



# 64-040 Modul InfB-RS: Rechnerstrukturen

[https://tams.informatik.uni-hamburg.de/  
lectures/2016ws/vorlesung/rs](https://tams.informatik.uni-hamburg.de/lectures/2016ws/vorlesung/rs)

– Info zur Organisation –

Andreas Mäder



Universität Hamburg  
Fakultät für Mathematik, Informatik und Naturwissenschaften  
Fachbereich Informatik  
**Technische Aspekte Multimodaler Systeme**

Wintersemester 2016/2017



## 1. Modul InfB-RS: Rechnerstrukturen

## 2. Organisatorisches

Vorlesung

Übungen

Tutorien

Praktikum

Klausur

## 3. Literaturempfehlungen

## 4. Software



## Modulhandbuch

*Dieses Modul behandelt im Rahmen der Vorlesung ein begrenztes und wohl ausgewähltes Theorie- und Methodenrepertoire für die Konfigurierung, den Entwurf, die Realisierung, und die angemessene Nutzung von Rechnern- und Kommunikationsnetzen sowie ihrer Basiskomponenten, u.a. unter Berücksichtigung technologischer, ökonomischer und anwendungsspezifischer Randbedingungen.*

*... das Verstehen der grundlegenden Konzepte, Organisationsformen und Entwurfsmethoden von Rechnerarchitekturen und deren Vernetzung, einschließlich der Betriebssoftware. . .*

[www.inf.uni-hamburg.de/de/studies/bachelor](http://www.inf.uni-hamburg.de/de/studies/bachelor)

WS1	Software-entwicklung I	Mathematik für Studierende der Informatik	Rechnerstrukturen		Informatik im Kontext
SS1	Software-entwicklung II		Meth. komp.	Pro-seminar	Formale Grundlagen der Informatik I
WS2	Algorithmen & Datenstrukturen	Wahlpflicht 24 LP			
SS2	Praktikum	Wahlpflicht 24 LP			
WS3	Wahlpflicht 9 LP	Wahl	Projekt	Semi-nar	
SS3	Wahlpflicht 9 LP	Wahl	Abschlussmodul (Bachelorarbeit)		

# Software-System-Entwicklung BSc

Modul InfB-RS: Rechnerstrukturen

Organisatorisches

Literaturempfehlungen

Software

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

## Einführende Grundlagen

- ▶ Grundbegriffe der Informationsverarbeitung:  
Information, Repräsentation von Information,  
Zahlensysteme, Codierung
- ▶ Rechnerarithmetik:  
Operationen mit Zahlen, Ganzzahl- und Gleitkomma-Arithmetik
- ▶ Konzepte der Digitaltechnik:  
Boolesche Algebra, Schaltnetze, Automatentheorie,  
synchrone und asynchrone Schaltwerke,  
kooperierende Schaltwerke (Operationswerk, Steuerwerk)
- ▶ Hardwarestrukturen:  
Register, Registerbank, Zähler, Speicher,  
Datenpfade, Bus-Systeme

## Rechnerarchitektur

- ▶ von-Neumann-Rechner:  
Prinzipien, Architektur, Befehlssatz
- ▶ ISA-Konzepte: CISC, RISC
- ▶ Maschinen- und Assembler-Code und -programmierung
- ▶ Betriebssystemebene
- ▶ Pipelining, superskalare Architekturen, Parallelrechner
- ▶ Speicherhierarchie:  
Speichertypen, virtueller Speicher, Cache-Organisation
- ▷ Kommunikation, Schnittstellen und Peripheriegeräte
- ▷ IC-Schaltungstechnik, VLSI-Entwurf und -Fertigung

# Feedback erwünscht

maeder@informatik.uni-hamburg.de

- ▶ Zwischenfragen: bitte Feedback bei Unklarheiten etc.!
- ▶ Fehler und Ungenauigkeiten in den Folien und Materialien bitte melden
- ▶ Vorschläge und Hinweise auf Tools, Lehrmaterialien etc. sind immer willkommen!

Problem: stark unterschiedliches Vorwissen!

- ▶ generell: keine speziellen Voraussetzungen
  - ▶ betrifft insbesondere **Rechnerstrukturen**
- ⇒ Geduld, wenn (am Anfang) Stoff schon bekannt
- ⇒ kein Frust, wenn sehr speziell, sondern: Eigeninitiative (Fragen, Tutorium, Übungsgruppen ...)





