

# Integrationsseminar und Projekt: IoT und Embedded-Workshop mit Raspberry Pi

- Embedded und IoT-Devices im Alltag
- Kurzer Einblick in die Geschichte
- Vorstellung gängiger Hardwareplattformen
- Fallbeispiel für die Nutzung des Raspberry Pi
- Grobe Architektur einer Linux-basierten Firmware

<https://else.dhbw-karlsruhe.de/moodle/course/view.php?id=4719>

# Embedded und IoT-Devices im Alltag



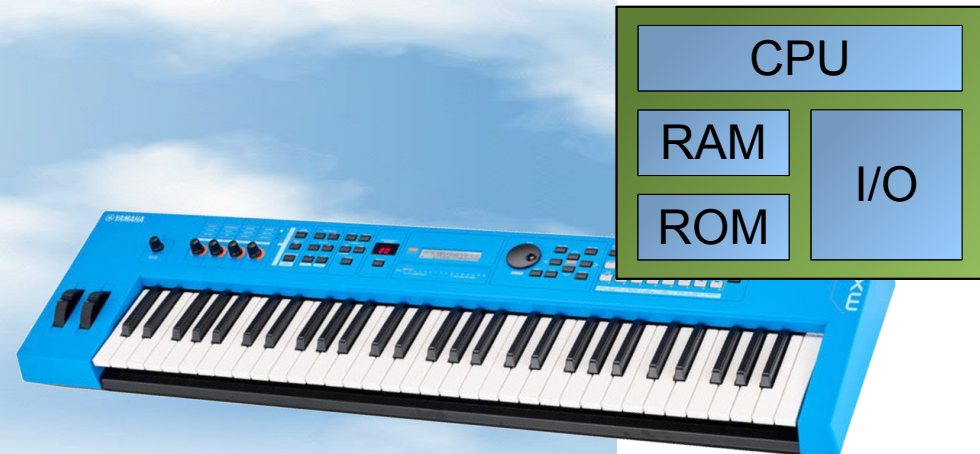
# Defintion: Embedded und IoT-Devices

## Eingebettete Systeme

- Kleine Computersysteme innerhalb eines größeren Geräts
- Hoch spezialisierte, auf die Anwendung zugeschnittene Hardware
- Von kleinen Micro-Controllern bis hin zu leistungsfähigen SoC
- Können steuernde Aufgaben besitzen oder einem Gerät „Leben einhauchen“

## Internet of Things-Devices

- Eingebettete Systeme mit permanenter Internetverbindung
- Tauschen untereinander und mit Servern im Internet Daten aus
- Messen häufig verschiedene Umweltfaktoren (sog. Sensornetze)





