

Aufgabe 1 (10 Punkte):

a) Gegeben ist das folgende Karnaugh-Diagramm einer Schaltfunktion f in den vier Variablen X_1 bis X_4 .

	X_1				
X_2	0	0	0	1	X_4
	1	0	0	1	
	1	0	1	1	
	1	0	1	1	
	X_3				

Vervollständigen Sie die Wertetabelle! (3 P.)

Hinweis: Wie im Kurstext verwenden wir die verkürzende Schreibweise für Karnaugh-Diagramme, bei der gerade die Spalten bzw. Zeilen mit einer Variablen markiert werden, bei denen die entsprechende Variable den Wert 1 hat. Zum Beispiel gilt in den obersten beiden Zeilen $X_2 = 1$, und in den beiden äußeren Spalten $X_3 = 0$.

X_1	X_2	X_3	X_4	$f(X_1, \dots, X_4)$
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

Name:

Vorname:

Matr.-Nr.:

3

b) Geben Sie die Anzahl der Primimplikanten der Funktion f aus Teil a) an. (3 P.)

c) Nennen Sie einen der Primimplikanten der Funktion f aus Teil a). (1 P.)

d) Kreuzen Sie in der folgenden Primtermtabelle an, welche der Primimplikanten auch Kernimplikanten sind. (2 P.)

		Minterme/Trägerelemente						
		M1	M2	M3	M4	M5	M6	M7
<input type="checkbox"/>	P1	X	X			X	X	
<input type="checkbox"/>	P2	X	X				X	X
<input type="checkbox"/>	P3	X	X		X	X		
<input type="checkbox"/>	P4					X		X
<input type="checkbox"/>	P5	X	X	X		X		

e) Kreuzen Sie in der folgenden Primtermtabelle an, welcher der sogenannten Primimplikanten gar kein Primimplikant ist. (1 P.)

		Minterme/Trägerelemente						
		M1	M2	M3	M4	M5	M6	M7
<input type="checkbox"/>	P1	X	X				X	X
<input type="checkbox"/>	P2	X	X			X	X	
<input type="checkbox"/>	P3	X	X		X	X		
<input type="checkbox"/>	P4		X				X	
<input type="checkbox"/>	P5	X	X	X	X			

Aufgabe 2 (9 Punkte):

a) Beweisen Sie: Zu einer Zahl $z \in \{0, \dots, 2^n - 1\}$ kann es **höchstens eine** n -stellige Binärdarstellung $(a_{n-1}, \dots, a_0) \in \{0, 1\}^n$ geben mit $\langle a_{n-1}, \dots, a_0 \rangle_2 = z$. (4 P.)

Hinweis: Führen Sie den Beweis durch Widerspruch zu der Annahme, dass es eine zweite n -stellige Binärdarstellung $(b_{n-1}, \dots, b_0) \neq (a_{n-1}, \dots, a_0)$ gibt mit $\langle b_{n-1}, \dots, b_0 \rangle_2 = z$.

b) Rechnen Sie jeweils aus (5 P.):

$$[0101]_2 = \boxed{}$$

$$[1101]_2 = \boxed{}$$

$$\langle 1101 \rangle_2 = \boxed{}$$

$$\text{bin}_6(18) = \boxed{}$$

$$\text{twoc}(-1) = \text{mit 8 Stellen } \boxed{}$$

Name:

Vorname:

Matr.-Nr.:

5

Aufgabe 3 (10 Punkte):

a) Erstellen Sie die Wertetabelle für einen Halbaddierer. (2 P.)

<i>a</i>	<i>b</i>	<i>c</i>	<i>s</i>
0	0		
0	1		
1	0		
1	1		

b) Wir betrachten n -stellige Binärzahlen mit Betrag und Vorzeichen, sog. signed binary digits (sbd). Formal ist die sbd-Darstellung gegeben durch die Funktion

$$sbd : \{0, 1\}^{n+1} \rightarrow \{-2^n, \dots, +2^n\} \text{ mit } sbd(a_n, \dots, a_0) = \begin{cases} \langle a_{n-1}, \dots, a_0 \rangle & \text{wenn } a_n = 0 \\ -\langle a_{n-1}, \dots, a_0 \rangle & \text{wenn } a_n = 1 \end{cases}$$

d.h. die oberste Stelle gibt das Vorzeichen an, die restlichen n Stellen den Betrag. Für den Fall $n = 1$ gilt also die folgende Tabelle

$a_1 a_0$	$sbd(a_1, a_0)$
00	0
01	1
10	-0
11	-1

Für den Fall $n = 2$ gilt beispielsweise $sbd(101) = -1$ und $sbd(010) = +2$.

