

RPi: Die (SoC) Spannungswerte vom Raspberry Pi auslesen

geschrieben von Andreas Potthoff | 25. Juni 2024

SoC Sources

Raspberry Pi OS

Mögliche Sources: core, sdram_c, sdram_i, sdram_p

Dabei bedeutet:

- core = Kern
- sdram_c = SDRAM controller
- sdram_i = SDRAM input/output
- sdram_p = SDRAM physical

Einzelnen Spannungs-Status ausgeben

```
pi@pi5b:~ $ vcgencmd measure_volts core  
volt=0.9496V
```

Alle Spannungswerte auslesen

```
for id in core sdram_c sdram_i sdram_p ; do \  
    echo "$id:\t$(vcgencmd measure_volts $id)" ; \  
done
```

```
pi@pi5b:~ $ for id in core sdram_c sdram_i sdram_p ; do echo  
"$id:\t$(vcgencmd measure_volts $id)" ; done  
core:\tvolt=0.7200V  
sdram_c:\tvolt=0.6000V  
sdram_i:\tvolt=0.6000V  
sdram_p:\tvolt=1.1000V
```

RPi: Die Board Revision vom Raspberry Pi auslesen

geschrieben von Andreas Potthoff | 25. Juni 2024

Anhand des Revisionscodes des Raspberry Pis erhalten Sie Informationen über das Releasedatum, das Modell, die Platinenversion, den RAM Speicher und den Hersteller. Mit folgenden Befehlen könnt ihr den Revisions-Code eures Raspberry Pis auslesen. Das funktioniert mit Raspberry Pi OS.

Mit `cpuinfo` den Revisionscode ausgeben

In den CPU-Infos findet ihr den Revisions-Code und auch die Seriennummer (ganz unten)

```
pi@pi5b:~ $ cat /proc/cpuinfo
processor          : 0
BogoMIPS         : 108.00
Features         : fp asimd evtstrm aes pmull sha1 sha2 crc32
atomics fphp asimdhp cpuid asimdrdm lrcpc dcpop asimddp
CPU implementer  : 0x41
CPU architecture: 8
CPU variant      : 0x4
CPU part         : 0xd0b
CPU revision     : 1

processor        : 1
BogoMIPS        : 108.00
Features        : fp asimd evtstrm aes pmull sha1 sha2 crc32
atomics fphp asimdhp cpuid asimdrdm lrcpc dcpop asimddp
CPU implementer : 0x41
```

```
CPU architecture: 8
CPU variant      : 0x4
CPU part        : 0xd0b
CPU revision    : 1

processor       : 2
BogoMIPS       : 108.00
Features        : fp asimd evtstrm aes pmull sha1 sha2 crc32
atomics fphp asimdhp cpuid asimdrdm lrcpc dcpop asimddp
CPU implementer : 0x41
CPU architecture: 8
CPU variant     : 0x4
CPU part       : 0xd0b
CPU revision   : 1

processor       : 3
BogoMIPS       : 108.00
Features        : fp asimd evtstrm aes pmull sha1 sha2 crc32
atomics fphp asimdhp cpuid asimdrdm lrcpc dcpop asimddp
CPU implementer : 0x41
CPU architecture: 8
CPU variant     : 0x4
CPU part       : 0xd0b
CPU revision   : 1

Revision       : d04170
Serial        : 0fcc7c123fada123
Model        : Raspberry Pi 5 Model B Rev 1.0
```

Nur die Zeile mit dem Revisionscode ausgeben

```
pi@pi5b:~ $ cat /proc/cpuinfo | grep "Revision"
Revision      : d04170
```

Nur den Revisionscode ausgeben 1

```
pi@pi5b:~ $ cat /proc/cpuinfo | grep 'Revision' | awk '{print $3}'  
d04170
```

Nur den Revisionscode ausgeben 2

```
pi@pi5b:~ $ cat /proc/cpuinfo | grep Revision | cut -d ':' -f  
2 | sed -e "s/ //g"  
d04170
```

Tabelle Revisionscodes

Revision	Release Datum	Modell	PCB Rev.	Speicher	Hersteller
Beta	2012 Q1	Pi 1 Model B (Beta)	-	256 MB	Beta Board
0002	2012 Q1	Pi 1 Model B	1.0	256 MB	?
0003	2012 Q3	Pi 1 Model B	1.0	256 MB	?
0004	2012 Q3	Pi 1 Model B	2.0	256 MB	Sony UK
0005	2012 Q4	Pi 1 Model B	2.0	256 MB	Qisda
0006	2012 Q4	Pi 1 Model B	2.0	256 MB	Egoman
000d	2012 Q4	Pi 1 Model B	2.0	512 MB	Egoman
000e	2012 Q4	Pi 1 Model B	2.0	512 MB	Sony UK
000f	2012 Q4	Pi 1 Model B	2.0	512 MB	Qisda
0007	2013 Q1	Pi 1 Model A	2.0	256 MB	Egoman
0008	2013 Q1	Pi 1 Model A	2.0	256 MB	Sony UK

Revision	Release Datum	Modell	PCB Rev.	Speicher	Hersteller
0009	2013 Q1	Pi 1 Model A	2.0	256 MB	Qisda
0011	2014 Q2	Compute Module 1	1.0	512 MB	Sony UK
0014	2014 Q2	Compute Module 1	1.0	512 MB	Embest
0010	2014 Q3	Pi 1 Model B+	1.0	512 MB	Sony UK
0012	2014 Q4	Pi 1 Model A+	1.1	256 MB	Sony UK
0013	2015 Q1	Pi 1 Model B+	1.2	512 MB	?
0015	2015 Q?	Pi 1 Model A+	1.1	256 MB / 512 MB	Embest
a01041	2015 Q1	Pi 2 Model B	1.1	1024 MB	Sony UK
a21041	2015 Q1	Pi 2 Model B	1.1	1024 MB	Embest
900092	2015 Q4	Pi Zero	1.2	512 MB	Sony UK
a01040	2015 Q?	Pi 2 Model B	1.0	1024 MB	Sony UK
a02042	2015 Q?	Pi 2 Model B (BCM2837)	1.2	1024 MB	Sony UK
a02082	2016 Q1	Pi 3 Model B	1.2	1024 MB	Sony UK
a22082	2016 Q1	Pi 3 Model B	1.2	1024 MB	Embest
900032	2016 Q2	Pi 1 Model B+	1.2	512 MB	Sony UK
900093	2016 Q2	Pi Zero	1.3	512 MB	Sony UK / Embest
920093	2016 Q2	Pi Zero	1.3	512 MB	Embest
900021	2016 Q3	Pi 1 Model A+	1.1	512 MB	Sony UK
a22042	2016 Q3	Pi 2 Model B (BCM2837)	1.2	1024 MB	Embest

