



Raspberry Pi

programmieren mit **Java**



Hinweis des Verlages zum Urheberrecht und Digitalen Rechtemanagement (DRM)

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

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

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

Wolfgang Höfer

Raspberry Pi programmieren mit Java



mitp

Bibliografische Information der Deutschen Nationalbibliothek

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

ISBN 978-3-95845-056-1

1. Auflage 2016

www.mitp.de

E-Mail: mitp-verlag@sigloch.de

Telefon: +49 7953 / 7189 - 079

Telefax: +49 7953 / 7189 - 082

© 2016 mitp Verlags GmbH & Co. KG

Dieses Werk, einschließlich aller seiner Teile, ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

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

Lektorat: Sabine Schulz

Sprachkorrektur: Petra Heubach-Erdmann

Coverbild: Michael Weigend

Satz: III-satz, Husby, www.drei-satz.de

Inhaltsverzeichnis

	Einleitung	13
I	Der Raspberry Pi	19
I.1	Das Board	19
I.1.1	Der Prozessor	20
I.1.2	Die GPU (Graphics Processing Unit)	20
I.1.3	Audio	21
I.1.4	Netzwerk und USB	22
I.1.5	CSI (Camera Serial Interface)	22
I.1.6	DSI (Display Serial Interface)	22
I.1.7	Status-LEDs	24
I.1.8	Der Schacht für die SD-Karte	24
I.1.9	GPIO (General Purpose Input and Output)	25
I.2	Ein Gehäuse für den Raspberry Pi	27
I.3	Das richtige Netzteil	27
I.4	Tastatur, Maus und HDMI-Kabel	28
I.5	Die SD-Karte	28
2	Booten und konfigurieren	29
2.1	Das Betriebssystem auf die SD-Karte laden	29
2.2	Booten	30
2.2.1	Reboot und richtig ausschalten	36
2.2.2	Mögliche Fehler	38
2.3	Der Raspberry Pi und das Netzwerk	39
2.4	Der Desktop	42
2.5	Den Raspberry Pi headless betreiben	44
2.6	Der Raspberry Pi und das WLAN	52
2.7	Ein Backup	57
3	Linux auf dem Raspberry Pi	59
3.1	Verzeichnisse	60
3.2	Dateien	63
3.3	Suchen	65
3.4	Berechtigungen	67
3.5	Benutzer verwalten	69

3.6	Administrator werden	71
3.7	Dateien und Verzeichnisse packen	72
3.8	Softwarepakete verwalten	73
3.9	Netzwerk	74
3.10	Prozesse	74
3.11	Gemischtes	75
4	Projektvorbereitung	77
4.1	BlueJ installieren	78
4.2	Pi4J installieren	79
4.3	Pi4J in BlueJ einbinden	79
4.4	Das Buchprojekt installieren	82
4.5	Ein Projekt anlegen und Pi4J testen	82
4.6	Ein Überblick über das Buchprojekt	87
4.7	Das Gertboard	89
4.7.1	Das Testprogramm	93
4.7.2	Die Klasse TestLED	96
5	Datum und Zeit	99
5.1	Die Date-Klasse	99
5.1.1	Jede Zeitangabe gehört zu einer Zeitzone und einem Sprachraum	100
5.2	Die Klassen Calendar und GregorianCalendar	100
5.3	Datum und Zeit formatieren und parsen	102
5.4	An welchem Wochentag habe ich Geburtstag?	104
5.5	Wie viele Arbeitstage liegen zwischen Weihnachten und Silvester?	106
5.6	Relais schalten	108
5.7	Akkus zeitgesteuert laden	113
5.8	Zeitsteuerung einer Kamera	117
5.9	Die Navigation in BlueJ	121
6	Digitale Eingänge und Sensoren	123
6.1	Den Raspberry Pi über einen Taster steuern	123
6.2	Externe Prozesse über einen Taster steuern	129
6.3	Der Bewegungsmelder HC-SR501	134
6.4	Licht und Musik über den Bewegungsmelder steuern	135
6.5	Abstandsmessung mit dem Ultraschallsensor HC-SR04	143
6.6	Schmitt-Trigger für analoge Sensoren	147
7	Swing lernen	151
7.1	Ein Fenster erzeugen	151
7.2	Swing-Komponenten	155
7.3	Auf Ereignisse reagieren und Text verarbeiten	157

7.4	Auf Ereignisse in einer Auswahlliste reagieren	160
7.5	Auf Ereignisse in Checkboxes und Radiobuttons reagieren	162
7.6	Auf Ereignisse in einer Liste reagieren	166
7.7	Noch etwas über Ereignisse	169
8	Analoge Eingänge und Sensoren	171
8.1	Den MCP3002 an den Raspberry Pi anschließen	173
8.2	Temperaturmessung mit einem NTC-Widerstand	178
8.2.1	Kennlinie linearisieren	178
8.2.2	Der passende Verstärker	182
8.2.3	Die Schutzschaltung	185
8.3	LDR-Sensoren	188
8.4	Analoge Distanzsensoren	193
9	Messwerte speichern	199
9.1	Auf Dateien und Verzeichnisse zugreifen	199
9.1.1	Dateisystem	199
9.1.2	In eine Datei schreiben	200
9.1.3	Aus einer Datei lesen	202
9.2	Listen, Mengen und assoziative Speicher	203
9.2.1	Mit Listen arbeiten	204
9.2.2	Mit Mengen arbeiten	205
9.2.3	Arbeiten mit assoziativem Speicher	206
9.3	Zeichenketten formatieren	208
9.4	Einen Datenlogger schreiben	211
9.5	Temperatur und Beleuchtungsstärke loggen	227
10	Messwertgrafiken mit Swing	233
10.1	Das Prinzip der Komponentenentwicklung	233
10.2	Die Hilfsklasse ComputeDiagram	235
10.3	Die Swing-Komponente JDiagram	243
10.4	Diagramme in einer Swing-Anwendung anzeigen	256
10.5	Geloggte Messwerte anzeigen	259
11	Threads und reguläre Ausdrücke	263
11.1	Threads implementieren	263
11.1.1	Threads anlegen und starten	264
11.1.2	Threads beenden	266
11.1.3	Daten teilen	267
11.1.4	Auf das Ende warten	272
11.2	Reguläre Ausdrücke	273
11.2.1	Zeichenklassen	275

11.2.2	Vordefinierte Zeichenklassen	276
11.2.3	Häufigkeit	276
12	Füllstandsanzeige für eine Zisterne	279
12.1	Die Schaltung	280
12.2	Der HTTP-Server stellt sich vor	283
12.3	Den Webserver bauen	286
12.3.1	Die Klasse RaspiHttpServer	286
12.3.2	Die Klasse HttpHandlerZisterne	288
12.3.3	Die Klasse Scheduler	294
12.3.4	Die Klasse Sensor	297
12.4	Der Client	298
12.4.1	Die Klasse HttpClient	298
12.4.2	Die Swing-Anwendung	301
13	Eine Heizungssteuerung	309
13.1	Die Schaltung	309
13.2	Das Programmkonzept	312
13.2.1	Die Klasse RaspiHttpServer	313
13.2.2	Die Klasse HttpHandlerHeizung	314
13.2.3	Die Klasse Scheduler	321
13.2.4	Die Klasse Sensor	326
13.3	Der Client	328
13.3.1	Die Klasse HttpClient	328
13.3.2	Die Klasse JTempPanel	332
13.3.3	Die Klasse Gui	336
14	Eine Zeitschaltuhr	347
14.1	Die Schaltung	347
14.2	Das Programmkonzept	347
14.2.1	Die Klasse RaspiHttpServer	350
14.2.2	Die Klasse SchedulerPort	350
14.2.3	Die Klasse HttpHandlerZeitschaltuhr	351
14.3	Der Client	359
14.3.1	Die Klasse HttpClient	362
14.3.2	Die Klasse ZeitschaltuhrProperties	363
14.3.3	Die Klasse GuiUtil	364
14.3.4	Die Klasse Gui	368
14.4	Einen Autostart für den Server einrichten	378
14.5	Hinweise zum Betrieb des Servers	379
14.6	Eine Desktop-Verknüpfung erstellen	380

15	E-Mail-Alarm	381
15.1	Das Package raspi.mail	382
15.1.1	Die Klasse MailService	382
15.1.2	Die Klasse EMail	390
15.2	Das Projekt	391
15.2.1	Die Klasse AlarmProperties	394
15.3	Die Klasse AlarmService	396
15.4	Die Klasse Shutdown	399
15.5	Die Klasse Alarm	400
15.6	Autostart	405
16	Der I²C-Bus	407
16.1	Den I ² C-Bus einrichten	411
16.2	Den I ² C-Bus mit Pi4j steuern	414
16.3	Eine Hilfsklasse für I ² C-Geräte	416
16.4	Der Temperatursensor MCP9808	421
16.5	Die Arbeitsweise des MCP9808	424
16.6	Die Klasse MCP9808	426
16.7	Den Sensor testen	434
16.8	Den Sensor im Komparator-Modus testen	440
17	Die serielle Schnittstelle RS232	445
17.1	Die Schnittstelle konfigurieren	448
17.2	Mit Pi4j die serielle Schnittstelle steuern	449
17.3	Der Ultraschall-Entfernungssensor SRFo2	451
17.4	Die Klasse SRFo2	454
17.5	Entfernungsmessung	461
17.6	Kurze Distanzen	463
17.7	Die Sensoradresse ändern	464
18	Textausgabe mit LCD-Displays	467
18.1	Der Befehlssatz des HD 44780	470
18.1.1	Clear Display	470
18.1.2	Cursor Home	470
18.1.3	Entry Mode Set	470
18.1.4	Display On/Off Control	471
18.1.5	Cursor / Display Shift	471
18.1.6	Function Set	472
18.1.7	RAM Address Set und Data write	472
18.1.8	CG RAM Address Set	474
18.1.9	Und noch zwei Lesebefehle	474

18.2	Die Initialisierung des Displays	474
18.3	Das Programmkonzept	476
18.4	Die Klasse DisplayHD44780	477
18.5	Die Klasse PinDirect	483
18.6	Das Display testen	487
18.6.1	Die Klasse Test8Bit	488
18.6.2	Die Klasse Test4Bit	489
18.6.3	Die Klasse TestDisplay	490
18.7	Das Display über den MCP23017 ansteuern	491
18.7.1	Die Klasse PinMCP23017	493
18.7.2	Die Klasse Test8BitMCP23017	496
18.7.3	Die Klasse Test4BitMCP23017	497
A	Java	499
A.1	Der erste Programmcode	499
A.2	Java-Konzepte	502
A.2.1	Objekte	502
A.2.2	Klassen und Instanzen	503
A.2.3	Vererbung	504
A.2.4	Interface	506
A.2.5	Package	507
A.3	Java-Grundlagen	508
A.3.1	Variable	508
A.3.2	Arrays	511
A.3.3	Operatoren	513
A.3.4	Programmfluss	517
A.4	Java ist objektorientiert	520
A.4.1	Klasse	520
A.4.2	Variablen	521
A.4.3	Methoden	522
A.4.4	Konstruktoren	524
A.4.5	this	525
A.4.6	Zugriffsrechte	525
A.4.7	static und final	526
A.4.8	Anonyme innere Klassen	527
A.4.9	ENUM-Datentypen	530
A.4.10	Interfaces	531
A.4.11	Vererbung	536
A.4.12	Methoden überschreiben	541
A.4.13	Super	544
A.4.14	Überschreiben unterbinden	545
A.4.15	Abstrakte Klassen und Methoden	545

A.5	Wrapper-Klassen für numerische Datentypen	546
A.6	Zeichenkette (String)	549
A.7	Character	552
A.8	Boolean	553
A.9	Exceptions	554
A.10	Garbage-Collector	560
A.11	Packages	561
	Stichwortverzeichnis	565

Für Tina, Marcel und Jan



Einleitung

Der Raspberry Pi ist ein Minirechner, den Sie sehr flexibel für die unterschiedlichsten Aufgaben einsetzen können. Zum einen haben Sie mit dem gerade mal scheckkartengroßen Rechner einen vollwertigen Desktop-PC, auf dem Sie Office-Anwendungen laufen lassen und Ihre E-Mails bearbeiten können. Auch Drucken, Surfen, Netzwerkverbindung über Kabel, Wireless LAN und Bluetooth gehören dazu. Zum anderen können Sie an dem Minirechner eine Maus und Tastatur über die USB-Ports und einen Bildschirm über HDMI oder einen Fernseher über Composite anschließen. Oder Sie betreiben den Raspberry Pi über einen Remote-Zugriff. Für diese Aufgaben unterstützt Sie dabei das für den Raspberry Pi optimierte Debian-Linux. Aber keine Angst, auch der ungeübte Anwender, der mit der Konfiguration eines Linux-Rechners nicht vertraut ist, wird so weit unterstützt, dass das Debian-Linux ohne Vorkenntnisse installiert und konfiguriert werden kann.

Die zweite Welt, in die Sie mit dem Raspberry Pi eintauchen können, ist die faszinierende Verbindung von Software und Hardware. Diese reicht von einfachen digitalen Ein- und Ausgängen bis hin zur Ansteuerung von komplexen Schaltkreisen, die über den I²C-Bus, den SPI-Bus oder die RS232-Schnittstelle angesteuert werden können. Über diese Schnittstellen lassen sich dann die unterschiedlichsten Sensoren ansprechen. Den Zugang zu diesen Systemen stellt der Raspberry Pi über eine 26-polige oder je nach Modell auch über eine 40-polige Stiftleiste zur Verfügung. Und genau dahin möchte ich Sie mit diesem Buch begleiten.

Alle Projekte aus diesem Buch werden mit der Programmiersprache Java umgesetzt und es soll nicht nur den erfahrenen Java-Entwickler, sondern auch Umsteiger, die vielleicht bisher mit C oder Python Erfahrung haben, ansprechen. Aber auch alle Technikfreaks, die einfach neugierig sind und etwas Neues ausprobieren wollen. Für diesen Leserkreis gibt es im Anhang dieses Buches eine Einführung in Java, in der die grundlegenden Strukturen und Prinzipien dieser Hochsprache erläutert werden. In den Projekten finden Sie dann weiterführende Themen zu Java.

Wie Sie mit diesem Buch arbeiten können

Das Buch ist so konzipiert, dass die Kapitel aufeinander aufbauen. So gibt es zum Beispiel Java-Klassen, die einmal ausführlich erläutert und dann in nachfolgenden Kapiteln eingesetzt werden. Der gesamte Programmcode teilt sich dabei in einen allgemeinen und in einen für das jeweilige Kapitel spezifischen Teil. Den allgemeinen Teil des Programmcodes können Sie dann später sehr einfach in eigene Projekte einbinden und er bildet in seiner Summe ein kleines Framework für die Java-Programmierung auf dem Raspberry Pi.

Den ganzen Programmcode, den Sie für die Umsetzung der Projekte benötigen, können Sie sich von dem Verlagsserver unter www.mitp.de/055 herunterladen (unter dem Reiter Downloads) und auf dem Raspberry Pi installieren. Eine Anleitung dazu finden Sie in dem Kapitel »Projektvorbereitung«.

Das Buchprojekt mit dem vollständigen Programmcode finden Sie auf dem Verlagsserver

http://media.mitp.de/downloads/9783958450554_Raspberry_Pi_und_Java.tar

Alle Programme sind direkt auf dem Raspberry Pi entwickelt worden. Hierzu wurde die Entwicklungsumgebung BlueJ verwendet, die auch in diesem Buch vorgestellt wird und mit der Sie die wichtigsten Grundlagen für den Umgang an die Hand bekommen.

Da in diesem Buch die Verbindung zwischen Soft- und Hardware beschrieben wird, sind die Projekte fast ausschließlich technischer Natur. Zu jedem Programmcode gehört demnach auch immer ein Stück Elektronik. Hierbei habe ich stets versucht, den Aufwand für Sie möglichst gering zu halten. Es lassen sich daher alle Versuche sehr einfach auf einem Steckbrett nachbauen.

Für einen ersten Überblick über die Themen finden Sie im Folgenden eine kurze Beschreibung zu jedem Kapitel.

Kapitel 1: Der Raspberry Pi

In dem ersten Kapitel lernen Sie die Hardware des Raspberry Pi kennen, indem die verschiedenen Komponenten und Schnittstellen auf dem Board beschrieben werden. Hierzu zählen der Prozessor, die Grafikeinheit, die USB-Anschlüsse, das Netzwerk und die Audio- und Video-Ausgabe. Hierbei wird auch auf die Unterschiede zwischen den verschiedenen Modellen eingegangen. Außerdem wird beschrieben, wo Sie das Kameramodul und ein Display anschließen können.

Für den ersten Start des Raspberry Pi benötigen Sie eine SD-Karte, eine Tastatur, eine Maus und ein HDMI- oder Composite-Kabel und ein Netzteil. Damit Sie hier die richtigen Komponenten anschließen, zeige ich Ihnen, auf welche technischen Werte Sie achten müssen.

Kapitel 2: Booten und konfigurieren

Nachdem alle Komponenten für einen ersten Testlauf des Raspberry Pi verfügbar sind, wird beschrieben, wie Sie das richtige Betriebssystem für den Raspberry Pi auf eine SD-Karte kopieren können und wie Sie den Raspberry Pi konfigurieren. Anschließend werden noch verschiedene Möglichkeiten vorgestellt, um den Raspberry Pi in Ihrem Heimnetzwerk über LAN oder WLAN zu integrieren. Danach wird der Mini-Rechner so konfiguriert, dass Sie über SSH mit PuTTY und WinSCP auf den Raspberry Pi zugreifen können. Weiterhin beschreibe ich, wie Sie mit TightVNC eine Remote-Verbindung aufbauen, um so von einem anderen Rechner auf den grafischen Desktop des Raspberry Pi zuzugreifen.

Kapitel 3: Linux auf dem Raspberry Pi

Auch wenn der Raspberry Pi über einen grafischen Desktop verfügt, werden Sie für bestimmte Aufgaben mit der Linux-Konsole arbeiten müssen. Um hier die wichtigsten Befehle kennenzulernen, finden Sie diese in diesem Kapitel nebst ihrer Anwendung.

Kapitel 4: Projektvorbereitung

In der Projektvorbereitung wird die Java-Entwicklungsumgebung BlueJ installiert und ich beschreibe die wichtigsten Funktionen. Sie installieren dann die Pi4J-Bibliothek, die Sie benötigen, damit Sie die GPIO-Ports des Raspberry Pi ansprechen können. Dies ist die Schnittstelle für alle technischen Projekte.

