Technische Universität Chemnitz
Fakultät für Informatik
Professur Technische Informatik
Prof. Dr. Wolfram Hardt

HW/SW Codesign II

Introduction

Prof. Dr. Wolfram Hardt
Dipl.-Inf. Michael Nagler

Sommersemester 2015

Version: 15.01.V.r-1.0-150407

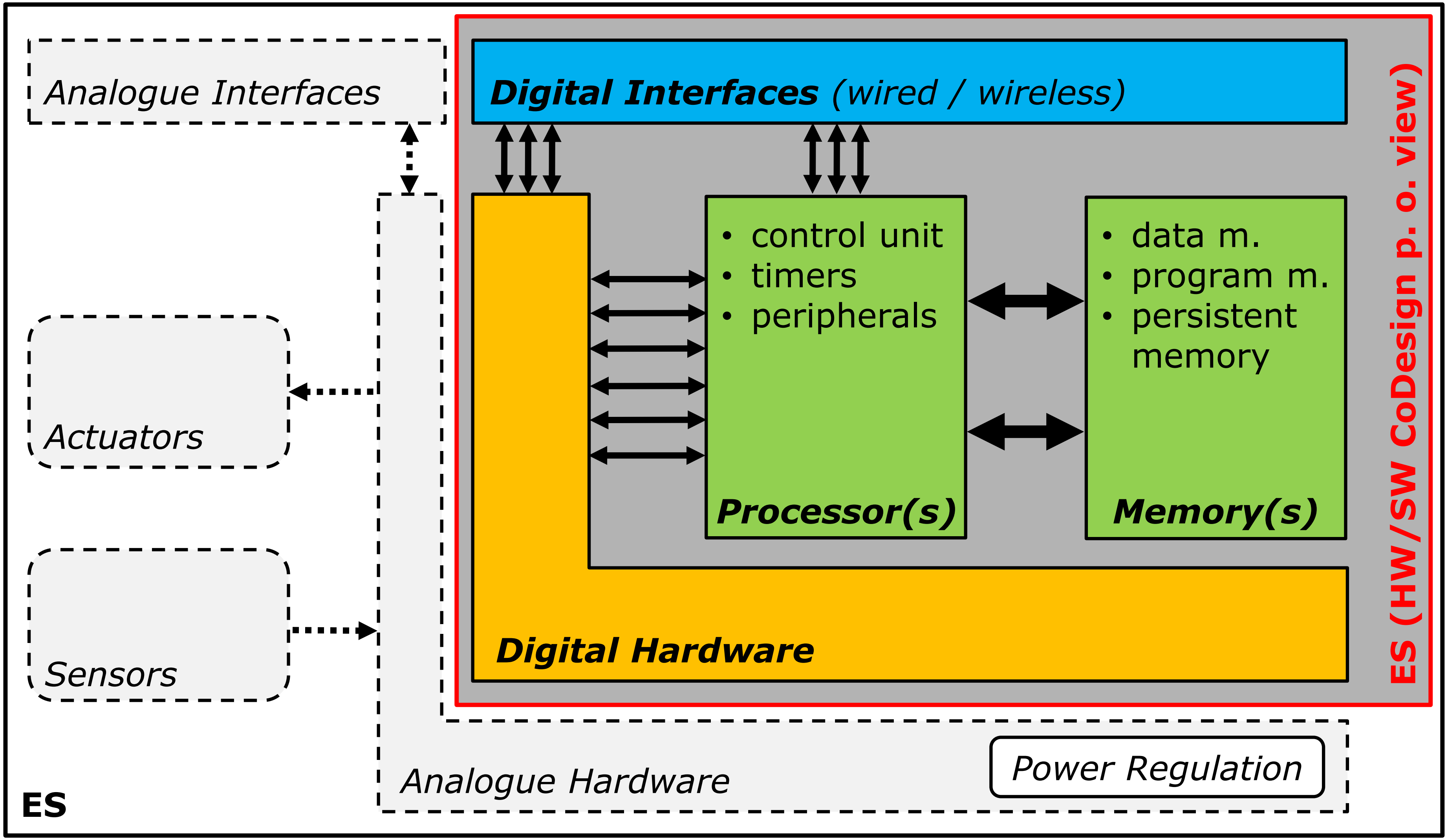
Contents

- Summary HW/SW Codesign I
- Content of Lecture
- Practical Course
- Organisational

Embedded Systems

- Hansson: Embedded systems are computers not looking like computers
  - » example: mobile phone, electronic control of washing machine, telephone switchboard
- Broy: An embedded systems is a
  - » SW/HW unit connected via
  - » sensors and actors with a whole system and
  - » scans, controls and adjusts therein
- more often used properties:
  - » reactive, hybrid and distributed systems
  - » real-time requirements

