

Embedded Systems (ES)

1. Introduction

Prof. Dr. Marco Platzner
Computer Engineering Group

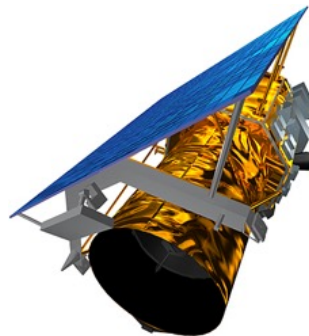


Embedded Systems

- Embedded system = information processing system embedded into a larger product



automotive functions



satellite



hearing aid



industrial control



fax machine / printer



prosthesis control



production robot

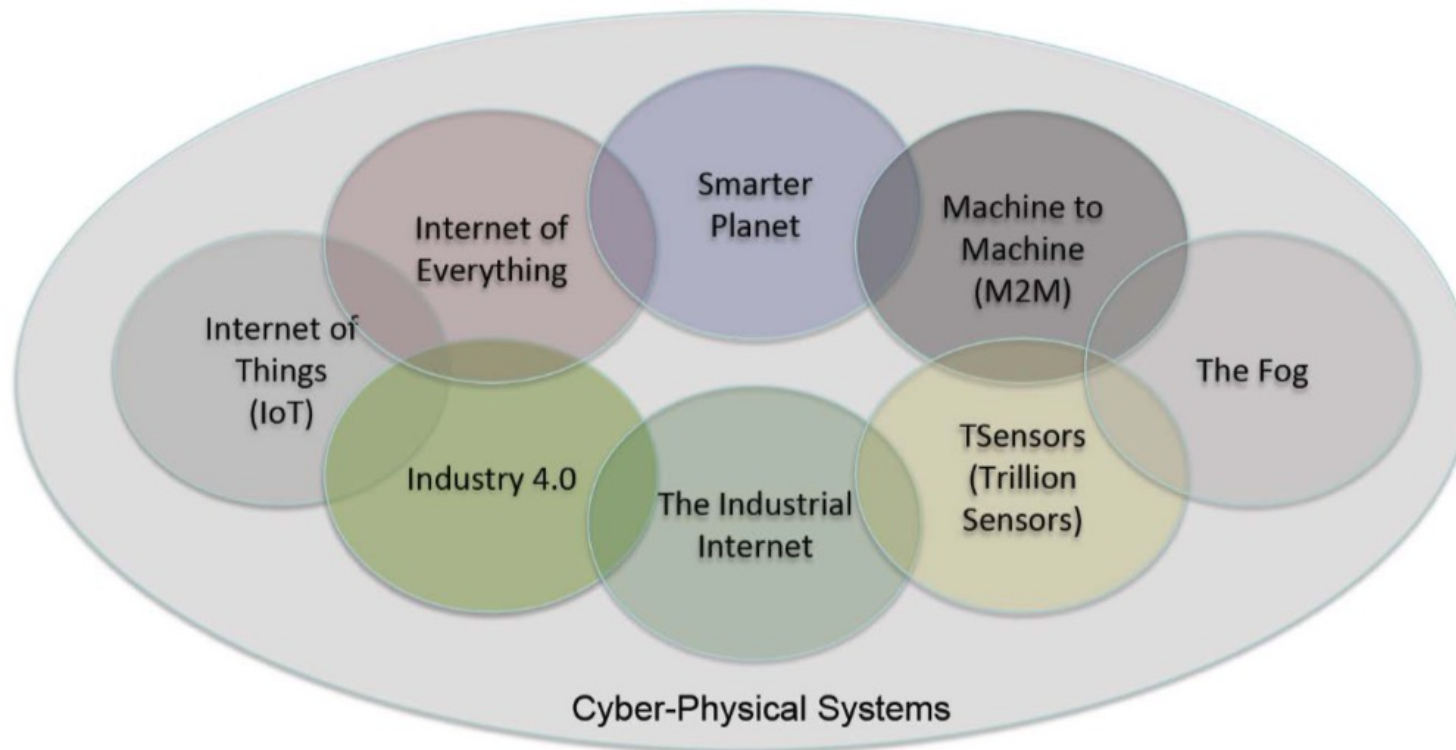