Wenn Sie nicht direkt mit einem Steckbrett arbeiten wollen, um die ersten Versuche aufzubauen, zeige ich Ihnen, wie Sie das Erweiterungsboard Gertboard mit dem Raspberry Pi verbinden und mit einem einfachen Java-Programm testen können. Auf diesem Board finden Sie einige Schaltkreise, die auch in den verschiedenen Projekten benutzt werden.

Kapitel 5: Datum und Zeit

Im Kapitel »Datum und Zeit« werden Ihnen der Umgang mit Datum und Zeit in Java und die dafür bereitgestellten Klassen `Date`, `Calendar` und `GregorianCalendar` vorgestellt. Hierzu gehört auch das Formatieren und Parsen von Datums- und Zeitangaben. Als praktische Übung schreiben Sie zuerst ein Programm, das den Wochentag Ihres Geburtstags für ein beliebiges Jahr berechnen kann. Und danach kommt die Berechnung der Arbeitstage für ein beliebiges Jahr, die zwischen Weihnachten und Silvester liegen.

Als Nächstes werden Sie das GPIO-Interface des Raspberry Pi über Java ansprechen und Relais ansteuern. Als praktische Übung werden hierzu mit zeitgesteuerten Relais Akkus geladen und eine Spiegelreflexkamera gesteuert.

Kapitel 6: Digitale Eingänge und Sensoren

Das Thema »Digitale Eingänge und Sensoren« zeigt Ihnen, wie Sie mithilfe der Pi4J-Bibliothek auf externe Ereignisse reagieren können. Ein Programm muss also einen Pegelwechsel, der an einem der GPIO-Pins ausgeführt wird, erfassen und darauf reagieren. Als erstes Beispiel werden Sie hierzu ein Programm schreiben, das einen Tastendruck auswertet und daraufhin einen externen Prozess startet. Unter einem externen Prozess versteht man hier zum Beispiel das Ausführen eines Linux-Befehls oder eines beliebigen Programms. In diesem Beispiel wird durch den Tastendruck der Raspberry Pi heruntergefahren und der TightVNC-Server gestartet oder beendet.

Als zweites Anwendungsbeispiel wird ein Bewegungsmelder eingesetzt, um in einer Toilette das Licht beim Betreten einzuschalten und Musik zu spielen. Sie lernen also auch, wie man mit Java Audio-Dateien abspielt.

Den Ultraschallsensor HC-SR04, den man zur Abstandsmessung einsetzt, werden Sie mit einem weiteren Programm testen. Hierbei muss der Raspberry Pi zwei sehr schnell hintereinander auftretende Signale auswerten.

Als Letztes werden dann analoge Sensoren über einen Schmitt-Trigger an den Raspberry Pi angeschlossen, um somit als Schwellenwerterkennung eingesetzt zu werden.

Kapitel 7: Swing lernen

Das Kapitel »Swing lernen« ist ein reiner Java-Kurs, den Sie auch auf jedem anderen Computer testen können. Hier lernen Sie, wie mit Java grafische Oberflächen erzeugt werden und auf Ereignisse reagiert werden kann. Da die Entwicklung grafischer Oberflächen ein sehr komplexes Thema ist, wird hier nur auf einen kleinen Bereich eingegangen, der aber

alles abdeckt, was Sie für die Projekte in diesem Buch wissen müssen. Es dient Ihnen aber auch als Grundlage, um tiefer in dieses Thema einzusteigen.

Kapitel 8: Analoge Eingänge und Sensoren

Um mit dem Raspberry Pi analoge Signale verarbeiten zu können, benötigen Sie einen externen Analog-Digital-Wandler. Im Falle des Raspberry Pi können Sie Wandler verwenden, die entweder über einen SPI- oder einen I²C-Bus verfügen. Als Einstieg in dieses Thema wird der Wandler MCP3002 vorgestellt, der auch auf dem Gertboard zu Verfügung steht und über den SPI-Bus angesteuert wird. Hierzu beschreibe ich zuerst die grundlegende Arbeitsweise des SPI-Bus und wie Sie diesen mit Java über die Pi4J-Bibliothek ansprechen können.

Als analoge Sensoren werden hier ein NTC-Widerstand zur Temperaturmessung und ein LDR-Widerstand zur Messung der Lichtstärke eingesetzt. Außerdem zeige ich Ihnen, wie Sie die Kennlinie solcher Sensoren linearisieren und entsprechende Verstärker zur Anpassung an den Wandler berechnen können.

Da es auch vorkommen kann, dass in einem Datenblatt zu einem Sensor keine Kennlinie, sondern eine Tabelle mit Stützwerten angegeben wird, zeige ich Ihnen, wie Sie diese Stützwerte in Java verarbeiten und Zwischenwerte interpolieren können. Als Beispiel verwenden Sie hierzu ein Programm für die Distanzsensoren der Firma Sharp.

Kapitel 9: Messwerte speichern

Immer wenn Sie mit Sensoren arbeiten, werden Sie die unterschiedlichsten Messwerte speichern und aus einer Datei auslesen müssen. Hierzu habe ich einen Java-Kurs geschrieben, der das Arbeiten mit Dateien und Verzeichnissen erläutert. Außerdem lernen Sie die wichtigsten Java-Klassen kennen, um Daten als Liste, Menge oder in einem assoziativen Speicher abzulegen und formatiert zu speichern. Als praktisches Beispiel werden Sie hierzu einen Datenlogger schreiben, der beliebige Messwerte verarbeiten kann.

Kapitel 10: Messwertgrafiken mit Swing

In diesem Kapitel zeige ich Ihnen, wie Sie mit Java eigene grafische Komponenten entwickeln können. Diese können Sie dann genauso in eigene Programme einsetzen, wie Sie es mit den Swing-Komponenten aus der Java-Bibliothek gewohnt sind. Als Beispiel werden Sie hier ein Diagramm programmieren, mit dem Sie Messwerte im zeitlichen Verlauf grafisch darstellen.

Kapitel 11: Threads und reguläre Ausdrücke

Mit Threads werden in Java nebenläufige Prozesse abgebildet, um eine quasi parallele Verarbeitung von Daten in einer Anwendung zu realisieren. Ich zeige Ihnen in diesem Einführungskurs, wie Sie Threads starten und beenden. Außerdem lernen Sie, wie man auf Daten zugreift, die von mehreren Threads gelesen und geschrieben werden.

Der zweite Block in diesem Kapitel beschäftigt sich mit regulären Ausdrücken, mit deren Hilfe Sie ein sehr mächtiges Werkzeug an die Hand bekommen, um auch komplexe Suchen von Zeichenketten zu realisieren.

Beide Themen aus diesem Kapitel bilden die Grundlage für die nächsten Projekte.

Kapitel 12: Füllstandsanzeige für eine Zisterne

Mit einem Distanzsensor können Sie den Wasserstand in einer Zisterne ermitteln und zum Beispiel als Funktion über die Zeit grafisch darstellen. Die Bereitstellung der Messwerte und die Anzeige werden hier in zwei getrennte Java-Anwendungen aufgeteilt. Die Messwerte werden über einen HTTP-Server zur Verfügung gestellt, den Sie mit ganz einfachen Mitteln selber schreiben. Die Darstellung und das Abholen der Daten von dem Server übernimmt hier ein HTTP-Client, der sich die Daten über einen HTTP-GET-Befehl erfragt. Die grafische Anzeige wird durch die selbst geschriebene Swing-Komponente realisiert.

Mit diesem Konzept kann der HTTP-Client auf jedem Rechner in Ihrem Heimnetzwerk laufen und sich die Daten von dem HTTP-Server, der auf dem Raspberry Pi gestartet wurde, abholen.

Kapitel 13: Eine Heizungssteuerung

Die Heizungssteuerung, die in diesem Kapitel beschrieben wird, folgt dem gleichen Softwarekonzept wie im vorherigen Kapitel, nur dass Server und Client um den HTTP-POST-Befehl erweitert werden und Daten vom Client auf den Server geschrieben werden.

Mit der vorgestellten Schaltung können Sie eine Heizung oder einen Heizkörper steuern.

Kapitel 14: Eine Zeitschaltuhr

Und auch die Zeitschaltuhr teilt sich in Server und Client auf und kann beliebig viele Schaltvorgänge verarbeiten. Sie können damit also Relais für jeden Wochentag beliebig oft ein- und ausschalten. Außerdem zeige ich Ihnen, wie Sie die Anwendung in den Autostart aufnehmen und Desktopverknüpfungen erstellen.

Kapitel 15: E-Mail-Alarm

Im Kapitel »E-Mail-Alarm« lernen Sie die grundlegenden Methoden, um mit Java E-Mails zu verarbeiten. Die E-Mails werden hierbei über einen Provider verschickt und empfangen. Als Anwendung werden Sie eine kleine Alarmanlage programmieren, die einen Alarm durch einen Bewegungsmelder auslöst und eine E-Mail verschickt. Außerdem lässt sich die Alarmanlage über eine E-Mail einschalten.

Kapitel 16: Der I²C-Bus

Der I²C-Bus ist neben dem SPI-Bus der zweite Bus, den der Raspberry Pi auf seinem GPIO-Interface bereitstellt. Wie Sie diesen Bus aktivieren und über die Pi4J-Bibliothek ansprechen, lernen Sie in diesem Kapitel. Außerdem wird das Protokoll des I²C-Bus eingehend behandelt, damit Sie das grundlegende Verständnis für die Arbeit mit diesem Bus erlangen. Als Beispiel werden Sie den Temperatursensor MCP9808 an den I²C-Bus anschließen und den Sensor in verschiedenen Betriebsmodi ansprechen.

Kapitel 17: Die serielle Schnittstelle RS232

Auf dem GPIO-Interface des Raspberry Pi liegt auch ein UART-Baustein bereit, mit dem Sie eine RS232-Schnittstelle aufbauen können und damit die Möglichkeit haben, andere Geräte über eine lange Leitung ansprechen zu können. In einer einfachen Anwendung wird hier der Ultraschallsensor SRF02 über die RS232-Schnittstelle mit dem Raspberry Pi

verbunden und über die Pi4J-Bibliothek angesprochen. Außerdem lernen Sie die Leitungstreiber kennen, die Sie für die Anpassung an lange Leitungen benötigen.

Kapitel 18: Textausgabe mit LCD-Displays

Für viele kleine Anwendungen ist es oft hilfreich, wenn Sie die Möglichkeit haben, kurze Texte über ein einfaches Display anzeigen zu können. Hierdurch können Sie oft auf ein teures TFT-Display verzichten. Die Grundlage für dieses Kapitel ist der HD-44780-Displaytreiber, der in sehr vielen Displays verbaut wird. Sie werden hier den Befehlssatz des HD 44780 kennenlernen und verschiedene Testprogramme schreiben. Dabei werden Sie ein Display im 4-Bit- und 8-Bit-Modus ansteuern. Und Sie werden den IO-Expander MCP23017 einsetzen, um ein einfaches Display über den I²C-Bus anzusteuern.

Anhang A: Java

Wenn Sie mit Java noch keine Erfahrung haben, sollten Sie sich im Anhang den Grundkurs Java anschauen. Dort habe ich die wichtigsten Grundlagen in einem Kurs zusammengefasst. Sie lernen als Erstes das Grundkonzept von Java und der objektorientierten Programmierung kennen. Anschließend erörtere ich, wie in Java mit Variablen, Arrays und Operatoren gearbeitet wird und wie Sie den Programmfluss kontrollieren können. Danach erfolgt eine Einführung in das objektorientierte Programmieren mit Java und die dafür notwendige Syntax. Abschließend bekommen Sie einen Überblick über Wrapper-Klassen, die Fehlerbehandlung in Java und die Organisation von Klassen in sogenannte Packages. Tiefer gehende Themen zu Java werden dann noch in verschiedenen Kapiteln behandelt, die einen konkreten Bezug zu den beschriebenen Projekten haben.

Über den Autor

Nach dem Studium der Elektrotechnik arbeitet Wolfgang Höfer seit über 20 Jahren als Software-Entwickler. Im Laufe seines Berufslebens hat er Projekte in den Programmiersprachen C, Pascal, Basic, Assembler, PL/SQL und Java umgesetzt. Hierbei hat er in den letzten 15 Jahren mit Java hauptsächlich Datenbank-gestützte Client/Server- und Webanwendungen entwickelt. Aus Interesse zur Hardware hat er sich in seiner Freizeit mit der Programmierung von ATMEL-Controllern beschäftigt und damit verschiedene Projekte umgesetzt. Dabei ging es hauptsächlich um die Auswertung von Sensoren und einfachen regelungstechnischen Aufgaben aus der Hausautomation. Bei diesen Projekten fehlte aber immer wieder die Anbindung eines Controllers an einen leistungsstarken Rechner, mit dem Benutzeroberflächen, Webanwendungen, Webservices und die Speicherung von größeren Datenmengen realisiert werden konnten. Da ein normaler PC wegen des relativ hohen Stromverbrauches für den Dauereinsatz ausschied, suchte er nach preiswerten Alternativen. Und genau diese hat er dann mit dem Raspberry Pi gefunden. Die leichte Integration in ein Computernetzwerk und die Fähigkeit, verschiedene Bussysteme und Schnittstellen ansteuern zu können, machen den Raspberry Pi zu einem perfekten System, um Steuer- und Sensorelektronik mit modernen Anwendungen zu verbinden.

Der Raspberry Pi

In diesem Kapitel werden Sie die verschiedenen Komponenten des Raspberry Pi kennenlernen. Ich werde Ihnen die Anschlüsse und Interfaces vorstellen, die den Raspberry Pi mit der Außenwelt verbinden. Außerdem werde ich Ihnen kurz die zwei wichtigsten ICs auf dem Board und deren Funktion erläutern.

1.1 Das Board

Wenn Sie noch keinen Raspberry Pi besitzen, stellt sich zum einen die Frage, wo Sie ihn erwerben können und ob Sie etwas beim Kauf beachten müssen. Da sich der Raspberry Pi mittlerweile so großer Beliebtheit erfreut, ist er in fast jedem gut sortierten Elektronik-Versandhandel zu bekommen. Wenn Sie nach dem Raspberry Pi suchen, werden Ihnen wahrscheinlich unterschiedliche Modelle angezeigt, die sich erst mal im Preis unterscheiden. Die Modelle A und A+ sind mittlerweile schon ein bisschen aus der Mode gekommen, da sie über keinen Netzwerkanschluss verfügen und sich preislich kaum von den größeren Modellen mit Netzwerk unterscheiden. Oft wird das A-Modell auch nicht mehr zum Verkauf angeboten.

Der wesentliche Unterschied zwischen den Modellen B und B+ liegt an der Anzahl der GPIO-Ports. Was sich genau hinter den GPIO-Ports verbirgt, werde ich Ihnen weiter hinten noch aufzeigen. Im Gegensatz zu den A-Modellen bieten die B-Modelle doppelt so viel RAM-Speicher.

Das Flaggschiff der Raspberry-Pi-Modelle ist das Modell 2 B, das seit 2015 auf dem Markt ist. Es bietet den meisten Arbeitsspeicher und hat einen erheblichen Geschwindigkeitsvorteil gegenüber den kleineren Modellen, ist aber auch das teuerste Modell. Wenn man die Modelle mit dem 40-poligen Pfostenstecker miteinander vergleicht, so liegt der ungefähre Preis bei 23 Euro für das A+-Modell, 28 Euro für das B+-Modell und 38 Euro für das Modell 2 B. Für welches Modell man sich nun entscheidet, hängt auch von der Verwendung ab. Alle Beispiele, die in diesem Buch vorgestellt werden, laufen auch mit den B-Modellen. Die A-Modelle fallen bei einigen Anwendungen wegen des fehlenden Netzwerkanschlusses raus.

Modell	RAM-Speicher	USB	Ethernet	Pfostenstecker
A	256MB	1x	-	26
A+	256MB	1x	-	40
B	512MB	2x	10/100 Mbit	26

Tabelle 1.1: Die Modelle im Vergleich

Modell	RAM-Speicher	USB	Ethernet	Pfostenstecker
B+	512MB	4x	10/100 Mbit	40
2 B	1024MB	4x	10/100 Mbit	40

Tabelle 1.1: Die Modelle im Vergleich (Forts.)

Die B-Modelle unterstützen an ihren USB-Anschlüssen die meisten USB-2.0-Geräte. Wenn Ihnen die Anzahl der USB-Anschlüsse nicht ausreicht, können Sie über einen USB-Hub mit eigener Spannungsversorgung weitere hinzufügen. Wollen Sie zum Beispiel eine Tastatur, eine Maus und einen WLAN-Stick an den Raspberry Pi anschließen, kommen Sie bei den Modellen A, A+ und B nicht ohne einen USB-Hub aus.

1.1.1 Der Prozessor

Das Herzstück des Raspberry Pi ist ein sogenannter SoC. Die Abkürzung SoC steht für System-on-a-Chip und bedeutet so viel wie, dass in einem einzelnen Chip die wichtigsten Komponenten für ein Rechnersystem untergebracht sind. Das Konzept, mehrere Hardware-Komponenten in einem IC zu integrieren, wird in fast allen mobilen Geräten genutzt. Der Grund dafür ist natürlich die platzsparende Bauweise. Im Falle des Raspberry Pi handelt es sich außer bei dem Modell 2 B um den SoC BCM2835 der Firma Broadcom. In ihm ist die 32-Bit-CPU ARM1176JZFS untergebracht. Sie wird mit 700 Mhz getaktet und unterstützt die Verarbeitung von Gleitkommazahlen.

Das Modell 2 B hat den SoC BCM2836, in dem sich eine ARM-Cortex-A7-CPU befindet. Sie besitzt vier Kerne und wird mit 900 Mhz getaktet. Damit ist sie über sieben Mal schneller als die der einfachen B-Modelle.

1.1.2 Die GPU (Graphics Processing Unit)

In dem BCM2835 und BCM2836 befindet sich auch der Grafikprozessor Videocore 4 von Broadcom. Er ist in der Lage, BluRay-Qualität mit H.264 bei 40MBits/s zu verarbeiten. Um 3D-Grafiken schneller verarbeiten zu können, greift die GPU auf die mitgelieferte OpenGL-ES2.0-Bibliothek zu. Die Verarbeitung von Vektor- und Rastergrafiken wird durch die OpenVG-Bibliothek realisiert. Die hinter diesen Bibliotheken stehenden Spezifikationen wurden hauptsächlich für eingebettete Systeme wie Mobiltelefone und Spielkonsolen entwickelt.