Structure of Embedded Systems



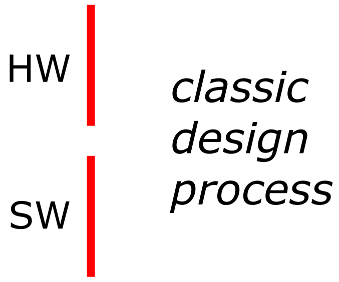
Implementation of ES

Domain of ES	***Behaviour*** ***(developer point of view)***	***Typical Implementation*** ***(developer point of view)***
Actuators / Sensors	converter between voltage / current and other physical values (kinetics, optics, …)	• purchase
Analogue HW / IF	physics	• differential equations • simulation by framework (SPICE, …)
Digital Hardware	finite state machine	• Boolean equations • HW description language + synthesis framework
Processor / Memory	sequential execution of machine code	• assembly language • high-level language + compiler
Digital Interfaces	(partial) embedded system	• standardised → purchase • integrated in other parts • synthesise

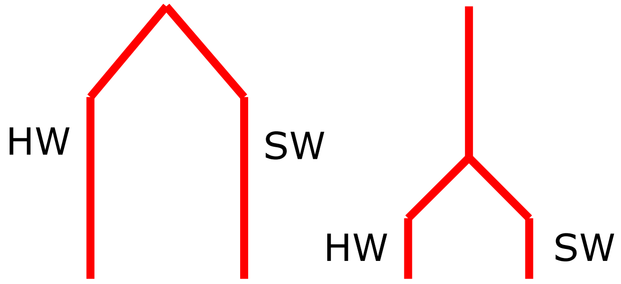
basic knowledge / HSC I + II

HW/SW Codesign

- integrated development of embedded system consisting of
  - hardware-components (HW) ***and***
  - software-components (SW)



