

Hochschule Osnabrück  
University of Applied Sciences

# Hardware/Software-Codesign

## Einführung

**Prof. Dr.-Ing. Markus Weinhardt**

*Labor für Digital- und Mikroprozessortechnik*

---

## Übersicht über diese Veranstaltung SS 2017

---

**Vorlesung: 2 SWS (AA 206)**

**Praktikum: 2 SWS = 10 Termine à jeweils 90 Minuten (SB 219)**

Mittwoch 14.30 – 16.00 und 16:30 – 18:00

- Die ersten 3-4 Wochen finden an beiden Terminen Vorlesungen statt, um genügend Grundlagen für das Praktikum zu erarbeiten.
- Die Praktika finden (nach Ankündigung) zu denselben Terminen wie die Vorlesungen statt.

**Prüfung:** „Projektbericht“ (Praktische Arbeit) oder Klausur

**Prof. Dr.-Ing. Markus Weinhardt**

Raum: SB 223 (Artilleriestr. 46)

Tel.: 969 – 3445

Email: [m.weinhardt@hs-osnabrueck.de](mailto:m.weinhardt@hs-osnabrueck.de)

Web: <http://www.ecs.hs-osnabrueck.de/weinhardt.html>

Sprechstunde: nach Vereinbarung

# Was ist Hardware/Software-Codesign?

---

... der integrierte (d. h. gemeinsame und gleichzeitige) Entwurf (*design*) der Hardware- und Software-Komponenten eines digitalen Systems

## Themen:

### → Gemeinsamkeiten und Unterschiede von Hardware und Software

- Abgrenzung zwischen Hardware und Software in einem System
- Hardware/Software-Schnittstellen
- Zielarchitekturen für digitale Systeme

### → Entwurf eines Gesamtsystems

- Ausgangspunkt: Anwendung bzw. Anwendungsklassen
- Entwurfsmethoden und -modelle
- Leistungs-Abschätzung und Evaluation von Entwurfs-Alternativen

# Warum Codesign?

---

## Abgrenzung Hardware/Software

### ▶ Universalrechner (*general purpose systems*), z. B. PCs

- Relativ klare Trennung:
  - Prozessor-Entwurf  $\Leftrightarrow$  Compiler, Betriebssystem  $\Leftrightarrow$  Software

### ▶ Moderne "Eingebettete Systeme" (*embedded systems*)

- bestehen aus kooperierenden Hardware- und Software-Komponenten
- z. B.: Industrie-/Kfz-Steuerungen, Mobiltelefone, Unterhaltungselektronik
- Systementwurf lässt sich nicht mehr strikt in Hardware- und Software-Entwurf trennen:
  - Entwurf von Spezial-Prozessoren  $\Leftrightarrow$  Compiler  $\Leftrightarrow$  Software
  - Funktionalität kann in Hardware *oder* Software implementiert werden
    - Verwendung von Standard- und Spezialprozessoren, Entwurf von Spezial-Hardware
    - Software-Entwicklung (ggf. Treiber, Compiler, Betriebssystem) muss an das spezielle System angepasst werden!

### ▶ Hochleistungsrechner (*Supercomputing/High-Performance Comp.*)

- Verwenden teilweise FPGA-Hardware, um spezielle Funktionen zu implementieren → Hardware-Entwurf, Anpassung der Treiber etc. nötig!

# Warum Codesign?

## Allgemeiner Systemaufbau:



Die Funktionalität des Systems besteht aus einem oder mehreren (kooperierenden) Prozessen. Implementierung eines Prozesses prinzipiell sowohl möglich als

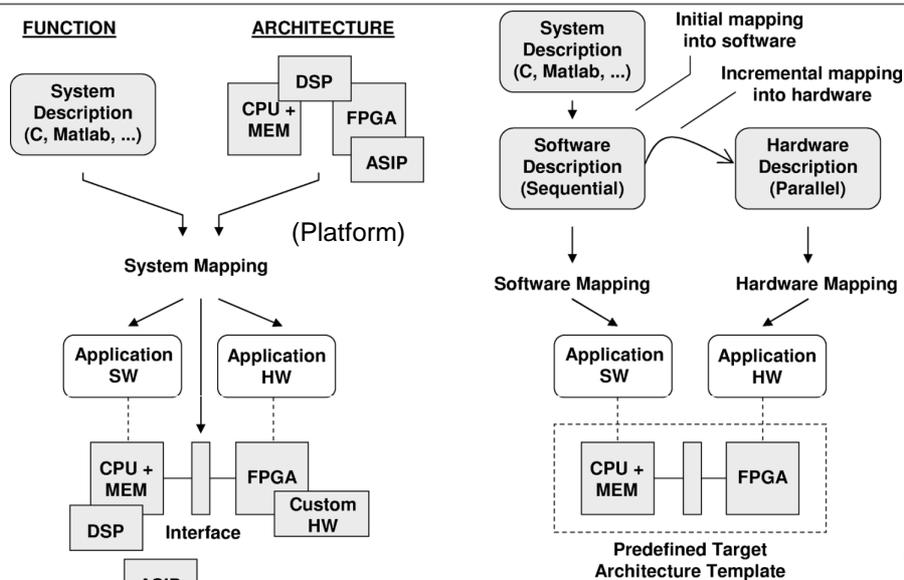
- Dedizierte Hardware (d. h. digitale Schaltung, die nur diese Funktion ausführt)
- oder als Software (d. h. Maschinenprogramm, das auf Standard-Hardware, also einem General-Purpose-Prozessor läuft)

**Trennung in HW und SW ist also relativ willkürlich!**

Entscheidung hängt von den Leistungs-Anforderungen an das System ab.