Dr. Andreas Mäder

maeder@informatik.uni-hamburg.de

+49 40 42883 2502

Informatikum, Haus F-317



- ▶ Mi. 16:15-17:45 Erzwiss. H, VMP-8
- ▶ Fr. 12:15-13:45 Audimax 2, VMP-4 (4 SWS)  
**Rechtshaus**, Rothenbaumch. 33
- ▶ Folien (pdf) **vor der Vorlesung** zum Download
- ▶ Lecture2Go: [lecture2go.uni-hamburg.de/12gos](http://lecture2go.uni-hamburg.de/12gos)
- ▶ diverse gute Lehrbücher — Empfehlungen s.u.
- ▶ Software: JAVA VM, C-Compiler, GNU-Toolchain
- ▶ Informationen und Downloads auf der Webseite — **aktuell!**
- ▶ eingestreute Hinweise auf aktuelle Themen und Vertiefung



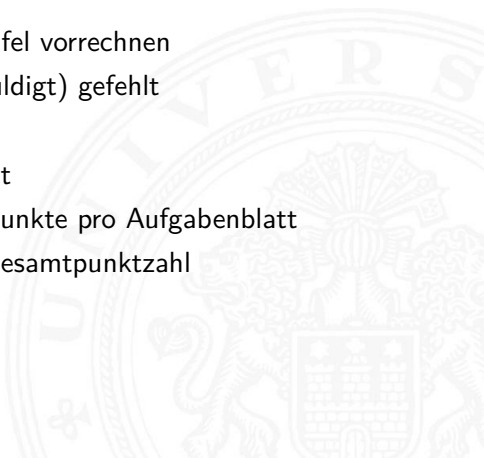
# Übungen

[tams.informatik.uni-hamburg.de/lectures/2016ws/vorlesung/rs/uebung](https://tams.informatik.uni-hamburg.de/lectures/2016ws/vorlesung/rs/uebung)

- ▶ neunzehn Gruppen geplant, Details siehe STiNE/Webseite
- ▶ 1 SWS, wöchentlich
- ▶ Beginn: nächste Woche!
  
- ▶ Diskussion und Nachbereitung der Vorlesung
- ▶ Gruppenarbeit: 2-3 Teilnehmer pro Gruppe
  
- ▶ Übungsaufgaben zum Vertiefen und Erarbeiten des Stoffes
- ▶ Aufgabenblätter jeweils Mittwochs zum Download verfügbar
- ▶ Abgabe der Lösungen: email bis nächsten Mittwoch 24:00 beim Gruppenleiter (zur Not schriftlich im Sekretariat F-311)



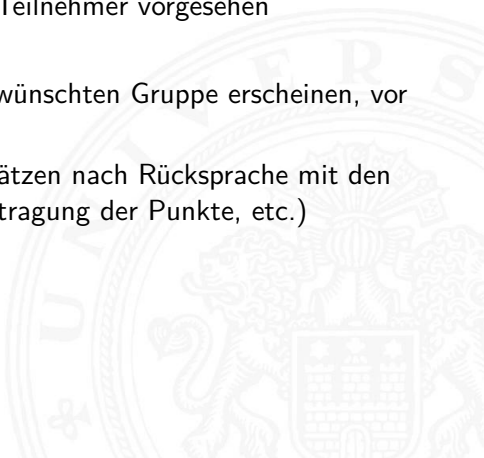
- ▶ [tams.informatik.uni-hamburg.de/lectures/2016ws/vorlesung/rs/doc/scheinkriterien.pdf](https://tams.informatik.uni-hamburg.de/lectures/2016ws/vorlesung/rs/doc/scheinkriterien.pdf)
- ▶ regelmäßige aktive Teilnahme
- ▶ mindestens zweimal an der Tafel vorrechnen
- ▶ höchstens zweimal (unentschuldigt) gefehlt
  
- ▶ alle Aufgabenblätter bearbeitet
- ▶ jeweils mindestens 30 % der Punkte pro Aufgabenblatt
- ▶ und mindestens 50 % der Gesamtpunktzahl





# Übungen: Gruppeneinteilung / -wechsel?

- ▶ Übungen beginnen in der zweiten Vorlesungswoche
- ▶ Anmeldung und Ranking letzte Woche über STiNE
- ▶ derzeit insgesamt 304 Anmeldungen
- ▶ neunzehn Gruppen á 15 (20) Teilnehmer vorgesehen
  
- ▶ 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: Gruppeneinteilung / -wechsel? (cont.)

