bytes transferred in 27.034211 secs (7324077 bytes/sec)
Installation complete.
Finalising SD card, please wait...
Disk /dev/rdisk1 ejected
Raspbmc is now ready to finish setup on your Pi, please insert the
SD card with an active internet connection
```

Das Installationsprogramm zeigt eine Liste aller an den PC angeschlossenen Laufwerke an, einschließlich des Kartenlesers. Auf Ihrem PC wird die Ausgabe etwas anders aussehen, aber in unserem Beispiel besteht die SD-

4) <http://svn.stmlabs.com/svn/raspbmc/testing/installers/python/install.py>

Karte aus drei Partitionen namens disk1s1, disk1s2 und disk1s3. Geben Sie den Namen der ersten Partition an – in diesem Fall disk1s1 – und bestätigen Sie, dass Sie Raspbmc installieren wollen. Verwenden Sie unbedingt das richtige Laufwerk, da der Installer alle Daten auf dem ausgewählten Gerät löscht.

Wenn Sie die startfähige SD-Karte erzeugt haben, stecken Sie sie in den Pi und schalten ihn ein. Der Pi startet XBMC nicht direkt, sondern lädt ein minimalistisches Linux-System und beginnt die eigentliche Installation von Raspbmc. Zuerst wird die SD-Karte neu partitioniert und dann formatiert. Danach wird das Root-Dateisystem heruntergeladen und es werden der Kernel sowie einige Kernelmodule und Bibliotheken installiert. Nach einem Neustart wird schließlich die aktuelle Version von XBMC heruntergeladen und installiert.

Alle diese Schritte laufen ohne Benutzereingriffe ab. Abhängig von der Geschwindigkeit Ihrer SD-Karte und Ihrer Internetverbindung kann das bis zu 20 Minuten dauern. Sie können also ruhig einen Spaziergang machen oder einen Kaffee oder Tee trinken.

7.2 Raspbmc das erste Mal starten

Nachdem die Installation abgeschlossen ist, startet Raspbmc automatisch und führt XBMC aus. Abbildung 7–2 zeigt das Hauptmenü:



Abb. 7–2 Hauptmenü von XBMC

Auf den ersten Blick sieht XBMC aus wie viele andere Media-Player. Es gibt Menüeinträge zum Anzeigen von Fotos, Abspielen von Videos oder Musik und zur Konfiguration einiger Systemeinstellungen. Diese Funktionen erklären sich meist von selbst. Um Inhalte anzusehen oder abzuspielen, wählen Sie einfach Mediendateien von der SD-Karte oder einem USB-Gerät und XBMC zeigt sie an.

Um ein USB-Gerät wie eine Festplatte oder einen USB-Stick an den Pi anzuschließen, müssen Sie einen USB-Hub verwenden oder vorübergehend Ihre Maus abziehen und XBMC mit der Tastatur steuern. Statt einen Menüeintrag mit der Maus anzuklicken, können Sie den Fokus mit den Cursortasten bewegen und den Eintrag mit Return auswählen. Mit der Esc-Taste gelangen Sie in der Menühierarchie um eine Ebene nach oben.

XBM C ist jedoch mehr als ein einfacher Media-Player. Sie können ihn durch viele Add-ons verbessern und erweitern, die gratis im Web erhältlich sind. Einfach gesagt geben Ihnen Add-ons Zugriff auf Medien im Web. Zum Beispiel finden Sie Add-ons, die den Inhalt bestimmter Fernsehsender sammeln, oder Add-ons, mit denen Sie Zugriff auf die Musik toller Videospiele bekommen. XBMC bietet sogar einen bequemen Weg zur Verwaltung der Add-ons. In der folgenden Abbildung sehen Sie das Add-on TED, das die neuesten und besten TED-Konferenzvideos anzeigt.



Abb. 7-3 Das Add-on TED

Nehmen Sie sich ein paar Minuten Zeit und schauen Sie sich die Add-ons an, um zu sehen, ob etwas Interessantes für Sie dabei ist. Im Zweifel installieren Sie sie und werfen einen Blick darauf. Nicht gewünschte Add-ons lassen sich leicht wieder entfernen. Beachten Sie, dass Sie für viele Add-Ons genügend Bandbreite benötigen, da sie eine Menge Daten übertragen.

Abhängig von der Geschwindigkeit Ihrer SD-Karte und Ihrer Internetverbindung werden Sie Verzögerungen bemerken, wenn Sie in XBMC Menüeinträge wählen. Das wird sich in zukünftigen Versionen verbessern, momentan müssen Sie jedoch damit leben und geduldig sein, wenn Sie sich durch die XBMC-Menüs bewegen. Die Wiedergabe von Inhalten funktioniert dagegen prima, ohne Verzögerungen oder Ruckeln.

Schließlich sollten Sie einen Blick auf das Menü *Systems > Settings* werfen und prüfen, ob alle Einstellungen mit ihrer lokalen Konfiguration übereinstimmen. Wenn Sie z.B. Composite Video verwenden, müssen Sie *analog* als Audiogerät in *Systems > Settings > Audio* wählen.

7.3 Hinzufügen von Dateien zu XBMC

In XBMC können Sie einfach neue Filme, Fernsehsendungen oder Musik hinzufügen, indem Sie in den Menüs *Videos* oder *Music* den Eintrag *Add Files* verwenden. Bevor Sie beginnen, Mediendateien hinzuzufügen, sollten Sie wissen, wie XBMC intern funktioniert. XBMC ist mehr als ein einfacher Media-Player. Es ist eine vollständige Medienbibliothek, die versucht, automatisch so viele Informationen über Ihre Mediendateien zu bekommen wie möglich. Zum Beispiel ruft XBMC Zusatzinformationen über Ihre Lieblingssendungen aus Webdatenbanken ab und fügt sie Ihrer Bibliothek hinzu. Dabei kann es sich um das Datum der Erstausstrahlung handeln oder um eine kurze Zusammenfassung. Um das zu ermöglichen, verwendet XBMC ein bestimmtes Benennungsschema. Sie finden alle Informationen darüber im Projekt-Wiki⁵. Beachten Sie, dass die Verwendung der richtigen Dateinamen und Verzeichnisstrukturen auch das Hauptmenü von XBMC beeinflusst. Wenn Sie z.B. das Verzeichnis *TV-Shows* anlegen, fügt XBMC in das Hauptmenü einen Eintrag für Fern-

5) http://wiki.xbmc.org/index.php?title=Adding_videos_to_the_library/Naming_files

sehsendungen ein. Um das Maximum aus XBMC herauszuholen, sollten Sie Ihre Mediendateien entsprechend benennen, bevor Sie sie importieren.

Video- und Musikformate

XBMC unterstützt beinahe alle Containerformate und Codecs, die auf dem Markt sind. Es gibt kaum eine Multimediatei, die Sie nicht auf dem Pi abspielen können. Trotzdem kann es problematisch werden: Die Raspberry-Foundation hat Hardwarebeschleunigung nämlich nur für den H.264-Video-Codec lizenziert. Glücklicherweise ist das einer der beliebtesten verfügbaren Codecs. Liegen Videos jedoch in einem anderen Codec vor, können Probleme auftreten. Mittlerweile können Sie aber auf der Raspberry-Webseite Lizenzen für weitere Video-Codecs erwerben.^a Geben Sie beim Kauf die Seriennummer des Pi ein und sie erhalten einen Lizenzschlüssel. Diesen müssen Sie dann im Einstellungsmenü von Raspbmc eingeben.

- a. <http://www.raspbmc.com/license-keys/>

Am einfachsten fügen Sie XBMC Dateien hinzu, indem Sie ein USB-Gerät mit den entsprechenden Daten direkt an den Pi anschließen. Das funktioniert gut, hat jedoch einige Nachteile. Das USB-Gerät ist häufig größer als der Pi selbst und benötigt vermutlich auch mehr Strom. Außerdem belegt es mindestens einen der wertvollen USB-Anschlüsse. Eine bessere Lösung ist, Medien auf einer SD-Karte zu speichern oder sie über das lokale Netzwerk zu übertragen. Wegen der hervorragenden NetzwerkinTEGRATION von XBMC können Sie beide Lösungen einfach umsetzen.

XBMC unterstützt FTP, SFTP, SSH, NFS und Samba von Haus aus und Sie müssen nur wenig konfigurieren, damit sie laufen. XBMC hat SSH per Voreinstellung aktiviert. Um also Daten auf die SD-Karte zu kopieren, können Sie auch scp verwenden, wie in Abschnitt 6.2 beschrieben. Führen Sie folgenden Befehl im Terminal auf Ihrem PC aus und erstellen Sie einen Ordner namens Movies im Stammverzeichnis von XBMC:

```
maik> ssh pi@192.168.2.109 "mkdir /home/pi/Movies"
```

Ersetzen Sie die IP-Adresse durch die Adresse Ihres Pi. Beachten Sie, dass Raspbmc mit einem Benutzer namens *pi* geliefert wird, der momentan das Passwort *raspberry* hat. Jetzt können Sie Mediendateien mit `scp` kopieren und sie später der XBMC-Bibliothek hinzufügen.

```
maik> scp Pulp\ Fiction\ (1994).avi pi@192.168.2.109:Movies
```

Auch wenn Sie eine sehr große SD-Karte verwenden, reicht sie vermutlich nicht aus, um Ihre gesamte Medienbibliothek zu speichern. Außerdem ist es nicht sinnvoll, Filme oder Musik immer erst kopieren zu müssen, bevor Sie sie sehen oder hören können. Dafür wurden Netzwerkdateisysteme wie NFS und Samba entwickelt, die XBMC alle unterstützt.

Mit NFS oder Samba können Sie alle Mediendateien auf Ihrem regulären PC speichern und sie zum Pi übertragen, wenn Sie sie verwenden möchten. Das Konfigurieren von NFS⁶ oder Samba⁷ geht über den Rahmen dieses Buchs hinaus. Im XBMC-Wiki findet sich jedoch eine hervorragende Dokumentation für alle wichtigen Plattformen.

Sobald Ihre Mediendateien in Ihrem Heimnetzwerk über NFS oder Samba zur Verfügung stehen, können Sie mit XBMC leicht auf sie zugreifen. Bei NFS müssen Sie nichts konfigurieren. Die Samba-Einstellungen finden Sie unter *System > Service > SMB Client*.

7.4 Fernbedienung von XBMC

Wenn Sie den Pi als Multimediacenter in Ihrem Wohnzimmer einsetzen wollen, kommt früher oder später der Wunsch nach einer Fernbedienung auf. Dazu können Sie Spezialhardware mit einem Infrarotsender⁸ verwenden, leichter geht es aber mit Ihrem Smartphone.

XBMC bietet nicht nur Add-ons zur Verwaltung von Mediendateien, sondern ermöglicht Ihnen auch, nette Webschnittstellen zu installieren, mit denen Sie XBMC über eine Fernbedienung steuern können. Gehen Sie ins Menü *System > Settings > Services > Webserver* und aktivieren Sie die Option »Allow Control of XBMC via HTTP«. Klicken Sie dann auf die Schaltfläche »Web interface« und wählen Sie »Get More...« In Abbil-

6) <http://wiki.xbmc.org/index.php?title=NFS>

7) <http://wiki.xbmc.org/index.php?title=Samba>

8) <http://www.raspbmc.com/wiki/user/configuring-remotes/>

dung 7–4 sehen Sie die momentan verfügbaren Schnittstellen. Aktivieren Sie einfach alle, sodass Sie sie ausprobieren und Ihren Favoriten herausfinden können.

Wenn Sie die Webschnittstelle aktiviert haben, können Sie sie mit jedem Browser verwenden, der Zugang zu Ihrem Netzwerk hat. Die Schnittstelle läuft in der Regel auf Port 8080, sodass Sie in Ihren Browser eine URL wie 192.168.2.109:8080 eingeben können, um auf XBMC zuzugreifen (verwenden Sie auch hier bitte die IP-Adresse Ihres Pi).

In Abbildung 7–5 sehen Sie z.B. die AWXi-Schnittstelle. Sie bietet alle üblichen Bedienknöpfe wie Wiedergabe, Pause und Stopp und ermöglicht es Ihnen, die gesamte Medienbibliothek zu durchsuchen.



Abb. 7–4 XBMC bietet mehrere Schnittstellen.



Abb. 7–5 Die AWXi-Webschnittstelle in Aktion

Wenn Sie ein iPad oder iPhone haben⁹ oder ein Android-Smartphone¹⁰, können Sie sogar eine Fernbedienungs-App für XBMC installieren. Abbildung 7–6 und Abbildung 7–7 zeigen die Android-Version. Sie sieht nicht nur prima aus, sondern bietet auch einen bequemen Zugriff auf alle XBMC-Funktionen. In vielen Belangen ist sie besser als die normale Fernbedienung für Fernseher.

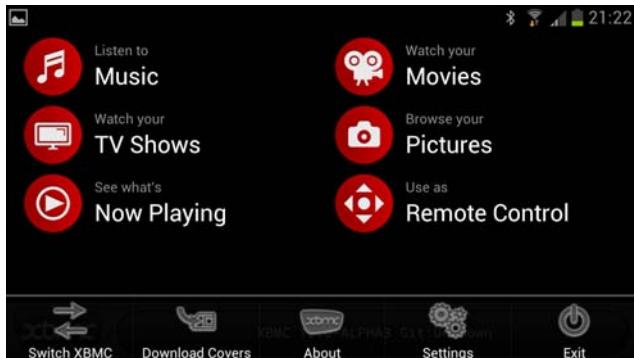


Abb. 7–6 XBMC von Android aus steuern



Abb. 7–7 Ganz wie eine klassische Fernbedienung

9) <http://itunes.apple.com/gb/app/official-xbmc-remote/id520480364>

10) <https://play.google.com/store/apps/details?id=org.xbmc.android.remote>

7.5 Die nächsten Schritte

In diesem Kapitel haben Sie gelernt, etwas ganz Neues mit dem Pi zu machen. Sie haben ihn erstmalig nicht als regulären PC eingesetzt, sondern ihn in ein Spezialgerät umgewandelt: ein Multimediacenter. Im nächsten Kapitel lernen Sie, wie Sie noch mehr Multimedia-Anwendungen auf Ihrem Pi laufen lassen können, und spielen einige spannende Spiele.

8 Spiele auf dem Pi spielen

Linux war noch nie eine populäre Spieleplattform. Auch wenn sich die Lage mit den Jahren gebessert hat, dauert es vermutlich noch etwas, bis die neuesten Renner auf Linux erscheinen. Trotzdem können Sie schon heute unterhaltsame und manchmal auch süchtig machende Spiele auf dem Raspberry Pi spielen.

Zum Beispiel können Sie Tausende von Textadventures auf dem Pi spielen. Die kommerziellen Unternehmen haben zwar dieses Genre längst verlassen, dennoch hat es eine enthusiastische Fangemeinde, die regelmäßig neue Spiele herausbringt. Und wenn Sie die Klassiker wie *Zork* noch nicht gespielt haben, sollten Sie sie ausprobieren.

Ein weiteres Klassikergenre ist das Point-and-Click-Adventure, zu dem Spiele wie *The Secret of Monkey Island* und *Day of the Tentacle* gehören. Dank der Arbeit der Open-Source-Community können Sie die meisten davon auf Ihrem Pi spielen.

Auch wenn der Pi für moderne Spiele nicht leistungsfähig genug ist, hat er doch genug Rechenleistung für native Linux-Spiele wie *Quake III*. Er kann sogar einige Heimcomputer und Spielkonsolen der Vergangenheit emulieren. Zum Beispiel können Sie alle Spiele für das Atari VCS 2600 auf dem Pi spielen.

8.1 Textadventures

In der Ära der ersten Heimcomputer waren Textadventures sehr beliebt. Im Gegensatz zu modernen Spielen mit 3D-Grafik und Surround-Sound wirken Textadventures sehr spartanisch. Sie geben nur Texte aus und Sie steuern das Spiel, indem Sie Befehle über die Tastatur eingeben. Nachfol-

gend sehen Sie den Text des Eröffnungsbildschirms von *Zork*, einem der berühmtesten Textadventures.

```
< ZORK I: The Great Underground Empire
Copyright (c) 1981, 1982, 1983 Infocom, Inc. All rights reserved.
ZORK is a registered trademark of Infocom, Inc.
Revision 88 / Serial number 840726
West of House
You are standing in an open field west of a white house, with a boarded
front door.
There is a small mailbox here.

→ >open mailbox
<Opening the small mailbox reveals a leaflet.

→ >take leaflet
<Taken.

→ >read leaflet
<"WELCOME TO ZORK!