Revision	Release Datum	Modell	PCB Rev.	Speicher	Hersteller
a32082	2016 Q4	Pi 3 Model B	1.2	1024 MB	Sony Japan
a220a0	2016 Q?	Compute Module 3	1.0	1024 MB	Embest
9000c1	2017 Q1	Pi Zero W	1.1	512 MB	Sony UK
9000c1	2017 Q1	Pi Zero W	1.3	512 MB	Embest
a020a0	2017 Q1	Compute Module 3 / Lite	1.0	1024 MB	Sony UK
a020d3	2018 Q1	Pi 3 Model B+	1.3	1024 MB	Sony UK
a22083	2018 Q1	Pi 3 Model B	1.3	1024 MB	Embest
9020e0	2018 Q4	Pi 3 Model A+	1.1	512 MB	Sony UK
a52082	2018 Q?	Pi 3 Model B	1.2	1024 MB	Sony UK / Stadium
a02100	2019 Q1	Compute Module 3+	1.0	1024 MB	Sony UK
a03111	2019 Q2	Pi 4 Model B	1.1	1024 MB	Sony UK
b03111	2019 Q2	Pi 4 Model B	1.1	2048 MB	Sony UK
b03112	2019 Q2	Pi 4 Model B	1.2	2048 MB	Sony UK
c03111	2019 Q2	Pi 4 Model B	1.1	4096 MB	Sony UK
c03112	2019 Q2	Pi 4 Model B	1.2	4096 MB	Sony UK
b03114	2020 Q2	Pi 4 Model B	1.4	2048 MB	Sony UK
c03114	2020 Q2	Pi 4 Model B	1.4	4096 MB	Sony UK
d03114	2020 Q2	Pi 4 Model B	1.4	8192 MB	Sony UK
902120	2021 Q4	Pi Zero 2 W	1.0	512 MB	Sony UK
b03115	2022 Q1	Pi 4 Model B	1.5	2048 MB	Sony UK
c03115	2022 Q1	Pi 4 Model B	1.5	4096 MB	Sony UK
d03115	2022 Q1	Pi 4 Model B	1.5	8192 MB	Sony UK

Revision	Release Datum	Modell	PCB Rev.	Speicher	Hersteller
c03130		Pi 400	1.0	4096 MB	Sony UK
a03140		Compute Module 4	1.0	1024 MB	Sony UK
b03140		Compute Module 4	1.0	2048 MB	Sony UK
c03140		Compute Module 4	1.0	4096 MB	Sony UK
d03140		Compute Module 4	1.0	8192 MB	Sony UK
902120		Pi Zero 2 W	1.0	512 MB	Sony UK
c04170		Pi 5 Model	1.0	4096 MB	Sony UK
d04170		Pi 5 Model	1.0	8192 MB	Sony UK

Pi OS: Kernel- und Firmware (SoC) Update

geschrieben von Andreas Potthoff | 25. Juni 2024

Getestet mit Raspberry Pi OS

```
pi@pi5b:~ $ cat /etc/os-release | grep PRETTY_NAME  
PRETTY_NAME="Debian GNU/Linux 12 (bookworm)"
```

```
pi@pi5b:~ $ uname -a  
Linux pi5b 6.6.31+rpt-rpi-2712 #1 SMP PREEMPT Debian  
1:6.6.31-1+rpt1 (2024-05-29) aarch64 GNU/Linux
```

rpi-update

Ursprünglich hat Hexxeh ein Script (rpi-update) geschrieben um ein Kernel- und Firmware-Update durchzuführen. Das Script wird auf [Github](https://github.com/raspberrypi/rpi-update) zur Verfügung gestellt:
<https://github.com/raspberrypi/rpi-update>

rpi-update ist eigentlich nur für den Gebrauch mit Raspberry Pi OS gedacht. Wenn Sie eine andere Distribution verwenden, dann vergewissern Sie sich, ob rpi-update sicher ist.

Vorher ist unbedingt ein Update / Upgrade durchzuführen, sonst kann es zu Problemen kommen.

```
sudo apt update  
sudo apt upgrade
```

Es wird der neueste Kernel (Entwicklerversion) und die neueste Firmware für den SoC (GPU) installiert:

```
sudo rpi-update  
sudo reboot
```

Warnung

Auch wenn Sie Raspberry Pi OS benutzen, sollten Sie sich im Klaren darüber sein, dass ein rpi-update Sicherheitsprobleme mit sich bringen kann und Ihr System unter Umständen instabil wird. Bug-Fixes und Verbesserungen werden ausschließlich über die offiziellen Raspberry Pi OS Releases mit apt-get zur Verfügung gestellt, wenn sie als ausreichend getestet betrachtet wurden.

Führen Sie ein rpi-update nur aus, wenn Sie genau wissen was Sie tun! Es gibt keine Undo-Funktion.

Es ist ratsam, vor einem rpi-update, ein **Backup** des aktuellen Betriebssystems (Image) zu erstellen!

Nachträgliche Installation

In den neueren Raspberry Pi OS Distributionen ist rpi-update bereits vorhanden. Falls dieses Paket nicht vorhanden sein sollte, kann man mit

```
sudo apt install rpi-update
```

das rpi-update Script nachträglich installieren.

Update-Verlauf

Kernel-Version:

```
pi@pi5b:~ $ uname -a
Linux pi5b 6.6.31+rpt-rpi-2712 #1 SMP PREEMPT Debian
1:6.6.31-1+rpt1 (2024-05-29) aarch64 GNU/Linux
```

Firmware-Version:

```
pi@pi5b:~ $ vcgencmd version
2024/04/20 11:53:30
Copyright (c) 2012 Broadcom
version d1744d21 (release) (embedded)
```

Firmware- und Kernel-Update:

```
pi@pi5b:~ $ sudo rpi-update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by
AndrewS and Dom
*** Performing self-update
*** Relaunching after update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by
AndrewS and Dom
FW_REV:01be90cf9623fb08984a37ead6e6b4e703f61003
BOOTLOADER_REV:e430a41e7323a1e28fb42b53cf79e5ba9b5ee975
*** We're running for the first time
```

```

*** Backing up files (this will take a few minutes)
*** Remove old firmware backup
*** Backing up firmware
*** Remove old modules backup
*** Backing up modules 6.6.31+rpt-rpi-2712
WANT_32BIT:0 WANT_64BIT:1 WANT_PI4:1 WANT_PI5:1
#####
WARNING: This update bumps to rpi-6.6.y linux tree
See: https://forums.raspberrypi.com/viewtopic.php?p=2191175

```

'rpi-update' should only be used if there is a specific reason to do so - for example, a request by a Raspberry Pi engineer or if you want to help the testing effort and are comfortable with restoring if there are regressions.

```

DO NOT use 'rpi-update' as part of a regular update process.
#####
Would you like to proceed? (y/N)

```

```

Downloading bootloader tools
Downloading bootloader images
*** Downloading specific firmware revision (this will take a
few minutes)
% Total      % Received % Xferd  Average Speed   Time    Time
Time  Current
                                Dload  Upload  Total  Spent
Left  Speed
  0      0    0     0    0     0     0     0  --:--:--  --:--:--
- --:--:--    0
100  142M  100  142M    0     0  21.5M    0  0:00:06
0:00:06  --:--:--  23.7M
*** PREPARING EEPROM UPDATES ***

```

```

BOOTLOADER: update available
CURRENT: Sat 20 Apr 10:53:30 UTC 2024 (1713610410)
LATEST: Wed 5 Jun 15:41:49 UTC 2024 (1717602109)
                RELEASE:                latest
(/lib/firmware/raspberrypi/bootloader-2712/latest)
        Use raspi-config to change the release.
CURRENT: Sat 20 Apr 10:53:30 UTC 2024 (1713610410)
UPDATE: Wed 5 Jun 15:41:49 UTC 2024 (1717602109)