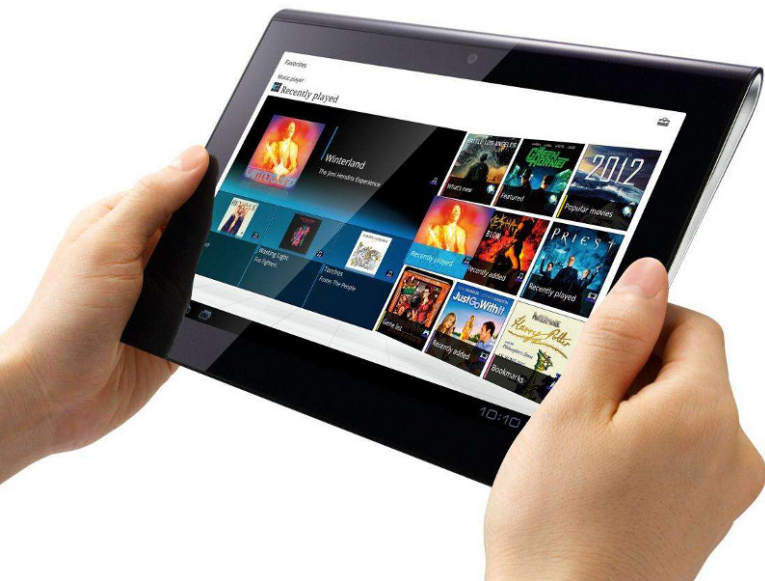
# Micro-Controller vs. System-On-A-Chip

## Micro-Controller



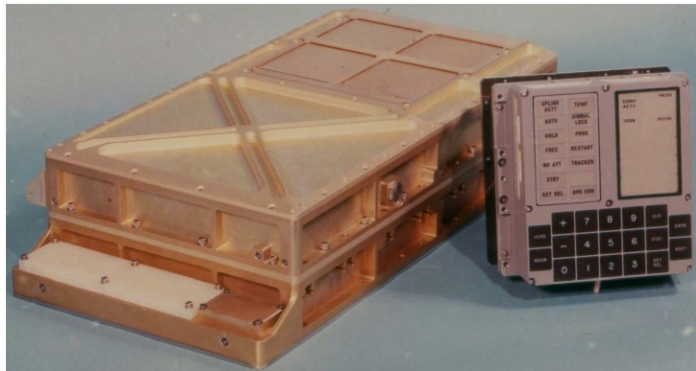
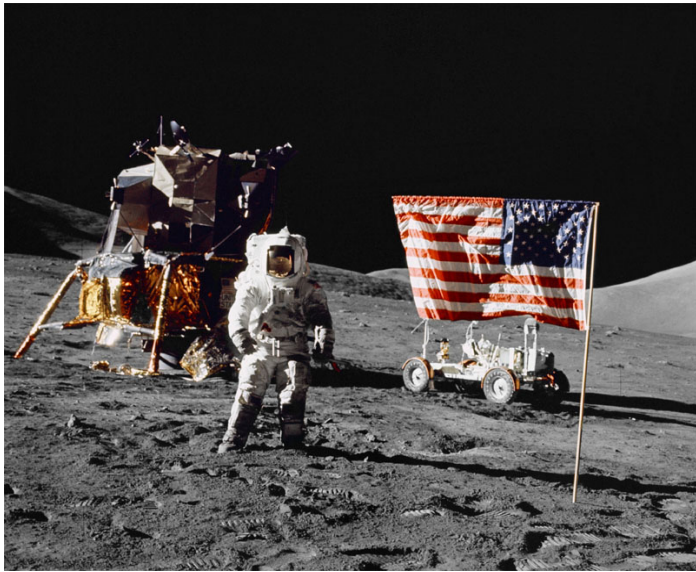
- Einfache CPUs mit wenigen MHz
- In der Regel nur 8 oder 16-Bit
- Wenige kB selten MB Hauptspeicher
- Programmcode ebenso nur wenige kB
- Direkte Programmierung der Hardware ohne Betriebssystem

## System-On-A-Chip



- Sehr leistungsfähige Hardware
- 32-Bit oder 64-Bit CPUs
- Single Core, Dual Core, Quad Core, ...
- Mehrere MHz bis GHz Rechenleistung
- Mehrere MB oder auch GB RAM
- Umfangreiche Peripherie

# Eine kleine Computer-Geschichte



**21. Juli 1969:** Die ersten Menschen auf dem Mond

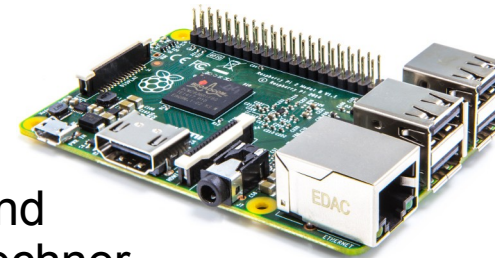
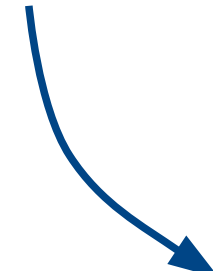
Unterstützt wurden sie vom Apollo Guidance Computer.



**1970er:** „Mini-Computer“



**1980er:** Home Computer  
Unten, der erste Macintosh



**2012:** Raspberry Pi und ähnliche Einplatinenrechner

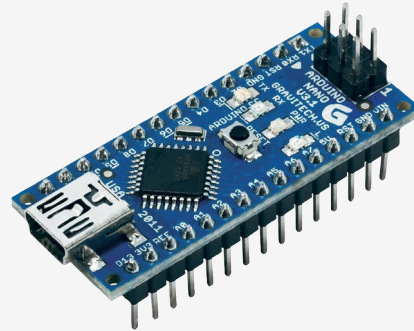


# Aktuelle Hardware-Plattformen



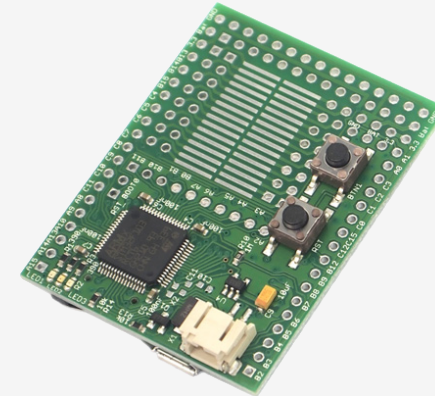
**Raspberry Pi**

32-Bit oder 64-Bit ARM-CPU  
 QuadCore mit bis zu 1,5 GHz  
 1 GB Hauptspeicher  
 USB, LAN, HDMI, WIFI, ...



