Chemnitzer Linux-Tage
22. und 23. März 2025

Irgendwas mit Medien

Daten Speichern unter Linux

Holger Assmann – h.assmann@pengutronix.de

Kurzvorstellung

- Embedded Linux Engineer bei Pengutronix
- PTXdist und Yocto BSP Integration
- Yak Barbier

Pengutronix.

- Embedded Linux Consulting & Support seit 2001
- > 7500 patches in Linux kernel

Kontext

"Daten Speichern unter Linux"

- Motivation aus dem Bereich eingebettete Systeme
  - "Wo landen die Daten auf meinem Speicher?"
- Heute leider ohne: Raid, LVM, DM, Crypt, Kompression, ...
- Heute mit Fokus auf: ext4 und Flash
- Nachmachen erbeten

Beispieldatei

© *https://pingus.github.io* Ingo Ruhnke

- "single_walker.png"
- 23x30 px
- Größe: 814 Bytes

Beispieldatei

```
89 50 4e 47 0d 0a 1a 0a 00 00 00 0d 49 48 44 52
00 00 00 17 00 00 00 1e 08 06 00 00 00 c7 25 85
68 00 00 00 09 70 48 59 73 00 00 0b 13 00 00 0b
13 01 00 9a 9c 18 00 00 00 07 74 49 4d 45 07 e9
03 12 05 16 1d 39 7a 1e cb 00 00 02 cd 49 44 41
54 48 c7 cd 95 a1 6f e3 48 14 87 bf e9 2e 18 b3
09 a8 34 86 03 27 cc 65 2e f4 b2 2d 4b 99 0b 73
ac e4 a4 c2 90 03 f7 27 dc b1 2e dc 63 7b ac 85
0b 13 96 b2 9a d5 e0 4e 6a 98 0d 56 f2 80 48 6f
81 e3 34 6d e2 36 1b ad 74 f7 a4 91 ec b1 e6 7b
e3 37 bf df 1b f8 3f 84 f7 5e 00 01 64 f5 fc 73
c2 18 b3 06 77 c3 39 f7 53 12 48 df 58 25 ed 8d
a3 3d 4a b1 15 79 9e 53 55 15 17 17 17 87 6f 59
6b bd 2e 81 b5 56 00 19 8d 46 52 55 95 58 6b c5
5a 2b cd fc 8f de dd ab 3d 4a 82 73 8e 34 4d 89
a2 88 8f 49 c5 e8 7c cc 9f 5f 4a 2e 07 bf 83 5d
81 3e 2c d4 41 f0 2e ae 2f 3d e3 8f 06 e2 18 cc
09 14 37 50 ce 20 36 e0 4e 81 04 c2 0c 75 fa 55
01 bc 7f a3 e6 14 45 b1 7e 9f cd 0b 86 18 52 77
07 71 37 af a1 aa a1 ba 05 6e 21 3c ad 7f f7 1a
fc f8 f8 f8 b7 e5 72 89 d6 9a 10 02 77 ff c2 f4
9f 76 7d a8 17 b8 f7 df 60 19 a0 06 be 01 4b 8d
fa 75 a9 f6 36 8e 73 4e 76 e9 5c 83 5c 5f 65 72
b0 23 3b 85 bc a6 f3 1f 76 ab f7 7e 6b b7 c6 18
49 92 44 3a 79 02 92 24 89 58 6b 7f 2c c1 2e 70
9e e7 92 65 d9 b3 fe 32 9f cf e5 fe fe 5e b4 d6
fb 25 d8 6c 50 9b e0 24 49 d6 73 d6 5a 99 4c 26
d2 34 8d 88 34 32 1e 8f df 6e 68 2f c1 80 e4 79
2e 69 9a 3e 1d a4 d6 92 65 99 dc dc dc 88 48 0b
9f 4e a7 eb 72 ed 4c d0 d9 b9 b3 7a 9a a6 5b e0
ae 15 4c 26 13 79 78 78 58 c3 9b a6 91 d1 68 b4
05 5d 9b 68 b1 58 a8 2e ab f7 9e 10 02 b3 d9 8c
b2 2c 37 cf 02 ef 3d c3 e1 f0 45 0f 6a 5b 44 2f
1c a0 28 0a 05 c8 a6 2b 37 63 30 18 e0 9c c3 18
b3 97 30 8e 76 59 be a7 43 62 8c 21 8e e3 9d df
77 cd 6f c1 fb 76 ad b5 26 8e 63 8c 31 84 10 0e
83 bf d1 df d1 5a 03 10 42 60 33 47 d3 34 87 c3
3b 70 14 45 68 ad a9 eb 9a b2 2c db 04 a1 64 3c
fc 0b 99 3f 57 cc 56 cb 75 ce 3d 53 48 5f 92 a6
69 da e7 af 2b 95 0c 62 a8 4a e4 8b 11 34 a8 b3
5a 1d 6d 5e 67 de 7b d9 f5 7b 5d 19 ba 88 a2 88
28 8a 30 d5 a7 27 70 03 84 76 ad 3a ab 9f 2e 8b
6e 61 df 61 76 51 d7 0b 34 8f 18 02 be ba 5a 81
87 d0 54 50 97 c0 29 ea fc 56 bd 2c 8b 5a 59 7b
a7 12 5a 70 4d 59 3e 52 17 9f b9 2b c0 3b c0 c6
50 de 43 13 c0 9d a3 3e 7c 52 7d 07 aa 42 08 ea
fa 2a 63 e4 59 ab e2 99 07 28 70 8f 05 b9 b9 6b
af b6 a2 00 7d 82 fa a5 56 2f c1 af 5e d0 d3 4b
23 d3 c7 c0 b4 68 ff e4 d4 6b 2e 4f 02 5a 03 da
80 cb 50 67 7f 2b fe ab f8 0e cc 03 89 75 ae 1f
e9 38 00 00 00 00 49 45 4e 44 ae 42 60 82
```

