



**Klausur Technische Grundlagen der Informatik  
Studiengang Medieninformatik**  
H. Linnemann

Montag, 16. Juli 2007, 12.00 - 14.00, Raum B301

- Zugelassene Hilfsmittel: Keine.
- Versehen Sie bitte jedes Lösungsblatt mit Ihrem Namen und mit einer fortlaufenden Seitennummer.
- Falls Teillösungen über mehrere Seiten verteilt sind, versehen Sie diese bitte mit entsprechenden Querverweisen. Nicht gekennzeichnete oder nicht eindeutig zugeordnete Lösungsfragmente werden nicht gewertet!
- Reklamationen der Korrektur und Bewertung nur bei Klausurrückgabe!

Name:  **chip273**

Vorname:

Matrikel-Nr.

Unterschrift:

Aushändigen der korrigierten Klausur (zutreffendes bitte ankreuzen):

- Nur an mich persönlich oder an Kommilitonen/innen mit schriftlicher Vollmacht
- An Frau / Herrn:
- An alle, die danach fragen

Dritter (letzter zulässiger) Versuch:

- Nein. ✓
- Ja.

**Fehlende Antworten erweitert  
für 100 Punkte. Extras in Blau.**

Frage	Max. Punkte	Erreichte Punkte
Frage 1	6	6
Frage 2	8	8
Frage 3	14	14
Frage 4	15	14
Frage 5	8	4
Frage 6	9	9
Frage 7	18	8
Frage 8	12	12
Frage 9	10	10
<b>SUMME</b>	<b>100</b>	<b>85</b>

NOTE: **1,7**



766C

- 1.) Wandeln Sie die Oktalzahl 73154 in das Dual- und das Hexadezimalsystem!
- 2.) Es sind zwei Hexadezimalzahlen gegeben:  $a = A3C_{16}$ ,  $b = 35A_{16}$ .
  - a) Berechnen Sie im hexadezimalen Zahlensystem  $a + b$  und  $a - b$ .
  - b) Führen Sie die Subtraktion zusätzlich im 16er Komplement aus.
- 3.) Stellen Sie die Dezimalzahl 3,25 dual im Fließkommaformat dar. Verwenden Sie das 32-Bit-Fließkommaformat mit 1 Bit Vorzeichen, 8 Bit Exponent mit Offset 127 und 23 Bit Fraktion.
- 4.) Ein Komparator vergleicht zwei 1-Bit-Werte ( $a$  und  $b$ ) und liefert als Ergebnis  $(a=b)$ ,  $(a<b)$  oder  $(a>b)$ .
  - a) Erstellen Sie die Funktionstabelle für den Komparator.
  - b) Ermitteln Sie die Funktionen für  $(a=b)$ ,  $(a<b)$  und  $(a>b)$ .
  - c) Erstellen Sie für die drei Funktionen die entsprechenden Schaltungen.
- 5.) Ein zentraler Begriff der Informatik ist die sog. Von Neumann Architektur.
  - a) Nennen Sie vier wesentliche Eigenschaften des von Neumann-Rechners!
  - b) Was versteht man unter dem „von Neumann’schen Flaschenhals“?
  - c) Wie kann man ihn vermeiden und wie heißt diese Rechnerarchitektur?
- 6.) Interrupts von Intelprozessoren können in drei Klassen eingeteilt werden. Nennen Sie die drei Klassen, die Art ihrer Auslösung und geben Sie jeweils ein Anwendungsbeispiel.
- 7.) Aktuelle Grafikkarten verfügen über 3D-Beschleuniger zur Unterstützung des Rendering.
  - a) Was versteht man unter Rendering?
  - b) Nennen Sie die drei Stufen des Ebenenmodells der so genannten Rendering-Pipeline für 3D-Darstellungen und charakterisieren Sie die in den einzelnen Stufen vorgenommenen Arbeiten.
- 8.) Hinsichtlich der Anzahl der gleichzeitig zu verarbeitenden Daten und gleichzeitig ausführbaren Befehle können nach Flynn vier Grundtypen von Rechnern unterschieden werden. Nennen Sie die vier Grundtypen (sowohl die Abkürzung als auch die „Langform“) sowie die Art der Abarbeitung von Befehls- und Datenstrom.