classic design process

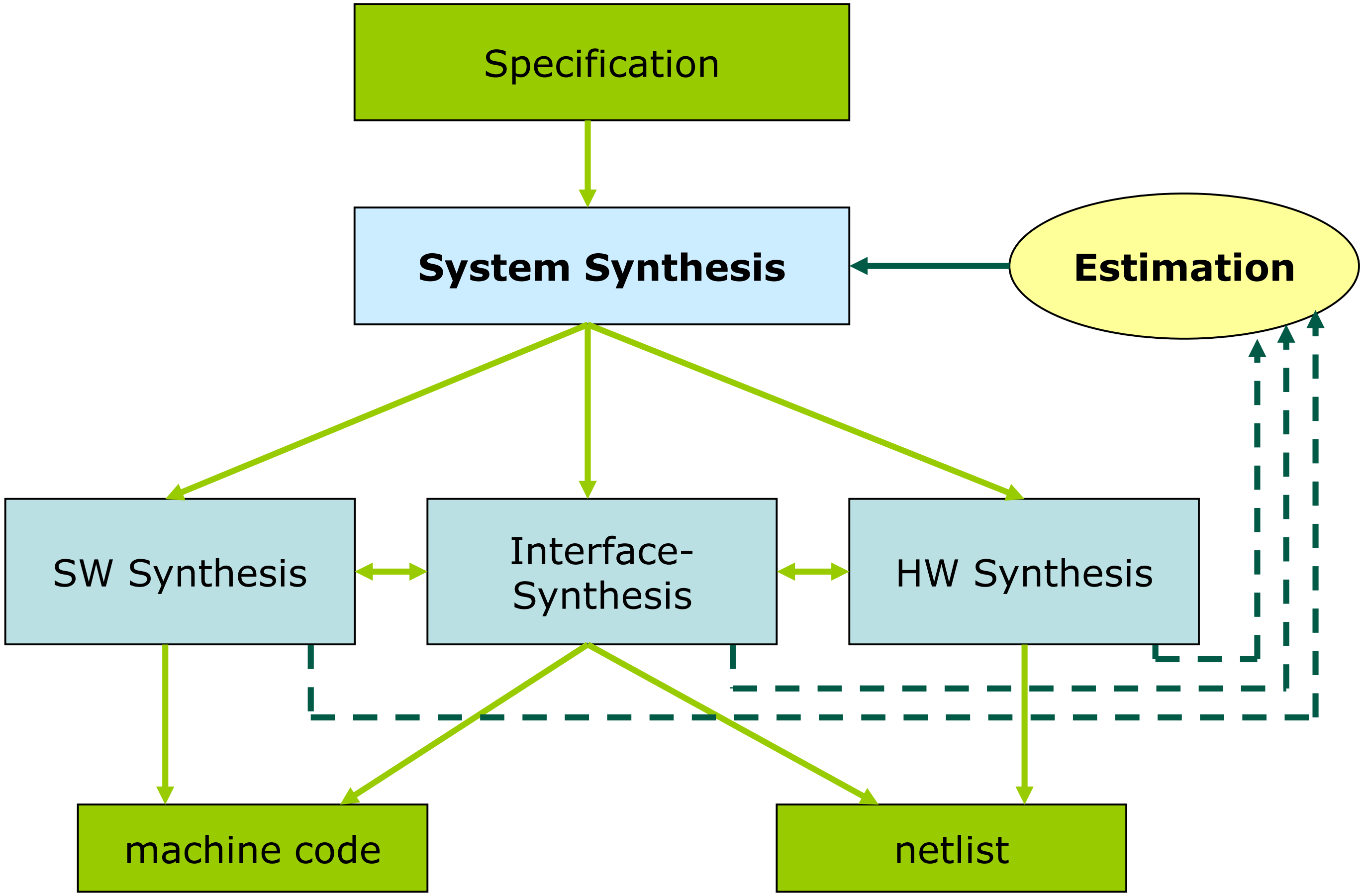


codesign process

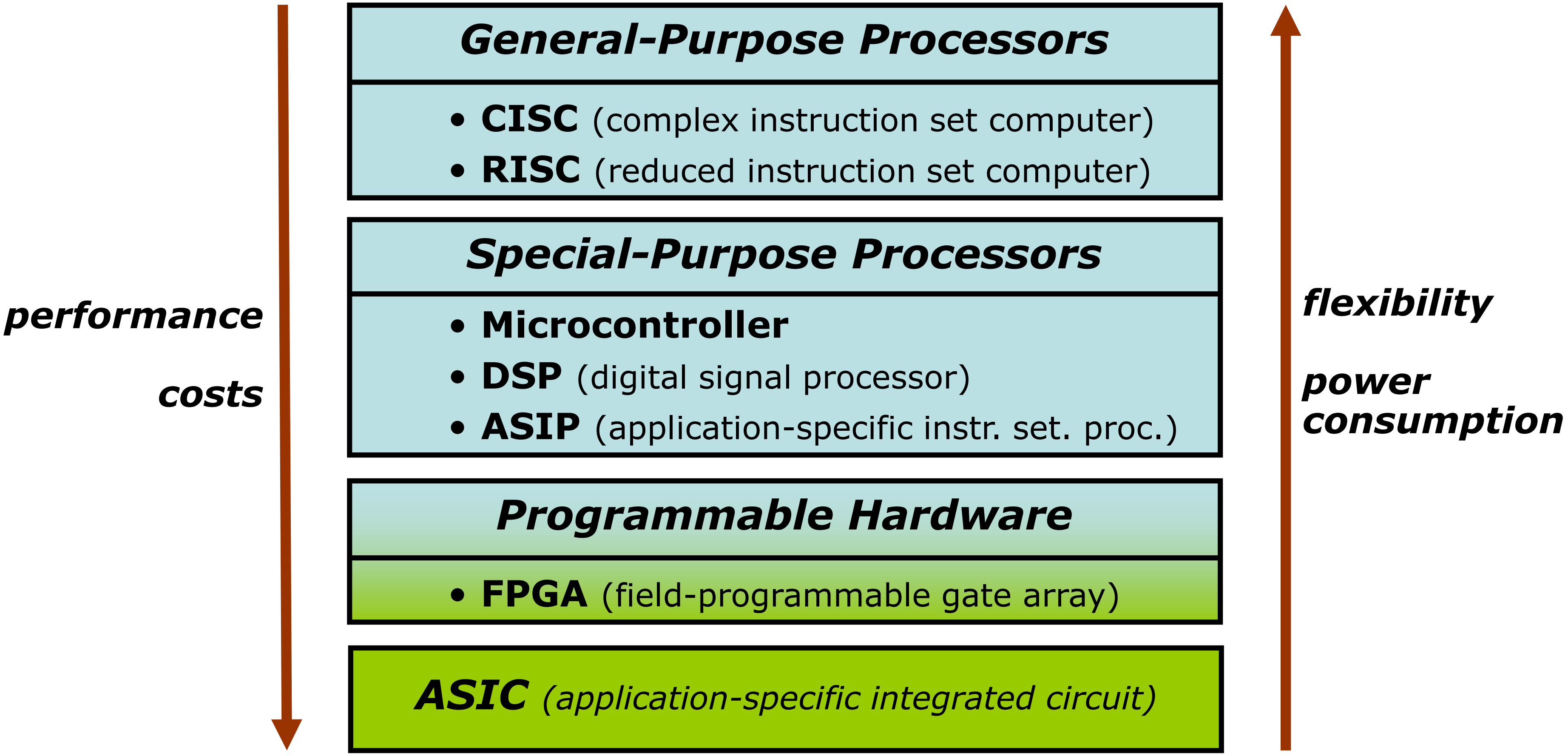
→**design process optimisation**

- special requirements to designer
  - analyse the restrictions of HW and SW
  - evaluation of alternative development solutions
  - integration of HW and SW components

HW/SW Codesign Process



Platform Alternatives



performance

costs

General-Purpose Processors

- **CISC** (complex instruction set computer)
- **RISC** (reduced instruction set computer)

Special-Purpose Processors

- **Microcontroller**
- **DSP** (digital signal processor)
- **ASIP** (application-specific instr. set. proc.)

