Technische
Fachhochschule
Berlin

Klausur Technische Grundlagen der Informatik
Studiengang Medieninformatik
H. Linnemann

Freitag, 06. Februar 2009, 10.00 Uhr, Raum D E15

- 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: .....

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.

Frage	Max. Punkte	Erreichte Punkte
Frage 1	8	8
Frage 2	14	14
Frage 3	15	15
Frage 4	16	14
Frage 5	9	9
Frage 6	18	18
Frage 7	10	10
Frage 8	10	10
SUMME	**100**	98

NOTE: 1,0

1.) Es sind zwei Hexadezimalzahlen gegeben: $a = B3D_{16}$, $b = 28A_{16}$.

a) Berechnen Sie im hexadezimalen Zahlensystem a + b und a – b.

b) Führen Sie die Subtraktion zusätzlich im 16er Komplement aus.

2.) Stellen Sie die Dezimalzahl 19,25 dual im Fließkommaformat dar.
Verwenden Sie das 32-Bit-Fließkommaformat mit 1 Bit Vorzeichen, 8 Bit Exponent mit Offset 127 und 23 Bit Fraktion.

3.) 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.

4.) Zentrale Begriffe der Informatik sind die sog. Von Neumann - Architektur und die Harvard - Architektur.

a) Wodurch ist die Von Neumann - Architektur gekennzeichnet?

b) Wodurch ist die Harvard - Architektur gekennzeichnet?

c) Zeichnen Sie für die Von Neumann – Architektur ein Blockschaltbild, das das Zusammenwirken von Speicher, Ein-/Ausgangseinheiten, Steuereinheit und ALU darstellt.

5.) Interrupts von Intelprozessoren können in drei Klassen eingeteilt werden. Nennen Sie die drei Klassen, die Art ihrer Auslösung und geben Sie jeweils ein Anwendungsbeispiel.

6.) 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.

7.) Nach der Umfang und Komplexität des Befehlsvorrats unterscheidet man zwei Grundtypen von Prozessoren.

a) Nennen Sie die beiden Grundtypen (sowohl die Abkürzung als auch die „Langform").

b) Stellen Sie die wesentlichen Eigenschaften sowie Vor- und Nachteile der beiden Grundtypen gegenüber.

8.) Gegeben ist der folgende Ausschnitt eines Assemblerprogramms
(in einem durch ein $-Zeichen begrenzten String werden enthaltene Kleinbuchstaben in Grossbuchstaben umgewandelt;
die ASCII-odes sind: 'a' = 61H, 'D' = 44H, 'd' = 64H, 'E' = 45H, 'e' = 65H, 'F'= 46H, 'f' = 66H, 'z' = 7AH, '$' = 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 'Def$' ;Der OFFSET von String ist 0!								
.CODE								
MOV AX,@DATA								
MOV DS,AX								
MOV SI, OFFSET STRING	00H							
AUSGABE:								
MOV DL, [SI]		44H		65H		66H		24H
CMP DL, '$'								
JE ANZEIGE								
CMP DL, 'a'								
JL KEIN_KLEINBUCH								
CMP DL, 'z'								
JNBE KEIN_KLEINBUCH								
SUB DL, 20H				45H		46H		
MOV [SI], DL								
KEIN_KLEINBUCH:								
INC SI	01H		02H		03H			
JMP AUSGABE								
ANZEIGE:								
. . .								

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

Aufgabe 1.)

a = $B3D_{16}$ b = $28A_{16}$

A = 10
B = 11
C = 12
D = 13
E = 14
F = 15

a) a + b

$$\begin{array}{r} B3D_{16} \\ +\ 28A_{16} \\ \hline DC7_{16} \end{array}$$ ✓

a − b

$$\begin{array}{r} B3D_{16} \\ -\ 28A_{16} \\ \hline 8B3_{16} \end{array}$$ ✓

b) 15er-Komplement von $28A_{16}$

$$\begin{array}{r} FFF_{16} \\ -\ 28A_{16} \\ \hline D75_{16} \end{array}$$ ✓

16er Komplement von $28A_{16}$

$$\begin{array}{r} D75_{16} \\ +\ 001_{16} \\ \hline D76_{16} \end{array}$$ ✓ ≙ 16er-Komplement

a + (−b) = $8B3_{16}$

$$\begin{array}{r} B3D_{16} \\ +\ D76_{16} \\ \hline 18B3_{16} \end{array}$$ ✓

8/8

Aufgabe 2.)