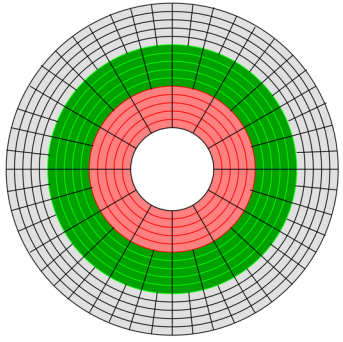
- "single_walker.png"
- 23x30 px
- Größe: 814 Bytes

Beispieldatei

```
$ ls -l ./single_walker.png
-rw-r--r-- 1 uid gid 814 18. Mar 06:22 single_walker.png

$ du -h ./single_walker.png
4,0K   single_walker.png
```

Dateisysteme - Blockgröße

- Daten werden auf der Hardware in Sektoren abgelegt
  - meist 512 Byte, heute eher 4096 Byte (AF)
- Das Dateisystem fasst Sektoren zu Blöcke zusammen
  - Standardwert für ext4: 4096 Byte
- Sektoren und Blöcke reduzieren Overhead bei Systemaufrufen, ECC, etc.
- Reduziert Anzahl benötigter Inodes

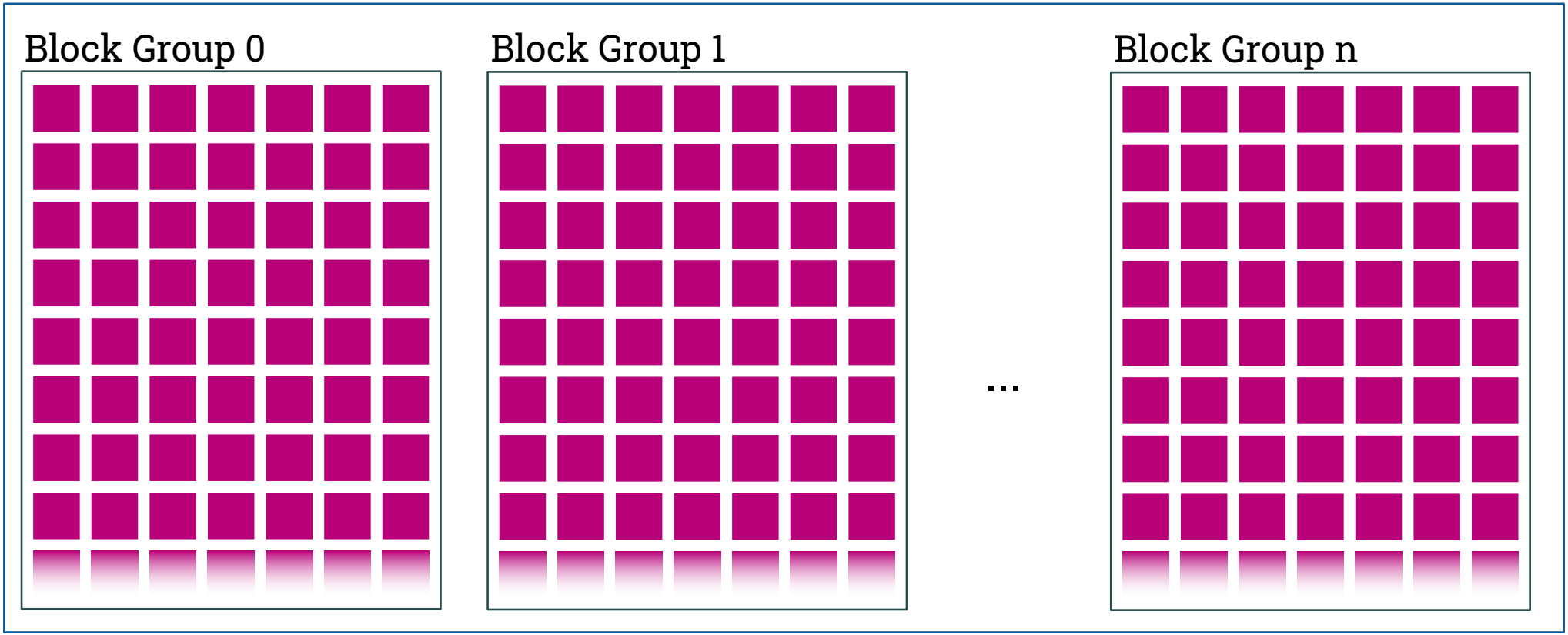


Wikimedia, Jan Schaumann

Inode (ext4)

- Metadaten der Dateien
  - Logische Adresse
  - Dateigröße
  - Typ (Datei | Symlink | Verzeichnis)
  - Besitzer und Rechte
  - Zeitstempel
  - …
- typ. 256 Bytes groß (ext4)

Struktur von ext4 Dateisystemen



ext4 - Block Groups

Block Group, 128 MiB

P 1024 Byte Padding für Boot Sectoren etc.

S Superblock, Informationen für das gesamte FS
- Anzahl Blöcke
- Anzahl Inodes
- Features des Dateisystems
- …
- Kann mit "`dumpe2fs`" ausgelesen werden

D Group Descriptors, Informationen über die Block Group
- Adressen von **bB**, **iB** und **I**
- Anzahl freier Blöcke und Inodes
- Anzahl der Verzeichnisse

bB **iB** Block- bzw. Inode Usage Bitmap

I Inode-Tabelle

ext4 - Eckdaten

Block Group, 128 MiB

32768 x 4 KiB

- Blockgröße 1 - 64 KiB, typ. 4 KiB
- 8192 - 524228 Blöcke pro Gruppe
- mehrere Block Gruppen können als "Logical Block Group" zusammengefügt werden
- max. 2^{64} Blöcke pro Partition
- max. 2^{32} Inodes Pro Partition

Journal - das Tagebuch des Dateisystems

- Protokoll über aktive Prozesse im Dateisystem
- Ziel: Murphy's Law abschwächen
  - Integrität des Dateisystems bei einem Absturz etc. erhalten
  - Schaden an betroffenen Dateien gering halten
  - Wiederherstellung beim Neustart beschleunigen
- Nebeneffekt: gesteigerte Schreibleistung im Normalbetrieb
- Alternative: Copy-on-Write Dateisysteme wie btrfs oder ZFS

