

#makers
DO IT.

Inkl. Einsatz
von WiringPi,
ALSA & OpenCV



Raspberry Pi

Harald Schmidt

RASP- BERRY PI

programmieren mit
C/C++ und **BASH**

Mehr als 50 Programme rund um
Foto, Video & Audio

HANSER

Harald Schmidt

Raspberry Pi programmieren mit C/C++ und Bash



BLEIBEN SIE AUF DEM LAUFENDEN!

Hanser Newsletter informieren Sie regelmäßig über neue Bücher und Termine aus den verschiedenen Bereichen der Technik. Profitieren Sie auch von Gewinnspielen und exklusiven Leseproben. Gleich anmelden unter

www.hanser-fachbuch.de/newsletter

Harald Schmidt

Raspberry Pi programmieren mit C/C++ und Bash

Mehr als 50 Programme rund um
Foto, Video & Audio

HANSER

Der Autor:

Harald Schmidt, München

Alle in diesem Buch enthaltenen Informationen wurden nach bestem Wissen zusammengestellt und mit Sorgfalt getestet. Dennoch sind Fehler nicht ganz auszuschließen. Aus diesem Grund sind die im vorliegenden Buch enthaltenen Informationen mit keiner Verpflichtung oder Garantie irgendeiner Art verbunden. Autor und Verlag übernehmen infolgedessen keine Verantwortung und werden keine daraus folgende oder sonstige Haftung übernehmen, die auf irgendeine Weise aus der Benutzung dieser Informationen – oder Teilen davon – entsteht, auch nicht für die Verletzung von Patentrechten, die daraus resultieren können.

Ebenso wenig übernehmen Autor und Verlag die Gewähr dafür, dass die beschriebenen Verfahren usw. frei von Schutzrechten Dritter sind. Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt also 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 benützt werden dürften.

Bibliografische Information der deutschen Nationalbibliothek:

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

Dieses Werk ist urheberrechtlich geschützt.

Alle Rechte, auch die der Übersetzung, des Nachdruckes und der Vervielfältigung des Buches, oder Teilen daraus, vorbehalten. Kein Teil des Werkes darf ohne schriftliche Genehmigung des Verlages in irgendeiner Form (Fotokopie, Mikrofilm oder ein anderes Verfahren), auch nicht für Zwecke der Unterrichtsgestaltung, reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

ISBN 978-3-446-45342-5

E-Book-ISBN 978-3-446-45349-4



© 2018 Carl Hanser Verlag, München

Lektorat: Julia Stepp

Herstellung: Isabell Eschenberg

Umschlagrealisation: Stephan Rönigk

Satz: Kösel Media GmbH, Krugzell

Druck und Bindung: Druckerei Hubert & Co. GmbH und Co. KG, Göttingen

Printed in Germany

www.hanser-fachbuch.de

Inhaltsverzeichnis

Vorwort	XVII
1 Die Welt des Raspberry Pi	1
1.1 Die Modelle des Raspberry Pi	1
1.1.1 Energieversorgung	5
1.1.2 Der Raspberry Pi Zero W	6
1.1.3 Raspberry Pi 2 und Pi 3	10
1.2 SD-Karten und microSD-Karten	14
1.3 Von Wheezy zu Stretch – Downloads für Raspbian	15
1.3.1 Die Release Notes als Chronik	16
1.4 Über NOOBS zur Raspbian SD-Karte	17
1.4.1 NOOBS aus dem Internet holen und entpacken	18
1.4.2 NOOBS auf die leere (micro)SD-Karte kopieren	20
1.5 NOOBS auf dem Raspberry Pi	21
1.5.1 Wohin mit dem Bild?	21
1.5.2 Energie für NOOBS	22
1.6 Der NOOBS Recovery-Modus	28
2 Der Raspberry Pi und Linux	31
2.1 Das Abbild von Raspbian für die SD-Karte	32
2.1.1 Die Abbilddatei mit einem Linux-PC übertragen	36
2.1.2 Der Win32 Disk Imager	36
2.2 Die Grundkonfiguration des Raspbian Desktops	38
2.2.1 Tastaturlayout und Spracheinstellung	39
2.2.2 Konfiguration von Autologin, Hostname und Bootmodus	47
2.2.3 Passwortänderung	48
2.2.4 Freigabe für SSH und Abfrage der IP-Adresse	49

2.3	Der Raspberry Pi Desktop als Bedienoberfläche	50
2.3.1	Die Lage der Panelleiste ändern	51
2.3.2	Der Aufbau der Panelleiste	51
2.3.3	Programmstart über Tastaturkürzel	52
2.3.4	Eine symbolische Desktop-Verknüpfung hinzufügen	53
2.3.5	Eine neue Desktop-Datei erzeugen	54
2.3.6	Ein Programm der Anwendungsstartleiste hinzufügen	56
2.3.7	Miniprogramme im Panel	57
2.3.8	Das Konsolenkommando lxpanelctl	59
2.4	Der Main Menu Editor Alacarte	59
2.5	Das Erscheinungsbild anpassen mit LxAppearance	60
2.6	Der Raspbian Desktop auf dem PC oder Mac	62
2.7	Nützliche Programme ohne und mit grafischer Oberfläche	64
2.8	Autostart mit und ohne GUI	71
2.8.1	Die versteckte Datei .bashrc	72
2.8.2	Die Datei rc.local	74
2.9	Die Verwendung von LXTerminal in Desktop-Dateien	76
2.10	Pakete - Installation und Verwaltung	78
2.10.1	Der vorinstallierte Paketmanager pi-packages	79
2.10.2	Der Paketmanager Synaptic	80
2.10.3	Paketverwaltung über Konsolenkommandos	81
2.11	Die Bildschirmauflösung mit und ohne den RPi-Desktop einstellen	89
2.11.1	Über HDMI zum Fernseher	93
2.11.2	Über den HDMI/VGA-Adapter zum Monitor	94
2.11.3	Einstellung der Auflösung mit raspi-config	96
2.12	Audiumschialtung und Bluetooth-Lautsprecher	99
2.13	Systembackup im laufenden Betrieb	101
3	Betriebssysteme, Partitionen, Backup und Restore	103
3.1	SD-Karte, USB-Stick, Festplatte - was erkennt der Raspberry Pi?	105
3.2	SD-Karten mit Betriebssystem	108
3.2.1	Hinein in die Box	108
3.2.2	Der passende Aufkleber	109
3.2.3	Binäre Einheiten und die Kapazität von SD-Karten	109
3.3	SD-Karten identifizieren mit sd-label.sh	110
3.4	Hinweise zu Backup und Restore	113
3.5	Formatierung von SD-Karten	115
3.6	Das USB Image Tool	117
3.7	Direktkopie der SD-Karte mit piclone	119

3.8	Der Partitionsmanager GParted	121
3.9	Eine bootfähige SD-Karte mit sd-restore.sh erzeugen	124
3.10	Backup von SD-Karten mit sd-back.sh	128
3.10.1	Nullen vor dem Backup	132
3.11	Ein Blick in das Abbild der SD-Karte	134
3.12	Festplatten am Raspberry Pi	137
3.12.1	Der Einhängepunkt und die Systemkonsole	140
3.12.2	Einhängen der Partition über /etc/fstab	141
4	Programmieren mit dem Pi	143
4.1	Dateien und Programme zum Buch	144
4.2	Paketinstallation über die Skriptdateien zum Buch	146
4.3	Die Skriptdateien zum Buch – was ist wo?	148
4.4	Die Shell als Kommandointerpreter	150
4.4.1	Umgebungsvariablen	153
4.4.2	vcgenclmd	154
4.5	Strings in Bash, C und C++	156
4.6	Programmieren mit Bash	157
4.6.1	Eingabe, Ausgabe und if else	157
4.6.2	Feldtrenner und float-Werte	160
4.6.3	Formatierte Ausgabe mit printfcalc.sh	162
4.6.4	Ausgabekommandos für ASCII-Dateien	162
4.6.5	Bash-Arrays mit numerischem Index	163
4.6.6	Assoziative Bash-Arrays mit Wortindex	165
4.6.7	Funktionen, Parameter und Exit-Codes	166
4.6.8	Ausgabeumleitung und Rückgabewerte	167
4.6.9	Ein Bash-Skript mit Funktionen und Indexsuche	170
4.6.10	Indezelemente sortieren und finden mit assosort.sh	173
4.6.11	Ausgabespalten neu anordnen mit dosdir.sh	174
4.6.12	Die Ausgabe Einlesen und die IFS-Trennzeichen	176
4.6.13	Filtern mit cut und grep	179
4.6.14	Farben in der Konsolenausgabe	186
4.6.15	Zeichen entfernen mit trim.sh	188
4.6.16	Dateinamen oder Pfade extrahieren	189
4.6.17	Die Position einer Zeichengruppe oder eines Strings finden	189
4.6.18	Ein Blick in raspi-config	191
4.7	Bash in Beispielen	193
4.7.1	Wie copydesk.sh funktioniert	193
4.7.2	Farbige Listings mit colordir.sh	194

4.7.3	Analyse von Partitionsgrößen mit sd-info.sh	196
4.7.4	Feldinhalte aus Audiodateien kompakt anzeigen	199
4.7.5	EXIF-Daten anzeigen mit jhead.sh	201
4.7.6	Metadaten ermitteln mit mediainfo.sh	204
4.7.7	Diagnose der Bildschirmauflösung mit scrreso.sh	208
4.7.8	Eine Bildschirmecke zur Videowiedergabe nutzen (OMXPlayer) ...	210
4.7.9	Prozesse anzeigen mit pgrep_htop.sh	212
4.8	Skriptdateien mit ASCII-GUI (whiptail/dialog)	217
4.8.1	Vom Skript zum Kommando	221
4.8.2	Eine allgemeingültige Dateiauswahl im Skript verwenden	223
4.8.3	Programmauswahl über Checkboxes	225
4.8.4	Bilder als ASCII-Art anzeigen	228
4.8.5	Programmstart mit fotomenu.sh	234
4.8.6	Programmstart über Desktop-Dateien und ASCII-Dialoge	237
4.8.7	Dateiauswahl bei sd-mount.sh	240
4.8.8	Der Programmstarter opencvgui.sh als ASCII-GUI	243
4.9	C-Programme mit Raspbian nutzen (hello_pi/GPU)	245
4.9.1	Videoprojektion auf eine rotierende Teekanne (hello_teapot.c) ...	247
4.9.2	Mathematische Kunst mit hello_triangle2.c	248
4.9.3	Das Skript helloGPU.sh auf einem „großen“ Linux-PC starten ...	249
4.9.4	Mit make zum eigenen C-Programm	250
4.9.5	Das Skript helloGPU2.sh für weitere GPU-Programme	253
4.9.6	Programme starten mit system und popen	254
4.9.7	Zeit ist relativ	257
4.9.8	Mausklicks, Bewegungen und Scrollevents auswerten	259
4.9.9	Von der Mausbewegung zur Audioausgabe	264
4.10	Der Qt Creator als Entwicklungsumgebung	268
4.11	Die Entwicklungsumgebung Code::Blocks	273
4.12	GTK+ Programmierung	276
4.13	Die Geany-Entwicklungsumgebung	279
4.14	Python und IDLE als Entwicklungsumgebung	282
4.15	Zenity-Dialoge	286
4.16	GitHub und Raspbian	287
5	Elektronik und Programme für den GPIO-Port	289
5.1	Schaltungen aufbauen	292
5.1.1	Drähte oder Drahtbrücken	295
5.1.2	SMD-Adapter und ICs	297
5.1.3	Widerstände, Widerstandsnetzwerke und Kondensatoren	298
5.1.4	Der LED-Test	300

5.2	Eine LED per Software schalten (Ein/Aus)	301
5.3	Eine LED per Software dimmen (Pulsweitenmodulation)	304
5.4	Eine LED am Optokoppler	306
5.5	GPIO-Pins im Read-Modus (Bash)	307
5.5.1	GPIO-Pins zyklisch abfragen (Polling)	308
5.5.2	GPIO-Pins im Interrupt-Modus (Flankenerkennung)	309
5.5.3	Flankenerkennung bei mehreren Schaltern	312
5.6	WiringPi-Beispiele mit C-Quelltext	315
5.6.1	WiringPi im Interrupt-Modus	316
5.7	Experimente zur Impulswahl	317
5.7.1	Audiodateien durch Impulswahl abspielen	321
5.7.2	Impulsdiagramme mit GnuPlot	322
5.8	Von der Helligkeit zur Frequenz	325
5.8.1	Frequenzen messen mit dem Oszilloskop und mit r2f.cpp	326
5.9	Stufenschalter, Tastenfelder, Encoder und Codierschalter	331
5.10	Analog/Digital-Wandler für die SPI-Schnittstelle	333
5.10.1	Analoge Messwerte mit spiloop.c erfassen	336
5.11	Die I ² C-Schnittstelle	338
5.12	Analog/Digital-Wandler für I ² C	339
5.13	Ein LCD-Display am I ² C-Bus	344
5.13.1	Ausgaben mit sysinfo.sh	349
5.13.2	PIDs auf dem LCD-Display mit pgrep_menuctrl.sh	351
5.14	Die C-Library bcm2835 für den GPIO-Port	353
5.15	Mehr Ampere mit externen Spannungsquellen	355
5.16	Vom Schaltplan zum Layout	357
5.16.1	EAGLE Light	358
6	Sensoren und Software für Infrarot und Ultraschall	363
6.1	LIRC als Empfänger	364
6.1.1	Kommandos per IR-Fernbedienung auslösen	370
6.1.2	Ein Auswahlmenü auf dem LCD-Display	373
6.2	Entfernungsmessung per Ultraschall	376
6.3	Ein Abstandswarner mit ultraloop.c im Qt Creator	381
6.4	Entfernungsmessung mit Infrarotlicht	386
6.4.1	Distanzmessung über den MCP3426 als A/D-Wandler	388

7	Netzwerke und der Fernzugriff auf den Raspberry Pi	389
7.1	Die automatische Vergabe von IP-Adressen	392
7.2	Netzwerkkabel, Übertragungsraten und ethtool	401
7.3	Eine Direktverbindung zum Raspberry Pi	402
7.3.1	Die Link-Local-IP	406
7.4	Netzwerkkameras	407
7.4.1	Eine Direktverbindung für Netzwerkkameras	409
7.5	Der SSH-Server und ein Linux-PC als Client	410
7.5.1	SSH und die Weiterleitung des X11-Fensters	415
7.5.2	Fingerabdrücke auf (micro)SD-Karten	417
7.6	Zwei Schlüssel zum Vergleich – SSH ohne Passwort	418
7.7	PuTTY als Windows-Client	420
7.8	Remote-Zugriff mit WinSCP und Krusader als Client	424
7.9	Der SSH-Fernzugriff per ASCII-GUI	428
7.9.1	Ein Desktop-Icon auf dem Client-PC	428
7.10	Ein Remote-Desktop mit TightVNC	429
7.11	Energie sparen mit TMux	432
7.12	Die ASCII-GUI pgrep_gui.sh	437
7.13	TMux und die ASCII-GUI pgrep_gui.sh	441
7.13.1	TMux-Skripting	443
7.14	WLAN-Verbindungen und WLAN-Sticks	445
7.14.1	WLAN-Verbindungen und wpa_gui	449
7.15	Die Netzwerk-Tools arp-scan und nmap	451
7.16	Netcat – vom Chatprogramm zum Videoempfänger	453
7.17	Fünf Wege zum Raspberry Pi	454
7.18	Über Android zum Raspberry Pi	456
7.18.1	Über F-Droid zu ConnectBot	457
7.19	Über USB OTG zum Raspberry Pi Zero	460
8	Audio und Composite Video am Pi	463
8.1	Buchsen, Kabel, Signale	463
8.2	Audiowiedergabe	467
8.3	Der ALSaMixer	474
8.4	Audioaufzeichnung und ALSA-Tools	476
8.4.1	SoX mit der Webcam als Audioquelle	479
8.4.2	SoX passt auf – Reaktion auf Geräusche	485
8.4.3	Informationen zur Audiodatei mit soxi	488
8.4.4	Audioaufzeichnung im Remote-Modus	489
8.4.5	Lautsprecherdurchsagen aus sicherem Abstand (remote)	492
8.4.6	Externe USB-Soundkarten und das Programm speaker-test	493

8.5	Sprachsynthese auf dem Linux-PC	496
8.5.1	Sprachausgabe mit eSpeak	497
8.5.2	Sprachausgabe über tts.sh mit eSpeak, SVOX-Pico und Festival ...	499
9	Videodaten – Aufzeichnung und Wiedergabe	503
9.1	Der OMXPlayer	504
9.1.1	Kommandos für den OMXPlayer (DBUS-Interface)	506
9.2	Videodateien aus dem Internet holen	508
9.3	Archivierte Videos abspielen mit guiPlay.sh	512
9.4	Videoplayer und multiplay.sh	520
9.5	Wiedergabe mit videoselect.sh und avplay	524
9.6	Wo ist das Videogerät?	525
9.7	Daten zum Videogerät mit dem V4L-Treiber	528
9.8	MPEG-Spezifikationen, Codecs und Containerdateien	531
9.9	Fotografieren und Filmen mit guvcview	532
9.10	Fotografieren mit fswebcam	536
9.11	Die libav-tools und avconv (bzw. ffmpeg)	542
9.11.1	Der Webcam-Zugriff mit avconv	543
9.11.2	Vertonung mit avconv	545
9.11.3	Bildschirmvideos mit dem Raspberry Pi	546
9.11.4	Videos konvertieren mit WinFF	548
9.12	Motion auf dem Pi	550
9.12.1	Der Programmstarter motiongui.sh	553
9.12.2	Der Blick auf die Uhr	558
9.12.3	Motion-Parameter geeignet festlegen	560
9.12.4	Namen für Bilder und Videodateien	566
9.12.5	Einblendung von Zeitstempeln und Texten	567
9.12.6	Markierte Bewegungen	568
9.12.7	Bewegungsalarm auf Teilbereichen des Bildes	569
9.12.8	Automatische E-Mails bei Ereignissen	572
9.12.9	Den Videostream anzeigen und weiterleiten	573
9.12.10	Parameteränderung per ASCII-GUI und setvalue.sh	576
10	Kameramodule zum Raspberry Pi	577
10.1	Das 8-Megapixel-Kameramodul mit Sony-IMX219-Sensor	581
10.2	Fotos mit raspistill	583
10.2.1	EXIF-Daten ermitteln	588
10.2.2	Fotografieren mit fotomenu.sh	589
10.2.3	Der manuelle Belichtungsmodus	591

10.2.4	Fotografieren mit Zeit-/ISO-Automatik	593
10.2.5	Farben und Spezialeffekte	596
10.2.6	Fotos bei sehr wenig Licht und im Signalmodus	600
10.3	Konverter für Einzelbilder	604
10.4	Bilder und das FBI	605
10.5	Videos aufzeichnen mit raspivid	608
10.6	Videodateien mit MP4Box und avconv verarbeiten	611
10.7	Der UV4L-Treiber zum Kameramodul des Raspberry Pi	613
10.7.1	Fotografieren mit multicam.sh	617
10.7.2	Filmen mit dd	621
10.8	Binning bei raspistill und raspivid	622
10.9	Motion und das Kameramodul	625
11	Automatischer Datentransfer	631
11.1	SFTP-Kommandos per Skript ausführen	632
11.2	Eine RAM-Disk als Zwischenpuffer	636
11.2.1	Bilder auf der RAM-Disk	637
11.2.2	Die RAM-Disk überwachen	639
11.3	Datentransfer mit FileZilla	641
12	Audiosignale analysieren	643
12.1	Audacity	645
12.2	Die Verwendung von libsndfile	647
12.2.1	Die Abtastwerte einer Audiodatei in Textform	649
12.2.2	Ein Blick auf die Dokumentation	650
12.2.3	Ausgabeumleitung für Audiodaten mit sndout.cpp	651
12.3	Pausenerkennung und Segmentierung	655
12.3.1	Audiosegmentierung und Sprachausgabe mit audiosegm.cpp	656
12.4	Die Library tinyalsa	661
12.4.1	Abtastraten mit tinypcm_info.c ermitteln	663
12.4.2	Audioaufnahmen mit capaudio.sh und tinycap.c	664
12.5	Der Pi als Papagei	668
12.6	Spektrogramme mit sndfile-tools und Audacity	670
12.7	Vokaltraining mit voxvokal.sh	679
12.8	Das Audioquiz	683

13	OpenCV und der Pi	687
13.1	Die Konfiguration von OpenCV	688
13.2	Mit dem Compiler zur ausführbaren Datei	691
13.3	OpenCV – Dokumentation und Beispielprogramme	695
13.4	OpenCV, CMakeGUI und Code::Blocks	698
13.5	Die Qt-Projektdateien zu den Beispielprogrammen	700
13.6	Das Programm fpsec.cpp auf dem Pi benutzen	703
	13.6.1 Fenster und Tastaturcodes von fpsec.cpp	704
	13.6.2 Konturbilder vom Webcambild	705
	13.6.3 Konturbilder vom Videobild	706
	13.6.4 FloodFill auf dem Videobild	707
13.7	Programmierung mit OpenCV	708
	13.7.1 Grundlegende Datentypen und OpenCV-Klassen	708
	13.7.2 Fenster, Steuerelemente und cv::waitKey	710
	13.7.3 Optimierung von Farb- und Grauwertbildern per Tastendruck ...	711
	13.7.4 Binarisierung von Bildern mit cv::threshold	714
	13.7.5 Regionen und Beschriftungen	716
13.8	Konfiguration für die WITH_QT-Oberfläche	717
14	Objektsegmentierung mit OpenCV	719
14.1	Segmentierung von Textzeilen	719
14.2	Symbolerkennung mit iconreader.cpp	722
14.3	Symbolerkennung auf Webcambildern	725
	14.3.1 Einsatzzweck, Bedienung und Tastaturcodes	727
	14.3.2 Vom Graustufenbild zum Erkennungsergebnis	732
	14.3.3 Der Alarm und die Statistik	738
14.4	Die sprechende Lichtschranke	740
15	Experimente	741
15.1	Booten mit Sprachausgabe	741
15.2	Ein Eingabemodul mit IR-Detektor und LCD-Display	744
	15.2.1 Den Schaltungsaufbau planen	747
	15.2.2 Lötarbeiten für die Elektronik zum DOGM204 LCD-Display	749
	15.2.3 Das LCD-Display testen	750
	15.2.4 Das Eingabemodul eichen	752
	15.2.5 Die Drehung des Codierschalters auswerten	755
15.3	Ein Client/Server-Konzept für das Eingabemodul	756
	15.3.1 Nur mit Maus und Display?	762

15.4	Ein Client/Server-Konzept mit menuctrl.sh	765
15.4.1	Fotos mit MenuCtrl	770
15.5	Über MenuCtrl fswebcam starten/beenden	773
15.6	Die hello_pi GPU-Programme mit MenuCtrl starten/beenden	774
15.7	MenuCtrl im Infrarotlicht	776
15.8	MenuCtrl schon beim Booten starten	781
16	Schlusswort	783
17	Übersicht der Programme und Skriptdateien zum Buch	785
	Stichwortverzeichnis	791

Meinem Patenonkel Josef Schmidt gewidmet

Vorwort

Der Raspberry Pi ist zum Computer mit Kultstatus geworden. Das mag einerseits an der umfangreichen Literatur liegen, die für dieses Gerät verfügbar ist. Ganz sicher ist dafür aber auch das von der Raspberry Pi Foundation getragene Konzept verantwortlich, auch die Hardware als Open Source zu betrachten. So konnte ein kleiner Computer entstehen, der sich als Steuerungscomputer für Elektronikbastler einsetzen lässt und dabei nur wenig Energie verbraucht.

Softwarebasis des Pi sind Betriebssysteme, die ihre Wurzeln in Linux haben. Am bekanntesten ist dabei vielleicht Raspbian, das aus einer Debian-Linux-Distribution entwickelt wurde. Des- sen Bedienoberfläche ist einfach aufgebaut und ähnelt der klassischen Oberfläche von Win- dows XP.

Pi ist aber auch ein Benutzername (**pi@raspi3**) – und mit dessen Augen werden Sie in diesem Buch spannende Blicke auf Ihren Mini-PC und seine Umgebung werfen. Dieses Buch führt Sie zugleich auch in die Welt zweier recht unterschiedlicher Programmiersprachen ein:

- Bash: für Ablaufsteuerungen und als Programmstarter mit Benutzeroberfläche
- C/C++: für alle Fälle, in denen man tiefer einsteigen will und in denen Geschwindigkeit wichtig ist

Sie werden noch sehen, dass sich diese Programmiersprachen nicht widersprechen, sondern wunderbar ergänzen, denn letztlich ist auch Linux eine Kombination daraus.

Da wir Menschen hauptsächlich über das Sehen, Hören und Sprechen kommunizieren, ist es angemessen, auch einen Computer in dieser Hinsicht zu erweitern. Der Schwerpunkt der für dieses Buch erstellten Programme liegt deshalb in diesen Bereichen.

Was erwartet Sie in diesem Buch?

Kapitel 1 erklärt, welche Modelle des Raspberry Pi existieren und wie Sie Raspbian Jessie/ Stretch konfigurieren und erstmals starten. Zahlreiche Verweise auf andere Kapitel und Abschnitte erleichtern Ihnen den Einstieg.

Kapitel 2 bringt Ihnen Linux und Raspbian näher. Raspbian ist derzeit das beliebteste Betriebs- system auf dem Raspberry Pi. Sie lernen viele Details der grafischen Oberfläche kennen, aber auch die Verwendung der Konsole und vieler interessanter Programme werden erklärt.

Kapitel 3 erklärt den praktischen Umgang mit der (micro)SD-Karte als Boot-Medium des Raspberry Pi. Sie erfahren einiges über Backup und Restore und wie einfach es ist, in die „Computerproduktion“ einzusteigen, denn Raspbian Jessie hat mit PiClone ein hübsches Backupprogramm bekommen.

Kapitel 4 erklärt die Verwendung vieler Linux-Kommandos. Sie erhalten eine Einführung in Bash, welche Sie dazu befähigt, kleine effektive Werkzeuge bzw. Skriptdateien zu bauen. Selbst wenn Sie daran nur begrenzt Interesse haben sollten, erfahren Sie, wie Sie viele der Programme zum Buch durch Anklicken von Desktop-Icons ganz einfach starten können. Mit Bash können Sie sehr leicht ASCII-Dialoge aufbauen, die als einfacher und informativer Programmstarter verwendbar sind. Auch C/C++ Programme lassen sich so ganz ohne Vorkenntnisse über Compiler starten. In diesem Kapitel lernen Sie außerdem die GPU-Programme von Raspbian kennen – von der rotierenden Teekanne bis zum mathematischen Kunstwerk.

Kapitel 5 führt Sie in die Welt der Elektronik ein. Die Namen Faraday, Ohm und Tesla sollten Ihnen bekannt sein. Sie müssen aber kein Elektronikexperte sein, um funktionsfähige Schaltungen aufzubauen und anzusteuern. Lassen Sie eine LED blinken. Verwenden Sie einen lichtgesteuerten Widerstand, Optokoppler oder eine antike Wählscheibe. Nutzen Sie einen A/D-Wandler und die zugehörige Software am Raspberry Pi. Auch ein LCD-Display lässt sich verwenden.

