

RetroPie auf x86-Basis (USB-Stick —▶ Festplatte)

Von der Original-Webseite <https://retropie.org.uk/> :

Welcome to RetroPie. RetroPie allows you to turn your Raspberry Pi, ODroid C1/C2, or PC into a retro-gaming machine. It builds upon Raspbian, EmulationStation, RetroArch and many other projects to enable you to play your favourite Arcade, home-console, and classic PC games with the minimum set-up. For power users it also provides a large variety of configuration tools to customise the system as you want.

RetroPie sits on top of a full OS, you can install it on an existing Raspbian, or start with the RetroPie image and add additional software later. It's up to you.

Der Anstoss zu diesem Projekt liegt schon im ersten Satz: Raspberry, Odroid *or* PC. Fertige Images zu RetroPie gibt es in erster Linie aber dennoch nur für die Raspberrys; wer ein anderes System nutzt und Retro-Games zocken will, muss sich selber daran setzen.

Gut, ein Ubuntu/Debian installieren ist heutzutage keine Schwierigkeit mehr, aber welche Distribution nehmen? Welche Version? Welchen Fork nehmen...Ubuntu, Xubuntu, Lubuntu, Kubuntu? Wie? Was? Eben. Eine für RetroPie empfohlene Distribution sollte es ja sein, aber schön klein und einfach Handzuhaben, ohne Schnickschnack etc. Denn was will man mit Browsern und ganzen Office-Paketen auf einer Zock-Station?

Deshalb zeigen wir hier, wie man ohne Gefummel (na gut, von Schritt 1 und 2 einmal abgesehen) relativ schnell ein einfaches, offenes und legales und vor allem leistungsstarkes Betriebssystem aufbaut und mit nahezu zahllosen Emulatoren versieht.

Außerdem soll diese Anleitung zeigen und Denkanstöße geben, wie und mit welchen Werkzeugen man aus eigenen Ideen ein „eigenes“ Betriebssystem erstellen kann, welches ganz bestimmten Einsatzzwecken dient.

Gutes Beispiele, was aus solchen Ideen werden kann, zeigen beispielsweise die Projekte <https://kodi.tv/> und <https://libreelec.tv/>, und selbstverständlich <https://lede-project.org/>, sowie <http://ipxe.org/> und <http://www.syslinux.org>, um nur wenige und herausragende zu nennen.

Für diese Projekt hier wurden nur wenige Hilfsmittel verwendet: Der Bootloader SYSLINUX, ein minimalistisches Ubuntu, ein paar Quellen von RetroPie selber, ein von dritten stammender Boot-Splash-Screen und etwas Hirnschmalz – schon ist ein selbstgebauter RetroPie auf x86-Basis fertig!

Schritt 1 – Den USB-Stick vorbereiten (kritisch)

Der USB-Stick muss eine Größe von mindestens 2GB haben. Das Image von <https://www.com-in.de/retropie/retropie.img> herunterladen und auf den Stick transferieren:

Unter Linux geht das per dd, beispielsweise `[sudo] dd if=/PFAD/retropie.img of=/dev/sdX bs=1M` (als `sdX` ist das korrekte Gerät anzugeben)

Window-User benutzen Rufus oder einem anderen Imager (ungetestet).

Beim Hantieren mit den Laufwerken und Partitionen aufpassen, dass man das richtige Laufwerk nimmt! Schlimmstenfalls sind sonst gerade mal alle Daten des (falsch ausgewählten) Laufwerks futsch!

Schritt 2 – Vom USB-Stick starten (einfach-mittel)

