

Lehrform (*teaching format*) / **SWS** (*hours per week*): 2VL + 2UE

Kreditpunkte (*credit points*): 6

Turnus (*frequency*): i.d.R. jedes WiSe

Inhaltliche Voraussetzungen (*content-related prior knowledge/skills*): Technische Informatik 1

Sprache (*language*): Deutsch

Lehrende (*teaching staff*): AG Rechnerarchitektur (Prof. Dr. Rolf Drechsler u.a.)

Studiengang (<i>degree program</i>)	Module	Semester
Informatik (Bachelor VF)	IBAP	ab 5. Sem.
Informatik (Bachelor KF)	KINF-A1/A2	ab 5. Sem.
Systems Engineering (Bachelor)	V07-AuR-V, V07-ESS-V, V07-RF-V	ab 5. Sem.
Informatik (Master)	<i>General Studies</i>	ab 1. Sem.
Systems Engineering II (Master)	M07-AM-INF	1. Sem.
(Industr.)Mathematics (Master)	Anwendungsfach Informatik	

Lernergebnisse:

- Den detaillierten Aufbau moderner Rechner analysieren und erklären können
- Den modernen Systementwurf analysieren können
- Die Funktionsweise von Compilern und Codegenerierung grundlegend verstehen
- Syntheseansätze für Hardware kennen und darstellen können
- Qualität von Systementwürfen beurteilen können
- Aufgabenlösungen und Beispiele in den wöchentlichen Tutorien eigenständig bearbeiten und präsentieren können
- Probleme beim Entwurf eines komplexen Systems selbständig erkennen können

Learning Outcome:

- Ability to analyze and explain the organization of modern computers in detail
- Ability to analyze modern system design
- Understanding the basic working principles of compilers and code generation tools
- Ability to elaborate on synthesis approaches for hardware components
- Ability to judge the quality of system designs
- Ability to autonomously solve exercises/examples and present corresponding solutions in the weekly meetings
- Ability to identify problems in designing a complex system

Inhalte:

Aufbau eines Rechners

- Maschinensprachen
- Datenpfad und Kontrollpfad
- Pipelining

Systementwurf - Modelle und Methoden

- Zielarchitekturen für HW/SW-Systeme
- Allokation, Bindung, Ablaufplanung
- Partitionierung

Software-Entwurf

- Compiler
- Codegenerierung
- Registerallokation

Hardware-Entwurf

- Synthese
- Verifikation
- Verdrahtung
- Test

Schätzung der Entwurfsqualität

Contents:

Computer Organization

- Instruction Sets
- Datapath and Control Unit
- Pipelining

System Design - Models and Methods

- Target architectures for HW/SW systems
- Allocation, Binding, Scheduling
- Partitioning

Software Design

- Compiler
- Code generation
- Register allocation

Hardware Design

- Synthesis
- Verification
- Placement and Routing
- Test

Judging the design quality

Hinweise (*remarks*): In der Tabelle sind nur die primären/spezifischsten Module aufgelistet, denen diese Veranstaltung zugeordnet ist.