

# Das andere 3d

## Bluetooth aktivieren

- `sudo apt update && sudo apt upgrade -y && sudo apt install git bluez bluetooth pulseaudio pulseaudio-utils pulseaudio-module-bluetooth blueman pavucontrol pi-bluetooth`
- `sudo nano /boot/config.txt`
  - Diese beiden Elemente auskommentieren :

```
enable_uart=1
dtoverlay=disable-bt
```

- Treiber laden  
`sudo nano /etc/modules-load.d/modules.conf`
  - `btusb` einfügen
- Dienste aktivieren
  - `sudo systemctl enable hciuart.service`
  - `sudo systemctl enable bluetooth.service`
- `sudo usermod -a -G bluetooth pi`
- `sudo reboot`

## Joystick koppeln

- `bluetoothctl`
  - `scan on`
    - [NEW] Device DC:0C:2D:5A:FC:EE Wireless Controller
  - `pair DC:0C:2D:5A:FC:EE`

```
[bluetooth]# pair DC:0C:2D:5A:FC:EE
Attempting to pair with DC:0C:2D:5A:FC:EE
[CHG] Device DC:0C:2D:5A:FC:EE Connected: yes
[CHG] Device DC:0C:2D:5A:FC:EE UUIDs:
00001124-0000-1000-8000-00805f9b34fb
[CHG] Device DC:0C:2D:5A:FC:EE UUIDs:
00001200-0000-1000-8000-00805f9b34fb
[CHG] Device DC:0C:2D:5A:FC:EE ServicesResolved: yes
[CHG] Device DC:0C:2D:5A:FC:EE Paired: yes
Pairing successful
Authorize service
[agent] Authorize service 00001124-0000-1000-8000-00805f9b34fb
(yes/no): yes
```

- `trust DC:0C:2D:5A:FC:EE`

```
[Wireless Controller]# trust DC:0C:2D:5A:FC:EE
[CHG] Device DC:0C:2D:5A:FC:EE Trusted: yes
```

```
Changing DC:0C:2D:5A:FC:EE trust succeeded
```

- connect DC:0C:2D:5A:FC:EE

```
[Wireless Controller]# connect DC:0C:2D:5A:FC:EE
Attempting to connect to DC:0C:2D:5A:FC:EE
Connection successful
```

### • Gamepad (PS4) testen

- Tools installieren  
sudo apt install evtest joystick jstest-gtk -y
- Verfügbare Controller auflisten  
ls /dev/input/js\*
- Test starten

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

In der Ausgabe nach "Axes" schauen die nicht auf 0 gehen.

**Hinweis** : Axes 2 & 5 sind meistens die Shoulderbuttons und stehen auf -32676

```
pi@Pi3Test:~$ jstest /dev/input/js0
Driver version is 2.1.0.
Joystick (Sony Interactive Entertainment DualSense Wireless Controller) has 8 axes (X,
and 13 buttons (BtnA, BtnB, BtnX, BtnY, BtnTL, BtnTR, BtnTL2, BtnTR2, BtnSelect, BtnSt
Testing ... (interrupt to exit)
Axes: 0: 258 1: 774 2:-32767 3: 774 4: 516 5:-32767 6: 0 7: 0
```

## Lautsprecher koppeln

### • bluetoothctl

- scan on

```
[NEW] Device B8:D5:0B:C7:F1:0A JBL Charge 3
```

- pair B8:D5:0B:C7:F1:0A

```
[bluetooth]# pair B8:D5:0B:C7:F1:0A
Attempting to pair with B8:D5:0B:C7:F1:0A
[CHG] Device B8:D5:0B:C7:F1:0A Connected: yes
[CHG] Device B8:D5:0B:C7:F1:0A UUIDs:
00001101-0000-1000-8000-00805f9b34fb
[CHG] Device B8:D5:0B:C7:F1:0A UUIDs:
00001108-0000-1000-8000-00805f9b34fb
[CHG] Device B8:D5:0B:C7:F1:0A UUIDs:
0000110b-0000-1000-8000-00805f9b34fb
[CHG] Device B8:D5:0B:C7:F1:0A UUIDs:
0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device B8:D5:0B:C7:F1:0A UUIDs:
0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device B8:D5:0B:C7:F1:0A UUIDs:
0000111e-0000-1000-8000-00805f9b34fb
[CHG] Device B8:D5:0B:C7:F1:0A UUIDs:
00001200-0000-1000-8000-00805f9b34fb
[CHG] Device B8:D5:0B:C7:F1:0A UUIDs:
```

```
00001801-0000-1000-8000-00805f9b34fb
[CHG] Device B8:D5:0B:C7:F1:0A ServicesResolved: yes
[CHG] Device B8:D5:0B:C7:F1:0A Paired: yes
Pairing successful
```

