

64-040 Modul IP7: Rechnerstrukturen

0. Organisatorisches

Norman Hendrich

Universität Hamburg
MIN Fakultät, Department Informatik
Vogt-Kölln-Str. 30, D-22527 Hamburg
hendrich@informatik.uni-hamburg.de

WS 2013/2014

Inhalt

1. Übersicht

Themen

Vorläufige Terminplanung

2. Organisatorisches

Vorlesung

Übungen

Tutorium

Praktikum

Klausur

3. Literaturempfehlungen

4. Software

Modul Rechnerstrukturen: Motivation

Das Pflichtmodul Rechnerstrukturen vermittelt ein begrenztes und wohl ausgewähltes Theorie- und Methodenrepertoire für Konfigurierung, Entwurf, Realisierung und angemessene Nutzung von Rechnern und Kommunikationsnetzen sowie ihrer Basiskomponenten unter Berücksichtigung technologischer, ökonomischer und anwendungsspezifischer Randbedingungen.

... Grundlegende Konzepte, Organisationsformen und Entwurfsmethoden von Rechnerarchitekturen und deren Vernetzung, einschließlich der Betriebssoftware. . .

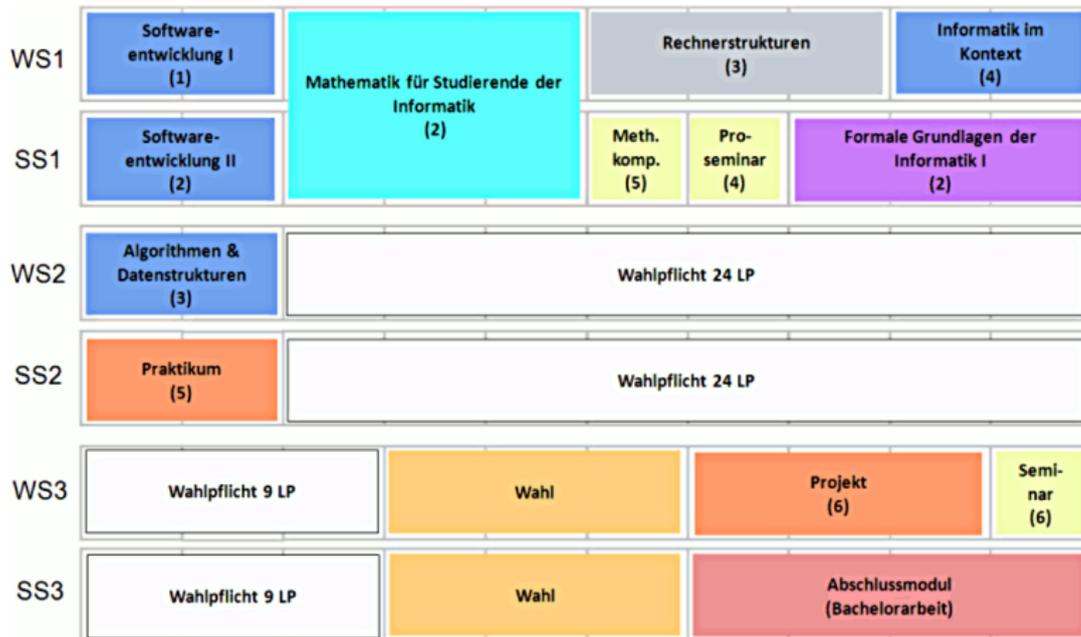
www.informatik.uni-hamburg.de/Info/Studium/BSc/Module/



Modul Rechnerstrukturen: Lernziele

- ▶ Leitbild: *eine der tragenden Säulen der Informatik*
- ▶ Faktenwissen: *Grundkenntnisse über digitale Rechner*
- ▶ Methodenwissen: *Analyse und Synthese von technischen Systemen*
- ▶ Transferkompetenz: *Anwendungen der Methoden der technischen Informatik, Verständnis von zeitlichen Abfolgen, Zusammenspiel von Software- und Hardwarekomponenten*
- ▶ Normativ-bewertende Kompetenz: *Urteilsvermögen zur Analyse von Rechnerarchitekturen und deren Komponenten*
- ▶ Schlüsselqualifikationen: *Kooperations- und Teamfähigkeit, Problemlösungskompetenz, Abstraktionsvermögen, Anwendung von Entwurfsmethoden, Befähigung zur Analyse und Synthese von technischen Systemen*

