

Installation RetroPie auf Raspberry Pi

Evil

m.stroh@softhome.net

<http://evil.hn.vc>

23. September 2013

Inhaltsverzeichnis

1	Vorwort	2
2	Information	4
3	Installation	4
4	Wie kommt man zu den Spielen für die Emulatoren	5
4.1	Original Module auslesen	6
4.2	Gebraucht CD	6
4.3	Extrahieren von Module-Images aus einer Sega Game Collection	6
4.4	Freies Spiel „Project MD“ (von Sik) für Sega Mega Drive / Genesis	6
5	Installation Spiele (ROM-Image/CD-Image)	6
6	Parametrierung	7
6.1	Allgemein	7
6.1.1	Joypad Konfiguration	7
6.1.2	RetroPie (Emulationstation)	7
6.1.3	Splash screen	8
6.1.4	System Overclocking	8
6.2	RetroArch - Sony PS1/PSX	9
6.3	RetroArch - Nintendo SNES	10
6.4	DGen - Sega Mega Drive / Genesis)	10

1 Vorwort

RetroPie ist eine Betriebssystem (Raspian Basis) das speziell auf die Anwendung als Retro Spiel Konsole ausgelegt ist. RetroPie enthält einige Emulatoren und eine passende Menüsteuerung dazu.

Folgende Emulatoren bzw. Spiele werden unterstützt:

- Apple][
- Atari 2600
- Basilisk II
- Cave Story
- C64
- Doom
- Duke Nukem 3D
- Game Boy
- Game Boy Advance
- Game Boy Color
- Sega Game Gear
- Intellivisio
- MAME
- FinalBurn Alpha
- PC (x86)
- ScummVM
- Sega Master System II
- Sega Mega Drive / Genesis
- NeoGeo
- Nintendo Entertainment System (NES)
- PC Engine/TurboGrafx 16
- Sony Playstation 1 (PSX/PS1)
- Super Nintendo (SNES)

- ZX Spectrum

Unterschied zwischen einigen Spiel Portierung auf Basis von Sega Spielen werden in diesem Buch behandelt:

Sega Arcade Classics Vol. 1 (Hardcore Gaming 101): <http://www.hardcoregaming101.net/segabook.htm>

Ich möchte hauptsächlich auf die Emulatoren „Sony Playstation 1“, „Super Nintendo (SNES)“ und „Sega Mega Drive / Genesis“ eingehen.

„Sony Playstation 1“ bietet aus einer Sicht die beste Hardware und liegt knapp an der Grenze dessen was die Hardware ermöglicht. „Super Nintendo“ und „Sega Mega Drive / Genesis“ sind aus „Retro“ Sicht für mich interessant.

Für mich hat auch der Aspekt Verfügbarkeit von original Kopien der Spiele eine Rolle gespielt. „Sony Playstation 1“ Spiele gibt es am Gebrauchtmart und können mit ein CD-Laufwerk ausgelesen werden. „Sega Mega Drive / Genesis“ verwendet zwar Module. Viele Spiele wurde aber auch als Collections für den PC herausgebracht und sind somit teilweise aktuell erwerbbar.

Nützliche Links:

ScummVM: <http://www.scummvm.org/>

2 Information

Projektseite: <http://blog.petrockblock.com/retropie/>

Download: <http://blog.petrockblock.com/retropie/retropie-downloads/>

Min. Anforderung SD-Speicher: 4 GB

Empf. Anforderung SD-Speicher (Bei PS1 Spielen): 32 GB

USB Cartridge Reader: <https://www.retrode.com/>

Sega Arcade Classics Vol. 1 (Hardcore Gaming 101): <http://www.hardcoregaming101.net/segabook.htm>

3 Installation

```
wget --trust-server-names http://blog.petrockblock.com/?wpdmdl=17
unzip RetroPieImage_v1.7.zip
```

```
dmesg | tail -n 6
```

```
[335694.196898] sd 3:0:0:0: [sdc] 63404032 512-byte logical blocks: (32.4 GB/30.2 GiB)
[335694.198641] sd 3:0:0:0: [sdc] No Caching mode page present
[335694.198946] sd 3:0:0:0: [sdc] Assuming drive cache: write through
[335694.206524] sd 3:0:0:0: [sdc] No Caching mode page present
[335694.206829] sd 3:0:0:0: [sdc] Assuming drive cache: write through
[335694.398175] sdc: sdc1
```

```
dd if=RetroPieImage_v1.7.img of=/dev/sdc
```

Nun kann die SD-Karte in die RASpberry Pi eingesteckt werden und gestartet werden. Zuerst muss man die Eingabegeräte (Joypads, Tastatur) konfigurieren. Diese Konfiguration werden wir aber später noch manuell anpassen.