Dank SYSLINUX kann vom Stick sowohl im alten BIOS-, wie auch im UEFI-BIOS-Modus gestartet werden:



```
Starting Load/Save Random Seed...
Starting Create System Users...
Starting Flush Journal to Persistent Storage...
[ OK ] Started Load/Save Random Seed.
[ OK ] Started Flush Journal to Persistent Storage.
[ OK ] Started Create System Users.
Starting Create Static Device Nodes in /dev...
[ OK ] Started udev Coldplug all Devices.
[ OK ] Started Create Static Device Nodes in /dev.
[ OK ] Reached target Local File Systems (Pre).
[ OK ] Reached target Local File Systems.
Starting Create Volatile Files and Directories...
Starting Rebuild Dynamic Linker Cache...
Starting Rebuild Journal Catalog...
Mounting Temporary /etc/pacman.d/gnupg directory...
Starting udev Kernel Device Manager...
[ OK ] Started RPC Bind.
[ OK ] Started Create Volatile Files and Directories.
Starting Update UTMP about System Boot/Shutdown...
[ OK ] Mounted Temporary /etc/pacman.d/gnupg directory.
[ OK ] Started Rebuild Journal Catalog.
[ OK ] Started Update UTMP about System Boot/Shutdown.
[ OK ] Started udev Kernel Device Manager.
[ OK ] Created slice system-dhcpd.slice.
[ OK ] Started Rebuild Dynamic Linker Cache.
Starting Update is Completed...
[ OK ] Started Update is Completed.
[ OK ] Reached target System Initialization.
[ OK ] Started Daily Cleanup of Temporary Directories.
[ OK ] Started Daily verification of password and group files.
[ OK ] Started Daily man-db cache update.
[ OK ] Started Daily rotation of log files.
[ OK ] Reached target Timers.
[ OK ] Listening on 0-Bus System Message Bus Socket.
[ OK ] Reached target Sockets.
[ OK ] Reached target Basic System.
Starting SSH Key Generation...
Starting Login Service...
[ OK ] Started Entropy Harvesting Daemon.
Starting Initializes Pacman keyring...
Starting dhcpd on eth0...
[ OK ] Started 0-Bus System Message Bus.
[ OK ] Started Login Service.
[ OK ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
```

Das Arch-Linux zeigt seinen Startvorgang detailliert an. Falls der PC nämlich zickt, erhalten wir hier ggf. nützliche Hinweise darauf, wo der Schuh drückt.

Gab es keinen Ärger, begrüßt uns ein selbstlaufendes Skript, das es gleich losgehen kann. Hier finden wir auch ein paar nützliche Informationen zu unserem System, wie beispielsweise die dem PC zugewiesene IP-Adresse:

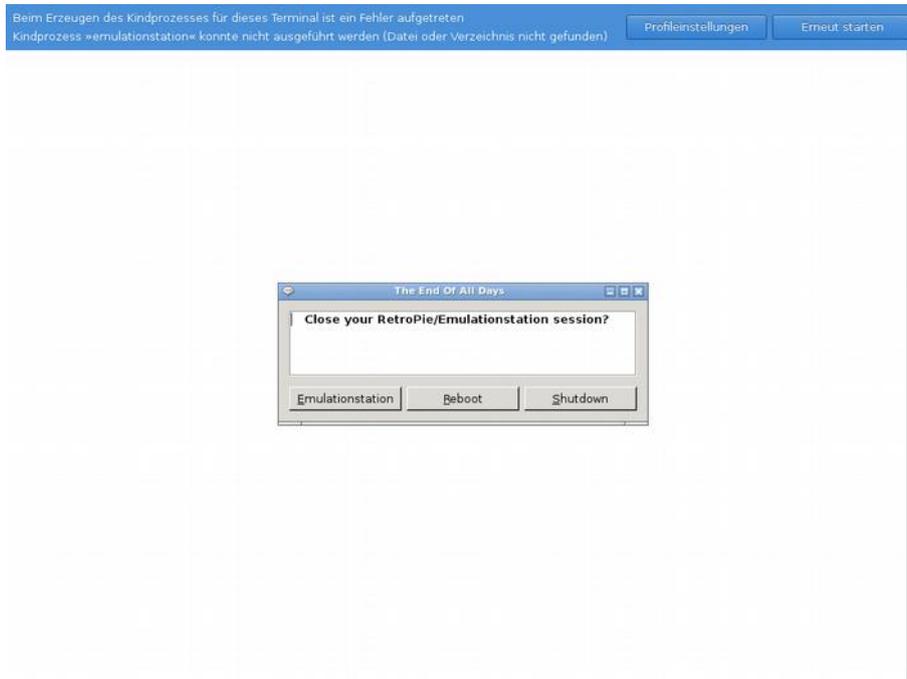
```
*** Mittwoch, 6 Dezember 2017, 04:26:57
**** Linux 4.14.3-1-RRCH x86_64 GNU/Linux
***
|_| Dateisystem Größe Benutzt Verf. Verw4 Eingehängt auf
|_| airoofs 256M 296K 256M 1% /
..|_|.. Uptime..... 0 days, 0h0m43s
*** x *** Memory..... 7286388K (Free) / 8176852K (Total)
***** Running Processes... 99
***** IP Address..... 192.168.0.86
***** Temperature..... CPU: 10C/F
***** The RetroPie Project, https://retropie.org.uk

Zum Ausführen des Skripts JA eingeben und bestätigen. Jede andere Eingabe führt zum Abbruch.
Sollen wir loslegen?
```