- trust B8:D5:0B:C7:F1:0A

- [JBL Charge 3]# trust B8:D5:0B:C7:F1:0A  
[CHG] Device B8:D5:0B:C7:F1:0A Trusted: yes  
Changing B8:D5:0B:C7:F1:0A trust succeeded

- connect B8:D5:0B:C7:F1:0A

- [bluetooth]# connect B8:D5:0B:C7:F1:0A  
Attempting to connect to B8:D5:0B:C7:F1:0A  
[CHG] Device B8:D5:0B:C7:F1:0A Connected: yes  
Connection successful  
[CHG] Device B8:D5:0B:C7:F1:0A ServicesResolved: yes  
[JBL Charge 3]#

- exit

- **Lautsprecher testen**

- speaker-test -c2 -twav -l3
- wget <https://download.samplelib.com/mp3/sample-15s.mp3>
- cvlc sample-15s.mp3

## Bluetooth Checkup

Wenn alles erfolgreich gekoppelt ist, dann solltet ihr jetzt folgende Ausgabe bekommen bei `bluetoothctl devices`:

```
pi@Pi4Test:~ $ bluetoothctl devices
Device DC:0C:2D:5A:FC:EE Wireless Controller
Device B8:D5:0B:C7:F1:0A JBL Charge 3
```

## Doom kompilieren (chocolate)

- basierend auf [https://www.chocolate-doom.org/wiki/index.php/Building\\_Chocolate\\_Doom\\_on\\_Debian](https://www.chocolate-doom.org/wiki/index.php/Building_Chocolate_Doom_on_Debian)
- `sudo apt install gcc make libsdl2-dev libsdl2-net-dev libsdl2-mixer-dev automake autoconf libtool git pkg-config`
- `git clone https://github.com/chocolate-doom/chocolate-doom.git`
- `cd chocolate-doom`
- `./autogen.sh`
- `make -j4`
- `sudo make install`

# Doom einrichten

## WAD Datei kopieren

Ihr braucht für Doom eine WAD Datei. Darin enthalten sind letztlich die Maps die ihr spielen könnt. Da diese Dateien immer noch unter Copyright stehen habt ihr folgende Möglichkeiten:

### 1. Shareware Version

- wget

[http://www.doomworld.com/3ddownloads/ports/shareware\\_doom\\_iwad.zip](http://www.doomworld.com/3ddownloads/ports/shareware_doom_iwad.zip)

- unzip shareware\_doom\_iwad.zip

### 2. Original Version

Ihr habt das Spiel gekauft und kopiert euch die WAD Datei.

### 3. Internet Version

*Zensiert wegen Copyright*

## Klipperscreen stoppen

- `sudo systemctl stop KlipperScreen.service`

## Test

- `DISPLAY=:0 chocolate-doom -iwad DOOM1.WAD`

## Probleme

- `sudo nano /etc/bluetooth/main.conf`

- `FastConnectable = true`

- Probleme mit Audio :

`sudo nano /etc/pulse/default.pa`

- ```
# automatically switch to newly-connected devices
load-module module-switch-on-connect
```

- `sudo nano /etc/dbus-1/system.d/bluetooth.conf`

- ```
<!-- allow users of bluetooth group to communicate with hcid -->
<policy group="bluetooth">
  <allow send_destination="org.bluez"/>
</policy>
```

- ```
[bluetooth]# scan on
Failed to start discovery: org.bluez.Error.InProgress
```

<https://github.com/Hexxeh/rpi-update>  
kann auch org.bluez.Error.NotReady sein

From:  
<https://drklipper.de/> - **Dr. Klipper Wiki**

Permanent link:  
[https://drklipper.de/doku.php?id=3d\\_druck:das\\_andere\\_3d&rev=1711811109](https://drklipper.de/doku.php?id=3d_druck:das_andere_3d&rev=1711811109)

Last update: **2024/03/30 16:05**