```

```
B00TFS: /boot/firmware
'/tmp/tmp.34dkfYAh64' -> '/boot/firmware/pieeprom.upd'
```

UPDATING bootloader. This could take up to a minute. Please wait

```
*** Do not disconnect the power until the update is complete
***
```

If a problem occurs then the Raspberry Pi Imager may be used to create a bootloader rescue SD card image which restores the default bootloader image.

```
flashrom -p linux_spi:dev=/dev/spidev10.0,spispeed=16000 -w
/boot/firmware/pieeprom.upd
```

UPDATE SUCCESSFUL

```
*** Updating firmware
*** Updating kernel modules
*** depmod 6.6.35-v8-16k+
*** depmod 6.6.35-v8+
*** Updating VideoCore libraries
*** Running ldconfig
*** Storing current firmware revision
*** Deleting downloaded files
*** Syncing changes to disk
*** If no errors appeared, your firmware was successfully
updated to 01be90cf9623fb08984a37ead6e6b4e703f61003
*** A reboot is needed to activate the new firmware
```

```
sudo reboot
```

Kernel-Version:

```
pi@pi5b:~ $ uname -a
Linux pi5b 6.6.35-v8-16k+ #1779 SMP PREEMPT Tue Jun 25
11:38:45 BST 2024 aarch64 GNU/Linux
```

Firmware-Version:

```
pi@pi5b:~ $ vcgencmd version
2024/06/05 16:41:49
Copyright (c) 2012 Broadcom
```

Tipps

Sollte vor dem rpi-update kein Update / Upgrade durchgeführt worden sein, kann es zu Problemen kommen. Es wird ein älterer Kernel bzw. ältere Firmware angezeigt, obwohl rpi-update ein up to date meldet und ein neuere Firmware vorhanden ist. Gehen Sie dann wie folgt vor:

```
sudo rm /boot/.firmware_revision
```

Wenn Sie dann das rpi-update starten wird der Update auch ausgeführt.

```
sudo rpi-update  
sudo reboot
```

RPi: Alternative Betriebssysteme für den Raspberry Pi

geschrieben von Andreas Potthoff | 25. Juni 2024

Betriebssysteme

Für den Raspberry Pi sind viele Open-Source-Betriebssysteme verfügbar. Die meisten Betriebssysteme basieren auf Linux, dies hat den Hintergrund, dass diese speziell auf die ARM-Architektur des RasPi angepasst sind. Hier können Sie sich einen schnellen Überblick über **mehr als 120 verschiedene**

Betriebssysteme verschaffen.

Bei der Auswahl des Betriebssystems ist für jeden Geschmack etwas dabei – ob einfaches Allrounder-System, benutzerfreundliches Multi-Media-Center, ultraschlankes Entwicklersystem oder exotische Linux-Weiterentwicklungen.

Das Debian-basierte Raspbian wird offiziell von der Raspberry Pi Foundation unterstützt und gilt als Standard-Distribution. Es ist mit Abstand am weitesten verbreitet, da es auf die Hardware des RasPi optimiert wurde, es eine fensterbasierte GUI besitzt und somit sehr einsteigerfreundlich ist.

Die Installation eines Betriebssystems geschieht entweder mit dem einfach zu verwendenden *Pi Imager* oder durch das Klonen eines Images auf die SSD, HD, SD-Karte.

Dies ist sicherlich keine vollständige Liste der Betriebssysteme für den RasPi. Auf nicht kostenlose Betriebssysteme habe ich in der Liste verzichtet. Die mit einem * gekennzeichneten Betriebssysteme werden nicht mehr weiterentwickelt oder sind obsolet und werden nur der Vollständigkeit halber gelistet.

Es wäre nett, wenn Sie defekte Links, fehlende Betriebssysteme oder Tabellen-Informationen bitte in den Kommentaren hinterlassen würden. Danke!

Tabelle Betriebssysteme

Betriebssystem	Basis	Einsatz	ARMHF	User:Pass	Download Webseite
9front System	Unabhängig Fork von: Plan 9	Verteilte Rechnerumgebung, Forschung	?	?	Download Webseite
Adafruit Occidentalis *	Debian	Maker	?	?	Download Webseite
AEROS	Debian + AROS	AmigaOS Emulation	?	?	Download Webseite
Alpine Linux ARM	Unabhängig	Netzwerk, Sicherheit	ja	root:	Download Webseite

Betriebssystem	Basis	Einsatz	ARMHF	User:Pass	Download Webseite
Arch Linux ARM	Unabhängig	Leichtgewicht	ja	root:root	Download Webseite
Archphile *	Arch	Entwicklung	ja	root:archphile	Download Webseite
arkOS *	Arch	Leichtgewicht, Web- und Cloud- Dienste	?	?	Download Webseite
BerryBoot	-	Multi OS Installer	-	-	Download Webseite
BerryTerminal *	?	Thin Client, LTSP (DHCP)	?	?	Download Webseite
BlackArch Linux	Arch	Testing	?	root:blackarch	Download Webseite
Bodhi Linux ARM	Debian	Leichtgewicht	ja	?	Download Webseite
CentOS ARM Minimal	Fedora	Leichtgewicht	ja	root:(Setup)	Download Webseite
ChibiOS/RT	?	Echtzeit OS	?	?	Download Webseite
Commodore Pi *	?	C-64 OS Emulation	?	?	Download Webseite
CRUX ARM	Unabhängig	Leichtgewicht	?	?	Download Webseite
DarkeLEC	Unabhängig Fork von: OpenELEC	Media Center	?	?	Download Webseite
Debian ARM	Debian	Allrounder	ja	root:(Setup)	Download Webseite
deepin	Debian	Allrounder	ja	?	Download Webseite
Devuan GNU+Linux	Debian	Entwicklung	ja	?	Download Webseite
DietPi	?	Leichtgewicht	?	?	Download Webseite
FedBerry	Fedora	Allrounder	ja	root:(Setup) root:fedberry	Download Webseite
Fedora ARM	Fedora	Allrounder	ja	root:(Setup)	Download Webseite
Finnix	Debian	Administrator	ja	?	Download Webseite
Firefox OS *	Linux	Browser OS	?	?	Download Webseite
FreeBSD	Unabhängig	Allrounder	?	root:(Setup)	Download Webseite
FreeRTOS	?	Echtzeit OS	?	?	Download Webseite
Funtoo Linux ARM	Gentoo	Allrounder	ja	?	Download Webseite
GeeXboX	Unabhängig	Leichtgewicht, Media Center	?	?	Download Webseite
Gentoo Linux ARM	Gentoo	Allrounder	ja	?	Download Webseite
Ha-Pi	?	Netzwerk, Sicherheit	?	?	Download Webseite
Haiku OS *	Früher: OpenBeOS	Personal Computing	?	?	Download Webseite
HypriotOS	Debian	Docker Container	?	?	Download Webseite
I2PBerry	Debian	I2P-Knoten, Router	?	?	Download Webseite
Instant WebKiosk	Debian	Web Kiosk	?	?	Download Webseite