Zu entwerfen ist ein Addierer für zwei 1-stellige sbd-Zahlen ohne Eingangsübertrag, d.h. die Eingaben stellen Zahlen aus dem Bereich $\{-1, 0, +1\}$ dar, das Ergebnis stellt eine Zahl dar, die sich im Bereich $\{-2, \dots, +2\}$ befindet. Das Ergebnis soll als sbd mit $n = 2$ Stellen ausgegeben werden (ohne Ausgangsübertrag).

b1) Erstellen Sie die Wertetabelle, wobei beim Ergebnis Null dieses mit 0 0 0 kodiert sein soll (d.h. das Vorzeichen der Null spielt keine Rolle). (4 P.)

$a_1 a_0$	$b_1 b_0$	s_2	s_1	s_0
00	00			
00	01			
00	10			
00	11			
01	00			
01	01			
01	10			
01	11			
10	00			
10	01			
10	10			
10	11			
11	00			
11	01			
11	10			
11	11			

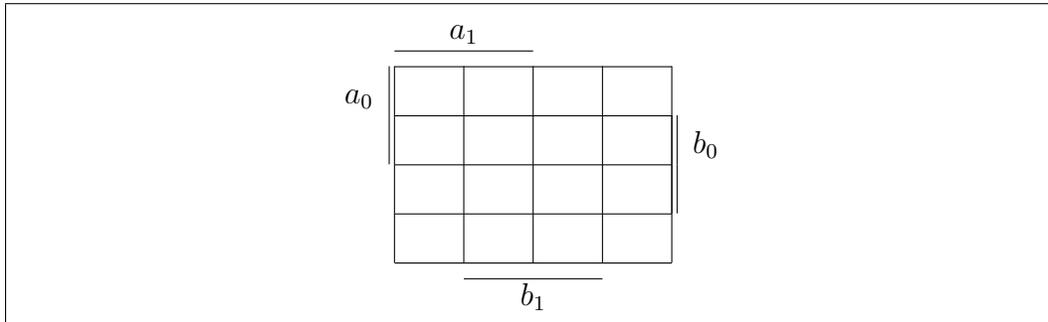
Name:

Vorname:

Matr.-Nr.:

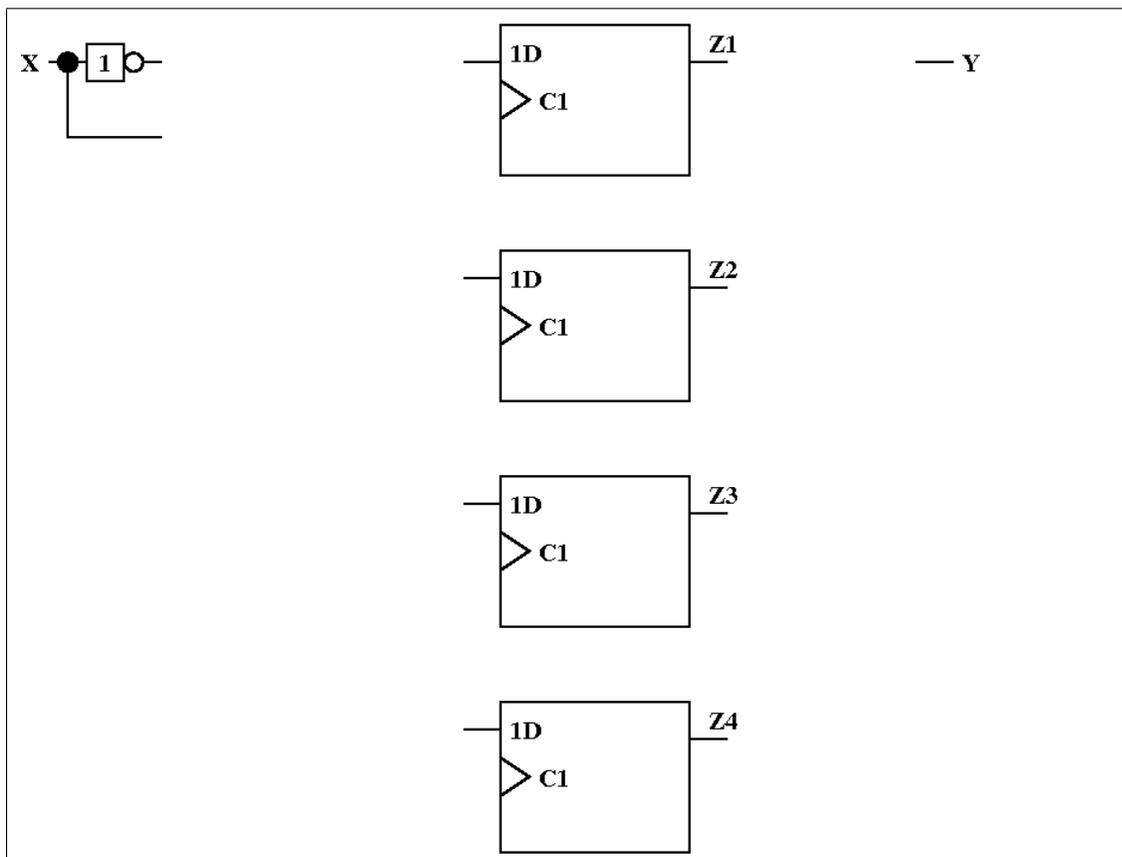
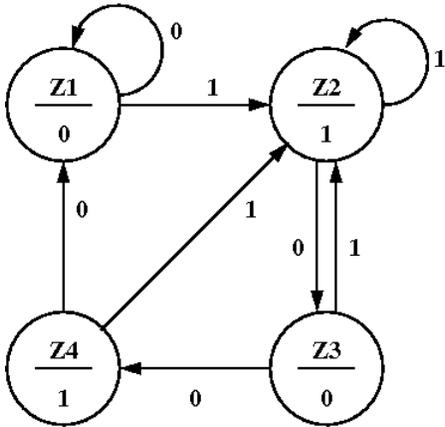
7