ZORK is a game of adventure, danger, and low cunning.
In it you will explore some of the most amazing territory ever seen by
mortals. No computer should be without one!"
```

Lassen Sie sich nicht durch die Optik der Spiele abschrecken. Viele von ihnen bieten spannende Geschichten und unterhalten Sie viele Stunden lang.

Auch wenn seit Jahrzehnten keine kommerziellen Textadventures mehr erschienen sind, hat das Genre noch immer eine aktive Community, die spannende Spiele herausbringt. Die meisten dieser Spiele basieren auf langen und gut erzählten Geschichten, weshalb manche ihrer Autoren sie lieber als *interaktive Fiktion* bezeichnen.

Eines der ersten Unternehmen dieses Genres war Infocom. Mehrere Jahre lang brachte es die besten Textadventures aller Zeiten heraus. Die Infocom-Entwickler erkannten frühzeitig, dass sie sich viel Arbeit ersparen konnten, indem sie eine umgebungsunabhängige Sprache für die Beschreibung der Spiele entwickelten. Sie nannten die Sprache *Z-Language* und die Autoren von Textadventures verwenden sie noch heute, um solche Spiele zu schreiben.

Um in Z-Language geschriebene Programme auszuführen, benötigen Sie eine virtuelle Maschine namens Z-Maschine¹. Eine der besten ist Frotz². Sie installieren sie wie folgt:

```
pi@raspberrypi ~ $ sudo apt-get install frotz
```

Um über Frotz ein Textadventure zu spielen, benötigen Sie nur die Z-Language-Datei des Spiels. Ihre Suche nach Textadventures beginnen Sie am besten im Interactive Fiction Archive³, denn dort gibt es Tausende von Spielen.

Wenn Textadventures etwas Neues für Sie sind, sollten Sie Ihre Reise mit der *Zork-Trilogie* beginnen. Diese Spielreihe machte Infocom berühmt, und obwohl sie schon ein paar Jahrzehnte alt ist, sind die Spiele so frisch wie am ersten Tag. Inzwischen sind sie kostenlos zu bekommen.⁴ Laden Sie sich also *Zork I*⁵ herunter und starten Sie es wie folgt:

```
pi@raspberrypi ~ $ unzip zork1.zip  
pi@raspberrypi ~ $ frotz zork1/DATA/ZORK1.DAT
```

Dadurch wird der Z-Maschinen-Interpreter gestartet, der das in *ZORK1.DAT* gespeicherte Spiel ausführt. In wenigen Minuten werden Sie mit dieser Art von Spiel vertraut sein, unter <http://pr-if.org/doc/play-if-card/play-if-card.html> finden Sie auch eine kleine Hilfe.

Wenn Ihnen Textadventures gefallen, haben Sie vielleicht Interesse daran, selbst eines mit aktuellen Werkzeugen zu schreiben.⁶ Es ist wirklich einfach, zumindest auf technischer Ebene. Eine gute und originelle Story brauchen Sie natürlich trotzdem.

8.2 Point-and-Click-Adventures

Ein anderes Genre, das immer populär gewesen ist, sind Point-and-Click-Adventures. In diesen Spielen steuern Sie den Protagonisten mit der Maus. Sie können ihn zu einer bestimmten Stelle laufen lassen, indem Sie

1) <http://en.wikipedia.org/wiki/Z-machine>

2) <http://frotz.sourceforge.net/>

3) <http://www.ifarchive.org/>

4) <http://www.infocom-if.org/downloads/downloads.html>

5) <http://www.infocom-if.org/downloads/zork1.zip>

6) <http://inform7.com/>

darauf klicken, und Aktionen durchführen, indem Sie auf dem Bildschirm auf den Namen der Aktion klicken. Beliebte Point-and-Click-Adventures sind *The Secret of Monkey Island*, *Day of the Tentacle* und *Maniac Mansion*.

Es hat immer eine Nachfrage nach neuen Adventurespielen dieser Art gegeben, in den letzten Jahren sind aber nur noch wenige erschienen. Die großen Unternehmen glauben nicht mehr an den Erfolg solcher Spiele und vertreiben lieber Ego-Shooter wie *Call of Duty* oder *Battlefield*.

Tim Schafer, einer der Schöpfer von *The Secret of Monkey Island*, war von dieser Situation frustriert und versuchte auf Kickstarter.com Geld zu sammeln, um ein neues Point-and-Click-Adventure zu erstellen. Er bekam mehr als 3,3 Millionen US-Dollar zusammen und zeigte damit, dass es noch immer eine große Fangemeinde für diese Spiele gibt.

Zur Drucklegung dieses Buchs war Tims Spiel noch nicht erschienen, glücklicherweise können Sie aber viele andere Klassiker spielen. Wie Textadventures laufen auch die meisten Point-and-Click-Adventures auf einer virtuellen Maschine. Die beliebteste ist SCUMM, das für *Script Creation Utility for Maniac Mansion* steht.

SCUMM stammt ursprünglich von den Entwicklern von LucasArts und diente zur Umsetzung des Spiels *Maniac Mansion*; mit ihr wurden seitdem noch viele andere Spiele realisiert.

Das Projekt ScummVM⁷ implementiert eine virtuelle Maschine, die SCUMM-Spiele interpretiert, und ist kostenfrei erhältlich. Sie installieren sie wie folgt:

```
pi@raspberrypi ~ $ sudo apt-get install scummvm  
pi@raspberrypi ~ $ sudo apt-get install beneath-a-steel-sky  
pi@raspberrypi ~ $ sudo apt-get install flight-of-the-amazon-queen
```

Dadurch wird nicht nur ScummVM installiert, sondern auch zwei tolle Spiele. Diese Spiele erscheinen im Abschnitt *Games* im Startmenü Ihres Desktops. Abbildung 8-1 zeigt ein Screenshot des Spiels *Beneath a Steel Sky*.

7) <http://www.scummvm.org/>

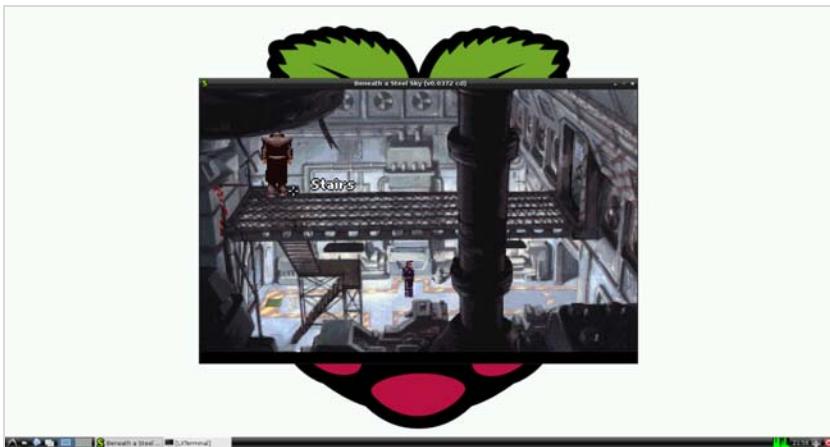


Abb. 8-1 »Beneath a Steel Sky« ist noch immer ein tolles Spiel.

Beneath a Steel Sky und *Flight of the Amazon Queen* sind Freeware, Sie können sie also unbedenklich installieren. Die meisten anderen Spiele sind nicht gratis erhältlich, Sie dürfen sie also nur installieren, wenn Sie ein Original besitzen. Wenn Sie andere Spiele, die mit ScummVM kompatibel sind, besitzen, können Sie ScummVM aufrufen und sie hinzufügen.

8.3 Emulieren anderer Plattformen

Eine andere Möglichkeit, interessante Spiele auf dem Pi zu spielen, ist das Emulieren anderer Plattformen. Für Linux sind viele Emulatoren erhältlich und sie erwecken klassische Computer und Spielkonsolen wie den Commodore 64, das Sega Mega-Drive, das Nintendo Entertainment System und viele andere wieder zum Leben. Für jedes alte System aus der Vergangenheit gibt es mindestens einen Emulator.

Ein Emulator simuliert die Hardware eines bestimmten Computers oder einer Spielkonsole mit Software. Wenn Sie den Emulator auf dem Pi ausführen, können Sie mit dem emulierten System arbeiten, als wenn Sie die Originalhardware verwenden würden. Am wichtigsten ist jedoch, dass Sie die alte Software und die Spiele laufen lassen können, die es für das alte System gab.

Selbst bei einfachen Geräten ist die Emulation extrem schwierig, und die meisten Emulatoren leiden unter zwei Hauptproblemen. Das erste ist die Genauigkeit. Oft ist ein Emulator nicht in der Lage, das Originalsystem zu 100 % nachzubilden. Das zweite Problem besteht in der Leistung. Auch die Emulation sehr alter und langsamer Hardware verschlingt große Mengen an Ressourcen. Der Commodore 64 z.B. hatte eine Taktrate von nur 1 MHz, Sie benötigen aber viel Rechenleistung, um ihn zu emulieren. Zurzeit ist selbst die Pi-Hardware nicht in der Lage, den C64 mit vernünftigen Frameraten zu emulieren, obwohl der Pi ein Vielfaches der Leistung des C64 besitzt. Dies könnte sich mit besseren Grafiktreibern für den Pi ändern.

Trotzdem ist der Pi stark genug, einige coole Spielkonsolen zu emulieren, und eine davon ist das Atari VCS 2600⁸. Dieses Gerät war zwischen 1977 und den frühen 1990er-Jahren sehr beliebt und bot großartige Spiele wie *Pac-Man*, *Centipede* und *Pitfall*. Die Konsole war so beliebt, dass mehrere Emulatoren dafür existieren, und einer der besten ist Stella⁹. Sie installieren und starten Stella wie folgt:

```
pi@raspberrypi ~ $ sudo apt-get install stella  
pi@raspberrypi ~ $ stella
```

Zuerst fragt Sie Stella nach der Position der Spiele-ROMs. Die Spiele für das VCS 2600 wurden auf Modulen geliefert, die ein paar Kilobyte Nur-Lese-Speicher (ROM) besaßen. Um mit Stella ein Spiel zu spielen, müssen Sie das ROM des Moduls kopieren. Sie können den Inhalt eines Moduls mit einem Spezialgerät auf Ihren PC übertragen. Glücklicherweise finden Sie all diese ROM-Dateien im Web¹⁰, aber es gibt damit ein großes Problem: Obwohl die meisten Spiele für das VCS 2600 sehr alt sind, unterliegen sie noch dem Urheberrecht. In den meisten Ländern ist es nicht legal, ROM-Dateien von Spielen herunterzuladen, die Sie nicht selbst besitzen.

Sie können die Spielemodule im Web für kleines Geld kaufen, und einige Anbieter verkaufen Atari-Spielesammlungen auf CDs. Diese Sammlungen machen nichts anderes, als die Original-ROM-Dateien über einen Emulator auszuführen.

8) http://en.wikipedia.org/wiki/Atari_2600

9) <http://stella.sourceforge.net/>

10) <http://atariage.com/>

Die Größe dieser ROMs liegt normalerweise zwischen 4 KB und 8 KB und die Dateinamen haben die Endung .bin. Die Pac-Man-ROM-Datei heißt also beispielsweise pacman.bin. Wenn Sie eine ROM-Datei auf den Pi kopiert haben, können Sie sie im Stella-Hauptmenü auswählen und das Spiel wird sofort gestartet. In der Voreinstellung verwenden Sie die Pfeiltasten für Bewegungen und die Leertaste für Aktionen. Stella ermöglicht das Umbelegen der Tasten und bietet auch Unterstützung für Joysticks. Zusätzlich können Sie zahlreiche Video- und Audiooptionen ändern. Beachten Sie jedoch, dass der Pi in den anspruchsvollsten Videomodi das VCS 2600 nicht mehr ordentlich emulieren kann.

Das Spielen einiger Spieleklassiker kann Kindheitserinnerungen zurückbringen, das VCS 2600 hat aber auch eine unglaublich aktive Fangemeinde, die immer noch Spiele entwickelt.¹¹ Viele dieser selbst gemachten Spiele sehen sogar besser aus und klingen besser als viele der Originalspiele und die meisten sind umsonst. In Abbildung 8–2 sehen Sie z.B. *A-VCS-tec Challenge*¹². Einige dieser selbst entwickelten Spiele sind noch heute als Modul erhältlich.

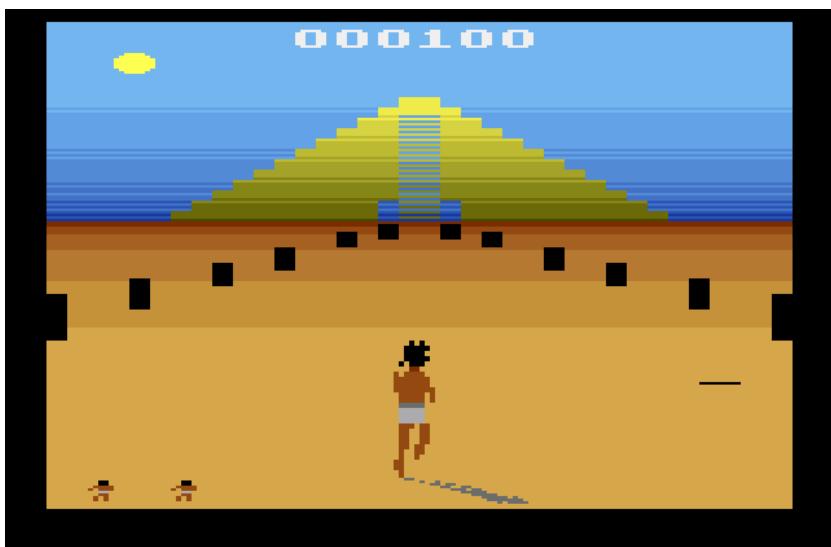


Abb. 8–2 Es entstehen noch immer tolle Spiele für das VCS 2600.

11) http://en.wikipedia.org/wiki/Atari_2600_homebrew

12) <http://www.quernhorst.de/atari/ac.html>

Spiele oder Demos für das VCS 2600 zu entwickeln ist sehr schwierig, Sie können aber eine Menge dabei lernen und viel Spaß damit haben! Die meisten Personen haben keine Vorstellung, wie wenig die Hardware bot. Sie hatte eine Taktfrequenz von 1,19 MHz, nur 128 Bytes an Speicher und keinen Bildpuffer für die Videoausgabe. Für dieses Gerät Software zu entwickeln war damals wirklich anstrengend. Mit den heutigen Werkzeugen und der Dokumentation ist es viel einfacher geworden. Zum Beispiel bietet Stella einen Debugger, der es Ihnen ermöglicht, den Status eines Spiels im Betrieb anzusehen und zu ändern. Um den Debugger zu aktivieren, starten Sie Stella wie folgt:

```
pi@raspberrypi ~ $ stella -debug
```

Für den Aufruf des Debuggers drücken Sie die Akzent-Taste. Vergessen Sie nicht, dass Sie die Tastenfunktionen in Stella frei umbelegen können, wenn Sie sie nicht auf Ihrer Tastatur finden. In Abbildung 8–3 sehen sie den Debugger in Aktion.

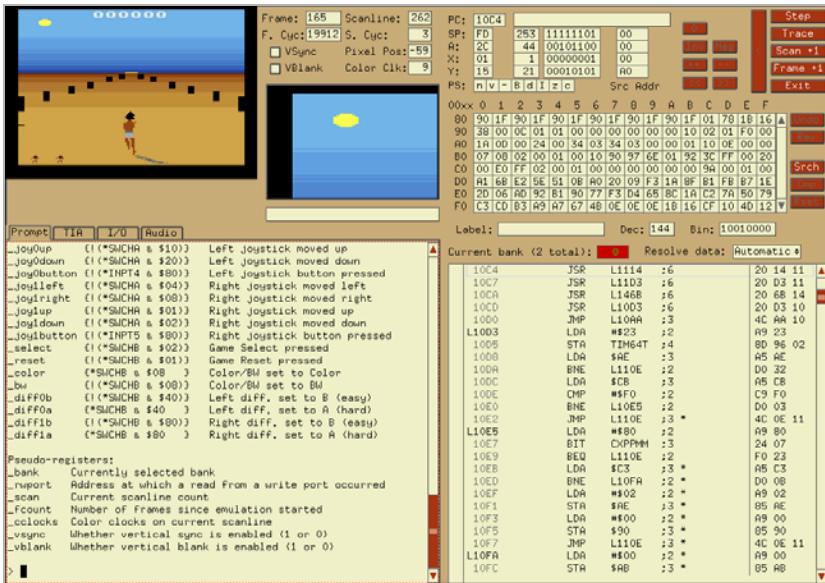


Abb. 8–3 Stella bietet einen leistungsfähigen Debugger.

Alles in allem läuft Stella ziemlich gut auf dem Pi, denn das VCS 2600 war kein sehr leistungsfähiges Gerät. Emulatoren für andere Geräte funktionieren zurzeit nicht so gut. Der C64-Emulator Vice¹³, läuft beispielsweise auf dem Pi, seine Framerate ist für die meisten Spiele aber zu langsam. Dies gilt auch für Spielemulatoren wie MAME¹⁴, aber die Situation könnte sich bald ändern. Bis dahin können Sie Ihr eigenes VCS 2600-Spiel entwickeln.

8.4 Native Spiele

In den vorangegangenen Abschnitten haben Sie Technologien kennengelernt, die Ihnen dabei helfen, Spiele mittels virtueller Maschinen und Emulatoren zu spielen. Es gibt aber auch eigens Spiele für Linux. Auf dem LXDE-Desktop finden Sie z.B. einige Spieleklassiker, die in Python geschrieben wurden: *Vier gewinnt*, *Snake* usw.

Da der Pi ein reguläres Linux-System ist, können Sie jedes Spiel, das mit Ihrer aktuellen Distribution kompatibel ist, starten, solange die Pi-Ressourcen ausreichend sind. Wenn Sie das Web nach klassischen Spielen wie *Tetris* oder *Pac-Man* durchsuchen, werden Sie schnell gute Klone finden.

