

# Kernel

Hans Buchmann FHNW/IME

9. Oktober 2018

## Ziele

### Neuer **kernel** auf **BeagleBoneBlack**

- ▶ Download
- ▶ Setup
- ▶ Konfiguration
- ▶ Kompilation
- ▶ Installation

# The Big Picture

## grosses Projekt

**Gegeben** Eine grosse Anzahl *source* Files

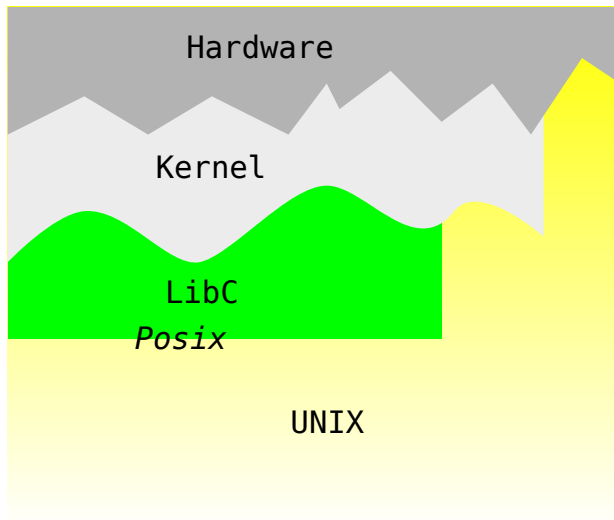
**Gesucht** 2 Files:

- ▶ das *Image*
- ▶ der *Devicetree*

**Lösung** Klassisches Verfahren

- ▶ Toolchain
- ▶ Makefile

## Die Schichten



## Kernel Grosses Projekt

### Was ist einfach ?

- ▶ **kernel** hängt nicht von anderen Software Komponenten ab
  - ▶ stand alone
- ▶ Braucht nur `make` und *toolchain*

### Was ist schwierig ?

- ▶ Konfiguration
  - ▶ Wahl der richtigen *source* Files für das **Image**

[github.com/beagleboard/linux](https://github.com/beagleboard/linux)

Mehrere Möglichkeiten

- ▶ das ganze git repository
- ▶ nur die letzten  $n$  Versionen `--depth= $n$`
- ▶ zip File

## Tools

`toolchain` [sourceforge.net/projects/fhnw-tinl/files](https://sourceforge.net/projects/fhnw-tinl/files)

- ▶ `tc-tinl-gcc-8.1.0-2018.05.21.tar.gz`
  - ▶ auf [drive.switch.ch/index.php/s/A6H382zEGDrgfAL](https://drive.switch.ch/index.php/s/A6H382zEGDrgfAL)
- ▶ Prefix: `arm-linux-gnueabihf-`
  - ▶ beschreibt:
    - ▶ Architektur: `armv7`
    - ▶ **A**pplication **B**inary Interface: `gnueabihf`

`make` Normales `make`

- ▶ **kernel** Herstellung:
  - ▶ `make cmd`

## Wo ist was ?

```
tinL
├── 5-kernel
│   ├── build ..... generated kernel files
│   │   ├── .config ..... die aktuelle Konfiguration
│   ├── tools ..... for making
│   │   ├── kernel.sh ..... wrapper to kernel Makefile
│   ├── config
│   │   ├── config.sh ..... for kernel.sh
│   │   ├── kernel.config ..... 'gute' kernel Konfiguration
│   ├── tc ..... link to toolchain
├── resources
│   ├── beaglebone-black
│   │   ├── linux ..... the source tree
```



## Erste Konfiguration

- ▶ Hilfe
  - ▶ `./tools/kernel.sh help`
- ▶ Vordefinierte Konfiguration
  - ▶ `./tools/kernel.sh bb.org_defconfig`
- ▶ Anpassung der Konfiguration
  - ▶ `./tools/kernel.sh menuconfig`
  - ▶ `./tools/kernel.sh xconfig`

# Kompilation

- ▶ `./tools/kernel.sh zImage`
  - ▶ erzeugt `build/arch/arm/boot/zImage`
- ▶ `./tools/kernel.sh dtbs`
  - ▶ erzeugt `build/arch/arm/boot/dts/am335x-boneblack-wireless.dtb`  
*Devicetree*

Remark: *Devicetree* später behandelt

# Installation auf SD-Card

## ► Kopiere

**Image** `build/arch/arm/boot/zImage`

**Devicetree** `build/arch/arm/boot/dts/am335x-boneblack-wireless.dtb`  
auf

## ► SD-Card Partition 2:Root File System

```
/ ..... Partition 2
├── boot
│   ├── zImage
│   └── am335x-boneblack-wireless.dtb
```

## Workflow

### schrittweise Herstellung

#### 0 Setup der Toolchain

#### 1 Default Konfiguration (falls vorhanden)

▶ `sh tools/kernel.sh bb.org_defconfig`

#### 2 Herstellung

▶ `tools/kernel.sh zImage`

#### 3 Transfer/Start/Test auf **BeagleBoneBlack**

▶ U-Boot

#### 4 (Re)Konfiguration

▶ `sh tools/kernel.sh menuconfig`

→ 2 `ev. cp build/.config config/kernel.config`

# Workflow

## ▶ **BeagleBoneBlack** *default* Konfiguration

- ▶ herstellen
- ▶ auf SD-Karte
- ▶ ausprobieren

## ▶ Die *default* Konfiguration ändern:

- ▶ Internet über USB:

Device Drivers

└─ USB support

└─ USB Gadget Support

└─ USB Gadget Drivers

- ▶ nur eine CPU
- ▶ keine ALSA Soundkarte
- ▶ ...

## Die beteiligten Files

- ▶ `zImage` der *kernel*
- ▶ `am335x-boneblack-wireless.dtb` der *device tree*