Damit Sie sich vorstellen können, wie leistungsfähig die GPU in Bezug auf die 3D-Verarbeitung ist, möchte ich Ihnen ein paar Eckdaten vorstellen. Für eine technische Bewertung des Grafikprozessors dienen die Werte GFLOPs/s, Gpixel/s und Gtexel/s. GFLOPs/s sind die maximale Anzahl von Fließkomma-Operationen in Giga pro Sekunde. Gpixel/s und Gtexel/s geben die maximale Anzahl zu verarbeitenden Pixel und Texturelemente in Giga pro Sekunde an. Da hier die reinen Zahlenwerte nicht wirklich ein Gefühl dafür geben, was der Grafikprozessor zu leisten vermag, habe ich die Videocore-4-GPU mit zwei Grafikkarten, wie sie in einem PC verwendet werden, verglichen und in der folgenden Tabelle gegenübergestellt.

Model	Rechenleistung in GFLOPS/s	Pixelfüllrate in Gpixel/s	Texelfüllrate in Gtexel/s
Raspberry Pi mit der Videocore 4	24	1	1,5
Geforce 8200 MG	19,2	1,6	3,2
Geforce 9800 MGTS	288	9,6	16

Tabelle 1.2: Die GPU des Raspberry Pi im Vergleich mit zwei anderen Grafikkarten

Den Video-Ausgang der GPU können Sie entweder über den HDMI-Anschluss mit einem Monitor oder wahlweise über den Composite-Ausgang für einen TV-Scart-Anschluss mit einem Fernseher verbinden. Wenn ich mal davon ausgehe, dass es in fast jedem Haushalt einen Fernseher oder ein Computermonitor gibt, der mindestens eine der beiden Schnittstellen unterstützt, sind dies wohl die einfachsten und auch günstigsten Varianten, um eine Ausgabe für den Raspberry Pi zu realisieren. Wenn Sie den HDMI-Anschluss verwenden, wird der Audio-Ausgang über den Klinkenstecker nicht benötigt, da das HDMI-Interface auch Audio unterstützt.

Die zur Verfügung stehenden Videoausgänge unterscheiden sich etwas bei den verschiedenen Modellen. So hat das Modell B für den Composite-Ausgang eine eigene Buchse. Bei den Modellen B+ und 2 B gibt es eine Buchse für einen Klinkenstecker, auf dem der Composite-Ausgang und der Audio-Ausgang zur Verfügung liegen. Die HDMI-Ausgänge sind aber gleich geblieben. Schauen Sie sich dazu Abbildung 1.1 an. Dort ist auf der linken Seite das Modell B mit den zwei getrennten Buchsen für Audio und Composite und auf der rechten Seite das Modell 2 B mit einer Buchse für Audio und Composite zu sehen.

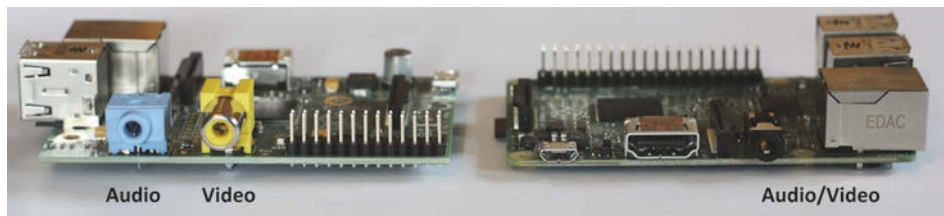


Abb. 1.1: Audio- und Videoausgänge beim Modell B und den Modellen 2 B und B+

1.1.3 Audio

Der Audio-Ausgang steht über einen 3,5-mm-Klinkenstecker zur Verfügung und kann an einen Verstärker oder über ein Adapterkabel zusammen mit dem Composite-Ausgang auf einen Scart-Stecker gelegt werden. Die Ausgangsleistung ist aber so gering, dass Sie einen Lautsprecher nicht direkt ansteuern können. Sie benötigen hier also einen zusätzlichen Verstärker oder eine Aktivbox.

Die Tonqualität des Audio-Ausgangs über den Klinkenstecker ist nicht bei allen Modellen gleich. Bei den Modellen A+, B+ und 2 B wurde die Audioqualität gegenüber den Modellen A und B verbessert. Dies macht sich aber erst bemerkbar, wenn Sie den Raspberry Pi an ein rauscharmes Netzteil anschließen. Wenn Sie auf bessere Audioqualität Wert legen, sollte das

Netzteil einen Filter besitzen. Für diese Zwecke ist dann ein klassisches Netzteil mit Trafo, Festspannungsregler und Filter gegenüber einem einfachen Schaltnetzteil vorzuziehen.

Neben dem Audio-Ausgang werden Sie vielleicht eine Eingangsbuchse vermissen, die Sie auf dem Board auch nicht finden werden. Wenn Sie Audio-Eingänge benötigen, können Sie diese z.B. über ein USB-Mikrofon realisieren. Für zusätzliche Audio-In/Out-Geräte steht außerdem der I²S-Bus auf dem GPIO-Interface zur Verfügung.

1.1.4 Netzwerk und USB

Bei einem Blick auf das Board fällt neben dem SoC noch ein zweites IC ins Auge. Und zwar das LAN9512 der Firma SMSC. In ihm steckt ein Netzwerk-Adapter mit 10/100-Ethernet- und Full-duplex-Betrieb-Unterstützung. Außerdem wird darüber der USB-2.0-Hub bereitgestellt.

In Abbildung 1.2 sehen Sie den Netzwerk- und USB-Anschluss des Raspberry Pi. Auf der linken Seite ist das Modell B und auf der rechten Seite das Modell 2 B im Vergleich dargestellt. Das Modell 2 B besitzt an seinem Netzwerkstecker zwei LEDs, wobei eine den Betriebsmodus des Netzwerkadapters und die andere die Netzwerkaktivität anzeigt. Bei dem Modell B sitzen diese LEDs direkt auf dem Board.

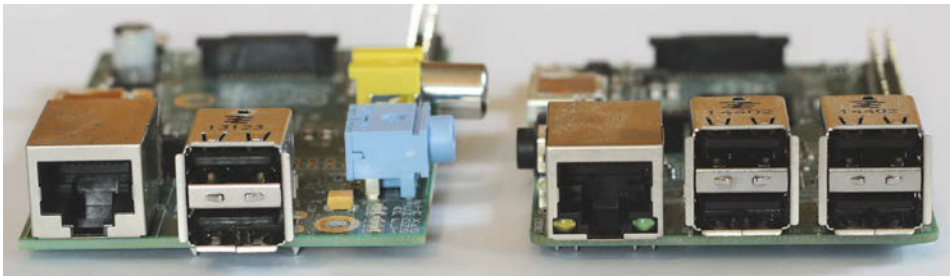


Abb. 1.2: Netzwerk- und USB-Anschlüsse beim Modell B und 2 B

1.1.5 CSI (Camera Serial Interface)

Für den Raspberry Pi gibt es zwei verschiedene 5-Megapixel-Keramodule, die über ein Folienbandkabel angeschlossen werden. Der verwendete Sensor ist bei beiden Modulen gleich und hat eine Auflösung von 2592×1944 Pixeln und ist in der Lage, Videos mit einer Auflösung von 1080p und 30 Bildern in der Sekunde aufzunehmen. Zum Betrieb des Kameramoduls muss aber vorher ein Treiber installiert werden.

Die beiden Kameramodule unterscheiden sich nur dadurch, dass das NoIR-Keramodul keinen Infrarotfilter besitzt. Hierdurch können Sie mit dieser Kamera Infrarotaufnahmen und Aufnahmen bei schlechter Beleuchtung durchführen.

1.1.6 DSI (Display Serial Interface)

Über den DSI-Anschluss können Sie Displays mit Folienbandkabel anschließen. Beim Kauf eines solchen Displays sollten Sie aber unbedingt darauf achten, dass es für das Display einen Treiber gibt, der für das Betriebssystem des Raspberry Pi getestet ist.

Das offizielle Raspberry-Pi-Display hat die Raspberry Pi Foundation im Juni 2015 vorgestellt. Es handelt sich dabei um ein 7-Zoll-Display mit einer Auflösung von 800 x 480 Punkten und wird über das DSI-Interface mit dem Raspberry Pi verbunden. Wenn Sie die aktuelle Betriebssystemversion nutzen, brauchen Sie für das Display auch keinen Treiber zu installieren. Die Kosten für dieses Display betragen ungefähr 80 Euro.

Einige Firmen bieten auch Displays an, die über die 26- oder 40-polige Steckerleiste des Raspberry Pi angeschlossen werden. So bietet die Firma HwLevel für den Raspberry Pi direkt vier verschiedene TFT-LCD-Displays zwischen 3,2" und 5" an, die alle direkt über den GPIO-Stecker mit dem Raspberry Pi verbunden werden. Für das günstigste Display liegt der Preis bei ca. 40 Euro.

Modell	Größe	Auflösung
LCD-PI32	3,2"	320 × 240
LCD-PI33	3,3"	320 × 480
LCD-PI43	4,3"	480 × 272
LCD-PI50	5,0"	480 × 272

Tabelle 1.3: TFT-LCD-Displays der Firma HwLevel

In Abbildung 1.3 können Sie sich die Lage des Display- und Kamera-Interface von Modell B und 2 B anschauen.

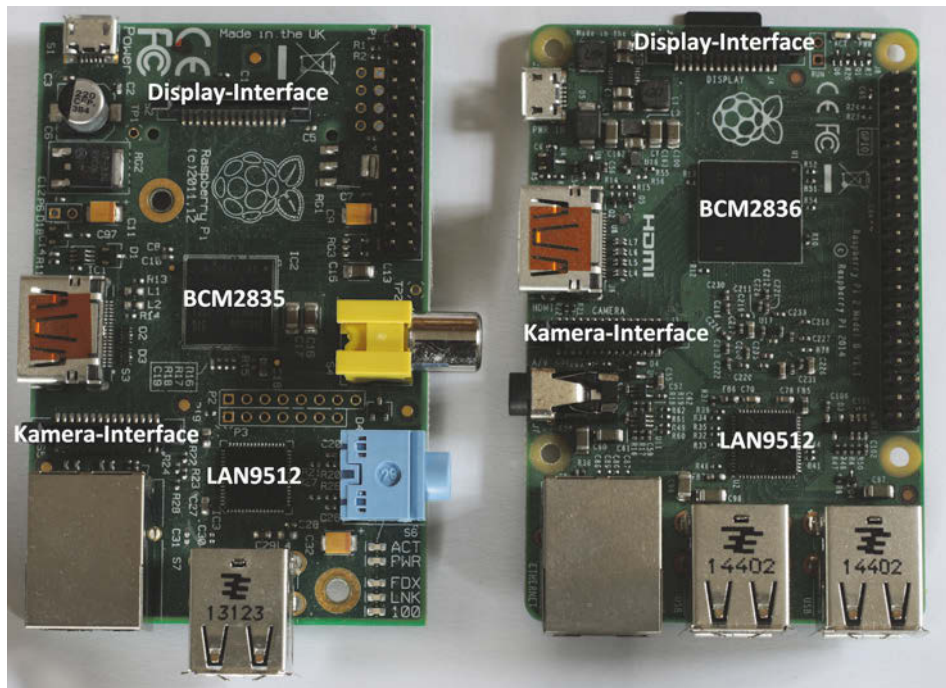


Abb. 1.3: Überblick über die Modelle B und 2 B

1.1.7 Status-LEDs

Auf dem Board B befinden sich in der rechten oberen Ecke 5 Status-LEDs, die Ihnen einige wichtige Informationen liefern. Eine kurze Erklärung zu den Status-LEDs finden Sie in der folgenden Tabelle.

ACT	Leuchtet immer dann, wenn auf die SD-Karte zugegriffen wird.
PWR	Zeigt an, dass der Raspberry Pi mit der Stromversorgung verbunden ist.
FDX	Durch diese LED wird der Full-duplex-Betrieb des Netzwerkadapters angezeigt.
LNK	Immer wenn Netzwerk-Aktivitäten ausgeführt werden, zeigt LNK dies an.
100	Der Netzwerkadapter arbeitet im 100-Mbps-Modus.

Tabelle 1.4: Die Bedeutung der Status-LEDs

Bei den Modellen A+, B+ und 2 B befinden sich nur noch die zwei Status-LEDs ACT und PWR direkt auf dem Board. Zwei Status-LEDs für das Netzwerk finden sich bei diesen Boards direkt am Netzwerkstecker. Den Vergleich zwischen den beiden Modellen B und 2 B zeigt Abbildung 1.4.

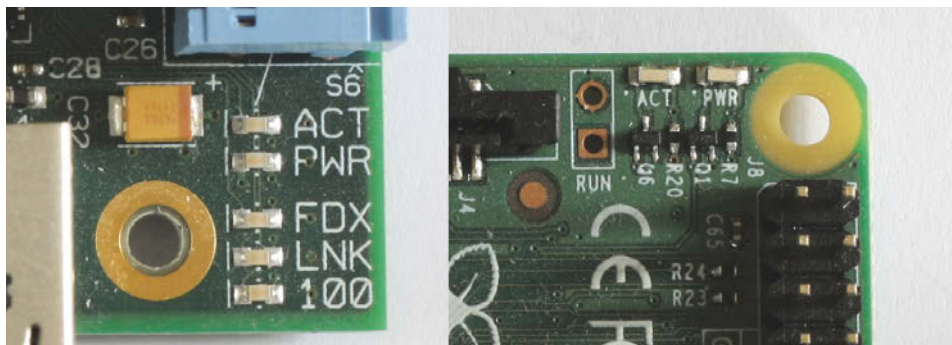


Abb. 1.4: Status-LEDs bei den Modellen B und 2 B

1.1.8 Der Schacht für die SD-Karte

Die Aufnahme für die SD-Karte befindet sich bei allen Modellen auf der Unterseite des Boards. Die SD-Karte wird hier von der Seite in den Schacht eingeschoben und ragt ein Stück über den Rand der Platine hinaus. Bei den Modellen A und B besitzt der Raspberry Pi einen SD-Schacht und bei den Modellen A+, B+ und 2 B einen microSD-Schacht. Den Vergleich zwischen beiden Varianten können Sie in Abbildung 1.5 sehen.

Im nächsten Kapitel werde ich Ihnen zeigen, wie Sie das Betriebssystem des Raspberry Pi auf die SD-Karte kopieren können. Auf ihr wird aber nicht nur das Betriebssystem installiert, sondern sie nimmt auch alle Daten und Programme auf, die Sie auf dem Raspberry Pi benötigen.



Abb. 1.5: Der SD-Schacht beim Modell B und der microSD-Schacht beim Modell 2 B

1.1.9 GPIO (General Purpose Input and Output)

Die 26- beziehungsweise 40-polige Steckerleiste ist die Schnittstelle zu einer ganzen Reihe von Bus-Systemen und einfachen digitalen Ein- und Ausgängen, die Sie über selbst geschriebene Programme ansprechen können. Die Modelle A und B besitzen eine 26-polige Steckerleiste und die Modelle A+, B+ und 2 B eine 40-polige Steckerleiste. In Abbildung 1.6 sehen Sie Position und Nummerierung der Steckerleiste.

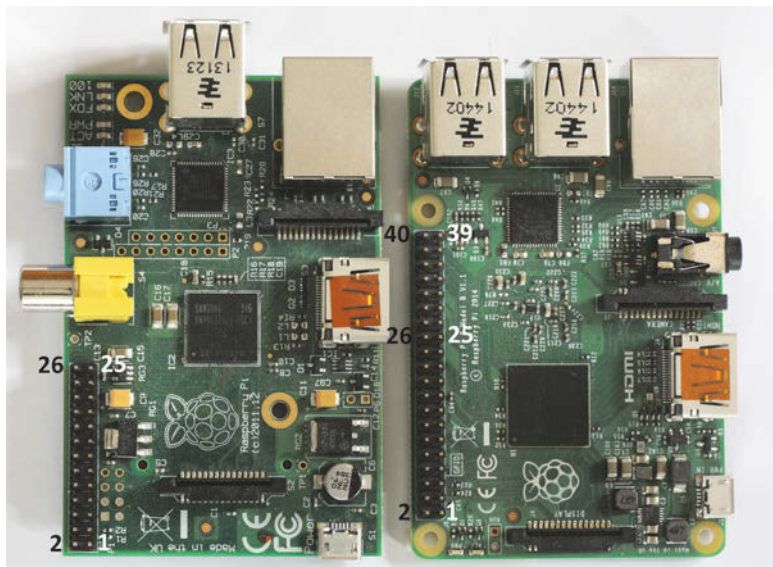


Abb. 1.6: Nummerierung der 26- und 40-poligen Steckerleiste

Einen ersten Überblick über die Belegung der einzelnen Pins zeigt die nächste Tabelle. Beachten Sie bitte, dass in der Tabelle die Belegungen für die 26- und 40-polige Steckerleiste angegeben sind.

Leider gibt es unterschiedliche Bezeichnungen für die einzelnen Pins. Dies liegt ganz einfach daran, dass oft die Pinbezeichnung angegeben wird, wie sie in der Spezifikation des Broadcom SoC definiert ist. Diese sehr Hardware-nahe Bezeichnung kann bei verschiedenen Releases zu einem Modell variieren. Die Bezeichnungen, die in der folgenden Tabelle zu sehen sind, beziehen sich alle auf die in diesem Buch benutzte Java-Bibliothek Pi4J. In Pi4J wurden diese Bezeichnungen abstrahiert und sind deshalb von der eigentlichen Hardwarebezeichnung unabhängig und machen es uns als Java-Programmierer einfacher, da Sie sich nicht um verschiedene Releases eines Boards kümmern müssen.

Die Bedeutung der einzelnen Ports auf der Steckerleiste werden Sie im Laufe des Buches noch kennenlernen.

A+/B+/2 B	A/B	GPIO		A/B	A+/B+/2 B
3,3 V		1	2	5,0 V	
GPIO8 SDA _I (I ² C)	GPIO8 SDA _O (I ² C)	3	4	5,0 V	
GPIO9 SCL _I (I ² C)	GPIO9 SCL _O (I ² C)	5	6	0,0 V	
GPIO7		7	8	GPIO15 Tx _D	
0,0 V		9	10	GPIO16 Rx _D	
GPIO0		11	12	GPIO1	
GPIO2		13	14	0,0 V	
GPIO3		15	16	GPIO4	
3,3 V		17	18	GPIO5	
GPIO12 MOSI (SPI)		19	20	0,0 V	
GPIO13 MISO (SPI)		21	22	GPIO6	
GPIO14 SCLK(SPI)		23	24	GPIO10 CE _O (SPI)	
0,0 V		25	26	GPIO11 CE _I (SPI)	
SDA _O (EEPROM)		27	28		SCL _O (EEPROM)
GPIO21		29	30		0,0 V
GPIO22		31	32		GPIO26
GPIO23		33	34		0,0 V
GPIO24		35	36		GPIO27
GPIO25		37	38		GPIO28
0,0 V		39	40		GPIO29

Tabelle 1.5: Die Pinbelegung der Steckerleiste

Wenn Sie weitere Information zu den verschiedenen Board-Versionen suchen, empfehle ich Ihnen die folgende Webseite

http://elinux.org/Rpi_Low-level_peripherals.

