

Moduldetails

TIIT2003: Rechnertechnik

Modulname	Rechnertechnik
Modulnummer	TIIT2003
Modultyp	Kernmodul
ECTS Creditpoints	7
Studienjahr	2
Dauer	1
Semesterwochenstunden	11
Workload Präsenz (h)	132 h
Workload Selbststudium (h)	78 h
Lehrveranstaltungen (Units)	TIIT2003.1 Rechnertechnik I TIIT2003.2 Rechnertechnik II
Prüfungsleistungen benotet	2
Prüfungsleistungen unbenotet	0
Lernziele	<p>Methodik und Funktionsweise digitaler Rechengänge (Integerarithmetik und logische Funktionen)</p> <p>Aufbau und Funktion von Rechen- und Steuerwerken</p> <p>Interner Aufbau von Digitalprozessoren</p> <p>Fundamentale Prozessorarchitekturen (RISC, CISC, DSP)</p> <p>Speichersysteme, -technologien und Aufbau (SRAM, DRAM, ROM-Typen, ...)</p> <p>Kommunikationssysteme (parallele und serielle Kommunikation, Busse mit HW-Handshaking)</p> <p>Systemaufbau und -entwurf von Digitalrechnern</p> <p>Konzepte zur Steigerung der Verarbeitungseffizienz (Pipelining, Caching, virtueller Speicher, Fließkommaverarbeitung)</p> <p>Aktuelle und zukünftige Entwicklungen für die Rechnerarchitektur von Workstations, Servern und eingebetteten Systemen</p> <p>Programmiermodel von Digitalrechnern auf Maschinenebene</p> <p>Befehlssatz und Adressierungsmethoden von Digitalrechnern</p> <p>Programmkontrolle auf Maschinenebene</p> <p>Entwurfsmethoden für Maschinenprogramme</p> <p>Vorgenannte Punkte eines konkreten Beispielprozessors / Übungsrechners</p> <p>Software-Entwicklungswerkzeuge für den Beispielprozessor / Übungsrechners</p> <p>Entwicklung mehrerer Maschinenprogramme mit steigenden Schwierigkeitsgrad</p>
Lerninhalte	<p>TIIT2003.1: Rechnertechnik I</p> <p>Teil A: Prozessorarchitektur</p> <p>Einführung</p> <p>Historie (mechanisch, analog, digital)</p> <p>Architektur nach von Neumann</p> <p>Systemkomponenten im Überblick</p> <p>Grobstruktur der Prozessorinterna</p> <p>Rechenwerk</p> <p>Addition: Halbaddierer, Volladdierer, Wortaddierer, Bedeutung des Carrybits, Ripple + Look-ahead Carry</p> <p>Subtraktion: Transformation aus Addition, Bedeutung des Carrybits</p> <p>Multiplikation: Parallel- und Seriell-Multiplizierer</p> <p>Division: Konzept</p> <p>Arithmetische-logische Einheit (ALU)</p>

ALU mit Rechenregister und Ergebnisflags (CCR, Statusbits)

Steuerwerk

Aufbau und Komponenten

Befehlsdekodierung und Mikroprogrammierung

Struktur von Prozessorbefehlssätzen

Klassifizierung und Anwendung Prozessorregistern (Daten, Adressen und Status)

Businterface

Buskomponenten: Daten-, Adress- und Steuerleitungen

Buszyklen: Lese- und Schreib-Zugriff, Handshaking (insbesondere Waitstates)

Busarbitrierung

Busmultiplexing

Befehlssatz und Maschinenprogrammierung

Programmiermodell: Befehlssatz und Adressierungsarten

Umsetzung von Kontrollstrukturen, Auswertung von Ergebnisflags

Unterprogrammstruktur mit Hilfe des Hardwarestacks: Mechanismen, Aufruf-Konventionen

Konzept und Umsetzung Unterbrechungsbehandlung (HW- und SW-

Interrupts): Diskussion von HW- und SW-Mechanismen und

Automatismen, IR-Vektortabelle; Spezialfall: Bootvorgang

Diskussion User- und Supervisor-Modus von Prozessoren

Teil B: Rechnersystemarchitektur

Fundamentalarchitekturen

von Neumann und Harvard

Konzept Systemaufbau und Komponenten: CPU, Hauptspeicher, I/O:

Diskussion Anbindung externer Geräte (Grafik, Tastatur, Festplatten, DVD, ...)

Halbleiterspeicher

Wahlfreie Speicher: Aufbau, Funktion, Adressdekodierung, interne Matrixorganisation

RAM: statisch, dynamisch, aktuelle Entwicklungen

ROM: Maske, Fuse, EPROM, EEPROM, FEPRM, aktuelle Entwicklungen

Systemaufbau

Aufteilung des Adressierungsraumes

Entwerfen von Speicherschemata und der zugehörigen Adress-Dekodierlogik

Vitale System-Komponenten: Stromversorgung, Rücksetzlogik, Systemtakt, Chipsatz

I/O-Schaltkreise: Interrupt- und DMA-Controller, Zeitgeber- und

Uhrenbausteine

Schnittstellen: Parallel und seriell, Standards (RS232, USB, ...)

Performancekonzepte

Pipelining

Caching

Virtueller Speicher

CPU-Strategien: CISC, RISC, DSP, FPU; aktuelle Beispiele (Pentium, Itanium, Power-PC, Strong-ARM)

Integrierte Systeme: Mikrocontroller, ASIC, ASPP

Beispielsysteme

PC (Desktop-Rechner, Laptops)

Mobile Geräte (PDA, Handy)

Steuersysteme (KFZ-Motorsteuerung)

TIIT2003.2: Rechnertechnik II

Theorie

Befehlsarten digitaler Rechenanlagen

Adressierungsarten digitaler Rechenanlagen

Steuerung des Programmlaufs

Schnittstellen von Funktionen und Unterbrechungsroutrinen
Methoden des maschinennahen SW-Entwurfs

Praktische Übungen

Einführung eines Beispielprozessors

Aufbau des Übungsrechners

Einarbeitung und Softwareentwicklungs- und Testumgebung für den Übungsrechner

Selbständige Entwicklung von Maschinenprogrammen mit steigenden Schwierigkeits- und Strukturierungsgrad

Zu den Modultypen:

Kernmodul

Pflichtfach für diesen Studiengang (an allen Standorten)

Allgemeines Profilmodul

Pflichtfach für diesen Studiengang in der speziellen Vertiefung / Schwerpunkt an allen Standorten

Lokales Profilmodul

Pflichtfach für diesen Studiengang in der speziellen Vertiefung / Schwerpunkt am gewählten Standort

Die Änderungen der neuen Prüfungssatzung sind hier nur teilweise abgebildet. Für detaillierte Informationen wenden Sie sich bitte an Ihren Studiengangsleiter.