Rasterizer

9.) Aufgabe ans Ende vom PDF verschoben.

1) Octal 23154  
 Binär 111 011 001 101 100  
 0111 0110 0110 1100 ✓  
 Hex 7 6 6 C ✓ 4/6  
 766C

2) a = A3C<sub>16</sub>  
 a) b = 35A<sub>16</sub>  
 (a+b) A3C<sub>16</sub>  
 + 35A<sub>16</sub>  
 -----  
 D96<sub>16</sub> ✓  
 (a-b) A3C<sub>16</sub>  
 - 35A<sub>16</sub>  
 -----  
 6E2<sub>16</sub> ✓

2b) FFF  
 - 35A  
 -----  
 CA5  
 1  
 +  
 -----  
 CAB ✓  
 (a-b) A3C  
 + CAB  
 -----  
 16E2<sub>16</sub> ✓ 8/8

3) 3,25<sub>10</sub> = 11,01<sub>2</sub> binär ✓  
 V2 = +

1,101 × 2<sup>1</sup> ✓  
 hidden bit 127 ✓  
 exponent = 128<sub>10</sub> → 10000000<sub>2</sub> ✓  
 Fraktion/Mantisse ✓  
 01000000101000000000000000000000 ✓  
 exp: nicht sicher könnte 0000001 sein. Nein! ✓

14/14

4) a)

a	b	$(a=b)$	$(a < b)$	$(a > b)$	Ergebnis
0	0	1	0	0	1
0	1	0	1	0	1
1	0	0	0	1	1
1	1	1	0	0	1

1 und  
V oder

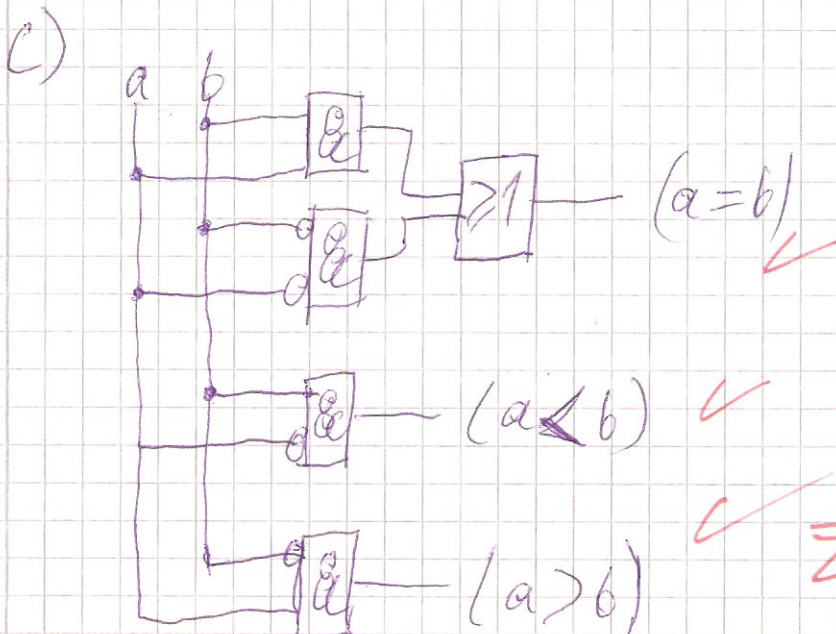
b)