Informatik-BSc: Übersicht



Software-System-Entwicklung-BSc: Übersicht

WS1	Software-entwicklung I (1)	Informatik im Kontext (3)	Rechnerstrukturen (3)	Mathematik für Studierende der Informatik (2)	
SS1	Software-entwicklung II (2)	Meth. komp. (5)	Pro-seminar (4)		
WS2	Software-entwicklungs-praktikum (5)	Grundlagen von Datenbanken (5)	Projekt-manag. (5)	Wahlpflicht Informatik	Wahlpflicht Informatik
SS2	Softwaretechnik (4)	Wahlpflicht Informatik	Wahl / Anwendungsgebiet	Projekt (6)	
WS3	Wahl / Anwendungsgebiet	Seminar (6)	Industriepraktikum (6)		
SS3	Wahlpflicht Informatik	Wahl / Anwendungsgebiet	Abschlussmodul (Bachelorarbeit)		



Rechnerstrukturen: Themen

- ▶ Einführung
- ▶ Grundprinzip des von-Neumann Rechners
- ▶ Abstraktionsschichten, virtuelle Maschinen
- ▶ Hardware-/Software-Schnittstelle

- ▶ Informationsbegriff und -theorie, Codierung
- ▶ Entwurf von digitalen Schaltungen
- ▶ Grundkomponenten des Rechners, Realisierungsaufwand
- ▶ Rechnerarchitektur

Themenübersicht

Teil 1

- ▶ Information und Repräsentation
- ▶ Zahldarstellung und Arithmetik
- ▶ Boole'sche Algebra und -Funktionen
- ▶ Schaltnetze und Schaltwerke
- ▶ Entwurf digitaler Schaltungen
- ▶ Komponenten der Register-Transfer Ebene

Teil 2

- ▶ Rechnerarchitektur
- ▶ Befehlssätze, CISC- und RISC-Architektur
- ▶ Maschinen- und Assemblerprogrammierung
- ▶ Speicherhierarchie und Speicherverwaltung
- ▶ I/O-Operationen, Interrupts
- ▶ Leistungsbewertung, Parallelrechner

Terminübersicht Teil 1

	18.10	Einführung, von-Neumann Konzept
23.10	25.10	Information, Zahldarstellung
30.10	01.11	Arithmetik, Textcodierung
06.11	08.11	Informationstheorie, Optimalcodes
13.11	14.11	Boole'sche Algebra, bitweise Operationen
20.11	22.11	Schaltfunktionen, Normalformen
27.11	29.11	Schaltnetze, Gatter, Rechenwerke
04.12	06.12	Schaltwerke, Flipflops, Entwurf von Schaltungen

Terminübersicht Teil 2

11.12	13.12	Register-Transfer-Ebene, ISA, Befehlsformate
18.12	20.12	x86-Architektur, Assemblerprogrammierung
08.01	10.01	Funktionsaufrufe, Stack
15.01	17.01	Datenstrukturen, Speicherhierarchie
22.01	24.01	Speicherhierarchie, Virtueller Speicher
29.01	31.01	Pipeline, Parallelrechner

Feedback erwünscht

Bitte alle Fehler und Ungenauigkeiten in den Folien und Materialien melden. Ebenso bitte Feedback bei Unklarheiten etc.!

Vorschläge und Hinweise auf Tools, schöne Lehrmaterialien etc. sind immer willkommen!

hendrich@informatik.uni-hamburg.de
 zhang@informatik.uni-hamburg.de

Kontakt

Dr. Norman Hendrich

hendrich@informatik.uni-hamburg.de

+4940 42883 2399

Informatikum, Haus F314

Prof. Dr. Jianwei Zhang

zhang@informatik.uni-hamburg.de

+4940 42883 2430

Informatikum, Haus F310

Sprechstunde: Donnerstag 14:00-16:00

Vorlesung

- ▶ Mi. 16:15-17:45 Erzwiss. H, VMP-8
Fr. 14:15-15:45 Phil. A, VMP-6 (4 SWS)
- ▶ Folien (pdf) und Materialien werden gestellt
- ▶ Lecture2Go: lecture2go.uni-hamburg.de
- ▶ diverse gute Lehrbücher verfügbar — Empfehlungen s.u.
- ▶ eingestreute Hinweise auf aktuelle Themen und Vertiefung

- ▶ Assemblerprogrammierung: x86 mit GNU-Toolchain
- ▶ unter Windows: Cygwin mit GNU-Toolchain

- ▶ Informationen und Downloads auf der Webseite:

Übungen

- ▶ dreizehn Gruppen geplant, Details siehe KVV/Webseite/STiNE
- ▶ Anmeldung über STiNE
- ▶ 1 SWS, wöchentlich