1.2 Ein Gehäuse für den Raspberry Pi

In vielen Veröffentlichungen sieht man den Raspberry Pi in interessanten Projekten die unterschiedlichsten Aufgaben ausführen. Nur leider liegt er dabei oft vollkommen schutzlos auf einem Schreibtisch oder einer Werkbank. Dabei reicht ein Stück Kupferdraht oder eine Büroklammer aus, um den kleinen Linux-Rechner ins Jenseits zu befördern. Ein Kurzschluss kann dann dazu führen, dass die sensible Elektronik zerstört wird und das Board nicht mehr zu reparieren ist. Und dies wäre doch echt schade, da ein einfaches Gehäuse für ca. 5 Euro den Raspberry Pi schon gut schützen kann. Aber nicht nur im Betrieb, sondern auch durch statische Aufladung oder zu starker mechanischer Beanspruchung der Multi-layer-Platine kann er für immer unbrauchbar werden. Für den Einsatz im Elektroniklabor bieten sich besonders die Gehäuse an, bei denen die Steckerleiste des Raspberry Pi von außen zugänglich ist. Bei einem solchen Gehäuse können Sie so die Ports der Steckerleiste mit einem Adapter auf ein Steckbrett führen.



Abb. 1.7: Der Raspberry Pi im Gehäuse

1.3 Das richtige Netzteil

Für den Raspberry Pi benötigen Sie ein 5-V-Netzteil mit Micro-USB-Stecker. Eine einwandfreie Funktion ist aber nur gewährleistet, wenn das Netzteil auch ausreichend Strom liefern kann. Der minimale Verbrauch ohne angeschlossene Geräte ist hier für Modell B ab 700 mA und für Modell A ab 500 mA zu berücksichtigen. Sie sollten sich vor dem Anschließen weiterer Komponenten unbedingt über den Stromverbrauch informieren. So kann der Stromverbrauch von Maus und Tastatur zwischen 100 mA und 1000 mA liegen. Für die richtige Auswahl hilft hier die eLinux-Webseite, auf der viele getestete Komponenten zu finden sind.

http://elinux.org/RPi_VerifiedPeripherals

Verbinden Sie den Raspberry Pi über ein HDMI-Kabel mit einem Monitor, müssen Sie hier mit einem Verbrauch von 50 mA und beim Anschluss des Kameramoduls mit 250 mA rechnen. Die GPIO-Ports dürfen zusammen nicht mehr als 50 mA liefern. Ein einzelner Port kann aber maximal 16 mA bereitstellen.

Wenn Sie für den ersten Probelauf Ihres Raspberry Pi eine Maus und eine Tastatur mit geringem Stromverbrauch und einen Monitor über HDMI anschließen, reicht ein 1,0-A-Steckernetzteil aus. Bei einigen Projekten in diesem Buch muss das Netzteil aber nicht nur den Raspberry Pi, sondern auch zusätzliche Hardware mit Strom versorgen. Aus diesem Grunde habe ich mich hier für ein 2-A-Schaltnetzteil entschieden. Wollen Sie später zusätzliche Geräte über die USB-Anschlüsse nutzen, sollten Sie hierfür einen USB-Hub mit eigener Stromversorgung verwenden.

1.4 Tastatur, Maus und HDMI-Kabel

Wenn Sie eine Tastatur und Maus für den Raspberry Pi kaufen wollen, sollten Sie sich, wie im vorherigen Abschnitt beschrieben, die Liste mit getesteten Geräten auf der eLinux-Webseite anschauen. Auch wenn wir den Raspberry Pi so konfigurieren wollen, dass er headless betrieben werden kann, benötigt man doch immer wieder die Eingabe über Tastatur und Maus.

Ich habe alle Java-Programme, die ich in diesem Buch erläutern werde, direkt am Raspberry Pi entwickelt. Die Entwicklungsumgebung, die ich Ihnen später noch vorstellen werde, läuft über einen direkten Zugang viel flüssiger als über eine Remote-Verbindung.

Wenn Sie keinen Monitor mit HDMI-Eingang besitzen, gibt es für den Anschluss an einen Monitor mit DVI-Eingang Adapterkabel. Für die Verbindung zu einem Monitor mit VGA-Eingang reicht ein einfaches Adapterkabel nicht aus. Hier benötigen Sie dann einen aktiven Adapter mit eigener Stromversorgung. Als letzte Alternative können Sie Ihren Raspberry Pi natürlich auch über den Composite-Ausgang mit einem Fernseher verbinden.

1.5 Die SD-Karte

Um den Raspberry Pi starten zu können, benötigen Sie eine SD-Karte mit einem Linux-Betriebssystem. Die Raspberry Pi Foundation stellt auf ihrer Webseite eine Reihe von verschiedenen Betriebssystemen zum Download zur Verfügung, die alle auf dem Raspberry Pi laufen und für ihn optimiert worden sind. Egal, für welches System man sich entscheidet, man braucht mindestens eine Speicherkapazität von 4 GB. Da man neben dem reinen Betriebssystem noch weitere Software installieren und natürlich auch eigene Daten speichern möchte, sollte die SD-Karte ausreichend groß sein. Auch hier empfehle ich Ihnen einen Blick auf die eLinux-Webseite, auf der es eine riesige Auswahl getesteter Karten gibt.

http://elinux.org/RPi_SD_cards

Neben der Größe sollten Sie auch auf die Klasse der Karte achten, die ein Maß für die Geschwindigkeit der Schreib-/Lesezugriffe angibt. Ich würde Ihnen hier immer eine schnelle Karte empfehlen, die positiv getestet wurde.

Stellen Sie beim Arbeiten am Raspberry Pi fest, dass Sie Text nicht zügig schreiben können oder die Eingabe immer wieder stockt, wird dies wahrscheinlich an einer langsamen SD-Karte liegen und nicht an der Verarbeitungsgeschwindigkeit des Prozessors.

Booten und konfigurieren

Jetzt wollen wir den ersten Probelauf des Raspberry Pi wagen. Und wie Sie sehen werden, ist das Betriebssystem sehr schnell installiert und konfiguriert. Selbst für Menschen, die im Administrieren von Linux-Systemen keine Übung haben, wird es nicht schwer sein, den kleinen Rechner zum Laufen zu bekommen. Und wenn Sie das Betriebssystem mal gründlich durcheinander konfiguriert haben, lässt sich der Grundzustand oder das letzte von Ihnen gespeicherte Betriebssystem-Backup sehr schnell wieder einspielen. Viel Spaß!

2.1 Das Betriebssystem auf die SD-Karte laden

Als Erstes müssen Sie sich die richtige Linux-Distribution für den Raspberry Pi herunterladen. Auf der Webseite der Raspberry Foundation finden Sie unter DOWNLOADS verschiedene Distributionen, die alle auf dem Raspberry Pi installiert werden können.

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

Wir entscheiden uns hier für die Raspbian Distribution, die von der Raspberry Community gepflegt und in sehr kurzen Abständen aktualisiert wird. Das Raspbian-Wheezy-Image ist ein freies Betriebssystem, das auf Debian-Linux aufbaut und für den Raspberry Pi optimiert wurde. Alle anderen Distributionen auf der Download-Seite sind für mehr oder weniger spezielle Aufgabe entwickelt worden.

Hinweis

Downloaden Sie das Raspbian-Wheezy-Image auf Ihren PC und entpacken Sie die Zip-Datei in ein Verzeichnis.

<https://www.raspberrypi.org/downloads/raspbian/>

Nach dem Entpacken sehen Sie die `.img`-Datei, die bei dem Raspbian-Wheezy-Image vom 05.05.2015 3,2 GB groß ist. Für dieses Image muss die SD-Karte 4 GB oder größer sein. Damit Sie das Image auf die SD-Karte kopieren können, benötigen Sie ein Programm, das ein 1:1-Abbild auf die SD-Karte schreiben kann. Windows-Benutzer können sich hierzu den Win32DiskImager auf ihrem PC installieren.

Hinweis

Win32DiskImager herunterladen und die Zip-Datei entpacken.

<http://www.sourceforge.net/projects/win32diskimager/>

Sie benötigen jetzt einen SD-Karten-Leser, um das Raspbian-Image mit dem Win32DiskImager auf die SD-Karte kopieren zu können. Falls Ihr Rechner keinen eigenen SD-Karten-Leser hat, können Sie einen externen Karten-Leser über die USB-Schnittstelle Ihres PC anschließen. Wenn sich jetzt die SD-Karte im Karten-Leser befindet, starten Sie aus dem Installationsverzeichnis die EXE-Datei des Win32DiskImager.

Hinweis

SD-Karte in den Karten-Leser stecken und die Datei `Win32DiskImager.exe` ausführen.

Nachdem Sie den Win32DiskImager gestartet haben, geben Sie den Namen des Raspbian-Image mit Pfadangabe in das Feld `IMAGE FILE` ein. Anschließend noch den Laufwerksbuchstaben des Karten-Lesers auswählen und die Schaltfläche `WRITE` aktivieren. Bevor der Schreibvorgang beginnt, kommt noch eine Sicherheitsabfrage, die Sie darauf aufmerksam macht, dass das Schreiben auf das ausgewählte Gerät dieses eventuell beschädigen kann. Diese Meldung ist für unsere SD-Karte nicht von Bedeutung und wird quittiert. Sie können jetzt erst einmal tief durchatmen und sich entspannt zurücklehnen, bis das Image auf die SD-Karte geschrieben wurde.

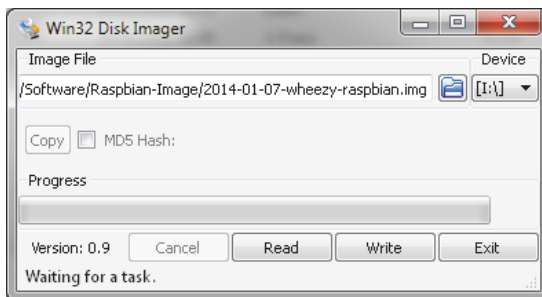


Abb. 2.1: Win32-Disk-Imager mit ausgewähltem Image und Laufwerksbuchstaben

2.2 Booten

Bevor Sie anfangen, den Raspberry Pi mit der Stromversorgung und den Ein-/Ausgabegeräten zu verbinden, kommt hier noch eine kurze Warnung.

Warnung

Alle Geräte und die SD-Karte bitte immer nur mit dem Raspberry Pi verbinden oder entfernen, wenn er sich im stromlosen Zustand befindet.

Stecken Sie jetzt die Tastatur und die Maus in die USB-Anschlüsse des Raspberry Pi. Einen Monitor können Sie über den HDMI-Anschluss oder einen Fernseher über den Composite-Ausgang mit dem Raspberry Pi verbinden. Dann fehlt nur noch die SD-Karte mit dem Raspbian-Image und zu guter Letzt das Steckernetzteil. Sobald die Stromversorgung des Raspberry Pi aktiv ist, beginnt der Boot-Vorgang. Auf dem Monitor sehen Sie jetzt die ganzen Initialisierungsprozesse des Betriebssystems, die Sie erst mal nicht weiter berücksichtigen müssen.

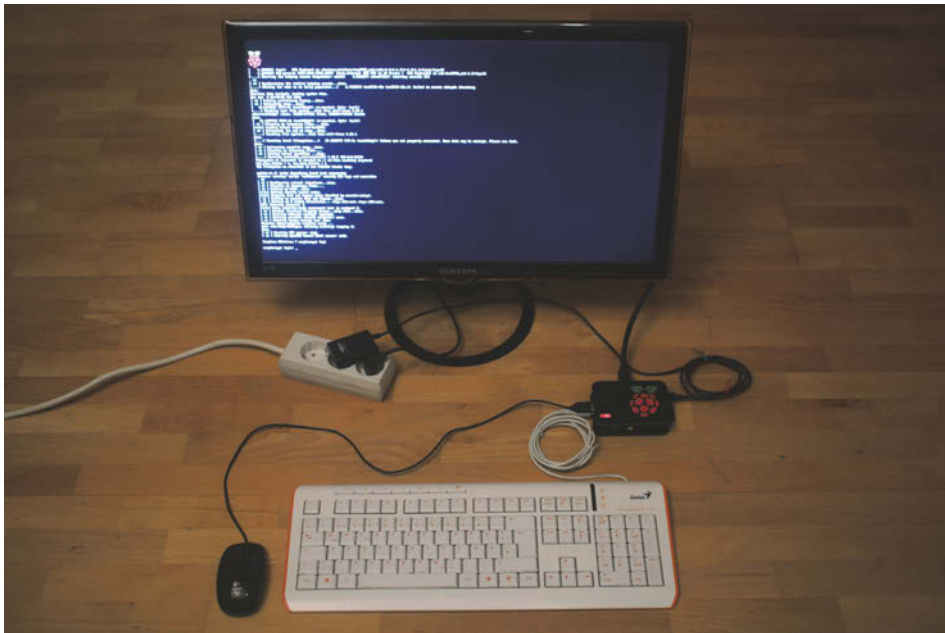


Abb. 2.2: Das erste Mal booten mit Maus, Tastatur und Monitor

Nach der Initialisierung wird automatisch das Konfigurations-Tool *raspi-config* gestartet und erlaubt Ihnen eine menügesteuerte Konfiguration des Raspberry Pi. Nachfolgend werden jetzt alle Menüpunkte kurz beschrieben und die erforderlichen Einstellungen vorgenommen.

Hinweis

Alle Menüangaben beziehen sich auf das Raspbian-Wheezy-Image vom 05.05.2015.

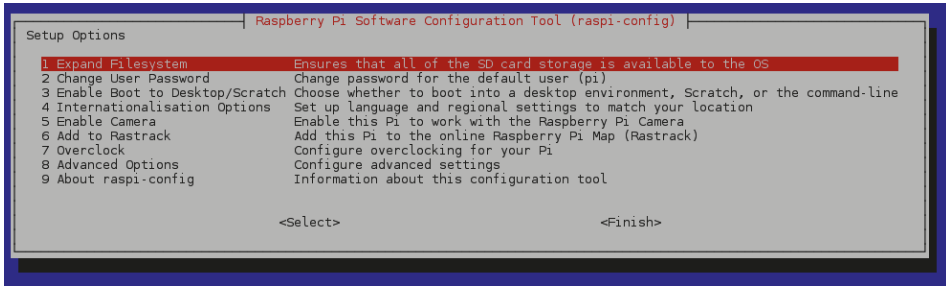




Abb. 2.3: Das Hauptmenü des Konfigurations-Tools `raspi-config`

1. EXPAND FILESYSTEM

Mit EXPAND FILESYSTEM können Sie den ungenutzten Speicherplatz auf der SD-Karte dem Betriebssystem zur Verfügung stellen. Wenn Sie z.B. das Raspbian-Image auf eine 8 GB große SD-Karte geschrieben haben, werden 4,2 GB nicht genutzt.

8 GB (SD-Karte) – 3,2 GB (Raspbian-Image) = 4,8 GB (ungenutzt)

Im Hauptmenü können Sie mit  auf <SELECT> springen und dann mit  das Menü EXPAND FILESYSTEM aktivieren. Danach stehen Ihnen zum Speichern von Daten und weiteren Programmen zusätzlich 4,8 GB Speicher auf dem Raspberry Pi zur Verfügung. Wenn Sie eine größere SD-Karte benutzen, entsprechend mehr. Sie sollten aber bedenken, dass ein Backup des Betriebssystems mit dem Win32-Disk-Imager auf Ihrem PC immer ein Image der Größe Ihrer SD-Karte speichert.

2. CHANGE USER PASSWORD

Auf dem Raspian-Image ist der Benutzer *pi* mit dem Kennwort *raspberry* gesetzt. Wenn Sie möchten, können Sie das Kennwort jetzt ändern. Dies können Sie jedoch auch später an anderer Stelle erledigen.

Default-Einstellung von Benutzer und Kennwort

Benutzer ist **pi** und das Kennwort **raspberry**.

3. ENABLE BOOT TO DESKTOP/SCRATCH

Unter dem Punkt ENABLE BOOT TO DESKTOP/SCRATCH können Sie festlegen, mit welchem Programm der Raspberry Pi nach dem Booten gestartet werden soll. Mit der ersten Option wird direkt die Konsole gestartet. Bei dieser Variante muss man sich mit dem Benutzer *pi* und dem eingestellten Kennwort anmelden. Als zweite Möglichkeit kann der Raspberry Pi in den grafischen Desktop ohne Anmeldung gebootet werden. Als letzte Möglichkeit kann man auch direkt in die grafische Entwicklungsumgebung Scratch starten, die dafür gedacht ist, jungen Menschen den spielerischen Einstieg in die Software-Entwicklung zu erleichtern. Wir wählen hier die zweite Option, damit nach dem Start sofort der grafische Desktop zur Verfügung steht.



Abb. 2.4: Mit dieser Einstellung wird direkt in den grafischen Desktop gebootet.

4. INTERNATIONALISATION OPTIONS

Wenn Sie aus dem Hauptmenü in den Punkt INTERNATIONALISATION OPTIONS navigieren, kommen Sie in ein Untermenü, in dem Ihnen drei weitere Navigationspunkte zur Verfügung stehen. Hier stellen Sie die Sprache mit regionalen Varianten, die Zeitzone und das Tastatur-Layout ein.

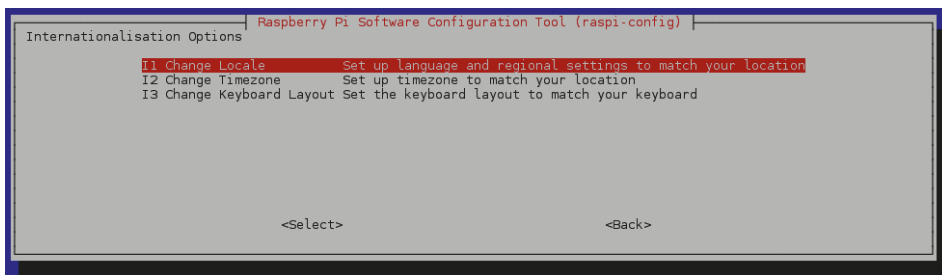


Abb. 2.5: Sprache, Zeitzone und Tastatur-Layout einstellen

Ich habe im Menü CHANGE LOCALE als Land und Sprache *deutsch* gewählt. Und den Zeichensatz wie im Menü empfohlen auf *UTF-8*. Im Menü CHANGE TIMEZONE habe ich die Zeitzone *Europa* mit der Stadt *Berlin* ausgewählt.