```
pi@raspberrypi ~ $ sudo apt-get install ltris pacman
```

Es lohnt sich, alle gefundenen Spiele zu installieren, viele verlangen dem Pi jedoch zu viel ab. Die Pakete *pinball* und *extremetuxracer* laufen beispielsweise nicht auf dem Pi.

So überrascht es fast schon, dass der Pi *Quake II* und *Quake III* (zwei bekannte Ego-Shooter) mit brauchbaren Frameraten ausführen kann. Als dieses Buch geschrieben wurde, gab es zwar ein paar Probleme mit der Tonausgabe, die Ego-Shooter sind jedoch spielbar. Halten Sie also Ausschau nach neuen Pi-Spielen.

Neben all dem Genannten finden Sie auch coole Spiele im Pi Store (siehe Abschnitt 3.5). *Iridium Rising* zum Beispiel ist ein exklusives Spiel für den Pi und es ist sehr professionell.

13) <http://vice-emu.sourceforge.net/>

14) <http://mamedev.org/>

Schließlich hat Mojang¹⁵ eine besondere *Minecraft*-Edition für den Pi¹⁶ veröffentlicht. Die ersten Videos sehen sehr vielversprechend aus, es hat einige einzigartige Features und ist sogar kostenlos. Es lohnt sich also, auf die Suche nach neuen Pi-Spielen zu gehen.

8.5 Die nächsten Schritte

In diesem Kapitel haben Sie gelernt, wie Sie sich mit Spieleklassikern auf dem Pi die Zeit vertreiben. Der Pi ist keine Xbox oder PlayStation, aber er bietet unterhaltsame Spiele, die Sie auf den meisten modernen Spielkonsolen nicht finden werden. Das nächste Kapitel wendet sich einem ganz anderen Thema zu. Sie lernen, wie Sie Elektronikprojekte entwickeln und an die GPIO-Ports des Pi anschließen.

15) <http://www.mojang.com/>

16) <http://www.raspberrypi.org/archives/2603>

9 Mit den GPIO-Pins herumbasteln

Die Raspberry-Foundation entwickelte den Pi nicht nur, damit Kinder das Programmieren lernen, sondern auch, um Elektronikexperimente durchzuführen. Deshalb gibt es die Erweiterungsanschlüsse, über die Sie Ihre eigenen Elektronikprojekte einfach anschließen können.

In diesem Kapitel lernen Sie, wie Sie eigene kleine Elektronikgeräte bauen und sie mit dem Pi steuern. Sie beginnen mit einem einfachen Schaltkreis, der eine LED zum Leuchten bringt. Danach steuern Sie die LED mit den Pi-Erweiterungsanschlüssen, indem Sie sie per Pi-Befehl an- und ausschalten.

Anschließend bauen Sie einen kleinen Speicheralarm, ein Gerät, das anzeigt, wie viel Speicher im Pi noch frei ist. Es funktioniert wie eine Ampel. Ein rotes Licht bedeutet in diesem Fall, dass nur noch wenig freier Speicher vorhanden ist. Zusätzlich stellen Sie das Ergebnis des Alarms in Ihrem Webbrower dar.

9.1 Was Sie benötigen

Für die Projekte dieses Kapitels benötigen Sie nur wenige kostengünstige Bauteile (die Sie alle in Abb. 9–1 sehen):

- Ein Steckbrett in halber Baugröße
- Drei 5-mm-LEDs (rot, gelb und grün)
- Drei Widerstände im Bereich zwischen $220\ \Omega$ und $1\ k\Omega$
- Vier Jumperkabel mit Stecker und Buchse

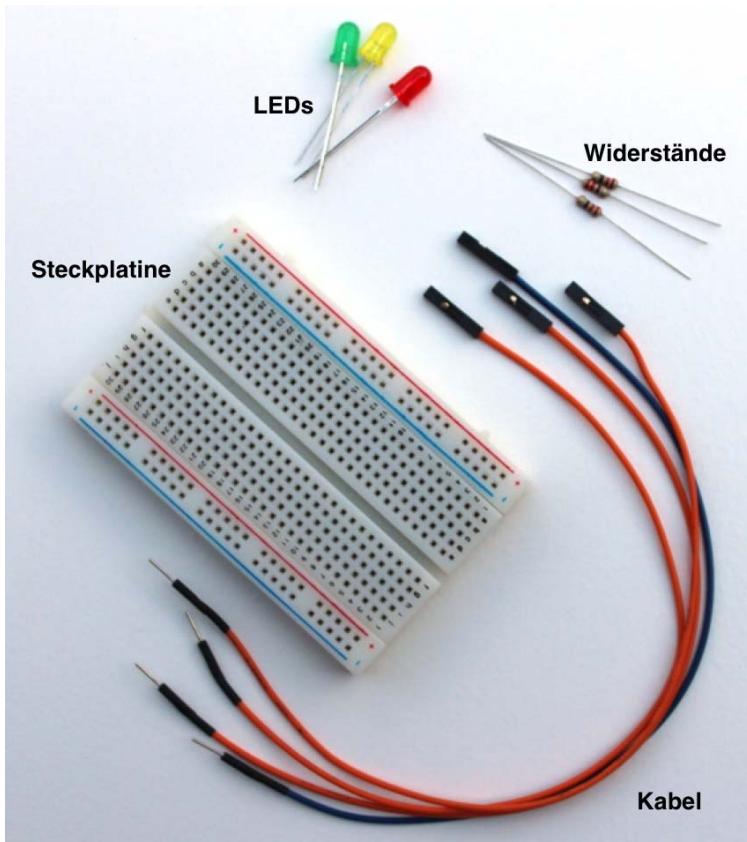


Abb. 9-1 Die benötigten Bauteile

Sie erhalten diese Bauteile in jedem Elektronik- und Bastelladen, beispielsweise bei Reichelt¹, Conrad² oder Watterott³.

Es lohnt sich nicht, LEDs, Widerstände oder Jumperkabel einzeln zu erwerben. Sie bekommen diese Teile wesentlich günstiger, wenn Sie sie in Paketen kaufen.

1) <http://reichelt.de/>

2) <http://conrad.de>

3) <http://watterott.com/>

9.2 Die GPIO-Pins des Pi kennenlernen

Um Ihre eigenen Elektronikprojekte an den Pi anzuschließen, verwenden wir die Erweiterungsleiste oben links am Pi (siehe Abb. 1–1). Sie besteht aus 26 Pins, die in zwei Reihen zu je 13 angeordnet sind. Die obere Reihe enthält die gerade nummerierten Pins und die untere die mit den ungeraden Zahlen. Das bedeutet, der erste Pin der unteren Reihe hat die Nummer 1 und Sie finden die Beschriftung »P1« auf dem Pi genau darunter.

In Abbildung 9–2 sehen Sie die Bezeichnung und Nummerierung der Pins. Beachten Sie, dass sich die Etiketten der Pins zwischen Version 1 und 2 leicht verändert haben. Über die Pins mit der Bezeichnung GROUND können Sie eine gemeinsame Masseverbindung zu Ihren Elektronikprojekten erstellen. Mit den Pins mit der Bezeichnung 3v3 bzw. 5V können Sie externe Geräte, die an den Pi angeschlossen sind, mit 3,3 V oder 5 V versorgen. Der Pi begrenzt die Ausgangsleistung von Pin 1 auf 50 mA, während Pin 2 einen Strom zulässt, der vom USB-Eingangsstrom abhängig ist. Wenn Sie den Pi z.B. mit einem 1-A-Netzteil mit Strom versorgen, können Sie an Pin 2 bis zu 300 mA abnehmen, weil Modell B selbst 700 mA benötigt.

Rev 2	5V	5V	Ground	GPIO14	GPIO15	GPIO18	Ground	GPIO23	GPIO24	Ground	GPIO25	GPIO8	GPIO7
Rev 1	5V	-	Ground	GPIO14	GPIO15	GPIO18	-	GPIO23	GPIO24	-	GPIO25	GPIO8	GPIO7
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
Pin	2	4	6	8	10	12	14	16	18	20	22	24	26
Pin	1	3	5	7	9	11	13	15	17	19	21	23	25
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Rev 1	3v3	GPIO0	GPIO1	GPIO4	-	GPIO17	GPIO21	GPIO22	-	GPIO10	GPIO9	GPIO11	-
Rev 2	3v3	GPIO2	GPIO3	GPIO4	Ground	GPIO17	GPIO27	GPIO22	3v3	GPIO10	GPIO9	GPIO11	Ground

Abb. 9–2 Die GPIO-Pins des Pi

Die Pins 4, 9, 14, 17, 20 und 25 sind für zukünftige Erweiterungen reserviert, sodass Sie sie nicht in eigenen Projekten verwenden können. Die verbleibenden Pins sind universelle Ein- und Ausgabepins (GPIO), die Sie als digitale Eingabe- oder Ausgabeleitungen verwenden können. Beachten Sie, dass die Namen der GPIO-Pins nicht denen auf der Erweiterungsleiste entsprechen.

Sie können die GPIO-Pins z.B. verwenden, um den Zustand einer Drucktaste zu ermitteln oder eine LED an- und auszuschalten. In den Beispielen dieses Kapitels gehen wir davon aus, dass alle GPIO-Pins gleich funktionieren, Sie sollten jedoch wissen, dass einige der Pi-Pins besondere Funktionen haben. Pin 12 unterstützt z.B. die Pulsweitenmodulation (PWM)⁴,

die zum Steuern von Motoren nützlich sein kann. Wenn Sie komplexere Projekte bauen wollen, sollten Sie eine ausführlichere Beschreibung der Pi-Pins zurate ziehen.⁵

9.3 Einen einfachen Schaltkreis erstellen

Zum Aufwärmen bauen wir einen der einfachsten Schaltkreise. Sie schließen eine LED an den Pi an und lassen sie leuchten, solange der Pi eingeschaltet ist. Dazu benötigen Sie eine LED, einen Widerstand, ein Steckbrett und zwei Jumperkabel. Mit diesen Teilen bauen Sie den in Abbildung 9–3 gezeigten Schaltkreis.

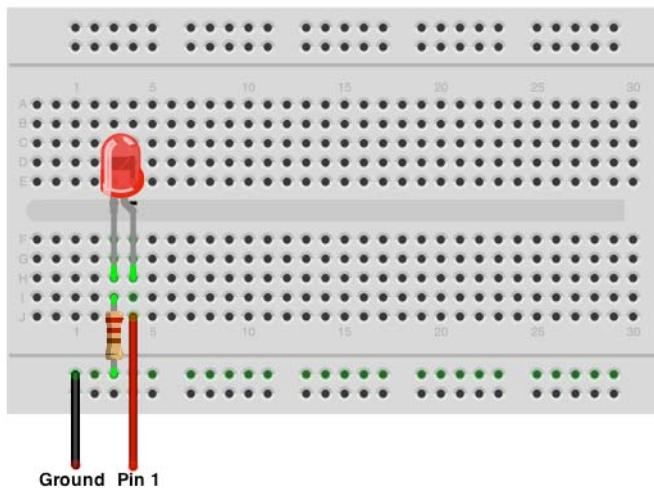


Abb. 9–3 Ein einfacher Schaltkreis

Bevor Sie die Schaltung tatsächlich bauen, sollten Sie wissen, wozu die einzelnen Bauteile dienen und wie sie funktionieren. Steckbretter sind nützlich für Prototypen von Schaltungen. Sie können Bauteile wie LEDs und Widerstände einfach einstecken und müssen sie nicht anlöten. Steckbretter gibt es in verschiedenen Größen und alle sehen sich ähnlich. Auf allen finden Sie viele Stecksockel, die in Spalten angeordnet sind. Die meisten Steckbretter haben außerdem oben und unten noch zwei Sockelreihen.

5) http://elinux.org/RPi_Low-level_peripherals

Der Trick beim Steckbrett besteht darin, dass es alle Sockel in einer bestimmten Reihe und einer bestimmte Spalte miteinander verbindet. In der Schaltung in Abbildung 9–3 verbinden Sie den Massepin des Pi mit der vorletzten Reihe des Steckbretts. Dadurch werden alle Sockel in dieser Reihe mit der Masse des Pi verbunden (und daher haben alle Sockel dieser Reihe eine hellgrüne Farbe). Das Gleiche passiert in den zwei Spalten, die mit der LED verbunden sind. Der Widerstand verbindet den Massepin des Pi indirekt mit einem der beiden LED-Anschlüsse. Zusätzlich verbinden Sie Pin 1 des Pi direkt mit dem anderen Anschluss der LED, indem Sie ihn in dieselbe Spalte stecken.

Übrigens steht LED für *Licht emittierende Diode*, es handelt sich also eigentlich um eine Diode. Dioden sind sehr nützlich, denn sie lassen Strom nur in eine Richtung passieren. Das gilt auch für LEDs, und zusätzlich leuchten diese dabei.

Mit LEDs zu arbeiten ist nicht besonders schwierig, Sie müssen aber auf zwei Dinge achten. Erstens müssen Sie sie richtig anschließen. LEDs haben zwei Anschlüsse, von denen einer etwas kürzer als der andere ist. Der kürzere Anschluss wird Kathode (negativ) genannt und Sie müssen ihn an den Massepin des Pi anschließen. Der längere Anschluss heißt Anode (positiv). Diesen müssen Sie an einen der spannungsführenden Pins oder einen GPIO-Pin anschließen. Sie können Anode und Kathode auch durch einen Blick auf die LED-Fassung unterscheiden. Die flache Seite gehört zur Kathode und die runde zur Anode. In Abbildung 9–3 ist die Anode leicht gebogen.

Außerdem müssen Sie vor eine LED immer einen Widerstand schalten. Tun Sie das nicht, nimmt die LED zu viel Strom auf und wird zerstört. Vereinfacht gesagt begrenzt ein Widerstand den Strom, der durch eine elektrische Verbindung fließt, und schützt die LED. Die Berechnung des benötigten Widerstandswerts für eine gegebene LED ist nicht schwer, sprengt aber den Rahmen dieses Buchs. Merken Sie sich einfach, dass ein Widerstand mit kleinem Wert die LED heller zum Leuchten bringt. Im Zweifelsfall verwenden Sie $330\ \Omega$ oder $470\ \Omega$.

Jetzt können wir die Schaltung basteln. Zuerst verbinden Sie die LED mit dem Steckbrett. Achten Sie auf die richtige Ausrichtung der LED und stecken Sie sie ein. Sie müssen spürbar, aber nicht zu fest drücken, sonst ver-

biegen Sie die Anschlüsse und sie passen nicht mehr. Normalerweise ist es einfacher, Bauteile einzustecken, wenn Sie die Kontakte gekürzt haben. Wenn Sie die Kontakte abschneiden, schützen Sie Ihre Augen mit einer Brille!

Nun kommt der Widerstand, bei dem die Richtung keine Rolle spielt. Bevor Sie den Widerstand einstecken, müssen Sie seine Kontakte umbiegen. Auch hier hilft es, sie zu kürzen.

Dann verbinden Sie die beiden Jumperkabel mit dem Pi und dem Steckbrett. Verbinden Sie die Buchse mit dem Pi und den Stecker mit dem Steckbrett. Achten Sie darauf, am Pi die richtigen Pins zu verwenden, und schalten Sie den Pi ein. Wenn Sie alles richtig angeschlossen haben, leuchtet die LED. Wenn nicht, werfen Sie einen Blick in Abschnitt 9.7.

9.4 Mit den GPIO-Pins eine Leuchtdiode ansteuern

Eine LED zum Leuchten zu bringen ist zwar eine nette Übung, sie wird aber recht schnell langweilig. In diesem Abschnitt lernen Sie, wie Sie eine LED mittels Software steuern. Sie schalten sie über Befehle am Pi an und aus.

Hardware direkt zu programmieren ist normalerweise eine schwierige Sache. Mit dem Projekt WiringPi⁶ wird die Sache zum Kinderspiel. WiringPi ist ein Open-Source-Projekt, das die hässlichen Low-Level-Funktionen hinter einer hübschen und sauberen Schnittstelle verbirgt. Wenn Sie mit dem beliebten Arduino-Projekt⁷ gearbeitet haben, wird Ihnen WiringPi bekannt vorkommen, denn es versucht, die Annehmlichkeiten von Arduino auf den Pi zu bringen. WiringPi macht nicht nur die Programmierung der Pi-Hardware leichter, sondern bietet auch ein kleines Hilfsprogramm für die Kommandozeile namens *gpio*, mit dem Sie die Hardware steuern können, ohne Code zu schreiben.

6) <https://projects.drogon.net/raspberry-pi/wiringpi/>

7) <http://arduino.cc>

Sie installieren WiringPi wie folgt auf dem Pi:

```
pi@raspberry:~$ cd /tmp  
pi@raspberry:~$ sudo apt-get install git-core  
pi@raspberry:~$ sudo apt-get update  
pi@raspberry:~$ sudo apt-get upgrade  
pi@raspberry:~$ git clone git://git.drogon.net/wiringPi  
pi@raspberry:~$ cd wiringPi  
pi@raspberry:~$ ./build
```

Diese Befehle installieren die WiringPi-Bibliotheken und den Befehl `gpio`. Sie können WiringPi aus vielen Programmiersprachen nutzen, wie C, C++, Python, Ruby usw. In diesem Kapitel verwenden Sie es ausschließlich in der Befehlszeile und von kurzen, aber effektiven Shell-Skripten aus.

Alles, was Sie für Ihre ersten interaktiven Elektronikexperimente benötigen, ist der Befehl `gpio`. Er unterstützt viele Optionen und Befehle. Die wichtigsten sind `mode`, `read` und `write`. Mit dem folgenden Befehl setzen Sie den Pin GPIO18 auf den Ausgabemodus:

```
pi@raspberry:~$ gpio -g mode 18 out
```

Alle GPIO-Pins können sich in einem der folgenden Modi befinden: `in`, `out`, `pwm`, `up`, `down` oder `tri`. Für den jetzigen Anwendungsfall sind nur `in` und `out`, also Ein- und Ausgabe interessant. Um an einem GPIO-Pin digitale Signale einzulesen, setzen Sie seinen Modus auf `in`. Setzen Sie ihn auf `out`, wenn Sie Digitalsignale ausgeben wollen. Sie müssen den Modus eines Pins nur einmal festlegen, und er behält ihn, bis Sie einen anderen einstellen.

Nachdem Sie den Modus von GPIO18 auf `out` gesetzt haben, aktivieren Sie ihn wie hier angegeben:

```
pi@raspberry:~$ gpio -g write 18 1
```

Das Ausschalten funktioniert ebenso:

```
pi@raspberry:~$ gpio -g write 18 0
```

Schließlich können Sie den aktuellen Status von GPIO18 mit folgendem Befehl auslesen:

```
pi@raspberry:~$ gpio -g read 18
```

Der Befehl gibt 0 zurück, wenn zurzeit kein Signal anliegt, ansonsten 1.

Mit gpio können Sie die LED im Steckbrett einfach steuern. Sie müssen sie nur an die GPIO-Pins des Pi anschließen, statt direkt an einen Pin zur Stromversorgung. Zum Beispiel können Sie dazu GPIO18 verwenden, das ist Pin Nummer 12. Die richtigen Pins auszuwählen und anzusprechen kann etwas verwirrend sein, da es unterschiedliche Benennungsschemata gibt. Per Vorgabe verwendet WiringPi sein eigenes Nummerierungsschema.⁸ Für Ihre ersten Experimente sollten Sie die »offiziellen« GPIO-Pinnamen verwenden. GPIO akzeptiert diese erfreulicherweise, wenn Sie die Option -g angeben.

Entfernen Sie in Ihrer Schaltung jetzt das Jumperkabel von Pin 1 und verbinden Sie es stattdessen mit Pin 12. Führen Sie dann die folgenden Befehle aus:

```
pi@raspberry:~$ gpio -g mode 18 out  
pi@raspberry:~$ gpio -g write 18 1
```

Diese Befehle schalten die LED ein, und der folgende Befehl schaltet sie aus:

```
pi@raspberry:~$ gpio -g write 18 0
```

gpio unterstützt noch viel mehr Kommandos und Optionen und ein Blick ins Handbuch lohnt sich. Beispielsweise gibt das readall-Kommando den akutellen Zustand aller GPIO-Pins aus.