digital tv set



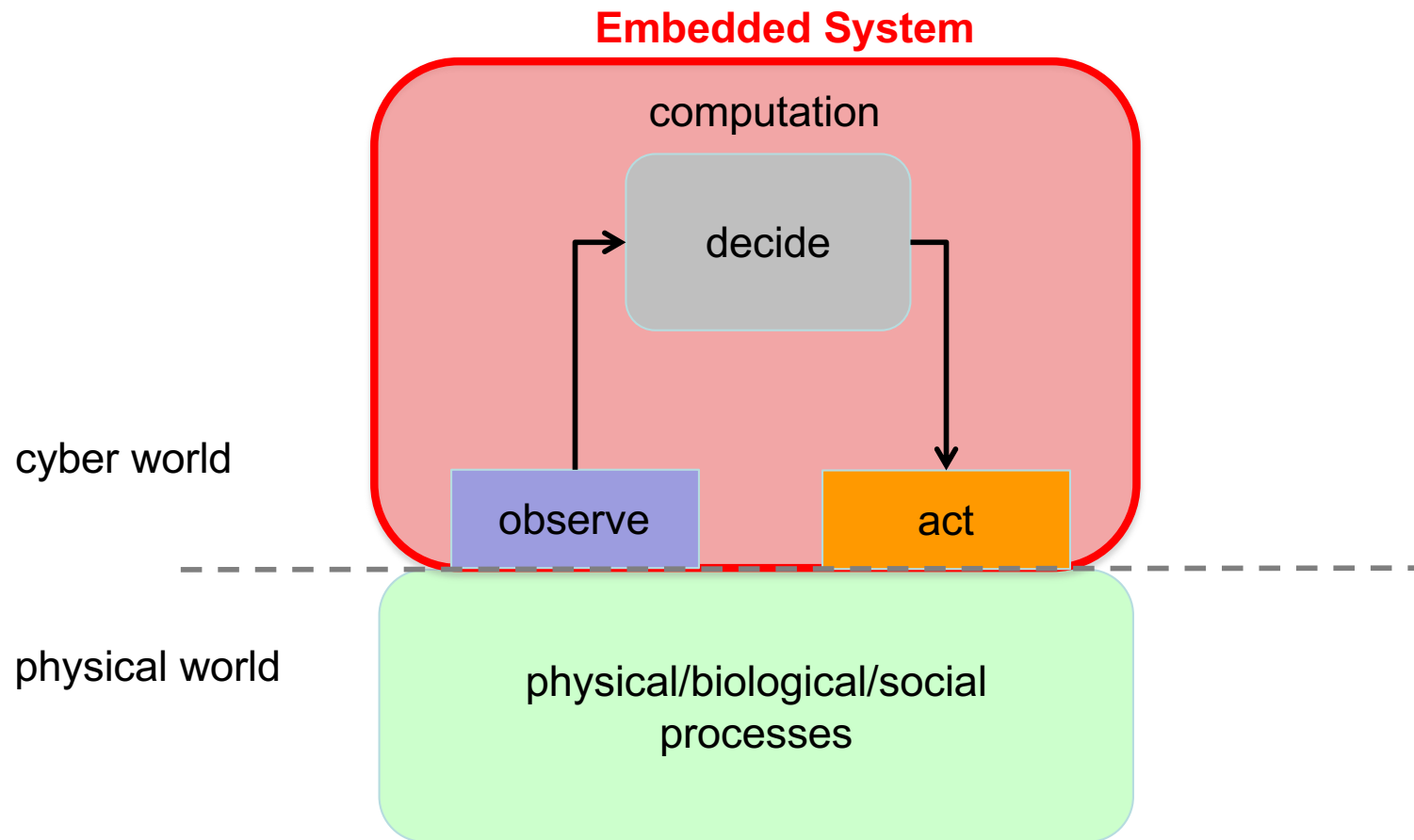
Cyber-Physical Systems

- Cyber-physical system (CPS)
 - integration of **computation and physical processes** [Lee07]
 - often distributed: integration of **computing and communication**



[Lee07] E. A. Lee. Computing Foundations and Practice for Cyber Physical Systems: A Preliminary Report. Technical Report No. UCB/EECS-2007-72. 2007.

