Bootdatenträger klonen (Pi 4/5)

Diese Anleitung sollte es ermöglichen, eine SD-Karte aus einem Pi 4 / Pi 5 auf ein neues Medium zu klonen. Es gibt ein passendes grafisches Tool namens `sdcopier`. Da das aber eine grafische Oberfläche benötigt, wird hier darauf verzichtet. Stattdessen wird der komplette Klonvorgang in der Konsole durchgeführt.

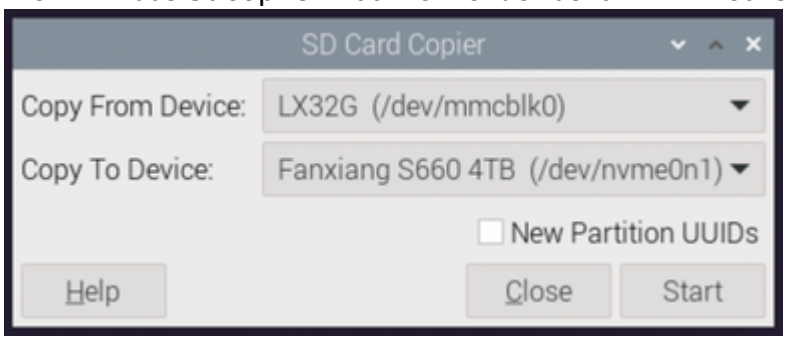
> **Hinweis**
> 1) Die Anleitung ist auf dem Pi 4 noch ungetestet. Ebenso auf dem Pi 5 mit NVMe's > 512GB.
> 2) Die Anleitung funktioniert (zumindest beim Pi 5) auch mit einem Kloning von SD-Karte zu USB-Stick. Theoretisch sollte man in alle Richtungen klonen können (SD > USB, SD > NVMe, USB > SD, USB > NVMe, NVMe > SD, NVMe > USB).
> 3) Aufpassen bei dem Quell- und Ziellaufwerk. Man kann sich ganz schnell das bestehende System zerschießen (Leidvolle Eigenerfahrung ...)

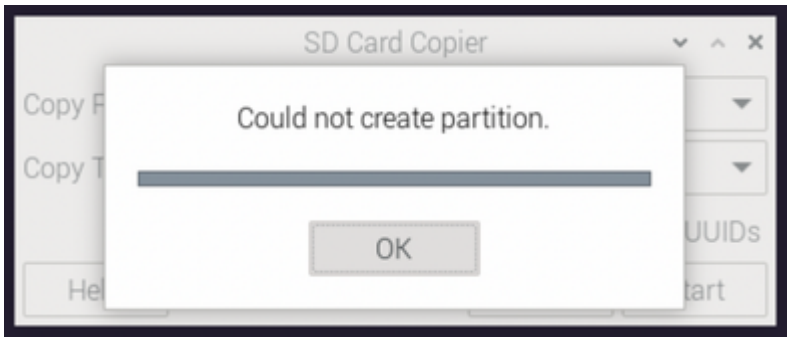
YouTube Video #84

NVMe SSD's > 2 TB

Wenn ihr eine NVMe SSD verwendet, die mehr als 2 TB an Speicher hat, dann funktioniert das Tool `sdcopier` nicht. Das liegt daran, dass das Tool ein MBR (Master Boot Record) Partitionsschema verwendet. Das kann aber maximal Partitionen mit 2 GB unterstützen. Um diese großen NVMe's verwenden zu können, muss GPT als Partitionsschema verwendet werden. Genau das macht diese Anleitung.

Wenn ihr das `sdcopier` Tool verwendet bekommt ihr sonst folgende Meldung:





Das Klonen ist in dem Fall auch nicht erfolgreich!