SSH Login:

User: pi

Password: raspberry

```
sudo apt-get clean
sudo fdisk -l /dev/mmcblk0
```

```
Disk /dev/mmcblk0: 32.5 GB, 32462864384 bytes
4 heads, 16 sectors/track, 990688 cylinders, total 63404032 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000c7b31
```

Device	Boot	Start	End	Blocks	Id	System
/dev/mmcblk0p1		8192	122879	57344	c	W95 FAT32 (LBA)
/dev/mmcblk0p2		122880	6676479	3276800	83	Linux

```
sudo fdisk /dev/mmcblk0
```

```
Command (m for help): n
```

```
Select (default p): p
```

```
Partition number (1-4, default 3): 3
```

```
First sector (2048-63404031, default 2048): 6676480
```

```
Last sector, +sectors or +sizeK,M,G (6676480-63404031, default 63404031): <Enter>
```

```
Command (m for help): p
```

```
Disk /dev/mmcblk0: 32.5 GB, 32462864384 bytes
```

```
4 heads, 16 sectors/track, 990688 cylinders, total 63404032 sectors
```

```
Units = sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disk identifier: 0x000c7b31
```

Device	Boot	Start	End	Blocks	Id	System
/dev/mmcblk0p1		8192	122879	57344	c	W95 FAT32 (LBA)
/dev/mmcblk0p2		122880	6676479	3276800	83	Linux
/dev/mmcblk0p3		6676480	63404031	28363776	83	Linux

```
Select (default p): w
```

```
sudo reboot
```

```
sudo mkfs.ext4 /dev/mmcblk0p3
```

```
sudo mount /dev/mmcblk0p3 /mnt/
```

```
sudo cp -rv /home/pi/RetroPie/roms/* /mnt/
```

```
sudo umount /mnt
```

```
sudo mount /dev/mmcblk0p3 /home/pi/RetroPie/roms
```

```
sudo chown pi:pi /home/pi/RetroPie/roms
```

```
sudo chown pi:pi /home/pi/RetroPie/roms/*
```

```
sudo umount /home/pi/RetroPie/roms
```

/etc/fstab

proc	/proc	proc	defaults	0	0
/dev/mmcblk0p1	/boot	vfat	defaults	0	2
/dev/mmcblk0p2	/	ext4	defaults,noatime	0	1
/dev/mmcblk0p3	/home/pi/RetroPie/roms	ext4	defaults,noatime	0	3

```
sudo umount -a
```

4 Wie kommt man zu den Spielen für die Emulatoren

Im Internet besteht die Möglichkeit Spiele-Images von Seiten wie FreeROMS.com, CoolROM.com usw. herunter zu laden, dies ist allerdings illegal.

Es gibt aber auch Möglichkeiten legal zu den Spiele-Images zu kommen.

4.1 Original Module auslesen

Nintendo SNES und Sega Mega Drive / Genesis verwenden Module als Quelle für die Spiele. Diese können mit dem Adapter „Retrode“ ausgelesen werden. Weitere infos erhält man auf <https://www.retrode.com/>

4.2 Gebraucht CD

Für die Playstation (PS1 oder PSX) können einfach alte Spiele CD gekauft und mit z.B. ImgBurn mit einem CD-Laufwerk in eine Datei ausgelesen werden. Dieses CD-Image (bin- und cue-Dateien) kann im Emulator geladen werden. Für Sega Mega Drive hat es zwar auch ein Mega CD als Zusatz zur Konsole gegeben. Leider kann der Emulator DGen aber keine Mega CD Spiele emulieren. Somit können Mega CD Spiele nicht genutzt werden!