Kapitel 6 erklärt die Verwendung einer IR-Fernbedienung am Raspberry Pi. Zusätzlich werden Entfernungen mit preisgünstigen Sensoren für Ultraschall und Infrarot gemessen.

Kapitel 7 bietet Ihnen Grundwissen zu Netzwerken. Wie werden IP-Adressen vergeben? Wie entsteht eine Direktverbindung zwischen dem Raspberry Pi und einem anderen PC? Wie gelingt ein Netzwerkzugriff? Nutzen Sie SSH, PuTTY, TMux und weitere Programme.

In **Kapitel 8** lernen sie ALSA kennen – das Linux-Soundsystem. Dabei geht es weniger darum, aus dem Pi eine Stereoanlage zu machen, als den Minicomputer zum Lauschen und Zuhören zu bringen.

In **Kapitel 9** erfahren Sie, wie Sie Videodateien abspielen und Ihre Webcam nutzen können: zum Fotografieren oder Filmen – auf Wunsch auch ereignisgesteuert.

Kapitel 10 zeigt die Verwendung der Kameramodule zum Raspberry Pi. Fotografieren und filmen Sie bei unterschiedlichen Lichtverhältnissen oder in Kombination mit einer Webcam.

Kapitel 11: Was geschieht mit den gewonnenen Daten? Auf der RAM-Disk ablegen, übertragen oder löschen? Cronjobs helfen bei der Organisation.

Kapitel 12 wendet sich nochmal dem Thema Audio zu, jetzt aber mit C/C++ Programmen, die es ermöglichen, Audiosignale zu analysieren. Lernen Sie Audiotools und Libraries kennen. Benutzen Sie den Raspberry Pi als Papagei, und lassen Sie sich von einem selbst gebauten Audioquiz überraschen.

In **Kapitel 13** lernen Sie die Bildverarbeitung mit OpenCV kennen. Viele Open-Source-Programme werden mit Tools wie CMakeGUI konfiguriert und direkt aus dem C++ Quelltext erzeugt. Bei mehr als 100 OpenCV-Beispielprogrammen ist garantiert etwas Passendes für Sie dabei! Ergänzend wird das ebenfalls von mir verfasste Buch „OpenCV Basiswissen“ empfohlen.

Kapitel 14 erläutert die Objekterkennung mit OpenCV – von der Kontrastspitzung und einfachen Segmentierung von Textzeilen bis hin zur anspruchsvolleren Symbolerkennung. Erleben Sie eine sprechende Lichtschranke!

In **Kapitel 15** wird MenuCtrl als Client-/Server-Software vorgestellt. Schrittweise wird eine Platine aufgebaut, die als Eingabegerät mit LCD-Display verwendet wird. So entsteht eine autarke Hardware und Software (gegebenenfalls für den Akkubetrieb), bei der auf Tastatur, Maus und Monitor verzichtet werden kann.

Nach dem Schlusswort in **Kapitel 16** folgt in **Kapitel 17** eine Übersicht der Programme und Skriptdateien zum Buch.

Hinweise zur Arbeit mit diesem Buch

Die Programme zum Buch, weitere Zusatzmaterialien sowie einen **elektronischen Anhang** mit ergänzende Informationen können Sie sich hier herunterladen: <http://downloads.hanser.de/978-3-446-45342-5>

Käufer der gedruckten Ausgabe dieses Buches erhalten das E-Book kostenfrei zum Buch. Die erste Seite des Buches enthält einen Zugangscode für das E-Book. Das E-Book ist komplett in Farbe und enthält in der EPUB-Version auch ein farblich hervorgehobenes Syntax-Highlighting.

Für den Raspberry Pi hat sich die Kurzbezeichnung Raspi eingebürgert. Wenn in diesem Buch vom Raspberry Pi oder Raspi gesprochen wird, sind damit alle bisher erschienenen Modelle gemeint. Wenn nur vom Pi gesprochen wird, ist sowohl der Raspberry Pi als auch der Banana Pi gemeint – auch wenn dieser Mini-PC nicht den Schwerpunkt des Buches bildet.

Im Buch sind viele Links zu Webseiten angegeben. Autor und Verlag können nicht für die dauerhafte Gültigkeit garantieren. Oft führt eine Suche nach Schlüsselwörtern zur genannten Webseite ebenfalls zu guten Treffern. Manchmal ist auch nur der Server vorübergehend nicht erreichbar.

Autor und Verlag haften nicht für eventuelle Schäden, die durch die Benutzung der im Buch beschriebenen Hardware oder Software entstehen könnten. Die Informationen wurden allerdings mit größtmöglicher Sorgfalt zusammengestellt, und es sind Warnhinweise zum Umgang mit Elektronik und elektrischen Spannungen enthalten.

Danksagungen

So wie das Raspbian-Betriebssystem eine bestimmte Zeit benötigt hat, um zum heutigen Entwicklungsstand zu gelangen, geht es einem manchmal auch als Autor. Deshalb möchte ich ganz besonders Herrn Frank Baur danken, der mich in zahlreichen Fachgesprächen immer wieder auf neue Ideen gebracht hat. Zur grafischen Umsetzung dieses Buches hat Frau Barbara Schmitz mit guten Tipps beigetragen. Meinem Bruder Erhard Eugen danke ich für zahlreiche Tests der Software, die er als völliger Laie auf dem Gebiet der Programmierung erfolgreich gemeistert hat. Ein ganz großer Dank gebührt auch Frau Julia Stepp vom Hanser Verlag, die die Entstehung dieses Buches über einen recht langen Zeitraum geduldig begleitet hat.

1

Die Welt des Raspberry Pi

Als der Bauingenieur Konrad Zuse 1941 den Z3 als ersten programmierbaren Computer mit binärer Gleitkommarechnung und insgesamt 2200 Relais erfand, waren Betriebssysteme noch nicht existent. Bei einem Hauptspeicher aus 1600 Relais dürfte das auch nicht weiter verwundern. Weniger bekannt ist, dass Zuse auch einen A/D-Wandler und erste Prozesssteuerungen erfand.

75 Jahre später arbeiten die meisten Computer fast unbemerkt und immer noch ohne Betriebssystem. Wenn Ihre Waschmaschine Temperatur, Drehzahl und Wasserstand ermittelt, dann leiten einige Sensoren die gemessenen Daten an einen Embedded Computer weiter, der einen vom Hersteller programmierten Mikrocontroller enthält und so eine Spezialaufgabe erfüllt.

Der Arduino ist ein sehr preisgünstiger Minicomputer, der ohne angeschlossene Peripherie allerdings absolut nutzlos ist. Mit angeschlossenen Sensoren und nach dem Transfer eines am PC entwickelten Programms können Sie damit hingegen Messwerte erfassen oder Relais ansteuern. Ein 8-Bit-Mikrocontroller führt das Programm aus, wobei der Arduino weder einen Monitoranschluss noch ein Betriebssystem hat.

Der Raspberry Pi ist im Gegensatz zum Arduino durch die Nutzung von Betriebssystemen viel universeller verwendbar und bindet Sie nicht an eine bestimmte Programmiersprache. Wie bei einem großen PC können Sie einen Bildschirm anschließen, eine Internetverbindung herstellen oder USB-Geräte anstecken. Zusätzlich kann dieser Mini-PC aber auch relativ leicht mit bestimmten Sensoren verbunden werden. Dort, wo ein Laptop zu groß, zu schwer und zu teuer ist, beginnt die Welt des Raspberry Pi.

■ 1.1 Die Modelle des Raspberry Pi

Als im Jahr 2012 der erste Raspberry Pi Computer auf den Markt kam, ahnte wohl noch niemand, dass sich daraus eine ganze Produktfamilie (siehe Bild 1.6) mit bislang mehr als 19 Millionen verkauften Exemplaren (März 2018) entwickeln würde. Alle Raspberry Pi Einplatinen-

computer haben einen recht geringen Energieverbrauch, der von der Prozessorleistung des jeweiligen Modells, aber auch von den angeschlossenen Geräten abhängt.

Kann man für Preise zwischen 11 und 45 € (je nach Modell und Bezugsquelle) auch wirklich einen vollwertigen Computer erwarten? Nicht ganz, denn Sie bekommen nur ein vollständig bestücktes Mainboard. Gehäuse, Netzteil, (micro)SD-Karte, HDMI-Kabel und ein USB-Hub sind nicht enthalten.



Bild 1.1 Ein Raspberry Pi 1 mit PS2/USB-Adapter, passivem USB-Hub, USB-Stick und WLAN-Stick

Was macht den Raspberry Pi so attraktiv? Es ist nicht nur der Preis, sondern vielleicht auch die Möglichkeit der Erweiterung und die Fähigkeit, externe Hardware vergleichsweise einfach anzusteuern. Wofür ist dieser kleine Computer zu empfehlen? Auf jeden Fall für den lautlosen Dauerbetrieb und für viele Arten von Spezialaufgaben, bei denen Tastatur und Monitor entfallen können – aber nicht müssen. Zusammen mit OpenCV können Sie z. B. eine Alarmanlage basteln, bei der verdächtige Bilder drahtlos zu einem anderen Rechner übertragen werden. Sehr oft wird der Raspberry Pi im eigenen Netzwerk eingesetzt und ist als einziger Computer permanent eingeschaltet.

A, A+, B, B+, Pi 2, Pi 3, Pi Zero W – welcher Raspberry Pi ist nun der richtige? Sämtliche lieferbaren Modelle werden auf der Website der Raspberry Pi Foundation beschrieben: <https://www.raspberrypi.org>

Der Raspberry Pi 3 und der ganz neue Pi Zero W sind mit einem WLAN-Chip ausgestattet. Beim Pi Zero W ist das besonders wichtig, denn dieses Minimodell hat keine Ethernet-Buchse. Letztlich hängt die Entscheidung davon ab, welche Rechenleistung und welche Schnittstellen Sie benötigen. Auf den Monitor kann man gegebenenfalls verzichten, wenn Sie den Raspberry Pi über SSH (SecureShell) fernsteuern. Ein WLAN-Stick kann bei allen Modellen über USB angeschlossen werden.

Im Jahr 2013 war der Raspberry Pi (Modell B) wahrscheinlich der weltweit bekannteste Einplatinencomputer mit folgenden Anschlüssen und Eigenschaften:

- Broadcom BMC2835-Prozessor, 700 MHz (ohne Overclock)
- Broadcom VideoCore-4-GPU mit OpenGL ES 2.0, h.264/MPEG-4
- 512 MB Hauptspeicher (DDR2-SDRAM)
- USB 2.0 (leider nur 2-fach)
- Micro-USB zur Spannungsversorgung (min. 700 mA)
- Netzwerkbuchse RJ 45 (Ethernet mit 10/100 MBit/s)
- HDMI-Ausgang (adaptierbar auf DVI)
- Cinchbuchse als analoger FBAS-Videoausgang
- 3,5 mm Klinensteckerbuchse für den Ton (Ausgang)
- CSI-Interface für das Kameramodul
- DSI-Interface für spezielle externe LCD-Displays
- GPIO-Port mit 26 Pins (z. B. für Sensoren, Relais etc.)
- Slot für eine SD-Karte mit Betriebssystem
- Energieverbrauch: 3,5 Watt bei 700 mA

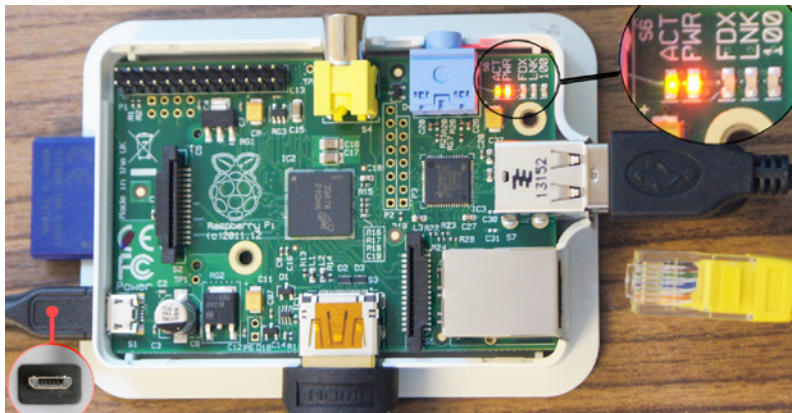


Bild 1.2 Raspberry Pi 1 (Modell B) mit aktiven LEDs für Power und Action

Inzwischen wird dieses weitverbreitete, aber ältere Modell mit SD-Karte als Raspberry Pi 1 (Modell B) bezeichnet (Bild 1.2). Die Modellreihe A ist preisgünstiger und etwas sparsamer ausgestattet. Zunächst geschah das durch Weglassen von Bauteilen auf der Platine. Die A+ Modellreihe (Bild 1.3) hat dagegen eine kleinere, fast quadratische Platine.

Die älteren Modelle A und B des Raspberry Pi (vor Juni 2014) werden nur noch selten als Neeware angeboten, haben aber immer noch einen teilweise erstaunlichen Sammlerwert. Bei beiden Modellen wird eine SD-Karte zum Booten des Betriebssystems benötigt. Die neueren Modelle (A+, B+, Zero, Pi 2, Pi 3) verwenden stattdessen eine microSD-Karte (Bild 1.4).

Beachten Sie bitte auch, dass es unterschiedliche Gehäuseformen gibt. Das Gehäuse aus Bild 1.2 ist für lediglich zwei USB-Buchsen vorgesehen, während die neueren Modelle der Reihe B über vier USB-Buchsen verfügen, wie in Bild 1.5 zu sehen. Nicht alle Gehäusehersteller haben Öffnungen für das Displaymodul und das Kameramodul vorgesehen. Die seriellen Anschlüsse dazu werden als DSI und CSI bezeichnet.

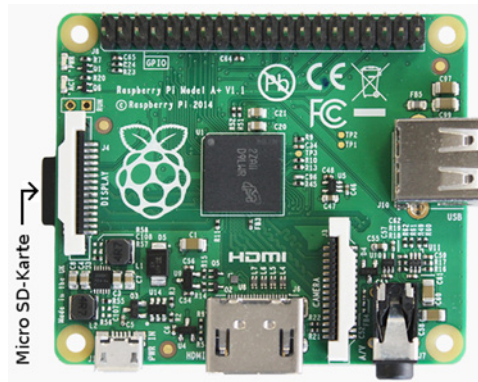


Bild 1.3 Das Modell A+ des Raspberry Pi 1 (© Bild: Raspberry Pi Foundation)

Ein signifikantes Merkmal aller Raspberry Pi Computer ist eine Stiftleiste, die als GPIO-Port bezeichnet wird. In Bild 1.2 sind das $2 \times 13 = 26$ Kontakte für den GPIO-Port. Es gibt jede Menge Hardware, die über General Purpose Input/Output (GPIO) anschließbar ist. Da die neueren Raspberry Pi Modelle 40 Kontakte haben, passt die eine oder andere Aufsteckplatine nur über einen zwischengeschalteten Adapter. Wie Sie elektronische Schaltungen über GPIO ansteuern, erfahren Sie in Kapitel 5.

	Raspberry Pi				
	B+	Zero W	2B	3B	3 B+
Prozessor	BMC2835				
Architektur ^{Bit}	ARMv6 ³²	ARMv7 ³²	ARMv8 ⁶⁴		
CPU-Takt (MHz) ^{Co}	700 •	1000 •	900 ••	1200 •••	1400 •••
GPU	VideoCore IV mit OpenGL ES 2.0, H264 / MPEG-4				
Hauptspeicher	512 MB		1024 MB		
USB-Ports (Anzahl)	4	1	4		
LAN (maximal)	100 MBit/s	—	100 MBit/s	300 MBit/s	
WLAN-Standard	—	2,4 GHz b/g/n	—	2,4 GHz b/g/n	2,4/5 GHz, ac
Audio-Out	Klinkenstecker	nur	Klinkenstecker		
Video-Out	4-polig	Kontakte	4-polig		
erhältlich seit	6/2014	2/2017	2/2015	2/2016	3/2018

Bild 1.4 Die neueren Modelle des Raspberry Pi (ab Juni 2014)

Der Raspberry Pi B+ ist ein Nachfolger des allerersten Raspberry Pi B Modells und kostet derzeit ca. 25 €. Der Hauptunterschied zum älteren Modell (Bild 1.2) liegt in einer 40-poligen Stiftleiste und in der nun 4-poligen Klinkensteckerbuchse. Es spricht aber aus meiner Sicht nur

noch wenig für dieses Modell, das von „unten“ und von „oben“ Konkurrenz aus dem eigenen Haus bekommen hat. Aber wer nicht viel Geld ausgeben möchte und eine Ethernet-Buchse braucht, ist mit diesem Modell gut bedient. Die Anschlüsse stimmen mit den wesentlich schnelleren Modellen Pi 2 und Pi 3 überein.

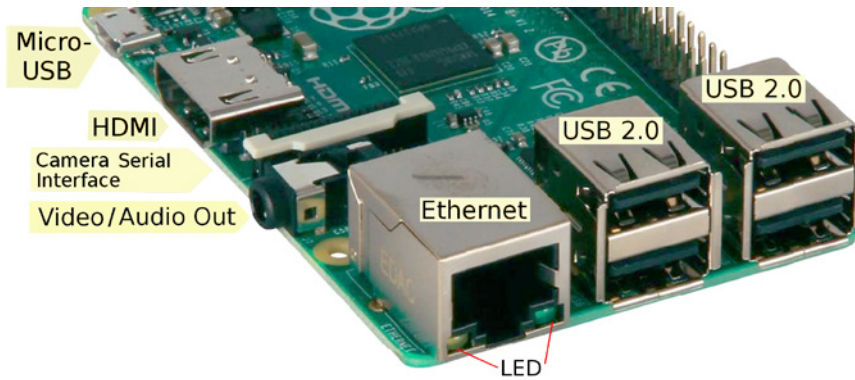


Bild 1.5 Anschlüsse am Modell B+, dem Raspberry Pi 2 und dem Pi 3

Obwohl Sie in Bild 1.5 vier USB-Buchsen sehen, hat sich in diesem Punkt nicht so viel geändert. Es gibt seit 2012 einen gemeinsamen Controller für LAN und USB, was dazu führen kann, dass sich die Geräte gegenseitig ausbremsen. Aber es bleibt die Hoffnung, dass ein noch nicht existenter Raspberry Pi 4 endlich auch USB 3.0 unterstützen könnte und auch getrennte Controller mitbringt.



Bild 1.6 Produkte der Raspberry Pi Foundation (Stand: Juni 2016)

1.1.1 Energieversorgung

Das am Raspberry Pi über die Micro-USB Buchse angeschlossene 5 Volt Netzteil sollte mindestens 1000 mA liefern können. Ein derartiges Netzteil reicht für die Raspberry Pi Modelle Zero, A, A+, B, B+ und knapp auch für den Pi 2 aus.

Dann kann es aber immer noch z.B. bei Festplatten kritisch werden, die keine eigene Spannungsversorgung haben und eine Energiezufuhr über USB erwarten. Es ist besser, wenn Sie von Anfang an ein Netzteil mit minimal 2000 mA und Micro-USB Kabel verwenden, das dann auch zum Banana Pi passen würde. Beachten Sie bitte, daß der Raspberry Pi 3 ein stärkeres Netzteil (2500 mA) als die Vorgängermodelle braucht. Nicht nur Festplatten konsumieren vergleichsweise viel Energie. Auch manche WLAN-Sticks und Soundkarten können nur über einen **aktiven** USB-Hub (siehe Bild 9.11) am Raspberry Pi betrieben werden.

In Bild 1.1 wird der Raspberry Pi 1 mit einem schwachen Netzteil und mit passivem USB-Hub betrieben. Würden Sie den WLAN-Stick erst im laufenden Betrieb anstecken, dann brähe die Spannung kurzzeitig zusammen, was zum erzwungenen Reboot führen würde. Es kommt also auch darauf an, welche Last ein Peripheriegerät beim Anstecken verursacht.

1.1.2 Der Raspberry Pi Zero W

Der Raspberry Pi Zero ist der kleinste und preisgünstigste Raspi (so der Spitzname) und wurde bislang in drei Varianten mit denselben Abmessungen (65 * 30 * 5 mm) gebaut:

Pi Zero (ohne Kameraanschluss)	ab 11/2015
Pi Zero (mit Kameraanschluss)	ab 5/2016
Pi Zero W (Kameraanschluss + WLAN)	ab 2/2017

Für die aktuelle Variante (ab 28.02.2017) wurde erstmals eine höhere Stückzahl eingeplant, und es gibt inzwischen pi3g als Distributor für den deutschsprachigen Raum. Der Pi Zero W kann unter www.buyzero.de (gegebenenfalls mit Zubehör) bezogen werden. Je nach mitgeliefertem Zubehör liegt der Preis zwischen 11 € und ca. 30 €.

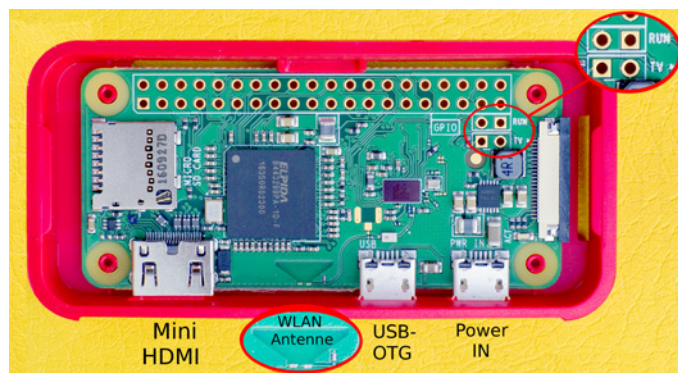


Bild 1.7 Der Raspberry Pi Zero W im Gehäuseunterteil

Der Preis hängt auch davon ab, welche Adapter(kabel) Sie benötigen. Der bei den größeren Modellen verwendete HDMI-Anschluss wurde aus Platzgründen durch Mini-HDMI ersetzt, ebenso USB durch Micro-USB. Kennzeichnungen am Gehäuseunterteil helfen, die Micro-USB-Buchsen voneinander zu unterscheiden (Bild 1.7). Auf die Verwendung von USB-OTG (On The Go) wird in Abschnitt 7.19 genauer eingegangen.

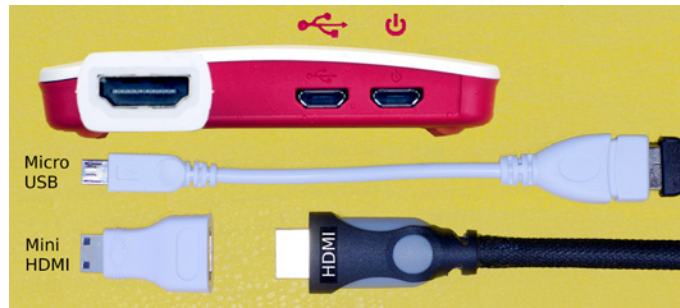


Bild 1.8 Adapter(kabel) für USB und HDMI

Die Raspberry Pi Foundation hat auch ein Gehäuseset entwickelt, das die in Bild 1.9 abgebildeten Teile (ohne den Pi Zero W) beinhaltet.

Der Pi Zero hält dabei ohne Schrauben im roten Gehäuseunterteil, muss aber (mit microSD-Karte) so eingesetzt werden, dass keine der Miniaturbuchsen in das Gehäuse hineinragt. Die gegenüberliegende Seite dient dabei quasi als Drehachse, wie in Bild 1.9 zu sehen.

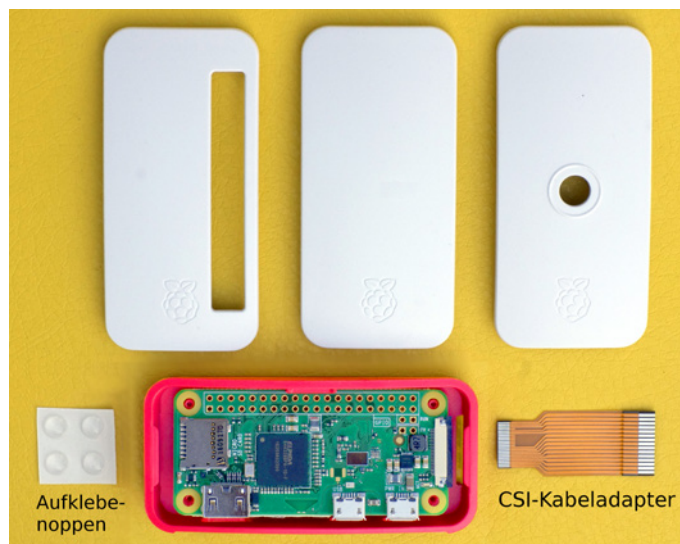


Bild 1.9 Raspberry Pi Zero W mit dem offiziellen Gehäuseset

Für den Kameraanschluss (siehe Bild 10.7) wird ein schmäleres Kabel als bei den großen Raspberry Pi Modellen benötigt. Das über pi3g erhältliche Gehäuseset enthält nicht nur diesen CSI-Kabel-Adapter, sondern neben dem roten Gehäuseunterteil auch drei Abdeckungen zur Auswahl. Die links in Bild 1.9 abgebildete Abdeckung enthält eine Öffnung für die GPIO-Stiftleiste. Die runde Öffnung im rechten Gehäuseoberteil ist für das Kameramodul vorgesehen.

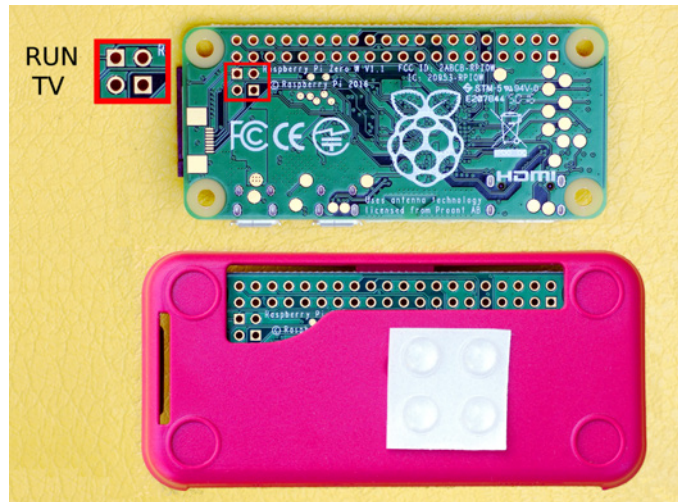


Bild 1.10 Platinenrückseite und Gehäuseboden mit zugänglichen Kontakten

Am auffälligsten sind am Raspberry Pi Zero vielleicht die Bohrungen für eine 40-polige Stiftleiste, deren Kontakte 2,54 mm Abstand zueinander haben. Die gesamte Platine ist nur 5 mm hoch, da die GPIO-Stiftleiste nicht eingelötet wurde und auch die Netzwerkbuchse fehlt. Der DSI-Anschluss für das Raspberry Pi Touchdisplay ist beim Pi Zero entfallen.

