Course Description   Computer Architecture

Keywords: computer architecture, microprocessor, microcontroller, instruction set architecture, assembler programming

Target Group:	4th Semester TIB	**Module Number:**	**TIB 428**

Workload:	**5 ECTS**	**150 h**
Divided into:	**Contact time**	**75 h**
	Self-study	**45 h**
	Exam preparations	**30 h**
Course language:	**German and English**	
Module director:	**Prof. Dr.-Ing. Werner Zimmermann**	

Valid from:	**01.03.2014**

Requirements:

- Structure of computer systems (computer unit, control unit, memory, periphery, bus system), Von Neumann architecture - Harvard, CISC und RISC concepts (from Programming, Computer Science)
- Design of procedural and object-oriented programs (from Programming, Computer Science, Software Engineering)
- Software development and software testing in C/C++ with integrated tool chains, systematic software testing (from Programming and Software Engineering)
- Coding and number representation, data types and data structures in advanced programming languages and their mapping of basic data types in computer systems, arithmetic and logical operations in programming languages, the limits of digital arithmetic (number domain, resolution, overflow) (from Computer Science)
- Operating systems' tasks and functions including operations structuring and protection functions in multi-tasking and in multi-user systems, particularly synchronisation and communication concepts

Overall Aims of the Module:

This lecture introduces the architecture of computer systems with microprocessors and microcontrollers. Students will develop a basic understanding for the Instruction Set Architecture of computers and understand how the programming constructs of advanced languages is mapped onto the hardware language. This understanding should help to map the interrelation of programming languages, operating systems and hardware.
In practical exercises, students will implement the hardware programming basics into C/C++ and assembler languages

Contents:

- Computer systems creation, arithmetic-logical operations, basic tasks of operating systems (review)
- Programming model (register set, addressing modes, memory map, instruction set) of an example microprocessor
- Introduction to the machine language, advanced language conceptual mapping, estimated storage capacity and speed of execution
- Hardware and software interfaces for typical periphery building blocks, digital and analog input/output, timer, simple network interfaces
- Modular programming, interface for the interaction between various programming languages
- Operating systems support, for example memory protection, virtual memory, through microprocessors
- Overview of current micro- and signal processor architecture: technical and economic significance

Literature:

Patterson,D.; Hennesey, J.: Computer Architecture and Design, Morgan Kaufmann Verlag.
Tanenbaum, A.: Structured Computer Organization, Prentice Hall Verlag.
Huang, H.W.: The HCS12/9S12. An Introduction to the hardware and software interface,
Thomson Learning Verlag.

Offered:

Every semester

Submodules and Assessment:

Type of instruction/learning:	Lecture with self-study and exam preparations
Type of assessment:	Written exam (90 minutes)
Hours per week:	4 SWS
Estimated student workload:	120 hours

Type of instruction/learning:	Laboratory exercises
Type of assessment:	Attendance certificate
Hours per week:	1 SWS
Estimated student workload:	30 hours

Overall Assessment:

Written exam, non-graded attendance certificate