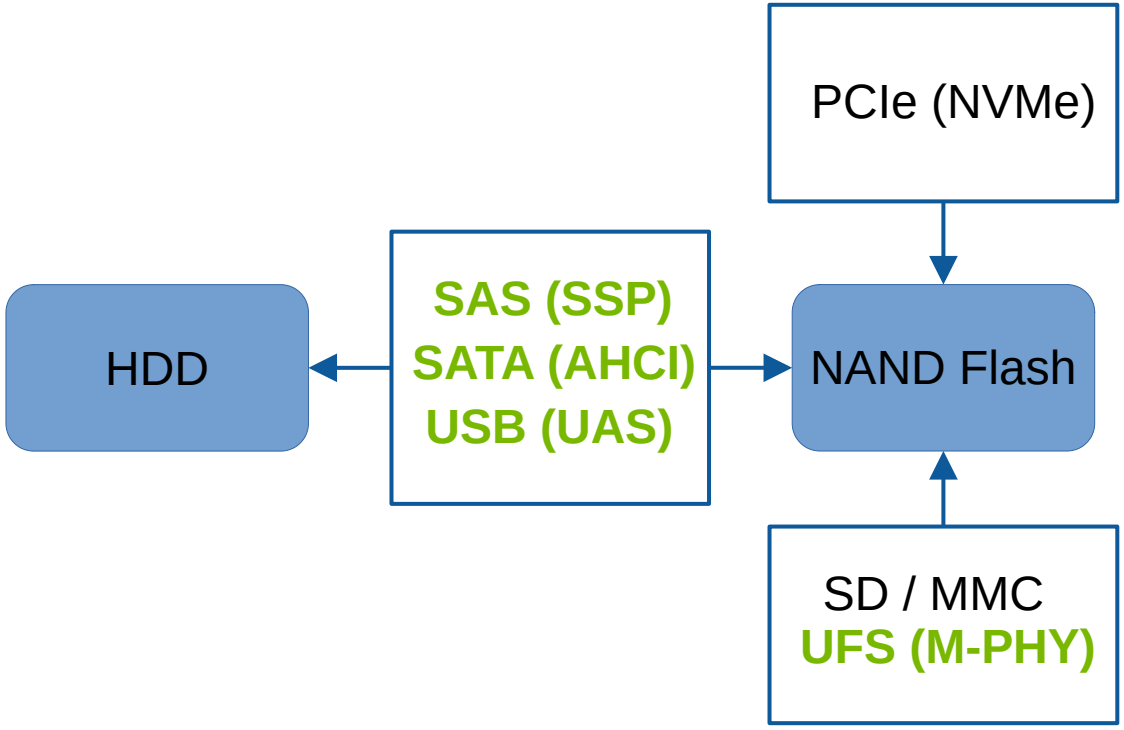
Journal bei ext4 (JBD2)

- Journal liegt in einer Block Group oder extern
  - Inode-Nummer ist typ. <8>
  - Meta-Daten des Journals liegen im Super-Block
- Verschiedene Modi möglich:
  - writeback (Metadaten im Journal)
  - ordered (default, Metadaten nach Dateiänderung im Journal)
  - journal (alle Dateien und Metadaten im Journal)
- Änderungen erst nur im Page Cache
  - Bündelung einzelner Aktionen
  - Schreiben, wenn Journal voll oder Zeitintervall vergangen