System vorbereiten

- Es muss der aktuelle Bootloader auf dem Pi 5 installiert sein! Siehe dazu Bootloader Update
  - Check mittels `sudo rpi-eeprom-update`
- Evtl. muss auch die Bootreihenfolge (Pi 5) angepasst werden. Siehe dazu Bootloader Konfig
- ein paar Tools installieren auf dem zu klonenden System
  `sudo apt install gparted gdisk dosfstools mtools iotop -y`
- Dienste stoppen die ggf. eine größeren IO Traffic auf einem Massenspeicher generieren könnten Hier in paar Beispiele:
  - Klipper `sudo systemctl stop klipper`
  - Datenbanken MySQL `sudo systemctl stop mysqld`
  - Webserver Apache `sudo systemctl stop apache2`
  - Samba Service `sudo systemctl stop smbd.service`
- Nicht vergessen auf dem Klon später die Dienste wieder zu starten 😉
- Prüfen was ggf. hohe IO Last erzeugt geht mittels `sudo iotop –only`

neues Medium vorbereiten

> **ACHTUNG**
> Alle Daten auf dem neuen Medium werden gelöscht!!
>
> **Hinweis**
> In diesem Beispiel wird auf einem Pi 5 von einem USB-Stick (Bootmedium) auf eine SD-Karte geklont!

- den Pi booten und das neue Laufwerk mit anschließen
- aktuelles Bootmedium (also das Laufwerk, was wir auf das neue Medium klonen wollen) ermitteln
  `lsblk`

```
pi@Pi5Test:~ $ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda           8:0    1 28.6G  0 disk
├─sda1        8:1    1  512M  0 part /boot/firmware
└─sda2        8:2    1 28.1G  0 part /
```

```
mmcblk0     179:0    0 58.9G  0 disk
├─mmcblk0p1 179:1    0  512M  0 part /media/pi/bootfs
└─mmcblk0p2 179:2    0 58.4G  0 part /media/pi/rootfs
```

    - Das Bootmedium ist immer das mit dem “Mountpoint /”. Im obigen Fall liegt `/` auf der Partition `sda2` und die liegt auf dem Device sda.
      Das **Bootmedium** ist als hier **sda**!
- Das Medium, auf das wir klonen wollen, können wir ebenfalls in der `lsblk` Ausgabe sehen - in dem Fall `mmcblk0`. Man kann auch gleich sehen, dass die beiden Partitionen auf dem Laufwerk automatisch gemountet wurden.
- Ergebnis bis hier:
    - **Quelllaufwerk : sda**
    - **Ziellaufwerk : mmcblk0**
    - Ziellaufwerk enthält Partitionen → Partitionen aushängen
- Unmount der Partitionen auf dem Ziellaufwerk
    - `sudo umount /media/pi/bootfs`
    - `sudo umount /media/pi/rootfs`
- Neue Partitionen anlegen (inkl. Umstieg auf GPT als Partitionsschema)
    - `sudo cgdisk /dev/mmcblk0`
    - Es können 1-2 Meldung kommen wie z.B.:
      `Warning! Non-GPT or damaged disk detected! This program will attempt to convert to GPT form or repair damage to GPT data structures, but may not succeed. Use gdisk or another disk repair tool if you have a damaged GPT disk.`

      Die quitieren wir aber einfach mit einem Tastendruck 🙂
    - Jetzt in der Oberfläche die Partitionen anwählen und durch `[ Delete ]` löschen.
      Das wiederholt man so lange, bis nur noch ein Eintrag mit `free space` vorhanden ist.
    - Nun 2 neue Partitionen anlegen jeweils über `[ New ]`
        - **Partition 1**
          First sector → Enter
          Size in sectors or {KMGTP} → 512M
          Hex code or GUID → 0700
          Enter new partition name → bootfs
        - Jetzt mit dem Cursor **unten auf free space** navigieren!
        - **Partition 2**
          First sector → Enter
          Size in sectors or {KMGTP} → Enter
          Hex code or GUID → Enter
          Enter new partition name → rootfs
    - Das sollte dann in etwa so aussehen:

```
Part. #     Size        Partition Type            Partition Name
----------------------------------------------------------------
            1007.0 KiB  free space
   1        512.0 MiB   Microsoft basic data      bootfs
   2        58.4 GiB    Linux filesystem          rootfs
            1007.5 KiB  free space
```

    - Jetzt auf `[ Write ]`, `yes` eintippen und Enter
    - Über `[ Quit ]` den Dialog verlassen

- Prüfen kann man das Ziellaufwerk jetzt nochmal mit `sudo sgdisk -p /dev/mmcblk0`
- Jetzt müssen die Partitionen noch formatiert werden ...
    - `sudo mkfs.vfat /dev/mmcblk0p1 -n bootfs -v`
    - `sudo mkfs.ext4 /dev/mmcblk0p2 -L rootfs -v` → Evtl. mit y bestätigen
    - **Hinweis:** die Partitionsnamen findet man immer mittels `lsblk`!