$$(a \wedge b) \vee (\bar{a} \wedge \bar{b}) = (a=b) = \boxed{a \odot b}$$

$$(\bar{a} \wedge b) = (a < b)$$

$$(a \wedge \bar{b}) = (a > b)$$

3/4



7/7  
Σ 14/15

a) 0/4

5) b) Der „vonNeumannsche Flaschenhals“ beschreibt das Problem, das in der Rechnerarchitektur das Überarbeiten von Befehlen an die andauernden Speicherlese- und Schreibprozesse geknüpft sind. Für jeden Rechenbefehl sind zwei Speicherzugriffe notwendig. Daher kann ein vonNeumannscher Rechner nicht die erwartete Leistung erzielen. 1/2

c) Harvard Cache einsetzen, für beschl. des speichers 2/2  
Σ 4/8

5a)

- Gemeinsamer Speicher für Daten und Befehle
- Gemeinsamer Daten- und Befehls- bus
- Sequenzielle Programmabarbeitung
- flexible Speicheraufteilung

5b)

Gemeinsamer Speicher und Datenbus daher langsame verarbeitung.  
CPU(ALU) muss immer auf den Speicher warten.

5c)

Harward Architektur, oder eine Mischung aus beidem benutzen.  
Einen schnelleren Speicher benutzen (z.B. in form von Cache)  
[Der jedoch aufgrund von Kosten langsamer sein muss]



ausgelöst durch externe ereignisse ✓

6) I) Hardwareinterrupts: z.B. maus, tastatur, timer, usw. ✓

II) ~~Software~~ Exceptions / Fehler: ausgelöst durch z.B. ~~exceptions~~ fehler in den programmen ✓ division durch null ✓

III) ~~Betriebssystem~~ Software Interrupts: Befehle des Betriebssystems ✓ zB int 21H für die DOS textausg. ✓

9/9

z) a) Rendering wird das Erzeugen von Bildern und Animationen mit Hilfe eines Computers genannt.  
Hierfür wird meistens spezielle Hardware eingesetzt.  
Heutzutage "Grafikkarte",

oder z.B. <http://de.wikipedia.org/wiki/Rendering>

v) I) Pixelizer (?) f. od. 3D Formen  
Aus den Koordinaten ~~wird~~ das Aussehen der Objekte bestimmt.

II) Texturizer f. od. 3D Formen  
Die Objekte werden mit "Texturen" überzogen.  
dadurch erhalten sie eine Oberfläche. Hier sollte auch Beleuchtung eingesetzt werden.

III) Rasterizer:   
Darstellung wird für die Pixel des Monitors aufbereitet.

## I Anwendungsebene

- Interaktion mit dem Anwender
- Ablaufsteuerung
- Kollisionsbehandlung ( in 2007 noch. Physik, nicht mehr Lange...

## II Geometrie

- Transform & Lighting, Projektion, Clipping
- Umsetzung / Projektion von 3D auf 2D
- Raytacing
- Z-Puffer berechnung (Tiefe)

## III Rasterizer

- Setzen der Pixel im Front-Buffer
- Berücksichtigung verdeckter und transparenter Pixel
- Texturen ( "Tapezierung" der Gitternetzmodelle )