Programmable Hardware

- **FPGA** (field-programmable gate array)

ASIC *(application-specific integrated circuit)*

flexibility

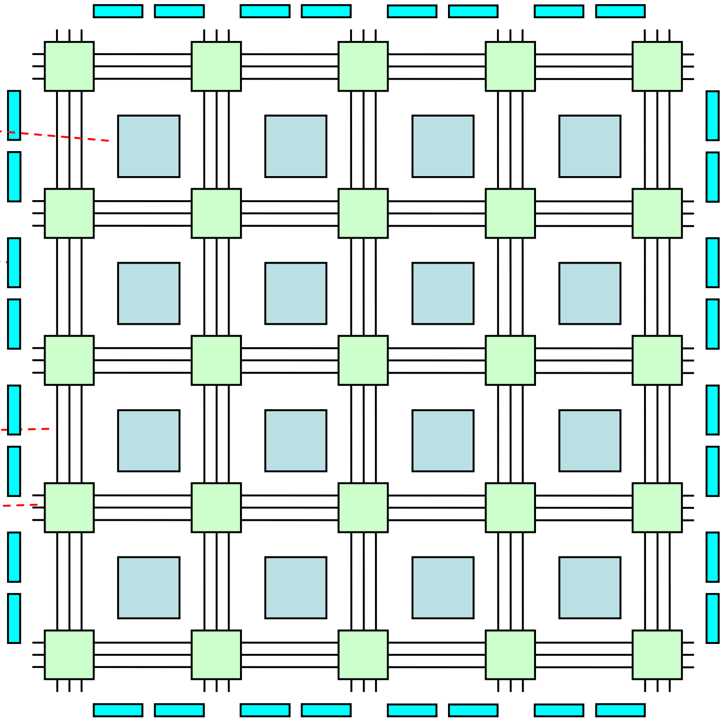
power consumption

Special-Purpose Processors

- microcontroller
  - control-dominant
  - optimised for bit- and logic-operations, interrupt handling, …
  - integrated peripheries (timer, UART, CAN, A/D, …)
  - available from low budget (and low performance) to high performance (and high costs)
- digital signal processors (DSP)
  - dataflow-dominant
  - optimised for arithmetic operations and high data throughput
  - parallel operations possible
- meanwhile both domains are mixed

Field Programmable Gate Arrays

- use (re)programmable hardware to implement sequential systems
  - logic blocks
  - I/O blocks
  - interconnection
    - lines
    - switch matrices