➔ Optimierung des Gesamtsystems notwendig!

## Verschiedene Herangehensweisen des HW/SW-Codesign



### "Function/Architecture Codesign"

- sehr allgemeine Vorgehensweise
- Überblick über Gesamtlösung nötig
- sehr großer "Entwurfsraum", erfordert gute Kenntnisse aller Prozessortypen (heterogene Zielarchitekturen)

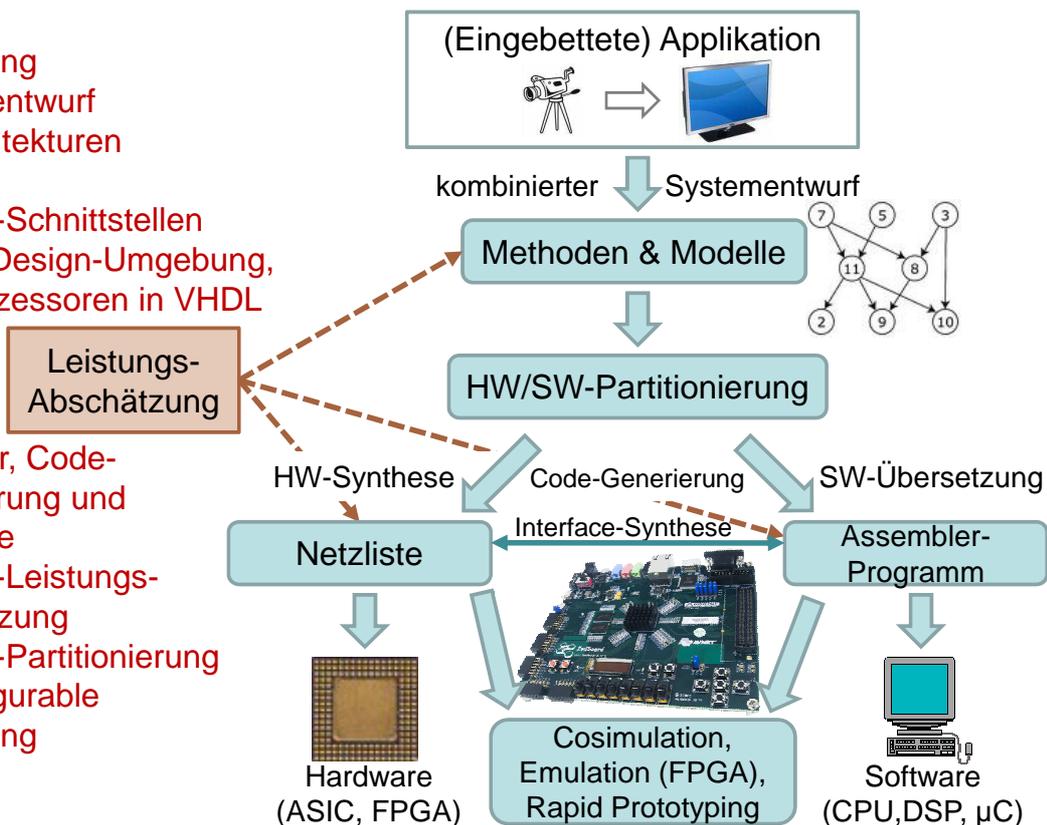
### "Inkrementelles Codesign"

- festgelegte Zielarchitektur (CPU/MEM/FPGA)
- ausgehend von SW-Implementierung wird das System schrittweise optimiert
  - ➔ gut für Übungen/**Praktikum** geeignet!
- **HW-Koprozessoren** (z.B. UART-I/O) entlasten CPU; Gesamtsteuerung durch CPU-Programm

# Hardware/Software-Codesign – Übersicht

## Gliederung:

1. Einführung
2. Systementwurf
3. Zielarchitekturen
4. Modelle
5. HW/SW-Schnittstellen
6. Vivado-Design-Umgebung, HW-Prozessoren in VHDL
7. Compiler, Code-Generierung und Synthese
8. HW/SW-Leistungs-Abschätzung
9. HW/SW-Partitionierung
10. Reconfigurable Computing



## Literaturhinweise

- ▶ [1] J. Teich, Chr. Haubelt: "Digitale Hardware/Software-Systeme – Synthese und Optimierung", 2. Auflage, Springer-Verlag, 2007
- ▶ [2] Chr. Plessl, M. Platzner: Vorlesungs-Skript "Hardware/Software Codesign", <http://homepages.uni-paderborn.de/plessl/lectures/2010-Codesign/script/Skript-Codesign.pdf>
- ▶ [3] P. R. Schaumont: "A Practical Introduction to Hardware/Software Codesign", Springer-Verlag 2010
- ▶ [4] F. Vahid, T. Givargis: "Embedded System Design - A Unified Hardware/Software Introduction", Wiley&Sons, 2002

## Hinweis:

Für die Vorlesung wird ein "Semesterapparat" in der Zentralbibliothek aufgebaut.