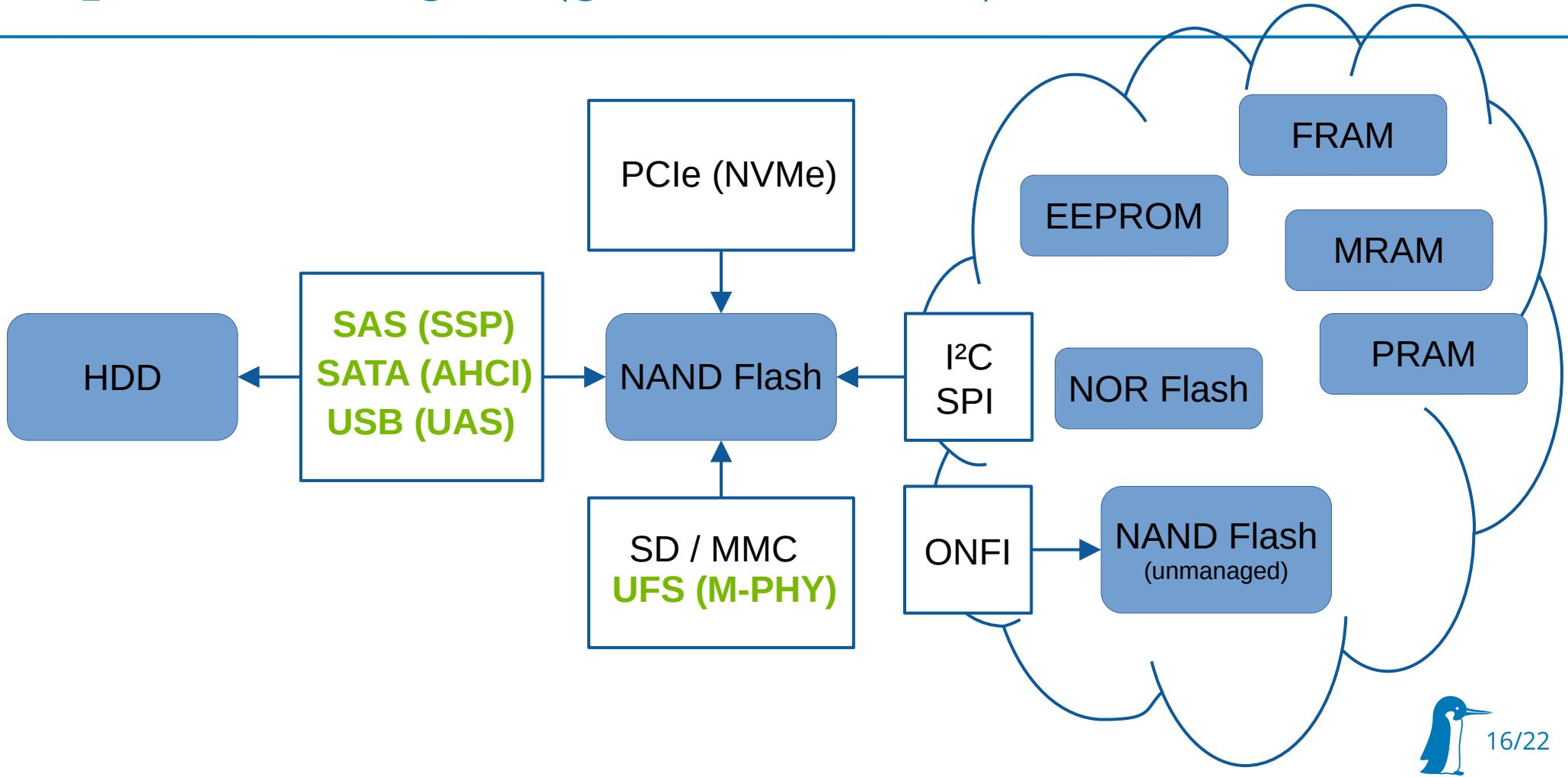
Wo liegt das Journal?

```
$ dumpe2fs /dev/sda1

[...]

Journal inode:            8
Journal backup:           inode blocks
Journal features:         journal_incompat_revoke journal_checksum_v3
Total journal size:       16M
Total journal blocks:     4096
Journal sequence:         0x0000007b
Journal start:            510
Journal checksum type:    crc32c
Journal checksum:         0xfd703782
```

Speicherwege... (generalisiert!)



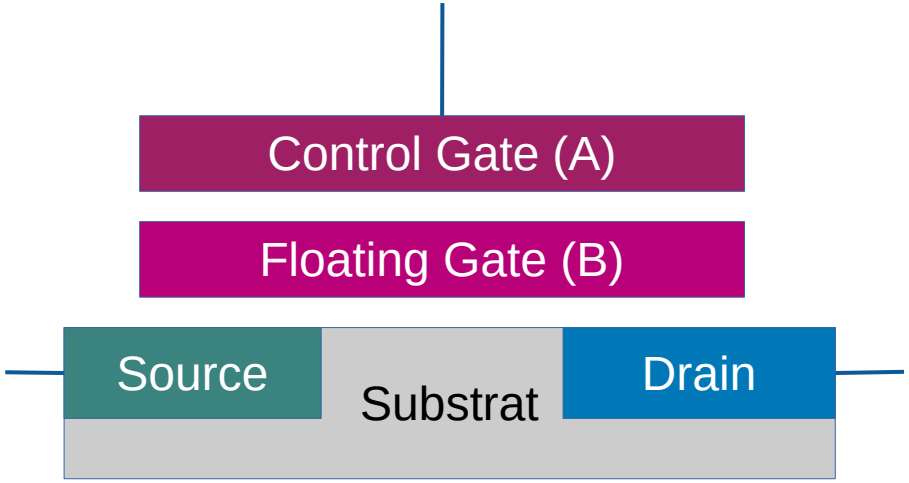
Speicherwege… (generalisiert!)