Der Zugang zur GPIO-Schnittstelle ist auch von der Gehäuseunterseite her möglich. So können Sie an den in den Bildern 1.7 und 1.10 vergrößert dargestellten Kontakten Drähte anlöten.

- Die Platinenbeschriftung „Run“ bezieht sich auf eine Reset-Möglichkeit durch Verbindung der beiden Kontakte.
- Die mit „TV“ beschrifteten durchkontaktierten Bohrungen liefern ein Composite Video-Signal, denn eine Klinkensteckerbuchse gibt es nicht.
- Eine bootfähige microSD-Karte muss vor dem Einbau in das Gehäuse eingesteckt werden. Es gibt keine Öffnung zum Wechsel der microSD-Karte.
- Die selbstklebenden Gumminoppen für den Gehäuseboden sind in Bild 1.10 noch nicht aufgeklebt.

Immerhin liegt die Prozessorleistung des Pi Zero etwa 40 % über der des Raspberry Pi 1 aus dem Jahr 2012. Dabei ist der Raspberry Pi Zero so klein, dass ein Einbau in nahezu jedes Gehäuse möglich erscheint.

Die im Vergleich zur A+ oder B+ Reihe weggefallenen Komponenten erleichtern den Akkubetrieb, wobei die Energieversorgung der HDMI-Buchse bei allen Modellen per Software abschaltbar ist (siehe Abschnitt 4.4.2).

Der Raspberry Pi Zero ist eher ein Bastelcomputer als ein klassischer Desktop-PC, für den die beiden Adapter(kabel) benötigt werden. Trotzdem sehen Sie in Bild 1.11 eine derartige Verkabelung, die Sie mit Bild 1.1 vergleichen können.

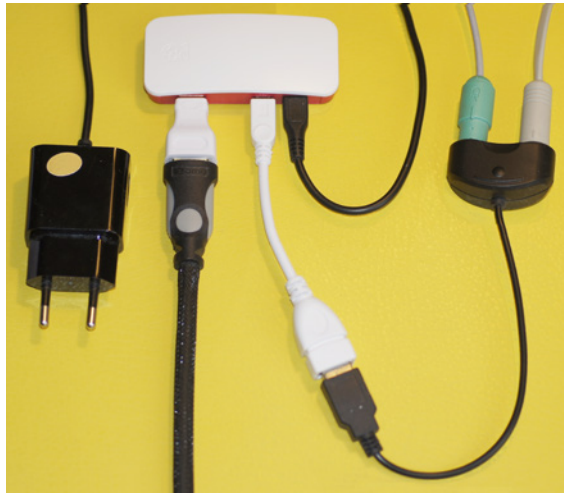


Bild 1.11 Verkabelung eines Pi Zero als Desktop-PC

Wer mehr als einen Pi-Zero-Computer hat, kann die Computer übereinander anordnen und z. B. über die serielle Schnittstelle miteinander vernetzen. Die nicht eingelötete Stiftleiste reduziert die Höhe eines solchen Arrays. Der Pi Zero ist als einzelne Platine viel zu leistungsschwach, um als dauerhafter Ersatz für einen älteren, vergleichsweise energiehungrigen Laptop zu dienen.

Für den Pi Zero W gelten folgende Daten:

- 1 GHz Single-Core-CPU
- 512 MB RAM
- Mini-HDMI-Port
- Micro-USB-On-the-Go-Port
- Micro-USB-Power
- HAT-kompatibler 40-Pin-Header
- Composite Video und Reset Header
- CSI Camera Connector
- 802.11n WLAN/Bluetooth v4.1/Bluetooth Low Energy
- Bluetooth 4.0

Man kann sich aber recht gut vorstellen, Tastatur und Maus über Bluetooth mit dem Pi Zero W zu verbinden. Am USB-Port kann dann z. B. ein USB-Stick als zusätzlicher Datenspeicher hängen. Bei bestimmten USB-Geräten (z. B. Soundkarten) werden Sie auch am Pi Zero W einen aktiven USB-Hub benötigen.

Benötigen Sie vielleicht auch noch einen Akku für mehr Mobilität? Alex Eames hat den Energiebedarf des Pi Zero W gemessen und die Daten auf folgender Webseite bereitgestellt: <http://raspi.tv/2017/how-much-power-does-pi-zero-w-use>

Ohne Last soll der Raspberry Pi Zero W im Mittel 120 mA konsumieren. Ein Akku mit 6000 mAh könnte den Pi Zero W damit zwei Tage lang versorgen – solange die Last gering bleibt.

Im Vergleich zu den Vorgängern der Zero-Modellreihe ist die Taktfrequenz angestiegen. Ein weiteres Unterscheidungsmerkmal ist die auf die Platinenunterseite gewanderte Himbeere. Bei höherer Prozessorlast kann es empfehlenswert sein, Kühlkörper zu verwenden. Ohne Lüftungsschlitze im Gehäuse sind Kühlkörper allerdings sinnlos.

1.1.3 Raspberry Pi 2 und Pi 3

Die Modelle Pi B+, Pi 2 B und Pi 3 B lassen sich auf den ersten Blick kaum unterscheiden. Für alle drei Modelle eignet sich exakt dasselbe Gehäuse, da alle Schnittstellen identisch angeordnet sind. Dort, wo beim Pi 2 die LEDs sitzen, hat der Pi 3 die WLAN-Antenne (siehe Bild 7.61).



Bild 1.12 System-on-a-Chip (B+, Pi 2 und Pi 3)

Ein genauer Blick auf den Broadcom-Chip zeigt beim Pi 3 die Beschriftung „BCM2837“, aber die Verschiebung der Himbeere ist auffälliger. Im Ruhebetrieb liegt der Energieverbrauch des Raspberry Pi 2 nicht höher als beim B+ Modell und damit weitaus niedriger als beim Banana Pi. Wenn lediglich ein Prozessorkern benutzt wird, liegt die Leistungssteigerung gegenüber dem Raspberry Pi B+ Modell bei über 40%. Bei Benutzung aller vier Prozessorkerne ist der Raspberry Pi 2 sogar um bis zu Faktor 6 schneller. Eine Tabelle zum Vergleich der Zeiten eines C++ Compilers finden Sie in Kapitel 13.

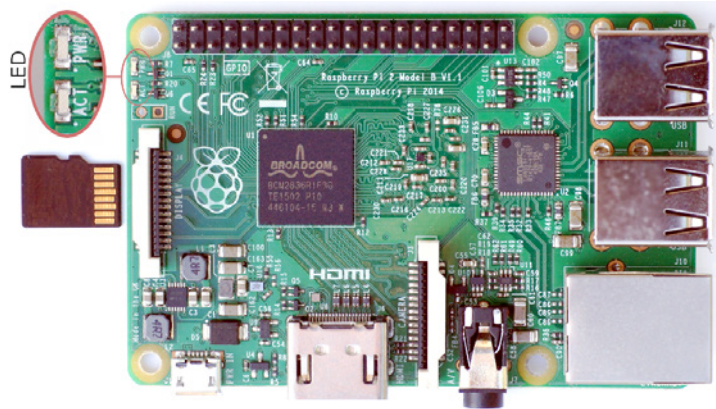


Bild 1.13 Raspberry Pi 2 mit 1 GB RAM und A7 4-Kern-Prozessor

Nochmal um bis zu 50 % zulegen kann der Raspberry Pi 3. Der Raspberry Pi 3 Modell B verfügt über einen 64-Bit-QuadCore-Prozessor, 1 GB Hauptspeicher und einen Cypress-CYW43438-Chip für das WLAN. Wie beim Pi Zero W werden WLAN 802.11n und Bluetooth 4.0 unterstützt. Dank des stärkeren Prozessors kann der Pi 3 im WLAN-Bereich eine höhere Transferrate herausholen als der Pi Zero W. Ein externer WLAN-Stick wie in Bild 1.1 ist nicht notwendig. Die Taktfrequenz ist mit 1,2 GHz höher als beim Pi 2, was aber auch ein stärkeres Netzteil mit 2,5 A erfordert.

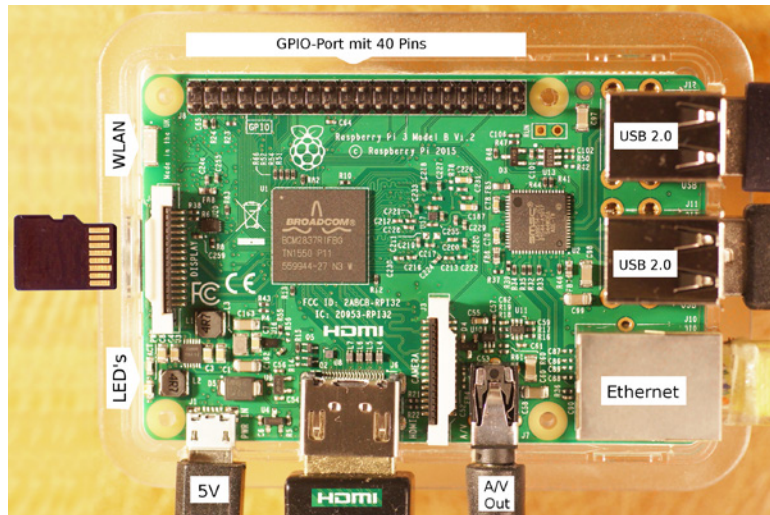


Bild 1.14 Ein Raspberry Pi 3 im transparenten Gehäuseunterteil

Die allermeisten Gehäuse für die B+ Serie, für den Pi 2 oder für den Pi 3 haben eine Öffnung zum Einlegen bzw. Entnehmen der microSD-Karte. Während der Raspberry Pi 2 eine Art Kugelschreibermechanik zur Entnahme der microSD-Karte hat, muss die Entnahme beim Pi 3 mit Hilfe einer Pinzette erfolgen, wie in Bild 1.15 angedeutet.

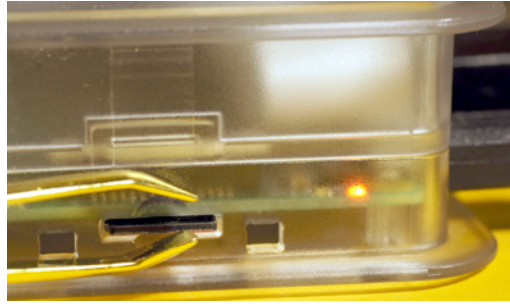


Bild 1.15 microSD-Karte im laufenden Raspberry Pi 3

Der etwas billigere Raspberry Pi 2 hat ebenfalls 1 GB Hauptspeicher, aber keinen WLAN-Chip. Dabei ist statt einer 64-Bit- lediglich eine 32-Bit-QuadCore-CPU in Verwendung. Raspbian unterstützt 64 Bit allerdings noch nicht. Für den Pi 2 und Pi 3 ist auch Windows 10 IoT verfügbar. Die älteren Modelle mit weniger Hauptspeicher sind fast ausschließlich für Linux-basierte Betriebssysteme vorgesehen.

Der Raspberry Pi 3B+

Am 14.3.2018 hat die Raspberry Pi Foundation das neue Modell Raspberry Pi 3B+ herausgebracht – gerade noch rechtzeitig vor Drucklegung dieses Buches. Die Verbesserungen liegen hauptsächlich bei der Taktfrequenz (1,4 GHz statt 1,2 GHz) sowie bei LAN und WLAN. Die gesteigerte Taktfrequenz erfordert zwingend ein Netzteil, das mindestens 2,5 A abgeben kann.

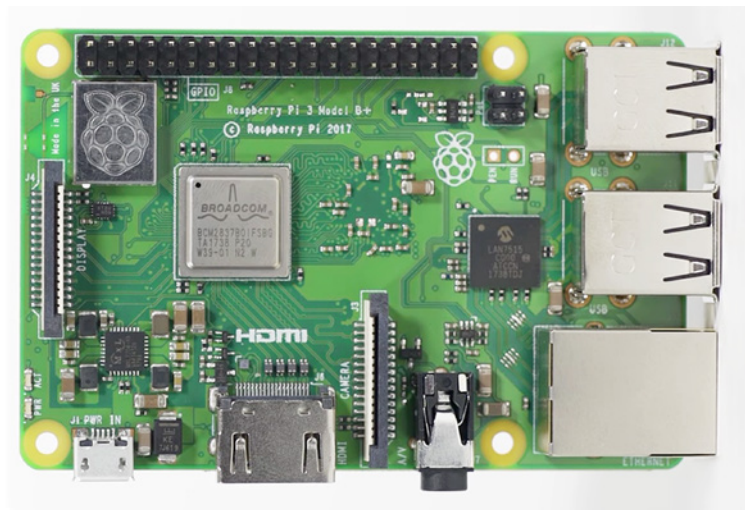


Bild 1.16 Der verbesserte Raspberry Pi 3B+ mit 1,4 GHz Taktfrequenz (Quelle: Raspberry Pi Foundation)

Auffällig ist dabei die metallene Kapselung, die sowohl den Prozessor als auch den WLAN-Chip betrifft. Sämtliche Anschlüsse sind unverändert positioniert. Für die vier Prozessorkerne wird weiterhin je eine ARM Cortex-A53 CPU verwendet. Durch die leicht gestiegene Taktfrequenz

ist der Pi 3B+ um ca. 15 % schneller und die metallene Verkapselung des SoC BMC2837B0 hat thermische Vorteile. ARMv8 bei 64 Bit ist als Befehlssatz des Prozessortyps weiterhin gültig.

Für das WLAN soll sich ein um etwa 30 % verbesserter Datendurchsatz bei 2,4 GHz ergeben, im 5 GHz-Band sogar eine Steigerung auf 102 Mb/sec, was Faktor 2,85 gegenüber dem Pi 3B entspricht. WLAN und Bluetooth sind vom Typ BMC43455 und unter dem in Blech eingepprägten Himbeersymbol verborgen.

Der LAN-Netzwerkchip mit der Beschriftung LAN7515 befindet sich dagegen in der Nähe der USB-Buchsen und erfüllt nun die Gigabit Ethernet-Spezifikation. Da die Anbindung weiterhin über USB 2.0 erfolgt, werden jedoch leider nur maximal 300 Mbps erreicht. Dies ist die Kurzfassung der Daten:

SoC	BCM2837B0
ARM-Kerne, Takt	4 × Cortex-A53 (ARMv8), 0,6 bis 1,4 GHz
Grafikeinheit	VideoCore IV
RAM	1 GByte
Gigabit-Ethernet	Microchip LAN7515 (USB 2.0)
WLAN 2,4/5 GHz	BCM43455 (SDIO, BT 4.2) IEEE 802.11ac Standard
Bluetooth	BLE 4.2 (Low Energie)
Power over Ethernet	PoE

Neu sind auch 4 Pins für Power over Ethernet. Damit wird es möglich, den Raspberry Pi über LAN mit Energie zu versorgen. Für 2018 ist übrigens auch ein Raspberry Pi 3 A+ angekündigt, der im Format des A+-Modells (Bild 1.3) herauskommen soll. Mit der Verkleinerung der Platine wird die Ethernetbuchse entfallen, WLAN und Bluetooth werden aber verfügbar bleiben. Ob Miniaturbuchsen wie beim Pi Zero W eingesetzt werden, ist noch unklar. Für 2019 ist ein Raspberry Pi 4 geplant, der wie üblich zum Preis von 39 \$ auf den Markt kommen soll.

Fazit: Wer Linux und den Raspberry Pi noch nicht kennt, sollte zum Raspberry Pi 2 oder Pi 3 greifen, denn bei den preisgünstigeren Modellen ist der Betrieb mit grafischen Benutzeroberflächen etwas langsam, was Ihnen den Spaß mit Linux verderben könnte. Der Hauptvorteil des Pi 3B+ liegt im verbesserten LAN/WLAN, wird aber mit einem höheren Energiebedarf erkauft. Der Pi 3B wird inzwischen etwas günstiger verkauft.

Wenn Sie in Bezug auf Elektronik ein Einsteiger sind, aber Schaltungen für den Raspberry Pi entwickeln wollen, kann es risikoloser sein, mit einem gebrauchten Raspberry Pi anzufangen und dessen GPIO-Port zu nutzen. Ein falsch angeschlossenes Kabel am GPIO-Port oder ein vergessener Vorwiderstand kann die Hardware zerstören. Der Raspberry Pi Zero W ist eine sehr preisgünstige Alternative, wenn Ihnen eine geringe Prozessorleistung genügt, oder wenn Sie zwingend einen Akkubetrieb brauchen. Bastler müssen dabei lediglich die GPIO-Stiftleiste oder die notwendigen Kabel selbst anlöten.

Die offizielle Dokumentation zum Raspberry Pi finden Sie auf folgender Webseite:
www.raspberrypi.org/documentation

■ 1.2 SD-Karten und microSD-Karten

Bei SD-Karten bzw. microSD-Karten wird zwischen HighCapacity und eXtendedCapacity unterschieden, was den letzten beiden Buchstaben von SDHC und SDXC entspricht. SD-Karten mit mehr als 32 GB, die dem SDXC-Standard entsprechen, sind auf dem Raspberry Pi nicht verwendbar. Preisgünstiger und verwendbar sind alle (micro)SD-Karten vom Typ SDHC:

- Alle SD-Karten bzw. microSD-Karten sind mit Class 4 bis Class 10 als Richtwert für die minimale Schreibgeschwindigkeit von 4 MB/s bis 10 MB/s gekennzeichnet.
- Die Leserate ist entweder direkt in MB/s aufgedruckt oder als Vielfaches von 150 kB/s, was der ursprünglichen Lesegeschwindigkeit einer CD-ROM entspricht. So entsprechen 200x 30 MB/s und 600x dem schon recht brauchbaren Wert von 90 MB/s.
- Je nach Typ (SDHC/SDXC) finden Sie ein entsprechendes Logo (siehe Bild 1.17).

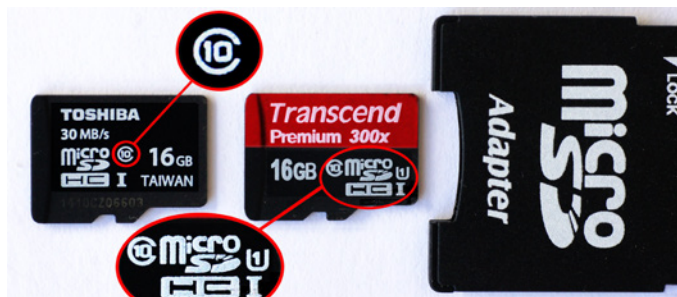


Bild 1.17 Class 10 microSD Karten mit nominell 16 GB

Der insgesamt verfügbare Speicherplatz auf (micro)SD-Karten ist herstellerabhängig. Eine aufgedruckte Größe von z. B. 16 GB ist nur ein grober Richtwert. Beim Backup (Abschnitt 3.4) der SD-Karte und bei der Direktkopie (Abschnitt 3.7) ist das von großer Bedeutung.

Über den microSD-Adapter können Sie einerseits eine microSD-Karte in jedem Kartenleser am PC beschreiben, andererseits aber auch eine fertig konfigurierte microSD-Karte (z. B. aus dem Raspberry Pi 2) an einem älteren Raspberry Pi 1 betreiben. Da die winzigen microSD-Karten sehr leicht aus den Fingern fallen, ist der Adapter auch eine schützende und beschriftbare „Garage“ (siehe Bild 3.5).

Falls die microSD-Karte im Raspberry Pi funktioniert, aber Ihr PC nicht reagiert, wenn die microSD-Karte über Adapter und Kartenleser verbunden ist, dann können Kontaktprobleme die Ursache sein.

■ 1.3 Von Wheezy zu Stretch – Downloads für Raspbian

Mit etwas Humor lässt sich Raspberry Pi als „Himbeertörtchen“ übersetzen. Auch die Bezeichnung für das Betriebssystem Raspbian setzt sich aus zwei Worten – nämlich Raspberry und Debian – zusammen, wie Bild 1.18 verdeutlicht.



Bild 1.18 Die Wortkomposition Raspbian

Debian existiert schon seit 1993 und ist als Betriebssystemkern Stammvater von bekannten Linux-Distributionen wie Ubuntu oder Linux Mint. Die Anpassung von Debian 7 (Debian Wheezy) für die Hardware des Raspberry Pi führte zum Begriff Raspbian Wheezy. Seit dem 28. 9. 2015 war auch Raspbian Jessie verfügbar. Die Umstellung auf Raspbian Stretch ist mit der Version vom 17. 8. 2017 erfolgt, was nun Debian 9 als Kern des Betriebssystems entspricht. Raspbian ist das derzeit beliebteste Linux-Betriebssystem für den Raspberry Pi. Deshalb wird in diesem Buch fast immer von Jessie oder Stretch als installiertem Betriebssystem ausgegangen.

Eine Auswahl sämtlicher bisher erschienener Versionen (siehe Bild 1.19) erhalten Sie unter <https://downloads.raspberrypi.org/raspbian/images>.

Name	Last modified	Description
Parent Directory		
2012-07-15-wheezy-raspbian/	16-Jul-2012 14:58	Wheezy
2012-08-16-wheezy-raspbian/	18-Aug-2012 23:39	
raspbian-2015-02-17/	17-Feb-2015 00:14	Jessie
raspbian-2015-05-07/	07-May-2015 16:54	
raspbian-2015-09-28/	28-Sep-2015 23:47	
raspbian-2015-11-24/	24-Nov-2015 02:51	Stretch
raspbian-2017-06-23/	2017-06-23 06:14	
raspbian-2017-07-05/	2017-07-05 16:43	
raspbian-2017-08-17/	2017-08-17 08:17	
raspbian-2017-09-08/	2017-09-08 11:13	
raspbian-2017-12-01/	2017-12-01 10:22	
raspbian-2018-03-14/	2018-03-14 05:24	

Bild 1.19 Downloadauswahl unter allen Raspbian-Versionen

Sie sehen daran, dass auch die älteste Version aus dem Jahr 2012 immer noch verfügbar ist. Das kann vor allem dann interessant sein, wenn Sie eine länger nicht benutzte (micro)SD-Karte haben und anhand von Dateien und Zeitstempeln die Version herausbekommen wollen.

Vieles, was Sie in diesem Kapitel über Raspbian erfahren, ist übrigens auch direkt auf den Banana Pi übertragbar. Die Installation des Betriebssystems ist bei beiden Pi-Computern vollständig von der Hardware entkoppelt. Es wird dafür nur eine geeignete SD-Karte (bzw. microSD-Karte) benötigt.

1.3.1 Die Release Notes als Chronik

In der Regel sind Sie mit der aktuellsten Raspbian-Stretch-Version am besten bedient. Doch da Raspbian Stretch noch recht neu ist, kann es Gründe geben bei einer persönlich erprobten Jessie-Version zu bleiben. Welche Änderungen stecken in der aktuellen Raspbian-Version? Welche Softwareprobleme wurden inzwischen gelöst? Darüber informieren die Release Notes, die Sie auf folgender Webseite finden: http://downloads.raspberrypi.org/raspbian/release_notes.txt

Sie finden dort eine recht lange Liste der Änderungen, die in etwa folgende Form hat:

```
2018-03-13:
* Raspberry Pi 3 B+ support
* WiFi is disabled until wireless regulatory domain is set (Pi 3 B+ only)
  - The domain can be done through 'Raspberry Pi Configuration' (rc_gui),
    'raspi-config' or by setting 'country=' to an appropriate ISO 3166
    alpha2 country code in /etc/wpa_supplicant/wpa_supplicant.conf.
* Default wireless regulatory domain is now unset
* Added support to desktop for different screen sizes and resolutions,
  including multiple preset options in Appearance Settings and pixel doubling
  option in Raspberry Pi Configuration
* Version 2.1.16 of Thonny included
* Version 29.0.0.113 of Adobe PepperFlash player included
* Version 1.2.post1 of Pygame Zero included
* Bluetooth plugin now supports connection to Bluetooth LE HID devices
* Network plugin now indicates 5G-compatible APs
* Latest changes to BlueZ ALSA service merged
  - service now started on CLI boot as well as GUI boot
* Latest changes to dhcpcd networking plugin merged
* Improved support for running on pi-top devices
* Small design changes to PiX theme and icons
* Bug fix - hide spurious window resize handles
* Bug fix - Scratch 2 remote GPIO state block now works correctly
* Updated WiFi Firmware
  - brcmfmac43455-sdio 7.45.154
  - brcmfmac43430-sdio 7.45.98.38
* New packages:
  - policykit-1
  - obconf
  - python-buttonshim python3-buttonshim
  - python-unicornhathd python3-unicornhathd
  - python-pantilthat python3-pantilthat
* Linux kernel 4.9.80+
* Raspberry Pi firmware 3347884c7df574bbabeff6dca63caf686e629699
2017-11-29:
* Added battery monitor plugin for taskbar - works on x86 images or first-generation Pi-Top
* Added shutdown mode to PCManFM file manager to reduce complexity
* Added ability to rename files in PCManFM by clicking name when selected
* Bug fix in Bluetooth ALSA module to reduce truncation of audio at end of playback
* Various small tweaks, bug fixes and theme modifications
* New kernel and firmware
```

```
2017-09-07:
* Disable predictable network interface names for Ethernet devices
* Bug fix for keyboard settings dialog in Raspberry Pi Configuration
* Bug fix for crash on some videos and animations in Chromium
* Bug fix for taskbar crash when running RealVNC server
* Bug fix for reloading projects with extensions in Scratch 2
* Bug fix for MAC address problem in Bluetooth
* Simple mode and new icons in Thonny
* New Japanese translations in Raspberry Pi Configuration
* Install fonts-droid-fallback for international fonts
2017-08-16:
* Based on Raspbian Stretch (Debian version 9)
....
```

Die Release Notes gehen zurück bis zur allerersten Raspbian-Wheezy-Version und geben einen guten Einblick von der Entwicklung und den Fähigkeiten des Raspbian-Betriebssystems. Für eine fertig konfigurierte SD-Karte bekommen Sie die Version so heraus:

```
uname -a
lsb_release -a
```

Probieren Sie diese beiden Kommandos bitte im LXTerminal aus.

■ 1.4 Über NOOBS zur Raspbian SD-Karte

Vielleicht ist der Raspberry Pi Ihr erster Linux-PC und Sie wissen noch nicht genau, wie Sie am besten loslegen. Da alle Raspberry Pi Modelle das Betriebssystem nicht auf einer Festplatte, sondern auf einer (micro)SD-Karte erwarten, muss vor dem Start eine vorbereitete SD-Karte bzw. microSD-Karte vorhanden sein. Dafür gibt es folgende Wege:

- Sie kaufen eine (micro)SD-Karte mit NOOBS.
- Sie verwenden Ihren PC, um NOOBS auf die (micro)SD-Karte zu bringen.
- Sie verwenden Ihren PC, um Raspbian auf die (micro)SD-Karte zu bringen.

Karl May hätte NOOBS mit seiner eigenen Wortschöpfung „Greenhorn“ übersetzt. Wer also ein Linux-Neuling ist, sollte NOOBS verwenden. Der Name NOOBS steht für *New Out Of the Box Setup*. NOOBS lässt sich mit jedem Dateimanager auf die SD-Karte kopieren.