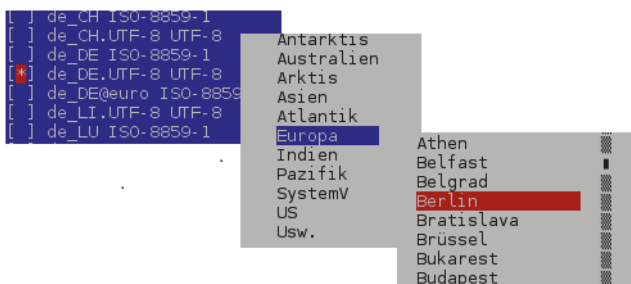


Abb. 2.6: Sprache und Zeitzone festlegen

Im Untermenü CHANGE KEYBOARD LAYOUT müssen Sie die Einstellungen Ihrer angeschlossenen Tastatur anpassen. Für eine deutsche Standard-Tastatur können Sie hier für das Keyboard-Layout GENERIC 105-KEY (INTEL) PC und für die Sprache GERMAN wählen. Für die Einstellungen der Tasten `Alt`, `Alt Gr` und `Control` habe ich alle Standards übernommen.

5. ENABLE CAMERA

In dem Menü ENABLE CAMERA brauchen Sie erst dann Einstellungen vorzunehmen, wenn Sie über den CSI-Stecker das Kameramodul anschließen wollen. Für den Anfang benötigen wir dies aber nicht.

6. ADD TO RASTRACK

Den nächsten Menüpunkt können Sie erst dann nutzen, wenn Sie Ihren Raspberry Pi mit dem Internet verbunden haben. Hier können Sie sich mit einem Benutzernamen auf der Webseite <http://rastrack.co.uk> registrieren und Ihren Standort eintragen. Dieser wird dann zusammen mit all den anderen Raspberry-Pi-Standorten auf einer Weltkarte angezeigt. Ist eine nette Idee, hat aber auf die Funktion Ihres Raspberry Pi keinen Einfluss.

7. OVERCLOCK

In diesen Menüpunkt sollten Sie als Einsteiger erst einmal keine Veränderungen vornehmen. Hier können Sie nämlich die CPU des Raspberry Pi übertakten, um noch die letzten Leistungsreserven zu nutzen. Dies hat aber zur Folge, dass sich die CPU stark erwärmen kann und sie ohne zusätzliches Kühlblech eventuell überhitzt. Im Rahmen dieses Buches ist ein solches Tuning auch nicht notwendig.

8. ADVANCED OPTIONS

Über das Menü ADVANCED OPTIONS kommen Sie zu einem Untermenü mit 7 weiteren Unterpunkten, in denen wir weitere Konfigurationen vornehmen. Vorweg möchte ich Sie aber auf eine etwas irritierende Navigation aufmerksam machen. Wenn Sie hier in einen Punkt hineinnavigieren und ihn verlassen, landen Sie immer wieder im Hauptmenü. Von dort aus müssen Sie sich dann immer wieder bis zu dem nächsten Unterpunkt durchhangeln.

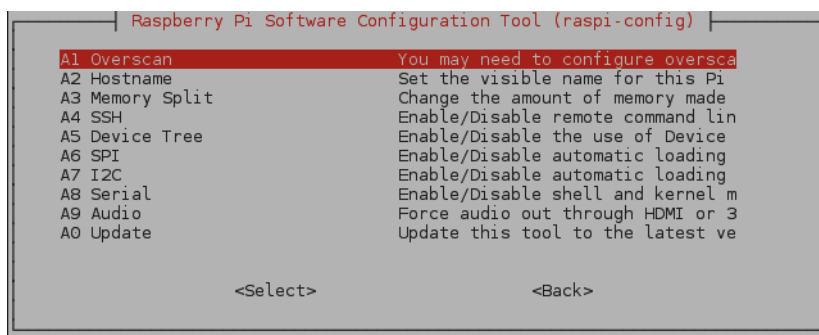


Abb. 2.7: Das Untermenü ADVANCED OPTIONS

■ ADVANCED OPTIONS: OVERSCAN

Der Menüpunkt OVERSCAN wird für Sie interessant sein, wenn Sie später mit dem grafischen Desktop des Raspberry Pi arbeiten und auf Ihrem Monitor am Rand schwarze

Streifen sehen, das Bild also zu klein dargestellt wird. Hier können Sie durch das Freigeben von OVERSCAN zu besseren Ergebnissen kommen.

■ ADVANCED OPTIONS: HOSTNAME

Der Raspberry Pi hat als Hostname *raspberrypi* eingestellt. Wenn Sie diesen Namen ändern möchten, können Sie ihn unter dem Punkt HOSTNAME überschreiben. Bei den weiteren Beschreibungen in diesem Buch gehe ich aber immer von dem Standardnamen aus.

■ ADVANCED OPTIONS: MEMORY SPLIT

Unter dem Punkt MEMORY SPLIT können Sie der GPU (Grafikeinheit) mehr Speicher zuweisen. Standardmäßig ist dieser auf 64 MB eingestellt. Wenn Sie den Raspberry Pi für 3D-Spiele nutzen, kann es erforderlich sein, diesen Wert nach oben zu ändern. Dies müssen Sie sich aber gut überlegen, da Sie dadurch dem Betriebssystem und allen laufenden Programmen Speicher entziehen.

■ ADVANCED OPTIONS: SSH

Im nächsten Punkt werden wir SSH aktivieren. Dies ist wichtig, wenn wir den Raspberry Pi später headless betreiben wollen, also alle Zugriffe über das Netzwerk laufen und Tastatur, Maus und einen zusätzlichen Monitor für den Raspberry Pi nicht mehr nötig sind.

■ ADVANCED OPTIONS: DEVICE TREE

Der Device Tree unterstützt das Laden von Kernelmodulen, die für Erweiterungsboards notwendig sind, die zum Beispiel am I²C- oder SPI-Bus angeschlossen sind. Hier nehmen wir keine Änderungen vor.

■ ADVANCED OPTIONS: SPI

Wenn Sie über das GPIO-Interface Geräte betreiben möchten, die die SPI-Schnittstelle nutzen, können Sie hier das Laden des SPI-Kernel-Moduls freigeben. Dies werden wir später nachholen, wenn der SPI-Bus über eigene Java-Anwendungen angesprochen werden soll.

■ ADVANCED OPTIONS: I²C

Auch in diesem Punkt müssen Sie jetzt keine Änderungen vornehmen, da die Konfiguration des I²C-Bus in Kapitel 16 erklärt wird.

■ ADVANCED OPTIONS: SERIAL

Auf der 26- und 40-poligen Steckerleiste des Raspberry Pi liegt auch die serielle Schnittstelle. In diesem Menüpunkt können Sie diese für ein Terminal freigeben. Sie haben dann die Möglichkeit, über ein Terminalprogramm von einem anderen Rechner auf die Konsole des Raspberry Pi zuzugreifen. Dazu müssen Sie aber beide Geräte mit einem entsprechenden Kabel verbinden und den Raspberry mit einem Kabeltreiber ausstatten. Wir nehmen in diesem Punkt keine Änderungen vor.

■ ADVANCED OPTIONS: AUDIO

Im Menüpunkt AUDIO können Sie die Ausgabe des Audiosignals fest auf HDMI, den 3,5 MM KLINKENSTECKER oder auf AUTOMATISCH einstellen lassen. Sie brauchen den Default-Wert AUTO nicht zu verändern.

■ ADVANCED OPTIONS: UPDATE

Im letzten Menüpunkt können Sie *raspi-config* auf die neueste Version updaten. Hierfür brauchen Sie aber Netzwerkzugriff, den Sie noch nicht haben. Diesen Punkt können Sie später durchführen. Sie können aber davon ausgehen, dass bei einem gerade herun-

tergeladenen Raspbian-Image auch *raspi-config* auf dem letzten Stand ist und ein Update erst später Sinn macht.

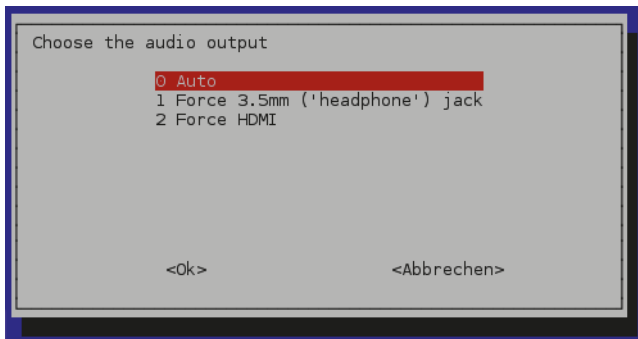


Abb. 2.8: Die Audio-Ausgabe konfigurieren

Jetzt geht es zurück ins Hauptmenü, wo Sie *raspi-config* über <FINISH> beenden können. Sie werden dann noch gefragt, ob neu gebootet werden soll, was unbedingt getan werden sollte, da sonst die vorgenommenen Konfigurationen erst beim nächsten Neustart wirksam werden.

2.2.1 Reboot und richtig ausschalten

Nach dem Neustart wird jetzt der grafische Desktop geöffnet und Sie könnten sich mit der Oberfläche des Desktops vertraut machen. Ich möchte Ihnen aber zuerst zeigen, wie Sie den Raspberry Pi richtig abschalten. Dies ist schon wichtig, da das Betriebssystem Schaden nehmen könnte, wenn Sie zum Ausschalten einfach die Stromversorgung unterbrechen.

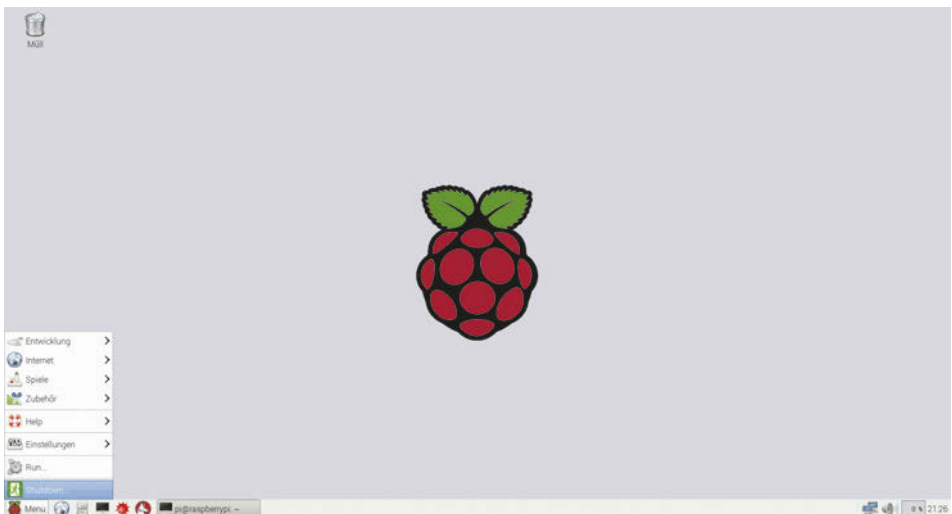


Abb. 2.9: Der grafische Desktop

Über den Menüpunkt SHUTDOWN öffnen Sie den kleinen Dialog END SESSION. Ihnen stehen nun drei Möglichkeiten zur Auswahl. Mit dem Punkt SHUTDOWN fahren Sie den Raspberry Pi herunter und können ihn dann vom Netz trennen. Warten Sie aber mit dem Trennen der Spannungsversorgung so lange, bis alle Status-LEDs außer der Power-LED erloschen sind. Warten Sie dann noch eine Sekunde und ziehen Sie dann den Netzstecker.

Mit REBOOT erzwingen Sie einen Neustart und mit LOGOUT können Sie sich von dem grafischen Desktop abmelden. Es erscheint dann ein Anmeldedialog, über den Sie sich mit dem Benutzer pi und dem aktuellen Kennwort (raspberrypi) anmelden können.

Übrigens erscheint die Taskleiste nach dem ersten Start am oberen Rand; wenn Sie dies ändern möchten, klicken Sie mit der rechten Maustaste auf die Taskleiste und ändern die Position unter dem Menüpunkt PANEL-EINSTELLUNGEN.

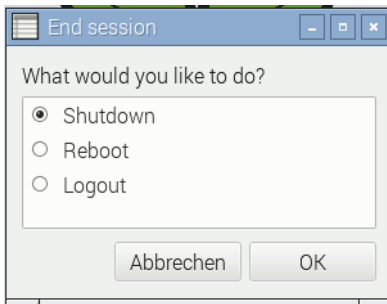


Abb. 2.10: Das Menü zum Ausschalten des Raspberry Pi

Eine zweite Möglichkeit, den Raspberry Pi herunterzufahren, ist über die Konsole. Dazu öffnen Sie das LXTerminal, das sich auch auf der Taskleiste befindet, und geben dort die folgende Befehlszeile ein:

```
$ sudo shutdown -h now
```

Den shutdown-Befehl können Sie nur als root-User ausführen. Aus diesem Grunde steht vor dem Befehl noch `sudo`, wodurch Sie für einen Befehl diese Rechte verliehen bekommen. Zu dem LXTerminal und dem Ausführen von Linux-Befehlen werden Sie später noch mehr erfahren.

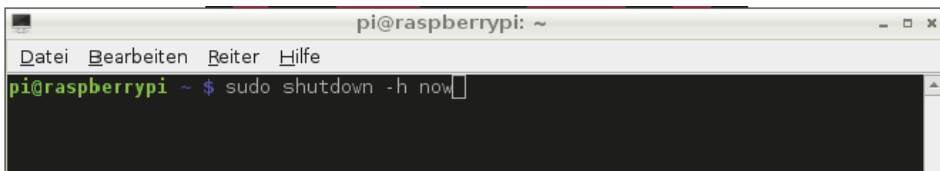


Abb. 2.11: Den Raspberry Pi über die Kommandozeile herunterfahren

2.2.2 Mögliche Fehler

Wenn Sie alles berücksichtigt haben und auch die Hardware einwandfrei funktioniert, müsste alles so laufen, wie ich es Ihnen beschrieben habe. Aber manchmal klappt es dann doch nicht sofort und dann brauchen Sie Hilfe. Aber woher? Eine sehr gute Quelle für solche Fälle bietet die eLinux-Webseite. Hier finden Sie unter TROUBLESHOOTING jede Menge Fehler beschrieben, die irgendwann schon einmal aufgetaucht sind.

http://elinux.org/R-Pi_Troubleshooting

Oft sind es aber nur Kleinigkeiten, die den richtigen Ablauf verhindern. Aus diesem Grunde habe ich Ihnen eine kleine Checkliste zusammengestellt, die Sie erst einmal durchschauen können. Wenn dies nicht hilft, sollten Sie auf der oben beschriebenen Webseite nachschauen.

Fehler	Abhilfe
Der Bildschirm bleibt dunkel.	Verbindung zwischen Raspberry Pi und Monitor prüfen. Ist am Monitor der richtige Eingang selektiert worden? Liefert das Steckernetzteil genug Strom für alle Komponenten? Tastatur und Maus benötigen eventuell zu viel Strom. Sitzt die SD-Karte richtig? Hat der Kopiervorgang des Raspbian-Image richtig funktioniert? Kopierschutz an der SD-Karte aufheben und eventuell neu kopieren.
Der Bootvorgang wird unterbrochen.	Stromversorgung reicht nicht aus. Tastatur und Maus benötigen zu viel Strom oder sind defekt.
Das Konfigurationswerkzeug <i>raspi-config</i> wird nicht automatisch gestartet und Sie werden zur Eingabe von Benutzernamen und Kennwort aufgefordert.	Benutzername: pi Kennwort: raspberrry <i>raspi-config</i> starten: sudo raspi-config
Bei der Eingabe des Kennworts werden Sie abgewiesen.	Wenn Sie in <i>raspi-config</i> noch kein deutsches Tastaturlayout angegeben haben, sind auf der Tastatur die Buchstaben y und z vertauscht. Dies führt zur Eingabe <i>raspberrrz</i> statt <i>raspberrry</i> .
Es wird nicht automatisch in den grafischen Desktop gebootet.	<i>raspi-config</i> starten und in den 3. Menüpunkt ENABLE BOOT TO DESKTOP/SCRATCH navigieren. Dort den Punkt DESKTOP LOG IN AS USER 'PI' AT THE GRAPHICAL DESKTOP selektieren und dann <i>raspi-config</i> beenden. Auf der Kommandozeile können Sie mit dem Befehl startx den Desktop starten.

Tabelle 2.1: Die Fehlersuche, wenn mal etwas nicht funktioniert

2.3 Der Raspberry Pi und das Netzwerk

Wir wollen den Raspberry Pi jetzt mit dem Netzwerk verbinden. Ich beschreibe hier das Vorgehen in einem typischen Heimnetzwerk, in dem alle Geräte an einen Router angeschlossen werden. Ich gehe davon aus, dass in Ihrem Netzwerk mindestens ein Rechner schon mit dem Router verbunden ist und dieser Zugriff auf das Internet hat. Fahren Sie den Raspberry Pi herunter und trennen Sie die Stromversorgung. Verbinden Sie jetzt den Router und den Raspberry Pi mit einem Netzkabel. Wenn Sie jetzt booten, sollte der grafische Desktop erscheinen und der Raspberry Pi mit dem Netzwerk beziehungsweise mit dem Internet verbunden sein.