Betriebssystem	Basis	Einsatz	ARMHF	User:Pass	Download Webseite
IPFire	Linux From Scratch	Firewall, Router	?	root:(Setup)	Download Webseite
KA-Pi *	?	Bildung	?	?	Download Webseite
Kali Linux ARM	Debian	Netzwerk, Sicherheit	ja	root:toor	Download Webseite 2
Kano OS	?	Entwicklung DIY Kits	?	root:kano	Download Webseite
Kodi	Verschiedene (früher XMBC)	Media Center	-	-	Download Webseite
Lakka	LibreELEC	Games (Retro)	?	root:root	Download Webseite
LibreELEC	Unabhängig Fork von: OpenELEC	Media Center	ja	root:libreelec	Download Webseite
LinutopOS	Debian	Werbe- und Informationssystem	?	?	Download Webseite
Lubuntu	Debian	Allrounder	ja	?	Download Webseite
Max2Play	?	Musik	ja	root:(Setup) pi:raspberrypi su:max2play	Download Webseite
MinePeon ARM *	?	Mining Plattform	?	?	Download Webseite
Mini vMac *	?	Mac Emulator	?	?	Download Webseite
Minibian	Debian	Leichtgewicht	?	?	Download Webseite
Moebius	?	Leichtgewicht	ja	?	Download Webseite
motionEyeOS	?	Sicherheit, Video Überwachung	?	?	Download Webseite
Nard SDK	?	Entwicklung, Pi Emulation	?	?	Download Webseite
NAS4Free	FreeBSD	NAS	?	?	Download Webseite
NetBSD	Unabhängig	Allrounder	ja	?	Download Webseite
NetBSD ARM	Unabhängig	Allrounder	?	root:(Setup)	Download Webseite
N00BS	-	Multi OS Installer	-	-	Download Webseite
N00BS Lite	-	Multi OS Installer	-	-	Download Webseite
nOS *	Debian	Leichtgewicht	?	?	Download Webseite
OctoPi	Debian	3D Printing	?	root:(Setup)	Download Webseite
OpenELEC	Unabhängig	Media Center	ja	root:openelec	Download Webseite
OpenMandriva Lx	Unabhängig	Allrounder	?	?	Download Webseite
OpenMediaVault	?	NAS	ja	root:(Setup) admin:openmediavault	Download Webseite
OpenSUSE ARM	Unabhängig	Allrounder	ja	root:linux	Download RPi 1 Download RPi 2 Download RPi 3

Betriebssystem	Basis	Einsatz	ARMHF	User:Pass	Download Webseite
OpenWrt	?	Netzwerk, Wireless Router	nein	root:(Setup)	Download Webseite
OSMC	Debian	Media Center	ja	osmc:osmc	Download Webseite
Pardus Topluluk ARM	Debian	Maker	ja	?	Download Webseite
Peach OSI	Debian	Allrounder	?	?	Download Webseite
Pi MusicBox	?	Musik	-	root:musicbox	Download Webseite
Pi-Point	Debian	Wireless Access Point	ja	pi:raspberry	Download Webseite
PiBang Linux *	Debian	Leichtgewicht	ja	?	Download Webseite
PiBox	?	Entwicklung	?	?	Download Webseite
Pidora	Fedora	Allrounder	?	?	Webseite Webseite
PiLFS	?	Entwicklung	?	?	Download Webseite
PiNet	?	Netzwerk, Bildung	ja	?	Download Webseite
pipaOS	Debian	Leichtgewicht	?	?	Download Webseite
PiPlay *	?	Spiele, Emulation	?	?	Download Webseite
Plan 9 *	Unabhängig	Verteilte Rechnerumgebung, Forschung	?	?	Download Webseite
Plop Linux	Unabhängig	Leichtgewicht, System-Tools	?	?	Download Webseite
PrimTux	Debian	Bildung	ja	?	Download Webseite
Puppy Linux	Unabhängig	Leichtgewicht	?	?	Download Webseite
PWN Pi	Debian	Testing Plattform	?	?	Download Webseite
Q4OS	Debian	Allrounder	?	?	Download Webseite
QtonPi	-	Qt SDK	-	-	Download Webseite
Quirky	Unabhängig	Allrounder	ja	?	Download Webseite
RancherOS	?	Docker Container	?	?	Download Webseite
RasDOS	?	DOS Emulation	?	?	Download Webseite
Raspberry Digital Signage	Debian	Werbe- und Informationssystem	ja	?	Download Webseite
Raspberry Pi Thin Client	?	Netzwerk, Client	?	?	Download Webseite
Raspberry Slideshow	Debian	Slideshow	ja	?	Download Webseite
Raspbian for Robots	Debian	Dexter Industries Roboter	?	?	Webseite
Raspbian Jessie	Debian	Allrounder	ja	root:raspberry	Download Webseite
Raspbian Jessie Lite	Debian	Leichtgewicht	ja	root:raspberry	Download Webseite
Raspbian Pices	Debian	Leichtgewicht	ja	root:raspbian raspbian:raspbian	Download Webseite

Betriebssystem	Basis	Einsatz	ARMHF	User:Pass	Download Webseite
Raspbian Server Edition *	Debian	Server	?	?	Download Webseite
RaspBSD	FreeBSD	Allrounder	ja	root:(Setup) raspberry:raspberry	Download Webseite
RaspEX	Debian	Media Center	?	?	Download Webseite
Raspicade *	?	Games (Retro)	?	?	Download Webseite
RasPlex	OpenELEC	Media Center	-	root:rasplex	Download Webseite
RazDroid	Android Port	Mobil OS	-	-	Download Webseite
RecalboxOS	?	Games (Retro), Media Center	-	root:recalboxroot	Download Webseite
resinOS	?	Docker Container	?	?	Download Webseite
RetroPie	Debian	Games (Retro)	-	root:(Setup)	Download Webseite
RISC OS Open	Unabhängig	RISC Entwicklung	nein	-	Download Webseite
ROSberryPi	Debian	Roboter	?	?	Download Webseite
RTAndroid	?	Echtzeit Entwicklung	ja	?	Download Webseite
RuneAudio	?	Musik	-	?	Download Webseite
Sabayon Linux ARM	Gentoo	Allrounder	ja	sabayon:sabayon root:root	Download Webseite
Sailfish OS	Unabhängig	Mobil OS	?	?	Download Webseite
Slackware ARM	Unabhängig	Server	?	?	Download Webseite
SlaXBMC	?	Media Center	?	?	Download Webseite
SliTaz GNU/Linux	Unabhängig	Leichtgewicht	?	?	Download Webseite
slrpi *	Debian	Leichtgewicht	?	?	Download Webseite
Snappy Ubuntu Core	Debian	Leichtgewicht	ja	ubuntu:ubuntu	Download Webseite
ThinBox *	?	Netzwerk	?	?	Download Webseite
ThingBox	?	IoT	?	?	Download Webseite
Tiny Core Linux	?	Leichtgewicht	?	?	Download Webseite
Tizen 3	Unabhängig	Mobil OS	-	root:root	Download Webseite
UBOS	Arch	Netzwerk, Server	ja	root:(Setup)	Download Webseite
Ubuntu 14.04 LTS	Debian	Allrounder	ja	ubuntu:ubuntu	Download Webseite
Ubuntu 16.04 LTS Classic	Debian	Allrounder	ja	ubuntu:ubuntu	Download Webseite
Ubuntu Mate	Debian	Allrounder	ja	root:(Setup)	Download Webseite
Ultimate Edition	Debian	Allrounder	ja	?	Download Webseite
Void Linux ARM	Unabhängig	Allrounder	?	?	Download Webseite
Volumio	Debian	Musik	ja	pi:raspberry volumio:volumio	Download Webseite
Windows 10 IoT Core	Windows	Windows IoT	nein	?	Download Webseite