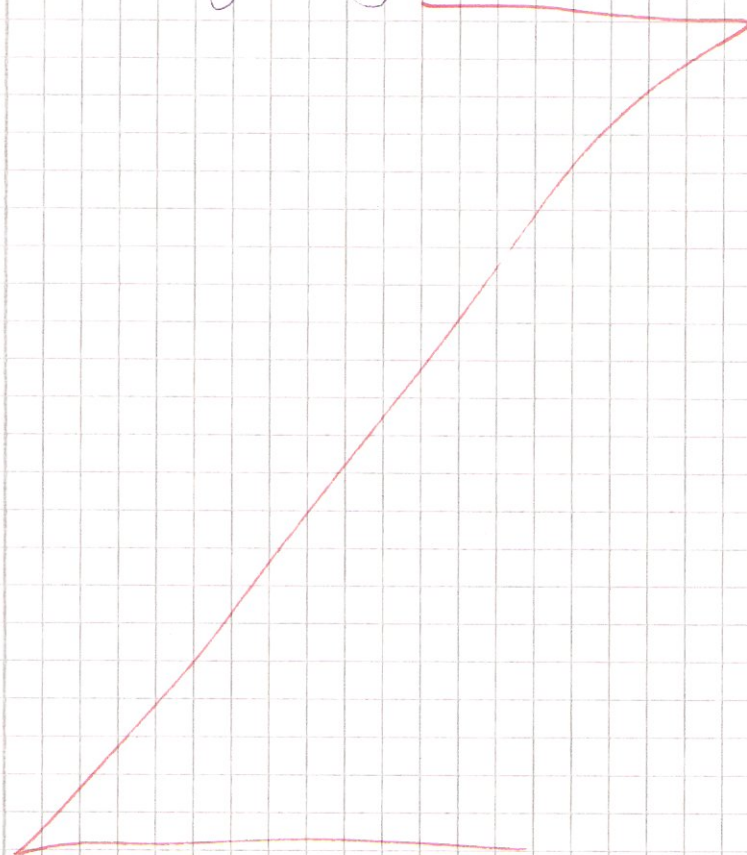
8) SISD ✓ Single Instruction Single Data ✓  
~~Each~~ Jeder Datenbyte wird mit einem Befehl  
bearbeitet. ✓

SIMD ✓ Single Instruction Multiple Data ✓  
Mit einem Befehl können mehrere Daten gleichzeitig  
bearbeitet werden. ✓

MISD ✓ Multiple Instruction Single Data ✓  
Ein Datenbyte/Datenwort wird mit mehreren Befehlen  
bearbeitet. Spart das immer wieder <sup>zwischen</sup> Kohrende Speicher  
der Daten. ✓

MIMD ✓ Multiple Instruction Multiple Data ✓  
Mehrere Datenwörter werden zugleich mit mehreren  
Wörtern bearbeitet. ✓  
12/12

9) Aufgabe 9 in der Tabelle ✓





- 9.) Gegeben ist der folgende Ausschnitt eines Assemblerprogramms  
 (in einem durch ein \$-Zeichen begrenzten String werden enthaltene Kleinbuchstaben in  
 Grossbuchstaben umgewandelt;  
 die ASCII-Codes sind: 'A' = 41H, 'a' = 61H, 'B' = 42H, 'b' = 62H, 'c' = 63H, 'C' = 43H, 'z' =  
 7AH, '\$' = 24H).

Befehl	Register							
	SI	DL	SI	DL	SI	DL	SI	DL
	1. Durch- lauf	2. Durch- lauf	3. Durch- lauf	4. Durch- lauf				
. . .								
.DATA								
STRING DB 'Abc\$' ;Der OFFSET von String ist 0!								
.CODE								
MOV AX,@DATA			/		/		/	
MOV DS,AX			/		/		/	
MOV SI, OFFSET STRING	00H ✓		/		/		/	
AUSGABE:	//							
MOV DL, [SI]	//	41H ✓	01H ✓	62H ✓	02H ✓	63H ✓	03H ✓	24H ✓
CMP DL, '\$' 24H	//							
JE ANZEIGE	//							
CMP DL, 'a' 61H	//							
JL KEIN_KLEINBUCH	//							
CMP DL, 'z' 7AH	//							
JNBE KEIN_KLEINBUCH	//							
SUB DL, 20H	//			42H ✓		43H ✓		
MOV [SI], DL	//							
KEIN_KLEINBUCH:	//							
INC SI	01H ✓		02H ✓		03H ✓			
JMP AUSGABE								
ANZEIGE:								
. . .								

10/10

Tragen Sie die NACH der Ausführung der einzelnen Befehle in den Registern vorhandenen Werte in die Tabelle ein.