Die Datei... im (managed) Flash

NAND Flash | NOR Flash | EEPROM

- MOSFET Transistor + isoliertes Gate (B)
- Unterschiede in Anordnung und Ansteuerung der Transitoren
- Bei Spannung an (A) tunneln Elektronen von Drain nach (B)
- Elektronen bleiben gefangen und erzeugen elektrisches Feld
- Unterschiedliche Feldstärken können als Information genutzt werden (MLC, TLC, ...)
- Nur Lesen is verschleißfrei



Die Datei... im (managed) Flash

NAND Flash | NOR Flash | EEPROM

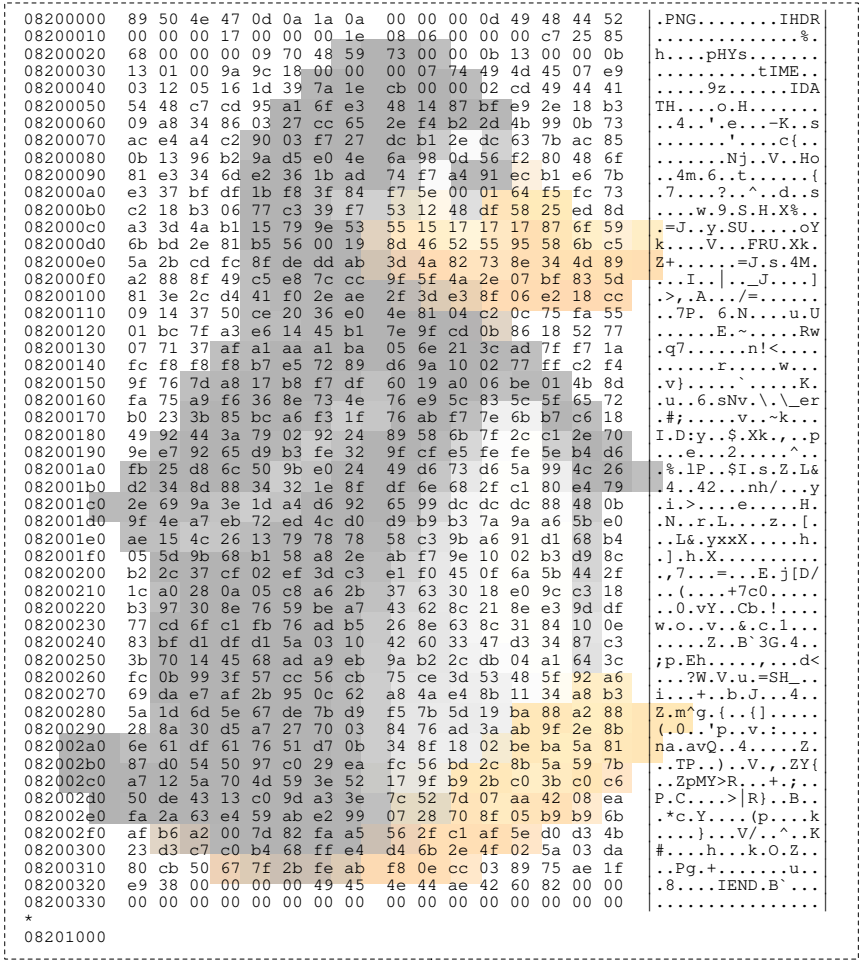
- Sektorgröße ursprünglich 512 Bytes
- Heute meist 4096 Bytes
- Dateisystem fasst Sektoren zu Blöcken zusammen
- Dateisystem adressiert die "Logical Block Address" (LBA)
- Controller führt "Physical Block Translation" durch (Mapping Table)
- Physische Block Adresse ändert sich durch
  - Dateiänderungen
  - Wear Leveling
  - Garbage Collection
  - Bad Blocks (auch bei HDDs)

Wo liegt nun die Datei?

```
$ ls -li ./single_walker.png
376 -rw-r--r-- 1 uid gid 814 18. Mar 06:22 single_walker.png
$ echo "stat <376>" | debugfs /dev/sda1
Inode: 376   Type: regular    Mode:  0644   Flags: 0x80000
Generation: 1795334941    Version: 0x00000000:00000002
User:     0   Group:     0   Project:     0   Size: 814
File ACL: 0
Links: 1   Blockcount: 8
Fragment:  Address: 0    Number: 0    Size: 0
 ctime: 0x67d9da3c:c4ad9e90 -- Tue Mar 18 21:40:28 2025
 atime: 0x67d9da3c:c4cd031c -- Tue Mar 18 21:40:28 2025
 mtime: 0x67d9da3c:c4ad9e90 -- Tue Mar 18 21:40:28 2025
crtime: 0x67d9da3c:c469814c -- Tue Mar 18 21:40:28 2025
Size of extra inode fields: 32
Inode checksum: 0x46374c04
EXTENTS:
(0):33280
```

```
$ hexdump -C -s $(( 33280 * 4096 )) -n 4096 /dev/sda1
```