```
pi@raspberry:~$ gpio readall
```

wiringPi	GPIO	Name	Value
0	17	GPIO 0	Low
1	18	GPIO 1	Low
2	27	GPIO 2	Low

8) <https://projects.drogon.net/raspberry-pi/wiringpi/pins/>

3	22	GPIO 3	Low
4	23	GPIO 4	Low
5	24	GPIO 5	Low
6	25	GPIO 6	Low
7	4	GPIO 7	Low
8	2	SDA	High
9	3	SCL	High
10	8	CEO	Low
11	7	CE1	Low
12	10	MOSI	Low
13	9	MISO	Low
14	11	SCLK	Low
15	14	TxD	High
16	15	RxD	High
-----+-----+-----+-----+			
17	28	GPIO 8	Low
18	29	GPIO 9	Low
19	30	GPIO10	Low
20	31	GPIO11	Low
-----+-----+-----+-----+			

Mit nur ein paar Bauteilen und einem Kommandozeilenwerkzeug haben wir unsere erste Schaltung gebaut. Und noch besser: Sie steuern sie mit dem Raspberry Pi! Im nächsten Abschnitt erstellen Sie ein komplexeres Projekt und setzen dabei die gelernten Techniken ein.

9.5 Einen Alarm für Speichermangel erstellen

Eine LED per Software an- und auszuschalten ist eine wichtige Übung, aber bestimmt kein sehr nützliches Projekt. Normalerweise wollen Sie LEDs nicht manuell steuern, sondern als Statusindikator verwenden. Zum Beispiel setzen viele USB-Geräte LEDs ein, um anzugeben, ob sie momentan Daten lesen oder schreiben.

In diesem Abschnitt verwenden Sie drei LEDs als Statusindikator für die aktuelle Speicherauslastung. Die LEDs haben die gleichen Farben wie eine Ampel. Wenn der freie Pi-Speicher bedenklich klein wird, schalten Sie die rote LED ein. Ist viel Speicher frei, schalten Sie das grüne Licht an. In den anderen Fällen leuchtet die gelbe LED.

In Abbildung 9–4 sehen Sie die Schaltung des Speicheralarms.

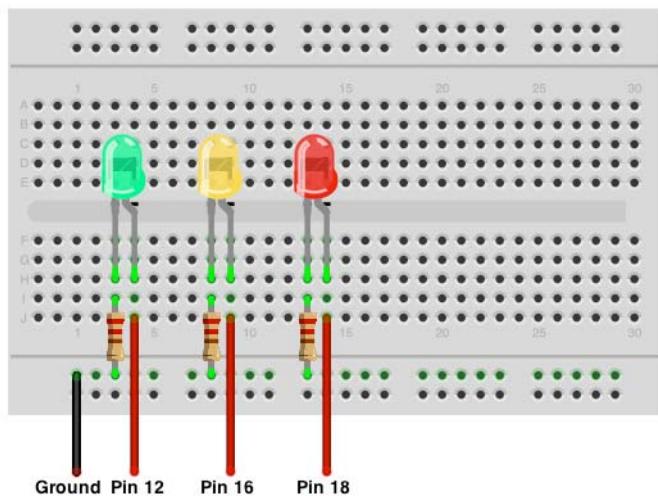


Abb. 9–4 Schaltung des Speicheralarms

Die folgende Abbildung zeigt eine Anwendung in der Praxis. Sie ist der vorher gebauten Schaltung sehr ähnlich. Sie müssen die erste LED-Schaltung einfach noch zweimal nachbauen, um drei LEDs anzusteuern.

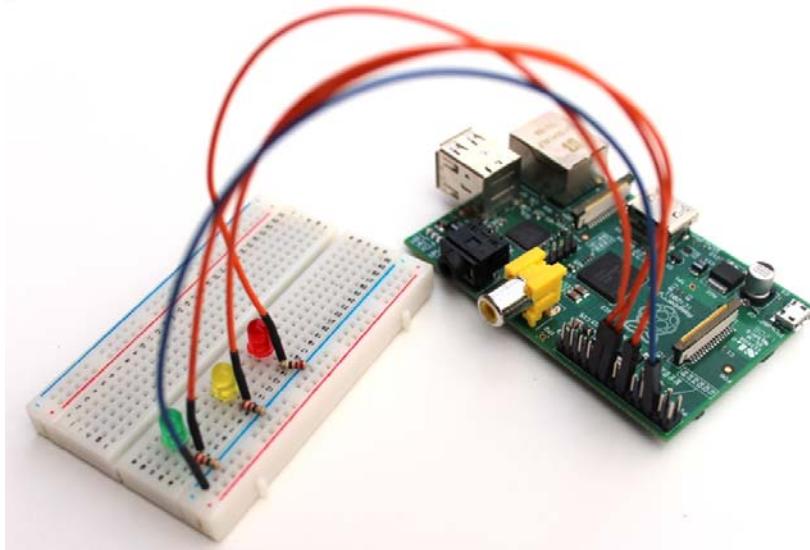


Abb. 9–5 Praktische Anwendung der Schaltung

Damit die Schaltung richtig funktioniert, müssen Sie auch ein Programm schreiben. Sie haben die Wahl aus einem breiten Angebot an Programmiersprachen. Dank dem Befehl gpio können Sie die Software für das Projekt aber auch mit einem einfachen Shell-Skript implementieren. Die folgenden Anweisungen definieren einige Konstanten und Funktionen. Sie nehmen außerdem einige Initialisierungen vor.

gpio/memwatch.sh

```
Zeile 1 #!/bin/bash
-
- green=18
- yellow=23
- red=24
5
- init_leds()
- {
-     for i in $green $yellow $red
-     do
10         gpio -g mode $i out
-         gpio -g write $i 0
-     done
- }
-
15 set_led()
- {
-     led_status=`gpio -g read $1`
-     if [ "$led_status" -ne 1 ]
-     then
20         init_leds
-         gpio -g write $1 1
-     fi
- }
-
25 cleanup()
- {
-     init_leds
-     exit 0
- }
30
- init_leds
- trap cleanup INT TERM EXIT
```

Die ersten drei Zeilen definieren Konstanten für die GPIO-Pins, an die Sie die LEDs angeschlossen haben. Beachten Sie, dass die Zahlen in den Konstanten sich auf die Pinbezeichnungen GPIO18, GPIO23 und GPIO24 beziehen. Sie beziehen sich nicht auf die Pinnummern der Erweiterungsanschlüsse 12, 16 und 18.

Die Funktion `init_leds()` in Zeile 6 setzt den Modus für alle drei LEDs auf Ausgabe und schaltet sie aus. Die Funktion `set_led()` schaltet eine bestimmte LED ein und die anderen aus. Bevor die LED eingeschaltet wird, prüft sie, ob die LED nicht schon brennt. Das stellt sicher, dass die LED nicht flackert, wenn sich der Status nicht verändert hat. Die Funktion `cleanup()` schließlich schaltet alle LEDs aus und beendet das Programm.

Um das Programm zu initialisieren, müssen Sie `init_leds()` aufrufen. Um sicherzustellen, dass das Programm aufräumt, bevor es anhält, verwenden Sie die `trap`-Anweisung in Zeile 32. Dieser Befehl bindet die Methode `cleanup()` an die meisten üblichen Stoppsignale. Wenn Sie also das Skript anhalten, räumt es auf, bevor es sich beendet.

Jetzt haben wir alles richtig initialisiert und Sie können die Alarmlogik programmieren. Dazu müssen wir einen Weg finden, den aktuellen Speicherstatus des Pi zu ermitteln. Der Befehl `free` ist dazu perfekt geeignet.

```
pi@raspberry:~$ free
              total        used        free      shared  buffers   cached
Mem:       190836     40232     150604          0      6252    21068
 -/+ buffers/cache:     12912     177924
Swap:          0          0          0
```

Um den Prozentwert des verfügbaren Speichers zu berechnen, müssen wir die Menge des Gesamtspeichers und die Menge des freien Speichers aus der Ausgabe des Befehls `free` ausschneiden. Danach müssen wir auf Basis des Prozentwerts die richtige LED einschalten. Das funktioniert folgendermaßen:

gpio/memwatch.sh

```
Zeile 1 while :
- do
-     total=`free | grep Mem | tr -s ' ' | cut -d ' ' -f 2` 
-     free=`free | grep Mem | tr -s ' ' | cut -d ' ' -f 4` 
5      available=$(( free * 100 / total ))
-     echo -n "$available% of memory available -> "
-
-     if [ "$available" -le 10 ]
-     then
10       echo "Critical"
-       set_led $red
-     elif [ "$available" -le 30 ]
-     then
-       echo "Low"
15       set_led $yellow
-     else
-       echo "OK"
-       set_led $green
-     fi
20     sleep 10
-   done
```

Der Speichermonitor sollte den aktuellen Speicherstatus regelmäßig prüfen, sodass die gesamte Logik in einer Endlosschleife läuft. Die Zeilen 3 bis 5 berechnen den Prozentwert des freien Speichers, indem sie den Befehl free zweimal aufrufen und die relevante Information ausschneiden. Diese Lösung hat eine potenzielle Fehlerquelle. Wenn sich der aktuelle Speicherverbrauch zwischen zwei free-Aufrufen signifikant ändert, stimmen die Ergebnisse nicht mehr. Das ist zwar nicht sehr wahrscheinlich, kann aber passieren. Bei einem Produktionssystem müssten wir einen zuverlässigeren Weg finden, um die Speicherbelegung zu ermitteln. Für einen Prototyp reicht diese Vorgehensweise aber aus.

Die folgenden Zeilen vergleichen den verfügbaren Speicher mit ein paar Schwellwerten. Ist der verfügbare Speicher kleiner oder gleich 10 Prozent des Gesamtspeichers, schaltet das Programm die rote LED ein. Ist er größer als 10 Prozent und kleiner oder gleich 30 Prozent, wird die gelbe LED eingeschaltet. Ansonsten brennt die grüne LED. Zum Schluss wartet das Programm zehn Sekunden, es verbraucht also nicht viel CPU-Zeit.

Geben Sie das Shell-Skript jetzt ein oder laden Sie es durch einen Klick auf den Dateinamen herunter und kopieren Sie es dann auf den Pi. Die folgende Anweisung macht das Skript ausführbar:

```
pi@raspberry:~$ chmod a+x memwatch.sh
```

Dann starten Sie es wie folgt:

```
pi@raspberry:~$ ./memwatch.sh
78% of memory available -> OK
```

Um den Speicheralarm zu testen, starten Sie den LXDE-Desktop und führen Sie das Skript im Terminal aus. Öffnen Sie dann einige Anwendungen und beobachten Sie, wie der verfügbare Speicher abnimmt.

9.6 Den GPIO-Status in einem Browser anzeigen

In Kapitel 6 haben Sie gelernt, wie Sie einen Webserver und PHP5 auf dem Pi aufsetzen. Jetzt können Sie diesen Webserver verwenden, um die aktuelle Speicherauslastung Ihres Pi in einem Webbrowser anzuzeigen. Kopieren Sie das folgende PHP-Programm in das Verzeichnis /var/www auf Ihrem Pi.

```
gpio/memwatch.php
```

```
<?php
function led_is_on($number) {
    $status = trim(@shell_exec("/usr/local/bin/gpio -g read " .
                               $number));
    if ($status == "0") {
        return False;
    } else {
        return True;
    }
}

$green = 18;
$yellow = 23;
$red = 24;