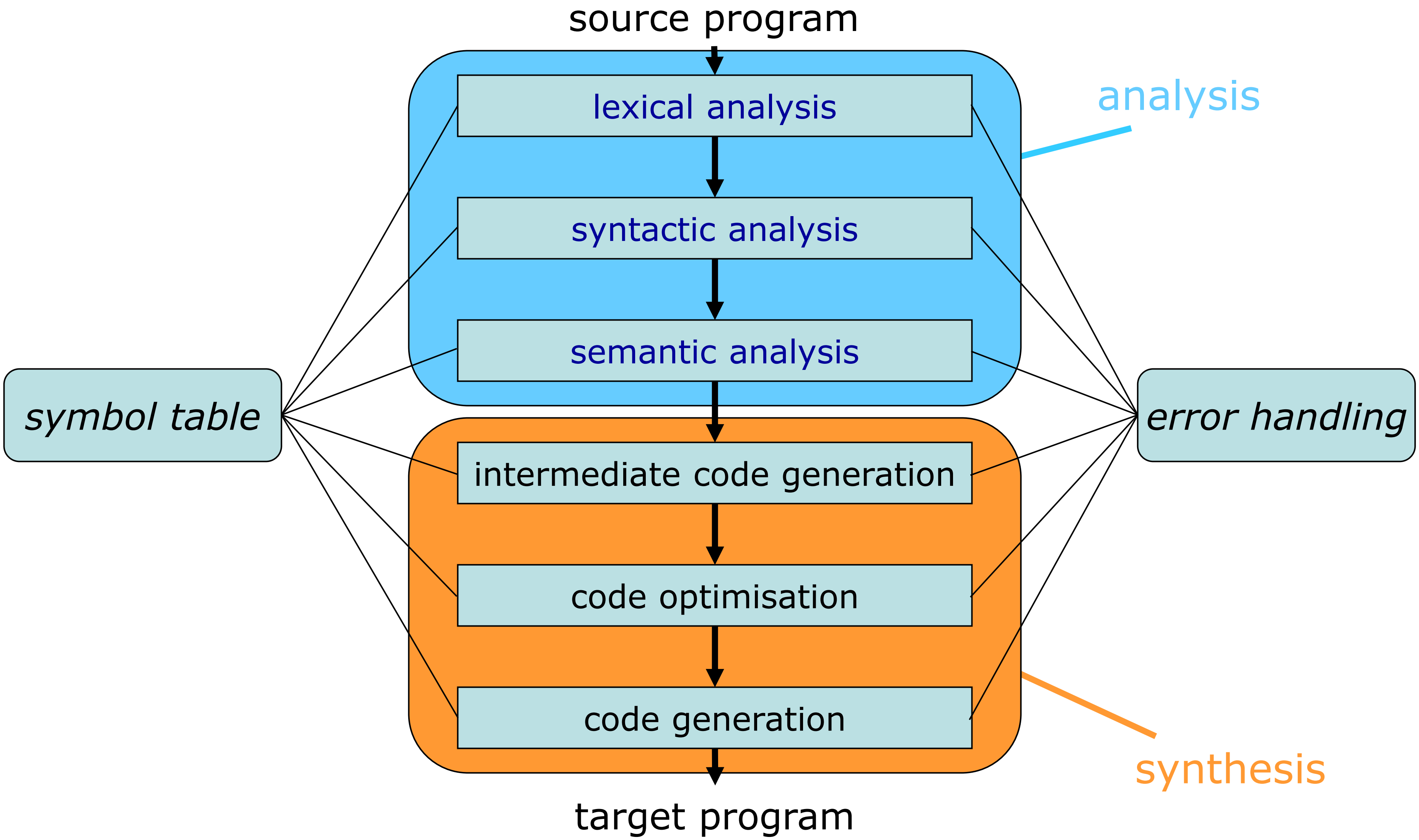


Code Generation

- code generation = *software synthesis*
  - allocation:
    - mostly the components are fixed (e. g. target CPU)
  - binding:
    - register binding (usage counter, graph colouring)
    - instruction selection
  - scheduling (ASAP, ALAP, mobility, list scheduling, …)

- requirements
  - correct code
  - efficient code
  - efficient generation

Compiler Phases



source program

lexical analysis

syntactic analysis

semantic analysis