b2) Bestimmen Sie ein Minimalpolynom für s_2 unter Nutzung eines KV-Diagramms. (4 P.)



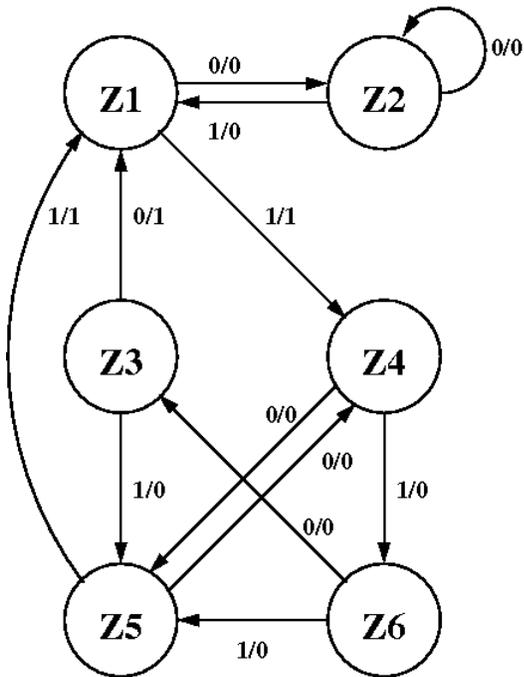
Aufgabe 4 (5 Punkte):

Gegeben sei ein Automat durch das folgende Zustandsdiagramm. Erstellen Sie ein Schaltwerk für diesen Automaten mit One-Hot-Codierung der Zustände unter Nutzung von D-Flipflops.



Aufgabe 5 (6 Punkte):

Gegeben ist ein Schaltwerk durch seinen Zustandsgraph.



a) Handelt es sich um einen Moore- oder einen Mealy-Automaten? (1 P.)

b) Erstellen Sie eine Liste aller Zustandspaare, die die Gleichung

$$(3.25) \forall x \in I : f(x, z_i) = f(x, z_j)$$

erfüllen, wobei f die Ausgangsfunktion des Automaten und I die Menge der möglichen Eingaben ist. (2 P.)

Name:

Vorname:

Matr.-Nr.:

10

c) Erstellen Sie für jedes Zustandspaar aus Teil b) die Liste der Folgezustandspaare, d.h. erstellen Sie die Tabelle Stufe 0 für eine Zustandsminimierung. (2 P.)