Abstract View of an Embedded System

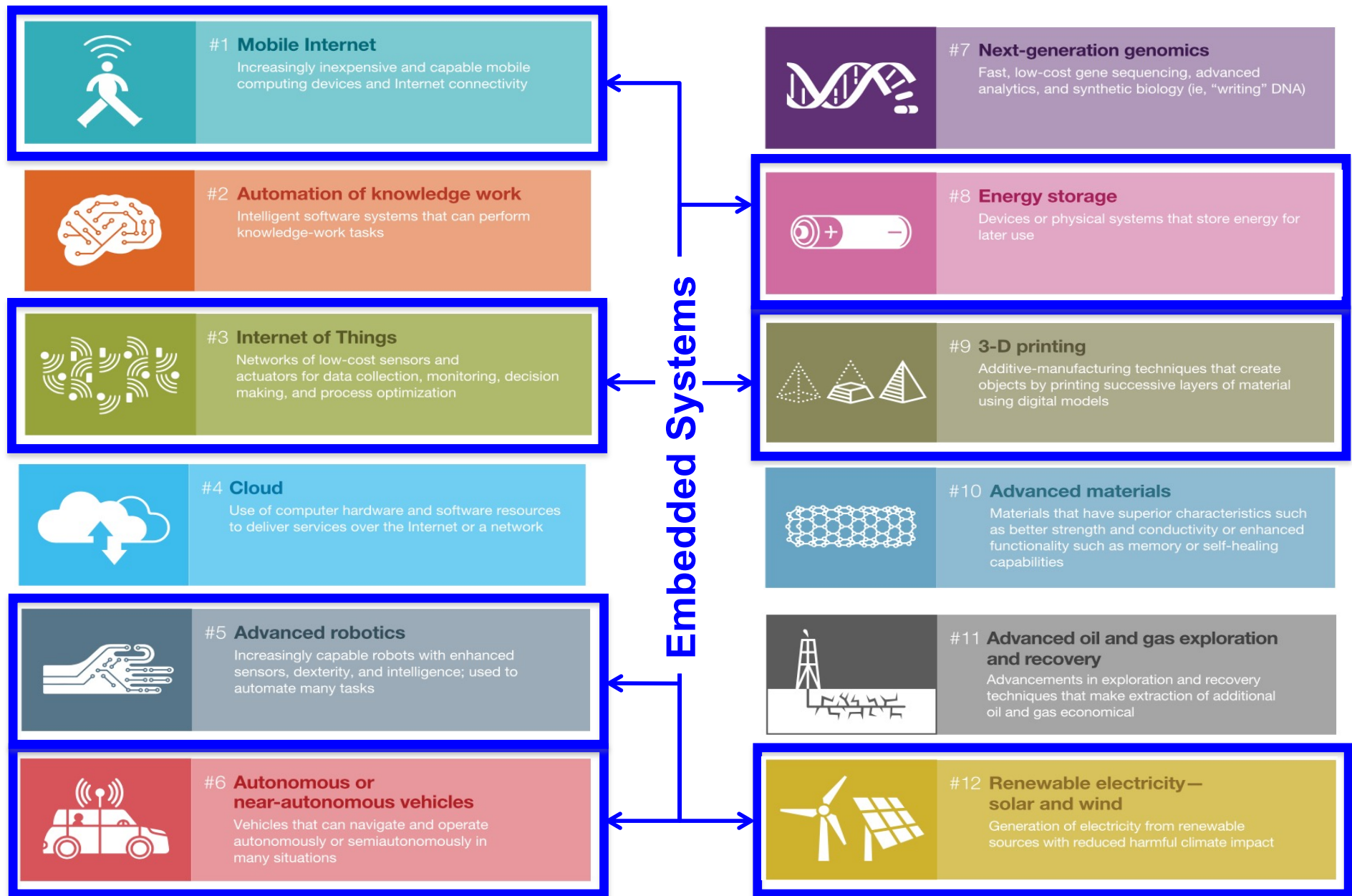


Reactive and Real-time Systems

- Embedded systems are often **reactive**
 - reactive systems must react to stimuli from the system environment
 - “a reactive system is in continual interaction with its environment and executes at a pace driven by that environment” [Ber+96]
- Embedded systems often have **real-time constraints**
 - “a real-time constraint is called hard, if not meeting that constraint could