Lösungsvorschläge zur Übungsklausur

zum Kurs 1708 „Technische Informatik II“

im WS ’01/02

Aufgabe 1: (10 Punkte)

Welche der folgenden Aussagen sind richtig ?

		ja	nein
a)	Unter dem Begriff „Mikroprogramm“ versteht man jedes Maschinenprogramm, das von einem Mikroprozessor ausgeführt werden kann. *s. KE1, S. 24*		X
b)	Eine Unterbrechungsanforderung, die durch die Bearbeitung eines ungültigen Befehlscode auftritt, heißt Software-Interrupt. *s. KE2, S. 15*		X
c)	Bei der indizierten Adressierung zeigt die im Befehl spezifizierte Adresse nicht unmittelbar auf den Operanden selbst, sondern auf ein Speicherwort, in dem die eigentliche Operandenadresse zu finden ist. *s. KE2, S. 63*		X
d)	RISC-Prozessoren sind nur dadurch gekennzeichnet, daß sie einen sehr kleinen Befehlssatz besitzen. *s. KE3, S. 3*		X
e)	Bei der Ausführung von logischen Befehlen (AND, OR, NOT, EOR) werden die logischen Werte ‘0’ bzw. ‘1’ immer durch $00 bzw. $01 repräsentiert. *s. KE 2, S. 43*		X

Aufgabe 2: (10 Punkte)

a) Unter einem „**Trap**“ versteht man *eine Ausnahmesituation, die durch bestimmte prozessorinterne Ereignisse verursacht wird und synchron zum Programmablauf auftritt. Dabei kann es sich um einen Fehler oder anomale Prozessorzustand im Zusammenhang mit einem Befehl handeln. (s. KE1, S. 15)*

b) Unter dem Begriff „**Lokalitätseigenschaft von Programmen** “ versteht man *die Beobachtung, daß zu jedem Zeitpunkt während des Programmlaufs bestimmte Befehle und Daten bevorzugt und wiederholt angesprochen werden. (s. KE5, S. 24)*

c) Unter dem Begriff „**Pipeline-Hemmnis**“ versteht man *alle Ereignisse, die verhindern, daß in jedem Taktzyklus ein Befehl die Pipeline vollständig ausgeführt verlassen kann. Dazu gehören insbesondere die Datenabhängigkeiten und verzögert ausgeführte Verzweigungsbefehle. (s. KE3, S. 17)*

d) Adressierungsarten, bei denen die im Befehl spezifizierte Adresse nicht unmittelbar auf den Operanden selbst zeigt, sondern erst auf eine Speicherzelle, unter der die eigentliche Operandenadresse zu finden ist, werden als

indirekte oder *zweistufige Speicher*-Adressierung

bezeichnet. *(s. KE2, S. 67)*

e) Ein Verfahren, bei dem der Datentransfer zwischen Speicher und Peripheriekomponenten ohne Einsatz der CPU durchgeführt wird, heißt

direkter Speicherzugriff, *Direct Memory Access - DMA*.

(s. KE5, S. 11)

Aufgabe 3:

a) 1.

	W3	W2	W1	W0	
R1:	6AE3	03D2	835A	7F34	
R2:	12F0	A234	AF06	1707	
	7DD3	A606	13260	963B	Übertrag 1 in W1
	⇓	⇓	⇓	⇓	wird abgeschnitten (Wrap-around)
R1:	7DD3	A606	3260	963B	

R1 : $7DD3A6063260963B

2.

	B7	B6	B5	B4	B3	B2	B1	B0	
R1:	6A	E3	03	D2	83	5A	7F	34	
R2:	12	F0	A2	34	AF	06	17	07	
	7C	1D3	A5	106	132	60	96	3B	
	⇓	⇓	⇓	⇓	⇓	⇓	⇓	⇓	B3 zum kleinsten negativen, B1 zum größten positiven Wert gesättigt
R1:	7C	D3	A5	06	80	60	7F	3B	

R1 : $7CD3A50680607F3B

b)

R1 = (DW1, DW0) → 6 bit rechts → R1’ = (DW1’, DW0’)

DW1: 10000111010101100101010000110010 →
DW1’: 11111110000111010101100101010000 = $ FE1D5950

DW0: 11110010001101000000000011110101 →
DW0’: 11111111110010001101000000000011 = $ FFC8D003

R1’ = $FE1D5950FFC8D003

Bemerkung: DW1 und DW0 sind negativ (MSB=1). Deshalb wird beim arithmetischen Rechtsschieben als Vorzeichen die ‘1’von links nachgezogen. Ein Übertrag zwischen den Wörtern findet nicht statt.