intermediate code generation

code optimisation

code generation

target program

symbol table

error handling

analysis

synthesis

Partitioning Problem

- **definition**

  The partitioning problem is to assign n objects $O = \{o_1, \ldots, o_n\}$ to *m* blocks (partitions) $P = \{p_1, \ldots, p_m\}$, such that

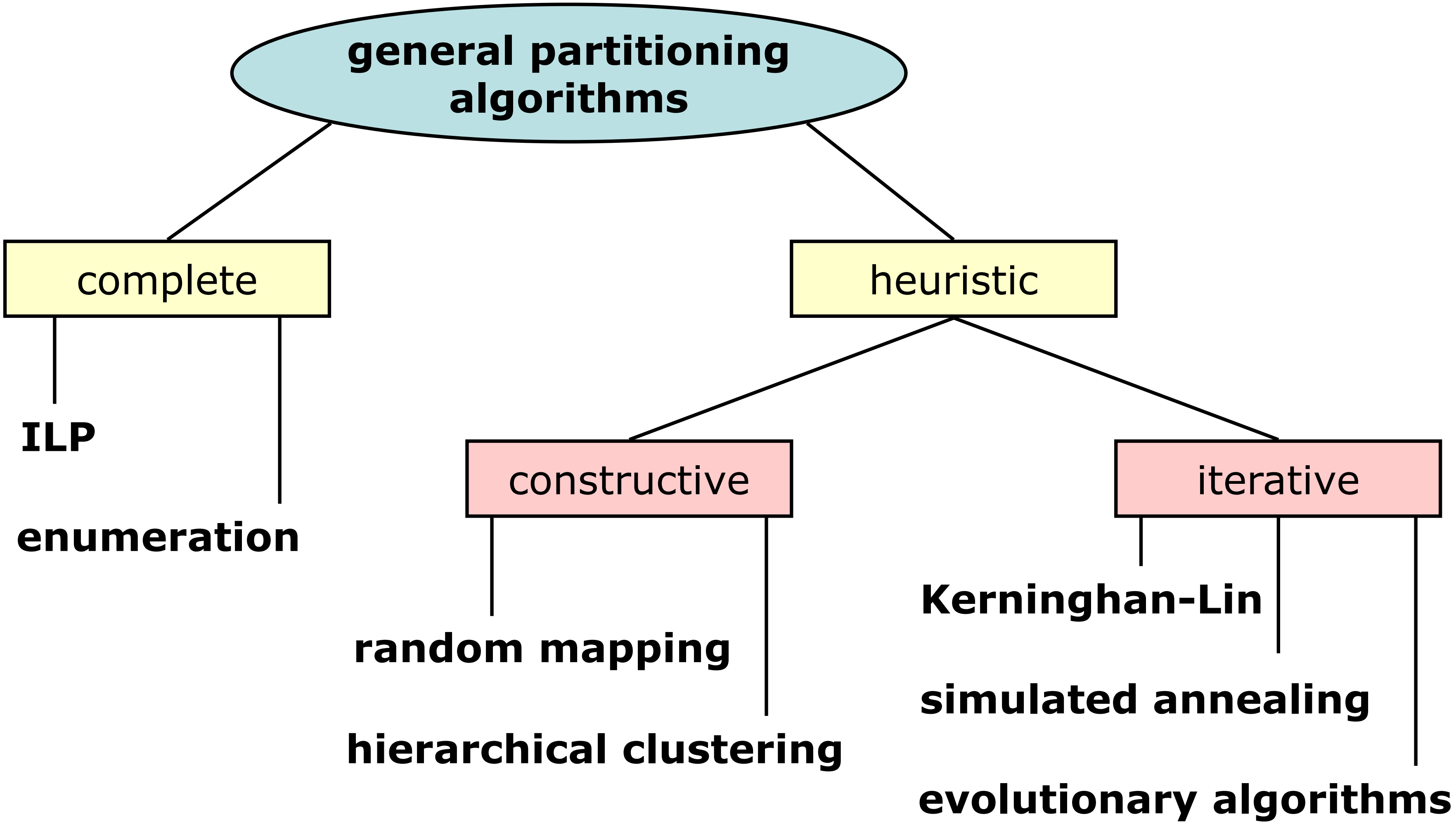
  - $p_1 \cup p_2 \cup \ldots \cup p_m = O$
  - $\forall i, j : i \neq j \Rightarrow p_i \cap p_j = \emptyset$
  - cost *c(P)* are minimised