echo "<h1>Memory Usage is ";
if (led_is_on($green)) {
    echo "OK";
}
```

```

elseif (led_is_on($yellow)) {
    echo "Low";
}
elseif (led_is_on($red)) {
    echo "Critical";
}
else {
    echo "Unknown";
}
echo ".</h1>";
?>

```

Gehen Sie dann mit Ihrem Browser auf die Datei `memwatch.php`. Wenn Ihre IP-Adresse beispielsweise 192.168.2.109 ist, verwenden Sie die URL `192.168.2.109/memwatch.php`. Ihr Browser zeigt eine kurze Meldung an, die den aktuellen Speicherstatus des Pi wiedergibt.

Selbst wenn Sie PHP noch nie verwendet haben, sollten Sie verstehen können, was das Programm macht. In der Funktion `led_is_on()` ruft es den Befehl `gpio` auf und liest den aktuellen Status eines GPIO-Pins. Ist der Pin bereits an, gibt die Funktion `True` zurück, wenn nicht, `False`. Dann prüft das Programm, welche der LEDs brennt, und gibt eine entsprechende Meldung aus.

Wie Sie gesehen haben, ist es einfach, in Ihrem Netzwerk den Status eines Elektronikgeräts anzuzeigen. Natürlich können Sie die Seiten farbenfroher gestalten und in eine richtige Ampel verwandeln, das geht jedoch über den Rahmen dieses Buchs hinaus.

9.7 Wenn es nicht klappt ...

Das Bauen eigener Elektronikgeräte ist keine Hexerei, ganz einfach ist es aber auch nicht. Wenn Sie noch nie mit einem Steckbrett, LEDs und Widerständen gearbeitet haben, kann viel schiefgehen. Auch mit sehr viel Erfahrung werden Sie noch Fehler machen.

Wenn etwas nicht funktioniert wie gewünscht – keine Panik! Meist ist die Ursache des Problems recht einfach. Zuerst sollten Sie prüfen, ob Sie alle Komponenten mit den richtigen Pins verbunden haben. Dann prüfen Sie die Polung der LEDs. Stellen Sie sicher, dass alle Bauteile korrekt eingesteckt sind. Manchmal ist es schwierig, die Bauteile in Steckbretter einzubauen, insbesondere, wenn das Steckbrett neu ist.

Vergessen Sie nicht, das Netzteil einzustecken, und fahren Sie mit Ihrem Projekt nicht fort, wenn nicht alle Vorstufen funktionieren.

9.8 Die nächsten Schritte

In diesem Kapitel haben Sie gelernt, wie Sie Ihre eigenen kleinen Elektronikgeräte bauen und sie mit dem Raspberry Pi steuern. Auch wenn Sie nur wenige und günstige Bauteile verwendet haben, konnten Sie etwas Nützliches herstellen, das auch noch Spaß macht.

Das nächste Kapitel wird Ihnen noch mehr spannende Dinge präsentieren. Sie werden digitale und analoge Sensoren an den Pi anschließen, um zum Beispiel Bewegungen zu erkennen oder die aktuelle Temperatur zu messen.

10 Arbeiten mit Digital- und Analogsensoren

Sensoren finden sich heute überall und viele von ihnen sind so allgegenwärtig, dass Sie sie nicht mehr wahrnehmen. Wenn Sie ein Lebensmittelgeschäft betreten, erwarten Sie, dass sich die Türen automatisch öffnen. Mit Temperatursensoren schaltet Ihr Auto die Heizung automatisch ein, um den Beschlag auf den Außenspiegeln zu entfernen. Abstandssensoren helfen Ihnen dabei, in die kleinste Parklücke zu gelangen.

Natürlich finden Sie auch viele Sensoren in Ihrem Tablet-PC und in Ihrem Smartphone. Beschleunigungssensoren drehen Ihr Display beispielsweise automatisch ins Hoch- oder Querformat und Sie steuern damit Videospiele.

Sensoren an einen PC anzuschließen ist normalerweise nicht so leicht. Erfreulicherweise ist der Pi anders und Sie können ihn ohne großen Aufwand mit vielen intelligenten Sensoren verbinden. Es ist zwar nicht ganz so einfach wie der Anschluss von Sensoren an ein Microcontroller-Board wie Arduino, in diesem Kapitel erfahren Sie jedoch, wie Sie den Pi mit Digital- und Analogsensoren verbinden.

10.1 Was Sie benötigen

Um die Projekte dieses Kapitels nachzubauen, benötigen Sie die in Abbildung 10-1 aufgelisteten Teile:

- Ein Steckbrett in halber Baugröße
- Einen PIR-Sensor
- Einen Temperatursensor Typ TMP36
- Einen Analog-Digital-Wandler Typ MCP3008-I/P
- Vier Jumperkabel mit Stecker und Buchse
- Ein paar Kabel

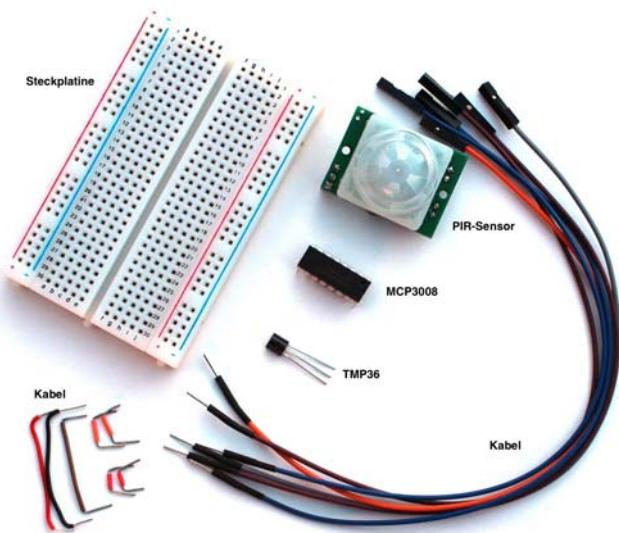


Abb. 10-1 Die benötigten Bauteile

In Abschnitt 9.1 »Was Sie benötigen« haben Sie bereits erfahren, wo Sie Steckbrett und Kabel erhalten. Die Sensoren und den MCP3008 erhalten Sie bei Digikey (<http://www.digikey.com/>), Mouser (<http://de.mouser.com>) oder bei Segor Electronics (<http://www.segor.de>). Wenn Sie auch den Parallax-PIR-Sensor verwenden, achten Sie darauf, dass Sie Rev A erhalten¹. Die neue Version (Rev B) ist etwas besser, seine Ausgangsspannung kann jedoch für den Pi zu hoch sein. Rev A kostet momentan weniger als 5 Euro und reicht für die meisten Anwendungen. Egal, welchen PIR-Sensor Sie verwenden: seine Signalspannung darf keinesfalls höher als 3,3V sein!

Achten Sie außerdem darauf, dass Sie die richtige Version des MCP3008 kaufen, denn es gibt ihn in verschiedenen Gehäuseformen. Sie benötigen den MCP3008-I/P (16 Pin), da Sie ihn mit dem Steckbrett einsetzen wollen.

Auch den TMP36 gibt es in verschiedenen Ausprägungen. Sie benötigen den TMP36GZ. Er ist auch bei deutschen Händlern verfügbar, aber manchmal müssen Sie etwas länger danach suchen. Alternativ können Sie den LM35 CZ verwenden. Sie bekommen ihn zum Beispiel bei Watterott

1) Suchen Sie bei www.parallax.com nach Produktnr. 910-28027.

(<http://www.watterott.com/de/LM35CZ>), müssen aber den Code für das Auslesen des Sensors ändern, weil der Sensor geringfügig andere Signale sendet als der TMP36. Ich werde das im entsprechenden Abschnitt erklären.

Warnung: Schließen Sie niemals Sensoren mit einer Ausgangsspannung von mehr als 3,3 V direkt an den Pi an. Sie beschädigen dadurch Ihren Pi! Seien Sie also vorsichtig, wenn Sie für Ihren Pi neue Bauteile anschaffen.

10.2 Bewegungserkennung mit dem Pi

Sie werden vermutlich mehrmals die Woche oder sogar mehrmals am Tag von Bewegungssensoren profitieren. Diese schalten vielleicht automatisch das Licht an, wenn Sie im Dunkeln nach Hause kommen, oder die Beleuchtung des Pausenraums an Ihrer Arbeitsstelle. In diesem Abschnitt erfahren Sie, wie diese Detektoren funktionieren, und lernen, wie Sie den Pi als Bewegungssensor betreiben.

Viele Bewegungssensoren verwenden passive Infrarotsensoren (PIR)². Ein PIR-Sensor misst ständig Infrarotlicht und erkennt, wenn sich etwas im Infrarotspektrum ändert. Das ist alles, was zur Bewegungserkennung erforderlich ist, da beinahe jedes Objekt Infrarotstrahlung abgibt. Dies gilt z.B. für alle Objekte vor Ihrem Haus: für den Boden, ein Fahrrad, eine Mülltonne usw. All diese Objekte geben eine recht konstante Menge an Infrarotstrahlung ab, die sich nicht spontan verändert. Wenn aber ein Mensch oder ein Tier vor Ihrem Haus auftaucht, erkennt der Sensor eine große Abweichung und gibt ein Signal aus.

2) http://en.wikipedia.org/wiki/Passive_infrared_sensor

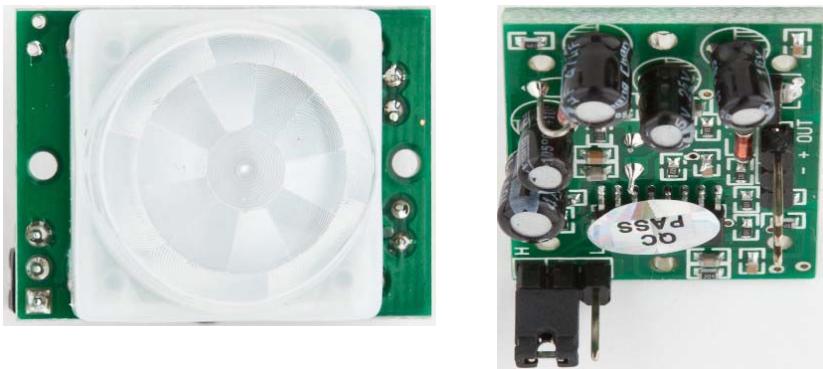


Abb. 10–2 Ober- und Unterseite eines passiven Infrarotsensors³

Den PIR-Sensor mit dem Pi verbinden

Der Aufbau eines PIR-Sensors ist zwar ziemlich komplex, er ist aber einfach einzusetzen. In Abbildung 10–2 können Sie den Sensor von Parallax (Rev A) sehen, den ich in den Beispielen dieses Kapitels verwende. Der Sensor hat eine Brücke, mit der Sie sein Verhalten verändern können. Stellen Sie sicher, dass sie in Position H steht, die Brücke also den Pin neben dem H belegt.⁴

Der Sensor verfügt über drei Pins, die Sie mit Kabeln an den Pi anschließen müssen (siehe Abbildung 10–3). Schließen Sie den 5V-Pin des Pi an den Versorgungsspannungspin des Sensors an und die Masse des Pi an den Masseanschluss des Sensors. Schließen Sie dann den Signalpin des Sensors an den Pin GPIO23 am Pi an. Normalerweise sind die Pins am Sensor beschriftet. Wenn Sie sich nicht sicher sind, werfen Sie einen Blick in das Sensor-Datenblatt.

3) Quelle: Schmidt, Arduino, dpunkt.verlag, 2012

4) Unter <http://www.ladyada.net/learn/sensors/pir.html> finden Sie eine exzellente Einführung in die Sensoren.

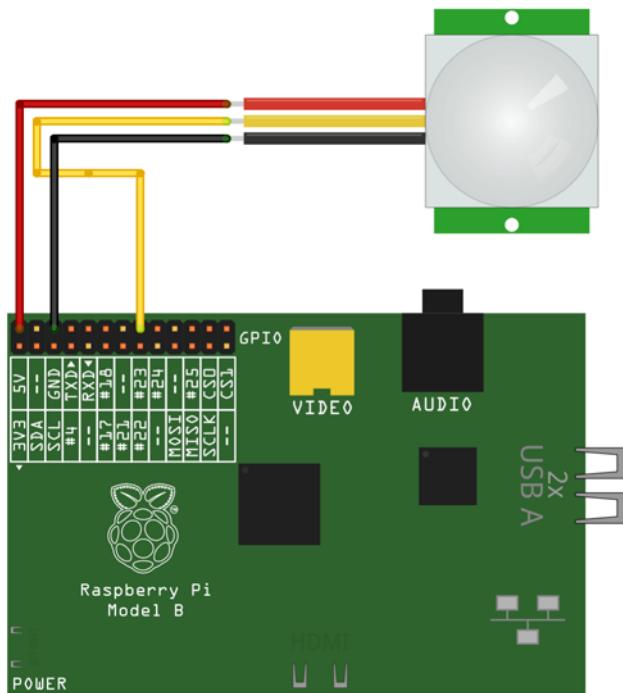


Abb. 10-3 Der PIR-Schaltkreis

Alle digitalen PIR-Sensoren funktionieren mehr oder weniger auf dieselbe Weise. Solange sie keine Bewegung erkennen, geben die Sensoren keine Spannung über ihren Signalpin aus. Wenn sie eine Bewegung wahrnehmen, geben sie ein positives Signal aus, also eine bestimmte Spannung, die Sie im Datenblatt des Sensors nachlesen können.

Nochmals eine Warnung! Schließen Sie niemals Sensoren mit einer Ausgangsspannung von mehr als 3,3 V direkt an den Pi an. Sie beschädigen dadurch Ihren Pi!

Einen PIR-Sensor steuern

Wenn die Verkabelung fertig ist, können wir den Sensor mit etwas Software steuern. Für viele Programmieraufgaben mit dem Pi ist die Sprache Python⁵ eine gute Wahl. Sie ist einfach zu erlernen und die RPi-Biblio-

5) <http://www.python.org/>

the⁶ bietet viele bequeme Funktionen, um die GPIO-Pins des Pi zu steuern. Raspbian wird bereits mit Python ausgeliefert, RPi müssen wir aber noch installieren.

```
pi@raspberry:~$ sudo apt-get update  
pi@raspberry:~$ sudo apt-get install python-dev python-rpi.gpio
```

Damit sind die Vorbereitungen abgeschlossen und wir können eine neue Python-Klasse für den PIR-Sensor definieren.

Sensors/pir.py

```
Zeile 1 import RPi.GPIO as GPIO  
2  
3 class PassiveInfraredSensor:  
4     def __init__(self, pin):  
5         self.pin = pin  
6         GPIO.setmode(GPIO.BCM)  
7         GPIO.setup(self.pin, GPIO.IN)  
8  
9     def motion_detected(self):  
10        return GPIO.input(self.pin)
```

Selbst wenn Sie Python noch nie verwendet haben, sollten Sie verstehen können, was das Programm macht. Bevor wir den Quellcode zeilenweise ansehen, sollten Sie wissen, dass Python Leerzeichen anders behandelt als die meisten anderen Programmiersprachen. Statt Codeblöcke mit geschweiften Klammern oder mit Schlüsselwörtern wie begin oder end abzugrenzen, verwendet Python Einrückungen. Es spielt keine Rolle, ob Sie Leerzeichen oder Tabulatoren verwenden, um einen Codeblock einzurücken, Sie müssen dabei nur konsistent arbeiten. Wenn Sie also die erste Zeile eines Blocks mit vier Leerzeichen eingerückt haben, müssen Sie auch die folgenden Zeilen um vier Leerzeichen einrücken.

In der ersten Zeile importieren wir die GPIO-Funktionen der RPi, sodass wir sie in unserem Code verwenden können. Dann definieren wir eine neue Klasse namens `PassiveInfraredSensor`. Immer wenn wir ein neues `PassiveInfraredSensor`-Objekt erzeugen, ruft Python die Funktion `__init__` auf. Die Funktion erwartet zwei Argumente: das neu erstellte Objekt (`self`) und die Nummer des Pins, den wir an den Sensor ange-

6) <http://code.google.com/p/raspberry-gpio-python/>

schlossen haben (`pin`). Das erste Argument wird von Python automatisch initialisiert.

In Zeile 5 legen wir die Pinnummer im aktuellen Objekt ab und danach setzen wir die Nummerierung für die Pins mit der Funktion `GPIO.setmode`. Wir übergeben ihr den Wert `GPIO.BCM`, sodass die RPi-Bibliothek die Pinnummern mittels der Broadcom-Definition (siehe die oberen und unteren Zeilen in Abb. 9–2) interpretiert. In unserem Fall ist die Pinnummer 23, da wir den Signalpin des Sensors an Pin GPIO23 am Pi angeschlossen haben.

Alternativ können wir den Modus auf `GPIO.BOARD` setzen. In diesem Fall interpretiert RPi die Pinnummern, wie sie auf der Platine des Pi definiert sind, und wir müssen 16 statt unserer aktuellen Einstellung verwenden (siehe die beiden Zeilen in Abb. 9–2, die mit Pin beginnen). Schließlich stellen wir den Pin, an den der Sensor angeschlossen ist, auf Eingang, indem wir `GPIO.setup` in Zeile 7 aufrufen.

Danach definieren wir eine Methode namens `motion_detected`, die `True` zurückgibt, wenn der Sensor eine Bewegung erkannt hat, und `False`, wenn nicht. Dazu wird `GPIO.input` aufgerufen und der Funktion wird unsere Signalpinnummer übergeben. Python setzt auch hier das Argument `self` automatisch für uns.

Unsere Klasse `PassiveInfraredSensor` ist jetzt vollständig, sodass wir damit den Bewegungssensor erstellen können. Das folgende Programm prüft ständig, ob der PIR-Sensor eine Bewegung erkannt hat. Es gibt eine Meldung aus, wenn sich jemand bewegt und wenn sich innerhalb eines Zeitraums von zwei Sekunden niemand bewegt hat.

Sensors/pir_demo.py

```
Zeile 1 from pir import *
- import time
-
- PIR_PIN = 23
5 pir = PassiveInfraredSensor(PIR_PIN)
-
- last_state = False
- while True:
-     if (pir.motion_detected() == True):
10         if (last_state == False):
```

```

-     print "Bewegung erkannt"
-     last_state = True
- else:
-     if (last_state == True):
15      print "Keine Bewegung erkannt"
-     time.sleep(2)
-     last_state = False;

```

In den ersten beiden Zeilen importieren wir die Klasse `PassiveInfrared-Sensor` und die Python-Funktionen, um Datum und Uhrzeit zu bearbeiten. Dann definieren wir eine Konstante namens `PIR_PIN` und setzen sie auf die Nummer unseres Signalpins. Wir verwenden die Konstante in der folgenden Zeile, wenn wir erstmalig ein neues `PassiveInfraredSensor`-Objekt erstellen.

Der Erkennungsalgorithmus beginnt in Zeile 7. Wir legen den letzten Status des PIR-Sensors in einer Variablen namens `last_state` ab. Dann starten wir eine Endlosschleife und prüfen, ob der PIR-Sensor gerade eine Bewegung erkannt hat. Wenn ja, prüfen wir, ob dies eine »neue« Bewegung ist oder ob wir sie bereits vorher erkannt haben. Ist sie neu, geben wir die Meldung »Bewegung erkannt« aus.

Hat der PIR-Sensor keine Bewegung erkannt, prüfen wir, ob vorher eine Bewegung erkannt wurde. In diesem Fall geben wir die Meldung »Keine Bewegung erkannt« aus und warten zwei Sekunden, bevor wir wieder von vorn beginnen. Damit stellen wir sicher, dass nur dann eine Meldung ausgegeben wird, wenn sich der Status tatsächlich verändert hat. Dadurch macht unser Bewegungssensor keinen so nervösen Eindruck.

Führen Sie das Programm jetzt aus, bewegen Sie sich etwas vor dem PIR-Sensor und bleiben Sie von Zeit zu Zeit stehen.

```

pi@raspberry:~$ sudo python pir_demo.py
Bewegung erkannt
Keine Bewegung erkannt

```

Das war alles, um den Pi in einen Bewegungssensor zu verwandeln. Eine Meldung auszugeben ist nicht besonders spannend, aber Sie können das Programm etwas verbessern, indem Sie sich eine E-Mail senden lassen oder eine Lampe einschalten. So verwandeln Sie Ihren Pi in eine Alarmanlage oder machen ihn zur Basis für Ihre Hausautomatisierung.

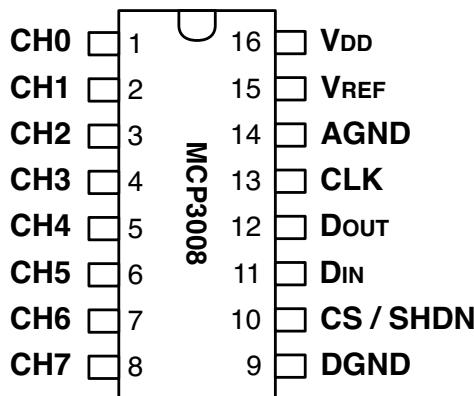


Abb. 10–4 Das Anschlusschema des MCP3008

In Abbildung 10–4 sehen Sie die Pins des MCP3008. Sie können die Pins 1 bis 8 auf der linken Seite verwenden, um bis zu 8 Analogsensoren anzuschließen. Die Hälfte der Pins auf der rechten Seite dient der Stromversorgung und Masse (VDD, VREF, AGND und DGND). Mit den verbleibenden vier Pins (CLK, DOUT, DIN und CS) können Sie die anliegende Spannung als Digitalzahl auslesen.

Der MCP3008 hat eine Auflösung von 10 Bit, sodass Sie sich vielleicht fragen, wie Sie 10 Bit mit nur vier Leitungen übertragen können. Man würde erwarten, dass der Chip 10 Pins für die Ausgabe des anliegenden Analogsignals besitzt. Das ist zwar grundsätzlich richtig, aber 10 Pins wären auch eine ziemliche Verschwendung. Nicht nur der MCP3008 müsste mehr Anschlüsse haben, sondern auch der Pi.

Aus diesem Grund wendet der MCP3008 einen Trick an, um seine Werte auszugeben. Er implementiert das Serial Peripheral Interface (SPI)⁸. SPI ermöglicht Ihnen eine synchrone serielle Verbindung zwischen einem Haupt- und mehreren Nebengeräten. Haupt- und Nebengeräte kommunizieren über einen seriellen Bus und in unserem Fall ist Pi der Master und der MCP3008 das Nebengerät. Immer wenn der Pi ein Analogsignal lesen möchte, sendet er eine Nachricht und der MCP3008 antwortet.

8) http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus

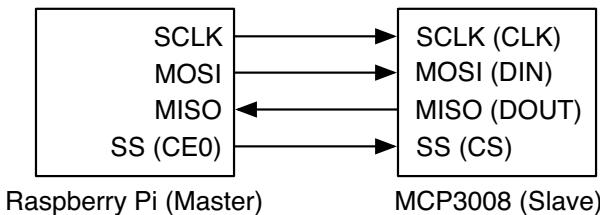


Abb. 10–5 So verbinden Sie zwei SPI-Geräte.

Um einen SPI-Bus zwischen zwei Geräten einzurichten, benötigen Sie vier Leitungen. In Abbildung 10–5 sehen Sie, wie Sie den Pi mit dem MCP3008 verbinden, sodass beide mittels SPI kommunizieren können. Machen Sie sich keine Sorgen. Sie sehen den Rest des Schaltplans im Abschnitt »Jetzt spielt alles zusammen«. Leider ist SPI keine besonders exakte Spezifikation, weshalb verschiedene Hersteller für ihre Pins unterschiedliche Bezeichnungen verwenden. Mit der SCLK-Leitung synchronisieren Haupt- und Nebengerät ihre Arbeit über ein gemeinsames Taktsignal. Die Leitung MOSI (Master Out, Slave In) dient zum Senden von Daten vom Haupt- zum Nebengerät, während MISO (Master In, Slave Out) für die Rückübertragung vom Neben- zum Hauptgerät sorgt. Mit der Leitung SS (Slave Select) wählt das Hauptgerät ein Nebengerät aus, das eine Aufgabe übernehmen soll.

Die SPI-Kommunikation in Software abzubilden ist nicht schwer, die Pi-Hardware kann das jedoch schon von Haus aus. Das gilt auch für Raspbian, weshalb Sie im nächsten Abschnitt erfahren, wie Sie sie aktivieren.

SPI auf dem Pi aktivieren

Per Voreinstellung aktiviert Raspbian SPI nicht, sodass Sie erst ein paar Konfigurationsdateien anpassen müssen. Zuerst müssen Sie SPI aus der Modulsperrliste entfernen. Dazu editieren Sie `/etc/modprobe.d/raspi-blacklist.conf`:

```
pi@raspberry:~$ sudo nano /etc/modprobe.d/raspi-blacklist.conf
```

Bei Erscheinen dieses Buchs enthielt diese Datei nur zwei Zeilen, um SPI und I²C zu deaktivieren. Sie sieht wahrscheinlich wie folgt aus:

```
blacklist spi-bcm2708  
blacklist i2c-bcm2708
```

Um die SPI-Unterstützung aus der Sperrliste zu entfernen, müssen Sie die entsprechende Zeile auskommentieren, sodass die Datei hinterher wie folgt aussieht:

```
#blacklist spi-bcm2708  
blacklist i2c-bcm2708
```

Alternativ können Sie die Zeile natürlich auch löschen. Speichern Sie anschließend die Datei und fügen Sie das SPI-Modul der Liste der Module hinzu, die Pi automatisch beim Systemstart lädt.