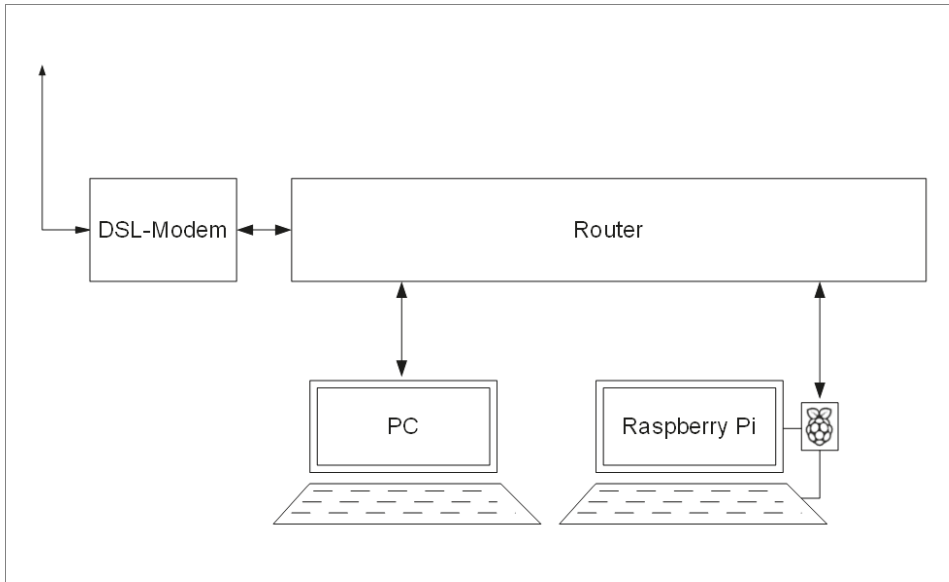
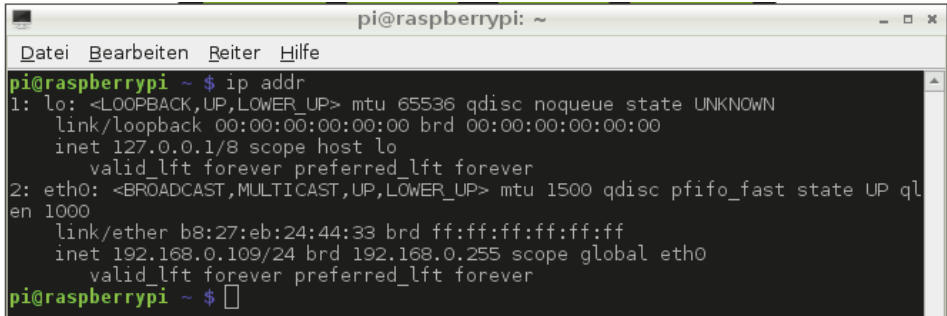


Abb. 2.12: Den Raspberry Pi mit dem Router verbinden

Bevor wir uns jetzt dem Raspberry Pi widmen, möchte ich noch ein paar Worte zur Router-Konfiguration verlieren. Um netzwerkfähige Geräte ganz einfach über den Router ins Heimnetzwerk zu integrieren, kann man in dem Router einen DHCP-Server aktivieren. Erkennt der Router einen Rechner, so vergibt er automatisch eine IP-Adresse an dieses Gerät. Um herauszufinden, welche IP-Adresse der Router an den Raspberry Pi vergeben hat, starten Sie das LXTerminal vom Desktop des Raspberry Pi und geben Sie den folgenden Befehl ein.

```
$ ip addr
```

Als Antwort erhalten Sie die Ausgabe wie in Abbildung 2.13.



```

pi@raspberrypi ~ $ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP ql
en 1000
    link/ether b8:27:eb:24:44:33 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.109/24 brd 192.168.0.255 scope global eth0
        valid_lft forever preferred_lft forever
pi@raspberrypi ~ $
  
```

Abb. 2.13: IP-Adresse ermitteln

Hinter INET können Sie die vergebene IP-Adresse ablesen. In unserem Fall ist dies die 192.168.0.109. Außerdem sehen Sie die MAC-Adresse des Netzwerkadapters. Beide Informationen wollen wir uns hier merken. Sie müssen die hier aufgeführten Werte natürlich durch die Werte ersetzen, die Sie in Ihrem Netzwerk ermittelt haben.

IP: 192.168.0.109

MAC: B8:27:2B:24:44:33

Noch wichtiger als die IP-Adresse ist die MAC-Adresse. Diese ist nämlich von der Konfiguration unabhängig und sie ist die Hardware-Adresse des Netzwerkadapters Ihres Raspberry Pi. Diese Adresse ist im ganzen Internet eindeutig.

Wenn wir später den Raspberry Pi headless betreiben möchten, haben wir keinen direkten Zugriff mehr. Den Raspberry Pi erreichen Sie dann nur noch über das Netzwerk. Aber woher bekommen Sie dann die vom Router vergebene IP-Adresse? Im Router können Sie der MAC-Adresse eine feste IP-Adresse zuordnen, hierzu suchen Sie in Ihrem Router nach dem entsprechenden Menü (z.B. IP & MAC BINDING) und fügen einen Eintrag mit der aktuellen IP-Adresse und der MAC-Adresse Ihres Raspberry Pi hinzu.

Nachdem Sie die Eintragungen vorgenommen haben, fahren Sie den Router und den Raspberry Pi herunter und wieder hoch. Im Raspberry Pi können Sie dann mit dem Befehl `ip addr` überprüfen, ob dieselbe Adresse vergeben wurde.

Es gibt aber noch eine andere Methode, den Raspberry Pi mit einer festen IP-Adresse zu versorgen. Hierbei wird die Netzwerkkonfiguration des Raspberry Pi so geändert, dass die IP-Adresse fest eingetragen wird. Man spricht dabei auch von einer statischen IP-Adresse. Falls Sie die IP-MAC-Bindung ausprobiert haben, müssen Sie dies unbedingt wieder rückgängig machen, bevor Sie die nächsten Änderungen vornehmen.

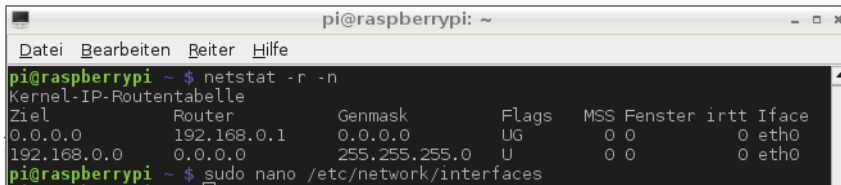
Dem Raspberry Pi eine statische IP-Adresse zuweisen

Bevor Sie dem Raspberry Pi eine statische IP-Adresse zuweisen, müssen Sie im Router erst einmal nachschauen, welcher IP-Bereich in Ihrem Netzwerk zur Verfügung steht. Eine gültige IP-Adresse bekommen Sie aus dem DHCP-Einstellungen Ihres Routers, wo es einen vorgegebenen Adressbereich gibt. Typisch ist z.B. ein IP-Bereich von 192.168.0.100 bis 192.168.0.199. Sie können also in Ihrem Heimnetzwerk 200 Adressen vergeben, die sich alle nur durch die letzte Zahl unterscheiden. Ich habe hier die Adresse 192.168.0.120 gewählt. Beachten Sie aber bitte, dass der IP-Bereich in Ihrem Router anders aussehen kann.

Für die Konfiguration auf dem Raspberry Pi müssen Sie erst einmal das Gateway und die Subnet-Mask ermitteln. Öffnen Sie hierzu das LXTerminal, das auf dem Desktop des Raspberry Pi liegt. Hier geben Sie dann den Befehl

```
$ netstat -r -n
```

ein und schauen sich die Ausgabe an. In der Zeile, die mit der IP-Adresse 0.0.0.0 beginnt, finden Sie unter der Spalte ROUTER die Adresse des Gateways. In unserem Fall ist dies die 192.168.0.1. In der Zeile, die nicht mit 0.0.0.0 beginnt, können Sie die Subnet-Mask mit der Adresse 255.255.255.0 ablesen.



```
pi@raspberrypi: ~  
Datei Bearbeiten Reiter Hilfe  
pi@raspberrypi ~$ netstat -r -n  
Kernel-IP-Routentabelle  
Ziel Router Genmask Flags MSS Fenster irtt Iface  
0.0.0.0 192.168.0.1 0.0.0.0 UG 0 0 0 eth0  
192.168.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0  
pi@raspberrypi ~$ sudo nano /etc/network/interfaces
```

Abb. 2.14: Der Linux-Befehl netstat

Mit den hier ermittelten Adressen können wir nun die Datei `interfaces` aus dem Verzeichnis `/etc/network` bearbeiten. Dafür brauchen wir einen Texteditor und die entsprechenden Rechte, um die Änderungen überhaupt vornehmen zu können. Dies können Sie mit der Eingabe der folgenden Befehlszeile erreichen.

```
$ sudo nano /etc/network/interfaces
```

Durch den `sudo`-Befehl können Sie einen Befehl als Administrator ausführen. In unserem Fall ist dies der Aufruf des Texteditors `nano` mit Pfad und Dateiangabe der zu bearbeitenden Datei. Schauen Sie sich jetzt die geöffnete Datei `interfaces` an und suchen Sie die Zeile:

```
iface eth0 inet dhcp
```

Wenn Sie ein Raspbian-Wheezy-Image ab dem 05.05.2015 benutzen, kann diese Zeile auch so aussehen:

```
iface eth0 inet manual
```

Diese ersetzen Sie dann durch die folgenden Zeilen:

```
iface eth0 inet static  
address 192.168.0.120  
netmask 255.255.255.0  
gateway 192.168.0.1
```

Wenn Sie die Änderungen eingetippt haben, speichern Sie alles mit **Strg**+**X**, **J** und **↩** ab. Starten Sie jetzt den Raspberry Pi und Ihren Router, damit die Änderungen übernommen werden. Der Raspberry Pi ist jetzt über die IP-Adresse 192.168.0.120 zu erreichen.

2.4 Der Desktop

Jetzt wollen wir uns die wichtigsten Funktionen des Desktops anschauen. Da der Desktop bei Raspbian-Wheezy-Images, die vor dem 05.05.2015 erstellt wurden, etwas anders aussieht, beziehe ich mich hier auf die neueren Versionen.

Schauen Sie sich nun den linken Teil der Taskleiste an. Hier finden Sie vier Symbole, die ich Ihnen kurz erkläre.



Abb. 2.15: Die Symbole von links nach rechts: MENÜ, WEBBROWSER, DATEIMANAGER, LXTerminal

Über das erste Symbol erreichen Sie das Hauptmenü des Desktops. Hier finden Sie alle vorinstallierten Programme mit grafischem Interface, also alles, was Sie über den Desktop öffnen können. Danach kommt das Symbol für den Webbrowser und als Drittes der Dateimanager, mit dem Sie sehr einfach im Verzeichnisbaum navigieren können. Und ganz rechts sehen Sie das LXTerminal, mit dem Sie direkt Befehle an das Betriebssystem absetzen können.

Auf der rechten Seite der Taskleiste finden Sie vier weitere Symbole. Hier wird Ihnen als Erstes das Netzwerksymbol angezeigt. Wenn in diesem Symbol in den beiden Bildschirmen jeweils drei helle Punkte zu sehen sind, ist der Raspberry Pi mit dem Netzwerk verbunden.

Über das Lautsprechersymbol können Sie die Lautstärke des Audioausgangs einstellen. In dem dritten Symbol sehen Sie die Prozessorauslastung in Form von einem kleinen Balkendiagramm. Und ganz rechts wird Ihnen die aktuelle Uhrzeit angezeigt, die sich der Raspberry Pi übrigens von einem Zeit-Server aus dem Internet bezieht. Wenn Sie den Raspberry Pi ausschalten, vergisst er Uhrzeit und Datum. Bei jedem Start holt er sich diese aber immer wieder neu aus dem Internet.



Abb. 2.16: Die Symbole von links nach rechts: NETZWERK, LAUTSTÄRKEREGLER, PROZESSORAUSLASTUNG und UHRZEIT

Alle Einstellungen am Desktop können Sie unter dem Menü **EINSTELLUNGEN** vornehmen. Hier kann das Erscheinungsbild des Desktops und der darauf dargestellten Fenster beeinflusst werden. Sie können außerdem Einfluss darauf nehmen, mit welchen Maus-Aktionen Fenster geschlossen oder vergrößert werden. Außerdem können Sie für Maus und Tastatur das Zeitverhalten ändern.



Abb. 2.17: Das Menü EINSTELLUNGEN

Es gibt auch einen Taskmanager, den Sie über den Menüpunkt ZUBEHÖR aufrufen können. Hier können Sie sich CPU- und RAM-Auslastung anschauen. Außerdem wird der Ressourcen-Bedarf von aktiven Prozessen in einer Liste angezeigt.

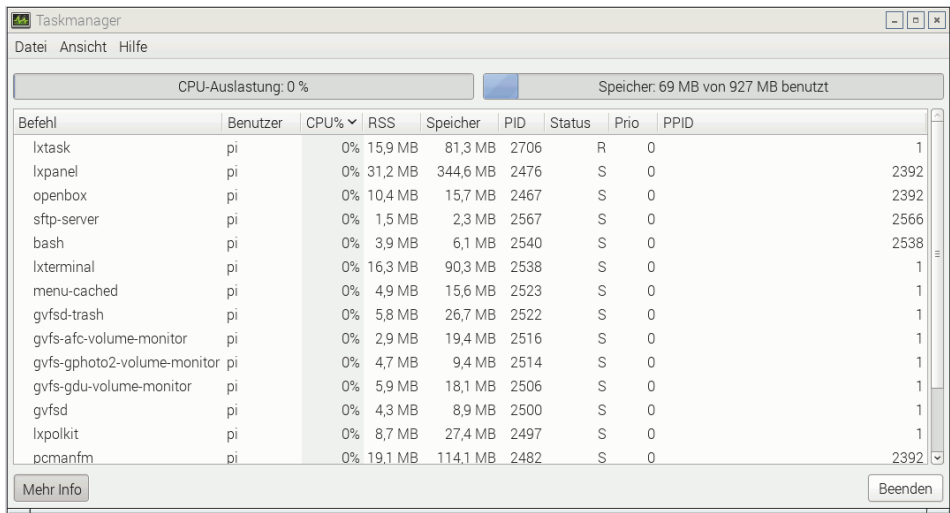


Abb. 2.18: Der Taskmanager

Zum Schluss beschreibe ich Ihnen noch eine Auswahl an Menü-Icons, die das Raspbian Image bereitstellt. Über den Menüpunkt HELP gelangen Sie zur DEBIAN-REFERENZ, in der Sie viele nützliche Informationen über das Betriebssystem nachlesen können. Die Re-

renz beginnt mit einem Linux-Tutorial und führt Sie Schritt für Schritt durch alle technischen Details der Debian-Distribution.



Abb. 2.19: Verschiedene Menü-Icons

Wenn Sie das Icon RASPBERRY PI HELP aktivieren, das sich ebenfalls im Menüpunkt HELP befindet, wird der Webbrowser geöffnet und die Hilfeseite der Raspberry Pi Foundation angezeigt.

<https://www.raspberrypi.org/help/>

Von hier aus finden Sie die Dokumentation, Projekte, Veranstaltungen und vieles mehr rund um den Raspberry Pi.

Über das Icon TEXT EDITOR aus dem Menüpunkt ZUBEHÖR öffnen Sie den Texteditor Leafpad, mit dem Sie Textdateien erzeugen und bearbeiten können. Und wenn Sie Archive erstellen oder öffnen wollen, können Sie dies über den Xarchiver sehr bequem ausführen.

Mit dem Icon PDF VIEWER können Sie den Xpdf öffnen und sich PDF-Dokumente anschauen. Dieses Programm werden Sie sehr oft benötigen, da technische Dokumentationen meist im PDF-Format vorliegen.

2.5 Den Raspberry Pi headless betreiben

Den Raspberry Pi headless betreiben, heißt, dass wir nur über das Netzwerk mit ihm kommunizieren. Tastatur, Maus und ein zusätzlicher Monitor sind nicht mehr notwendig. Alle Ein- und Ausgaben werden jetzt von einem anderen Rechner übernommen. Auf dem PC, der mit dem Raspberry Pi kommunizieren soll, müssen Sie aber zuerst die richtige Software installieren, damit auch eine Verbindung aufgebaut werden kann. Es gibt verschiedene Protokolle und Software, mit der Sie dies umsetzen können.

Mit SSH auf den Raspberry Pi zugreifen

Wir wollen hier mit dem SSH-Protokoll beginnen, weil dies schon über unser Konfigurationswerkzeug *raspi-config* freigeschaltet wurde. Falls Sie die Freischaltung noch nicht vorgenommen haben, sollten Sie dies wie weiter oben beschrieben nachholen. Über SSH können Sie sich direkt mit der Konsole des Raspberry Pi verbinden und dort genau so arbeiten, wie Sie es bereits mit dem LXTerminal getan haben.

Als Benutzer eines Windows-PC können Sie hier die Software PuTTY verwenden. Laden Sie dazu die `putty.exe` von der folgenden Webseite herunter und starten Sie diese:

<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

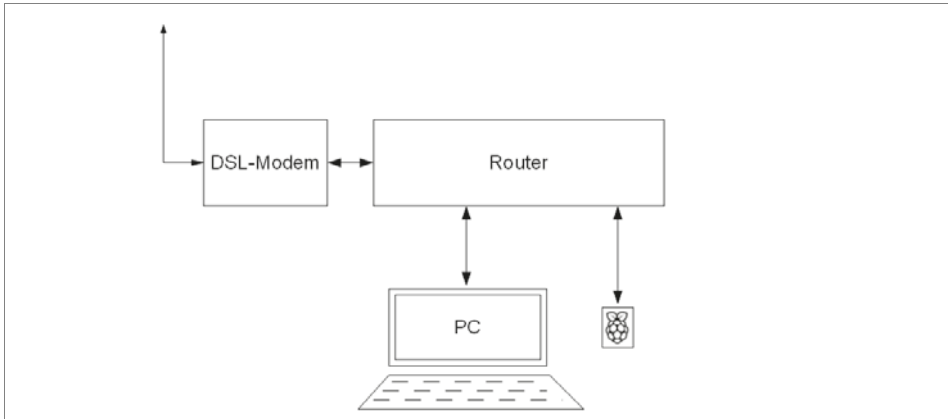


Abb. 2.20: Der Raspberry Pi ist nur noch über das Netzwerk mit der Außenwelt verbunden.

Ohne Installation wird PuTTY direkt ausgeführt und das Konfigurationsfenster wie in Abbildung 2.21 dargestellt. Zum Aufbau einer Session brauchen Sie jetzt die IP-Adresse, die Sie dem Raspberry Pi vergeben haben. In dem Konfigurationsfenster wählen Sie aus dem Baummenü den Punkt SESSION aus, damit Sie auf der rechten Seite die Maske BASIC OPTIONS FOR YOUR PUTTY SESSION erhalten. In das Feld HOST NAME geben Sie dann die IP-Adresse Ihres Raspberry Pi ein. Den Port tragen Sie mit 22 ein und wählen als CONNECTION TYPE SSH aus. Wenn Sie möchten, können Sie in SAVED SESSIONS einen Text eingeben und die Konfiguration dann mit der Schaltfläche SAVE speichern. Wie Sie in Abbildung 2.22 sehen, habe ich die Konfiguration unter dem Namen *Raspberry PI LAN* abgespeichert.