result in a catastrophe” [Kop97]; other real-time constraints are called soft
 - for hard-real time systems ...
 - **guaranteed system response** must be proven, without statistical arguments
 - correct answers that arrive too late are considered wrong

[Ber+96] J.-M. Bergé et al. (Eds.) High-level System Modeling: Specification Languages. Springer. 1995.

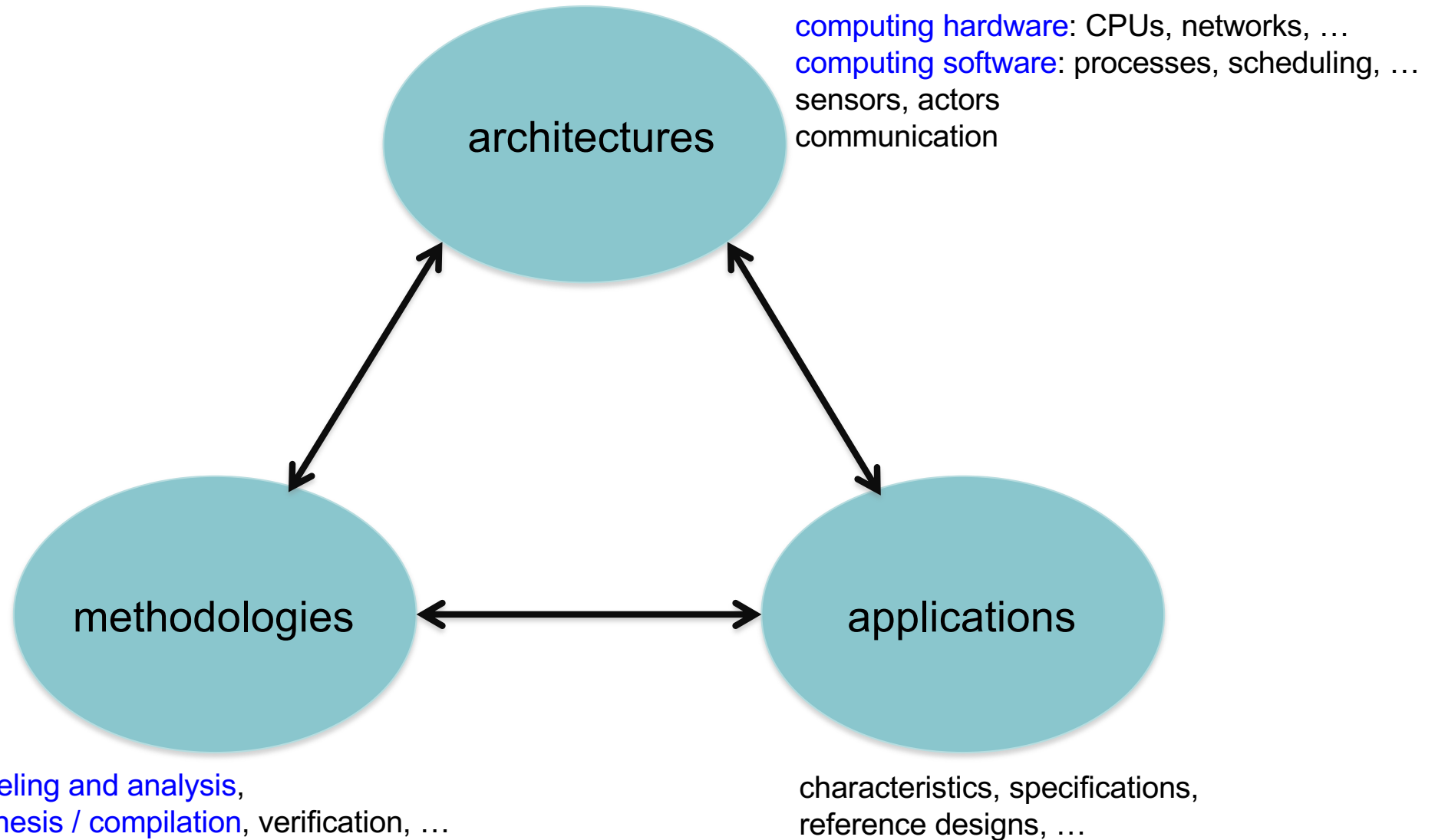
[Kop97] H. Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications. Springer. 1997.