```
pi@raspberry:~$ sudo nano /etc/modules
```

Fügen Sie eine Zeile hinzu, die nur das Wort »spidev« enthält, und speichern Sie die Datei. Dann starten Sie den Pi erneut.

```
pi@raspberry:~$ sudo reboot
```

Ihr Pi sollte jetzt zwei SPI-Geräte unterstützen, die Sie mit dem folgenden Befehl suchen können:

```
pi@raspberry:~$ ls /dev/spidev*  
/dev/spidev0.0 /dev/spidev0.1
```

Im folgenden Abschnitt erwecken wir eines dieser Geräte zum Leben, indem wir einen MCP3008 und einen Temperatursensor an den Pi anschließen.

Jetzt spielt alles zusammen

Da wir jetzt wissen, wie SPI prinzipiell funktioniert, müssen wir den Pi mit dem MCP3008 und den MCP3008 mit dem Temperatursensor TMP36 verbinden. In Abbildung 10–6 sehen wir, wie das geht.

Wir müssen die Versorgungsspannung und die Masse an verschiedenen Stellen auf dem Steckbrett anschließen. Daher verbinden wir sie mit den beiden Reihen links am Steckbrett. Von dort können wir die Spannung und Masse einfach mithilfe von kurzen Leitungen an andere Stellen verteilen.

Dann stecken wir den MCP3008-Baustein ein und achten auf die richtige Polung. Integrierte Schaltkreise (ICs) haben oben an einem Ende eine Einbuchtung im Gehäuse und der MCP3008 ist keine Ausnahme. Pin 1 befindet sich links von der Einbuchtung. Stecken Sie ihn vorsichtig ins Steckbrett. Sie müssen zwar etwas fester drücken, aber vorher sollten Sie sicherstellen, dass alle Beinchen in die Anschlüsse passen. Wenn Sie den Chip entfernen wollen, verwenden Sie einen kleinen Schraubenzieher als Hebel.

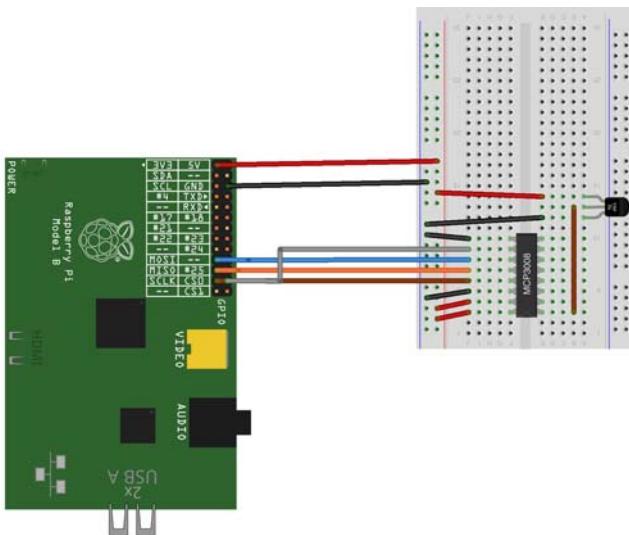


Abb. 10–6 Der TMP36-Schaltkreis

Seien Sie auch vorsichtig, wenn Sie den MCP3008 mit dem Pi verbinden. Tabelle 1 zeigt detailliert, wie Sie die GPIO-Pins des Pi an den MCP3008 anschließen.

Raspberry Pi Pin	MCP3008
1 (3.3V)	16 (VDD)
1 (3.3V)	15 (VREF)
6 (GND)	14 (AGND)
6 (GND)	9 (DGND)
23 (SCLK)	13 (CLK)

Raspberry Pi Pin	MCP3008
21 (MISO)	12 (DOUT)
19 (MOSI)	11 (DIN)
24 (CE0)	10 (CS)

Tab. 1 Pinbelegung für den Anschluss des Pi an den MCP3008

Das Anschließen des TMP36-Sensors ist einfach, denn er hat nur drei Pins. Stecken Sie den Sensor ins Steckbrett und achten Sie darauf, dass die Polung stimmt. Schließen Sie die Versorgungsspannung an den 3,3V-Pin des Pi an und Masse an die Masse des Pi. Schließen Sie dann den Signalpin des TMP36 an Pin 1 (CH0) des MCP3008 an.

Damit ist die Verdrahtung abgeschlossen. Jetzt müssen wir ein Programm schreiben, um letztendlich die Temperatur zu ermitteln.

Den MCP3008 steuern

Der MCP3008 ist ein SPI-Gerät, weshalb wir einen Weg finden müssen, mit ihm über die SPI-Hardware des Pi zu kommunizieren. Dies haben erfreulicherweise bereits mehrere Personen für uns erledigt, und py-spidev⁹ ist eine kleine Bibliothek, die uns alles Notwendige bietet. Sie installieren sie wie folgt:

```
pi@raspberry:~$ git clone git://github.com/doceme/py-spidev
pi@raspberry:~$ cd py-spidev
pi@raspberry:~$ sudo python setup.py install
```

Wir müssen nur Folgendes tun, um eine MCP3008-Klasse mittels Python zu implementieren:

Sensors/mcp3008.py

```
Zeile 1 import spidev
-
-
- class MCP3008:
-     def __init__(self, bus = 0, client = 0):
5         self.spi = spidev.SpiDev()
-         self.spi.open(bus, client)
-
-
```

9) <https://github.com/doceme/py-spidev>

```
- def analog_read(self, channel):
-     if (channel < 0 or channel > 7):
10         return -1
-     result = self.spi.xfer2([1, (8 + channel) << 4, 0])
-     return ((result[1] & 3) << 8) + result[2]
```

Es ist nur eine kleine Klasse, die aber ein paar Erklärungen benötigt. In der ersten Zeile importieren wir die spidev-Bibliothek, die wir zuvor installiert haben. Dann definieren wir eine neue Klasse namens MCP3008. Wie vorher bei der PassiveInfraredSensor-Klasse definieren wir eine Methode namens `__init__`, die immer dann aufgerufen wird, wenn wir ein neues MCP3008-Objekt erzeugen. `__init__` erwartet drei Parameter: die gerade erzeugte Instanz (`self`), die auf dem SPI-Bus zu belegende Nummer (`bus`) und die Nummer des Nebengeräts (`slave`), mit dem wir sprechen möchten (`client`). In Zeile 5 erzeugen wir ein neues SpiDev-Objekt und öffnen in der folgenden Zeile eine Verbindung mit dem Gerät.

Bis jetzt haben wir nur einen Kommunikationskanal mit einem SPI-Gerät eingerichtet, was bedeutet, dass der Aufruf unter `__init__` nicht spezifisch für den MCP3008 ist. Er wäre für jedes andere SPI-Gerät identisch. Die spezifischen Dinge passieren in `analog_read`. Diese Methode nimmt eine Portnummer und gibt den aktuellen Wert des Analogsensors zurück, der am Port anliegt. Wir prüfen, ob die Portnummer zwischen 0 und 7 liegt, und wenn das nicht der Fall ist, geben wir -1 zurück.

In Zeile 11 senden wir eine Meldung mit der Methode `xfer2` des `Spidev`-Objekts an den MCP3008. `xfer2` erwartet ein Byte-Array und sendet es über das SPI-Protokoll. In unserem Fall senden wir drei Bytes und Sie fragen sich vielleicht, was diese bedeuten. Sie finden die Antwort in den Kapiteln 5 und 6 des Datenblatts des MCP3008¹⁰. Das Datenblatt beschreibt detailliert, welche Daten der MCP3008 erwartet und welche Daten er zurücksendet. Die Struktur der Eingabemeldung sieht folgendermaßen aus:

$$0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad | \quad 1 \quad D2 \quad D2 \quad D0 \quad x \quad x \quad x \quad x \quad | \quad x \quad x$$

Die meisten Bits dieser drei Bytes haben einen konstanten Wert. Nur die drei Bits namens D0, D1 und D2 variieren und enthalten die Nummer des

10) <http://ww1.microchip.com/downloads/en/DeviceDoc/21295d.pdf>

auszulesenden Kanals. Das erste Byte ist immer 1, das letzte immer 0 und das zweite Byte enthält eine leicht veränderte Fassung der Kanalnummer. Wir setzen alle mit x markierten Bits auf 0.

Die Antwort des MCP3008 besteht ebenfalls aus drei Bytes. Sie bedeuten:

[?]	[?]	[?]	[?]	[?]	[?]	[?]	[?]	[?]	[?]	[?]	[?]	[?]	[0]	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	----	----	----	----	----	----	----	----	----	----

Für uns sind nur die unteren 10 Bits interessant, der Rest kann ignoriert werden. In Zeile 12 extrahieren wir die interessanten Bits und geben sie als einzelne Zahl zurück.

Mit der Klasse MCP3008 ist es einfach, die aktuelle Temperatur auszulesen:

Sensors/tmp36_demo.py

```
Zeile 1 from mcp3008 import *
- import time
-
- mcp = MCP3008()
5 TMP36_CHANNEL = 0
-
- while True:
-     analog_value = mcp.analog_read(TMP36_CHANNEL)
-     voltage = 3.3 * analog_value / 1024
10    temperature_c = (voltage * 1000 - 500) / 10
-     temperature_f = 9.0 / 5.0 * temperature_c + 32.0
-     print "Temperatur: %.1fC (%.1fF)" % (temperature_c,
-                                               temperature_f)
-     time.sleep(1)
```

Nach dem Importieren der Klasse MCP3008 und der Zeitfunktionen von Python erzeugen wir in Zeile 4 ein neues MCP3008-Objekt. Außerdem definieren wir eine Konstante für den Kanal, über den wir uns mit dem TMP36-Sensor verbinden.

Wir starten eine Endlosschleife und lesen den aktuellen Wert des TMP36 aus, indem wir `analog_read` aufrufen. In Zeile 9 wandeln wir diesen Wert in die tatsächlich vom Sensor ermittelte Spannung um. Die folgende Zeile berechnet die tatsächliche Temperatur in Grad Celsius über eine Formel, die Sie im Datenblatt des Sensors¹¹ finden. (Wenn Sie den LM35 CZ verwenden, müssen Sie die Zeile umwandeln: `temperature_c = voltage * 100;`

Beachten Sie, dass Sie auf diese Weise keine negativen Temperaturen messen können.) Wir konvertieren den Celsius-Wert zusätzlich in Grad Fahrenheit und geben sie beide aus. Dann warten wir eine Sekunde und messen die Temperatur erneut.

Führen Sie das Programm aus und die Ausgabe sollte wie folgt aussehen:

```
pi@raspberry:~$ sudo python tmp36_demo.py
Temperature: 20.9C (69.6F)
Temperature: 20.9C (69.6F)
```

Das war nicht ganz einfach, aber schließlich haben wir es geschafft! Jetzt können wir mit dem Pi die aktuelle Temperatur messen. Sie haben nicht nur gelernt, wie Sie einen Analogsensor an den Pi anschließen, sondern auch, wie Sie mit SPI-Geräten arbeiten, was wegen ihrer Beliebtheit eine tolle Sache ist.

10.4 Wenn es nicht klappt ...

Alle Ratschläge aus Abschnitt 9.7 gelten auch für das Projekt in diesem Kapitel. Zusätzlich müssen Sie sehr vorsichtig sein, wenn Sie den MCP3008 ins Steckbrett einstecken. Passen Sie auf, dass Sie seine Beinchen nicht versehentlich verbiegen. Sie sollten außerdem jede einzelne Verbindung zwischen MCP3008 und dem Pi gründlich überprüfen.

Sehen Sie sich die Pins des PIR-Sensors und des TMP36 genau an. Nicht alle PIR-Sensoren haben dieselbe Pinreihenfolge und die Pins des TMP36 lassen sich leicht miteinander verwechseln.

10.5 Die nächsten Schritte

Sie haben gelernt, wie Sie mit Analog- und Digitalsensoren umgehen und wie Sie SPI-Geräte steuern. Der Pi unterstützt außerdem I²C¹² (Inter-Integrated Circuit), einen anderen beliebten Standard zum Anschluss von Geräten. Es ist sinnvoll, sich auch damit zu beschäftigen.

11) http://www.analog.com/static/imported-files/data_sheets/TMP35_36_37.pdf

12) http://en.wikipedia.org/wiki/I%C2%BA_C

Wenn Sie anspruchsvollere Projekte bauen möchten, sollten Sie für den Pi ein Erweiterungsboard kaufen. Die *Prototyping Pi Plate*¹³ von Adafruit z.B. vereinfacht den Aufbau von Prototypen wesentlich. Das *Gertboard*¹⁴ macht die Erstellung von Prototypen noch sicherer und bietet eine Menge netter Zusatzfunktionen.

13) <http://adafruit.com/products/801>

14) <http://www.raspberrypi.org/archives/411>

A Eine Einführung in Linux

Das beliebteste Betriebssystem für den Pi ist Linux, und hierbei insbesondere die Debian-Distribution namens Raspbian. Wenn Sie bisher nur mit Betriebssystemen wie Windows oder Mac OS X gearbeitet haben, kann Linux für Sie leicht zum Kulturschock werden, vor allem, da die grafische Benutzeroberfläche (GUI) nur optional ist. Das bedeutet, dass Sie ein Linux-System ohne Maus und ohne Doppelklicks auf bunte Icons bedienen können.

Da Sie mit dem System aber irgendwie interagieren müssen, bietet Linux für diesen Zweck eine Shell an. Eine Shell ist ein Programm, das Ihre Befehle via Tastatur entgegennimmt und sie an das Betriebssystem (Linux) weitergibt. Nachdem Linux einen Befehl abgearbeitet hat, gibt Ihnen die Shell das Ergebnis des Befehls zurück.

Die Shell selbst läuft in einem Terminal, das auf die Anfänge der Computertechnik zurückgeht. Damals hatte man über relativ einfache Endgeräte (Terminals) Zugriff auf die »richtigen« Computer. Die Terminals sendeten die Eingaben an den Computer und dessen Ausgaben wurden auf den Terminals angezeigt. Heute müssen Sie nicht mehr mit solch speziellen Geräten arbeiten, aber die Metapher lebt weiter und Linux verwendet sie auch.

Wenn Sie sich also auf einem Linux-Computer anmelden, startet dieser normalerweise eine Shell in einer Terminalsitzung. In der Shell können Sie Befehle eingeben, die dann auf dem Linux-Computer ausgeführt werden.

In unserer modernen Zeit wird Linux mit grafischen Desktop-Systemen geliefert, die Windows und Mac OS X ziemlich ähnlich sind. Aber auch diese Systeme kommen mit einer Terminal-emulation daher, über die Sie Befehle direkt eingeben können. Der LXDE-Desktop verfügt z.B. über

ein Terminal namens LXTerminal. Sie finden sein Symbol auf dem LXDE-Desktop. Klicken Sie doppelt darauf, und der Pi startet eine neue Terminalsitzung.

A.1 Erste Begegnungen

Wenn Sie sich am Pi angemeldet haben oder nachdem Sie eine neue Terminalsitzung vom Desktop aus gestartet haben, sehen Sie die folgende Eingabeaufforderung:

```
pi@raspberrypi ~ $
```

Das sieht nicht nach viel aus, gibt Ihnen aber eine Menge Informationen. Der erste Teil zeigt Ihnen z.B. den Hostnamen Ihres Computers: *raspberrypi*. Sie erfahren auch, dass Ihr Benutzername *pi* lautet. Das ist ein wichtiger Hinweis, denn Linux ist ein Mehrbenutzersystem. Das heißt, dass mehrere Benutzer gleichzeitig mit demselben Computer arbeiten können (zum Beispiel über ein Netzwerk). Sie können auch jederzeit in ein anderes Benutzerkonto wechseln, wenn nötig. Daher ist es gut zu wissen, wer Sie gerade sind.

Der nächste Teil der Eingabeaufforderung enthält den Pfad des Dateisystems, in dem Sie sich gerade befinden. Er besteht hier nur aus dem Tilde-Zeichen (~), das die Abkürzung für das Stammverzeichnis des Benutzers ist. Jeder Linux-Anwender besitzt ein Stammverzeichnis, in dem er persönliche Daten und Konfigurationsdateien speichern kann. Es entspricht dem Ordner *Eigene Dokumente* unter Windows oder *Dokumente* unter Mac OS X. Das Dollarzeichen am Ende markiert das Ende der Eingabeaufforderung.

Um den Inhalt des aktuellen Verzeichnisses zu sehen, geben Sie `ls` ein und drücken Sie *Return*, um den Befehl `ls` auszuführen.