**Arduino**

Atmel AVR 8-Bit CPU  
 16 MHz, Single Core :-)  
 32 kB Programmspeicher  
 2 kB Hauptspeicher



**Espruino**

32-Bit ARM-CPU  
 72 MHz, Single Core  
 256 kB Flash, 48 kB RAM  
 Hardwareseitiges JavaScript



Boardcon

Arndale Board

Dragon Board

VIA Springboard

**BeagleBone**

BananaPi

C.H.I.P

Board

Hackberry

Orange Pi

Espruino Raspberry Pi

CubieBoard

pcDuino

Gizmo Board

nanoPi

**Arduino**

Intel Galileo

MarsBoard

**ODROID**

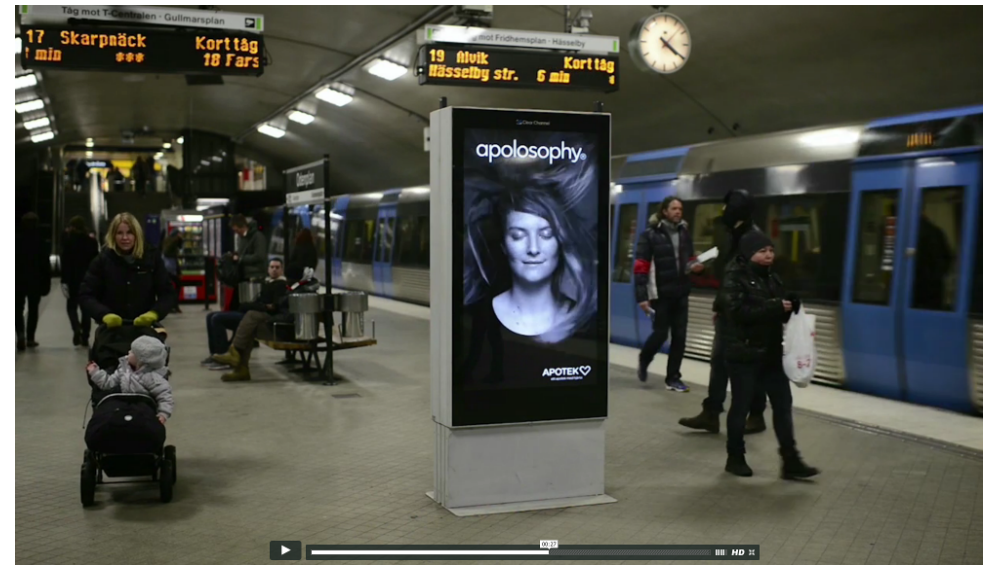
Go Warrior

Ouya

Cosmic + Board

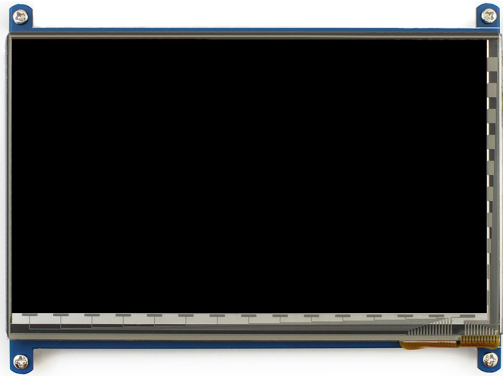
GameStick

# Fallbeispiel zum Raspberry Pi

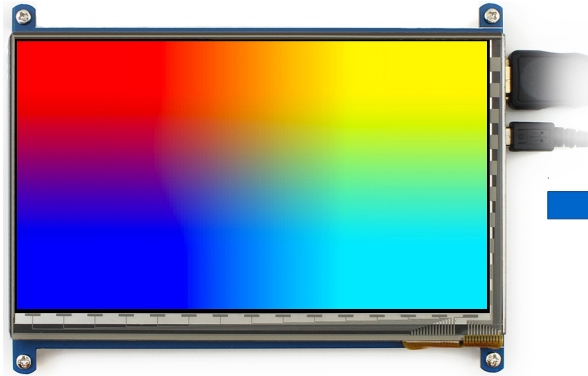


[https://blog.adafruit.com/2014/02/28/wonderful-subway-ad-shows-a-womans-hair-blowing-around-when-ever-a-train-arrives-raspberry\\_pi-piday-raspberrypi/](https://blog.adafruit.com/2014/02/28/wonderful-subway-ad-shows-a-womans-hair-blowing-around-when-ever-a-train-arrives-raspberry_pi-piday-raspberrypi/)