Eine gekaufte (micro)SD-Karte mit NOOBS hat allerdings folgende Nachteile:

1. Der Softwarestand ist möglicherweise über ein halbes Jahr alt.
2. Sie bezahlen mehr als für eine leere (micro)SD-Karte gleicher Kapazität.
3. Sie beginnen ohne Backup des Betriebssystems.

Sie können den Raspberry Pi nur über die eingesteckte SD-Karte booten. Über USB angeschlossene Geräte wie Kartenleser, USB-Stick, Festplatte, oder DVD-Laufwerk dienen nur dem Datenaustausch.

Wenn Sie eine fertige (micro)SD-Karte mit NOOBS verfügbar haben sollten, können Sie in Abschnitt 1.5 weiterlesen – es schadet allerdings nicht, über die Konfiguration der SD-Karte informiert zu sein.

Dabei ist NOOBS eher ein Installationsprogramm und so gesehen ein indirekter Weg zum passenden Betriebssystem. Eine schon benutzte SD-Karte sollte neu formatiert werden, wobei in Bild 1.20 zugleich „SD-Card“ als Labelname vergeben wurde.

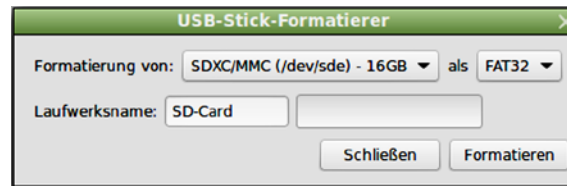


Bild 1.20 Formatierung einer 16 GB SD-Karte mit Linux Mint

Beachten Sie bitte, dass eine frisch formatierte (micro)SD-Karte entnommen und wieder eingesteckt werden sollte, damit die neu erzeugte Partition auch erkannt wird. Wenn die Formatierung aus irgendeinem Grund scheitern sollte, bleibt noch die Möglichkeit, die (micro)SD-Karte in der Digitalkamera zu formatieren. Im Zweifel greifen Sie bitte zu einer neuen Class-10-(micro)SD-Karte mit mindestens 8 GB.

1.4.1 NOOBS aus dem Internet holen und entpacken

NOOBS ist eine Installationssoftware zum Raspberry Pi. Es enthält in der Offlineversion das Raspbian-Betriebssystem und kann bei bestehender Onlineverbindung auch noch weitere Betriebssysteme auf die (micro)SD-Karte bringen. Wenn Sie sich für mehr als ein Betriebssystem entscheiden, wird eine Bootweiche eingerichtet. Sie finden NOOBS im Downloadbereich der Raspberry Pi Foundation (www.raspberrypi.org/downloads).

Dabei unterscheidet die Raspberry Pi Foundation zwischen den von ihr verwalteten Downloadmöglichkeiten für NOOBS, RASPBIAN¹ und RPI-DESKTOP (PC/MAC) sowie den extern verwalteten Third-Party-Betriebssystemen wie in Bild 1.21 zu sehen.

NOOBS und RASPBIAN sind für die Hardware des Raspberry Pi vorgesehen, während RASPBERRY PI DESKTOP zu einer *.iso-Datei führt, die für einen klassischen x86-PC vorgesehen ist (siehe Abschnitt 2.6).

Seit Erscheinen des Raspberry Pi 3 ist für Raspbian auch mit einer 64-Bit-Variante zu rechnen. Momentan existieren aber nur 32-Bit-Versionen, die für alle Generationen des Raspberry Pi verwendbar sind (Stand: Februar 2018). Wie RASPBIAN liegen die Third-Party-Betriebssysteme als sogenannte Abbilddateien vor. Wenn Sie RASPBIAN ohne NOOBS installieren wollen, lesen Sie bitte in Kapitel 2 weiter.

¹⁾ Mit „RASPBIAN“ ist die Abbilddatei bzw. die Downloadmöglichkeit gemeint, während „Raspbian“ ein schon funktionsfähiges Betriebssystem beschreibt.

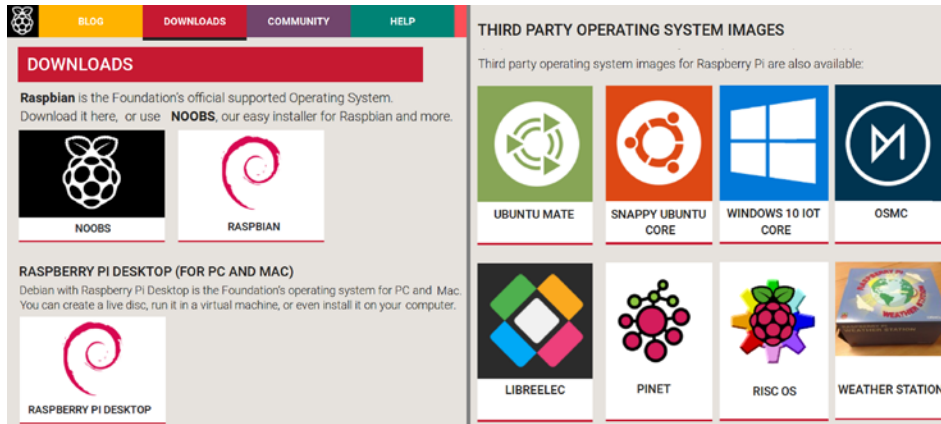


Bild 1.21 Betriebssysteme für den Raspberry Pi

Nach dem Anklicken von „NOOBS“ eröffnen sich zwei Downloadmöglichkeiten:

- Die Offlineversion braucht für Raspbian keine Internetverbindung während der Installation. *NOOBS_v2_4_5.zip* hat ca. 1,5 GiB und enthält zusätzlich auch LibreELEC als Multimedia-Betriebssystem. (Stand: 29. 11. 2017)
- Die Onlineversion *NOOBS_lite_v2_4.zip* ist wesentlich kompakter (ca. 32 MiB), erfordert aber generell eine Internetverbindung.

Welche Betriebssysteme in der jeweiligen NOOBS-Version enthalten sind, sehen Sie derzeit leider nicht auf der Webseite. Sie können aber davon ausgehen, dass Raspbian immer dabei sein wird. Welche Raspbian-Version enthalten ist, bekommen Sie bei NOOBS über einen Datumsvergleich mit dem Downloadbereich (Bild 1.19) heraus. Die Version von NOOBS ist hier vergleichsweise bedeutungslos. Bei NOOBS LITE ist „Release date“ vergleichsweise unwichtig, da alle Betriebssysteme nur online geholt werden.



Bild 1.22 Download von NOOBS über www.raspberrypi.org

Die nachfolgende Beschreibung geht überwiegend von „NOOBS Offline and network“ aus. Beim ersten Boot-Vorgang mit dem Raspberry Pi wird für Raspbian kein Internetzugang benötigt. Für weitere Betriebssysteme, die im Sinne von „Multiboot“ hinzukommen sollen, allerdings schon.

Ein englischsprachiges Video zur Installation und Benutzung von NOOBS finden Sie unter <https://www.raspberrypi.org/help/noobs-setup>.

Sie holen sich *NOOBS_v2_4_5.zip* (oder eine noch aktuellere Version) mit Ihrem PC aus dem Internet und entpacken die Datei zunächst in ein Verzeichnis der Festplatte. Viele Dateimanager (z.B. Dolphin bei Linux, Total Commander bei Windows) enthalten Tools wie 7-Zip zum Entpacken. Sie finden Links zu den Entpackprogrammen auch auf der Webseite der Raspberry Pi Foundation.

1.4.2 NOOBS auf die leere (micro)SD-Karte kopieren

Wenn Sie eine neu gekaufte SD-Karte (z.B. 16 GB SDHC, Class 10) verwenden, dann können Sie die bereits expandierten Dateien aus dem ZIP-Archiv direkt auf die SD-Karte kopieren (siehe Bild 1.23). Diesen Vorgang können Sie auch auf einem Windows-PC oder Mac durchführen. Eine schon verwendete, bootfähige SD-Karte muss vorher neu formatiert werden. Alternativ formatieren Sie die SD-Karte einfach mit der Digitalkamera.

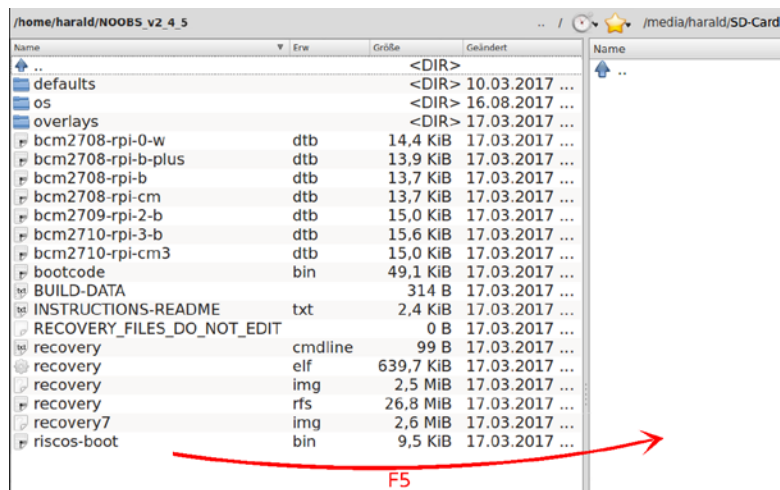


Bild 1.23 Der entpackte Inhalt von *NOOBS_v2_4_5.zip* im Krusader

Wenn Sie zum Kopieren den Dateimanager Krusader verwenden, wird die gesamte linke Seite aus Bild 1.23 markiert und dann per <F5>-Taste kopiert. Die recht einfache Vorgehensweise wird auch in der Datei *INSTRUCTIONS-README.txt* beschrieben. Es funktioniert problemlos mit jedem Windows-/Linux-PC oder Mac. Im Unterverzeichnis *os* sind die direkt nutzbaren Betriebssysteme in gepackter Form enthalten. So wissen Sie vorab, welche Betriebssysteme enthalten sind, und wofür eine Internetverbindung benötigt wird.

■ 1.5 NOOBS auf dem Raspberry Pi

Wenn die (micro)SD-Karte mit NOOBS fertiggestellt ist (oder so erworben wurde), kann die Inbetriebnahme des Raspberry Pi beginnen. Stecken Sie dazu zuerst die SD-Karte mit NOOBS in den Raspberry Pi. Schließen Sie nun auch Maus, Tastatur und Ihren Monitor an – aber warten Sie noch ein wenig mit dem Netzteil.

1.5.1 Wohin mit dem Bild?

Die weiteren Vorgänge hängen sehr stark von Ihrer Hardwarekonfiguration und teilweise auch von dem Modell des Raspberry Pi ab, das Sie gerade verwenden. NOOBS ist ein kleines Installationsprogramm für die gewünschten Betriebssysteme, hat eine grafische Oberfläche und erwartet, dass eine Maus verfügbar ist.

Dabei stellt sich NOOBS gleich zu Beginn ein paar Fragen:

- Ich will bereits seit 3000 μ s ein Bild ausgeben – aber über HDMI kommt keine Antwort rein.
- Muss ich mich jetzt mit anderen Computern um den Flachbildschirm streiten? Display-Port, HDMI, DVI, VGA – was ist denn gerade am Monitor eingestellt, komme ich auch mal dran?
- Wenn ich nicht innerhalb der nächsten 50 ms die Auflösung reinbekomme, dann werd ich bockig und geh in den VGA-Modus!

Und was „denkt“ Ihr Monitor?

- Mein Chef hat den Kanalwahlknopf gedrückt – ich soll auf HDMI gehen, aber da gibt's kein Signal.
- Jetzt schickt er mich zu DVI – da find ich auch kein Signal.
- Nun soll ich auf VGA-in, so was Altmodisches, da find ich aber auch nix.
- Jetzt mag ich nicht mehr: Am Display-Port ist doch so ein großer PC dran, den nehme ich jetzt automatisch.

Zugegeben die „Dialoge“ sind ein wenig zugespitzt, zeigen aber, dass sich der Raspberry Pi und Ihr Monitor irgendwie unterhalten müssen. Bevor Probleme aufkommen, kann folgender „Fragebogen“ helfen:

- Ist das Kabel zwischen Raspberry Pi und dem Bildschirm beschädigt oder irgendwo zu stark geknickt?
- Sitzen alle Stecker richtig?
- Welcher Eingang ist am Monitor wirksam?
- Steckt die bootfähige (micro)SD-Karte richtig?
- Ist das Netzteil stark genug?

Jeder Monitor verhält sich bei der Wahl des Eingangs ein wenig anders. Typisch ist die Meldung „kein Signal“ für eine Dauer von etwa 2 Sekunden, wenn z.B. bei HDMI-Eingang kein

Gerät angeschlossen ist. Der Monitor geht dann typischerweise zu der Quelle zurück, bei der ein gültiges Signal vorliegt. Die meisten Monitore haben solch eine Autodetect-Funktion. Beim ersten Boot-Vorgang mit dem Raspberry Pi sollte möglichst nur der Raspberry Pi angeschlossen sein.

Falls Ihr (wohl recht betagter) Monitor lediglich einen VGA-Anschluss hat, helfen die etwas teureren HDMI/VGA-Konverter, aus dem Digitalsignal ein Analogsignal zu machen. Der Raspberry Pi kann in solch einem Fall die erforderliche Auflösung nicht feststellen und wird dann zunächst 640 * 480 Pixel verwenden. Leider kann nicht garantiert werden, dass alle HDMI/VGA-Konverter mit dem HDMI-Signal des Raspberry Pi klarkommen. Auch hier gilt, dass möglichst nur ein einziger Signaleingang am Monitor beim ersten Boot-Vorgang genutzt werden sollte. Weitere Hinweise zu HDMI/VGA finden Sie in Abschnitt 2.11.



Bild 1.24 Anschlüsse an einem HDMI-zu-DVI-Kabel

Im Idealfall haben Sie einem Flachbildschirm mit HDMI-Eingang und ein HDMI/HDMI-Kabel. Aber auch ein etwas älterer Monitor mit DVI-Buchse ist problemlos verwendbar, wenn Sie ein HDMI-zu-DVI-Kabel haben oder über einen entsprechenden Adapter verfügen. HDMI wurde aus dem DVI-Standard heraus entwickelt. In beiden Fällen erhält der Monitor vom Raspberry Pi ein digitales Signal, was fast immer problemlos funktioniert.

1.5.2 Energie für NOOBS

Sie nehmen nun an, dass Ihrem Raspberry Pi nur noch die Energie fehlt? Aber halt – benutzen Sie „NOOBS offline“ oder „NOOBS LITE“?

Wenn sich „NOOBS LITE“ auf der SD-Karte befinden sollte, wird zwingend eine Internetverbindung benötigt. Wenn Sie erstmals einen Raspberry Pi booten wollen, sollten Sie „NOOBS offline“ verwenden. Das gilt ganz besonders für alle Raspberry Pi Modelle, denen die Ethernet-Buchse fehlt.

Wenn alles korrekt verkabelt ist (siehe Bild 1.1 oder 1.11) und die richtige SD-Karte im Raspi steckt, bringt die angelegte Spannung zumindest die PWR-LED zum Leuchten. Wenn auch die grüne Leuchtdiode ab und zu blinkt, dann bootet der Raspi, und Sie dürfen „Heureka“ ausrufen, sofern der Monitor nicht dunkel bleibt. Als Startsignal gibt der Raspberry Pi auf Ihrem Monitor kurzzeitig das Bild eines Spektrums aus.

Solange die NOOBS-SD-Karte noch nie im Raspberry Pi war, besteht lediglich eine einzige FAT32-Partition auf der SD-Karte. Die Anpassung der FAT-Partition ist das Erste, was NOOBS nach dem Einschalten des Raspberry Pi tut (siehe Bild 1.26).

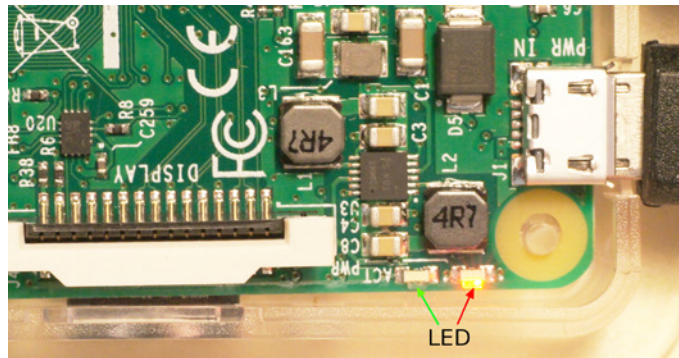


Bild 1.25 Leuchtdioden für Power und Action beim Raspberry Pi 3

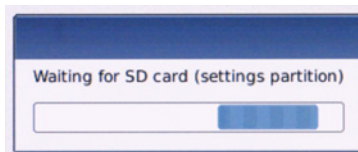


Bild 1.26 NOOBS modifiziert die Partitionen auf der (micro)SD-Karte.

Bei den Modellen mit 1 GB Hauptspeicher ist dieser Vorgang in weniger als 20 Sekunden erledigt. Bei einem Raspberry Pi 1 oder Zero kann der Vorgang bei einer 32 GB SD-Karte auch länger dauern.

Falls Sie nicht so weit kommen sollten und der Bildschirm schwarz bleibt, kann das an der SD-Karte liegen. Es gibt eine grüne LED auf dem Mainboard, die mit ACT bezeichnet ist (siehe Bild 1.2 oder 1.25). Diese LED blinkt im Betrieb bei jedem Speicherkartenzugriff, falls die Speicherkarte kompatibel ist.

Falls der Raspi die SD-Karte nicht verträgt (was sehr selten sein dürfte), finden Sie unter http://www.elinux.org/RPI_VerifiedPeripherals Hinweise zu (in)kompatibler Hardware. Von dort können Sie z.B. zu den Gruppen „SD-Cards“, „WiFiAdapter“ oder „USB_MouseDevice“ jeweils in den Ausführungen „working/problem“ verzweigen.

Beim Raspberry Pi Zero W strahlt die grüne ACT-LED seitlich ab und befindet sich direkt neben der Befestigungsbohrung, wie in Bild 1.27 zu sehen.

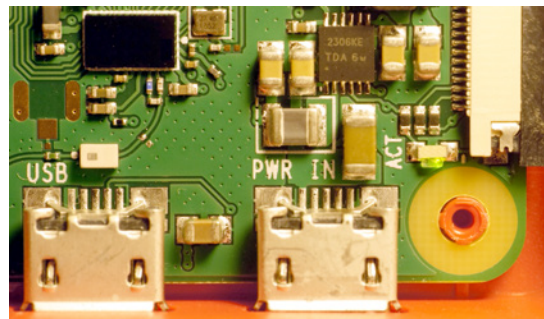


Bild 1.27 Micro-USB-Buchsen und Action-LED beim Raspberry Pi Zero W

Die Partitionen einer NOOBS-SD-Karte, von der schon gebootet wurde, sehen Sie in Bild 3.17. Eine englischsprachige Anleitung zur Verwendung von NOOBS (und die Quelltexte zu NOOBS) finden Sie unter <https://github.com/raspberrypi/noobs>.

NOOBS hat sich inzwischen für eine bestimmte Bildschirmauflösung entschieden. Wenn diese Auflösung nicht zur Auflösung des Monitors passt, dann skalieren die meisten Monitore automatisch. Bei einem Monitor mit 1920 horizontalen Pixeln und einem HDMI/HDMI- oder HDMI/DVI-Kabel wird das NOOBS-Fenster aus Bild 1.29 in etwa nur 1/3 der Bildschirmbreite benötigen. Wenn das HDMI-Kabel zu spät angesteckt oder der Monitor zu spät eingeschaltet wurde, dann passt die von NOOBS verwendete Bildschirmauflösung nicht zum Monitor. Das führt zu Verzerrungen, und gegebenenfalls wird der untere Bildrand abgeschnitten. Sie werden NOOBS wahrscheinlich trotzdem benutzen können.

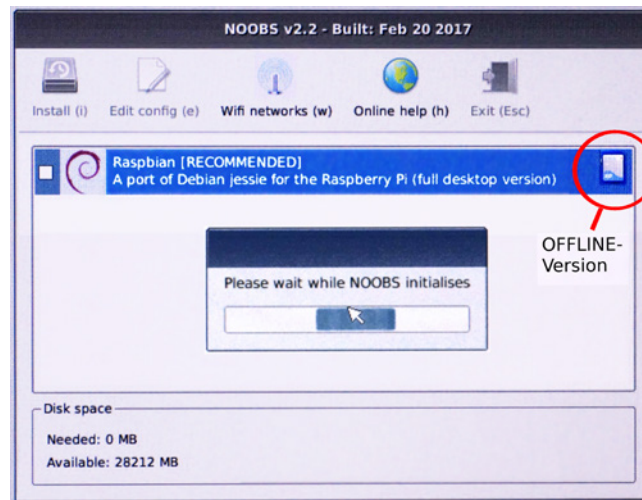


Bild 1.28 NOOBS-Initialisierung

NOOBS wird nun auf den Raspberry Pi Modellen mit Ethernet-Buchse versuchen, eine Internetverbindung aufzubauen. Wenn das Kabel fehlt, wird „Error downloading“ als Fehlermeldung erscheinen (Bild 1.29).

Bei „NOOBS offline and network“ stört diese Fehlermeldung aber nicht, denn Raspbian ist ja auf der (micro)SD-Karte vorhanden. Betriebssysteme, die nur online verfügbar sind, zeigt NOOBS durch eine Ethernet-Buchse an. Wenn Sie lediglich Raspbian benötigen, können Sie weitermachen und am unteren Bildschirmrand die Sprache einstellen. Ändern Sie zuerst die Sprache auf Deutsch, wie in Bild 1.30 zu sehen. Dadurch sollte auch das Tastaturlayout (Keyboard) automatisch von „gb“ auf „de“ springen. Die Menübeschriftung sollte nun deutschsprachig sein.

Bild 1.31 ist bei bestehender Internetverbindung entstanden, was zur ergänzenden Anzeige der online verfügbaren Betriebssysteme geführt hat. Beachten Sie bitte, dass „Raspbian LITE“ unter den Onlinebetriebssystemen auftaucht.

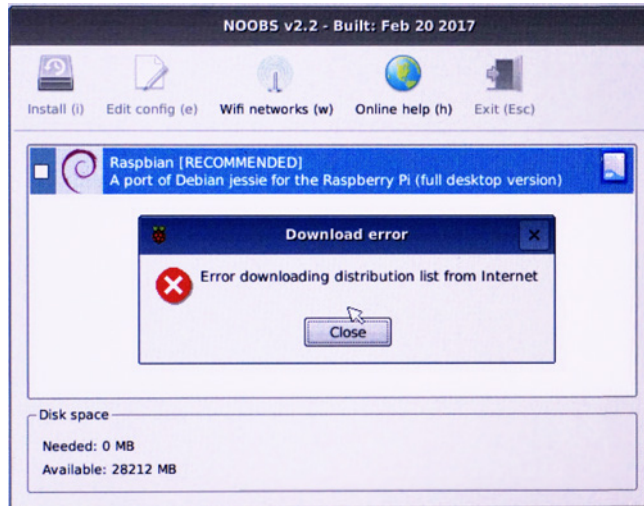


Bild 1.29 Fehlermeldung bei fehlender Internetverbindung



Bild 1.30 Umschaltung auf Deutsch

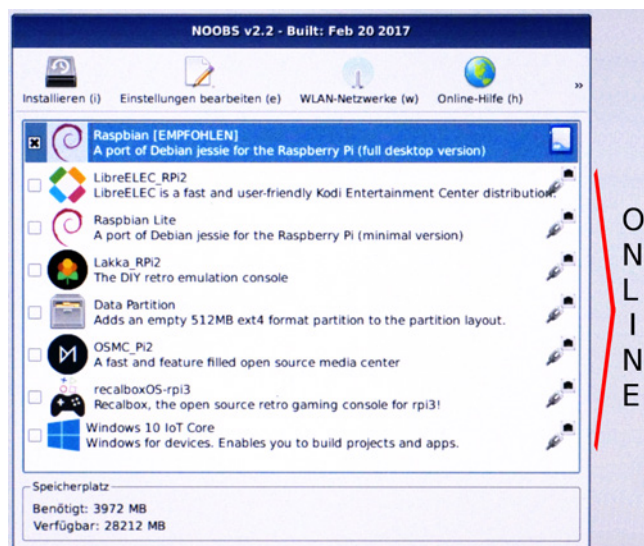


Bild 1.31 Betriebssysteme bei NOOBS 2.2 und bestehender Internetverbindung

Eine Boot-Weiche soll hier nicht wirksam werden. (Darauf wird in Abschnitt 1.6 eingegangen.) Markieren Sie daher bitte nur Raspbian, und betätigen Sie dann den Button „Installieren“.

Die Installation wird gestartet, und es erscheint eine Abfragebox, ob die SD-Karte gelöscht werden darf, was Sie bestätigen. Jetzt beginnt die Installation von Raspbian, die auf einem Raspberry Pi Zero W etwa 25 Minuten dauert, aber auf dem Pi 2 und Pi 3 wesentlich schneller ist.

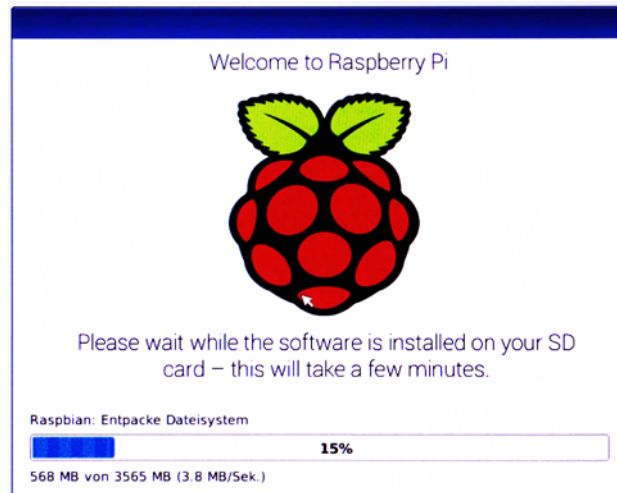


Bild 1.32 Fortschrittsbalken: NOOBS entpackt Raspbian.

Während der Installation sehen Sie eine prozentuale Fertigstellung (Bild 1.32) und darüber einige Infoseiten, die den Einstieg in die Arbeit mit dem Raspberry Pi erleichtern sollen. In Bild 1.32 werden 3,8 MB/s bei 3565 MB für einen Raspberry Pi 3 angegeben, was etwa 15 Minuten als Gesamtzeit entspricht. Wenn NOOBS seine Arbeit erledigt hat, erscheint die Meldung „Betriebssystem(e) erfolgreich installiert“, wie in Bild 1.33 zu sehen.