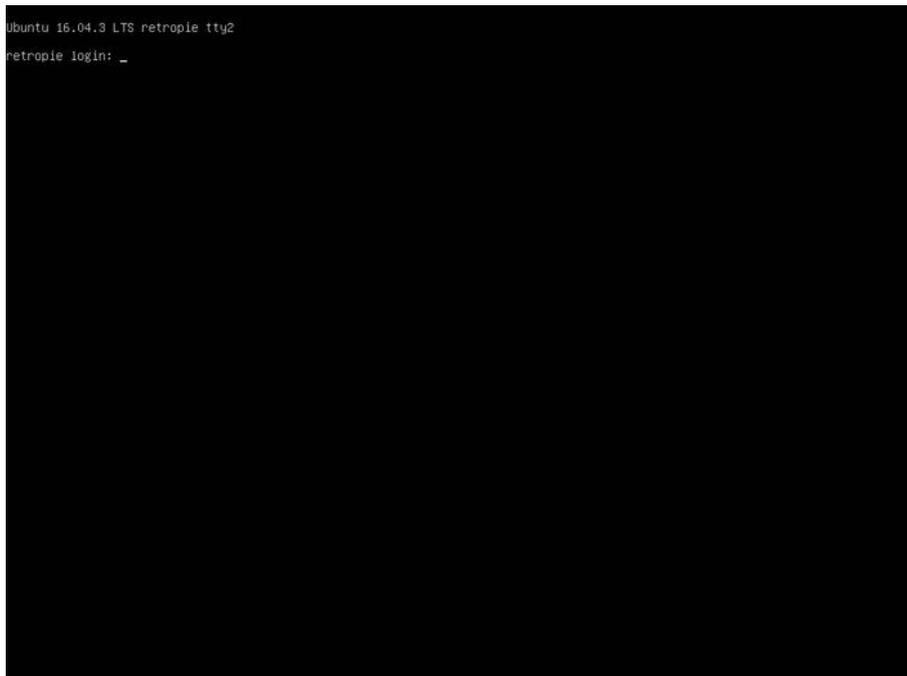
Das Skript fordert die Eingabe von **JA** (groß geschrieben!), alles andere wird als Abbruch gewertet. **Achtung: Nach der Eingabe von JA gibt es kein zurück mehr und die erste gefundene Festplatte wird vollständig gelöscht und neu partitioniert!**

Schritt 3 – Setup RetroPie (einfach)

Nach dem Neustart sehen wir eine Boot-Animation (hier aber nicht abgebildet) und werden gleich im Anschluss von einer Fehlermeldung begrüßt. Das ist zwar unschön, aber beabsichtigt (vielmehr Bequemlichkeit meinerseits, da noch keine elegante Alternative geschaffen), denn es fehlt ja noch der ganze RetroPie-Teil.



Um diesen RetroPie-Teil zu installieren, drücken wir kurzerhand die Tastenkombination STRG+ALT+F2 und landen auf dem Anmeldebildschirm der Konsole:



Hier melden wir uns mit dem Benutzernamen **pi** und Passwort **pi** am Betriebssystem an...

```
Ubuntu 16.04.3 LTS retroPie tty2
retroPie login: pi
Password:
Last login: Wed Dec 6 16:44:41 CET 2017 on tty1
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.4.0-101-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

.***.      Mittwoch, 6 Dezember 2017, 05:02:18
****.      Linux 4.4.0-101-generic x86_64 GNU/Linux
****.
|*|         Dateisystem   Größe Benutzt Verf. Verw% Eingehängt auf
|*|         /dev/sda2     7,9G  2,6G  4,9G  35% /
..|*|..     Uptime.....: 0 days, 00h17m55s
****. * ****. Memory.....: 7790492kB (Free) / 8174848kB (Total)
*****00**** Running Processes.: 150
*****00**** IP Address.....: 192.168.0.88
*****      Temperature.....: CPU: C/F
*****      The RetroPie Project, https://retroPie.org.uk

pi@retroPie:~$ _
```