## Derzeitige Belegung

19.Okt.

Mo	12-13	F-334	20	G-210	20
	13-14	F-334	15	G-210	14
	14-15	F-635	14	G-210	9
	15-16	F-635	20	G-210	20
Di	10-11	F-334	15	G-203	4
	11-12	F-334	10	G-203	9
	12-13	F-334	15	G-210	15
	13-14	F-334	14	G-210	15
Mi	8-9	F-334	15	F-534	15
	9-10	F-334	15		

- ▶ freiwillig, Nacharbeiten von Stoff und Übungen
  - ▶ Diskussion, Fragen: mehr Zeit als in der Übung
- 
1. ab nächster Woche, wöchentlich ein Doppeltermin:  
Donnerstag, 18:30 – 20:30 Uhr, F-132
    - + Live-Videostream:  
[www.youtube.com/channel/UC10j3PauhVSeAUgC39VYMsa](https://www.youtube.com/channel/UC10j3PauhVSeAUgC39VYMsa)
    - + Jabber-Gruppenchat auf dem Mafiasi-Server: [mafiasi.de](https://mafiasi.de)  
(Server: [conference.jabber.mafiasi.de](https://conference.jabber.mafiasi.de) Chatraum: [rstutorium](https://rstutorium))
  2. Vorbereitung auf die Klausur: „*Paniktutorium*“  
Termin: kurz vor der Klausur, Ort und Zeit werden noch bekanntgegeben

- ▶ nicht für Lehramtsstudiengänge
- ▶ im laufenden Semester empfohlen  
und in der vorlesungsfreien Zeit (Februar, März)
- ▶ 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



- ▶ Abschlussprüfung des Moduls Rechnerstrukturen
  - ▶ Note geht in BSc-Zeugnis ein
  
  - ▶ insgesamt zwei Klausurtermine in den Semesterferien
  - ▶ Mo. 20. Feb. 2017 9:30-11:30 Audimax 1 & 2
  - ▶ Mo. 20. März 2017 9:30-11:30 Audimax 1
- pünktlich kommen!**
- ▶ Tipp: möglichst bereits die erste Klausur mitschreiben
  - ▶ zweite Klausur fällt (leider) oft schlechter aus



- ▶ Umfang und Ablauf wie „echte“ Klausur
- ▶ 2-stündige Klausur
  - + anschließende gemeinsame Korrektur/Besprechung
- ▶ Termin: vorletzte Semesterwoche?



[BO15] R.E. Bryant, D.R. O'Hallaron:

*Computer systems – A programmers perspective.*

3rd global ed., Pearson Education Ltd., 2015.

ISBN 978–1–292–10176–7. [csapp.cs.cmu.edu](http://csapp.cs.cmu.edu)

Rechnerarchitektur mit Schwerpunkt Software und Systeme, leider nicht ganz billig. Viele C-Programme und Systemprogrammierung. Beispiele anhand Intel x86 Architektur.

[TA14] A.S. Tanenbaum, T. Austin: *Rechnerarchitektur – Von der digitalen Logik zum Parallelrechner.*

6. Auflage, Pearson Deutschland GmbH, 2014.

ISBN 978–3–86894–238–5

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

[TA13] A.S. Tanenbaum, T. Austin:

*Structured Computer Organization.*

6th edition, Pearson Education Limited, 2013.

ISBN 978–0–273–76924–8

s.o.; die Originalausgabe

- [Mu+09] T. Müller [u. a.]: *Technische Informatik I – Grundlagen der Informatik und Assemblerprogrammierung*.  
3. Auflage, vdf, 2009. ISBN 978-3-7281-3255-0
- [Gu+10] R. Gübeli [u. a.]: *Technische Informatik II – Mikroprozessor-Hardware und Programmier Techniken*.  
2. Auflage, vdf, 2010. ISBN 978-3-7281-3256-7
- [PH16a] D.A. Patterson, J.L. Hennessy: *Computer Organization and Design – The Hardware Software Interface: ARM Edition*.  
Morgan Kaufmann Publishers Inc., 2016.  
ISBN 978-0-12-801733-3
- Schönes Lehrbuch von den Entwicklern der RISC/MIPS Prozessoren.

