

# Hardware/Devices

Hans Buchmann FHNW/IME

21. Dezember 2016

# Devices

- ▶ Driver für Devices
  - ▶ Vereinfacht den Zugriff
  - ▶ alles ist ein File *Stream of Bits*
  - ▶ gehören zum *kernel*
  - ▶ Verbindung *userspace* ↔ *kernel-space*
- ▶ Userspace
  - ▶ komfortabler Zugriff via *Driver* auf die *Devices*
  - ▶ alles ist ein File *Stream of Bits*
  - ▶ `mmap` Verschiedene Arten von Memory

# Kernelspace

- ▶ Kernelspace
  - ▶ extrem privilegierter Zugriff
  - ▶ *kernelmodules*
  - ▶ Heimat der *Driver*
- ▶ Devicetree
  - ▶ Versuch einer Normierung
  - ▶ wo ist was

# GNU/Linux

Befehle: funktionieren nicht alle auf **BBB**

- ▶ dmesg
  - ▶ Meldungen von *kernel*
- ▶ ls\*
  - ▶ lspci, lsblk, lscpu ...
- ▶ hwdb
- ▶ ...

## GNU/Linux Verzeichnisse

- ▶ `/proc`
  - ▶ `/proc/iomem`: wo ist die Hardware
  - ▶ `/proc/devices`: die *major* Nummern
  - ▶ `/proc/config.gz`: die aktuelle *kernel* Konfiguration
- ▶ `/dev` Devices Namen
- ▶ `/sys` Devices im Detail
  - ▶ ziemlich komplex

## Um was geht es ?

### Zugriff auf mehrere Arten

- ▶ userspace
  - ▶ per `/sys/class/gpio`
  - ▶ per `mmap` mit eigenem Programm:
- ▶ kernel-space
  - ▶ mit eigenem module

# Die Aufgabe

die LEDs  $USR\{0,1,2,3\}$  setzen/löschen

## Die Informationen

- ▶ Schema & Info
  - ▶ `doc/beaglebone-black/BBB*.pdf`
- ▶ Prozessor System On Chip
  - ▶ AM335x Sitara Processors (Rev. L)  
`doc/beaglebone-black/spruh731.pdf`

# Die Komponente GPIO1

In AM335x Sitara Processor

- ▶ MemoryMap:
  - ▶ Abschnitt 2.1 ARM Cortex-A8 Memory Map
- ▶ Beschreibung
  - ▶ Abschnitt 25



# U-Boot

## die Befehle

- ▶ `mw` Memory Write
- ▶ `md` Memory Display
- ▶ `tools/u-boot.cmd`

## sys/class/gpio

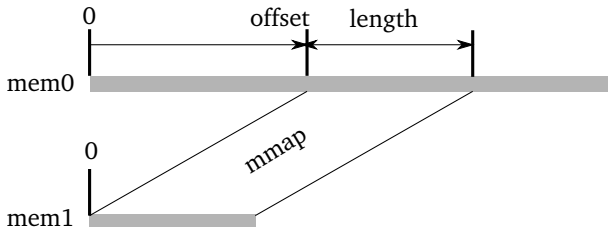
- ▶ kernel/Documentation/gpio.txt
- ▶ Gute Pins:
  - ▶ USR{0,1,2,3} auf GPIO1\_{21,22,23,24}
  - ▶ *kernel* anpassen: *kein* LED support

## mmap

### Direkter Zugriff auf die Hardware

- ▶ Beschreibung
  - ▶ Siehe Abschnitt *Direkt*
- ▶ Wo im Speicher
  - ▶ /proc/iomem
- ▶ Der wichtige Aufruf
  - ▶ `mmap`

# mmap



```
mem=(unsigned char*)mmap(0, /* addr hint */
    length,
    PROT_READ|PROT_WRITE,
    MAP_SHARED,
    memId,
    offset );
```

# Aufgaben

- ▶ Der Code `src/led-demo.c`

# Kernel**space**

## Kernelmodules

- ▶ 6-modules
- ▶ TODO