```
pi@raspberrypi ~ $ ls  
Desktop python_games
```

Das aktuelle Verzeichnis enthält zwei Objekte namens `Desktop` und `python_games`. Aus den Namen können Sie nicht ableiten, ob es sich um normale Dateien oder um Verzeichnisse handelt. Erfreulicherweise können Sie das Verhalten der meisten Linux-Befehle mittels Optionen steu-

ern. Diese Optionen bestehen meist aus einem einzelnen Buchstaben nach einem Minuszeichen. Der Befehl `ls` bietet viele Optionen, und wenn Sie ihm die Option `-l` übergeben (Kurzform für »lang«), werden detailliertere Informationen über die Dateien im aktuellen Verzeichnis ausgegeben.

```
pi@raspberrypi ~ $ ls -l
total 8
drwxr-xr-x 2 pi pi 4096 Jul 15 19:36 Desktop
drwxr-xr-x 2 pi pi 4096 Jul 15 19:36 python_games
```

Diese Ausgabe sieht zuerst ein wenig umständlich aus, ist aber schnell zu verstehen. Für jedes Objekt im Verzeichnis zeigt `ls` die in Abbildung A-1 dargestellten Informationen an.

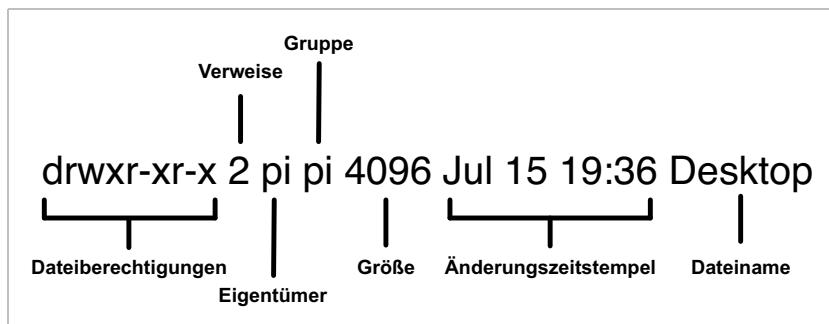


Abb. A-1 Die verschiedenen Komponenten der `ls`-Ausgabe

Der Dateimodus enthält den Dateityp und seine Berechtigungen. Ist das erste Zeichen ein Strich, handelt es sich um eine reguläre Datei. Ist es ein `d`, handelt es sich um ein Verzeichnis. Also sind `Desktop` und `python_games` beide Verzeichnisse.

Die folgenden neun Zeichen geben die Dateiberechtigungen für drei verschiedene Arten von Benutzern an: den Eigentümer, die Gruppe und andere. Die ersten drei Zeichen sind `rwx` und sie bedeuten, dass der Eigentümer der Datei die Berechtigung hat, die Datei zu lesen (`r`), zu schreiben (`w`) und auszuführen (`x`). Im Falle eines Verzeichnisses bedeutet Ausführen, das Verzeichnis »zu betreten«.

Jede Datei auf einem Linux-System gehört zu einem Benutzer und zu einer Gruppe. Gruppen helfen dabei, Teams zusammenzustellen, die miteinander die gleichen Ressourcen nutzen. Für jede Datei speichert Linux also auch Gruppenberechtigungen. In diesem Fall sind sie `r-x`, was bedeutet, dass die Mitglieder der Gruppe die Datei lesen und ausführen, sie aber nicht ändern können.

Schließlich speichert Linux Berechtigungen für Benutzer, die weder der Dateieigentümer sind noch zur Dateigruppe gehören. Auch hier bedeutet `r-x` wieder, dass andere Anwender die Datei lesen und ausführen dürfen, sie aber nicht ändern können.

Die nächste von `ls` ausgegebene Information ist die Anzahl von Verknüpfungen auf eine Datei. Für Ihre erste Tour durch Linux können Sie sie einfach ignorieren.

Danach finden Sie den Namen des Eigentümers und seiner Gruppe. In diesem Fall sind beide `pi`. Das liegt daran, dass es auf dem gezeigten Linux-System sowohl einen Anwender als auch eine Gruppe namens `pi` gibt.

Dann folgt die Dateigröße. Unter Linux sind Verzeichnisse ebenfalls Dateien und sie enthalten einfach die Namen der im Verzeichnis gespeicherten Dateien. Per Voreinstellung belegt Linux im Voraus etwas Speicher für die Dateiliste. Bei Debian auf dem Raspberry Pi sind es 4096 Bytes.

Rechts von der Dateigröße sehen Sie, wann die Datei das letzte Mal verändert wurde. Und schließlich gibt `ls` den Dateinamen aus.

A.2 Durch das Dateisystem navigieren

Der Befehl `pwd` (Print Working Directory) gibt an, in welchem Verzeichnis Sie sich momentan befinden.

```
pi@raspberrypi ~ $ pwd  
/home/pi
```

Wie Sie sehen, erweitert sich Ihr Stammverzeichnis (`~`) zum absoluten Dateipfad `/home/pi`.

Linux unterscheidet zwischen absoluten und relativen Pfaden. Absolute Pfade beginnen mit einem Schrägstrich (`/`) und referenzieren die Datei

unabhängig von ihrer Position im Dateisystem. Relative Pfade beziehen sich auf die Dateiposition ausgehend von Ihrer aktuellen Position im Dateisystem. Das folgende Beispiel zeigt die Unterschiede zwischen absoluten und relativen Pfaden.

Wie Sie im vorhergehenden Abschnitt gesehen haben, enthält das Stammverzeichnis des Anwenders zwei Verzeichnisse namens Desktop und python_games.

```
pi@raspberrypi ~ $ ls  
Desktop python_games
```

Mit dem Befehl cd (Change Directory) wechseln Sie vom aktuellen Verzeichnis in ein anderes.

```
pi@raspberrypi ~ $ cd Desktop/  
pi@raspberrypi ~/Desktop $
```

Jetzt hat sich Ihr Arbeitsverzeichnis geändert. Sie können sehen, dass sich die Eingabeaufforderung verändert hat. Überprüfen Sie das mit dem Befehl pwd.

```
pi@raspberrypi ~/Desktop $ pwd  
/home/pi/Desktop
```

Um zum Stammverzeichnis zurückzugelangen, haben Sie mehrere Möglichkeiten. Sie können beispielsweise cd mit einem absoluten Pfad zum Stammverzeichnis aufrufen.

```
pi@raspberrypi ~/Desktop $ cd /home/pi  
pi@raspberrypi ~ $
```

Alternativ verwenden Sie einen relativen Pfad wie hier:

```
pi@raspberrypi ~/Desktop $ cd ..  
pi@raspberrypi ~ $ pwd  
/home/pi
```

Die Abkürzung .. steht für das Elternverzeichnis des aktuellen Verzeichnisses. Im vorherigen Befehl ist Ihr aktuelles Arbeitsverzeichnis /home/pi/Desktop. Wenn Sie cd.. ausführen, wechseln Sie vom Arbeitsverzeichnis ins Elternverzeichnis von Desktop, das /home/pi ist.

A.3 Textdateien bearbeiten

Viele Linux-Werkzeuge basieren auf Konfigurationsdateien. Die meisten sind reguläre Textdateien, und Sie müssen sie von Zeit zu Zeit bearbeiten. Unter Linux gibt es für das Terminal viele leistungsfähige Texteditoren. Wenn Sie an grafische Texteditoren gewöhnt sind, sehen die meisten Linux-Editoren zuerst etwas eigenartig aus. Einer der einfachsten und intuitivsten Editoren ist *nano*. Er blendet die Tastaturkürzel für die wichtigsten Befehle dauerhaft ein, sodass Sie sie sich nicht merken müssen. Der folgende Befehl startet *nano* und erzeugt eine leere Textdatei namens *hello.txt*.

```
pi@raspberrypi ~ $ nano hello.txt
```

Im folgenden Screenshot sehen Sie, wie *nano* in Ihrem Terminal aussieht.

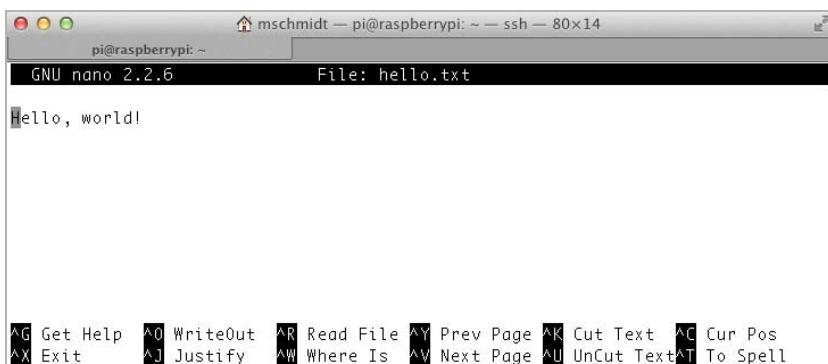


Abb. A-2 Der Texteditor nano

Sie können den Großteil der Bildschirmfläche zur Textbearbeitung verwenden, geben Sie also ein paar Wörter ein und bewegen Sie den Cursor mit den Pfeiltasten herum. Unten auf dem Bildschirm sehen Sie die wichtigsten *nano*-Befehle. Um sie aufzurufen, drücken Sie die Strg-Taste und den zum Befehl gehörenden Buchstaben. (Das ^-Zeichen ist eine Abkürzung für die Strg-Taste.) Zum Beispiel können Sie *nano* verlassen, indem Sie Strg-X drücken.

Wenn Sie dies tun, verwirft *nano* Ihre Änderungen nicht einfach und wird beendet, sondern Sie werden gefragt, ob Sie Ihre Änderungen speichern möchten (siehe Abb. A-3). Geben Sie y ein, wenn Sie Ihre Änderungen

speichern möchten, und wenn nicht, dann n. Wenn Sie y gedrückt haben, sind Sie jedoch noch nicht fertig, denn *nano* bittet Sie noch um die Bestätigung des Dateinamens (siehe Abb. A-4).

Normalerweise drücken Sie einfach Return, um die Datei dann zu speichern. Unten auf dem Bildschirm sehen Sie einige nützliche Optionen, die z.B. das Speichern der Datei in unterschiedlichen Formaten ermöglichen.

Wenn Sie häufiger mit Linux arbeiten, sollten Sie sich zuerst mit einem seiner Texteditoren vertraut machen. Für Anfänger ist *nano* eine exzellente Wahl, probieren Sie es also zumindest ein paar Minuten lang aus.



Abb. A-3 Eine Datei mit dem Texteditor *nano* speichern

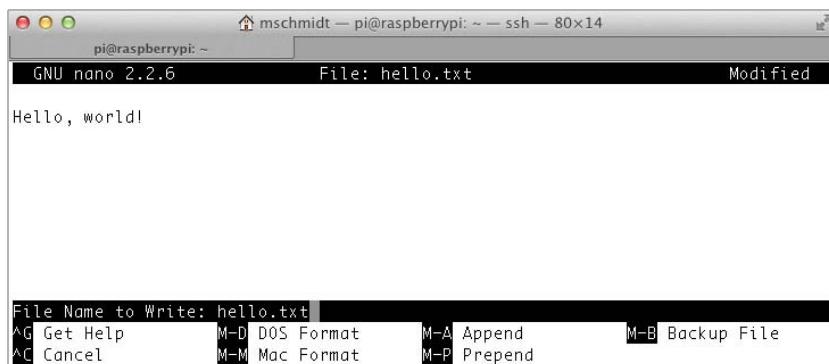


Abb. A-4 *nano* bittet Sie immer um die Bestätigung des Dateinamens.

A.4 Benutzer verwalten

Linux ist ein Mehrbenutzersystem. Es können mehrere Anwender gleichzeitig mit demselben System arbeiten. In diesem Buch sehen Sie am häufigsten den Benutzer *pi*, denn er wird mit Raspbian automatisch mitgeliefert. Das ist zwar bequem, aber manchmal ist es sinnvoll, für verschiedene Aufgaben verschiedene Benutzer einzurichten. Außerdem ist der Benutzer *pi* ein sehr mächtiger Anwender, der volle Administratorberechtigungen hat und beinahe alle Einstellungen des Systems verändern kann. Solche Rechte sollten Sie nicht allen Anwendern einräumen. Besser ist es, Administratorrechte nur dann einzusetzen, wenn Sie auch wirklich für den Zweck benötigt werden. So wird das System nicht versehentlich beschädigt.

In Linux einen neuen Benutzer einzurichten geht einfach mit dem Befehl `adduser`.

```
pi@raspberrypi ~ $ sudo adduser maik
< Adding user 'maik' ...
    Adding new group 'maik' (1002) ...
    Adding new user 'maik' (1002) with group 'maik' ...
    Creating home directory '/home/maik' ...
    Copying files from '/etc/skel' ...
    Enter new UNIX password:
    Retype new UNIX password:
passwd: password updated successfully
Changing the user information for maik
Enter the new value, or press ENTER for the default
→ Full Name []: Maik Schmidt
< Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
→ Is the information correct? [Y/n] Y
```

Sie müssen nur einen Benutzernamen angeben (per Konvention sollte er aus Kleinbuchstaben bestehen), ein Passwort und ein paar optionale Attribute wie Ihren vollen Namen. Wenn Sie bestätigt haben, dass die Informationen richtig sind, erstellt Linux diesen Benutzer und sein eigenes Stammverzeichnis. Wenn Sie den Pi das nächste Mal starten, können Sie sich mit diesem Namen anmelden. Falls Sie ungeduldig sind, können Sie den Befehl `su` (Substitute User Identity) verwenden, um auf einen neuen Benutzer umzuschalten.

```
pi@raspberrypi ~ $ su - maik  
Password:  
maik@raspberrypi ~ $ pwd  
/home/maik  
maik@raspberrypi ~ $ startx
```

Der Befehl `su` fragt Sie nach dem Passwort des Benutzers, und wenn es korrekt ist, wird auf den neuen Benutzer umgeschaltet. Der Befehl `pwd` gibt das aktuelle Arbeitsverzeichnis aus. In diesem Fall ist es das Stammverzeichnis des neu erstellten Benutzers. Wenn Sie den LXDE-Desktop mit dem Befehl `startx` aufrufen, begrüßt er Sie mit dem normalen LXDE-Hintergrundbild (siehe Abb. A-5), denn im Gegensatz zum Benutzer *pi* beginnt der neue Benutzer mit den Standardeinstellungen.

Als wir mit dem Benutzer *pi* arbeiteten, haben wir `sudo` häufig verwendet, um Befehle mit Administratorberechtigungen auszuführen. Schauen wir, was passiert, wenn wir versuchen, eine Datei mit dem Befehl `rm` (Remove) zu löschen, die uns nicht gehört.

```
maik@raspberrypi ~ $ sudo rm /boot/config.txt
```

We trust you have received the usual lecture from the local System Administrator. It usually boils down to these three things:

- #1) Respect the privacy of others.
- #2) Think before you type.
- #3) With great power comes great responsibility.

```
[sudo] password for maik:  
maik is not in the sudoers file. This incident will be reported.
```

Der Befehl gibt eine Warnung aus und fragt dann nach Ihrem Passwort. Offensichtlich ist der Benutzer nicht berechtigt, Dateien im Verzeichnis `/boot` zu löschen, weshalb sich Linux weigert, den Befehl auszuführen.

Obwohl es sinnvoll ist, neuen Benutzern gefährliche Operationen zu verweigern, benötigen sie manchmal Zusatzprivilegien. Wenn Sie Ihren neuen Anwendern dieselben Rechte wie dem Benutzer *pi* geben wollen, müssen Sie sie in die Datei `sudoers` einfügen. Diese Datei enthält eine Liste aller Benutzer, die den Befehl `sudo` verwenden dürfen und sie definiert auch, welche Operationen diese Anwender durchführen dürfen. Sie können die Datei `sudoers` nicht direkt bearbeiten, sondern müssen den Befehl `visudo` verwenden, der standardmäßig den Texteditor `vi` aufruft. Wenn



Abb. A-5 Das Standardaussehen von LXDE

Sie die Datei mit einem anderen Texteditor wie *nano* bearbeiten möchten, müssen Sie ihn in der Befehlszeile angeben (passen Sie auf, dass Sie dabei wieder den Benutzer *pi* verwenden).

```
pi@raspberrypi ~ $ sudo EDITOR=nano visudo
```

Dadurch wird die Datei /etc/sudoers mit dem Texteditor *nano* geöffnet. In der Datei finden Sie einen Abschnitt wie diesen:

```
# User privilege specification
root ALL=(ALL) ALL
suse ALL=(ALL) ALL
pi ALL=(ALL) ALL
```

Fügen Sie eine neue Zeile ein, die genauso aussieht wie die vorigen drei Zeilen, aber ersetzen Sie den Benutzernamen durch den Namen Ihres neuen Benutzers. Wenn Sie *nano* verwenden, um die Datei zu bearbeiten, drücken Sie Strg-X und bestätigen Sie, dass Sie die Änderungen speichern möchten. Nachdem Sie noch den Dateinamen bestätigt haben, sind Sie fertig. Jetzt hat Ihr neuer Benutzer dieselben Rechte wie Benutzer *pi*.

Wenn Sie einen bestimmten Benutzer nicht mehr benötigen, ist es sinnvoll, ihn zu löschen.

```
pi@raspberrypi ~ $ sudo userdel maik
```

Dieser Befehl löscht nur das Benutzerkonto, nicht aber die Dateien des Benutzers. Der Benutzer kann sich nicht mehr am System anmelden, aber alle von ihm im Stammverzeichnis angelegten Dateien bleiben erhalten. Wenn Sie die Dateien ebenfalls löschen möchten, führen Sie folgenden Befehl aus:

```
pi@raspberrypi ~ $ sudo userdel -r maik
```

Sollten Sie jemals die Benutzerattribute ändern müssen, wie z.B. das Stammverzeichnis, können Sie dazu den Befehl usermod verwenden. Sie können ihn z.B. einsetzen, um Konten zu sperren oder zu entsperren.