4.3 Extrahieren von Module-Images aus einer Sega Game Collection

Sega bietet mehrere Game Collection des Mega Drive / Genesis für den PC an. Diese Spiele enthalten die originalen Modul-Images der Spiele. Diese Modul-Images können aus dem geladenen Spiel in Windows extrahiert werden. Besonders empfiehlt sich dafür die Sega Mega Drive Classic Collection Gold Edition (siehe http://en.wikipedia.org/wiki/Sega_Mega_Drive_Classic_Collection). Sie enthält 46 Mega Drive Titel (Sammlung von 4 Einzell CDs) und ist relativ kostengünstig zu haben.

Wie man zu den ROM-Images kommt kann man den folgenden Seiten bzw. Links entnehmen.
Basisanleitung PC Collection: <http://forums.sonicrotro.org/index.php?showtopic=13514&st=0>
Basisanleitung Dreamcast Collection: <http://www.racketboy.com/forum/viewtopic.php?f=52&t=30647>
Memory Dump Tool: <http://technet.microsoft.com/en-us/sysinternals/dd996900.aspx>
Memory Viewer/Hex Editor: <http://mh-nexus.de/de/hxd/>
Beschreibung ROM Header: http://en.wikibooks.org/wiki/Genesis_Programming

4.4 Freies Spiel „Project MD“ (von Sik) für Sega Mega Drive / Genesis

Ja, es gibt sogar noch jemanden der in seiner Freizeit ein Spiel für das alte Sega Mega Drive / Genesis programmiert hat. Das Spiel nennt sich „Project MD“ und ist als ROM Image für Emulatoren verfügbar.
„Project MD“: <http://projectmd.mdscene.net>

5 Installation Spiele (ROM-Image/CD-Image)

```
cd /home/pi/RetroPie/roms/megadrive
wget http://projectmd.mdscene.net/download/projmd-20120429.zip
unzip projmd-20120429.zip
mv projmd-20120429/projmd-20120429.bin ./projmd.bin
rm -r projmd-20120429 projmd-20120429.zip

cd /home/pi/RetroPie/roms/snes

cd /home/pi/RetroPie/roms/psx
```

6 Parametrierung

6.1 Allgemein

6.1.1 Joypad Konfiguration

Beim Logitech RumblePad 2 darf die Mode Taste nicht gedrückt werden, damit alle Funktionen (auch analog) des Joypad funktionieren.

Beim GreenAsia Inc. USB Joystick muss die Analog Taste gedrückt werden, damit alle Funktionen (auch analog) des Joypad funktionieren.

```
jstest /dev/input/js0
jstest /dev/input/js1
```

```
udevadm info -q all -n /dev/input/js0 | egrep "ID_VENDOR|ID_MODEL"
```

```
E: ID_MODEL=Logitech_RumblePad_2_USB
E: ID_MODEL_ENC=Logitech\x20RumblePad\x202\x20USB
E: ID_MODEL_ID=c218
E: ID_VENDOR=Logitech
E: ID_VENDOR_ENC=Logitech
E: ID_VENDOR_ID=046d
```

```
udevadm info -q all -n /dev/input/js1 | egrep "ID_VENDOR|ID_MODEL"
```

```
E: ID_MODEL=USB_Joystick
E: ID_MODEL_ENC=USB\x20Joystick\x20\x20\x20\x20\x20
E: ID_MODEL_ID=0003
E: ID_VENDOR=GreenAsia_Inc.
E: ID_VENDOR_ENC=GreenAsia\x20Inc.\x20\x20\x20
E: ID_VENDOR_ID=0e8f
```

/etc/udev/rules.d/73-persistent-joystick.rules

```
KERNEL=="js?", ENV{ID_VENDOR_ID}=="046d", ENV{ID_MODEL_ID}=="c218", NAME="input/js1"
KERNEL=="js?", ENV{ID_VENDOR_ID}=="0e8f", ENV{ID_MODEL_ID}=="0003", NAME="input/js0"
```