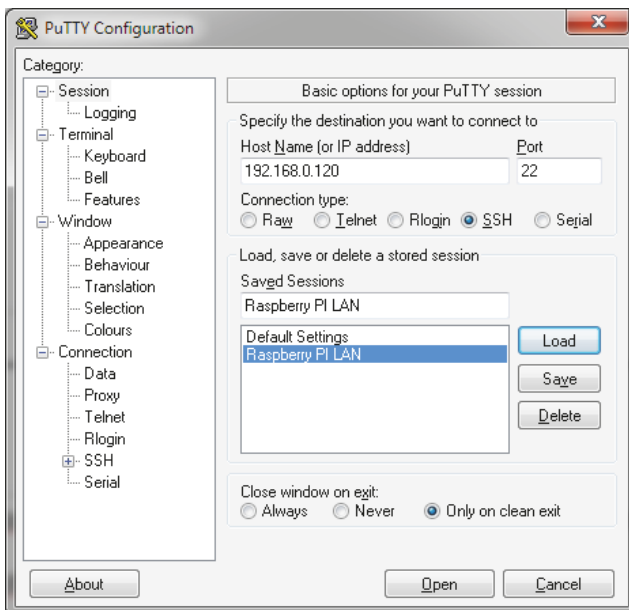


Abb. 2.21: Eine SSH-Verbindung mit PuTTY

Aktivieren Sie die Schaltfläche OPEN, um die Verbindung aufzubauen. Wenn der Raspberry Pi unter der angegebenen Adresse gefunden wurde, öffnet sich jetzt ein Fenster, in dem Sie sich mit *pi* als Benutzer und dem Kennwort *raspberry* anmelden können. Beachten Sie bitte, dass Sie ein anderes Kennwort eingeben müssen, wenn Sie dies über *raspi-config* geändert haben.

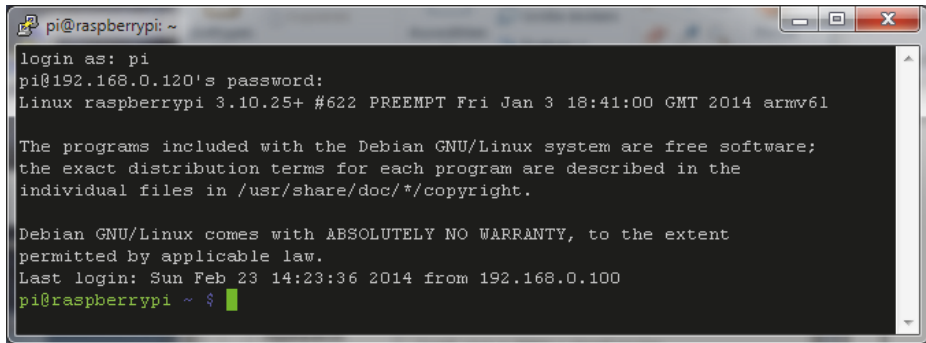


Abb. 2.22: Die Anmeldung über SSH

In der geöffneten Konsole sehen Sie:

```
pi@raspberrypi ~ $
```

Darin ist *pi* der aktuelle Benutzer und *raspberrypi* der Netzwerkname. Die Tilde vor dem Dollar-Zeichen zeigt Ihnen an, dass Sie sich im Home-Verzeichnis des angemeldeten Benutzers befinden. Um zu testen, dass der Raspberry Pi auch mit uns reden will, geben Sie den Befehl `ls -l` ein. Als Ergebnis erhalten Sie den Inhalt des Home-Verzeichnisses in Listenform.

```
pi@raspberrypi ~ $ ls -l
```

Auf der Konsole geben Sie die Befehle immer direkt hinter dem Dollar-Zeichen ein. Aus diesem Grund beginnen alle Beispiele auch immer mit einem Dollar-Zeichen und zeigen Ihnen damit an, dass Sie sich auf der Kommandozeile befinden und als normaler Benutzer angemeldet sind. Wenn Sie gleich einen Texteditor öffnen und dort Text eingeben, fehlt das Dollar-Zeichen. Dies signalisiert Ihnen, dass der eingegebene Text nicht direkt ausgeführt wird, sondern nach der Eingabe durch das aufgerufene Programm verarbeitet wird, im Fall eines Texteditors also in einer Datei gespeichert wird. Um die Verbindung zu beenden, geben Sie auf der Kommandozeile den Befehl `exit` ein und bestätigen mit `[↵]`. Das PuTTY-Fenster wird daraufhin geschlossen.

```
pi@raspberrypi ~ $ exit
```

Mit SFTP auf den Raspberry Pi zugreifen

Eine weitere Variante, um sich mit dem Raspberry Pi zu verbinden, ist über das SFTP-Protokoll, das sich hervorragend zum Übertragen von Dateien eignet. Da SFTP über SSH zur Verfügung steht, können wir dies ohne zusätzliche Konfiguration auf dem Raspberry Pi nutzen. Auf der PC-Seite können Sie dazu WinSCP benutzen, das Sie von der folgenden Webseite downloaden können:

<http://winscp.net/eng/download.php>

Installieren und starten Sie WinSCP auf Ihrem PC. Im Anmeldefenster aktivieren Sie dann in dem linken Baummenü den Punkt **SITZUNG** und wählen als Übertragungsprotokoll SFTP. Alle anderen Angaben geben Sie wie bei PuTTY ein und speichern die Sitzung unter einem für Sie sprechenden Namen.

Wählen Sie jetzt im Menü **GESP. SITZUNGEN** die gespeicherte Anmeldung für den Raspberry Pi aus und aktivieren Sie die Schaltfläche **ANMELDEN**. Wenn die Anmeldung funktioniert hat, wird Ihnen ein zweigeteiltes Fenster angezeigt. Im linken Teil sehen Sie die Verzeichnisstruktur Ihres PC und im rechten Teil die Verzeichnisstruktur des Raspberry Pi. Durch Drag&Drop können Sie jetzt Dateien zwischen den beiden Rechnern austauschen.

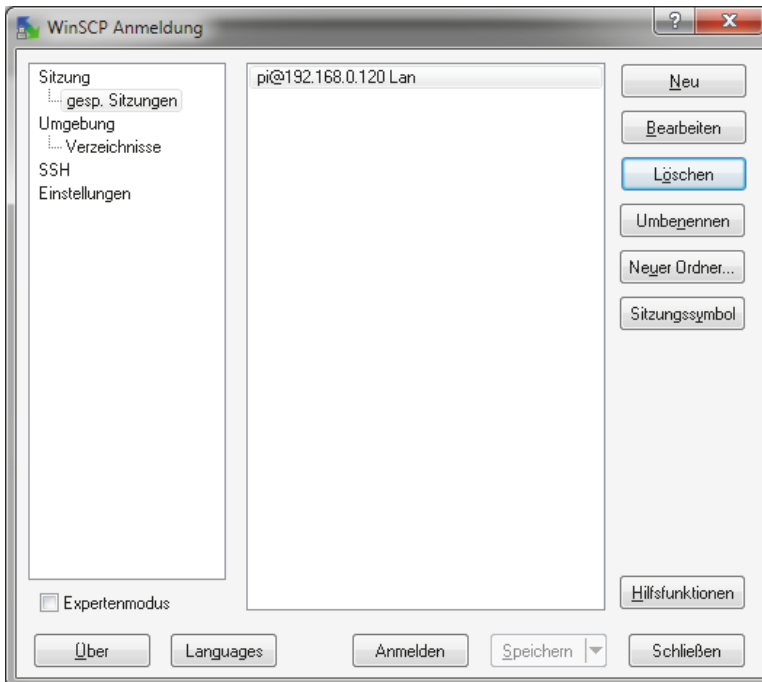


Abb. 2.23: WinSCP Anmeldung mit einer gespeicherten Sitzung

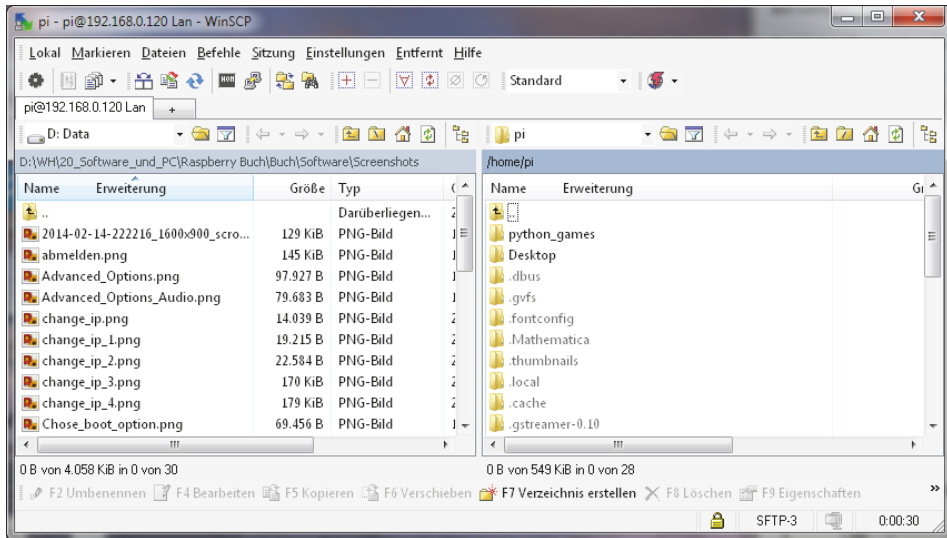


Abb. 2.24: In WinSCP können Sie Dateien mit Drag&Drop zwischen dem PC und dem Raspberry Pi austauschen.

Mit Virtual Network Computing (VNC) auf den Raspberry Pi zugreifen

Die dritte Variante, um sich von Ihrem PC aus mit dem Raspberry Pi zu verbinden, ist über eine Remote-Verbindung, mit der Sie den Desktop des Raspberry Pi in einem Fenster auf Ihrem PC angezeigt bekommen. In diesem Fenster können Sie dann so mit dem Desktop des Raspberry Pi arbeiten, als wären Sie direkt mit ihm verbunden. Die freie Software TightVNC kann eine solche Verbindung aufbauen und ist für Linux- und Windows-Systeme verfügbar. Um TightVNC nutzen zu können, müssen Sie es zuerst auf dem Raspberry Pi installieren. Da Sie die Software über die Konsole installieren müssen, benötigen Sie ein paar Linux-Befehle. Ich werde jeden Schritt kurz beschreiben, damit Sie die Software auch dann installieren können, wenn Sie mit dem Arbeiten auf der Linux-Konsole nicht geübt sind. Hierzu starten Sie zuerst PuTTY von Ihrem PC aus und melden sich an dem Raspberry Pi, wie weiter oben schon beschrieben, an. Nach dem Login befinden Sie sich im Home-Verzeichnis des Benutzers `pi`. Damit Sie die passende Software aus dem Internet herunterladen können, muss die Verbindung zum Internet stehen. Mit dem Befehl `apt-get` bietet Debian-Linux eine Benutzerschnittstelle, um Software-Pakete aus einem Repository im Internet herunterzuladen und zu installieren. Bevor die Software heruntergeladen wird, prüft die Paketverwaltung noch auf Abhängigkeiten zu anderen Modulen, die eventuell noch nachinstalliert werden müssen. Damit erhalten Sie eine sehr große Sicherheit, dass die installierte Software dann auch lauffähig ist. Um den TightVNC-Server aus dem Internet zu laden und auf dem Raspberry Pi zu installieren, geben Sie den folgenden Befehl ein:

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

Da Sie `apt-get` nur als Administrator ausführen können, muss `sudo` vor dem Befehl stehen. Bei der Installation erhalten Sie noch den Hinweis, wie viel Speicher benötigt wird. Dies bestätigen Sie mit `j` und warten, bis die Paketverwaltung alles für Sie erledigt hat.

Da der TightVNC-Server auf dem Raspberry Pi mit einigen Parametern gestartet werden muss, legen Sie sich ein Start-Script an. Um dieses zu erstellen, verwenden Sie den Texteditor *nano*. Geben Sie hierzu auf der Kommandozeile `nano` gefolgt von dem Namen der zu erstellenden Datei ein.

```
pi@raspberrypi ~ $ nano start_vnc.sh
```

Da `start_vnc.sh` noch nicht existiert, wird sie durch *nano* erzeugt. Später können Sie auf die gleiche Weise eine bestehende Datei öffnen. Geben Sie jetzt die folgenden zwei Zeilen in den Editor ein:

```
#!/bin/sh/
vncserver :1 -geometry 1280x800 -depth 24 -dpi 96
```

Mit `Strg`+`x`, `j` und `↵` speichern Sie die Datei und beenden *nano*. Sie haben den VNC-Server jetzt so parametrisiert, dass er die Ausgabe mit einer Auflösung 1280 x 1024 Pixeln, einer Farbtiefe von 24 bit und 96 dpi ausgibt. Die `1` hinter dem Doppelpunkt gibt den Ausgabekanal an, den wir später noch auf der PC-Seite benötigen. Mit `ls -l` überprüfen Sie noch, ob die Datei auch erzeugt worden ist.

```
pi@raspberrypi ~ $ ls -l
```

Dem Script fehlt jetzt aber noch die Berechtigung, ausgeführt werden zu dürfen. Dies können Sie daran erkennen, dass ganz links vor dem Dateinamen `start_vnc.sh` die Zeichenfolge `-rw-r--r--` zu sehen ist. Da die Zeichenfolge nicht mit einem `d` beginnt, erkennen Sie, dass es sich um eine Datei und nicht um ein Verzeichnis handelt. Danach folgen drei Dreier-Päckchen, die die Berechtigung auf die Datei angeben.

- `rw-`
Der Besitzer der Datei darf lesen (r) und schreiben (w).
- `r--`
Die Gruppe darf lesen (r).
- `r--`
Alle anderen Benutzer dürfen lesen (r).

Sie müssen die Berechtigung jetzt noch so erweitern, dass Sie als Besitzer der Datei diese auch ausführen dürfen. Geben Sie dazu die folgende Zeile ein.

```
pi@raspberrypi ~ $ chmod +x start_vnc.sh
pi@raspberrypi ~ $ ls -l
```

Wenn Sie sich jetzt die Berechtigung für die Datei anschauen, werden Sie sehen, dass der Besitzer, die Gruppe und alle anderen Benutzer nun das Recht zum Ausführen erhalten haben.

```
-rwxr-xr-x
```

Gekennzeichnet wird dies durch das x in jeder Berechtigungsstufe. Als Nächstes testen Sie, ob das Script auch läuft.

```
pi@raspberrypi ~ $ . start_vnc.sh
```

Sie müssen das Script mit einem vorangestellten Punkt und einem Leerzeichen aufrufen, damit kenntlich gemacht wird, dass die Datei im aktuellen Verzeichnis liegt. Wenn Sie das Script jetzt ausführen, werden Sie aufgefordert, ein Kennwort zu vergeben. Ich habe hier *desktop1* vergeben. Danach werden Sie noch gefragt, ob Sie noch ein *view-only*-Kennwort vergeben möchten. Dies habe ich mit ☐ quittiert.

Hinweis

Kennwort: *desktop1*; view-only: ☐

Beachten Sie bitte, dass das Kennwort nicht mehr als 8 Zeichen haben darf.

Nachdem Sie nun den VNC-Server erfolgreich über *start_vnc.sh* gestartet haben, brauchen Sie noch ein Script, um ihn zu stoppen. Dazu erzeugen Sie die Datei *stop_vnc.sh*.

```
pi@raspberrypi ~ $ nano stop_vnc.sh
```

Im Editor geben Sie dann die folgenden Zeilen ein

```
#!/bin/sh/  
vncserver -kill :1
```

und speichern mit ☐+☐, ☐ und ☐. Danach vergeben Sie die Rechte zum Ausführen der Datei und kontrollieren diese mit *ls -l*.

```
pi@raspberrypi ~ $ chmod +x stop_vnc.sh
```

Um den Server zu stoppen, geben Sie die folgende Zeile ein:

```
pi@raspberrypi ~ $ . stop_vnc.sh
```

Nachdem Sie auf dem Raspberry Pi alles für den TightVNC-Server installiert und konfiguriert haben, müssen Sie auf Ihrem PC noch einen TightVNC-Client installieren. Besuchen Sie dazu die folgende Webseite und downloaden Sie die entsprechenden Dateien.

```
http://www.tightvnc.com/?f=va
```

Für Windows-PCs liegen MSI-Pakete zur Installation bereit, die Sie auf Ihrem Rechner ausführen. Übernehmen Sie alle Standardeinstellungen bei der Installation und geben Sie für

den TightVNC-Server auch hier das Kennwort *desktop1* ein. Den Server benötigen wir allerdings nicht, um auf den Raspberry Pi zuzugreifen. Nach der Installation starten Sie den TightVNC-Viewer und geben in das Eingabefeld REMOTE HOST die folgende Verbindung ein und aktivieren die Schaltfläche CONNECT.

```
192.168.0.120::5901
```

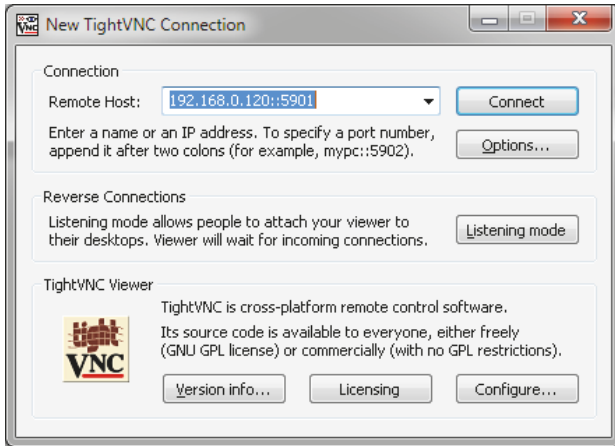


Abb. 2.25: Der TightVNC-Viewer auf Ihrem PC

Wenn auf dem Raspberry Pi der TightVNC-Server läuft und die IP-Adresse stimmt, öffnet sich ein Fenster, in dem die Verbindung angezeigt wird und in dem Sie das Kennwort des Tight-VNC-Servers eingeben müssen. Für das Kennwort geben Sie hier *desktop1* ein. Alles, was Sie in diesem Anmeldefenster sehen, bezieht sich auf die Einstellungen, die wir auf dem Raspberry Pi vorgenommen haben. So bezieht sich die 1 in 5901 auf den Wert hinter dem Doppelpunkt in dem Script *start_vnc.sh*, das Sie auf dem Raspberry Pi angelegt haben. Zur Verdeutlichung sehen Sie sich den Eintrag noch einmal an.

```
vncserver :1 -geometry 1280x800 -depth 24 -dpi 96
```

Die 5901 ist die Portnummer, über den die Tight-Verbindung läuft.

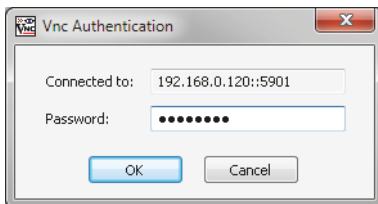


Abb. 2.26: Die Anmeldung am TightVNC-Server auf dem Raspberry Pi

Wenn Sie jetzt OK aktivieren, erscheint ein Fenster mit dem Desktop des Raspberry Pi, so wie Sie ihn über die direkte Verbindung schon kennen.