Wo liegt nun die Datei?



```
08200000  89 50 4e 47 0d 0a 1a 0a  00 00 00 0d 49 48 44 52  |.PNG........IHDR|
08200010  00 00 00 17 00 00 00 1e  08 06 00 00 00 c7 25 85  |..............%.|
08200020  68 00 00 00 09 70 48 59  73 00 00 0b 13 00 00 0b  |h....pHYs.......|
08200030  13 01 00 9a 9c 18 00 00  00 07 74 49 4d 45 07 e9  |..........tIME..|
08200040  03 12 05 16 1d 39 7a 1e  cb 00 00 02 cd 49 44 41  |.....9z......IDA|
08200050  54 48 c7 cd 95 a1 6f e3  48 14 87 bf e9 2e 18 b3  |TH....o.H.......|
08200060  09 a8 34 86 03 27 cc 65  2e f4 b2 2d 4b 99 0b 73  |..4..'.e...-K..s|
08200070  ac e4 a4 c2 90 03 f7 27  dc b1 2e dc 63 7b ac 85  |.......'....c{..|
08200080  0b 13 96 b2 9a d5 e0 4e  6a 98 0d 56 f2 80 48 6f  |.......Nj..V..Ho|
08200090  81 e3 34 6d e2 36 1b ad  74 f7 a4 91 ec b1 e6 7b  |..4m.6..t......{|
082000a0  e3 37 bf df 1b f8 3f 84  f7 5e 00 01 64 f5 fc 73  |.7....?..^..d..s|
082000b0  c2 18 b3 06 77 c3 39 f7  53 12 48 df 58 25 ed 8d  |....w.9.S.H.X%..|
082000c0  a3 3d 4a b1 15 79 9e 53  55 15 17 17 17 87 6f 59  |.=J..y.SU.....oY|
082000d0  6b bd 2e 81 b5 56 00 19  8d 46 52 55 95 58 6b c5  |k....V...FRU.Xk.|
082000e0  5a 2b cd fc 8f de dd ab  3d 4a 82 73 8e 34 4d 89  |Z+......=J.s.4M.|
082000f0  a2 88 8f 49 c5 e8 7c cc  9f 5f 4a 2e 07 bf 83 5d  |...I..|.._J....]|
08200100  81 3e 2c d4 41 f0 2e ae  2f 3d e3 8f 06 e2 18 cc  |.>,.A.../=......|
08200110  09 14 37 50 ce 20 36 e0  4e 81 04 c2 0c 75 fa 55  |..7P. 6.N....u.U|
08200120  01 bc 7f a3 e6 14 45 b1  7e 9f cd 0b 86 18 52 77  |......E.~.....Rw|
08200130  07 71 37 af a1 aa a1 ba  05 6e 21 3c ad 7f f7 1a  |.q7......n!<....|
08200140  fc f8 f8 f8 b7 e5 72 89  d6 9a 10 02 77 ff c2 f4  |......r.....w...|
08200150  9f 76 7d a8 17 b8 f7 df  60 19 a0 06 be 01 4b 8d  |.v}.....`.....K.|
08200160  fa 75 a9 f6 36 8e 73 4e  76 e9 5c 83 5c 5f 65 72  |.u..6.sNv.\.\_er|
08200170  b0 23 3b 85 bc a6 f3 1f  76 ab f7 7e 6b b7 c6 18  |.#;.....v..~k...|
08200180  49 92 44 3a 79 02 92 24  89 58 6b 7f 2c c1 2e 70  |I.D:y..$.Xk.,..p|
08200190  9e e7 92 65 d9 b3 fe 32  9f cf e5 fe fe 5e b4 d6  |...e...2.....^..|