- ▶ Diskussion und Nachbereitung der Vorlesung
- ▶ Gruppenarbeit erwünscht (max. 3 Teilnehmer pro Gruppe)

- ▶ Übungsaufgaben zum Vertiefen und Erarbeiten des Stoffes
- ▶ Aufgabenblätter jeweils Freitag zum Download verfügbar
- ▶ Abgabe der Lösungen schriftlich bis nächsten Freitag 12:00 beim Gruppenleiter oder im TAMS-Sekretariat

Übungen: Scheinkriterien

- ▶ regelmässige aktive Teilnahme
- ▶ mindestens zweimal an der Tafel vorrechnen
- ▶ höchstens zweimal (entschuldigt) gefehlt

- ▶ alle Aufgabenblätter bearbeitet
- ▶ jeweils mindestens 30 % der Punkte pro Aufgabenblatt
- ▶ und mindestens 50 % der Gesamtpunktzahl

Übungen: Gruppeneinteilung / Wechselwünsche?

- ▶ Übungen beginnen in der zweiten Vorlesungswoche
Achtung: erstes Übungsblatt ist bereits online!
- ▶ Anmeldung und Ranking (während OE) über STiNE
- ▶ Auslastung derzeit noch nicht bekannt
- ▶ jeweils zwei Gruppen parallel: einfache Wechselmöglichkeit
- ▶ Ausgleich der Gruppengröße in der ersten Woche
- ▶ Wechselwünsche: bitte zur gewünschten Gruppe erscheinen, vor Ort in Warteliste eintragen
- ▶ späterer Wechsel bei freien Plätzen nach Rücksprache mit den Gruppenleitern möglich (Übertragung der Punkte, etc.)

Übungen: Stundenplan

Derzeitige Anmeldungen:

Mo	12-13	F-009	F-334	30	×2
	13-14	F-009	F-334	30	×2
	14-15	R-031	F-635	31	×2
Di	10-11	D-129	F-334	24	×2
	11-12	D-129	F-334	25	×2
	12-13	F-534	F-334	30	×2
	13-14	F-534	F-334	15	×2

Tutorium

- ▶ ab dieser Woche, wöchentlich
- ▶ freiwillig, Nacharbeiten von Stoff und Übungen
- ▶ Diskussion, Fragen: mehr Zeit als in der Übung

- ▶ Di. 12-13 Pavillon-III, 007 (hinter Haus E)
- ▶ Di. 13-14 Pavillon-III, 007 (hinter Haus E)
- ▶ Di. 14-15 Pavillon-III, 007 (hinter Haus E)

- ▶ Paniktutorium: Vorbereitung auf die Klausur
- ▶ Anfang Februar 2014, Ort und Zeit wird bekanntgegeben

Praktikum

- ▶ Blockkurse á 4 Termine zu je 3 h (1 SWS)
- ▶ Vorbereitung **unbedingt** erforderlich
- ▶ Erarbeiten und Programmieren eines einfachen Prozessors
- ▶ Komponenten auf der Register-Transfer-Ebene
- ▶ Zeitverhalten, Speicheransteuerung
- ▶ Mikroprogrammierung
- ▶ Assemblerprogrammierung
- ▶ I/O-Operationen und Interrupts

<http://tams-www.informatik.uni-hamburg.de/lectures/2013ws/praktikum/rechprak/>

Klausur

- ▶ Abschlussprüfung des Moduls Rechnerstrukturen
- ▶ Note geht in BSc-Zeugnis ein

- ▶ insgesamt zwei Klausurtermine in den Semesterferien
- ▶ Do 06/02/2014 10:30-12:30 Audimax 1
- ▶ Do 06/03/2014 10:30-12:30 Audimax 2
- pünktlich kommen!**

- ▶ Tipp: möglichst bereits die erste Klausur mitschreiben
- ▶ zweite Klausur fällt regelmäßig schlechter aus (leider)

Probeklausur

- ▶ Umfang und Fragen wie „echte“ Klausur
- ▶ vermutlich im Rahmen des Tutoriums (2-stündig)
- ▶ anschließende gemeinsame Korrektur/Besprechung
- ▶ Termin: vorletzte Semesterwoche?

Literatur: Empfohlene Lehrbücher

- ▶ Randal E. Bryant and David O'Hallaron,
Computer Systems — A programmers perspective,
 Pearson Prentice Hall, 2nd. Ed., 2010

Rechnerarchitektur mit Schwerpunkt Software und Systeme, leider nicht ganz billig. Viele C-Programme und Systemprogrammierung. Beispiele anhand Intel x86 Architektur. Keine wesentlichen Änderungen gegenüber der Erstauflage von 2003.