Aufgabe 4: (15 Punkte)

a) Geben Sie für das beschriebene Format den darstellbaren Zahlenbereich an:

$$-1_{10} = 1.000.....00_2 \leq Z \leq 0.111.....11_2 = 1 - 2^{-15} < 1$$

kleinste Zahl: $-1 + 0*2^{-1} + ... + 0*2^{-15}$, größte Zahl: $1*2^{-1} + ... + 1*2^{-15} = 1 - 2^{-15}$

b) Welche Dezimalzahlen werden durch folgende, in hexadezimaler Darstellung gegebenen Festpunkt-Zahlen (Format 1.15) repräsentiert:

Z1 = \$7000 = $2^{-1} + 2^{-2} + 2^{-3} = 0.875$

Z2 = \$5B00 = $2^{-1} + 2^{-3} + 2^{-4} + 2^{-6} + 2^{-7} = 0.7109375$

Z3 = \$9000 = $-1 + 2^{-3} = - 0.875$

Z4 = \$FFFF = $-1 + 2^{-1} + ... + 2^{-15} = -1 + 1 - 2^{-15} = - 2^{-15} = - 0.000030517578125$

c) Stellen Sie die folgenden Zahlen als 16-bit-Festpunktzahlen in hexadezimaler Form dar:

Z1 = - 0.5625 = -1 + 0,4375 = $-1 + 2^{-2} + 2^{-3} + 2^{-4}$ = \$B800

Z2 = - 1 = \$8000

Z3 = 0.5625 = $2^{-1} + 2^{-4}$ = \$4800

Z4 = 0,0625 = 2^{-4} = \$0800

.

Aufgabe 5: (30 Punkte)

a)

Zugriff	Variable	hit/miss	Weg 3	Weg 2	Weg 1	Weg 0	B2	B1	B0
0	-		(frei)	(frei)	(frei)	(frei)	0	0	0
1	A	miss	**A**	(frei)	(frei)	(frei)	1	1	0(u)
2	B	miss	A	(frei)	**B**	(frei)	0	1(u)	1
3	A	hit	A	(frei)	B	(frei)	1	1	1(u)
4	C	miss	A	(frei)	B	**C**	0	1(u)	0
5	D	miss	A	**D**	B	C	1	0	0(u)
6	A	hit	A	D	B	C	1	1	0(u)
7	E	miss	A	D	**E**	C	0	1(u)	1
8	C	hit	A	D	E	C	0	1(u)	0
9	F	miss	A	**F**	E	C	1	0	0(u)
10	B	miss	A	F	**B**	C	0	0(u)	1
11	C	hit	A	F	B	C	0	0(u)	0

b)

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17	16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1	0
T A G	**I N D E X**	**B y t e**

Tag: Bits 31 - 17 Index: Bits 16 - 7 Byte-Adr.: Bits 4-0

Anzahl der Bytes pro Eintrag (*line length*):	$2^5 = 32$
Anzahl der Einträge pro Weg:	$2^{12} = 4096$ Einträge pro Weg
Kapazität der einzelnen Cache-Wege:	$2^{12} * 2^5 = 4096 * 32 = 128$ kbyte
Gesamtkapazität des Caches:	4 * 128 = 512 kbyte

c)

Speicheradresse	Tag	Index	Byte-Adresse
$CBC3 9AB1	65E1	CD5	11
$4635 9AA5	231A	CD5	05
$D965 D236	6CB2	E91	16

Nur die 1. und 2. Adresse belegen einen Eintrag mit derselben Nummer (Index), und zwar den Eintrag Nr. $CDF = 3295. (Dies ist der 3296. Eintrag, da mit 0 begonnen wird.)

„Nebenrechnung“: bitweise Darstellung der vorgegebenen Zahlen

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	0	0	1	0	1	1	1	1	0	0	0	0	1	1	1	0	0	1	1	0	1	0	1	0	1	1	0	0	0	1
0	1	0	0	0	1	1	0	0	0	1	1	0	1	0	1	1	0	0	1	1	0	1	0	1	0	1	0	0	1	0	1
1	1	0	1	1	0	0	1	0	1	1	0	0	1	0	1	1	1	0	1	0	0	1	0	0	0	1	1	0	1	1	0

Aufgabe 6: (20 Punkte)

a)