...und wechseln in das Verzeichnis des RetroPie-Setups per **cd RetroPie-Setup** (Groß-/Kleinschreibung beachten!):

```
Ubuntu 16.04.3 LTS retroPie tty2
retroPie login: pi
Password:
Last login: Wed Dec 6 16:44:41 CET 2017 on tty1
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.4.0-101-generic x86_64)

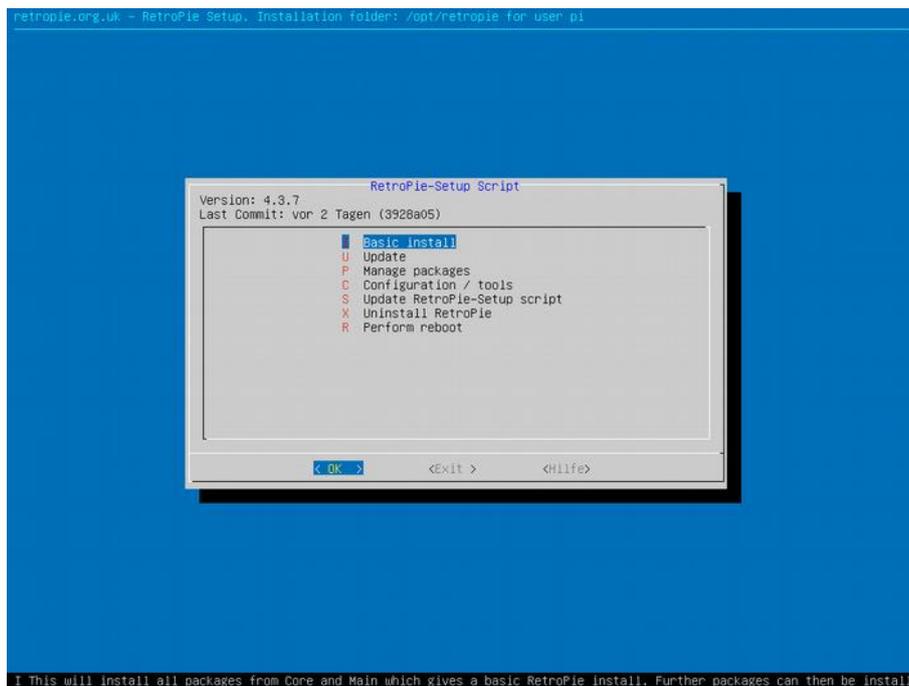
 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

.***.      Mittwoch, 6 Dezember 2017, 05:02:18
****.      Linux 4.4.0-101-generic x86_64 GNU/Linux
****.
|*|         Dateisystem   Größe Benutzt Verf. Verw% Eingehängt auf
|*|         /dev/sda2     7,9G  2,6G  4,9G  35% /
..|*|..     Uptime.....: 0 days, 00h17m55s
****. * ****. Memory.....: 7790492kB (Free) / 8174848kB (Total)
*****00**** Running Processes.: 150
*****00**** IP Address.....: 192.168.0.88
*****      Temperature.....: CPU: C/F
*****      The RetroPie Project, https://retroPie.org.uk

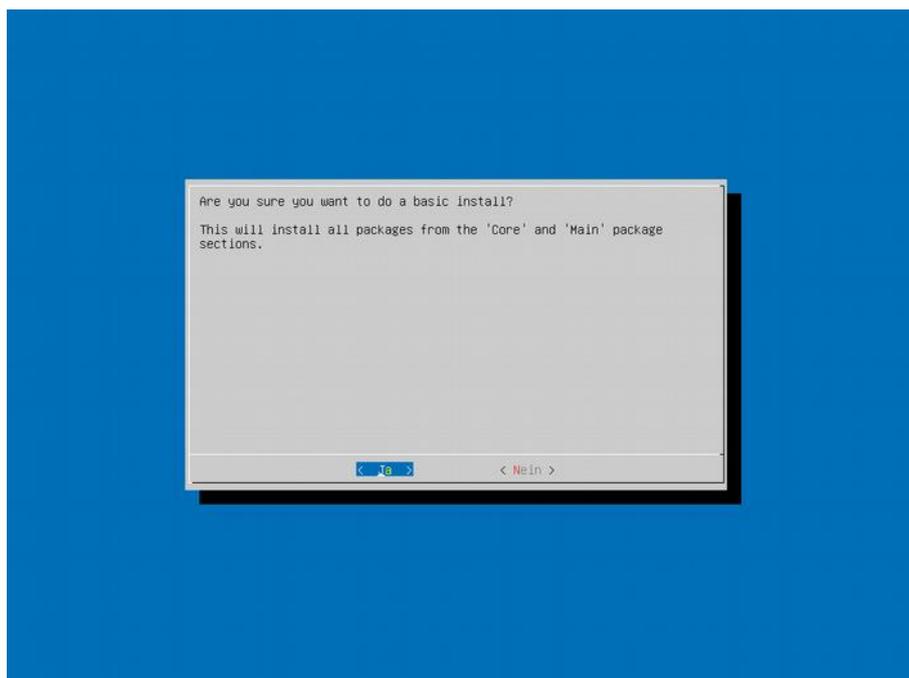
pi@retroPie:~$ cd RetroPie-Setup
pi@retroPie:~/RetroPie-Setup$
```