$19,25_{10}$ 32 Bit 1 Bit Signum, 8 Bit Exponent, 23 Bit Fraction

Offset 127

1. Vorkommastellen wandeln

19 / 2 = 9 Rest 1 ↑
9 / 2 = 4 Rest 1
4 / 2 = 2 Rest 0
2 / 2 = 1 Rest 0
1 / 2 = 0 Rest 1

19_{10} ≙ 10011_2

2. Nachkommastellen wandeln

$0,25 \cdot 2 = 0,5$

$0,5 \cdot 2 = 1,0$ $\quad 0,25_{10} \triangleq 0,01_2$

3. Zusammensetzen

$19,25_{10} \triangleq 10011,01_2$

4. Exponent bestimmen

$10011,01_2 \triangleq 1,001101_2 \cdot 2^4$

Offset + Exponent: $127 + 4 = 131$

131 / 2 = 65 Rest 1

65 / 2 = 32 Rest 1

32 / 2 = 16 Rest 0

16 / 2 = 8 Rest 0

8 / 2 = 4 Rest 0

4 / 2 = 2 Rest 0

2 / 2 = 1 Rest 0

1 / 2 = 0 Rest 1

$131_{10} \triangleq 1000\ 0011_2$

5. Vorzeichen bestimmen

positiv → 0

6. Zusammensetzen

0	10000011	00110100...0

S Exponent Fraction mit 0 aufgefüllt auf 23 Bit.

14/14

Aufgabe 3.)

a)

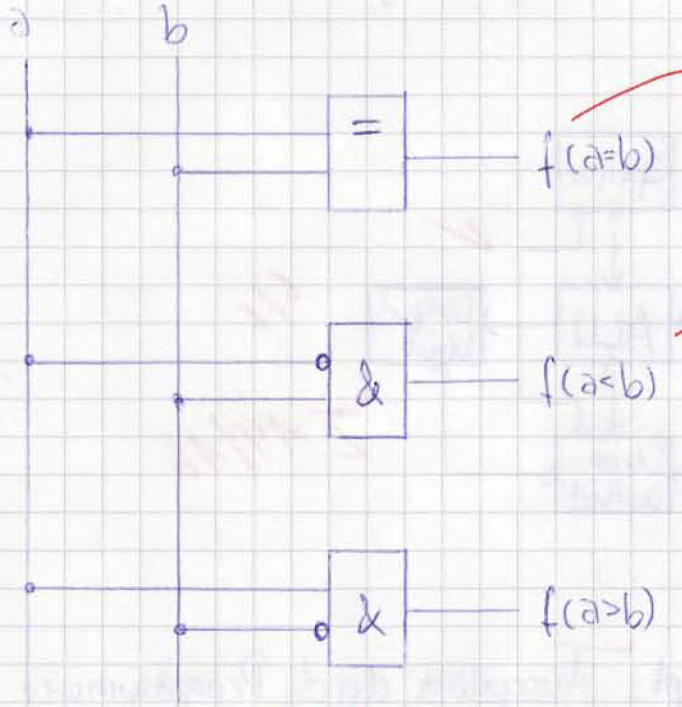
a	b	a = b	a < b	a > b
0	0	1	0	0
0	1	0	1	0
1	0	0	0	1
1	1	1	0	0

b) $f(a=b)$: $\bar{a}\cdot\bar{b} + a\cdot b = a \odot b = (a \oplus b)'$

$f(a<b)$: $\bar{a}\cdot b$

$f(a>b)$: $a\cdot\bar{b}$

c)



15/15

Aufgabe 4.)

Von Neumann-Architektur

a) 1) Nur ein Bus für Daten und Befehle —
2) Daten und Befehle in einem Speicher —
3) flexible Speicheraufteilung für Daten und Befehle
4) Sequentielle Programmabarbeitung —

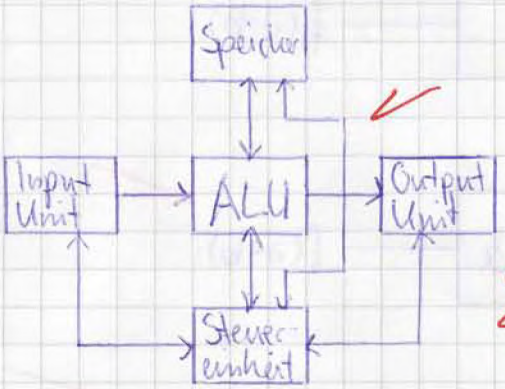
4/5 —

b) Harvard-Architektur

1) Jeweils ein Bus für Daten und Befehle
2) Daten und Befehle in getrennten Speichern
3) Daher: feste Speichergröße
4) Sequentielle Programmabarbeitung

4/5 —

c)



6/6

Σ 14/16

Aufgabe 5.)