[Man+13] J. Manyika et al. [Disruptive technologies](#): Advances that will transform life, business, and the global economy. McKinsey Global Institute. 2013

Embedded System Design

- Embedded system design comprises studying three fields



Design Goals and Challenges

- Design goals
 - functionality
 - deadlines: hard and soft real-time systems, multi-rate systems
 - power and energy consumption
 - cost: manufacturing and development
 - design time
 - dependability
 - ...
- Challenges
 - finding the right amount of hardware needed
 - meeting deadlines
 - minimizing power consumption
 - designing for upgradeability
 - achieving dependable operation
 - mastering restricted development environments
 - ...



Learning Objectives

- **Lecture**
 - learn about theoretical foundations and practical aspects of embedded system design
- **Exercises**
 - work on exercises (paper & pencil) to deepen understanding of the lecture material
- **Lab**
 - introduction to and practical experience with a real-time operating system (RTOS)

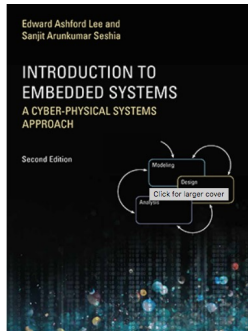
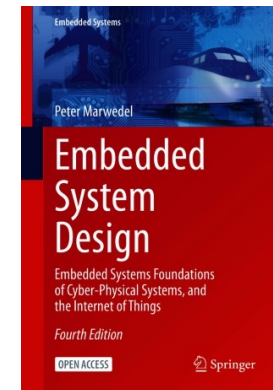


- Introduction
- Specification and modeling
 - models of computation
 - state-based models
 - dataflow-based models
- Target architectures
 - general-purpose processors
 - specialized processors: digital signal processors, microcontrollers, ASIPs
 - FPGAs and ASICs
 - multi-core processors and system-on-chip
- Reactive and real-time systems
 - tasks and task definitions
 - programming paradigms
 - real-time scheduling techniques
 - shared resources
- Performance and energy
 - processor performance, worst-case execution time analysis
 - power and energy



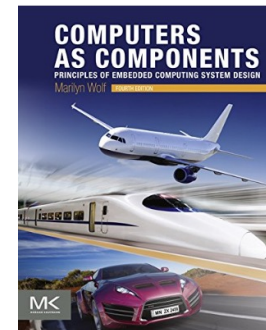
Literature (Selection)

Peter Marwedel: *Embedded System Design*, 4th Ed, Springer, 2021 ([Open Access](#))

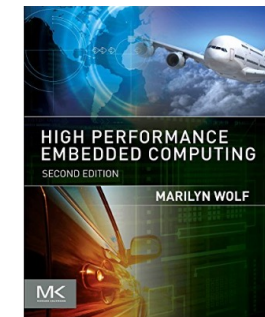


Edward A. Lee and Sanjit A. Seshia: *Introduction to Embedded Systems: A Cyber-Physical Systems Approach*, 2nd Ed, MIT Press, ISBN 978-0-262-53381-2, 2017 ([Open Access](#))

Marilyn Wolf: *Computers as Components*, 4th Ed, Morgan Kaufmann, ISBN 978-0-12-805387-4, 2017



Marilyn Wolf: *High-Performance Embedded Computing*, 2nd Ed, Morgan Kaufmann, ISBN 978-0-12-369485-0, 2014



Uwe Brinkschulte & Theo Ungerer: *Mikrocontroller und Mikroprozessoren*, 3rd Ed, Springer, ISBN: 978-3-642-05397-9, 2010

The lecture slides contain also ideas and material of Lothar Thiele, ETH Zurich.