- ▶ Andrew S. Tanenbaum,
Structured Computer Organization / Computerarchitektur,
 5th. edition, Pearson Prentice Hall, 2006

Guter Überblick, klares didaktisches Konzept. Java VM, Intel x86, SPARC. Mit jeder Auflage komplett überarbeitet und aktualisiert.

Literatur: weitere Lehrbücher

- ▶ David A. Patterson and John L. Hennessy, *Computer Organization and Design — the hardware/software interface*, 4th edition, Morgan Kaufmann, 2009

Schönes Lehrbuch von den Entwicklern der RISC/MIPS Prozessoren.

- ▶ David A. Patterson and John L. Hennessy, *Rechnerorganisation und -entwurf — die Hardware/Software-Schnittstelle* 3. Auflage, Spektrum Verlag, 2005

Die deutsche Übersetzung, leider eine Auflage zurück: die aktuellen Ergänzungen wie z.B. Multi-Core Maschinen fehlen. Mehrere Exemplare in der Informatik-Bibliothek.

Literatur: weitere Lehrbücher

- ▶ Schiffmann und Schmitz,
Technische Informatik 1, Grundlagen der digitalen Elektronik
Technische Informatik 2, Grundlagen der Computertechnik
Springer Verlag, 2004, 2005
- ▶ Schiffmann und Schmitz,
Übungsbuch zur Technischen Informatik 1 und 2
Springer Verlag, 2003

Dutzende von Übungsaufgaben mit detailliert entwickelten Lösungen.
- ▶ Miles Murdocca and Vincent Heuring,
Computer Architecture and Organization, An Integrated Approach John Wiley and Sons, 2007



Literatur: aus Hamburg

- ▶ Klaus Lagemann,
Rechnerstrukturen,
Springer Verlag, 1987.
- ▶ Dietmar Möller,
Rechnerstrukturen, Grundlagen der Technischen Informatik,
Springer Verlag, 2003
- ▶ Dietmar Möller und Martin Lehmann,
Skript zur Vorlesung Rechnerstrukturen
Uni Hamburg, 2008
- ▶ Andreas Mäder,
Skript zur Vorlesung Rechnerarchitektur und Mikrosysteme
Uni Hamburg, 2008



Literatur: Assemblerprogrammierung

- ▶ Randy Hyde,
Art Of Assembly Language Programming,
online zum Download verfügbar unter
<http://homepage.mac.com/randyhyde/webster.cs.ucr.edu/www.artofasm.com/index.html>



Literatur: Vertiefung

- ▶ John L. Hennessy and David A. Patterson,
Computer Architecture — A Quantitative Approach,
4th. edition, Morgan Kaufmann, 2007
Die Bibel zum Thema Rechnerarchitektur
- ▶ Donald E. Knuth, *The Art of Computer Programming:*
Volume 1 Fascicle 0: MMIX
Volume 4 Fascicle 0: Boolean Functions
*Volume 4 Fascicle 1: Bitwise Tricks and Techniques, Binary
Decision Diagrams* Addison-Wesley, 2006-2009
- ▶ Giovanni de Micheli *Synthesis and Optimization of Digital
Circuits*, McGraw-Hill, 1994

Literatur: über den Tellerrand hinaus

- ▶ Reiner Hartenstein, *Standort Deutschland: Wozu noch Mikro-Chips*, IT-Press Verlag, 1994 (vergriffen)

- ▶ Tracy Kidder, *The soul of a new machine*, diverse Verlage, 1981
- ▶ Jeff Hawkins, *On Intelligence*, Times Books, 2004
- ▶ Raul Rojas, *Neural Networks — A Systematic Introduction*, Springer (1996)

page.mi.fu-berlin.de/rojas/neural/

Software

- ▶ Java-VM für diverse Programmierbeispiele
- ▶ Hades Schaltungssimulator

<http://tams-www.informatik.uni-hamburg.de/applets/hades/>

- ▶ Assembler und Tools, Debugger
- ▶ C-Compiler

- ▶ GNU-Toolchain empfehlenswert: gcc, binutils, gdb
- ▶ Insight-Debugger als Frontend zu gdb
- ▶ Unter Windows: Cygwin mit Development-Tools installieren

<http://www.cygwin.com/>

- ▶ Links und weitere Infos auf der Webseite zur Vorlesung