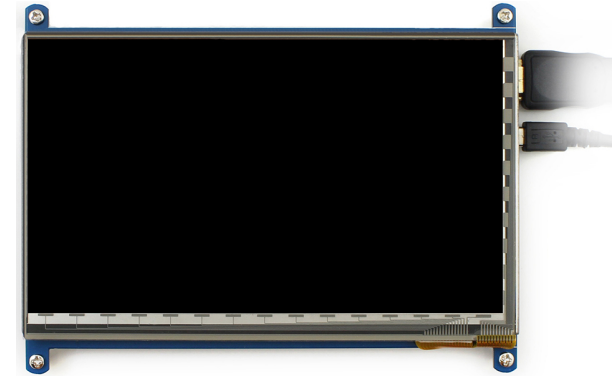
# Der Raspberry Pi-Bootvorgang



1st Stage Boot Loader  
Fest im ROM



2nd Stage Boot Loader  
bootcode.bin, läuft auf der GPU



3rd Stage Boot Loader  
Datei start.elf und config.txt



Linux Kernel  
Prüft die Hardware

**Initial RAM Disk**  
/linuxrc  
Hängt das Root FS ein

**Root Filesystem**  
/sbin/init  
Startet alle anderen  
Prozesse des Systems



Last but not least login oder irgend ein  
anderes Programm



# Grober Aufbau einer Linux-Firmware



Linux Kernel Binary  
kernel.img

## Initial RAM Disk

```
drwxr-xr-x 10 root root 4096 May 7 02:48 .
drwxr-x--- 15 root root 4096 May 7 00:54 ..
drwxr-xr-x 2 root root 4096 May 7 02:48 bin
drwxr-xr-x 2 root root 4096 May 7 02:48 dev
drwxr-xr-x 4 root root 4096 May 7 02:48 etc
-rwxr-xr-x 1 root root 812 May 7 02:48 linuxrc
lrwxrwxrwx 1 root root 3 May 7 02:48 sbin -> bin
drwxr-xr-x 2 root root 4096 May 7 02:48 sys
```

Minimales Dateisystem innerhalb des Kernels

## Kernel

## Userland

### Root File System

```
lrwxrwxrwx 1 root root 7 30. Sep 2015 bin -> usr/bin
drwxr-xr-x 4 root root 4096 19. Nov 17:08 boot
drwxr-xr-x 21 root root 3640 20. Nov 21:37 dev
drwxr-xr-x 107 root root 4096 19. Nov 17:50 etc
drwxr-xr-x 5 root root 4096 21. Mai 2016 home
lrwxrwxrwx 1 root root 7 30. Sep 2015 lib -> usr/lib
drwx----- 2 root root 16384 14. Dez 2014 lost+found
drwxr-xr-x 3 root root 4096 14. Dez 2014 media
drwxr-xr-x 2 root root 4096 29. Dez 2014 mnt
drwxr-xr-x 7 root root 4096 3. Okt 13:31 opt
dr-xr-xr-x 282 root root 0 20. Nov 13:00 proc
drwxr-x--- 15 root root 4096 8. Jul 16:25 root
drwxr-xr-x 29 root root 780 20. Nov 13:01 run
lrwxrwxrwx 1 root root 7 30. Sep 2015 sbin -> usr/bin
drwxr-xr-x 4 root root 4096 25. Okt 2014 srv
dr-xr-xr-x 13 root root 0 20. Nov 13:00 sys
drwxrwxrwt 19 root root 540 21. Nov 05:37 tmp
drwxr-xr-x 14 root root 4096 18. Okt 23:34 usr
drwxr-xr-x 14 root root 4096 8. Jul 16:29 var
```

Besteht aus dem eigentlichen Dateisystem mit den Bibliotheken, Systemprogrammen und allen anderen Dateien der Firmware