- [PH16b] D.A. Patterson, J.L. Hennessy: *Rechnerorganisation und Rechnerentwurf – Die Hardware/Software-Schnittstelle*. 5. Auflage, Oldenbourg, 2016. ISBN 978–3–11–044605–0  
Die deutsche Übersetzung; eine Auflage zurück.
- [SS04] W. Schiffmann, R. Schmitz: *Technische Informatik 1 – Grundlagen der digitalen Elektronik*. 5. Auflage, Springer-Verlag, 2004. ISBN 978–3–540–40418–7
- [Sch05] W. Schiffmann: *Technische Informatik 2 – Grundlagen der Computertechnik*. 5. Auflage, Springer-Verlag, 2005. ISBN 978–3–540–22271–2

- [SBH11] W. Schiffmann, H. Bähring, U. Hönig:  
*Technische Informatik 3 – Grundlagen der PC-Technologie.*  
Springer-Verlag, 2011. ISBN 978-3-642-16811-6
- [SSW04] W. Schiffmann, R. Schmitz, J. Weiland:  
*Übungsbuch zur Technische Informatik 1 und 2.*  
3. Auflage, Springer-Verlag, 2004. ISBN 978-3-540-20793-1  
Dutzende von Übungsaufgaben mit detailliert entwickelten Lösungen.
- [MH07] M.J. Murdocca, V.P. Heuring: *Computer architecture and organization – An integrated approach.*  
John Wiley & Sons, 2007. ISBN 978-0-471-73388-1

[HP12] J.L. Hennessy, D.A. Patterson:

*Computer architecture – A quantitative approach.*

5th edition, Morgan Kaufmann Publishers Inc., 2012.

ISBN 978-0-12-383872-8

Die Bibel zum Thema Rechnerarchitektur

[Knu05] D.E. Knuth: *The Art of Computer Programming, Volume 1, Fascicle 1, MMIX — A RISC Computer for the New Millennium.*

Addison-Wesley Professional, 2005. ISBN 978-0-201-85392-6





- [Knu08] D.E. Knuth: *The Art of Computer Programming, Volume 4, Fascicle 0, Introduction to Combinatorial Algorithms and Boolean Functions*. Addison-Wesley Professional, 2008. ISBN 978-0-321-53496-5
- [Knu09] D.E. Knuth: *The Art of Computer Programming, Volume 4, Fascicle 1, Bitwise Tricks & Techniques; Binary Decision Diagrams*. Addison-Wesley Professional, 2009. ISBN 978-0-321-58050-4
- [DM94] G. De Micheli: *Synthesis and Optimization of Digital Circuits*. Mc Graw-Hill, 1994. ISBN 0-07-016333-2.  
[si2.epfl.ch/~demichel/publications/mcgraw](http://si2.epfl.ch/~demichel/publications/mcgraw)

[Lag87] K. Lagemann: *Rechnerstrukturen*.

Springer-Verlag, 1987. ISBN 3-540-17618-7

[Mö103] D.P. Möller: *Rechnerstrukturen: Grundlagen der Technischen Informatik*.

Springer-Verlag, 2003. ISBN 3-540-67638-4

[Mäd11] A. Mäder: *Vorlesung: Rechnerarchitektur und Mikrosystemtechnik*. Universität Hamburg, FB Informatik, 2011, Vorlesungsfolien.

[tams.informatik.uni-hamburg.de/lectures/2011ws/vorlesung/ram](http://tams.informatik.uni-hamburg.de/lectures/2011ws/vorlesung/ram)



[Hyd10] R. Hyde: *The Art of Assembly Language Programming*.  
2nd edition, No Starch Press, 2010. ISBN 978-1-59327-207-4.  
[www.plantation-productions.com/Webster/www.artofasm.com](http://www.plantation-productions.com/Webster/www.artofasm.com)



- ▶ Java VM für diverse Programmierbeispiele
- ▶ HADES Schaltungssimulator — HAMBURG DEsign System  
[tams.informatik.uni-hamburg.de/applets/hades](http://tams.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 [www.cygwin.com](http://www.cygwin.com)
  - ▶ Linux VM mit vorinstallierter Software (VirtualBox, VMware ...)
  
- ▶ Links und weitere Infos auf der Webseite zur Vorlesung