Betriebssystem	Basis	Einsatz	ARMHF	User:Pass	Download Webseite
WTware	?	Netzwerk	?	?	Download Webseite
XBian	Debian	Media Center	ja	xbian:raspberry	Download Webseite
xv6	?	Bildung	?	?	Download Webseite

Raspberry Pi OS – Betriebssysteme & Tools

geschrieben von Andreas Potthoff | 25. Juni 2024

Einleitung

Das offiziell unterstützte Betriebssystem für den Raspberry Pi ist **Raspberry Pi OS** (früher Raspbian) und ist basiert auf Debian. Die aktuelle Version von Pi OS basiert auf Debian 12 (Bookworm), die älteren Versionen basieren auf Debian (Bullseye, Buster, Stretch, Jessie, Wheezy). Der Raspi unterstützt auch Betriebssysteme (Distributionen) von anderen Organisationen.




Um Speicherplatz und Traffic zu sparen werden hier nur die letzten beiden Versionen der jeweiligen Release angeboten.

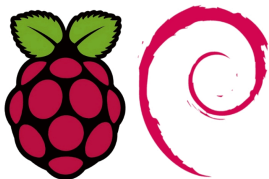


Raspberry Pi Imager

Raspberry Pi Imager ist die schnelle und einfache Möglichkeit, Raspberry Pi OS und andere Betriebssysteme auf einer microSD-Karte zu installieren, die sofort mit Ihrem Raspberry Pi verwendet werden kann. Sehen Sie sich das Video an, um zu erfahren, wie Sie ein Betriebssystem mit Raspberry Pi Imager installieren.

Laden Sie Raspberry Pi Imager herunter und installieren Sie es auf einem Computer mit einem SD-Kartenleser. Stecken Sie die SD-Karte, die Sie mit Ihrem Raspberry Pi verwenden, in das Lesegerät und führen Sie Raspberry Pi Imager aus.

	Titel	Kategorien	Datum der Veröffentlichung	Download
	Raspberry Pi Imager 1.8.5 (Ubuntu, x86) 1 Datei(en) 34 Downloads	Pi Imager	21. Januar 2024	Download578.63 KB
	Raspberry Pi Imager 1.8.5 (MAC) 1 Datei(en) 34 Downloads	Pi Imager	21. Januar 2024	Download63.08 MB
	Raspberry Pi Imager 1.8.5 (WIN) 1 Datei(en) 29 Downloads	Pi Imager	21. Januar 2024	Download19.33 MB








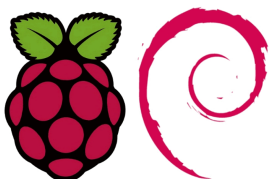
Raspberry Pi OS

Raspberry Pi OS

Images für die manuelle Installation.

- Standard: Desktop
- Full: Desktop + Recommended Software
- Lite: Ohne Desktop

	Titel	Kategorien	Datum der Veröffentlichung	Download
	Raspberry Pi OS Bookworm Lite 2023-12-11 (Image, 64-bit) 1 Datei(en) 51 Downloads	Pi OS	11. Dezember 2023	Download432.95 MB
	Raspberry Pi OS Bookworm Lite 2023-12-11 (Image, 32-bit) 1 Datei(en) 57 Downloads	Pi OS	11. Dezember 2023	Download519.70 MB
	Raspberry Pi OS Bookworm Full 2023-12-05 (Image, 64-bit) 1 Datei(en) 73 Downloads	Pi OS	5. Dezember 2023	Download2,732.28 MB
	Raspberry Pi OS Bookworm 2023-12-05 (Image, 64-bit) 1 Datei(en) 35 Downloads	Pi OS	5. Dezember 2023	Download818.32 MB
	Raspberry Pi OS Bookworm 2023-12-05 (Image, 32-bit) 1 Datei(en) 36 Downloads	Pi OS	5. Dezember 2023	Download1,253.78 MB
	Raspberry Pi OS Bookworm Full 2023-12-05 (Image, 32-bit) 1 Datei(en) 48 Downloads	Pi OS	5. Dezember 2023	Download2,601.68 MB











Raspberry Pi OS

Raspberry Pi OS (Legacy)


Obsoleter!

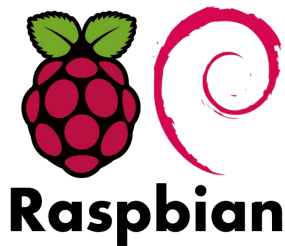
Images für die manuelle Installation.

- Standard: Desktop
- Full: Desktop + Recommended Software
- Lite: Ohne Desktop

	Titel	Kategorien	Datum der Veröffentlichung	Download
	Raspberry Pi OS Bookworm Full 2023-10-10 (Image, 32-bit) 1 Datei(en) 49 Downloads	Pi OS Legacy	10. Oktober 2023	Download2,605.53 MB
	Raspberry Pi OS Bookworm 2023-10-10 (Image, 32-bit) 1 Datei(en) 37 Downloads	Pi OS Legacy	10. Oktober 2023	Download1,238.90 MB
	Raspberry Pi OS Bookworm 2023-10-10 (Image, 64-bit) 1 Datei(en) 40 Downloads	Pi OS Legacy	10. Oktober 2023	Download1,093.44 MB
	Raspberry Pi OS Bookworm Full 2023-10-10 (Image, 64-bit) 1 Datei(en) 39 Downloads	Pi OS Legacy	10. Oktober 2023	Download2,725.42 MB
	Raspberry Pi OS Bookworm Lite 2023-10-10 (Image, 32-bit) 1 Datei(en) 54 Downloads	Pi OS Legacy	10. Oktober 2023	Download568.40 MB
	Raspberry Pi OS Bookworm Lite 2023-10-10 (Image, 64-bit) 1 Datei(en) 37 Downloads	Pi OS Legacy	10. Oktober 2023	Download435.48 MB
	Raspberry Pi OS Bullseye Full 2023-05-03 (Image, 32-bit) 1 Datei(en) 92 Downloads	Pi OS Legacy	3. Mai 2023	Download2,700.65 MB
	Raspberry Pi OS Bullseye 2023-05-03 (Image, 32-bit) 1 Datei(en) 44 Downloads	Pi OS Legacy	3. Mai 2023	Download871.88 MB

