

UNIX use

Hans Buchmann FHNW/IME

24. September 2014

Ziel

Entwicklung von Programmen auf dem **RaspberryPi**

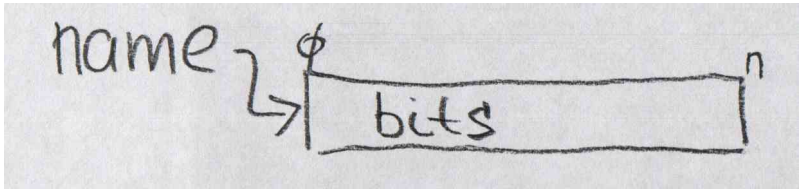
- ▶ alles ist ein File
 - ▶ 0 – te Näherung
- ▶ Filesysteme
 - ▶ mount
 - ▶ sshfs
- ▶ Cross development
 - ▶ *Host* ↔ **RaspberryPi**

Remark: Keine Toolchain auf dem **RaspberryPi**

Wichtig

- ▶ wo ist was
- ▶ Verzeichnisstruktur

Alles ist ein File



- ▶ *name* Referenz auf die Bits (Bytes)
 - ▶ Bits(bytes) der Reihe nach
- ▶ Files
 - ▶ Datenquelle
 - ▶ liefern Daten: Bits(Bytes)
 - ▶ Datensenke
 - ▶ absorbieren Daten: Bits(Bytes)

Ein paar Befehle

- ▶ `cat name`
- ▶ `hexdump -C name`

Devices sind auch Files

z.B. SD-Karte

`/dev/mmcblki` $i = 0, 1, 2 \dots$

Remark: Name vom Betriebssystem bestimmt

Datenquelle `hexdump -C /dev/mmcblk0`

Datensenke `cp name /dev/mmcblk0`

Remark: Aufpassen

Devices sind auch Files

z.B Zufallszahlen

`/dev/random` sammelt das Rauschen: langsam

Remark: Name vom Betriebssystem bestimmt

Datenquelle `hexdump -C /dev/random`

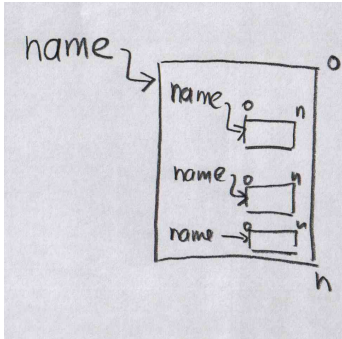
`/dev/urandom` berechnete (Pseudo) Zahlen: schnell

Datenquelle `play -b 16 -e signed-integer \`
`-t raw -r 44000 /dev/urandom`

Remark: der Befehl `play` hat viele Optionen

Filesystem

Files für Files



- ▶ File der weitere Files enthält

- ▶ Verschiedene Filesysteme

`vfat` Microsoft

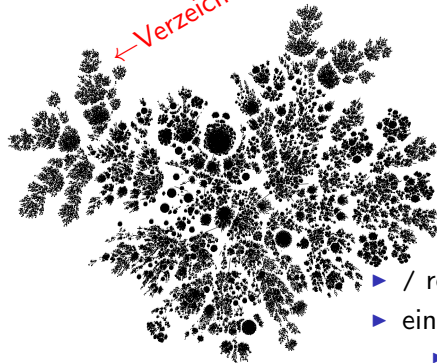
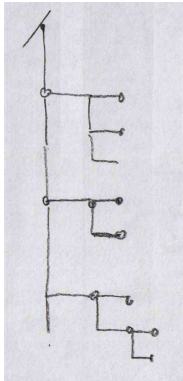
`ext4` UNIX

... noch viele andere

`cat /proc/filesystems`

Vereichnisstruktur

Hierarchie

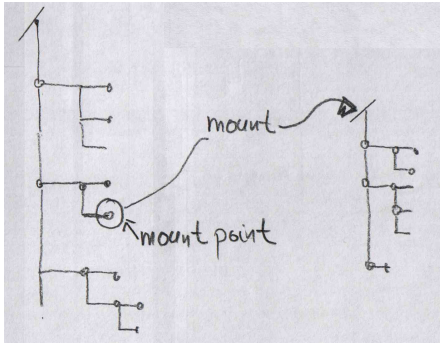


← Verzeichnisse einer Workstation

- ▶ / root
- ▶ ein paar Befehle
 - ▶ ls, tree, cd

`mount fileSystem mountPoint`

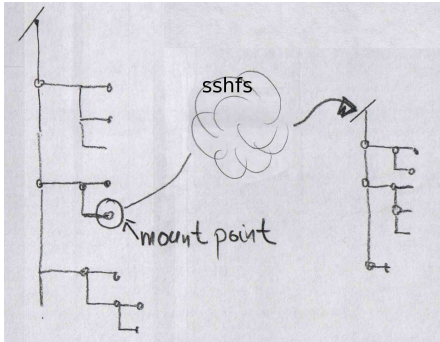
Verbindet Filesysteme



► `mount /dev/mmcblk0p1 \`
`mountPoint`

Remark: Sieht wie ein normales Verzeichnis aus

sshfs user@host mountPoint
via ssh



► braucht ssh

Remark: Sieht wie ein normales Verzeichnis aus

Verzeichnisstruktur

Host RaspberryPi

Host

```
devel ..... somewhere on the host
├── config
│   └── Makefile ..... for making RaspberryPi executables
├── java ..... source
├── src ..... c c++
├── tc ..... normally a link
└── work ..... connected with RaspberryPi current dir
```

RaspberryPi

```
user ..... somewhere on the RaspberryPi
└── work ..... connected with Host current dir
```

Aufgaben

- ▶ Verbindung mit **RaspberryPi** via ssh
- ▶ upgrade pacman -Suy
- ▶ *user* auf **RaspberryPi** useradd
- ▶ *toolchain* auf *Host*
- ▶ mount *Host* auf **RaspberryPi** mit sshfs
- ▶ HelloWorld.java, hello-world-cpp.cc, hello-world-c.c,
primes.cc
auf **RaspberryPi**