082001a0  fb 25 d8 6c 50 9b e0 24  49 d6 73 d6 5a 99 4c 26  |.%.lP..$I.s.Z.L&|
082001b0  d2 34 8d 88 34 32 1e 8f  df 6e 68 2f c1 80 e4 79  |.4..42...nh/...y|
082001c0  2e 69 9a 3e 1d a4 d6 92  65 99 dc dc dc 88 48 0b  |.i.>....e.....H.|
082001d0  9f 4e a7 eb 72 ed 4c d0  d9 b9 b3 7a 9a a6 5b e0  |.N..r.L....z..[.|
082001e0  ae 15 4c 26 13 79 78 78  58 c3 9b a6 91 d1 68 b4  |..L&.yxxX.....h.|
082001f0  05 5d 9b 68 b1 58 a8 2e  ab f7 9e 10 02 b3 d9 8c  |.].h.X..........|
08200200  b2 2c 37 cf 02 ef 3d c3  e1 f0 45 0f 6a 5b 44 2f  |.,7...=...E.j[D/|
08200210  1c a0 28 0a 05 c8 a6 2b  37 63 30 18 e0 9c c3 18  |..(....+7c0.....|
08200220  b3 97 30 8e 76 59 be a7  43 62 8c 21 8e e3 9d df  |..0.vY..Cb.!....|
08200230  77 cd 6f c1 fb 76 ad b5  26 8e 63 8c 31 84 10 0e  |w.o..v..&.c.1...|
08200240  83 bf d1 df d1 5a 03 10  42 60 33 47 d3 34 87 c3  |.....Z..B`3G.4..|
08200250  3b 70 14 45 68 ad a9 eb  9a b2 2c db 04 a1 64 3c  |;p.Eh.....,...d<|
08200260  fc 0b 99 3f 57 cc 56 cb  75 ce 3d 53 48 5f 92 a6  |...?W.V.u.=SH_..|
08200270  69 da e7 af 2b 95 0c 62  a8 4a e4 8b 11 34 a8 b3  |i...+..b.J...4..|
08200280  5a 1d 6d 5e 67 de 7b d9  f5 7b 5d 19 ba 88 a2 88  |Z.m^g.{..{].....|
08200290  28 8a 30 d5 a7 27 70 03  84 76 ad 3a ab 9f 2e 8b  |(.0..'p..v.:....|
082002a0  6e 61 df 61 76 51 d7 0b  34 8f 18 02 be ba 5a 81  |na.avQ..4.....Z.|
082002b0  87 d0 54 50 97 c0 29 ea  fc 56 bd 2c 8b 5a 59 7b  |..TP..)..V.,.ZY{|
082002c0  a7 12 5a 70 4d 59 3e 52  17 9f b9 2b c0 3b c0 c6  |..ZpMY>R...+.;..|
082002d0  50 de 43 13 c0 9d a3 3e  7c 52 7d 07 aa 42 08 ea  |P.C....>|R}..B..|
082002e0  fa 2a 63 e4 59 ab e2 99  07 28 70 8f 05 b9 b9 6b  |.*c.Y....(p....k|
082002f0  af b6 a2 00 7d 82 fa a5  56 2f c1 af 5e d0 d3 4b  |....}...V/..^..K|
08200300  23 d3 c7 c0 b4 68 ff e4  d4 6b 2e 4f 02 5a 03 da  |#....h...k.O.Z..|
08200310  80 cb 50 67 7f 2b fe ab  f8 0e cc 03 89 75 ae 1f  |..Pg.+.......u..|
08200320  e9 38 00 00 00 00 49 45  4e 44 ae 42 60 82 00 00  |.8....IEND.B`...|
08200330  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
08201000
```