	Titel	Kategorien	Datum der Veröffentlichung	Download
XZ -	Raspberry Pi OS Bullseye Lite 2023-05-03 (Image, 32-bit) 1 Datei(en) 81 Downloads	Pi OS Legacy	3. Mai 2023	Download363.88 MB
XZ -	Raspberry Pi OS Bullseye 2023-05-03 (Image, 64-bit) 1 Datei(en) 56 Downloads	Pi OS Legacy	3. Mai 2023	Download818.32 MB
XZ -	Raspberry Pi OS Bullseye Lite 2023-05-03 (Image, 64-bit) 1 Datei(en) 38 Downloads	Pi OS Legacy	3. Mai 2023	Download307.56 MB
XZ -	Raspberry Pi OS Buster Lite 2023-05-03 (Image, 32-bit) 1 Datei(en) 34 Downloads	Pi OS Legacy	3. Mai 2023	Download320.71 MB
XZ -	Raspberry Pi OS Buster 2023-05-03 (Image, 32-bit) 1 Datei(en) 37 Downloads	Pi OS Legacy	3. Mai 2023	Download788.03 MB
XZ -	Raspberry Pi OS Bullseye Full 2023-02-22 (Image, 32-bit) 1 Datei(en) 36 Downloads	Pi OS Legacy	22. Februar 2023	Download2,751.51 MB
XZ -	Raspberry Pi OS Bullseye 2023-02-22 (Image, 64-bit) 1 Datei(en) 39 Downloads	Pi OS Legacy	22. Februar 2023	Download816.43 MB
XZ -	Raspberry Pi OS Bullseye Lite 2023-02-22 (Image, 64-bit) 1 Datei(en) 43 Downloads	Pi OS Legacy	22. Februar 2023	Download307.05 MB
XZ -	Raspberry Pi OS Bullseye 2023-02-22 (Image, 32-bit) 1 Datei(en) 42 Downloads	Pi OS Legacy	22. Februar 2023	Download923.80 MB
XZ -	Raspberry Pi OS Bullseye Lite 2023-02-22 (Image, 32-bit) 1 Datei(en) 36 Downloads	Pi OS Legacy	22. Februar 2023	Download361.54 MB
XZ -	Raspberry Pi OS Buster 2023-02-22 (Image, 32-bit) 1 Datei(en) 121 Downloads	Pi OS Legacy	22. Februar 2023	Download758.42 MB
XZ -	Raspberry Pi OS Buster Lite 2023-02-22 (Image, 32-bit) 1 Datei(en) 47 Downloads	Pi OS Legacy	22. Februar 2023	Download285.58 MB

Titel	Kategorien	Datum der Veröffentlichung	Download
 Raspberry Pi OS Bullseye for Desktop 2022-07-01 (PC/MAC) 1 Datei(en) 58 Downloads	Pi OS Legacy	1. Juli 2022	Download3,440.00 MB






Raspbian

Obsoleter!

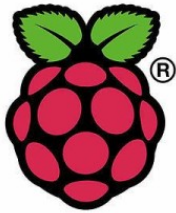
Images für die manuelle Installation.

- Standard: Desktop
- Full: Desktop + Recommended Software
- Lite: Ohne Desktop

Titel	Kategorien	Datum der Veröffentlichung	Download
 Raspbian Buster Full 2020-02-14 (Image, 32-bit) 1 Datei(en) 29 Downloads	Pi Raspbian	14. Februar 2020	Download2,530.79 MB
 Raspbian Buster 2020-02-14 (Image, 32-bit) 1 Datei(en) 23 Downloads	Pi Raspbian	14. Februar 2020	Download1,136.55 MB

	Titel	Kategorien	Datum der Veröffentlichung	Download
	Raspbian Buster Lite 2020-02-14 (Image, 32-bit) 1 Datei(en) 45 Downloads	Pi Raspbian	14. Februar 2020	Download433.01 MB



“NOOBS”
New Out Of Box Software



Raspberry Pi NOOBS

Obsoleter!

Dieser Installationsassistent wurde durch den **Pi Imager** ersetzt.

	Titel	Kategorien	Datum der Veröffentlichung	Download
	Raspberry Pi NOOBS 3.8.1 (Image, 32-bit) 1 Datei(en) 40 Downloads	Pi NOOBS	7. April 2022	Download2,814.21 MB
	Raspberry Pi NOOBS 3.8 Lite (Image, 32-bit) 1 Datei(en) 36 Downloads	Pi NOOBS	28. Januar 2022	Download40.01 MB

Die Verzeichnisstruktur von Linux (Debian based)

geschrieben von Andreas Potthoff | 25. Juni 2024

Dieser Beitrag beschreibt die typische Verzeichnisstruktur für Linux-Systeme, welche auf Debian basieren. Dazu gehören z.B.: Ubuntu, Raspberry Pi OS, Raspbian, Linux Mint, Kali Linux, Kubuntu, Proxmox u.v.m. Eine umfangreiche Liste von Debian based-Linux-Distributionen findet ihr hier: [Distrowatch.com](https://distrowatch.com) – Debian based

Alle elementaren Programme und Konfigurationsdateien liegen in verschiedenen Systemverzeichnissen. Anhand des Filesystem Hierarchy Standard (FHS) wird die Eingliederung der verschiedenen Dateiarten in die Verzeichnisstruktur gezeigt. Der FHS ist eine Richtlinie für die Verzeichnisstruktur unter Unix-ähnlichen Betriebssystemen.

Sie können den folgenden Informationen entnehmen, wo Sie Konfigurationsdateien finden, welche Verzeichnisse Programme enthalten und wo Dokumentationen zu finden sind.

Die folgende tabellarische Übersicht der Verzeichnisstruktur berücksichtigt nicht eine Installation mit gesonderten Partionen (z.B. boot, home, etc.). Außerdem wird Beschreibung der Dateisystem-Hierarchie durch den Konsolenbefehl **man** (manual) gezeigt.

Tabelle Verzeichnisstruktur

Die *wichtigsten* Verzeichnisse haben folgende Bedeutungen bzw. Inhalte:

Verzeichnis

Beschreibung

/

Wurzelverzeichnis: Das Wurzelverzeichnis und oberste Verzeichnis der Hierarchie.

Verzeichnis

Beschreibung

/bin	Systemprogramme: Die wichtigsten Systemkommandos. Dieses Verzeichnis darf keine Unterverzeichnisse enthalten.
/boot	Bootloader: Die statischen Dateien des Bootloaders und die Kernel.
/dev	Geräte-dateien: Schnittstellen zur Ansteuerung der gesamten Hardware.
/etc	Systemkonfiguration: Die Konfigurationsdateien.
/etc/init.d	Start- und Stopscripte.
/etc/network	Konfigurationsdateien des Netzwerkes. z.B. interfaces
/etc/opt	Konfigurationsdateien für Programme im Verzeichnis /opt
/home/Benutzername	Heimatverzeichnis: Die eigenen Dateien. Für jeden Benutzer finden Sie einen eigenes Verzeichnis mit dem Benutzernamen.
/lib	Bibliotheken: Dynamische Bibliotheken und Kernelmodule die für den Systemstart notwendig sind.
/lib/modules	Kernelmodule
/lost+found	Verloren+Gefunden: Dateien und Dateifragmente die bei der Reparatur eines defekten Dateissystems übrig geblieben sind.
/media	Wechselmedien: Einhängpunkt und Unterverzeichnisse für transportable Medienspeicher (z.B. USB-Stick, externe Festplatte, Floppy, CD-ROM, DVD, u.a).
/mnt	Mountpunkt: Einhängpunkt für ein temporär eingehängtes Dateisystem.