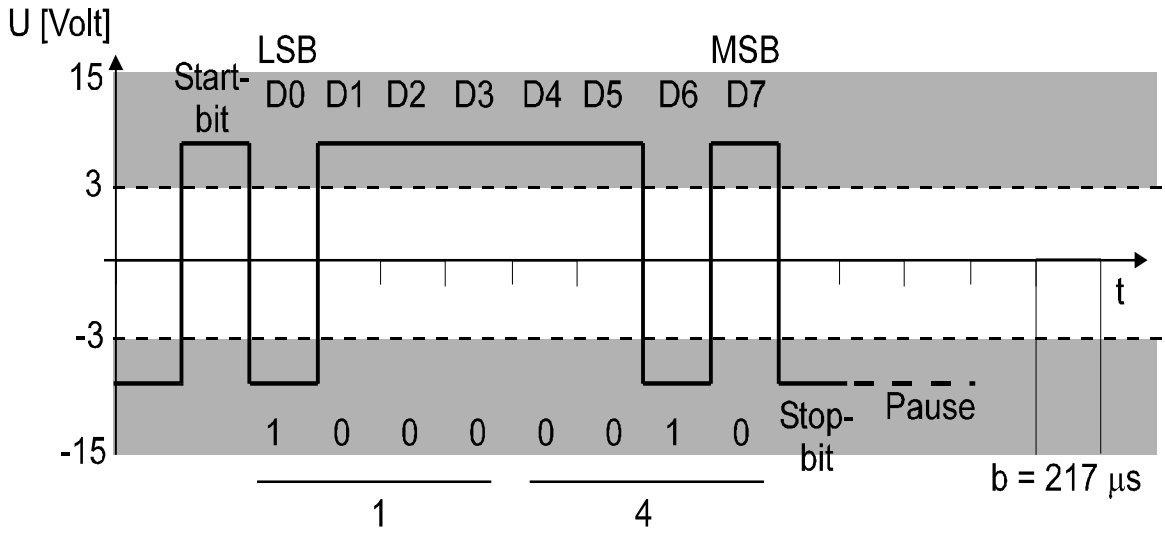
- 8 Datenbits, 1 Stopbit
- 4800 baud, interner Empfängertakt

CR = 0001 1100 (binär) = $1C

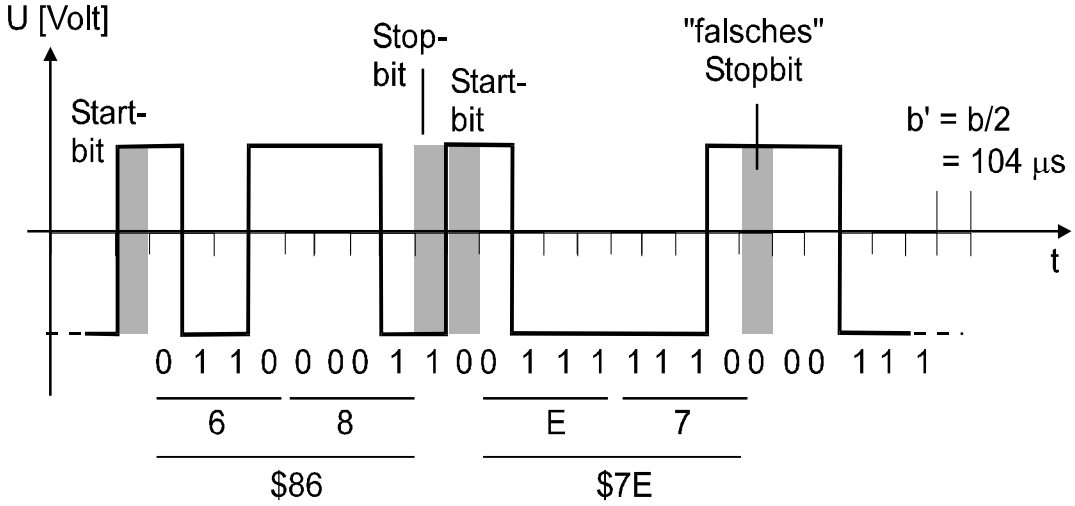
- kein Paritätsbit, PMC = 00 willkürlich
- Receiver Echo Mode disabled
- Sender aktiviert, Sender-Interrupt enabled, Empfänger-Interrupt disable
- DTR: Prozessor will Daten austauschen

IR = 0000 0111 (binär) = $07

b)



c)



1. Zeichen: Z1 = 1000 0110 (binär) = $ 86

   Statusregister: SR = 1001 1000 (binär) = $ 98

   kein Fehler, da Rahmen und Parität nicht verletzt, kein Overrun nach Voraussetzung

2. Zeichen: Z2 = 0111 1110 (binär) = $ 7E

   Statusregister: SR = 1001 1010 (binär) = $ 9A

   Fehler: Framing Error (FE=1), da nach letzten Datenbit ein Stopbit mit L-Pegel kommen müßte, Parität nicht verletzt, kein Overrun nach Voraussetzung