Und große Dateien?

```
$ ls -li ./file_of_randomness.random
415 -rw-r--r-- 1 uid gid 419430400 18. Mar 06:22 file_of_randomness.random
$ echo "stat <415>" | debugfs /dev/sda1
Inode: 415   Type: regular    Mode:  0644   Flags: 0x80000
Generation: 1319619711    Version: 0x00000000:00075239
User:     0   Group:     0   Project:     0   Size: 419430400
File ACL: 0
Links: 1   Blockcount: 819200
Fragment:  Address: 0    Number: 0    Size: 0
 ctime: 0x67db2bbc:ca0d4678 -- Wed Mar 19 06:53:19 2025
 atime: 0x67db2bbc:cc3294ec -- Wed Mar 19 06:53:19 2025
 mtime: 0x67db2bbc:ca0d4678 -- Wed Mar 19 06:53:19 2025
crtime: 0x67db3bbc:627838e8 -- Wed Mar 19 06:45:20 2025
Size of extra inode fields: 32
Inode checksum: 0x5a49933f
EXTENTS:
(0-30719):34816-65535, (30720-59391):69632-98303,
(59392-92159):100352-133119, (92160-102399):133120-143359
```

By the way...

… we are hiring

jobs.pengutronix.de

Vielen Dank

Fragen?