Verzeichnis

Beschreibung

/opt	<p>Optional: Für die manuelle Installation von Programmen die ihre eigenen Bibliotheken mitbringen und nicht zum Standard der Distribution gehören. Unterverzeichnisse sind z.B. minecraft-pi, sonic-pi, vs, Wolfram</p>
/proc	<p>Prozess- und Systeminformationen: Schnittstellen zum aktuell geladenen Kernel und seinen Prozeduren. Dateien lassen sich mittel cat auslesen z.B. „cat proc/version“ gibt die aktuelle Kernelversion aus.</p>
/root	<p>Root Heimatverzeichnis: Das Heimatverzeichnis des Superusers (root). Das root-Verzeichnis liegt im Wurzelverzeichnis, falls das home-Verzeichnis auf eine andere Partiton ausgelagert wurde oder ein Zugriff auf home aus irgendeinem Grund nicht möglich ist.</p>
/run	<p>Prozesse: Dateien für laufende Prozesse. Hier befinden sich auch die meisten PID-Dateien (Process Identifier).</p>
/sbin	<p>Systemprogramme (root): Die Systemkommandos zur Systemverwaltung für die Rootrechte benötigt werden. Dieses Verzeichnis darf keine Unterverzeichnisse enthalten.</p>
/srv	<p>Systemdienste: Daten für Systemdienste (in der Regel leer). Dieses Verzeichnis ist noch nicht genau spezifiziert.</p>
/sys	<p>System: Systemweites Geräteverzeichnis welches Informationen und Statistiken über das System und die Komponenten enthält.</p>

Verzeichnis

Beschreibung

/tmp	Temporär: Temporäre Dateien von Programmen und Benutzern. Dieses Verzeichnis wird nach jedem Neustart automatisch geleert.
/usr	UNIX Systemressourcen: Die meisten Systemprogramme, Bibliotheken und installierten Programme.
/usr/bin	Die meisten Benutzerbefehle . Das primäre Verzeichnis für ausführbare Dateien des Systems.
/usr/games	Spiele
/usr/include	Header-Dateien für C-Programme.
/usr/lib	Allgemeine Bibliotheken
/usr/local	Distributionsunabhängige lokale Hierarchie enthält noch einmal die gleiche Verzeichnisstruktur wie das /usr Verzeichnis. Für Programme und Daten gedacht, die von der entsprechenden Distribution des jeweiligen Systems unabhängig installiert worden sind, wie etwa selbstkompilierte oder unabhängig von der Distribution heruntergeladene Programme und Dateien.
/usr/sbin	Weniger wichtige Systemprogramme die im Gegensatz zu /sbin nicht während des Bootvorganges verwendet werden.
/usr/share	Statische architekturunabhängige Dateien z.B. Dokumentationen, Manpages und Wörterbücher.
/usr/src	Source Code zu den Paketen.
/var	Variable Daten: Diese Daten entstehen z.B. im Zuge einer Abarbeitung
/var/backups	Sicherungskopien der Liste der installierten Programme (dpkg).

Verzeichnis	Beschreibung
/var/cache	Zwischenspeicher von Programmen
/var/lib	Variable Statusinformationen
/var/local	Variable Daten im Zusammenhang mit /usr/local
/var/lock	Lock-Dateien zur Prozesssynchronisation
/var/log	Protokolldateien
/var/mail	Mailboxen der Benutzer
/var/opt	Variable Daten der optionalen Programme
/var/run	Dateien zu laufenden Prozessen
/var/spool	Von Programmen gespoolte Daten z.B. Druckaufträge.
/var/tmp	Variable Daten die zwischen Reboots erhalten bleiben.
/var/www	Standardverzeichnis für Inhalte des Webservers .

Verzeichnisstruktur mit dem Konsolenbefehl tree anzeigen

Um sich einen Überblick der Verzeichnisstruktur zu verschaffen kann man den Konsolenbefehl **tree** benutzen:

Als erstes wechseln wir in das Wurzelverzeichnis und dann lassen wir uns den Verzeichnisbaum mit einer Leveltiefe von 1 anzeigen. Die Leveltiefe kann natürlich auch erweitert werden, indem man die 1 durch die gewünschte Tiefe ersetzt.

```
pi@pi4b:~ $ cd /
pi@pi4b:/ $ tree -L 1
```

```
.
├── bin
├── boot
├── boot.bak
├── dev
```

```
|— etc
|— home
|— lib
|— lost+found
|— media
|— mnt
|— opt
|— proc
|— root
|— run
|— sbin
|— srv
|— sys
|— tmp
|— usr
|— var
```

20 directories, 0 files

Beschreibung der Dateisystem-Hierarchie mit dem Konsolenbefehl man (englisch)

Wenn Sie folgenden Konsolenbefehl eingeben:

```
man hier
```

erhalten Sie die Beschreibung der Dateisystem-Hierarchie mit folgender Beispielausgabe:

```
HIER(7) Linux Programmer's Manual HIER(7)
```

```
NAME
```

```
hier - description of the filesystem hierarchy
```

```
DESCRIPTION
```

```
A typical Linux system has, among others, the following directories:
```

```
/ This is the root directory. This is where the whole tree
```

starts.

`/bin` This directory contains executable programs which are needed in single user mode and to bring the system up or repair it.

`/boot` Contains static files for the boot loader. This directory holds only the files which are needed during the boot process. The map installer and configuration files should go to `/sbin` and `/etc`.

`/dev` Special or device files, which refer to physical devices. See `mknod(1)`.

`/etc` Contains configuration files which are local to the machine. Some larger software packages, like X11, can have their own subdirectories below `/etc`. Site-wide configuration files may be placed here or in `/usr/etc`. Nevertheless, programs should always look for these files in `/etc` and you may have links for these files to `/usr/etc`.

`/etc/opt`

Host-specific configuration files for add-on applications installed in `/opt`.

`/etc/sgml`

This directory contains the configuration files for SGML and XML (optional).

`/etc/skel`

When a new user account is created, files from this directory are usually copied into the user's home directory.

`/etc/X11`

Configuration files for the X11 window system (optional).

`/home` On machines with home directories for users, these are usually beneath this directory, directly or not. The structure of this

directory depends on local administration decisions.

`/lib` This directory should hold those shared libraries that are necessary to boot the system and to run the commands in the root filesystem.

`/media` This directory contains mount points for removable media such as CD and DVD disks or USB sticks.

`/mnt` This directory is a mount point for a temporarily mounted filesystem. In some distributions, `/mnt` contains subdirectories intended to be used as mount points for several temporary filesystems.

`/opt` This directory should contain add-on packages that contain static files.

`/proc` This is a mount point for the `proc` filesystem, which provides information about running processes and the kernel. This pseudo-filesystem is described in more detail in `proc(5)`.

`/root` This directory is usually the home directory for the root user (optional).

`/sbin` Like `/bin`, this directory holds commands needed to boot the system, but which are usually not executed by normal users.

`/srv` This directory contains site-specific data that is served by this system.

`/tmp` This directory contains temporary files which may be deleted with no notice, such as by a regular job or at system boot up.