Prüfen ob das neue Medium mit einem GPT Partitionsschema arbeitet. (Wichtig für Medien > 2TB Speicherkapazität):

- `sudo fdisk -l |grep Disklabel -A 1 -B 4`

```
pi@Pi5Test:~ $ sudo fdisk -l |grep Disklabel -A 1 -B 4
Disk model: GIGABYTE GP-GSM2NE3256GNTD
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: ADB4BD3E-02B3-4BCF-81E4-B00633730E50
```

gparted (grafisch)

Wer das Partitionieren grafisch erledigen möchte kann das mit MobaXTerm so machen:

- `export XAUTHORITY=~/.Xauthority`
- `sudo gparted`
    - Neues Medium oben rechts auswählen
    - alle bestehenden Partitionen löschen
    - mit dem grünen Haken übernehmen
    - umstellen auf GPT Partitionsschema
        - Device → Create Partition Table ... → gpt
    - neue FAT32 Partition anlegen mit 512MB (Label bootfs)
    - neue ext4 Partition anlegen mit dem Restspeicherplatz (Label rootfs)
    - mit dem grünen Haken übernehmen
    - fertig

Mounting

Für den Klonvorgang müssen die neuen Partionen gemountet werden:

- `sudo mkdir /klonboot`
- `sudo mkdir /klonfs`
- `sudo mount /dev/mmcblk0p1 /klonboot`
- `sudo mount /dev/mmcblk0p2 /klonfs`

Klonen

In diesem Schritt werden die Daten dateibasiert auf den Klon übertragen...

- Boot Partition klonen
  BookWorm Image
  `sudo rsync -rltDv --numeric-ids --info=progress2 /boot/firmware/ /klonboot`
  BullsEye Image
  `sudo rsync -rltDv --numeric-ids --info=progress2 /boot/ /klonboot`
- System Partition klonen
  `sudo rsync -axHAWXSv --numeric-ids --info=progress2 / /klonfs`

Man kann jetzt `df -h` verwenden, um zu schauen pb Quelle und Ziel in etwa gleich viele Daten enthalten (passt nicht 100%):

```
pi@Pi5Test:~ $ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            1.9G     0  1.9G   0% /dev
tmpfs           405M  5.8M  399M   2% /run
/dev/sda2        28G  4.3G   22G  17% /
tmpfs           2.0G  368K  2.0G   1% /dev/shm
tmpfs           5.0M   48K  5.0M   1% /run/lock
/dev/sda1       510M   73M  438M  15% /boot/firmware
tmpfs           405M  160K  405M   1% /run/user/1000
/dev/mmcblk0p1  511M   73M  439M  15% /klonboot
/dev/mmcblk0p2   58G  4.0G   51G   8% /klonfs
```

> **Hinweis**
> Beim BullsEye ist /boot eine eigene FAT32 Partition. Bei BookWorm ist /boot aber mit im Root Filesystem. Dort wird dann /boot/firmware als FAT32 Partition eingebunden anstatt /boot!

fstab abpassen

Jetzt muss auf dem Klon Laufwerk noch die Datei `fstab` angepasst werden.

- Neue PARTUUID's ermitteln
  `lsblk -o NAME,FSTYPE,SIZE,MOUNTPOINT,LABEL,PARTUUID`

```
pi@Pi5Test:~ $ lsblk -o NAME,FSTYPE,SIZE,MOUNTPOINT,LABEL,PARTUUID
NAME        FSTYPE  SIZE MOUNTPOINT     LABEL  PARTUUID
sda                28.6G
├─sda1      vfat    512M /boot/firmware bootfs 9c1b969d-01
└─sda2      ext4   28.1G /              rootfs 9c1b969d-02
mmcblk0            58.9G
├─mmcblk0p1 vfat    512M /klonboot      bootfs 4c5ca341-4453-47a6-8635-
b20e4380b4e7
└─mmcblk0p2 ext4   58.4G /klonfs        rootfs
78f45d12-57af-496f-9288-066e8694eee8
```