```
pi@raspberrypi ~ $ sudo usermod -L maik
```

Dadurch wird das Konto des Benutzers *maik* gesperrt. Der Benutzer kann sich dann nicht mehr am System anmelden. Um das Konto zu entsperren, verwenden Sie folgenden Befehl:

```
pi@raspberrypi ~ $ sudo usermod -U maik
```

Sie können die Handbuchseite (man page) von usermod lesen (oder die Handbuchseite jedes anderen Linux-Befehls), indem Sie den Befehl `man` verwenden (siehe auch Abschnitt A.7).

```
pi@raspberrypi ~ $ man usermod
```

Dadurch wird die Handbuchseite des Befehls angezeigt. Sie verlassen das Handbuch, indem Sie `q` eingeben.

Ein wichtiger Vorgang ist die Änderung des Passworts eines Benutzers. Hierzu verwenden Sie den Befehl `passwd`.

```
pi@raspberrypi ~ $ passwd maik
Changing password for maik.
Old Password:
New Password:
Retype New Password:
```

`passwd` fragt Sie nach Ihrem aktuellen Passwort und dann nach dem neuen. Wenn alles in Ordnung ist, wird keine Meldung ausgegeben und der Benutzer besitzt ein neues Passwort.

A.5 Prozesse verwalten

Immer wenn Sie einen Befehl oder eine Anwendung auf einem Linux-System ausführen, erzeugt der Betriebssystemkern einen neuen Prozess. Sie können Ihre aktuellen Prozesse mit dem Befehl `ps` auflisten.

```
pi@raspberrypi ~ $ ps
 PID TTY      TIME      CMD
1880 pts/2    00:00:00  bash
1892 pts/2    00:00:00  ps
```

Momentan haben Sie nur zwei Prozesse. Der erste hat die Prozess-ID (PID) 1880 und gehört zu einem Befehl namens `bash` (die Prozess-ID auf Ihrem System kann abweichen). Das ist der zur Shell gehörende Prozess, in dem Sie gerade arbeiten. Der Prozess mit der PID 1892 gehört zum Befehl `ps`, mit dem Sie die laufenden Prozesse aufgelistet haben. In dem Moment, in dem Sie die Ausgabe des Befehls `ps` sehen, wird Prozess 1892 schon wieder verschwunden sein. Um den Effekt noch einmal zu sehen, rufen Sie `ps` erneut auf.

```
pi@raspberrypi ~ $ ps
 PID TTY      TIME      CMD
1880 pts/2    00:00:00  bash
1894 pts/2    00:00:00  ps
```

Wie Sie sehen können, hat die Shell noch immer die PID 1880, aber Ihr letzter Aufruf von `ps` wurde von einem neuen Prozess mit der Nummer 1894 ausgeführt.

Mehr Informationen über Ihre Prozesse erhalten Sie mit der Option `-f`.

```
pi@raspberrypi ~ $ ps -f
UID        PID  PPID   C STIME  TTY      TIME CMD
pi        1880  1879   0 12:51  pts/2    00:00:00  -bash
pi        1895  1880   0 12:58  pts/2    00:00:00  ps -f
```

Jetzt können Sie die Benutzer-ID (UID) des Anwenders sehen, der einen bestimmten Prozess ausgelöst hat. Es überrascht nicht, dass die UID für alle Prozesse `pi` lautet. Zusätzlich zur PID sehen Sie die ID des Elternprozesses (PPID). Dies ist die ID eines Prozesses, der einen anderen ausgelöst hat. Zum Beispiel hat der Befehl `ps -f`, den Sie gerade aufgerufen haben, die PPID 1880. Das ist die PID der Shell, die Sie momentan benutzen. Die

Shell ist also der Elternprozess des durch den Befehl `ps -f` erstellten Prozesses.

Um alle Informationen über alle momentan auf Ihrem Pi laufenden Prozesse zu sehen, verwenden Sie folgenden Befehl:

```
pi@raspberrypi ~ $ ps -ef
```

Dadurch wird eine recht lange Liste von Prozessen ausgegeben, die jeden einzelnen Linux-Dienst umfasst, der auf Ihrem Pi gestartet wurde.

Eine Liste aller aktiven Prozesse kann durchaus nützlich sein, meist suchen Sie aber aus bestimmten Gründen nach einem speziellen Prozess. Möglicherweise verwendet der Prozess zu viele Ressourcen oder läuft zu lange und Sie möchten ihn beenden. Wie können Sie einen Prozess aber beenden?

Einen lang laufenden Prozess zu beenden ist einfach, wenn Sie ihn direkt von der Shell aus starten. Als Beispiel sucht der folgende Befehl nach allen Textdateien auf Ihrer SD-Karte, er wird also lange benötigen, bis er damit fertig ist.

```
pi@raspberrypi ~ $ find / -name '*.txt'
```

Während der Prozess läuft, können Sie ihn durch Drücken von Strg-C auf Ihrer Tastatur beenden. Wenn Sie Strg-C drücken, erkennt die Shell Ihre Tasteneingabe und sendet dem laufenden Prozess ein Signal. Signale sind kurze Meldungen, auf die alle Prozesse im Hintergrund achten. Das Drücken von Strg-C erzeugt ein Signal namens SIGINT, das dem Prozess mitteilt, dass er unterbrochen wurde. Immer wenn ein Prozess ein SIGINT-Signal erhält, räumt er auf und beendet sich.

Für gerade im Terminal laufende Prozesse ist Strg-C also eine gute Wahl. Was aber, wenn Sie einen Prozess beenden möchten, der im Hintergrund läuft? Die meisten Linux-Dienste laufen standardmäßig im Hintergrund, und Sie starten sie nicht selbst. In diesem Fall müssen Sie die PID des Prozesses herausfinden und sie an den Befehl `kill` übergeben.

```
pi@raspberrypi ~ $ kill 4711
```

Der vorhergehende Befehl hat dem Prozess mit der ID 4711 das Signal SIGTERM gesendet. Sie können mit dem Befehl `kill` auch andere Signale

senden. Zum Beispiel beendet der folgende Befehl mithilfe des SIGKILL-Signals den Prozess mit der PID 4711 in jedem Fall.

```
pi@raspberrypi ~ $ kill -KILL 4711
```

Natürlich müssen Sie die Berechtigung haben, Prozesse zu terminieren. Normalerweise dürfen Sie nur Ihre eigenen Prozesse beenden.

A.6 Den Pi herunterfahren und neu starten

Wenn Sie mit Ihrer Arbeit fertig sind, sollten Sie den Pi nicht einfach ausschalten. Dies kann zu Datenverlust führen. Fahren Sie ihn immer mit dem folgenden Befehl herunter:

```
pi@raspberrypi ~ $ sudo shutdown -h now
```

Wenn Sie den Pi neu starten müssen, machen Sie das mit folgendem Befehl:

```
pi@raspberrypi ~ $ sudo reboot
```

A.7 Wo Sie Hilfe erhalten

Seit seinen Anfängen wurde das Betriebssystem Unix/Linux mit einer tollen Handbuchfunktion namens *man pages* geliefert. Immer wenn Sie eine Option für einen bestimmten Befehl nachschlagen müssen, können Sie seine Handbuchseiten mit dem Befehl `man` anzeigen lassen. Um alle Optionen des Befehls `ls` nachzuschlagen, verwenden Sie z.B. folgenden Befehl:

```
pi@raspberrypi ~ $ man ls
```

Um eine Zeile herunterzurollen, drücken Sie die Pfeil-unten-Taste. Drücken Sie die Pfeil-oben-Taste, um eine Zeile nach oben zu gelangen. Mit der Leertaste erreichen Sie die nächste Seite. Mit Strg-B gelangen Sie zur vorhergehenden Seite. Um das Programm zu beenden, drücken Sie q.

Der Befehl `man` hat viele weitere Optionen. Um mehr darüber zu erfahren, geben Sie Folgendes ein:

```
pi@raspberrypi ~ $ man man
```

Index

A

A/D-Wandler 127
amixer 48, 49
Analog-Digital-Wandler (A/D) 127
Analogsensoren 127
Android 90
Android-Smartphone 90
Apache HTTP 70
apt-file 40
apt-get 35
Arch Linux 14, 41
Arch Linux ARM 15
Arduino i
Arduino-Projekt 108
ARM 14
ARM1176JZ-F-Prozessor 2
ARM-Prozessor 13
Atari VCS 2600 93, 98
Audiogerät 86
Audiohardware 48
Audiosystem
 konfigurieren 48
 testen 48
AWXi-Schnittstelle 89

B

Bauteile 120
BCM2835 2
Befehl ls 138
Beneath a Steel Sky 97
Benutzer verwalten 144
Benutzernamen 144
Beschleunigungssensoren 119
Betriebssystem ii, 13
Bewegungserkennung 121
Bewegungssensor 121, 125

Bildschirm 10

BIOS 6
Bluetooth 6
Bodhi Linux 14
Bring Your Own Peripherals 7
Broadcom 2
BYOP 7

C

C64-Emulator Vice 101
change_timezone 30
Chromium 58
Codebeispiele v
Codecs 87
Commodore 64 98
Components iv
Composite Video 3
CSI-Anschluss 4

D

Dateiberechtigungen 139
Dateisystem 140
Datum einstellen 30
Day of the Tentacle 93, 96
Debian Linux ii, iii, iv
Debian Wheezy 14
Debugger 100
Desktop starten 33
Desktop-Umgebung 34
Diode 107
DSI-Anschluss 4

E

Echtzeituhr 6, 30
Ein- und Ausgabepin 105
Ein-/Ausgabekontakte 4
Einchipsystem 2

- Einführung in Linux iii
Einrückungen 124
Elektronikgeräte 103, 117
Emulator 98
Emulieren 97
Ethernet-Schnittstelle 3
expand_rootfs 27
- F**
Farnell iv
FastCGI 72
FAT-Dateisystem 44
Fedora 41
Firmware 43
aktualisieren 43
Firmware-Konfigurationsparameter 49
Flash 58
Flight of the Amazon Queen 97
Frotz 95
FTP 87
- G**
Gehäuse 10
GitHub 44
Google Chrome OS 14
Google Mail 58
GPIO 4, 105, 109
gpio 108
GPIO-Pin iii, 105
GPIO-Ports 69
GPIO-Status 116
GPU 2
Grafik iii
- H**
H.264-Video-Codec 87
Handbuch ii
Händler iv
Hardware ii, 1
HDMI 3
Hilfe 150
HTTP-Server 70
Hypertext-Transfer-Protokoll 70
- I**
Image einer Linux-Distribution 14
Indiecity 41
Infrarotsender 88
- Infrarotsensoren 121
passive 121
install 37
Installation anpassen 27
Installationsvorgang 13
Interactive Fiction Archive 95
IP-Adresse 79
iPhone 90
Iridium Rising 42
- J**
Java 58
Java-Applets 59
JavaScript deaktivieren 59
Joystick 4
JTAG-Steckplätze 4
Jumperkabel 110
- K**
Kernel aktualisieren 43
Kiosk-Modus 56
Kiosk-System iii, 51
Konfigurationsdatei 45
Konfigurationseinstellungen 26, 46
Konfigurationsparameter iii
- L**
Ladegeräte 7
Ländereinstellung 28, 30
LED 107
Leuchtdiode 108
Lighttpd 70, 71
Linux ii, 13, 137
Linux-Editor 142
Linux-Kernel 43
lsusb 76
LXDE 34
LXDE-Desktop 145
- M**
Maniac Mansion 96
Maus 9
MCP3008 120, 127
Anschlusschema 128
Pinbelegung 132
steuern 132
Media-Player 86
Mehrbenutzersystem 144

- memory_split 31
Micro-USB-Anschluss 3
Midori 58
Modell A 1, 3
Modell B 1
 Ethernet-Schnittstelle 3
Modell-B-Platine 3
Mojang 102
Multimediacenter iii, 81
Musikformate 87
- N**
- nano 46, 71
Netzteil 7
Netzwerk iii, 57
Netzwerk-Equipment 10
NFS 87
Nginx 70
numid 49
- O**
- Onlinematerial v
Overclocking 32
Overscan 45
Overscan-Modus 27
Overscan-Probleme 47
- P**
- Pac-Man 99, 101
pacman 41
Paketmanager 15, 35
Passwort 31, 144
PDF-Reader 36
PHP 72
PHP-Infoseite 73
PHP-Interpreter 72
Pi 57
 herunterfahren 150
 Hilfe 150
 neu starten 150
 Oberfläche iii
Pi Store 41
Pi-Firmware iii
PIR-Schaltkreis 123
PIR-Sensor 120, 122, 126
PIR-Sensor steuern 123
Point-and-Click-Adventures 93, 95
pragprog 56
- Projekt-Wiki 8
Prozesse verwalten 148
PSPC 65
Public-Key-Verschlüsselung 63
PuTTY 61
PuTTYgen 64
Python 123
Python-Klasse 124
- Q**
- Quake II 101
Quake III 93
- R**
- Raspberry Pi i
Raspberry-Projekt 14
Raspberry-Stiftung i
Raspbian ii, 14, 137
 konfigurieren 25
 starten 25
Raspbmc 81, 84
Raspbmc-Installer 82
Raspbmc-Karte 82
Raspi-config 26, 27
RISC OS 14, 16
ROM 6
Root-Berechtigung 20
RPi-Bibliothek 124, 125
rpi-update 44
- S**
- Samba 87
Schaltkreis 106
Schlüssel
 öffentlich 63
Schlüsselpaar 63
Script Creation Utility for Maniac Mansion 96
SCUMM 96
ScummVM 96
SD-Karte 5, 8
 bootfähige 16
 Speicher 27
 unter Linux 19
 unter Mac OS X 20
 unter Windows 17
SD-Kartenimage 18
secure Shell 60

- Sensoren 119
Serial Peripheral Interface (SPI) 128
SFTP 87
Shell 137
Smartphone 88
Snake 101
Software
 aktualisieren 39
 entfernen 38
 installieren 36
Speicher 5
Speicheralarm 112
Speicherkonfiguration 31
Speicherstatus 114
SPI aktivieren 129
SPI-Bus 129
spidev-Bibliothek 133
Spiele 41, 93, 101
Spielekonsole iii
Spieleplattform 93
SPI-Gerät 132
SPI-Geräte verbinden 129
SSH 57, 87
 Einloggen auf dem Pi 65
ssh 60
SSH-Server 60
Startverzeichnis 44
Statusindikator 111
Status-LED 4
Steckbrett 106, 120, 130
Stella 98
- T**
Tastatur 9
Tastaturbelegung 28
Temperaturen messen 127
Temperatursensor 127
Terminal 137
Terminalemulation 137
Terminalsitzung 137
Tetris 101
Textadventures 93
Textdateien 142
Texteditor nano 46, 71
The Secret of Monkey Island 93, 96
TightVNC 66
- TMP36 120
 Datenblatt 134
TMP36-Schaltkreis 131
Twitter 52
Twitter-Suche-Widget 53
- U**
Ubuntu 13
Underscan 45
USB-Anschlüsse 3
USB-Gerät 7
USB-Hub 9
User-Agent 59
UTC-Darstellung 30
- V**
VCS 2600 98
Velocix 41
VGA-Anschluss 3
Videoausgabe konfigurieren 45
VideoCore IV 2
Videoformate 87
Video-Unterstützung 59
Virtual Network Computing (VNC) 66
VNC-Client 66
VNC-Server 66
- W**
Webbrowser 14, 52
Webseiten automatisch aktualisieren 55
Webserver iii, 69, 72
Widerstand 108
Widget 53
Widget-Code 54
WiFi Config 75
Win32DiskImager 17
WiringPi 108
WLAN 6, 74
wlan0 78
WLAN-Adapter 75, 77
- X**
XBMC iii, 81, 84, 86
 Add Files 86
 Fernbedienung 88
 Music 86
 Videos 86

Y

yum 41

Z

Zeit einstellen 30

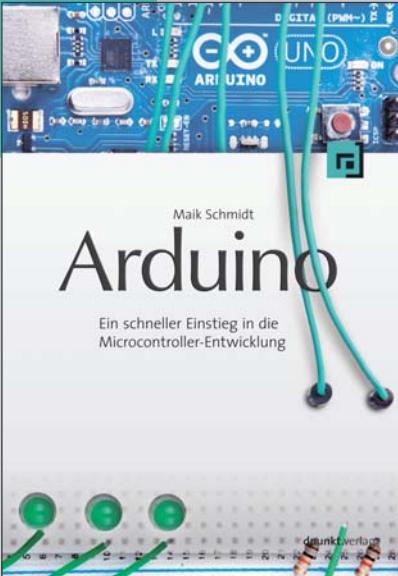
Zeitzone einstellen 30

Z-Language 94

Z-Maschine 95

Zork 93, 94

Zusatzhardware iv



2012, 260 Seiten, Broschur
€ 26,90 (D)
ISBN 978-3-89864-764-9

Maik Schmidt

Arduino

Ein schneller Einstieg in die Microcontroller-Entwicklung

Leser finden rasch einen Einstieg sowohl in die Software als auch in die Hardware des Arduino-Projekts. Bereits nach kurzer Zeit erstellen sie Schaltungen und Code für erste Projekte und erwecken sie zum Leben. Von da an geht es mit anspruchsvolleren Projekten weiter, bei deren Umsetzung die Leser immer mehr dazu lernen. Nach der Lektüre können Sie sich im Arduino-Umfeld orientieren und eigene Ideen sicher umsetzen.

Wenn Sie an Elektronik interessiert sind und besonders daran, Ihre eigenen Ideen umzusetzen, haben Sie das richtige Buch gekauft. Ideal für C/C++ oder Java-Programmierer!



Ringstraße 19 B · 69115 Heidelberg
fon 0 62 21/14 83 40
fax 0 62 21/14 83 99
e-mail hallo@dpunkt.de
<http://www.dpunkt.de>