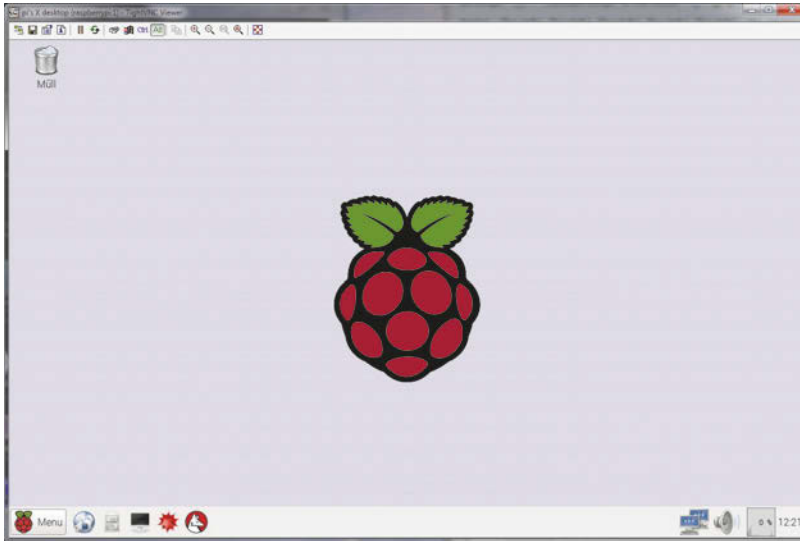


Abb. 2.27: Der Desktop des Raspberry Pi über VNC

Da Sie die SSH-Verbindung über PuTTY jetzt nicht mehr benötigen, können Sie das Fenster schließen. Über das SHUTDOWN-Icon auf dem Desktop lässt sich der Raspberry Pi herunterfahren. Es ist aber auch möglich, sich vom Desktop abzumelden und den TightVNC-Viewer zu schließen. Der TightVNC-Server auf dem Raspberry Pi ist davon nicht betroffen und arbeitet weiter. Zu einem späteren Zeitpunkt können Sie sich vom PC aus wieder mit dem TightVNC-Viewer an dem Raspberry Pi anmelden und auf dem Desktop weiterarbeiten. Zum Test können Sie mal den Browser des Raspberry Pi öffnen und das Fenster des TightVNC-Viewers schließen. Sie können sich dann wieder über VNC mit dem Raspberry Pi verbinden und werden sehen, dass der Desktop unverändert ist.

Wenn Sie sich noch eingehender mit dem VNC-Server auf dem Raspberry Pi beschäftigen möchten, finden Sie auf der eLinux-Webseite noch mehr Informationen.

http://elinux.org/RPi_VNC_Server

2.6 Der Raspberry Pi und das WLAN

Mit dem nächsten Schritt werden Sie auch noch das Netzkabel los und der Raspberry Pi wird nur noch an die Spannungsversorgung angeschlossen. Hierfür benötigen Sie einen USB-WLAN- bzw. USB-WiFi-Stick. Hier bietet die eLinux-Webseite eine Liste von getesteten Geräten an.

http://elinux.org/RPi_USB_Wi-Fi_Adapters

Bei der Auswahl des WLAN-Sticks sollten Sie darauf achten, dass Sie keinen Treiber installieren müssen und der Strombedarf des WLAN-Sticks mit allen angeschlossenen Geräten nicht höher als 300 mA sein darf. Wollen Sie große Reichweiten erzielen, braucht der WLAN-Stick eventuell mehr Strom, als der Raspberry Pi liefern kann. In diesem Fall können Sie den WLAN-Stick über einen USB-Hub mit eigener Stromversorgung anschließen und sind damit von der Stromversorgung des Raspberry Pi abgekoppelt.

Wenn Sie sich für einen WLAN-Stick entschieden haben, verbinden Sie diesen im stromlosen Zustand mit einem USB-Port des Raspberry Pi. Bei Raspberry-Pi-Modellen mit 4 USB-Ports können Sie die Konfiguration auch direkt über Tastatur und Maus ausführen. Wenn Sie aber nur zwei Ports frei haben, verbinden Sie sich von Ihrem PC aus über PuTTY mit dem Raspberry Pi. Auf der Kommandozeile starten Sie dann das Script `start_vnc.sh`. Denken Sie beim Aufruf des Scripts bitte an Punkt und Leerzeichen vor dem Script-Namen.

```
pi@raspberrypi ~ $ . start_vnc.sh
```

Jetzt können Sie sich von Ihrem PC aus über den TightVNC-Viewer mit dem Desktop des Raspberry Pi verbinden.

Bei Image-Versionen ab dem 05.05.2015 läuft die Konfiguration des Wireless-Netzwerks fast automatisch. Aktivieren Sie hierzu das NETZWERK-Icon in der Task-Leiste, um alle Wireless-Netzwerke angezeigt zu bekommen. Je nachdem müssen Sie etwas warten, bis alle verfügbaren Netze gescannt worden sind. Klicken Sie dann auf das gewünschte Netzwerk. Ist das Wireless-Netzwerk verschlüsselt, werden Sie über einen Dialog aufgefordert, den PRE SHARED KEY einzugeben. Diesen müssen Sie gegebenenfalls Ihrer Router-Konfiguration entnehmen.

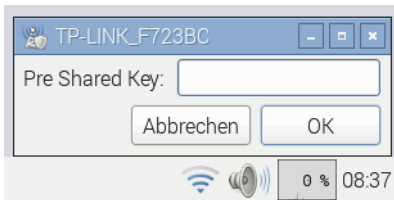


Abb. 2.28: Der Dialog zur Eingabe des PRE SHARED KEY

Wenn die Verbindung steht, können Sie den Raspberry Pi wieder herunterfahren und das Netzkabel entfernen. Nach dem Neustart ist der WLAN-Stick direkt aktiv.

Wenn Sie noch ein Rasbian-Wheezy-Image haben, das älter als 05.05.2015 ist, wird die Wireless-Konfiguration über die WPA-GUI ausgeführt. Diese starten Sie über das Icon WiFi CONFIG. Mit dieser Oberfläche können Sie nach dem WLAN-Router suchen und die WLAN-Verbindung konfigurieren. Wenn der WLAN-Stick erkannt worden ist, erscheint im Feld ADAPTER der Wert `wlan0`. Um die Suche nach dem WLAN-Router zu starten, aktivieren Sie die Schaltfläche SCAN.



Abb. 2.29: Die WPA-GUI

Nachdem Sie die Schaltfläche **SCAN** aktiviert haben, öffnet sich das Fenster **SCAN RESULTS**. Hier müssen Sie auch die Schaltfläche **SCAN** aktivieren, um den eigentlichen Suchlauf zu starten. Wurde der WLAN-Router gefunden, erhalten Sie einen entsprechenden Eintrag in der Liste **SCAN RESULTS**. In unserem Beispiel ist dies der TP-Link-Router.

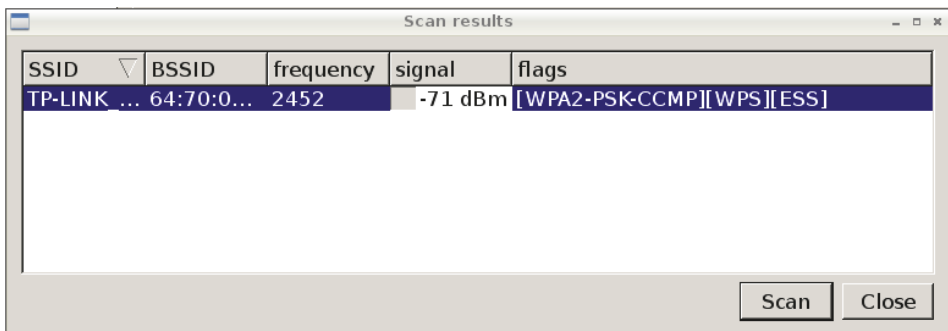


Abb. 2.30: Der Scan-Vorgang hat den TP-Link-Router gefunden.

Wenn Sie mehrere Router in der Liste angezeigt bekommen, wählen Sie Ihren Router mit einem Doppelklick aus der Liste aus. Danach erscheint das Konfigurations-Fenster, in das Sie noch das Kennwort eintragen müssen, das Sie im Router für WLAN-Verbindungen vergeben haben. Das Kennwort tragen Sie in das Feld **PSK** ein und aktivieren die Verbindung über **ADD**.

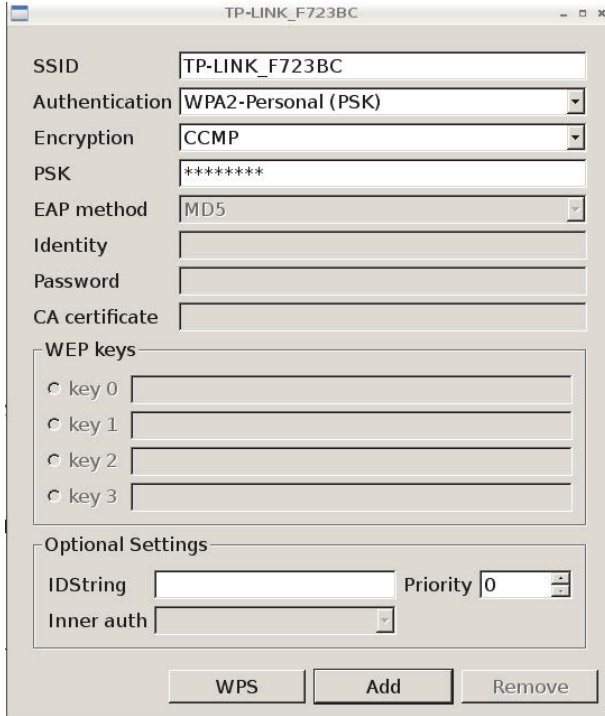


Abb. 2.31: Das Konfigurationsfenster für die WLAN-Verbindung

Wenn Sie das richtige Kennwort eingetragen haben, sehen Sie die aktive Verbindung im Fenster WPA-GUI. Unter IP ADDRESS wird die IP-Adresse angezeigt, die der Router dem Raspberry Pi für die WLAN-Verbindung zugewiesen hat.



Abb. 2.32: Tab-Reiter: CURRENT STATUS

In dem Tab-Reiter **MANAGE NETWORKS** können Sie mehrere Router verwalten. Sie können auch einen Scan-Vorgang anstoßen und verschiedene Router-Konfigurationen bearbeiten und freigeben.

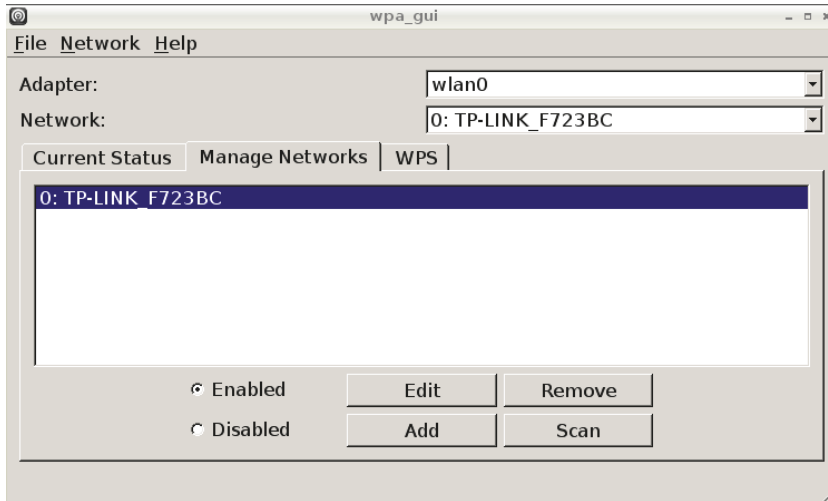


Abb. 2.33: Tab-Reiter: **MANAGE NETWORKS**

Über den Tab-Reiter **WPS** (Wi-Fi Protected Setup) können Sie die WPS-Funktion nutzen, um auf einfache Art eine Verbindung zu einem WiFi-Gerät aufzunehmen. Dabei wird im Router eine PIN erzeugt oder bestimmt, die von dem WiFi-Gerät aufgenommen werden muss, um eine Verbindung mit dem Router herstellen zu können.

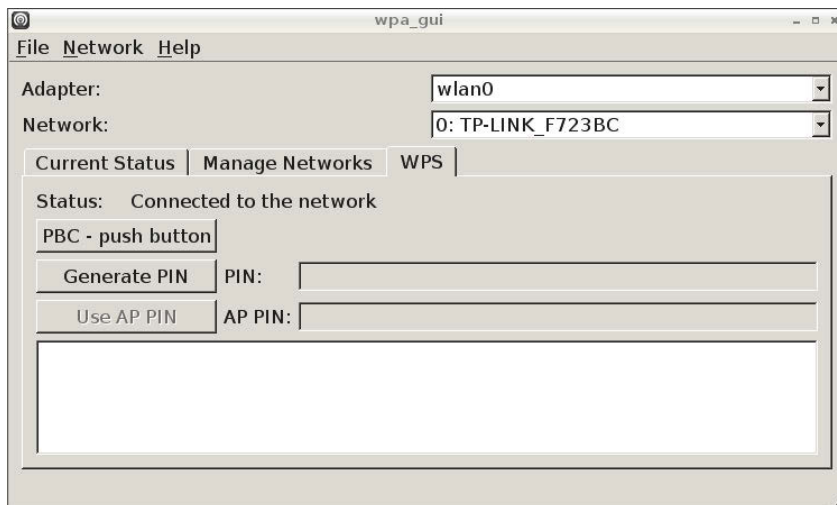


Abb. 2.34: Tab-Reiter: **WPS**

Als letzten Schritt fahren Sie den Raspberry Pi herunter und trennen die Stromversorgung. Entfernen Sie jetzt das Netzkabel und aktivieren Sie dann wieder die Stromversorgung des Raspberry Pi, damit er wieder startet. Wenn Sie Glück haben, vergibt der Router dieselbe IP-Adresse, wie Sie sie in der WPA-GUI angezeigt bekommen haben. Wenn nicht, müssen Sie im Router-Setup unter der DHCP-Client-Liste nach dem Raspberry Pi suchen, um die aktuell zugeteilte IP-Adresse herauszufinden. Der Verbindungsaufbau zum Desktop des Raspberry Pi ist genau so, wie Sie es schon über das Netzwerk ausgeführt haben.

Nachdem die WLAN-Verbindung steht, sehen Sie sich mit dem Linux-Befehl `less` die Ihnen schon bekannte Datei `interfaces` aus dem Verzeichnis `/etc/network` an. Mit `less` kann man Dateien anschauen und sie seitenweise durchblättern.

```
pi@raspberrypi ~ $ less /etc/network/interfaces
```

Um sich die Datei anzusehen, können Sie sich auch über PuTTY mit dem Raspberry Pi verbinden. In der Datei `interfaces` hat die WPA-GUI drei neue Zeilen hinzugefügt. Vielleicht werden Sie Parameter wie das Kennwort (PSK) und die Authentifizierung in der Datei vermissen. Diese hat die WPA-GUI in die Datei `wpa_supplicant.conf` eingetragen und den Verweis in `interfaces` geschrieben.

Alle Einträge, die zu dem WLAN-Adapter `wlan0` gehören, sollten nur über die WPA-GUI geändert werden. Es könnte sonst zu Komplikationen kommen, wenn manuelle Änderungen von der WPA-GUI nicht richtig erkannt werden oder bei einem späteren Aufruf des Programms die manuellen Änderungen überschrieben werden. Aus diesem Grund wollen wir hier auch auf das Einstellen einer statischen IP-Adresse für die WLAN-Verbindung verzichten und stattdessen im Router die MAC-Adresse des WLAN-Adapters einer IP-Adresse zuweisen.

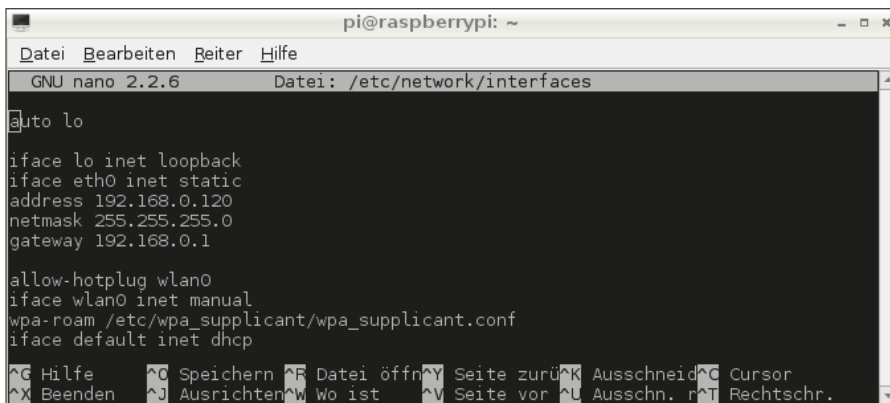


Abb. 2.35: Netzwerkkonfiguration in `/etc/network/interfaces`

2.7 Ein Backup

Nachdem Sie den Raspberry Pi jetzt fertig konfiguriert haben, sollten Sie ein Backup auf Ihrem PC sichern. Mit diesem Backup können Sie später immer wieder einen Grund-

zustand herstellen, in dem der Raspberry Pi lauffähig ist. Um das Backup anzulegen, fahren Sie den Raspberry Pi hinunter und trennen die Stromversorgung. Sie können jetzt die SD-Karte aus dem Raspberry Pi herausziehen und in den Kartenleser des PC einstecken. Starten Sie dann den Win32-Disk-Imager und tragen Pfad und Dateinamen in das Feld IMAGE FILE ein. Sie können auch über die Schaltfläche rechts neben dem Feld IMAGE FILE zu dem gewünschten Verzeichnis navigieren und einen Namen eintragen. Achten Sie bitte darauf, dass die Dateiendung immer `img` ist. Wenn Sie jetzt die Schaltfläche `READ` aktivieren, wird das Image von der SD-Karte gelesen und auf Ihrem PC gesichert.

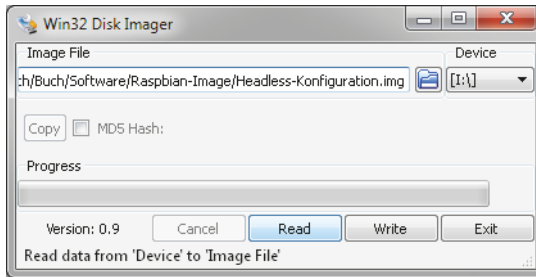


Abb. 2.36: Backup mit dem Win32 Disk Imager