`/usr` This directory is usually mounted from a separate partition. It should hold only sharable, read-only data, so that it can be mounted by various machines running Linux.

`/usr/X11R6`

The X-Window system, version 11 release 6 (optional).

`/usr/X11R6/bin`

Binaries which belong to the X-Window system; often, there is a symbolic link from the more traditional `/usr/bin/X11` to here.

`/usr/X11R6/lib`

Data files associated with the X-Window system.

`/usr/X11R6/lib/X11`

These contain miscellaneous files needed to run X; Often, there is a symbolic link from `/usr/lib/X11` to this directory.

`/usr/X11R6/include/X11`

Contains include files needed for compiling programs using the X11 window system. Often, there is a symbolic link from `/usr/include/X11` to this directory.

`/usr/bin`

This is the primary directory for executable programs. Most programs executed by normal users which are not needed for booting or for repairing the system and which are not installed locally should be placed in this directory.

`/usr/bin/X11`

is the traditional place to look for X11 executables; on Linux, it usually is a symbolic link to `/usr/X11R6/bin`.

`/usr/dict`

Replaced by `/usr/share/dict`.

`/usr/doc`

Replaced by `/usr/share/doc`.

`/usr/etc`

Site-wide configuration files to be shared between several machines may be stored in this directory. However, commands should always reference those files using the `/etc` directory. Links

from files in /etc should point to the appropriate files in /usr/etc.

/usr/games

Binaries for games and educational programs (optional).

/usr/include

Include files for the C compiler.

/usr/include/X11

Include files for the C compiler and the X-Window system. This is usually a symbolic link to /usr/X11R6/include/X11.

/usr/include/asm

Include files which declare some assembler functions. This used to be a symbolic link to /usr/src/linux/include/asm.

/usr/include/linux

This contains information which may change from system release to system release and used to be a symbolic link to /usr/src/linux/include/linux to get at operating-system-specific information.

(Note that one should have include files there that work correctly with the current libc and in user space. However, Linux kernel

source is not designed to be used with user programs and does not know anything about the libc you are using. It is very likely

that things will break if you let /usr/include/asm and /usr/include/linux point at a random kernel tree. Debian systems don't do

this and use headers from a known good kernel version, provided in the libc*-dev package.)

/usr/include/g++

Include files to use with the GNU C++ compiler.

/usr/lib

Object libraries, including dynamic libraries, plus some executables which usually are not invoked directly. More complicated pro-

grams may have whole subdirectories there.

`/usr/lib/X11`

The usual place for data files associated with X programs, and configuration files for the X system itself. On Linux, it usually

is a symbolic link to `/usr/X11R6/lib/X11`.

`/usr/lib/gcc-lib`

contains executables and include files for the GNU C compiler, `gcc(1)`.

`/usr/lib/groff`

Files for the GNU groff document formatting system.

`/usr/lib/uucp`

Files for `uucp(1)`.

`/usr/local`

This is where programs which are local to the site typically go.

`/usr/local/bin`

Binaries for programs local to the site.

`/usr/local/doc`

Local documentation.

`/usr/local/etc`

Configuration files associated with locally installed programs.

`/usr/local/games`

Binaries for locally installed games.

`/usr/local/lib`

Files associated with locally installed programs.

`/usr/local/include`

Header files for the local C compiler.

`/usr/local/info`

Info pages associated with locally installed programs.

`/usr/local/man`

Man pages associated with locally installed programs.

`/usr/local/sbin`

Locally installed programs for system administration.

`/usr/local/share`

Local application data that can be shared among different architectures of the same OS.

`/usr/local/src`

Source code for locally installed software.

`/usr/man`

Replaced by `/usr/share/man`.

`/usr/sbin`

This directory contains program binaries for system administration which are not essential for the boot process, for mounting `/usr`, or for system repair.

`/usr/share`

This directory contains subdirectories with specific application data, that can be shared among different architectures of the same OS. Often one finds stuff here that used to live in `/usr/doc` or `/usr/lib` or `/usr/man`.

`/usr/share/dict`

Contains the word lists used by spell checkers.

`/usr/share/doc`

Documentation about installed programs.

`/usr/share/games`

Static data files for games in `/usr/games`.

`/usr/share/info`

Info pages go here.

`/usr/share/locale`

Locale information goes here.

`/usr/share/man`

Manual pages go here in subdirectories according to the man page sections.

`/usr/share/man/<locale>/man[1-9]`

These directories contain manual pages for the specific locale in source code form. Systems which use a unique language and code

set for all manual pages may omit the `<locale>` substring.

`/usr/share/misc`

Miscellaneous data that can be shared among different architectures of the same OS.

`/usr/share/nls`

The message catalogs for native language support go here.

`/usr/share/sgml`

Files for SGML and XML.

`/usr/share/terminfo`

The database for terminfo.

`/usr/share/tmac`

Troff macros that are not distributed with groff.

`/usr/share/zoneinfo`

Files for timezone information.

`/usr/src`

Source files for different parts of the system, included with some packages for reference purposes. Don't work here with your own

projects, as files below `/usr` should be read-only except when installing software.

`/usr/src/linux`

This was the traditional place for the kernel source. Some distributions put here the source for the default kernel they

ship.

You should probably use another directory when building your own kernel.

/usr/tmp

Obsolete. This should be a link to /var/tmp. This link is present only for compatibility reasons and shouldn't be used.

/var This directory contains files which may change in size, such as spool and log files.

/var/adm

This directory is superseded by /var/log and should be a symbolic link to /var/log.

/var/backups

Reserved for historical reasons.

/var/cache

Data cached for programs.

/var/catman/cat[1-9] or /var/cache/man/cat[1-9]

These directories contain preformatted manual pages according to their man page section. (The use of preformatted manual pages is deprecated.)

/var/cron

Reserved for historical reasons.

/var/lib

Variable state information for programs.

/var/local

Variable data for /usr/local.

/var/lock

Lock files are placed in this directory. The naming convention for device lock files is LCK.<device> where <device> is the device's name in the filesystem. The format used is that of HDU UUCP lock files, that is, lock files contain a PID as a 10-byte

ASCII decimal number, followed by a newline character.

/var/log

Miscellaneous log files.

/var/opt

Variable data for /opt.

/var/mail

Users' mailboxes. Replaces /var/spool/mail.

/var/msg

Reserved for historical reasons.

/var/preserve

Reserved for historical reasons.

/var/run

Run-time variable files, like files holding process identifiers (PIDs) and logged user information (utmp). Files in this directory are usually cleared when the system boots.

/var/spool

Spooled (or queued) files for various programs.

/var/spool/at

Spooled jobs for at(1).

/var/spool/cron

Spooled jobs for cron(8).

/var/spool/lpd

Spooled files for printing.

/var/spool/mail

Replaced by /var/mail.

/var/spool/mqueue

Queued outgoing mail.

/var/spool/news

Spool directory for news.

/var/spool/rwho

Spooled files for rwhod(8).

/var/spool/smial

Spooled files for the smial(1) mail delivery program.

/var/spool/uucp

Spooled files for uucp(1).

/var/tmp

Like /tmp, this directory holds temporary files stored for an unspecified duration.

/var/yp

Database files for NIS.