- costs

  –measure quality of a design point

  - may include C … system cost (in $[\$]$)
L … latency (in $[sec]$)
P … power consumption (in $[W]$)
  - requires estimation to find C, L, P

Algorithms



general partitioning algorithms

complete

heuristic

ILP

enumeration

constructive

iterative

random mapping

hierarchical clustering

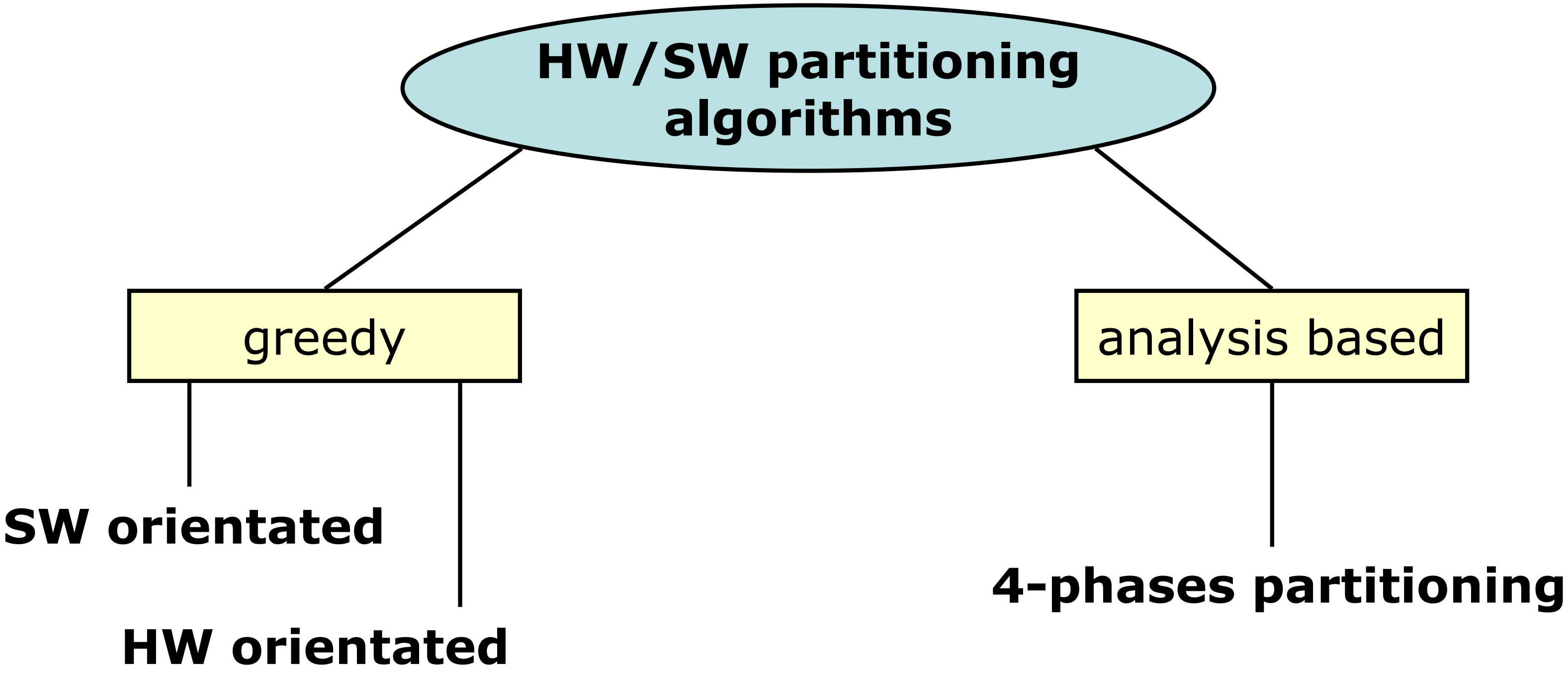
Kerninghan-Lin

simulated annealing

evolutionary algorithms

HW/SW Partitioning

- HW/SW partitioning is special case of partitioning: bi-partitioning $P = \{p_{SW}, p_{HW}\}$



Contents

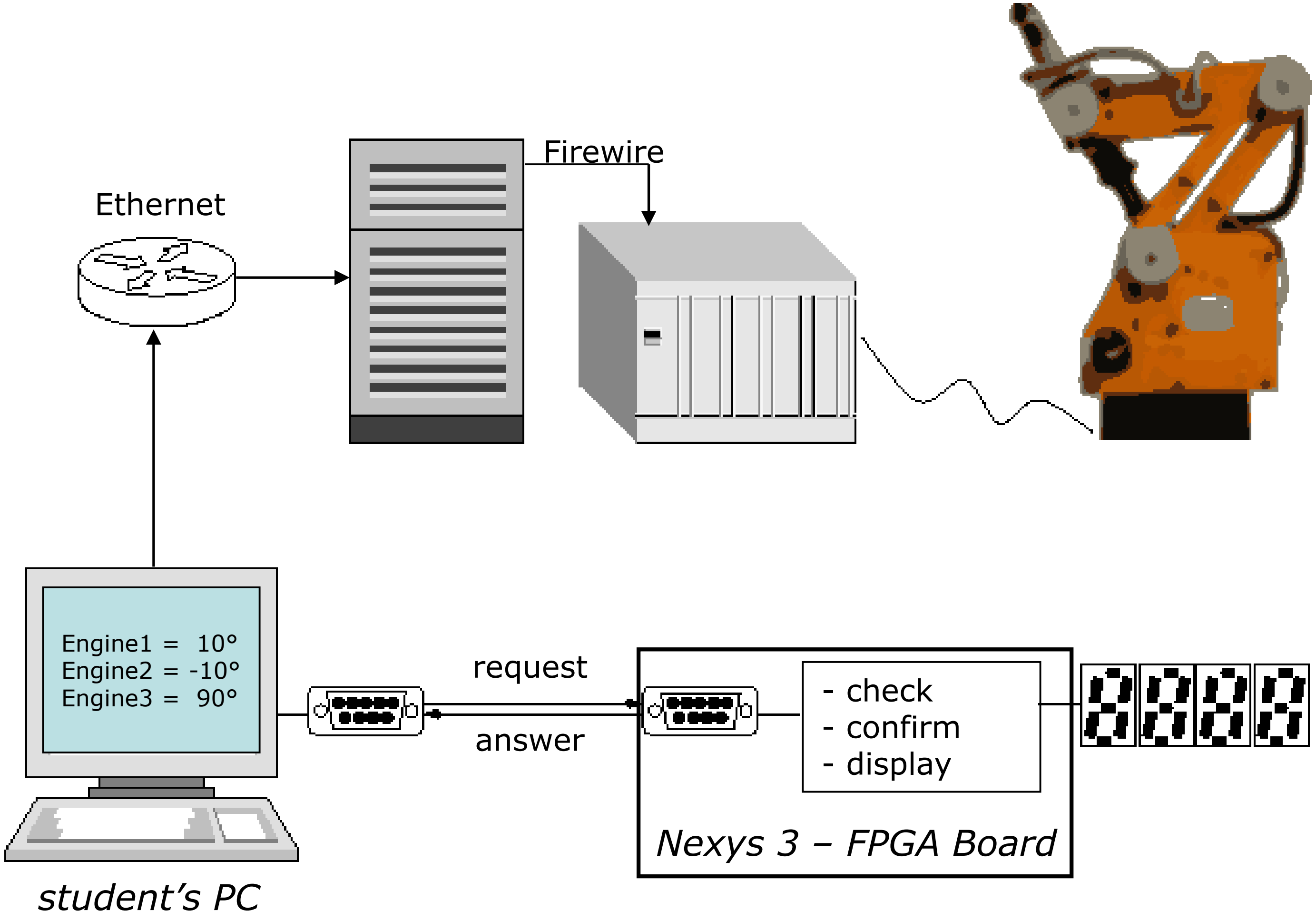
- Summary HW/SW Codesign I
- Content of Lecture
- Practical Course
- Organisational