Schritt 4 – Installation RetroPie (einfach)

Nun wird es spannend, denn nun kommt die eigentliche Software des RetroPie-Systems dran, infolge dessen weitere Software installiert werden wird. Zunächst drücken die Taste Enter und wählen im erschienenen Menü *Basic Install...*



und wählen *Ja* aus, um die Installation zu starten:



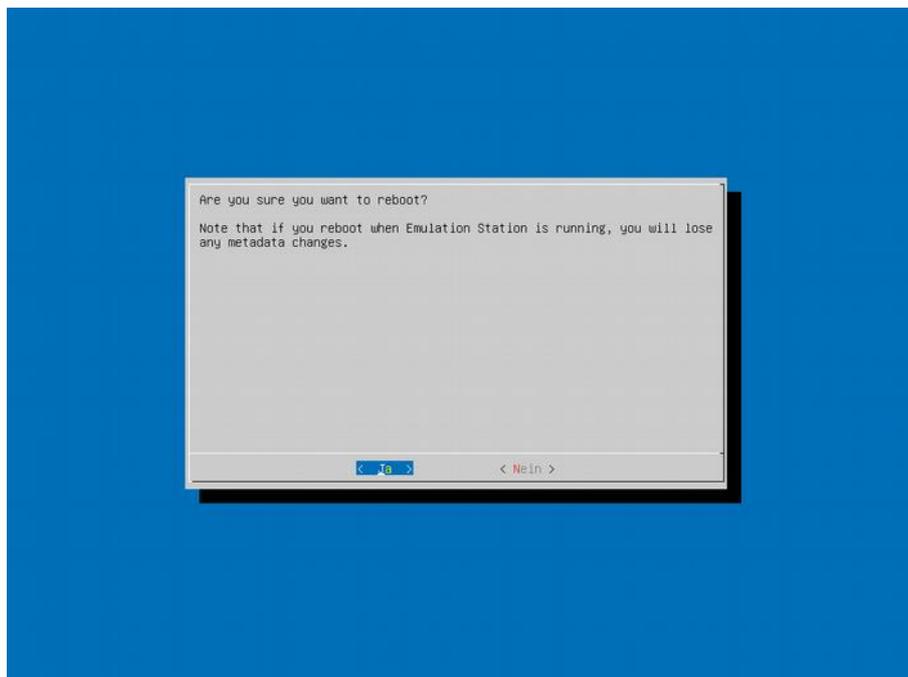
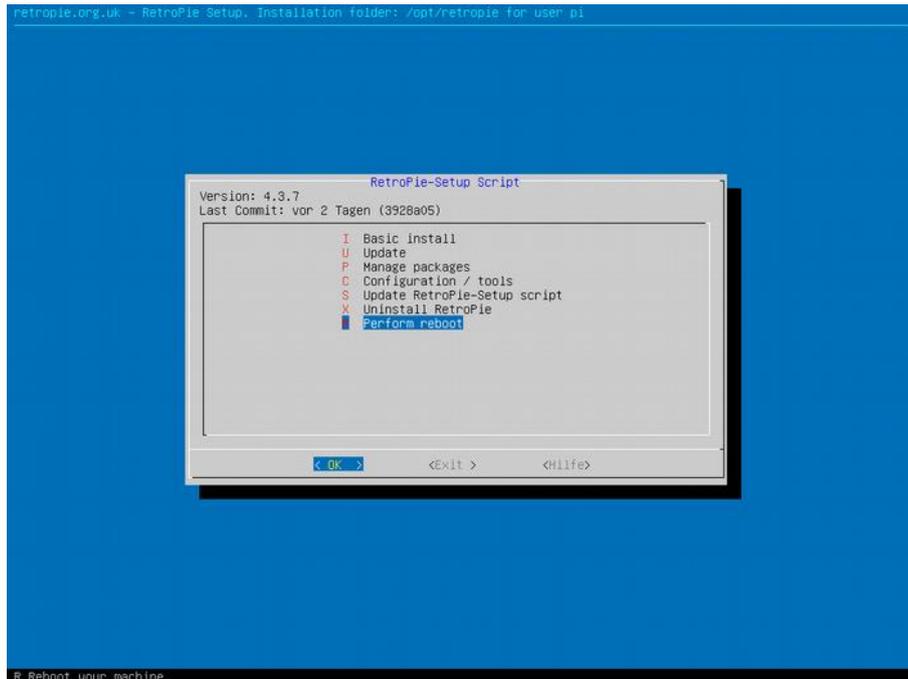
Wir sehen daraufhin ein uns vertrautes Bild und denken uns, dass das nun wirklich etwas dauern kann - es werden dieses Mal nicht einfach nur einige fertige Software-Pakete aus dem Internet geladen, nein, es werden die ganzen Quelltexte der (Libretro-) Emulatoren geladen und kompiliert. Je nach PC-Power kann das durchaus eine ganze Weile dauern. Auf meiner virtuellen Maschine der Testumgebung mit 4-Kern CPU und 8GB RAM dauerte das ganze Rund 30 Minuten. Also suchen wir uns derweil eine andere Beschäftigung, wie Katze kraulen, Staubsaugen oder Spazierengehen ☺

```
cvs-buildpackage diffoscope devscripts-el dose-extra gnuplot
libauthn-sasl-perl libfile-desktopentry-perl libnet-smtp-ssl-perl
libterm-size-perl libyaml-syck-perl mozilla-devscripts mutt svn-buildpackage
w3m debian-keyring equivs liblzo-protocol-https-perl libsoap-lite-perl fcitx
gettext-doc libasound2-doc nas libglb2.0-doc libice-doc
libqt4-declarative-folderlistmodel libqt4-declarative-gestures
libqt4-declarative-particles libqt4-declarative-shaders qt4-qmlviewer
libqt4-dev qt4-qtconfig libsm-doc libtool-doc gfortran | fortran95-compiler
gcj-jdk libxcb-doc libxext-doc libxt-doc libmail-box-perl
Empfohlene Pakete:
at oct1-tools dput | dupload libdistro-info-perl libencode-locale-perl
liburi-perl libwww-perl lintian patchutils python3-debian python3-magic
