

Organisation und Technologie von Rechensystemen IV / Technische Informatik II

Dr.-Ing. Volkmar Sieh

Institut für Informatik 3 (Lehrstuhl für Rechnerarchitektur)

Tel.: 85-27998

E-Mail: Volkmar.Sieh@informatik.uni-erlangen.de

Büro: 07.138

Organisatorisches

Übungen (10 Gruppen):

- Mo. 10-12 (2.037) und 15:45-17:15 (2.037)
- Di. 14-16 (2.037) und 16:00-17:30 (00.151)
- Mi. 17:00-18:30 (00.151)
- Do. 10-12 (00.152) und 14-16 (00.151) und 14-16 (00.152)
- Fr. 12-14 (2.037) und 14:00-15:30 (2.037)

erste Übungen am Di. 27.4.!

Organisatorisches

Klausur (Vordiplom bzw. studienbegleitend):

findet im Prüfungszeitraum statt (Termin vom Prüfungsamt)

Scheinprüfung:

am Anfang der vorlesungsfreien Zeit

je nach Anzahl der Teilnehmer mündlich bzw. schriftlich

generell:

zugelassen sind Hilfsmittel wie z.B.

- Skript, Bücher
- eigene oder fremde Mitschriften
- Übungsaufgaben, Klausuren usw. mit Lösungen

nicht jedoch

- Laptop, Taschenrechner, Handy usw.

Organisatorisches

Folien, Unterlagen, alte Klausuren usw. im Netz unter

<http://www3.informatik.uni-erlangen.de/Lectures/>

bzw.

<http://univis.uni-erlangen.de/>

Inhalt

1. Teil: Von der Hochsprache zum Assembler-Code

- Speicher, Register, I/O-Register, Adressierungsarten
- Arithmetik (Integer, Floating-Point)
- Kontrollstrukturen, Unterprogramme, Interrupts, Exceptions

2. Teil: Ausführung des Assembler-Codes durch ein Mikroprogramm

- Aufbau Speicher, I/O-Register
- Aufbau einer CPU (ALU, Register, Busse, Tore, Steuerwerk)
- Mikroprogramm „interpretiert“ Assembler-Code, Steuerwerk als Automat

3. Teil: Schutzmechanismen

- Priviligierungs-Stufen
- Segmentierung, Memory Management Unit

Voraussetzungen

Kenntnisse über

- Programmierung in einer problemorientierten Hochsprache (z.B. C, C++, Java, Pascal)
- Darstellungen von Zahlen (speziell 2er- und 16er-System)
- AND-, OR-, XOR- und NOT-Gatter, Register, Schalter
- **!!!! persönlicher Einsatz („Learning by Doing“) !!!!**

Aufgaben

!!! Aufgaben sind Prüfungs-relevant !!!

Teil 1: Hochsprache - Assembler

- Aufgaben für z.B. Intel- oder Motorola-Prozessoren
- Aufgaben für Mikro-Controller

Teil 2: Assembler - Mikroprogramm

- Aufgaben zur Hardware einer CPU
- Aufgaben zur Mikroprogrammierung

Teil 3: Schutzmechanismen

- Aufgaben für z.B. Intel-Prozessoren

Literatur

Es existieren viele Bücher über

- Assembler-Programmierung
- Hardware-Komponenten und Rechnerarchitektur
- Mikroprogrammierung.

alles Prüfungs-Relevante steht auch auf Folien / Aufgabenblättern

Sinnvoll: Blick in ein gutes Handbuch eines Prozessors

Literatur

Beispiele:

- H. Bähring, „Mikrorechner-Systeme“, Springer, 1991
- G. Karam, J. Bryant, „Prinziples of Computer Design“, Prentice-Hall, 1992
- W. Giloi, „Rechnerarchitektur“, Springer, 1993
- M. Mano, „Computer System Architecture“, Prentice-Hall, 1993
- J. Hennessy, D. Patterson, D. Jungmann, „Rechnerarchitektur“, Vieweg, 1994
- T. Flik, H. Liebig, „Mikroprozessortechnik“, Springer, 1994
- K.-U. Witt, „Elemente des Rechneraufbaus, Hanser, 1995
- P. Herrmann, „Rechnerarchitektur“, Vieweg, 1998
- W. Stallings, „Computer Organization & Architecture“, Prentice Hall, 2003
- ...