Lecture Content

- Estimation
- Interfaces (wired, wireless)
- Interface Synthesis
- Rapid Prototyping/ Emulation
- Co-Simulation
- Co-Specification with SystemC

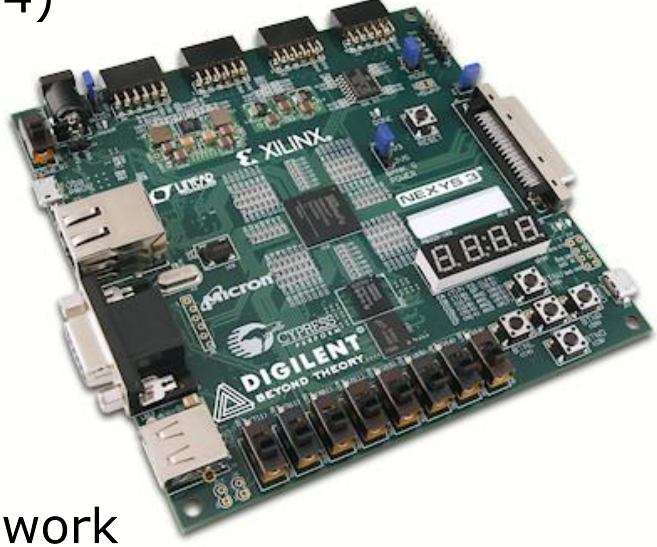
Practical Course (I)

- task: control a robot using a HW/SW implementation



Practical Course (II)

- *Digilent Nexys 3 Board*
  - Xilinx Spartan®-6 FPGA (XC6LX16-CS324)
  - Configurable MicroBlaze Micro Controller IP Core



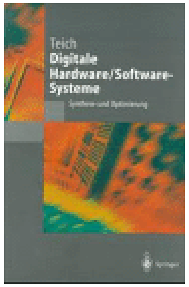
- detailed tasks:
  - learn how to use the development framework
  - implement an asynchronous, serial communication protocol on FPGA (using VHDL)
  - implement algorithms to handle data on microcontrollers (using an ANSI C compiler)

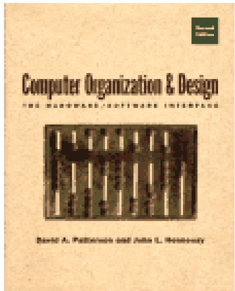
Contents

- Summary HW/SW Codesign I
- Content of Lecture
- Practical Course
- Organisational

Literature

- Teich, Jürgen: Digitale Hardware/Software-Systeme. Berlin: Springer, 1997
- Roth, Jörg: Mobile Computing. Heidelberg: Dpunkt Verlag, 2005
- Patterson, David A.; Hennessy, John L.: Computer Organization and Design: The Hardware/Software Interface. 2. Auflage. Oxford: Elsevier Books, 1997
- Ihmor, Stefan; Flade, Marcel: Rekonfigurierbare Schnittstellen. Dresden: TUDpress, 2005
- Gasteier, Michael: Cosimulation und Kommunikationssynthese im Entwurf gemischter Hardware, Software-Systeme. Aachen: Shaker, 1998
- Schürmann, Bernd: Grundlagen der Rechnerkommunikation. Vieweg+Teubner, 2004





Organisational (I)

• **Lecture:**	Thursday	09:15 – 10:45 (weekly) 1/305
(English)	*Prof. Dr. Wolfram Hardt*	
• **Practical Course:**	Tuesday	03:30 – 06:45 PM (week 1) 1/024
	Tuesday	03:30 – 06:45 PM (week 2) 1/024
	Wednesday	07:30 – 10:45 AM (week 1) 1/024
	Wednesday	07:30 – 10:45 AM (week 2) 1/024
	Friday	07:30 – 10:45 AM (week 1) 1/024
	Friday	07:30 – 10:45 AM (week 2) 1/024

Practical Courses will start 2015/04/20
and will finish individually when tasks are solved

For students of MASE (PO 2013), a previous and successful participation on "HW Development with VHDL" (module 555190) is mandatory.

Organisational (II)

- **Exam:** written test, 90 minutes

  Participation only after successful participation on practical course!

- **Contact:** Michael Nagler, Room 1/023a, Monday: 09:30 – 10:30

- **ALL mails:** **`ce-teaching@informatik.tu-chemnitz.de`**

- **Content:** **`https://www.tu-chemnitz.de/informatik/ce/lectures/lectures.php`**
  - register for practical groups (by OPAL)
  - download slides **BEFORE** lecture
    (They'll be incomplete, so you have to complete them!)
  - download AND **PREPARE** the tasks for practical course