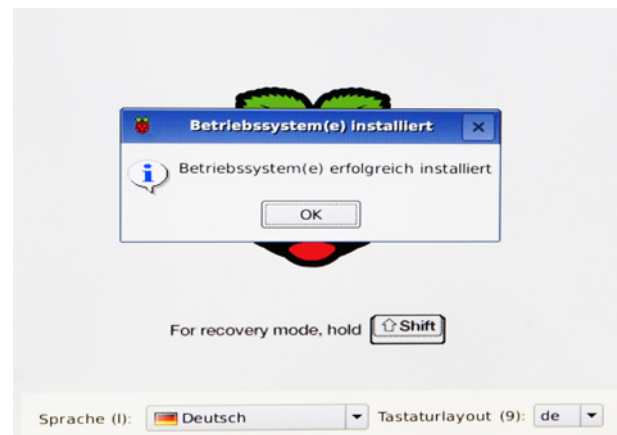
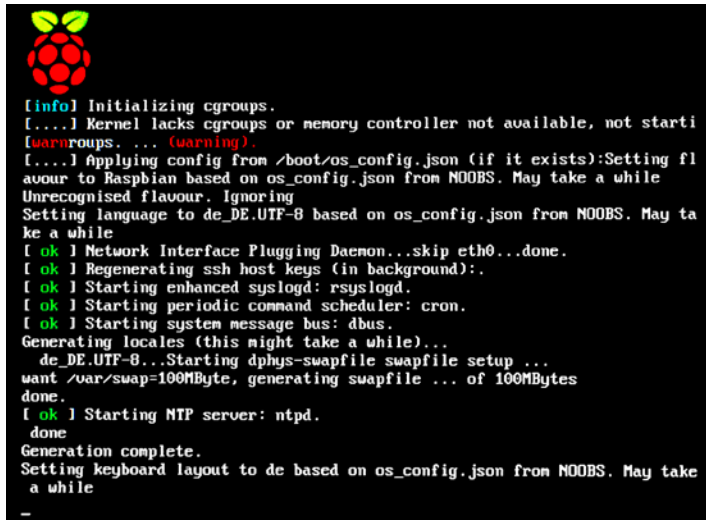


Bild 1.33 NOOBS meldet das Ende seiner Arbeit.

Nach einem Klick auf „OK“ bootet Raspbian Jessie erstmals, braucht aber noch etwas Zeit, um alles einzurichten.



```
[info] Initializing cgroups.
[... ] Kernel lacks cgroups or memory controller not available, not starting
[warning] ... (warning)
[... ] Applying config from /boot/os_config.json (if it exists): Setting flavour
to Raspbian based on os_config.json from NOOBS. May take a while
Unrecognised flavour. Ignoring
Setting language to de_DE.UTF-8 based on os_config.json from NOOBS. May take
a while
[ ok ] Network Interface Plugging Daemon...skip eth0...done.
[ ok ] Regenerating ssh host keys (in background):.
[ ok ] Starting enhanced syslogd: rsyslogd.
[ ok ] Starting periodic command scheduler: cron.
[ ok ] Starting system message bus: dbus.
Generating locales (this might take a while)...
de_DE.UTF-8...Starting dphys-swapfile swapfile setup ...
want /var/swap=100MByte, generating swapfile ... of 100MBytes
done.
[ ok ] Starting NTP server: ntpd.
done
Generation complete.
Setting keyboard layout to de based on os_config.json from NOOBS. May take
a while
_
```

Bild 1.34 Boot-Vorgang von Raspbian auf einem Raspberry Pi 1 (Systemkonsole)

- Wenn Sie „Raspbian LITE“ ausgewählt haben sollten, landen Sie nach dem Boot-Vorgang in der Systemkonsole (Bild 1.34). Sie werden dann üblicherweise das Programm `raspi-config` zur Konfiguration aufrufen und müssen von Anfang an auf die Tastatureinstellung achten. Die LITE-Version erfordert gute Linux-Kenntnisse.
- Bei „Raspbian (Full Desktop version)“ ist dagegen die grafische Oberfläche voreingestellt, und Sie erhalten einen „Welcome Screen“, wie in Bild 1.35 zu sehen.



Bild 1.35 Die grafische Oberfläche wurde umbenannt: „Welcome“ für den Raspbian Desktop

Das Ziel von NOOBS ist damit erreicht, denn kurz nach der Begrüßung erscheint die Benutzeroberfläche von Raspbian. Diese Oberfläche hatte eine Zeit lang den Namen PIXEL, wird nun aber schlicht als Raspberry Pi Desktop bezeichnet und mit RPi-Desktop abgekürzt. Die Konfiguration des Desktops und des Raspbian-Betriebssystems werden in Abschnitt 2.2 genauer beschrieben.

■ 1.6 Der NOOBS Recovery-Modus

Unter „Recovery“ könnte man verstehen, dass ein fehlerhafter Zustand durch einen besseren Zustand ersetzt wird. Erwarten Sie dabei aber bitte nicht, dass unlesbar gewordene Dateien funktionsfähig gemacht werden und der Rest der Daten unangetastet bleibt! Der Recovery-Modus ist keine „Zaubertaste“, die Ihren Raspberry Pi automatisch wiederbelebt. Sie müssen also selbst wissen, wo und wann man eingreifen kann und soll. Im NOOBS-Recovery-Modus gibt es folgende Möglichkeiten:

1. Sie können hierüber die Konfiguration eines schon installierten Betriebssystems ändern, indem Sie die Dateien *config.txt* oder *cmdline.txt* verändert speichern. (Nur zu empfehlen, wenn Sie wissen, was Sie tun!)
2. Sie können hierüber eine Verbindung zu einem WLAN-Netzwerk aufbauen.
3. Sie können ein oder mehrere Betriebssysteme komplett neu installieren, indem Sie die Betriebssysteme ankreuzen und dann „Install“ drücken. (Vorhandene Betriebssysteme gehen dabei verloren!) Sie brauchen dafür gegebenenfalls eine Internetverbindung, noch bevor Sie in den Recovery-Modus gehen.

Der „Recovery“-Modus ist ungeeignet für Einsteiger, die sich eine NOOBS SD-Karte gekauft und noch nie ein Backup (siehe Abschnitt 3.4) durchgeführt haben. Wenn Sie dagegen ein funktionsfähiges Backup haben, dürfen Sie mutig herangehen.

Wie gelangt man in den „Recovery“-Modus? Bei jedem Neustart von der NOOBS-SD-Karte erscheint kurz eine Himbeere auf dem Bildschirm, wie in Bild 1.36 zu sehen.



Bild 1.36 NOOBS wird mit der Shift-Taste betreten.

Sie sehen die Himbeere nicht länger als 1 Sekunde. Wenn Sie während des Boot-Vorgangs die Shift-Taste gedrückt halten, betreten Sie den „Recovery Mode“ von NOOBS und müssen abwarten, bis die Box „NOOBS initialisiert“ verschwindet.

Wenn Sie auf „Einstellungen bearbeiten“ klicken, dann geht ein Fenster „Konfigurationseditor“ auf, das auch mit der Maus bedienbar ist (Bild 1.37). Darin sehen Sie die Dateien *config.txt* und *cmdline.txt*, in denen die Boot-Konfiguration des schon vorhandenen Raspbian-Betriebssystems abgelegt ist. Wenn Sie die Bedeutung der jeweiligen Einträge kennen, können Sie z.B. die Audioausgabe über HDMI aktivieren.

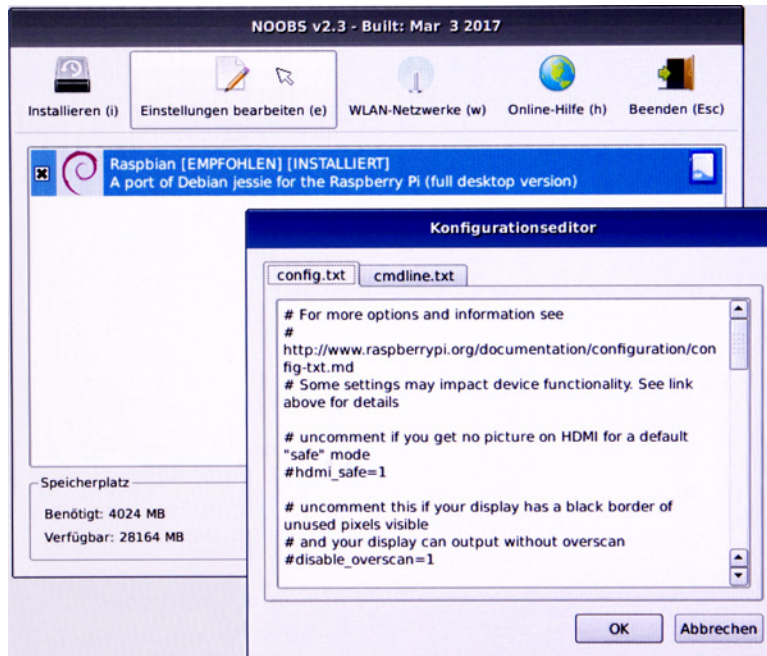


Bild 1.37 Über NOOBS die Einstellungen von Raspbian Jessie ändern

Über den Button „WLAN-Netzwerke“ stellen Sie eine Internetverbindung für den Fall her, dass Ihr Raspberry Pi keine Ethernet-Buchse hat. Um sich mit einem WLAN-Netzwerk zu verbinden, müssen Sie dessen Passwort kennen und eingeben, wie in Bild 1.38 zu sehen.

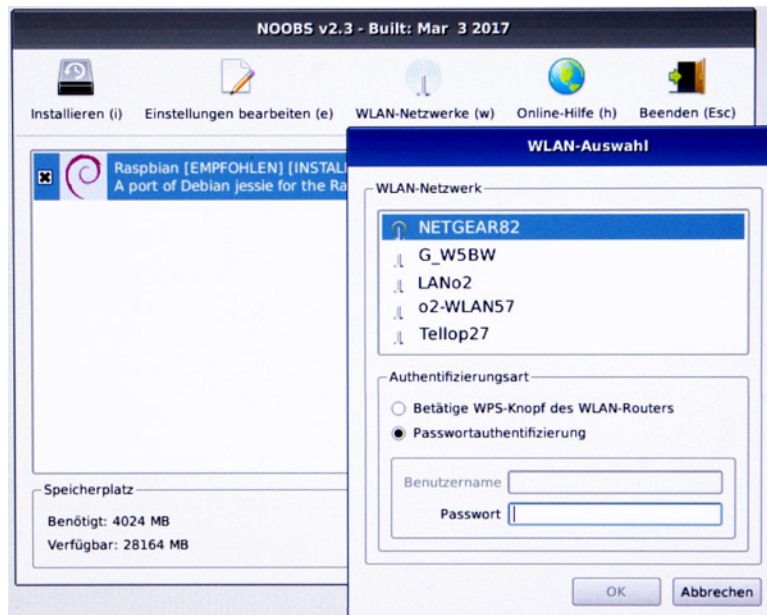


Bild 1.38 Aufbau einer WLAN-Verbindung mit NOOBS

Der Button „Installieren“ ist viel tiefgreifender und verändert die gesamte (micro)SD-Karte. Von Raspbian abgesehen, müssen bei NOOBS 2.3 alle Betriebssysteme über eine Internetverbindung nachgeladen werden. NOOBS Offline+Network vom 17.8.2017 enthält jedoch auch LibreELEC. Die Internetverbindung muss schon bestehen, damit alle Betriebssysteme aus Bild 1.39 erscheinen.

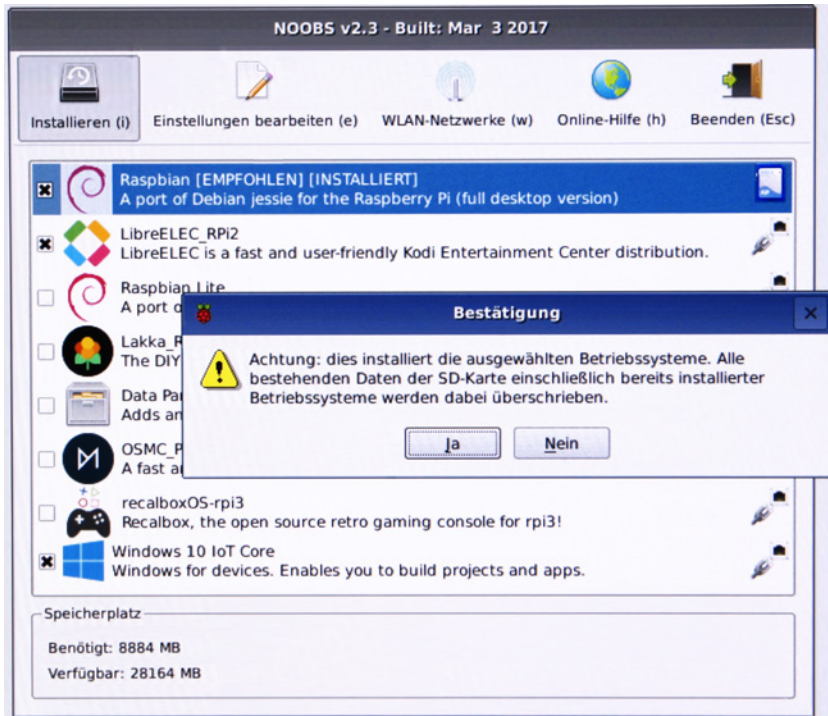


Bild 1.39 Raspbian, LibreELEC und Windows 10 IoT neu installieren

Es ist aber mit der Version 2.3 nicht möglich, ein Betriebssystem zu ergänzen. Die Einstellungen aus Bild 1.39 führen dazu, dass die SD-Karte gelöscht wird und die markierten Betriebssysteme (Raspbian, LibreELEC, Windows 10 IoT) neu installiert werden. In der Bestätigungsbbox wird darauf hingewiesen, dass das installierte Raspbian überschrieben wird. Wenn Sie „Ja“ anwählen, wird installiert, und Sie werden bei allen späteren Bootvorgängen eine Bootweiche für die markierten Betriebssysteme sehen.

2

Der Raspberry Pi und Linux

Wer Linux auf dem PC kennenlernen möchte, kann sich ein Computerheft kaufen, dem eine sogenannte Live-DVD mit mehreren Linux-Distributionen beigelegt ist (siehe Bild 3.1). Manchmal enthält solch eine DVD aber nicht alle der angegebenen Betriebssysteme in direkt bootbarer Form. Stattdessen könnte die DVD z. B. folgende Datei enthalten: *linuxmint-18.1-cinnamon-32bit-de-20161215.iso*

Solch eine Datei erhalten Sie auch bei einem Download des Betriebssystems aus dem Internet. Oft gibt es auch eine Onlineanleitung, die das Betriebssystem und den Weg beschreibt, auf dem aus der ISO-Datei ein Boot-Medium wird (https://www.linuxmint.com/documentation/user-guide/Cinnamon/german_18.1.pdf).

Wobei das zukünftige Bootmedium für den PC entweder eine DVD oder ein USB-Stick sein kann. Es gibt recht viele Linux-Distributionen, die aber immer einen verhältnismäßig großen Stamm an gemeinsamen Operationen haben. Mehr dazu finden Sie auf folgender Webseite: <https://de.wikipedia.org/wiki/Linux-Distribution>

Auf der Ebene einfacher Konsolenkommandos wie `cat`, `ls` oder `lsusb` gibt es faktisch keine Unterschiede. Linux könnte nicht existieren, wenn es keine Organisation gäbe, die entscheidet, welche Programme in eine neue Distribution aufgenommen werden bzw. schon beim ersten Start des Linux-Betriebssystems enthalten sind. Diese Entscheidung ist bei jedem Linux zweistufig. So können Sie sich vorstellen, dass eine Linux-Distribution 100 Programme hat, die schon beim ersten Start verfügbar sind. Es gibt aber noch die sogenannten Repositories (ein Internet-Archiv). Dort finden Sie z. B. 3000 weitere Programme, die als für Linux Mint zugelassen gelten.

So ist der Quelltexteditor Geany in Linux Mint 17 nicht enthalten, aber problemlos nachinstallierbar. Geany wird in Abschnitt 4.13 als vorinstalliertes Programm von Raspbian Jessie beschrieben. Die Vorgängerversion Raspbian Wheezy enthielt Geany noch nicht. Die Gemeinsamkeit ist, dass sowohl Raspbian als auch Linux Mint von Debian abstammen. Somit „erbt“ Raspbian Jessie von Debian 8. Diese Verwandtschaftsbeziehung entspricht auch der Wortkomposition Raspbian (siehe Bild 1.18).

■ 2.1 Das Abbild von Raspbian für die SD-Karte

Auf dem Raspberry Pi bilden NOOBS und Raspbian fast nur verschiedene „Verpackungen“ für dasselbe Betriebssystem, während sich der auf Debian basierende „RASPBERRY PI DESKTOP“ grundlegend davon unterscheidet und nach dem Download zu einer *.iso-Datei führt. Gehen Sie dazu auf folgende Webseite: <http://www.raspberrypi.org/downloads>.

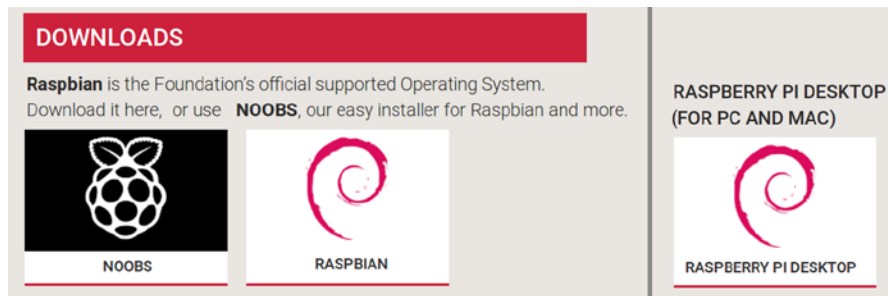


Bild 2.1 NOOBS, RASPBIAN oder die x86-Version?

In Kapitel 1 wurde lediglich NOOBS verwendet und beschrieben. Wo liegt der Unterschied zum Download von RASPBIAN? Die Schreibweise „RASPBIAN“ steht hier für eine Abbilddatei, die Sie sich per Download holen. Das lauffähige Betriebssystem „Raspbian“ fängt dagegen erst mit dem ersten Bootvorgang zu existieren an. Da NOOBS ein Installationsprogramm ist, das neben Raspbian weitere Betriebssysteme enthalten kann, unterscheidet sich das Endresultat bzw. die bootfähige (micro)SD-Karte. Zwar beziehen sich NOOBS und RASPBIAN aus Bild 2.1 fast immer auf exakt dieselbe Raspbian-Version, aber die Aufteilung der SD-Karte in Partitionen wird differieren.

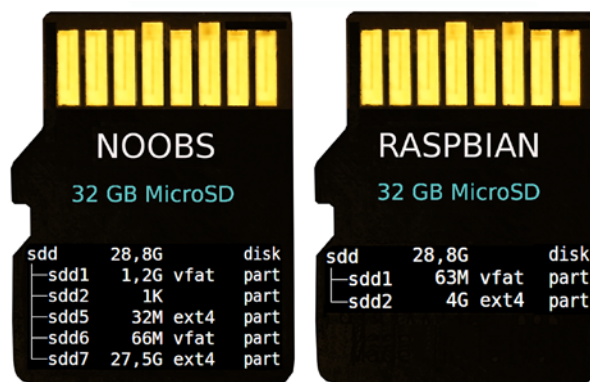


Bild 2.2 Partitionen bei NOOBS und Raspbian (lsblk)

In Bild 2.2 sehen Sie, dass ein lauffähiges Raspbian lediglich zwei Partitionen benötigt. NOOBS braucht da mehr, denn es ist dafür gedacht, „aus sich selbst heraus“ neue Betriebssysteme zu

bekommen, wie es in Bild 1.39 beim Recovery angedeutet wird. Oft ist ein Multiboot von der (micro)SD-Karte aber gar nicht erwünscht, und lediglich zwei Partitionen sind ein Vorteil. Aber das Thema Partitionen kann noch ein wenig warten ...

Sie haben eine frische (micro)SD-Karte und wollen daraus ein im Raspberry Pi verwendbares Bootmedium machen. Von Bild 2.1 ausgehend, wählen Sie nun „RASPBIAN“ im Internetbrowser Ihres PCs, was zu Bild 2.3 als Verzweigung vor dem Download führt.



Bild 2.3 Raspbian Stretch mit/ohne grafische Oberfläche

Auf diesem Weg erhalten Sie immer die aktuellste Version. Auf ältere Raspbian-Versionen greifen Sie wie in Abschnitt 1.3 beschrieben zu. Jetzt müssen Sie sich für oder gegen die grafische Oberfläche entscheiden, wobei Sie eine kleine SD-Karte mit lediglich 4 GB zur Version LITE zwingen würde. Wer einen Raspberry Pi 2 oder Pi 3 hat, kann sich klar für die Desktopversion entscheiden. Das gilt auch für Einsteiger, die einen leistungsschwächeren Pi Zero oder Pi 1 haben. Da der Einstieg in Linux bzw. Raspbian mit Bedienoberfläche leichter fällt, soll hier die Desktopversion verwendet werden. Nach dem Download von „RASPBIAN STRETCH WITH DESKTOP“ erhalten Sie z. B. folgende Datei: *2018-03-13-raspbian-stretch.zip*

Eine LITE-Version wäre z. B. für einen Raspberry Pi Zero W interessant, der mobil sein soll und deshalb kein Kabel am HDMI-Anschluss hat (→ HDMI ON/OFF, siehe Listing 4.2). Nur die Version mit DESKTOP wird später beim Boot-Vorgang auf den Desktop durchstarten, und soll deshalb Basis der weiteren Beschreibung sein.

Die *.zip-Datei muss nun entpackt werden, woraus eine *.img-Datei entsteht. Viele Dateimanager (z. B. Dolphin oder Krusader bei Linux, Total Commander bei Windows) enthalten Tools wie 7-Zip zum Entpacken (siehe Bild 2.4). Falls notwendig, können Sie sich 7-Zip von folgender Website holen: <http://www.7-zip.org>

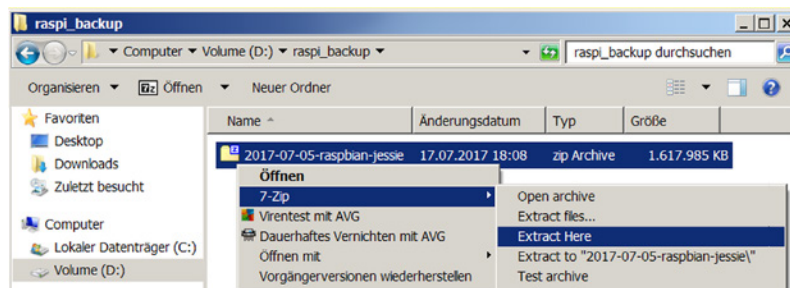


Bild 2.4 Entpacken der *.zip-Datei im Explorer durch 7-Zip

Für den Mac eignet sich The Unarchiver als Entpackprogramm. Bei Linux genügt folgende Kommandozeile:

```
sudo apt-get install p7zip-full
```

Das entstandene Image (oder Abbild) des Betriebssystems (*.img) wird dann mit einem Transferprogramm auf die (micro)SD-Karte geschrieben. Aber werfen wir zuerst noch ein Blick auf die Größenverhältnisse:

Datei:	Bytes	Datum/Zeit
2017-07-05-raspbian-jessie.zip	1656816607	2017-07-17 18:08
2017-07-05-raspbian-jessie.img	4661483821	2017-07-05 13:02
2017-08-16-raspbian-stretch.zip	1759433927	2017-08-18 19:32
2017-08-16-raspbian-stretch.img	4907675648	2017-08-16 11:35

Die entpackte Abbilddatei ist hier etwa um den Faktor 2,8 größer als die ZIP-Datei. Klar erkennbar ist auch, dass Sie mindestens eine 8 GB (micro)SD-Karte benötigen werden, um dort das Raspbian-Abbild unterzubringen. Auf der Festplatte Ihres „großen“ PCs muss natürlich auch genug Platz vor dem Entpacken frei sein.



Hinweis:

Beachten Sie bitte, dass sich Dateien mit über 4 GB nicht auf einen USB-Stick mit FAT32-Formatierung kopieren lassen. Der Vorgang wird bei Überschreitung des Limits abgebrochen. Das durch FAT32 vorgegebene Limit gilt auch, wenn Sie versuchen, eine *.zip-Datei zu entpacken, deren enthaltene Abbilddatei über dem Limit liegt.

Das Transferprogramm wird auf der (micro)SD-Karte als Zielmedium alle notwendigen Partitionen erzeugen, ebenso alle Verzeichnisse und Dateien des Betriebssystems. Für Raspbian Jessie/Stretch entstehen so zwei Partitionen, wie in Bild 2.2 zu sehen.

In der Praxis ist folgende Vorgehensweise für einen Linux-PC zu empfehlen:

1. Sie formatieren die (micro)SD-Karte (siehe Bild 3.9) und entfernen sie aus dem Kartenleser.
2. Dann downloaden Sie *2018-03-13-raspbian-stretch.zip* (oder eine neuere Version) mit einem „großen“ PC und entpacken die Datei.
3. Verbinden Sie die (micro)SD-Karte über einen Kartenleser mit dem PC.
4. Starten Sie ein Transferprogramm, wählen Sie die *.img-Datei aus, und „flashen“ Sie die (micro)SD-Karte.
5. Nach dem Schreibvorgang ziehen Sie den USB-Kartenleser ab.
6. Stecken Sie den USB-Kartenleser an einen Linux-PC an.
7. Kopieren Sie nun die Datei zum Buch *ProgPi.zip* innerhalb der ext4-Partition der SD-Karte nach `/home/pi`.
8. Auf dem Linux-PC lösen Sie ein „sicheres Entfernen“ der SD-Karte im Kartenleser aus.
9. Entnehmen Sie die (micro)SD-Karte, stecken Sie sie in den Raspberry Pi, und booten Sie.

Wenn Sie das Beschreiben der (micro)SD-Karte mit einem Windows-PC oder Mac durchführen, werden Sie die ext4 Partition aus Bild 2.2 nicht sehen können. Das ist eigentlich nicht weiter tragisch, verhindert aber, dass Sie *ProgPi.zip* schon jetzt im passenden Verzeichnis ablegen können.

Je nach Betriebssystem Ihres PCs können die in der folgenden Tabelle dargestellten Programme für das Formatieren der SD-Karte und das „Flashen“ der SD-Karte mit *.img verwendet werden.

Programm	Formatieren	Restore	Backup	Betriebssystem
USB-Stick-Formatierer	ja	nein	nein	Linux Mint
SDFormatter V4	ja	nein	nein	Windows
USB-Abbilderstellung	nein	ja	nein	Linux Mint
sd-restore	nein	ja	nein	Linux
Win32 DiskImager	nein	ja	bedingt	Windows
USB Image-Tool	nein	ja	ja	Windows
Etcher	nein	ja	nein	Windows/Mac/Linux

In der Regel werden Sie die hier tabellierten Programme auf Ihrem großen PC bei angestecktem Kartenleser ausführen. Die Liste erhebt keinen Anspruch auf Vollständigkeit. Fast jede Linux-Distribution enthält bereits ein Programm zum Flashen bzw. um das „Image“ zu übertragen. Die „USB-Abbilderstellung“ tut das übrigens auch für die zu Beginn des Kapitels erwähnten *.iso-Dateien. Programme, die einen Boot-USB-Stick erstellen können, sind meist auch fähig, eine SD-Karte mit dem Raspbian-Betriebssystem zu versehen. Ein Restore ist übrigens nichts anderes als der Transfer einer *.img-Abbilddatei auf das zukünftige Bootmedium (micro)SD-Karte.