- **Development platforms / technologies**

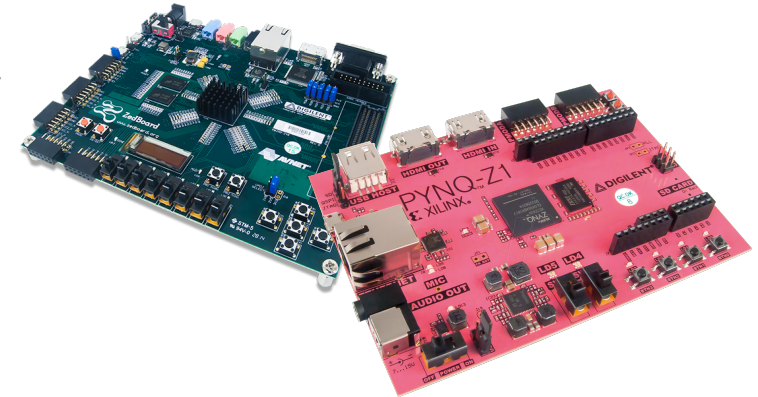
- PYNQ/ZedBoard with ARM/FPGA System-on-Chip Zynq 7020

- dual-core ARM Cortex-A9 processor
- Xilinx 7-series field programmable gate array

- freeRTOS real-time operating system for microcontrollers



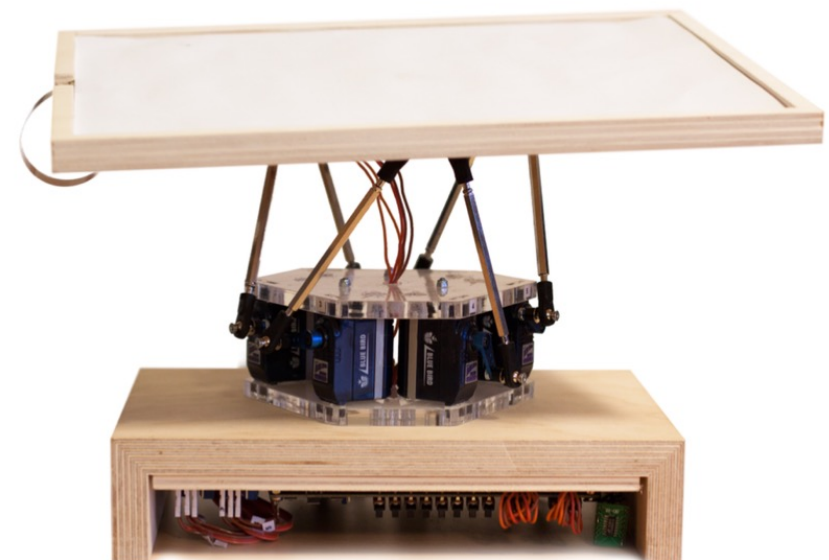
- Xilinx Vitis software platform for C programming



- **Development tasks**

- implement small applications on freeRTOS/ZedBoard

- control a mechatronics system (ball-on-plate)



Lecture Organization

- Materials and information in PANDA
- Lecture & Exercises
 - Wednesday 11:15 – 13:45
 - exercise sheets provided, try to solve the problems on your own, discussion of solutions in class
- Lab
 - announced in PANDA
- Contact
 - Marco Platzner, platzner@upb.de, 60-5250
 - Lennart Clausing, lennart.clausing@uni-paderborn.de, 60-5396 (lab)
- Grading
 - written exam
 - covers material from the lecture, exercises and lab
 - successful lab participation earns a bonus of one or two grade steps (if exam has been passed)