    - Wichtig sind die PARTUUID's von unserem Ziellaufwerk (mmcblk0). Wir brauchen beide PARTUUID's für die fstab!
- `sudo nano /klonfs/etc/fstab`
    - neue PARTUUID s eintragen für beide Partitionen
    - Alt

```
proc            /proc           proc    defaults          0
0
PARTUUID=9c1b969d-01  /boot/firmware  vfat    defaults          0
2
PARTUUID=9c1b969d-02  /               ext4    defaults,noatime  0
1
# a swapfile is not a swap partition, no line here
#   use  dphys-swapfile swap[on|off]  for that
```

    - Neu

```
proc            /proc           proc    defaults          0
0
PARTUUID=4c5ca341-4453-47a6-8635-b20e4380b4e7  /boot/firmware
vfat    defaults          0       2
PARTUUID=78f45d12-57af-496f-9288-066e8694eee8  /
ext4    defaults,noatime  0       1
# a swapfile is not a swap partition, no line here
#   use  dphys-swapfile swap[on|off]  for that
```

    - Editor beenden mit STRG-x, Y und Enter

cmdline.txt abpassen

Die Datei `cmdline.txt` muss auch angepasst werden, weil der Pi sonst sein Root Dateisystem nicht finden kann.

- `sudo nano /klonboot/cmdline.txt`
    - Alt
      `.. root=PARTUUID=9c1b969d-02 ..`
    - Neu
      `.. root=PARTUUID=78f45d12-57af-496f-9288-066e8694eee8 ..`
    - Editor beenden mit STRG-x, Y und Enter

Finalisierung

- `sudo umount /klonboot && sudo umount /klonfs`
- `sudo poweroff`
- Das alte Bootmedium entfernen (in diesem Beispiel der USB-Stick)
- Raspberry Pi booten und hoffen das er läuft 😎

Kontrolle

Wer Langeweile hat kann auch noch die Differenzen analysieren 😛
`sudo diff -qr / /media/pi/rootfs/`
Der hintere Pfad muss natürlich angepasst werden!

Probleme

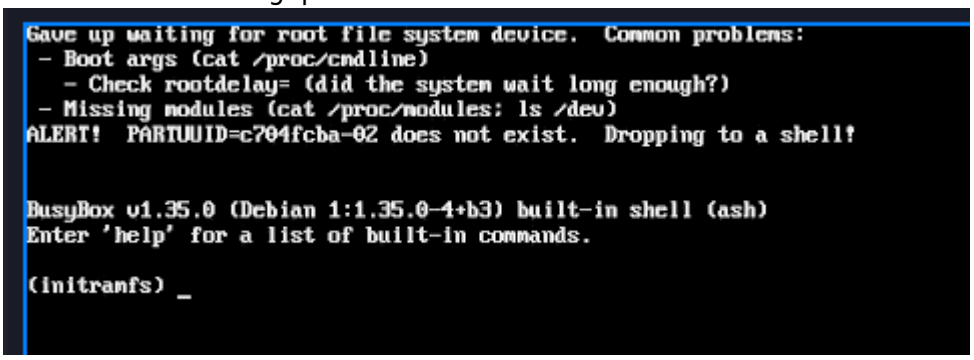
NVMe gebootet

Wenn die Bootreihenfolge auf NVMe, USB, SD eingestellt ist und von der NVMe gebootet wurde ...

- `sudo umount /dev/nvme0n1p1`
- `sudo gdisk /dev/nvme0n1`
  - d → 1 → w → Y
- `sudo reboot`

cmdline.txt

- cmdline.txt nicht angepasst



Host SSH Keys

- Gleicher Host SSH Keys
  Zum Erneuern der ssh-Host-Keys löscht man auf dem Server zunächst die bestehenden Host-Keys und generiert einen Satz neue Keys
  - `cd /etc/ssh`
  - `sudo rm ssh_host_*`
  - `sudo ssh-keygen -A`

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

Permanent link:
https://www.drklipper.de/doku.php?id=sbcs:raspberry_pi:bootdatentraeger_klonen_pi_4_5&rev=1706133762

Last update: **2024/01/24 23:02**