Das Skript `sd-restore.sh` hat den Vorteil, dass Sie sich das Entpacken sparen können, da das Skript auch die *.zip-Datei verwenden kann:

```
pi@raspi3:~/ProgPi/backup $ ./sd-restore.sh
Vorhandene Abbilddateien:
 0 /media/pi/Lexar/raspi_images/2017-08-16-raspbian-stretch.zip
 1 /media/pi/Lexar/raspi_images/2017-08-16-raspbian-stretch-lite.img
 2 /media/pi/Lexar/raspi_images/stretch17_10_171128_081040_7368.img.gz
Bitte Nummer der Datei eingeben : 0
```

Das Skript `sd-restore.sh` ist zwar nicht auf einem Windows-/Mac-PC verwendbar, aber auf jedem beliebigen Linux-PC. Wenn ein Raspberry Pi für den Restorevorgang verwendet wird, muss ein externer Kartenleser mit leerer, zu beschreibender SD-Karte und ein USB-Stick mit den *.zip-Dateien angeschlossen sein. Dieser Weg soll hier aber nicht fortgesetzt werden, da der Restorevorgang mit `sd-restore.sh` genauer in Abschnitt 3.9 geschildert wird.

Bevor Sie eine SD-Karte mit Raspbian Jessie versehen, können Sie übrigens auch einen Blick in die entpackte *.img-Abbilddatei werfen. Das dazu in Abschnitt 3.11 angewandte Skript nutzt ein virtuelles Laufwerk für diesen Zweck.

In den folgenden beiden Abschnitten werden Programme mit grafischer Oberfläche für den Restorevorgang (bzw. das Flashen der SD-Karte) verwendet.

2.1.1 Die Abbilddatei mit einem Linux-PC übertragen

Nachdem die (micro)SD-Karte mit ext4 formatiert ist (siehe Bild 1.20), erfolgt im nächsten Schritt die Übertragung des Betriebssystems als *.img-Datei. Bei Linux Mint kann das mit dem Programm USB-Abbilderstellung erfolgen (Bild 2.5). Bei Kubuntu können Sie den Startmedien-ersteller verwenden. Wenn Sie bei Ihrer Linux-Distribution kein passendes Programm finden, können Sie auf das schon kurz beschriebene Skript `ProgPi/backup/sd-restore.sh` zurückgreifen.

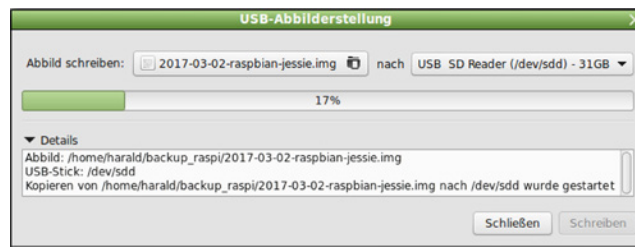


Bild 2.5 Übertragung der Abbilddatei auf eine SD-Karte

Während der Übertragung blinkt die Leuchtdiode am Kartenleser. Nach einigen Minuten ist das Schreiben der Abbilddatei beendet, und die (micro)SD-Karte kann aus dem Kartenleser am Desktop-PC oder Laptop entnommen und in den Raspberry Pi eingelegt werden.

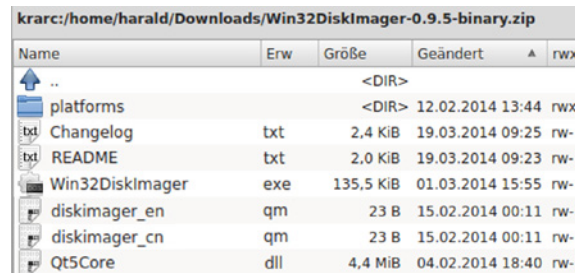
Zuerst sollten Sie aber noch auf die frisch präparierte SD-Karte schauen. Da sich die Partitionen verändert haben, ist es sinnvoll, den Kartenleser abzuziehen und dann wieder anzustechen. Nachdem das geschehen ist, gehen bei den meisten Linux-Distributionen zwei Fenster des Dateimanagers auf. Die Fenster sind mit „boot“ und „root“ betitelt, und Sie sehen darin die Dateien der jeweiligen Partition. Die Größe der ext4-Partition mit dem Namen „root“ entspricht jetzt ziemlich genau der Größe der Abbilddatei. Der erste Bootvorgang im Raspberry Pi wird das ändern.

2.1.2 Der Win32 Disk Imager

Wie wird aus einer formatierten SD-Karte ein Boot-Medium für den Raspberry Pi? Unter Windows geschieht das z. B. mit dem Win32 Disk Imager. Sie bekommen das Programm über eine der folgenden Webseiten:

- <https://wiki.ubuntu.com/Win32DiskImager>
- <http://sourceforge.net/projects/win32diskimager/files/latest/download>

Sie entpacken die Datei *Win32DiskImager-0.9.5-binary.zip* auf einem Windows-PC und erhalten ein Verzeichnis, in dem neben *README.TXT* und mehreren DLL-Dateien vor allem *Win32DiskImager.exe* liegt (Bild 2.6).



Name	Erw	Größe	Geändert	▲	rwx
..					<DIR>
platforms			12.02.2014 13:44		rwx
Changelog	txt	2,4 KiB	19.03.2014 09:25		rw-
README	txt	2,0 KiB	19.03.2014 09:23		rw-
Win32DiskImager	exe	135,5 KiB	01.03.2014 15:55		rw-
diskimager_en	qm	23 B	15.02.2014 00:11		rw-
diskimager_cn	qm	23 B	15.02.2014 00:11		rw-
Qt5Core	dll	4,4 MiB	04.02.2014 18:40		rw-

Bild 2.6 Einige Dateien von *Win32DiskImager-0.9.5-binary.zip*

Diese EXE-Datei ist ohne weitere Installation lauffähig und ermöglicht die direkte Auswahl einer Image-Datei und eines Geräts am USB-Port. Der Win32 Disk Imager kann mit „Write“ ein Medium am USB-Port beschreiben oder mit „Read“ ein Betriebssystem als Datei sichern. Ein per Internetdownload geholtes Betriebssystem zum Raspberry Pi wird zuerst entpackt (siehe Bild 2.4).

Sobald Sie aus der *.zip-Datei eine *.img-Datei gemacht haben, kann die Übertragung mit dem Win32 Disk Imager auf eine leere SD-Karte beginnen (Bild 2.7).

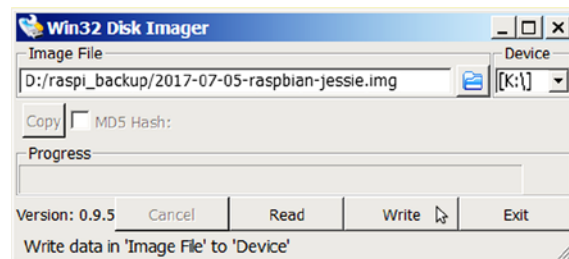


Bild 2.7 Transfer einer *.img-Datei über den *Win32DiskImager*

Als Zielmedium werden beim Win32 Disk Imager z.B. die Laufwerksbuchstaben H: und K: angeboten. Dabei kann H: ein USB-Stick sein, auf den z.B. die ISO-Datei von Linux Mint übertragen werden könnte. K: wäre eine zu beschreibende SD-Karte. Hier ist natürlich Vorsicht geboten, um keine Festplatte zu überschreiben.

Sobald Sie den Button Write drücken, wird das Zielmedium nach Rückfrage überschrieben – ganz unabhängig davon, wie viele Partitionen darauf enthalten sind. Der Schreibvorgang überträgt ein ganzes Betriebssystem. Sie brauchen auch bei Verwendung von USB 3.0 etwas Geduld. Sobald eine „Ready“-Meldung erscheint, kann die (micro)SD-Karte entnommen und in den noch nicht mit dem Netzteil verbundenen Raspberry Pi eingesetzt werden.

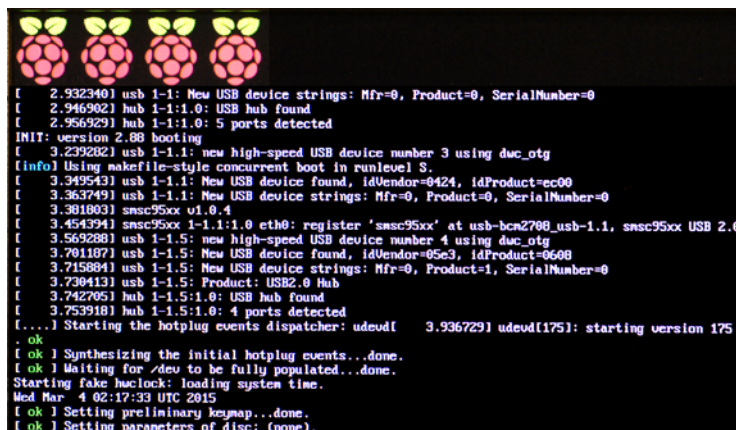
■ 2.2 Die Grundkonfiguration des Raspbian Desktops

Wenn Sie die (micro)SD-Karte mit dem Raspbian-Betriebssystem selbst erstellt haben, dann haben Sie sich für einen der folgenden Wege entschieden:

- NOOBS
- NOOBS LITE
- RASPBIAN JESSIE WITH DESKTOP (früher: WITH PIXEL)
- RASPBIAN JESSIE LITE
- RASPBIAN STRETCH WITH DESKTOP
- RASPBIAN STRETCH LITE

Jeder Boot-Vorgang auf dem Raspberry Pi beginnt mit der Anzeige von mindestens einer Himbeere. Vier Himbeeren entsprechen den vier Prozessorkernen auf einem Raspberry Pi 2 oder Pi 3. Der erste Bootvorgang verändert auch die Partitionen der SD-Karte. Bei RASPBIAN entspricht die ext4-Partition vor dem ersten Booten der Größe des Image (siehe Bild 2.2). Danach ist die ext4-Partition auf die gesamte SD-Karte expandiert.

Die LITE-Versionen arbeiten ohne grafische Oberfläche, was Platz auf der SD-Karte spart. Folglich gibt es bei LITE erheblich weniger an vorinstallierten Programmen. Der erste Bootvorgang muss deshalb zur Systemkonsole führen. Da alle Kommandos für die Systemkonsole auch in einem Terminalprogramm und in der grafischen Oberfläche eingegeben werden können, wird die Systemkonsole hier nicht näher beschrieben. Die Systemkonsole (Bild 2.8) nimmt immer den gesamten Bildschirm in Anspruch. In Bild 2.10 sehen Sie dagegen ein Terminalfenster.



```
[ 2.932340] usb 1-1: New USB device strings: Mfr=0, Product=0, SerialNumber=0
[ 2.946902] hub 1-1:1.0: USB hub found
[ 2.956929] hub 1-1:1.0: 5 ports detected
INIT: version 2.88 booting
[ 3.239282] usb 1-1.1: new high-speed USB device number 3 using dwc_otg
[info] Using makefile-style concurrent boot in runlevel S.
[ 3.349543] usb 1-1.1: New USB device found, idVendor=0424, idProduct=ec00
[ 3.363749] usb 1-1.1: New USB device strings: Mfr=0, Product=0, SerialNumber=0
[ 3.381803] smsc95xx v1.0.4
[ 3.454394] smsc95xx 1-1.1:1.0 eth0: register 'smsc95xx' at usb-bcm2708_usb-1.1, smsc95xx USB 2.0
[ 3.569288] usb 1-1.5: new high-speed USB device number 4 using dwc_otg
[ 3.701107] usb 1-1.5: New USB device found, idVendor=05e3, idProduct=0608
[ 3.715894] usb 1-1.5: New USB device strings: Mfr=0, Product=1, SerialNumber=0
[ 3.738413] usb 1-1.5: Product: USB2.0 Hub
[ 3.742705] hub 1-1.5:1.0: USB hub found
[ 3.753918] hub 1-1.5:1.0: 4 ports detected
[....] Starting the hotplug events dispatcher: udevd[ 3.936729] udevd[175]: starting version 175
.ok
[ ok ] Synthesizing the initial hotplug events...done.
[ ok ] Waiting for /dev to be fully populated...done.
Starting fake huclock: loading system time.
Wed Mar 4 02:17:33 UTC 2015
[ ok ] Setting preliminary keymap...done.
[ ok ] Setting parameters of disc: (none).
```

Bild 2.8 Booten mit dem Raspberry Pi 2 (Systemkonsole)

Während des Bootvorgangs zeigt RASPBIAN STRETCH WITH DESKTOP kurz ein „Welcome to“ als Begrüßung an (siehe Bild 1.35). Bald danach erscheint die Oberfläche des Betriebssystems.

Wie bei der LXDE-Oberfläche wird am oberen Bildschirmrand das Hauptmenü eingeblendet. Als Hintergrundbild wird derzeit *road.jpg* verwendet. Zunächst gelten noch die englischsprachigen Einstellungen, was an der Benennung „Wastebasket“ für das Papierkorb-Icon erkennbar ist (Bild 2.9).

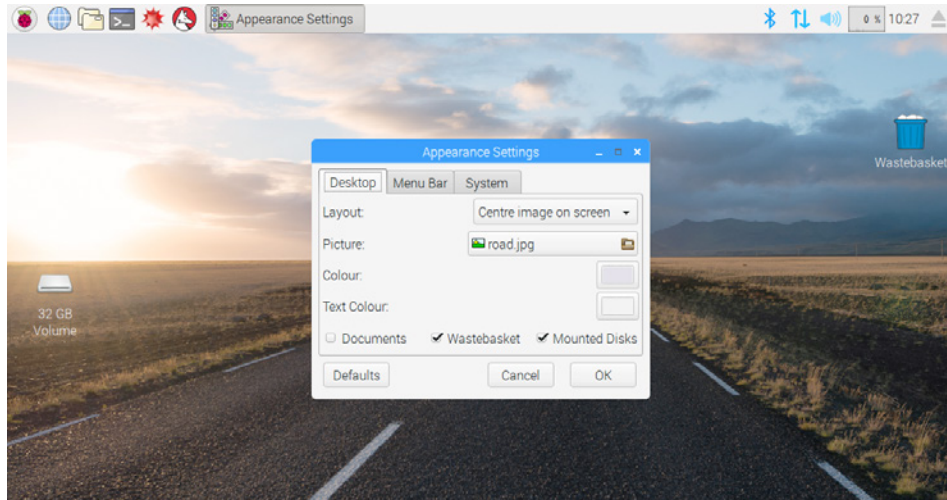


Bild 2.9 Der Raspbian Desktop mit *road.jpg* als Hintergrundbild

Es gilt nun folgende Fallunterscheidung:

- Wenn Sie über NOOBS zum RPi-Desktop gelangt sind, und in NOOBS die Spracheinstellung „Deutsch“ aktiv war, dann wird nach einem Reboot automatisch Deutsch für den Desktop wirksam.
- Bei RASPBIAN JESSIE/STRETCH WITH DESKTOP muss die Spracheinstellung dagegen noch manuell erfolgen.

Wenn Sie eine homogene Hintergrundfarbe oder ein anderes Hintergrundbild verwenden möchten, dann gehen Sie zuerst zum Hauptmenü (Bild 2.11) und dann auf „Appearance Settings“.

2.2.1 Tastaturlayout und Spracheinstellung

Nachdem Raspbian erstmals gebootet hat, ist üblicherweise ein „AutoLogin“ voreingestellt. Die Passwordeingabe ist dadurch abgeschaltet, und Sie wissen vielleicht noch nicht, welches Tastaturlayout nun gilt. Tastaturlayout und Spracheinstellungen lassen sich sehr leicht überprüfen, indem Sie ein LXTerminal starten und die Kommandos aus Bild 2.10 eingeben. Sie sehen am Resultat von `setxkbmap -query` dass ein britisches Tastaturlayout die Voreinstellung ist. Es musste also „b“ für „-“ und „z“ für „y“ getippt werden. Auch die Sprache steht auf „UK“, was an „Wastebasket“ erkennbar ist.

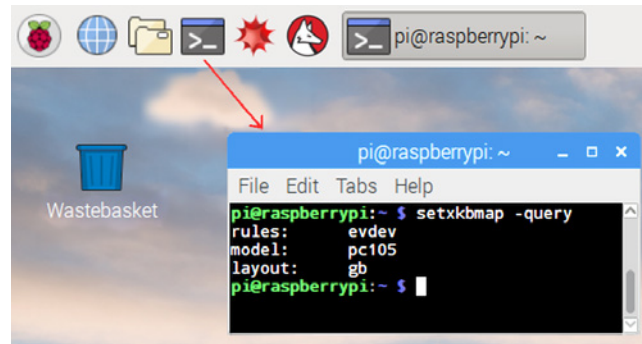


Bild 2.10 Überprüfung des gültigen Tastaturlayouts

Als Zwischenlösung können Sie folgende Kommandos eingeben, um ein deutsches Tastaturlayout einzustellen und zu überprüfen:

```
pi@raspberrypi ~ $ sudo setxkbmap de
pi@raspberrypi ~ $ setxkbmap -query
rules:    evdev
model:    pc105
layout:   de
```

Die Umschaltung der Tastatur wird so aber nur bis zu einem Reboot gültig bleiben. Das ist besonders bei der Passwortheingabe und beim Login zu beachten. Probieren Sie also ein „raspberrz“, wenn Sie sich mit „raspberrry“ nicht einloggen können. Doch noch ist der Autologin wirksam, so dass Sie kein Passwort eingeben müssen.

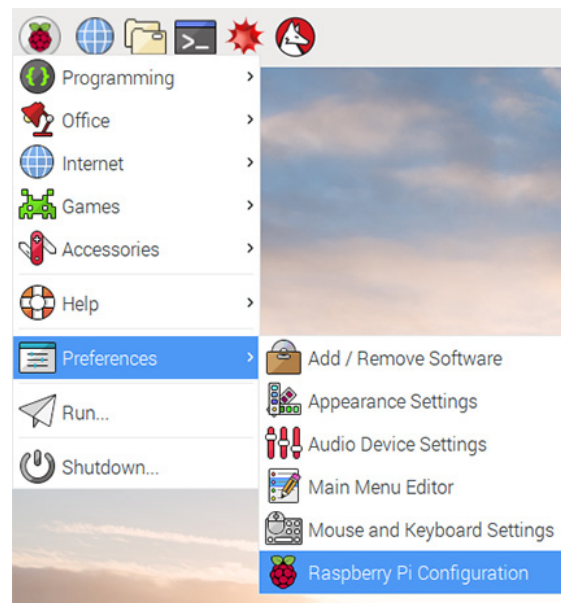


Bild 2.11 Das Hauptmenü mit Raspberry Pi Configuration

Wenn Sie Raspbian schon etwas länger kennen, haben Sie sicher schon das Konsolenprogramm `raspi-config` benutzt. Das Skript `raspi-config` ist bei einer Konfiguration ohne Desktop unter anderem für die richtige Sprache und ein passendes Tastaturlayout verantwortlich. Raspbian enthält inzwischen auch die GUI-Programme Add/Remove Software (siehe Abschnitt 2.10.1) und Raspberry Pi Configuration.

Wenn Sie für Raspberry Pi Configuration eine Desktop-Verknüpfung erstellen (rechte Maustaste im Menü), können Sie sehr leicht `rc_gui` als Namen des ausführbaren Programms ermitteln. Der Dialog „Dateieigenschaften“ erscheint, wenn Sie das neu erschaffene Himbeer-Icon mit der rechten Maustaste anklicken (Bild 2.12).

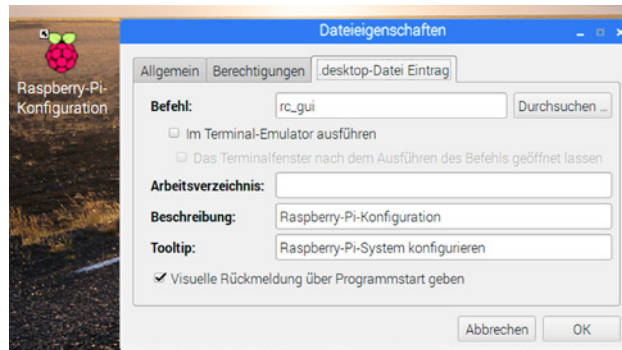


Bild 2.12 Eine Verknüpfung zu `rc_gui` durch `rc_gui.desktop`

Für die Benennung von Programmen gibt es drei Perspektiven:

- die Icon-Beschriftung (hier: Raspberry-Pi-Konfiguration)
- der Name der Desktop-Datei (hier: `rc_gui.desktop`)
- der Programmname selbst (hier: `rc_gui`)

Der Unterschied ist wichtig, weil die Icon-Beschriftung von der eingestellten Sprache abhängt.

Nach dem Start von `rc_gui` über *Preferences > Raspberry Pi Configuration* sehen Sie den in Bild 2.13 dargestellten (noch englischsprachigen) Dialog.

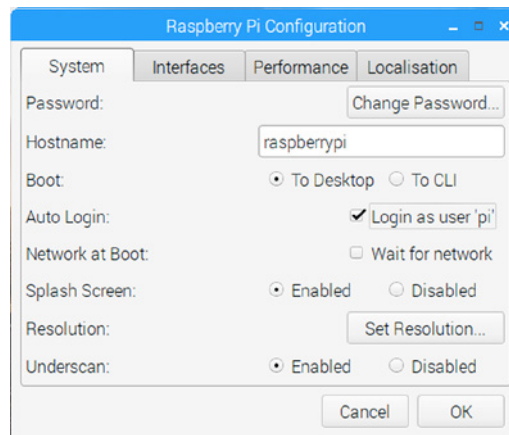


Bild 2.13 Das Programm Raspberry Pi Configuration

Das Programm `rc_gui` enthält im Wesentlichen dieselben Einstellmöglichkeiten, die auch `raspi-config` hat. Unter „System“ kann die Bildschirmauflösung („Set Resolution“) verstellt werden, was aber auch über `raspi-config` möglich ist (siehe Abschnitt 2.11).

- Sie sind beim Booten automatisch zur grafischen Oberfläche gelangt, weil „To Desktop“ aktiviert ist.
- Durch „Login as user **pi**“ mussten Sie (noch) kein Passwort eingeben („raspberrry“ ist trotzdem gültig).

Es gibt noch zahlreiche weitere Einstellungen, aber zuerst geht es jetzt um die Sprachumgebung. Wenn Sie zur Registerkarte „Localisation“ wechseln und den Button „Set Locale“ betätigen, erscheinen „en“ und „GB“ für Language und Country. Ändern Sie die Einstellungen auf Ihr Land und die gewünschte Sprache ab, wie in Bild 2.14 zu sehen. Falls „de(German)“ nicht verfügbar sein sollte, wählen Sie bitte „de(Austrian German)“ aus. Bestätigen Sie die Änderung mit „OK“.

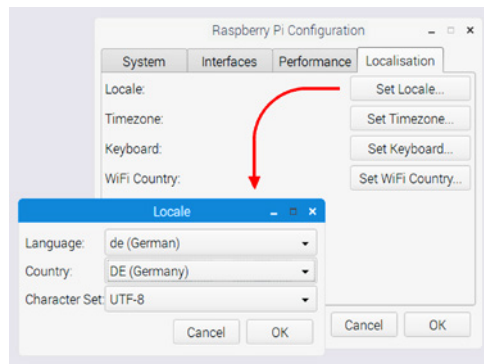


Bild 2.14 Auswahl von Deutsch als Sprache und UTF-8 als Zeichensatz

Kurzzeitig wird ein Bestätigungsfenster für „Set Locale“ erscheinen. Dann klicken Sie bitte auf „Set Timezone“ und wählen dort z. B. „Europe“ und „Berlin“ aus, wie in Bild 2.15 zu sehen. Bestätigen Sie die Änderung mit „OK“.

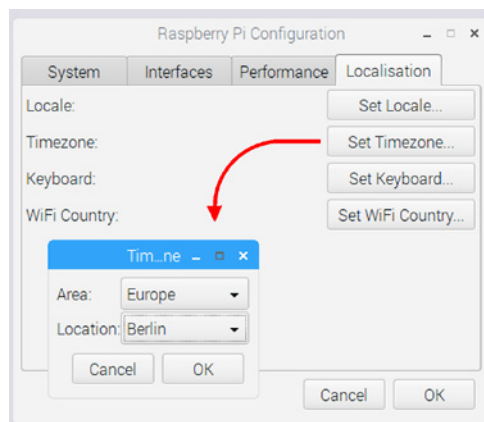


Bild 2.15 Auswahl von Berlin als Zeitzone

Da „Set Locale“ und „Set Timezone“ verändert wurden, betätigen Sie jetzt den „OK“-Button bei „Localization“, wie in Bild 2.16 zu sehen. Die Änderungen sollen wirksam werden, und Sie lösen mit „Yes“ einen Reboot aus.

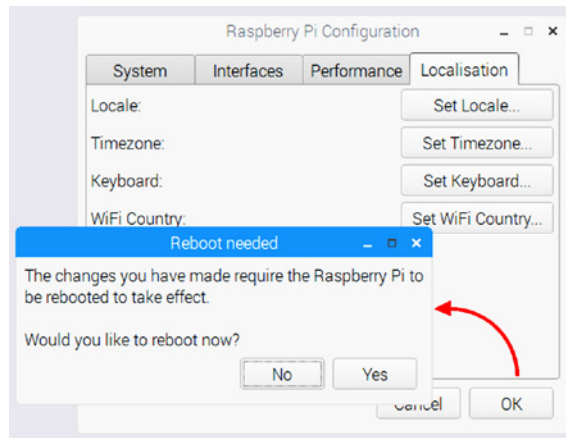


Bild 2.16 OK und Reboot

Nach etwa 30 Sekunden befindet sich ein Raspberry Pi 3 wieder in der grafischen Oberfläche. Sie können an Bild 2.17 erkennen, dass es jetzt ein Icon „Papierkorb“ gibt. Die Landessprache ist also bereits angepasst, und im Hauptmenü werden deutschsprachige Beschriftungen vorkommen. Doch noch kennt die Tastatur keine deutschen Umlaute.

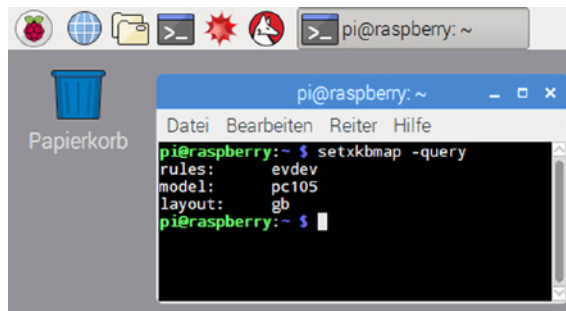


Bild 2.17 Die Sprache ist auch im LXTerminal auf „Deutsch“ umgeschaltet.

Da der Button „Set Keyboard“ in `rc_gui` nicht betätigt wurde, verwundert es nicht, dass das Tastaturlayout immer noch auf „gb“ steht. Die Sprachumschaltung war jedoch vollständig wirksam, was sich auch nach einem erneuten Aufruf von `rc_gui` ergibt.