1. Software-Interrupts: Ausgelöst durch Programmierer. Ähnlich einem Unterprogrammaufruf. Bsp.: INT 21H DOS-Funktionsaufruf
2. Exceptions: Ausgelöst vom Prozessor selbst. Bsp.: Interrupt Nr. 0 bei Division durch 0. —
3. Externe (asynchrone) Interrupts: Ausgelöst durch externe Ereignisse, Bsp.: Maus, Tastatur (Taste gedrückt), Zeitgabebaustein (Timer abgelaufen)

9/9

Aufgabe 6.)

a) Rendering: Bilderzeugung am Rechner aus geometrischen Modellen ✓ 2/2

b)

- Beschreibung der 3D-Welt ✓
- Kamerafahrten, Design, Drehbuch, Bewegungen, Script

CPU 1. **Application**

- Interaktion mit dem Anwender („kurzer Weg")
- Kollisionsberechnung
- Generierung von Rendering-Primitiven
- Polygone (Dreiecke, Linien, Punkte) beschrieben durch Vertices (Eckkoordinaten)

CPU / GPU 2. **Geometry**

- Transform: Modell- und Sichttransformierungen (Kamerabereich) in eine 3D-Welt / 3D-Koordinaten
- Lighting: Beleuchtungsrechnung (Lichteffekte, Reflexionen)
- Project: Transformierung in 3D-Kamerakoordinaten (3D-Einheitswürfel)
- Clipping: Abtrennen des Objekte außerhalb des Sichtbereichs der Kamera
- Screen Mapping: Transformierung in 2D-Kamerakoordinaten
- 2D-Raum mit z-Werten (Tiefenwerte der Polygone)

GPU 3. **Rasterizer**

- Sichtbarkeitsberechnung (z-Buffer)
- Einrechnen von Texturen, Texturing („Tapezieren" der 2D-Polygone)
- Setup: Umsetzen der 2D-Polygone in Pixel

- Front-Buffer

Monitor

16/16

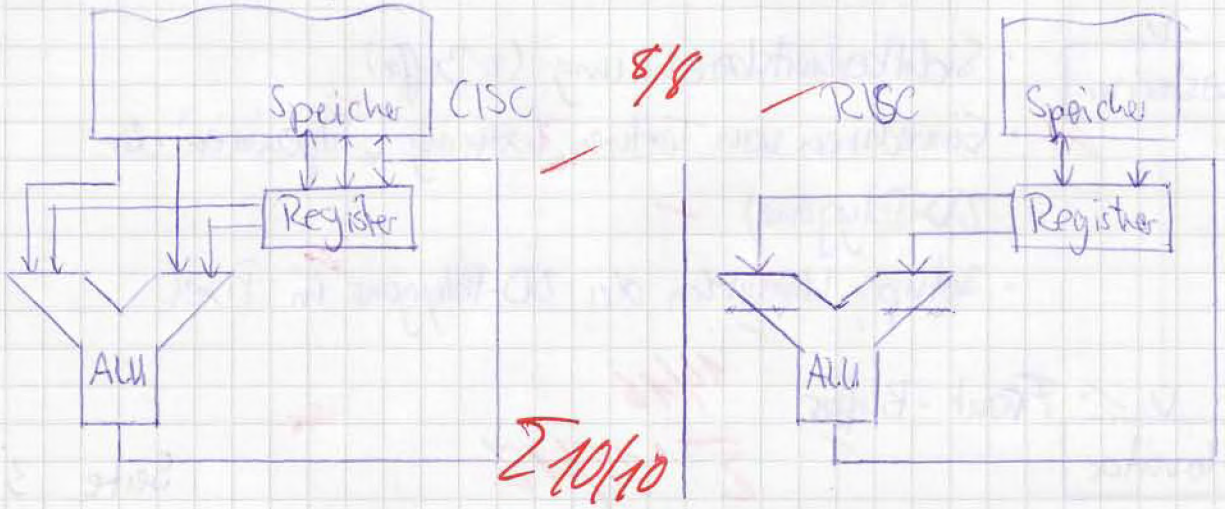
Σ 18/18

Aufgabe 7.)

a)

CISC - Complex Instruction Set Computer Register-Speicher-Architektur ✓ 4/2	RISC - Reduced Instruction Set Computer Register-Register Architektur ✓
1) kompakter Code -	Code weniger kompakt
2) Alle Befehle können auf den Speicher (langsam) zugreifen -	Nur Load / Store - Befehle ✓ können auf den Speicher zugreifen
3) Komplexität liegt im Mikroprogramm	Komplexität liegt im Compiler ✓
4) Instruktionen interpretiert durch Mikroprogramm ✓	Instruktionen ausgeführt durch festverdrahtete Hardware -
5) Instruktionen~~en~~ mit variabler Länge ✓	Instruktionen mit fester Länge -
6) einfacher Registersatz ✓	umfangreiche Registersätze -
7) viele Befehle und Adressierungs-arten -	wenige Befehle und Adressierungsarten -
8) Umfangreiche Befehle, benötigen mehrere Taktzyklen -	kurze Befehle, benötigen in der Regel nur 1 Taktzyklus -
9) Befehle: Register-Register Register-Speicher ✓ Speicher-Speicher	Befehle: Register-Register

RISC ist im Allgemeinen doppelt so schnell, da die Befehle ausschließlich in den schnellen Registern ausgeführt werden. Heutzutage gibt es meist Mischformen.



8/8

Σ 10/10