6.1.2 RetroPie (Emulationstation)

/home/pi/.emulationstation/es_input.cfg [-rw-r--]

```
<?xml version="1.0"?>
<inputList>
  <inputConfig type="keyboard">
    <input name="a" type="key" id="13" value="1" />
    <input name="b" type="key" id="8" value="1" />
    <input name="down" type="key" id="274" value="1" />
    <input name="left" type="key" id="276" value="1" />
    <input name="menu" type="key" id="109" value="1" />
    <input name="pagedown" type="key" id="281" value="1" />
    <input name="pageup" type="key" id="280" value="1" />
    <input name="right" type="key" id="275" value="1" />
    <input name="select" type="key" id="106" value="1" />
    <input name="up" type="key" id="273" value="1" />
  </inputConfig>
  <inputConfig type="joystick" deviceName="Logitech Logitech RumblePad 2 USB">
    <input name="a" type="button" id="0" value="1" />
  </inputConfig>
</inputList>
```

```

        <input name="b" type="button" id="2" value="1" />
        <input name="down" type="hat" id="0" value="4" />
        <input name="left" type="hat" id="0" value="8" />
        <input name="menu" type="button" id="9" value="1" />
        <input name="pagedown" type="button" id="4" value="1" />
        <input name="pageup" type="button" id="5" value="1" />
        <input name="right" type="hat" id="0" value="2" />
        <input name="select" type="button" id="3" value="1" />
        <input name="up" type="hat" id="0" value="1" />
    </inputConfig>
    <inputConfig type="joystick" deviceName="GreenAsia Inc.    USB Joystick    ">
        <input name="a" type="button" id="0" value="1" />
        <input name="b" type="button" id="3" value="1" />
        <input name="down" type="hat" id="0" value="4" />
        <input name="left" type="hat" id="0" value="8" />
        <input name="menu" type="button" id="1" value="1" />
        <input name="pagedown" type="button" id="4" value="1" />
        <input name="pageup" type="button" id="6" value="1" />
        <input name="right" type="hat" id="0" value="2" />
        <input name="select" type="button" id="9" value="1" />
        <input name="up" type="hat" id="0" value="1" />
    </inputConfig>
</inputList>

```

6.1.3 Splash screen

```

sudo mv /etc/splashscreen.png /etc/splashscreen.orig.png
sudo wget http://blog.petrockblock.com/wp-content/uploads/2013/06/splash.png -O /etc/splashscreen.png

```

6.1.4 System Overclocking

```

cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor

ondemand

cat /sys/devices/system/cpu/cpu0/cpufreq/cpuinfo_max_freq

900000

sudo cat /sys/devices/system/cpu/cpu0/cpufreq/cpuinfo_cur_freq

700000

cat /boot/config.txt | egrep "freq|volt"

arm_freq=900
core_freq=250
sdram_freq=450
over_voltage=2

sudo raspi-config

7 Overclock
<Ok>
High 950MHz ARM, 250MHz core, 450MHz SDRAM, 6 overvolt
<Ok>

```


Would you like to reboot now? <Yes>

```
cat /sys/devices/system/cpu/cpu0/cpufreq/cpuinfo_max_freq
```

```
950000
```

```
cat /boot/config.txt | egrep "freq|voltage"
```

```
arm_freq=950
core_freq=250
sdram_freq=450
over_voltage=6
```

6.2 RetroArch - Sony PS1/PSX

Mit dem Programm „retroarch-joyconfig“ können die Tasten des Joypads konfiguriert werden danach werden die Einstellungen am Bildschirm ausgegeben und man kann sie in die Konfigurationsdatei übernehmen.