wdiff fcitx-data fcitx-modules libasprintf-dev libgettextpo-dev
libqt4-sql-mysql | libqt4-sql-odbc | libqt4-sql-psql | libqt4-sql-sqlite
qt-at-sql libltdl-dev libx11-doc libmail-sendmail-perl
Die folgenden NEUEN Pakete werden installiert:
autoconf automake autopoint autotools-dev debhelper devscripts dh-autoreconf
dh-strip-nondeterminism fcitx-bin fcitx-libs-dev gettext gir1.2-ibus-1.0
intltool-debian libarchive-zip-perl libasound2-dev libaudio2 libdbus-1-dev
libdrm-dev libfcitx-config4 libfcitx-core0 libfcitx-gclient0 libfcitx-qt0
libfcitx-utilso libffi6-stripnondeterminism-perl libgettextpo0
libgl1-mesa-dev libglb2.0-bin libglb2.0-dev libibus-1.0-5 libibus-1.0-dev
libice-dev libmng2 libpcre16-3 libpcre3-dev libpcre32-3 libpcrecpp0v5
libpthread-stubs0-dev libpulse-dev libqt4-dbus libqt4-declarative
libqt4-network libqt4-script libqt4-sql libqt4-xml libqt4-xmllpatterns
libqtcore4 libqtdbus4 libqtgui4 libsigsegv2 libsm-dev libtimedate-perl
libtool libudev-dev libunistring0 libx11-dev libx11-xcb-dev libxau-dev
libxcb-dri2-0-dev libxcb-dri3-dev libxcb-glx0-dev libxcb-present-dev
libxcb-randr0 libxcb-randr0-dev libxcb-render0-dev libxcb-shape0-dev
libxcb-sync-dev libxcb-xfixes0-dev libxcb1-dev libxcursor-dev libxdamage-dev
libxmdcp-dev libxext-dev libxfixes-dev libxi-dev libxinerama-dev
libxrandr-dev libxrender-dev libxshmfence-dev libxss-dev libxt-dev
libxxf86vm-dev m4 mesa-common-dev po-debconf qdbus qtchooser qtcore4-l10n
x11proto-core-dev x11proto-damage-dev x11proto-dri2-dev x11proto-fixes-dev
x11proto-g1-dev x11proto-input-dev x11proto-kb-dev x11proto-randr-dev
x11proto-render-dev x11proto-scrnsaver-dev x11proto-xext-dev
x11proto-xf86vidmode-dev x11proto-xinerama-dev xorg-sgml-doctools xtrans-dev
zlib1g-dev