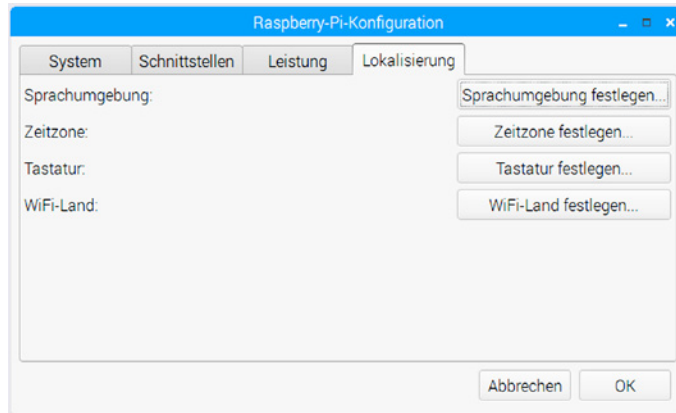


Bild 2.18 Das Programm `rc_gui` nach der Sprachumschaltung

Die Tastaturumschaltung soll nun aber nicht anhand von `rc_gui` erläutert werden. Das hat folgende Gründe:

1. Bei der LITE-Version müssen Sie mit `raspi-config` statt mit `rc_gui` arbeiten.
2. Die Tastaturumschaltung über `rc_gui` ist noch nicht ganz fehlerfrei (Version vom 16.08.2017), denn es kann sein, dass eine nach dem Abschalten/Reboot gegebenenfalls aktivierte Passworteingabe als „raspberrz“ erfolgen muss. Die Tastatur wäre dann beim Einloggen wieder auf „gb“.
3. Die Tastaturumschaltung kann stabil und dauerhaft über `raspi-config` oder `tastatur.sh` erfolgen.

Das Skript `tastatur.sh` ist in *ProgPi.zip* enthalten. Wie Sie *ProgPi.zip* entpacken, wird in Abschnitt 4.1 beschrieben. Doch zunächst zum Skript `raspi-config`, das im LXTerminal aufgerufen wird. Vergrößern Sie das LXTerminal nach dem Start auf etwa die halbe Bildschirmbreite. Dann geben Sie folgende Zeile ein:

```
sudo raspi-config
```

Da `raspi-config` die Systemkonfiguration ändert, ist „sudo“ als Präfix notwendig. Sie bekommen nach dem Start Bild 2.19 zu sehen.

Etwas komfortabler ist der Weg über das Desktop-Icon „TastaturDE“, das in Bild 2.21 eingebildet ist. Bei Raspbian LITE müssen Sie entweder `raspi-config` verwenden, oder das Skript `tastatur.sh` direkt aufrufen.

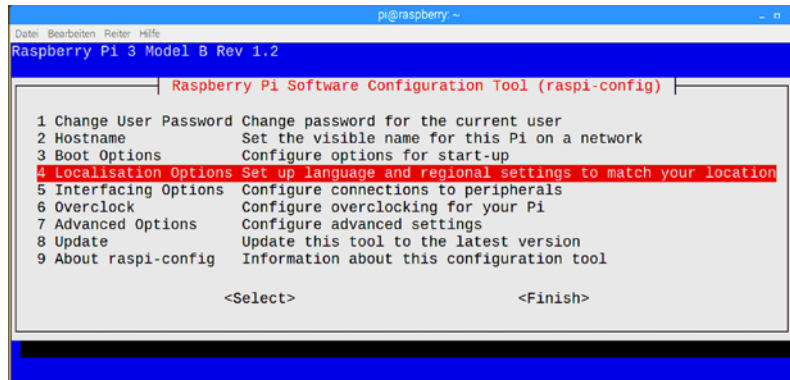


Bild 2.19 Das Hauptmenü von `raspi-config`

Obwohl die Menübeschriftung bereits deutschsprachig ist, erscheinen die Texte in Englisch. Nach der Auswahl von „4. Localisation Options“ bekommen Sie die ASCII-Oberfläche aus Bild 2.20 zu sehen.

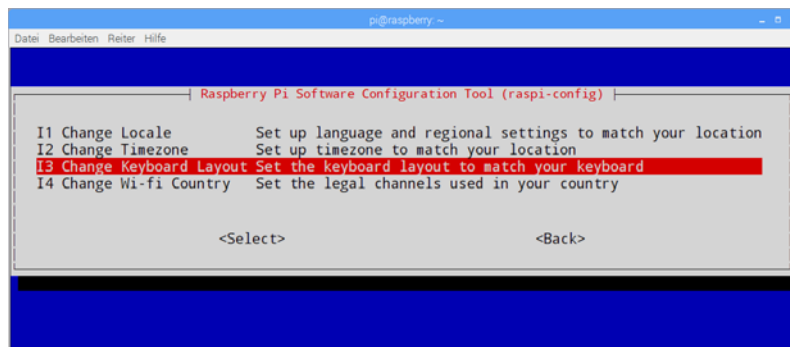


Bild 2.20 Das Tastaturlayout über `raspi-config` ändern

Darin wählen Sie „I3. Change Keyboard Layout“ und bestätigen die Auswahl mit <Enter>. Jetzt beginnt die eigentliche Einstellung des Tastaturlayouts, und es läuft genau der Teil von `raspi-config` ab, der auch in `tastatur.sh` enthalten ist. Der recht kurze Quelltext von `tastatur.sh` ist in Abschnitt 4.6.16 beschrieben. Die Anwendung sehen Sie in Bild 2.21.

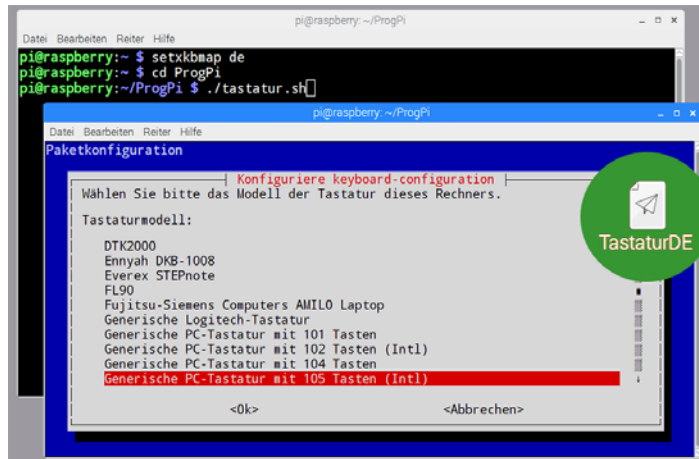


Bild 2.21 Das Skript `tastatur.sh` in der Konsole

Wenn Sie den Weg über `raspi-config` benutzt haben, führt das ebenso zu der ASCII-Oberfläche aus Bild 2.21, mit der zunächst die Hardware der Tastatur festgelegt wird. Sie können hier in aller Regel die „Generische PC-Tastatur mit 105 Tasten“ übernehmen.

In den nachfolgenden Schritten wählen Sie das Tastaturlayout Ihrer Landessprache aus, wie in Bild 2.22 zu sehen.

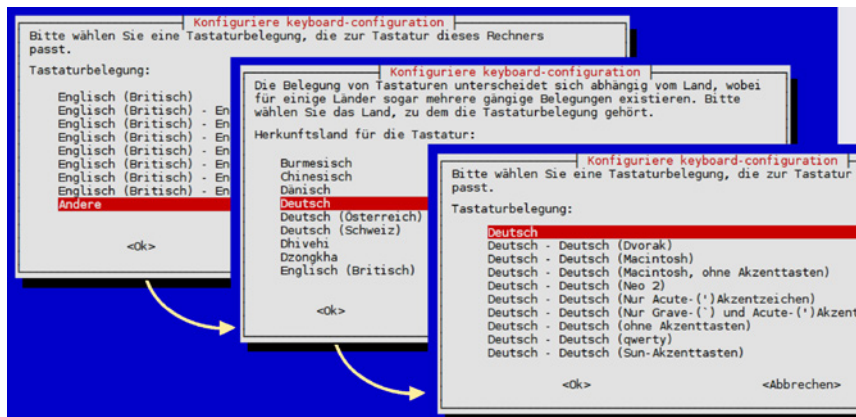


Bild 2.22 Tastaturnumschaltung über `tastatur.sh` bzw. `raspi-config`

Es folgen noch weitere Abfragen, bei denen Sie den jeweiligen Vorgaben folgen können:

- AltGr → Standard
- Compose → keine Compose-Taste
- Strg+Alt+Backspace für X-Server → Nein

Zuletzt beenden Sie `raspi-config` (bzw. `tastatur.sh`) mit „Finish“.

Bei `raspi-config` und `tastatur.sh` ist die Änderung des Tastaturlayouts dauerhaft. Beide Programme funktionieren auch über SSH – und damit ohne Monitor am Raspberry Pi.

2.2.2 Konfiguration von Autologin, Hostname und Bootmodus

Manchmal ist weder ein Autologin erwünscht, noch ein automatischer Start des Raspbian Desktop. Zusätzlich soll z.B. „raspi3“ als neuer Hostname verwendet werden, was zwingend erforderlich ist, falls es im Netzwerk noch einen weiteren Raspberry Pi geben sollte. Für diese Änderungen kann `raspi-config` oder innerhalb des Desktop auch `rc_gui` verwendet werden.

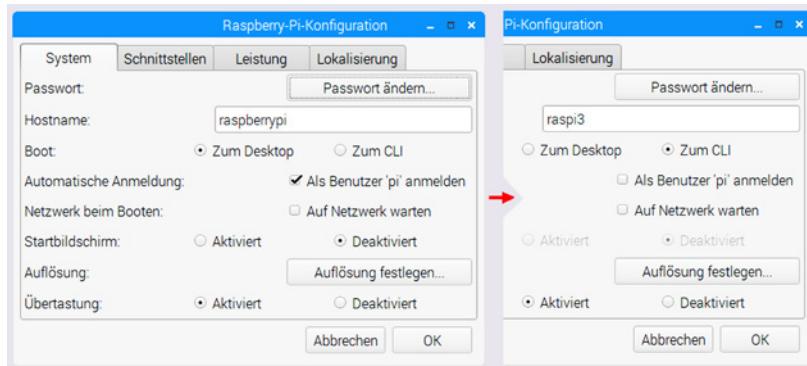


Bild 2.23 Änderungen der Konfiguration über `rc_gui`

Sie bestätigen die Änderungen aus Bild 2.23 mit „OK“ und fahren den Raspberry Pi herunter. Der nächste Bootvorgang wird dann direkt zur Systemkonsole führen. Die (micro)SD-Karte könnte nun entnommen und in einem Raspberry Pi eingesetzt werden, der keine Benutzeroberfläche braucht. Für einen Raspberry Pi 1 wäre nun „raspi1“ ein sinnvoller Hostname – auch wenn Sie bisher die gesamte Konfiguration mit einem Raspberry Pi 3 durchgeführt haben.

In Bild 2.24 sehen Sie die Systemkonsole des Raspberry Pi nach dem Booten von Raspbian Jessie, aber noch vor dem Login. Der Radiobutton „Zum CLI“ aus Bild 2.23 ist übrigens eine Abkürzung für Command Line Interface, was der Systemkonsole entspricht. Der Benutzername ist „pi“ und das Passwort lautet „raspberrypi“. Booten Sie jetzt Ihren Raspberry Pi (gegebenenfalls ist es jetzt ein anderes Modell).

```
[ OK ] Started Serial Getty on ttyAMA0.
[ OK ] Reached target Login Prompts.

Raspbian GNU/Linux 8 raspiz tty1

raspiz login: pi
Password:
Last login: Fri Sep 25 04:03:46 CEST 2015 from 192.168.1.2 on pts/0
Linux raspiz 4.1.7-07+ #817 SMP PREEMPT Sat Sep 19 15:32:00 BST 2015 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
pi@raspiz ~ $
```

Bild 2.24 Die Systemkonsole von „Jessie“ nach dem Einloggen

Der Raspberry Pi bootet jetzt in die Systemkonsole. Der unsichtbare Benutzer „root“ ist jetzt bereits aktiv und hat eine sogenannte „Login-Shell“ gestartet. Raspbian wartet nun auf eine Benutzeranmeldung. Geben Sie bei „login:“ statt „pi“ zunächst „piz“ ein. Wenn dabei „y“ statt „z“ erscheint, haben Sie noch eine englische Tastatur und verwenden bitte „rasberrz“ als Passwort (soweit Sie das Passwort nicht verändert haben).

Bild 2.24 ist übrigens mit einer Digitalkamera entstanden, da ein Screenshot über die Taste <Druck> und scrot nur innerhalb des Desktops funktioniert. An „GNU/Linux 8“ können Sie erkennen, dass es sich um Raspbian Jessie handelt. Bei Raspbian Stretch würden Sie „GNU/Linux 9“ sehen.

Die Ausgabe „armv7l“ deutet auf einen Raspberry Pi 2 hin. Das gilt für Jessie selbst dann, wenn Sie einen Raspberry Pi 3 verwenden, der 64 Bit und eine „armv8l“-Prozessorarchitektur hat. Durch das Kommando

```
uname --all
```

können Sie die Raspbian-Version und die Hardware jederzeit abfragen. Sie sehen in Bild 2.24 übrigens auch, dass der letzte Login „remote“ stattfand (from 192.168.1.2). Um wieder in der Desktop-Oberfläche zu landen, geben Sie folgendes Kommando auf der Systemkonsole ein:

```
startx
```

2.2.3 Passwortänderung

Sobald Sie wieder im RPi-Desktop sind, ist es sehr empfehlenswert, das Passwort abzuändern. Das funktioniert vielleicht am bequemsten über `rc_gui` (Bild 2.25).

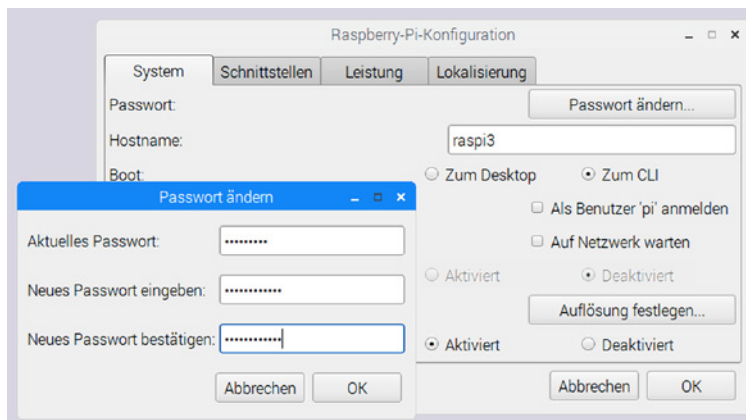


Bild 2.25 Passwortänderung über `rc_gui`

Doch auch das Konsolenkommando `passwd` funktioniert:

```
pi@raspi3:~ $ passwd
Ändern des Passworts für pi.
(aktuelles) UNIX-Passwort: *****
Geben Sie ein neues UNIX-Passwort ein: *****
Geben Sie das neue UNIX-Passwort erneut ein: *****
passwd: Passwort erfolgreich geändert
pi@raspi3:~ $
```

2.2.4 Freigabe für SSH und Abfrage der IP-Adresse

Nachdem das Passwort abgeändert ist, kann auch der SSH-Zugriff aktiviert werden. Dazu wechseln Sie in `rc_gui` zu „Interfaces“ (bzw. Schnittstellen) und nehmen die Änderung wie in Bild 2.26 dargestellt vor.

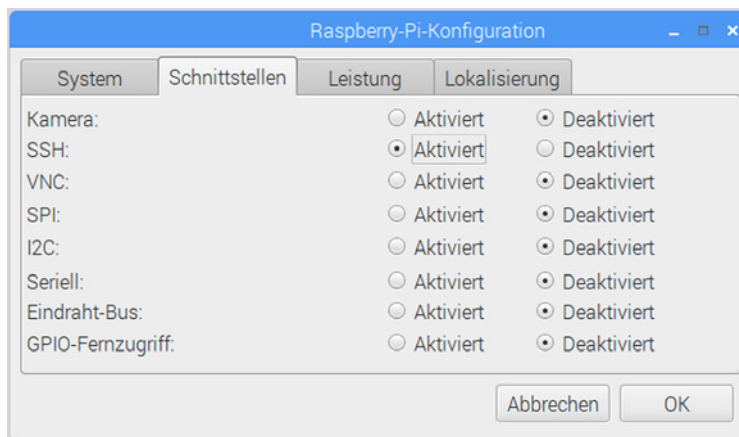


Bild 2.26 Aktivierung des SSH-Zugriffs über `rc_gui`

Falls noch nicht geschehen, verbinden Sie nun den Raspberry Pi über ein Netzkabel mit dem DSL-Modem. Soweit Ihr DSL-Modem-Router (der Begriff wird genauer in Kapitel 7 erklärt) auf DHCP eingestellt ist, wird der Router eine IP-Adresse vergeben, die Sie mit dem Raspberry Pi nach einem Reboot abfragen können:

```
pi@raspi3:~ $ ifconfig
eth0      Link encap:Ethernet  Hardware Adresse b8:27:XX:XX:XX:XX
          inet Adresse:192.168.1.4  Bcast:192.168.1.255
          Maske:255.255.255.0
          ....
lo        Link encap:Lokale Schleife
          inet Adresse:127.0.0.1  Maske:255.0.0.0
          inet6-Adresse: ::1/128  Gültigkeitsbereich:Maschine
          UP LOOPBACK RUNNING  MTU:65536  Metrik:1
          ....
wlan0    Link encap:Ethernet  Hardware Adresse b8:27:XX:XX:XX:XX
          ....
```

Der Raspberry Pi wird damit über „192.168.1.4“ im Netzwerk erreichbar sein. Ihr Router wird voraussichtlich eine andere IP-Adresse vergeben. Die Grundkonfiguration von Raspbian bzw. des Raspberry Pi ist damit vorläufig abgeschlossen.

■ 2.3 Der Raspberry Pi Desktop als Bedienoberfläche

Bei Linux gibt es nicht nur Bezeichnungen für das Betriebssystem. Innerhalb eines Betriebssystems gibt es unterschiedliche Bedienoberflächen. LXDE war dabei der Startpunkt. Danach haben ein paar Umbenennungen sowie Verbesserungen stattgefunden. Die Bezeichnung PIXEL (Pi Improved Xwindows Environment, Lightweight) war ab September 2016 für etwa ein Jahr gültig. Nun ist Raspberry Pi Desktop die offizielle Bezeichnung. In Abschnitt 2.2 haben Sie den RPi-Desktop schon ein wenig kennengelernt. Es wird nun angenommen, dass Sie Deutsch als Sprache eingerichtet und das Programm Raspberry-Pi-Konfiguration (bzw. `rc_gui`) schon einmal benutzt haben. Auch das Passwort haben Sie abgeändert. Möglicherweise haben Sie auch den Desktop-Hintergrund neu eingestellt und das Hauptmenü ein wenig erkundet. In den folgenden Abschnitten wird hauptsächlich beschrieben, wie Sie Programme starten und wie die Programme anklickbare Elemente der Bedienoberfläche werden können.

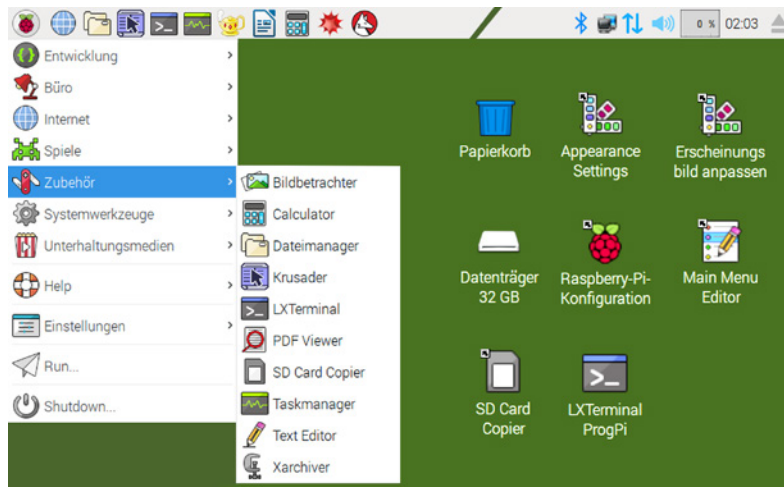


Bild 2.27 Der RPi-Desktop mit Zubehör-Programmen im Startmenü

2.3.1 Die Lage der Pannelleiste ändern

Wer längere Zeit mit Windows XP gearbeitet hat, wird sich sehr schnell im RPi-Desktop zurechtfinden und wahrscheinlich die Bedienleiste am unteren Bildschirmrand anordnen. Zum Ändern der Paneeleinstellungen klicken Sie mit der rechten Maustaste mittig auf das Panel (Bild 2.29). Es erscheint ein Auswahlm Menü, in dem Sie „Panel-Einstellungen“ anklicken, was zur Anzeige eines gleichnamigen Fensters führt, wie in Bild 2.28 zu sehen.

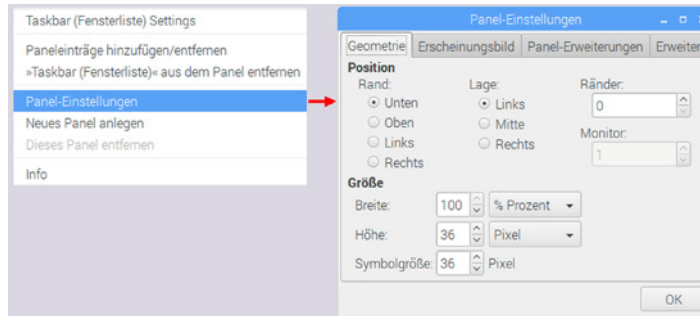


Bild 2.28 Einstellungen für die Pannelleiste

Sobald Sie den Radiobutton „Unten“ anklicken, springt die Pannelleiste zum unteren Bildschirmrand. Im Dialog aus Bild 2.28 lassen sich auch die Größen der Icons und die Höhe der Pannelleiste verstellen.

2.3.2 Der Aufbau der Pannelleiste

Die Pannelleiste kann in vier Gruppen oder Funktionseinheiten eingeteilt werden, die Sie in Bild 2.29 sehen.



Bild 2.29 Elemente in der Pannelleiste

Dabei sind in der aktuellen Raspbian-Stretch-Version folgende Programme in der Anwendungsstartleiste voreingestellt:

- Chromium Webbrowser
- Dateimanager (PCManFM)
- LXTerminal
- Mathematica
- Wolfram

Bei Programmen, im Hauptmenü oder aus der Anwendungsstartleiste, genügt ein einfacher Klick zum Programmstart. Programme, für die Sie eine Desktop-Verknüpfung oder ein Desktop-Icon angelegt haben, werden durch Doppelklick gestartet.

Wenn Programme über den RPi-Desktop gestartet werden sollen, gibt es folgende Möglichkeiten:

- Das Programm wird im Hauptmenü angesteuert und angeklickt bzw. mit <Enter> gestartet.
- Im Hauptmenü wird der „Run“-Dialog angesteuert und der Programmname darin eingetippt.
- Das Icon zum Programm ist Teil der Anwendungsstartleiste und wird dort angeklickt.
- Für das Programm wird eine Desktop-Verknüpfung erzeugt, deren Doppelklick den Programmstart auslöst.
- Für das Programm wird ein Desktop-Icon erzeugt, dessen Doppelklick den Programmstart auslöst.

In den folgenden Abschnitten werden Sie die genannten Startmethoden und deren Konfiguration näher kennenlernen.

2.3.3 Programmstart über Tastaturkürzel

Eine recht einfache Möglichkeit, Programme zu starten, ist der „Run“-Dialog, den Sie direkt im Startmenü finden oder durch <Alt>+<F2> auslösen können.

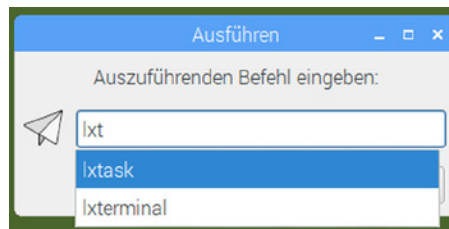


Bild 2.30 Programmstart über <ALT>+<F2>

Wie Sie in Bild 2.30 sehen können, genügt die Eingabe weniger Zeichen, um eine Auswahl der verfügbaren Programme zu erhalten. So würde z. B. die Zeichenfolge „lxa“ genügen, damit Sie „lxappearance“ (Abschnitt 2.5) angeboten bekommen. Im „Run“-Dialog können auch problemlos Parameter angehängt werden, wie in folgendem Fall:

```
lxterminal --working-directory=~/ProgPi
```

Das LXTerminal wird so in `/home/pi/ProgPi` geöffnet und kann dann Kommandos entgegennehmen. Sie könnten aber auch direkt ein Programm im LXTerminal aufrufen:

```
lxterminal -e geany
```

Der „Run“-Dialog ist bei vielen Linux-Distributionen mit <Alt>+<F2> aufrufbar.

Das Hauptmenü (oder Startmenü) ist ein sehr oft benutzter Weg für den Programmstart. Durch <Strg>+<Esc> erscheint das Hauptmenü, und Sie können das gewünschte Programm allein mit den Cursorstasten anwählen und mit <Enter> starten. Die Maus muss dafür nicht angeschlossen sein. Ein LXTerminal kann übrigens auch durch <Alt>+<Strg>+<T> gestartet werden.

2.3.4 Eine symbolische Desktop-Verknüpfung hinzufügen

Eine Desktop-Verknüpfung zu einem Programm im Startmenü ist sehr schnell erstellt. Sie müssen lediglich das Programm ansteuern und dann die rechte Maustaste betätigen. In Bild 2.31 geschieht das für den SD Card Copier. Nachdem Sie „Dem Desktop hinzufügen“ ausgewählt haben, entsteht eine symbolische Verknüpfung auf dem Desktop.

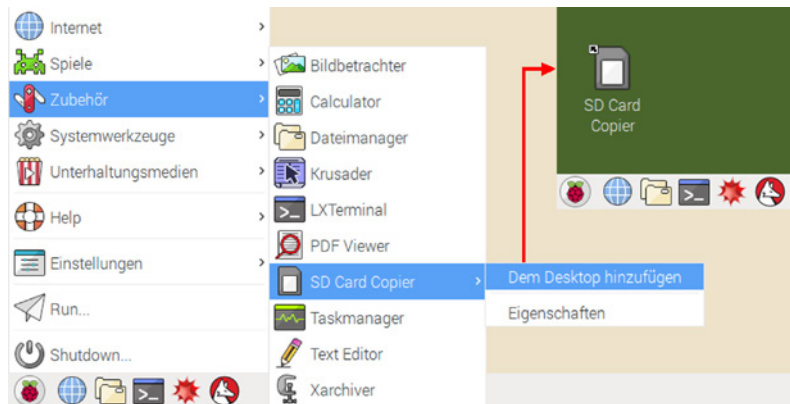


Bild 2.31 Eine Desktop-Verknüpfung zu SD Card Copier erzeugen

Eine Desktop-Verknüpfung hat einen kleinen Pfeil im Icon und entspricht nicht dem „echten“ Desktop-Icon. Der SD Card Copier ist nichts anderes als ein Link zur Datei *piclone.desktop* und über Doppelklick startbar. Die Funktion des Programms SD Card Copier wird kurz in Abschnitt 2.13 und genauer in Abschnitt 3.7 beschrieben.