/home/pi/RetroPie/configs/psx/retroarch.cfg

```
rewind_enable = false

input_player1_joypad_index = "0"
input_player2_joypad_index = "1"

#Logitech, Inc. Logitech RumblePad 2 USB
input_player1_b_btn = "1"
input_player1_y_btn = "0"
input_player1_select_btn = "8"
input_player1_start_btn = "9"
input_player1_up_axis = "-5"
input_player1_down_axis = "+5"
input_player1_left_axis = "-4"
input_player1_right_axis = "+4"
input_player1_a_btn = "2"
input_player1_x_btn = "3"
input_player1_l_btn = "6"
input_player1_r_btn = "7"
input_player1_l2_btn = "4"
input_player1_r2_btn = "5"
input_player1_l3_btn = "10"
input_player1_r3_btn = "11"
input_player1_l_x_plus_axis = "+0"
input_player1_l_x_minus_axis = "-0"
input_player1_l_y_plus_axis = "+1"
input_player1_l_y_minus_axis = "-1"
input_player1_r_x_plus_axis = "+2"
input_player1_r_x_minus_axis = "-2"
input_player1_r_y_plus_axis = "+3"
input_player1_r_y_minus_axis = "-3"

#GreenAsia Inc. MaxFire Blaze2
input_player2_b_btn = "2"
input_player2_y_btn = "0"
input_player2_select_btn = "8"
input_player2_start_btn = "9"
input_player2_up_axis = "-5"
```

```
input_player2_down_axis = "+5"
input_player2_left_axis = "-4"
input_player2_right_axis = "+4"
input_player2_a_btn = "3"
input_player2_x_btn = "1"
input_player2_l_btn = "5"
input_player2_r_btn = "7"
input_player2_l2_btn = "4"
input_player2_r2_btn = "6"
input_player2_l3_btn = "10"
input_player2_r3_btn = "11"
input_player2_l_x_plus_axis = "+0"
input_player2_l_x_minus_axis = "-0"
input_player2_l_y_plus_axis = "+1"
input_player2_l_y_minus_axis = "-1"
input_player2_r_x_plus_axis = "+3"
input_player2_r_x_minus_axis = "-3"
input_player2_r_y_plus_axis = "+2"
input_player2_r_y_minus_axis = "-2"
```

```
cat /home/pi/RetroPie/configs/all/retroarch.cfg | grep system_directory
```

```
system_directory = /home/pi/RetroPie/emulatorcores/
```

Bios-Datei (scph<Version>.bin mit 512 KB z.B. scph7001.bin) in das /home/pi/RetroPie/emulatorcores/ Verzeichnis kopieren.

```
/home/pi/RetroPie/configs/psx/retroarch.cfg
```

```
audio_enable = true
```

6.3 RetroArch - Nintendo SNES

```
cp /home/pi/RetroPie/configs/psx/retroarch.cfg /home/pi/RetroPie/configs/snes/retroarch.cfg
```

6.4 DGen - Sega Mega Drive / Genesis)

```
/home/pi/.emulationstation/es_systems.cfg [-rw-r--r-- pi pi]
```

```
DESCNAME=Sega Mega Drive / Genesis
NAME=megadrive
PATH=/home/pi/RetroPie/roms/megadrive
EXTENSION=.smd .SMD .gen .bin .BIN
```

```
/home/pi/RetroPie/configs/all/dgenrc [-rw-r--r-- pi pi]
```

```
# Joystick/joypad definitions for pad 1
# Defaults 10/12 button controller
joy_pad1_up = joystick0-axis1-min
joy_pad1_down = joystick0-axis1-max
joy_pad1_left = joystick0-axis0-min
joy_pad1_right = joystick0-axis0-max
joy_pad1_a = joystick0-button0
joy_pad1_b = joystick0-button1
joy_pad1_c = joystick0-button2
joy_pad1_x = joystick0-button3
```

```
joy_pad1_y = joystick0-button5
joy_pad1_z = joystick0-button7
joy_pad1_mode = joystick0-button8
joy_pad1_start = joystick0-button9

# M68K and Z80 cores to use at startup.
#emu_m68k_startup = musa
#emu_z80_startup = cz80
emu_z80_startup = drz80
emu_m68k_startup = cyclone

# These decide whether DGen should automatically load slot 0 on startup,
# and/or autosave to slot 0 on exit.
bool_autoload = yes
bool_autosave = yes

# Quit DGen or switch to the next ROM on the command-line.
key_quit = escape
joy_quit = joystick0-button11

# Reset Genesis
key_reset = tab
#12 button controller
joy_reset = joystick0-button10

key_save = f2
key_load = f3
joy_save = joystick0-button4
joy_load = joystick0-button6

# Display the current number of frames per second.
bool_fps = true

# Display pressed buttons. Can be used to help configuring them.
bool_buttons = false
```