0 aktualisiert, 103 neu installiert, 0 zu entfernen und 0 nicht aktualisiert.
Es müssen 22,7 MB an Archiven heruntergeladen werden.
Nach dieser Operation werden 94,5 MB Plattenplatz zusätzlich benutzt.
Holen:1 http://de.archive.ubuntu.com/ubuntu xenial/main amd64 libaudio2 amd64 1.9.4-4 [50,1 kB]
Holen:2 http://de.archive.ubuntu.com/ubuntu xenial/main amd64 libmng2 amd64 2.0.2-0ubuntu3 [169 kB]
Holen:3 http://de.archive.ubuntu.com/ubuntu xenial/main amd64 libpcrecpp0v5 amd64 2:8.38-3.1 [15,2 kB]
Holen:4 http://de.archive.ubuntu.com/ubuntu xenial/main amd64 libsigsegv2 amd64 2.10-4 [14,1 kB]
Holen:5 http://de.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libudev-dev amd64 2:3-4ubuntu21 [150 kB]
Holen:6 http://de.archive.ubuntu.com/ubuntu xenial/main amd64 libunistring0 amd64 0.9.3-5.2ubuntu1 [279 kB]
```

Nach Abschluss der ganzen Einrichtung gelangen wir wieder in das Hauptmenü des RetroPie-Setups und können, wenn wir wollen, noch weitere Emulatoren installieren und/oder verschiedene Konfigurationen vornehmen. Das und vieles andere mehr sind aber Themen, die weiter und ausführlich in der Original-Dokumentation unter <https://retropie.org.uk/docs/> zu finden sind.

Schritt 5 – Abschluss (zu einfach)

Nun noch flott ein Neustart des PC vorgenommen und man kann beginnen, das System mit BIOSen und ROMs zu bestücken – um dann endlich loszocken zu können! Viel Spaß dabei!



Quellennachweis

<http://www.syslinux.org>

für den Bootloader

<https://www.archlinux.org/>

als Minimalsystem, welches direkt vom USB-Stick gebootet wird

<https://retropie.org.uk/>

dem wir das eigentliche Retro-System verdanken

<https://github.com/HerbFargus/plymouth-themes.git>

für die wunderschöne Animation des Boot-Splash-Screens