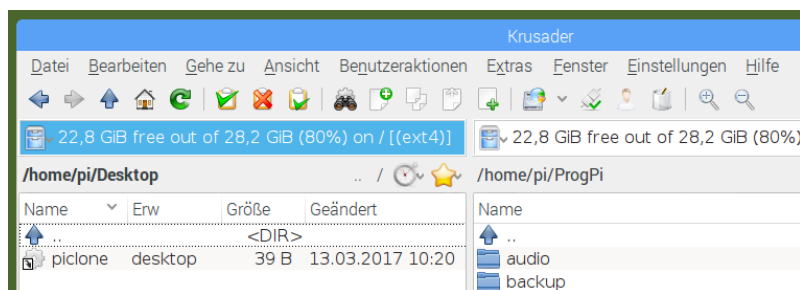


Bild 2.32 Desktop-Verknüpfung zu SD Card Copier im Krusader

Im Dateimanager Krusader (siehe Abschnitt 2.7) erscheint die Datei *piclone.desktop* mit lediglich 39 Byte (Bild 2.32). Diese geringe Größe ist ebenso wie der Pfeil im Icon ein Hinweis auf einen symbolischen Link und keine „echte“ Desktop-Datei. Beachten Sie bitte, dass sich der Benutzer **pi** zwar *piclone.desktop* in */home/pi/Desktop* ansehen kann, dass aber Änderungen darin (also rechte Maustaste, Dateieigenschaften im Kontextmenü) nicht übernommen werden.



Aufgabe:

Erzeugen Sie eine Desktop-Verknüpfung zum „Main Menu Editor“ Alacarte aus dem Untermenü „Einstellungen“.

2.3.5 Eine neue Desktop-Datei erzeugen

Welche Programme sind in Raspbian installiert? Einen guten Überblick erhalten Sie, wenn Sie den Dateimanager PCManFM starten und darin zum Verzeichnis *usr/share/applications* wechseln, wie in Bild 2.33 zu sehen.

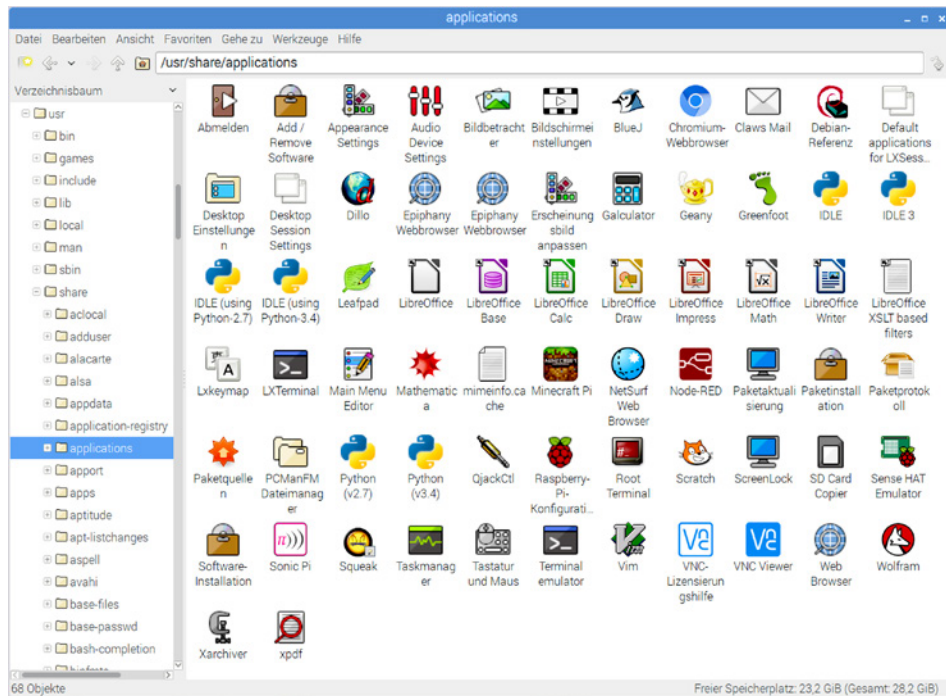


Bild 2.33 Startbare Programme bei Raspbian Jessie

Genau genommen handelt es sich dabei aber nicht um startbare Programme, sondern lediglich um Desktop-Dateien, deren Icons der Dateimanager anzeigt. Das wird gut erkennbar, wenn Sie

in PCManFM zum Menüpunkt „Ansicht“ gehen und dann die Ebene „Ordneransicht/Detailansicht“ auswählen (Bild 2.34).

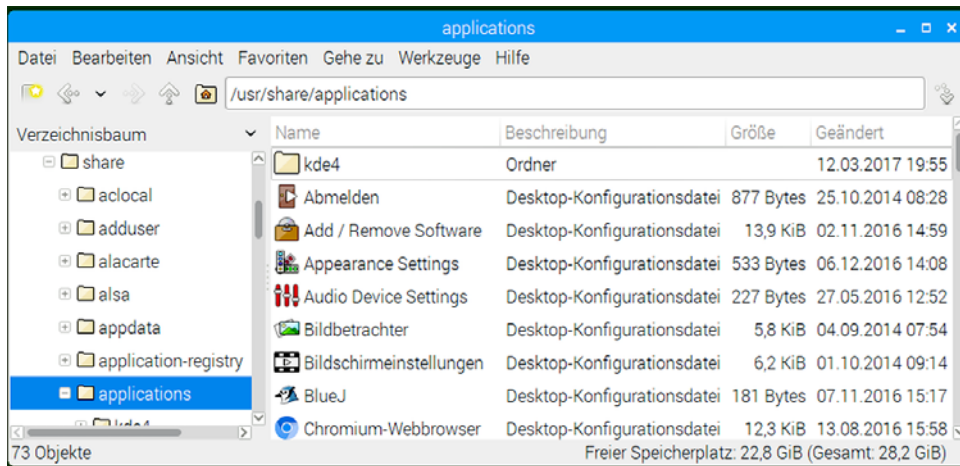


Bild 2.34 Die Icons der startbaren Programme

In der Spalte Beschreibung steht nun „Desktop-Konfigurationsdatei“, was *.desktop als Dateinamen entspricht. Es soll nun eine echte Kopie einer solchen Desktop-Datei für den Benutzer „pi“ entstehen.

Das LXTerminal ist ein Programm, das Sie für Konsoleneingaben benötigen und öfter starten werden. Es kann dabei lästig sein, immer wieder manuell in ein bestimmtes Verzeichnis zu wechseln. Die Programme zu diesem Buch werden in /home/pi/ProgPi abgelegt bzw. in dieses Verzeichnis entpackt (siehe Abschnitt 4.1). Was liegt also näher, als sich ein Desktop-Icon zum LXTerminal zu schaffen, das direkt dorthin führt?

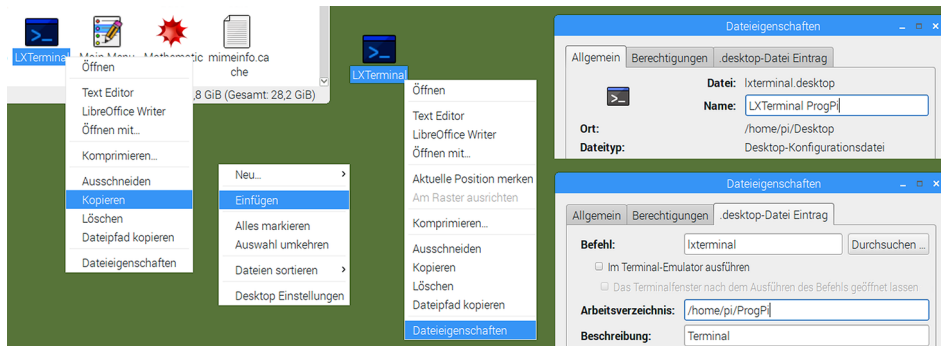
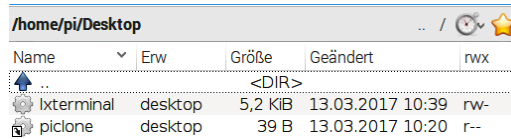


Bild 2.35 Ein Desktop-Icon für LXTerminal erzeugen

Im Dateimanager PCManFM klicken Sie „LXTerminal“ mit der rechten Maustaste an. Im Kontextmenü wählen Sie „Kopieren“, was die Desktop-Datei in der Zwischenablage ablegt. Dann klicken Sie mit der rechten Maustaste auf den Desktop-Hintergrund und wählen „Einfügen“. Jetzt befindet sich LXTerminal auf Ihrem Desktop. Zur Anpassung gehen Sie per Rechtsklick zu

den „Dateieigenschaften“. Ergänzen Sie im Register „Allgemein“ bei Name „ProgPi“. Dann tragen Sie noch bei „desktop-Datei Eintrag“ das Arbeitsverzeichnis ein, wie in Bild 2.35 zu sehen. Bei Doppelklick auf das nun vorhandene Desktop-Icon „LXTerminal ProgPi“ starten Sie im eingestellten Arbeitsverzeichnis (also z. B. /home/pi/ProgPi).



Name	Erw	Größe	Geändert	rwx
..		<DIR>		
lterminal	desktop	5,2 KiB	13.03.2017 10:39	rw-
piclone	desktop	39 B	13.03.2017 10:20	r--

Bild 2.36 Dateien in /home/pi/Desktop (Krusader)

In Bild 2.36 sehen Sie einen Vergleich der Dateigrößen bei der echten Desktop-Datei und einem symbolischen Link. Der Name jeder *.desktop-Datei kann problemlos in einem beliebigen Dateimanager geändert werden. Die Desktop-Icons können jetzt so wie in Bild 2.37 aussehen.

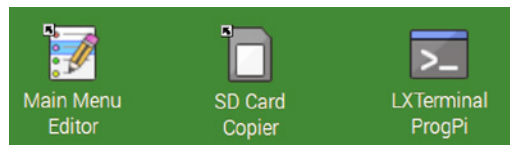


Bild 2.37 Desktop-Icons für Alacarte, PiClone und LXTerminal

2.3.6 Ein Programm der Anwendungsstartleiste hinzufügen

Um ein Programm der Anwendungsstartleiste hinzuzufügen, klicken Sie mit der rechten Maustaste in den mittigen, nicht gefüllten Panelbereich (Bild 2.29) und wählen dann „Paneleinträge hinzufügen/entfernen“ aus. Es erscheint das Fenster „Panel-Einstellungen“, in dem Sie den in Bild 2.38 markierten Eintrag „Anwendungsstartleiste“ doppelklicken.

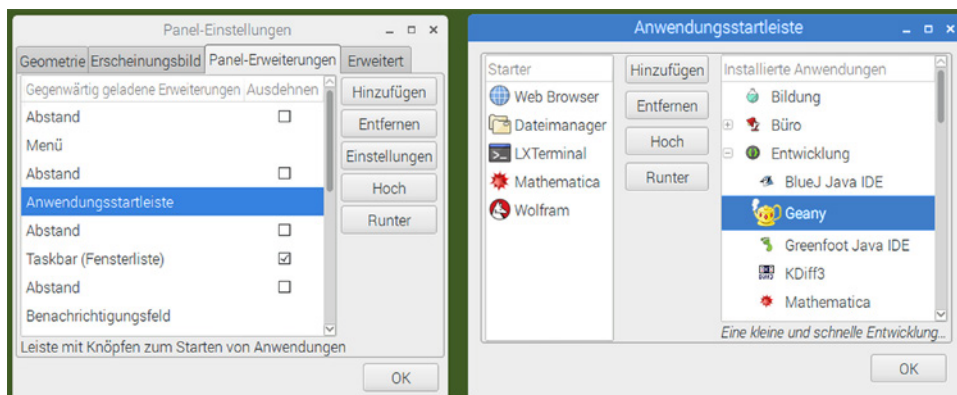


Bild 2.38 Ein Programm der Anwendungsstartleiste hinzufügen

Nach dem Doppelklick erscheint das Fenster „Anwendungsstartleiste“ mit vier mittig angeordneten Buttons. In der rechten Hälfte dieses Fensters navigieren Sie zu dem Programm, das Sie hinzufügen wollen. Wenn Sie das Zeichen „+“ neben „Entwicklung“ anklicken, werden die installierten Programme sichtbar. In Bild 2.38 wird der Quelltexteditor Geany ausgewählt. Sobald Sie den Button „Hinzufügen“ betätigen, erscheint Geany unter Wolfram in der linken Fensterhälfte. Dann klicken Sie das frisch kopierte Feld an, und drücken zweimal „Hoch“, um Geany neben dem LXTerminal zu positionieren. Mit „OK“ wird der Vorgang bestätigt, und Geany erscheint (als dampfende Teekanne) rechts neben LXTerminal in der Anwendungsstartleiste.



Bild 2.39 Anwendungsstartleiste mit Geany



Wichtiger Hinweis:

Bei neu installierten Programmen kann es sein, dass das Programm nicht im Fenster „Anwendungsstartleiste“ auftaucht, oder zwar vorhanden ist, sich aber nicht nach links zu den anderen Programmen kopieren lässt. Überprüfen Sie in solch einem Fall, ob das Programm im Main Menu Editor Alacarte (siehe Bild 2.43) aktiviert ist. Zusätzlich können Sie die Position im Hauptmenü mit Alacarte verändern. So wird das Programm auch im Fenster „Anwendungsstartleiste“ erscheinen.

2.3.7 Miniprogramme im Panel

Ein Miniprogramm hat keine große Bedienoberfläche, sondern einen beschränkten Satz von Anzeigeelementen, die z. B. als Kontextmenü oder eine Art Bubblehelp mehrzeilige Informationen anbieten. Die Zeilenzahl ist meist gering, und der Hauptzweck ist eine Kurzinformation.

Der RPi-Desktop von Raspbian nutzt derartige Miniprogramme unter anderem für folgende Zwecke:

- Zustand/Konfiguration von Bluetooth
- WiFi-Netzwerke anzeigen/verwalten
- Lautstärkereglern
- Prozessorlast-Monitor
- digitale Uhr
- Ejecter für USB-Geräte

Die Miniprogramme werden in Bild 2.29 der Gruppe „Panel-Icons“ zugeordnet und über *Panel-Einstellungen* > *Panel-Erweiterungen* verwaltet.

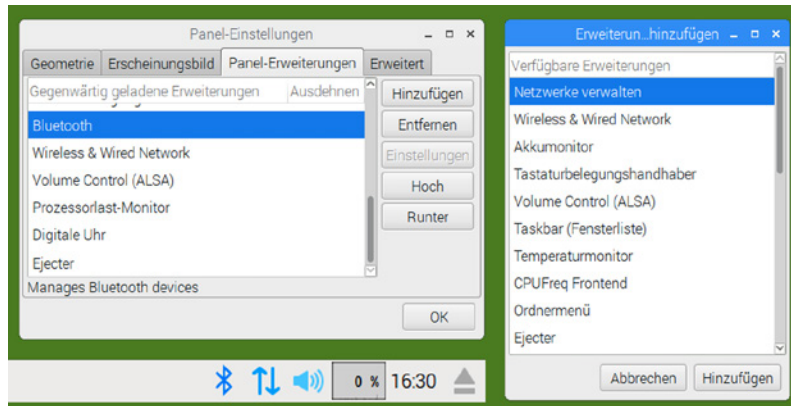


Bild 2.40 Miniprogramme im Panel (Originalzustand)

In Bild 2.40 sehen Sie am unteren Rand die Miniprogramme, die bei Raspbian Stretch derzeit voreingestellt sind. Sobald Sie den Button „Hinzufügen“ im Fenster „Panel-Einstellungen“ betätigen, wird ein Auswahlfenster „Erweiterung hinzufügen“ eröffnet. Der oberste Listeneintrag darin ist „Netzwerke verwalten“, und mit dem zweiten Button „Hinzufügen“ wird der Eintrag an das Listeneende der Panel-Erweiterungen gesetzt. Mit „Hoch“ bewegen Sie „Netzwerke verwalten“ so lange, bis der Eintrag unter „Bluetooth“ steht, was Sie in Bild 2.41 sehen.

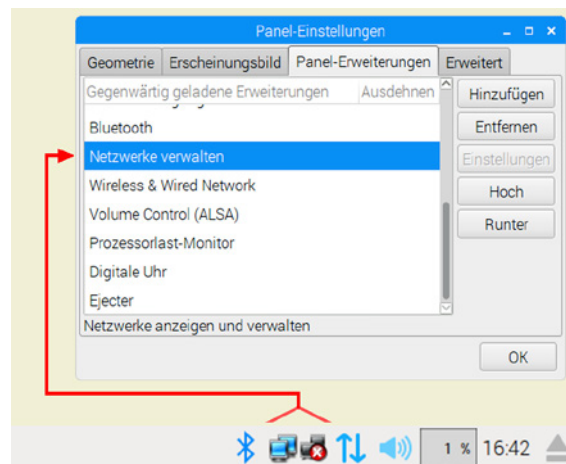


Bild 2.41 Miniprogramme nach Ergänzung von „Netzwerke verwalten“

Sie sehen in Bild 2.41, dass „Netzwerke verwalten“ aus zwei Icons für LAN/WLAN besteht, wobei die WLAN-Verbindung nicht aktiv ist. Details zu Netzwerken finden Sie in Kapitel 7.



Aufgabe:

Erweitern Sie die Panelleiste um den Temperaturmonitor.

2.3.8 Das Konsolenkommando lxpanelctl

Was macht Raspbian eigentlich, wenn Sie im Panel auf die „Himbeere“ bzw. den Button zum Startmenü klicken? Intern werden über das Kommando `panelctl` das (Start)Menü, der „Run“-Dialog und alle Einstellungen für die Bedienleiste verwaltet. In Bild 2.42 sehen Sie links einige Konsoleneingaben. Das Kommando `panelctl menu` führt zur Anzeige des Startmenüs, das in Bild 2.42 nicht enthalten ist.

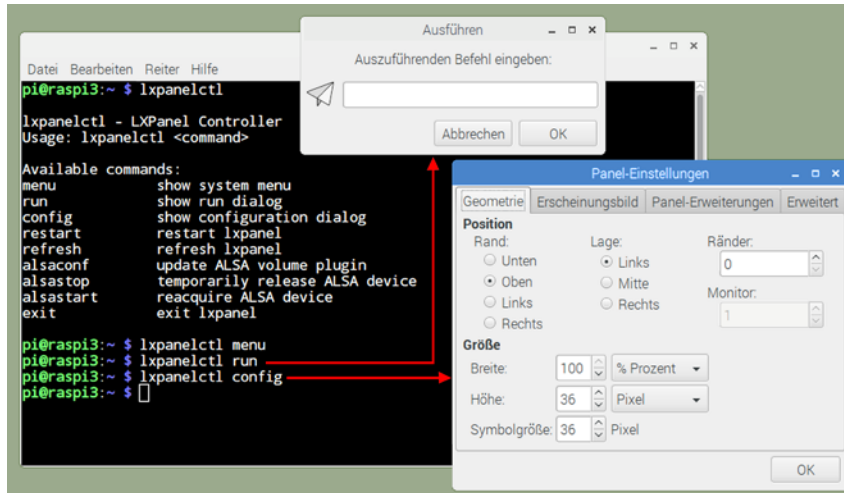


Bild 2.42 Konfiguration und Kommandos für die Panel- und Menüeinstellungen

Das Startmenü ist auch über `<Strg>+<Esc>` erreichbar.

Die Tastenkombination `<Alt>+<F2>` bewirkt, dass der „Run“-Dialog (siehe Bild 2.30) erscheint. Ebenso geschieht das, indem Sie `lxpanelctl run` im LXTerminal eintippen.

Der übliche Weg zu den Paneeleinstellungen ist das Anklicken der Bedienleiste mit der rechten Maustaste. Das Startmenü, in dem auch der Unterpunkt „Run“ enthalten ist, erreichen Sie über `lxpanelctl menu`.

Sie sehen an diesem Beispiel, wie stark Linux bzw. Raspbian von Terminalkommandos „leben“.

■ 2.4 Der Main Menu Editor Alacarte

Nicht alle Programme, deren Icons Sie in Bild 2.33 sehen, können und werden im Hauptmenü erscheinen. Es muss vielmehr immer eine Auswahl getroffen werden, wobei Sie mit Alacarte Programme aktivieren/deaktivieren können. Sie finden den „Main Menu Editor“ im Hauptmenü unter „Einstellungen“.

Programme, die bei Linux auch als Pakete bezeichnet werden, lassen sich leicht nachinstallieren. Aber nicht immer wird ein neu installiertes Programm sofort dem Untermenü zugeordnet sein, das Sie vielleicht erwarten.

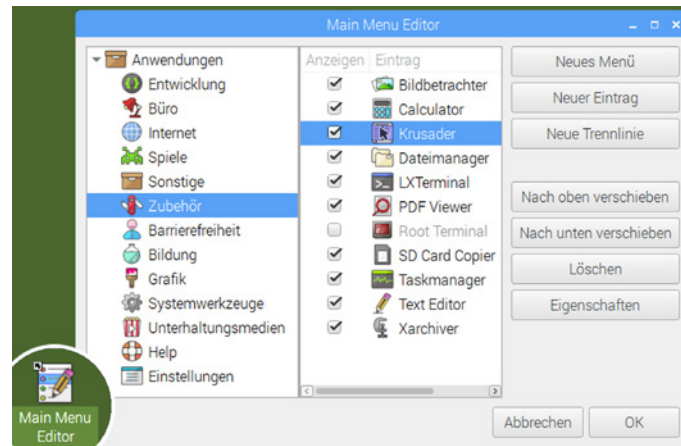


Bild 2.43 Programme im Zubehöroordinator von Alacarte

Dies sind zwei Anwendungsbeispiele für Alacarte:

- Um das „Root Terminal“ im Hauptmenü unter Zubehör erscheinen zu lassen, müssen Sie lediglich den noch fehlenden Haken setzen. Programme, die Sie voraussichtlich nicht im Hauptmenü haben wollen, können Sie in Alacarte natürlich auch deaktivieren. Das Programm bleibt weiterhin installiert und kann immer noch als Desktop-Icon vorkommen.
- Über den Button „Neues Menü“ aus Bild 2.43 könnten Sie z. B. OpenCV als neues Untermenü anlegen und dann darin mit „Neuer Eintrag“ Programme unterbringen.

■ 2.5 Das Erscheinungsbild anpassen mit LxAppearance

Zu den ein wenig versteckten Programmen von Raspbian Jessie gehört LxAppearance, das Sie in Bild 2.33 mit der Icon-Beschriftung „Erscheinungsbild anpassen“ sehen. Dort finden Sie auch ein identisches Icon mit „Appearance Settings“. Wo liegt der Unterschied?

- Den Dialog „Appearance Settings“ (*pipanel.desktop*) sehen Sie in Bild 2.9. Es kann damit z. B. ein Hintergrundbild aktiviert oder die Schriftgröße eingestellt werden.
- Durch „Erscheinungsbild anpassen“ (*lxappearance.desktop*) können Sie das sogenannte „Thema“ des Fensterdesigns anpassen. Dazu gehören das Design der Dialogelemente sowie Farbe und Aussehen der Fensterrahmen.

Wenn Sie „LXAppearance“ im Hauptmenü haben wollen, dann muss „Erscheinungsbild anpassen“ aktiviert sein, wie in Bild 2.44 zu sehen.

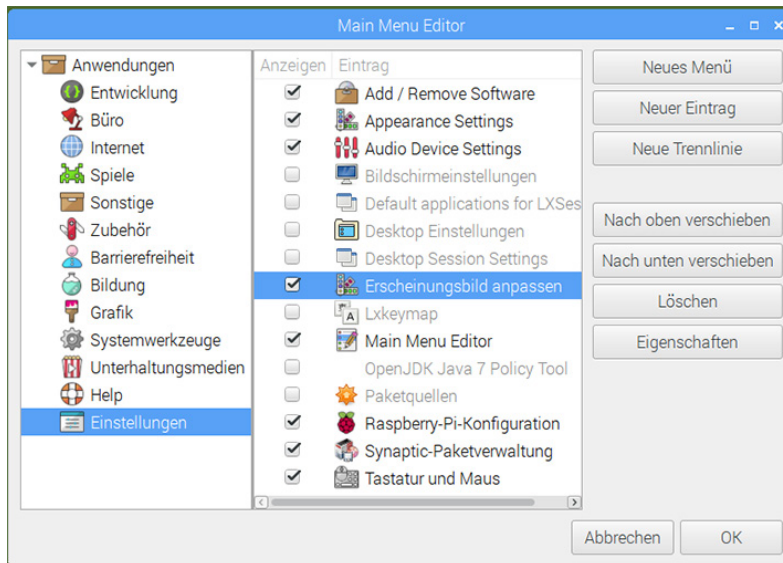


Bild 2.44 Im Hauptmenü LXAppearance aktivieren (Alacarte)

Das voreingestellte Fensterthema von „LXAppearance“ lautet PiX. Sie sehen in Bild 2.45 lediglich die Einstellungen für „Fenster“ und für „Fensterrahmen“. Je nach „Thema“ erscheint im „Fensterrahmen“ die Beschriftung linksbündig oder mittig und in unterschiedlichen Farben.

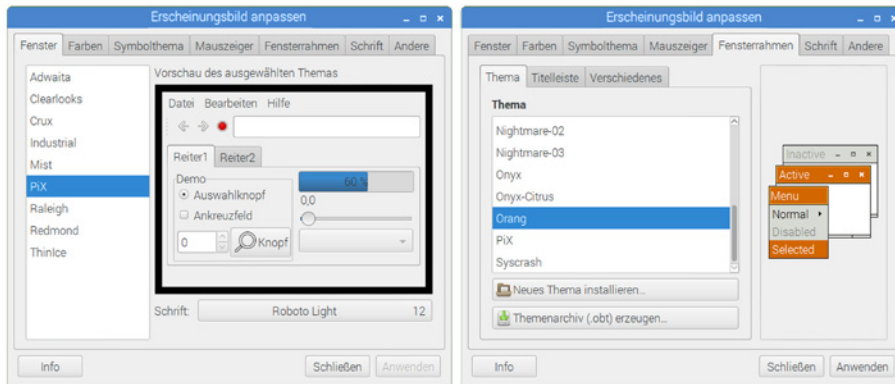


Bild 2.45 Das Fensterdesign mit LXAppearance anpassen

Bevor Sie Screenshots für eine Dokumentation (oder ein Buch) anfertigen, sollten Sie auf einen hohen Kontrast achten. Die Kopfzeile eines Fensters muss gegebenenfalls auch im Graustufen-Druck statt in Farbe gut lesbar bleiben. Für solch einen Fall wäre „Industrial“ als Fensterthema gut geeignet, weil darin mehr abgrenzende Linien vorkommen und das Design kaum Farben hat.