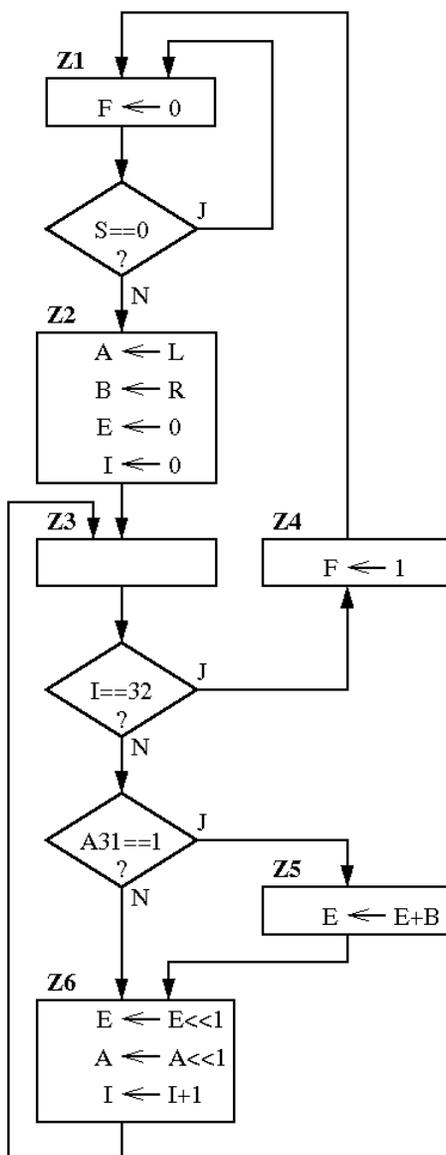
--

d) Kann es einen Zustandsgraphen mit 6 Zuständen, 1-Bit-Eingabe und 1-Bit-Ausgabe geben, bei dem kein Paar von Zuständen die obige Gleichung (3.25) erfüllt? Begründen Sie Ihre Antwort. (1 P.)

--

Aufgabe 6 (7 Punkte):

Gegeben sei das folgende ASM-Diagramm eines komplexen Schaltwerks, wobei die Eingabe aus zwei vorzeichenlosen 32-stelligen Binärzahlen L und R sowie einem Startsignal S besteht, es Register für Zahlen A , B , E , I und F gibt, und die Ausgabe aus dem Wert des Registers F besteht sowie aus dem Wert des Registers E , wenn das Ende von Zustand Z_4 erreicht wird, d.h. wenn der Inhalt des Registers F auf 1 gesetzt wird. Das Schaltwerk berechnet in E das Produkt aus L und R , wobei alle Register und die Arithmetik breit genug sein sollen, dass kein Überlauf stattfindet. Mit A_{31} ist das Bit 31 des Registers A mit Wertigkeit 2^{31} gemeint.



Name:

Vorname:

Matr.-Nr.:

12

a) Vervollständigen Sie die Tabelle des Steuerwerks, wobei lediglich die Steuersignale für die Register betrachtet werden. (3 P.)

Hinweis: Jedes Register verfügt über einen Steuereingang $S_{Registername}$, mit dem die Datenübernahme am Ende eines Zustands gesteuert wird. Eine 1 an diesem Steuereingang bewirkt die Übernahme des Wertes am Eingang des betreffenden Registers.

	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6
S_A						
S_B						
S_E						
S_I						
S_F						

b) Geben Sie weiterhin an, wieviele Statussignale das Steuerwerk empfängt, und wieviele Steuersignale zur Steuerung von Multiplexern im Operationswerk das Steuerwerk erzeugt. Hierbei soll auch bei Multiplexern mit mehr als zwei Wegen nur jeweils 1 Steuersignal gezählt werden. (1 P.)

Statussignale	
Steuersignale für Multiplexer	

c) Geben Sie an, wieviele Einheiten wie Addierer, Subtrahierer usw. jeweils im Operationswerk benötigt werden, wenn dieses gemäß der Vorgehensweise im Kurstext aufgebaut wird. (2 P.)

Vergleicher für Binärzahlen	
Addierer	
Subtrahierer	
Shifter um 1 Stelle	
Register	
Multiplizierer	

d) Bleibt bei einer Verlagerung der Zuweisung $I = I + 1$ von Zustand Z_6 nach Zustand Z_3 die Funktionalität des komplexen Schaltwerks erhalten? (1 P.)

JA

NEIN

Aufgabe 7: Maschinenbefehle (3 Punkte)

In Abschnitt 4.8.2 (Leitwerk) des Kurstextes wird die Abarbeitung verschiedener Arten von Maschinenbefehlen in mehreren Schritten beschrieben. Dabei wurde in manchen Fällen ein neuer Wert des Programmzählers PC dadurch bestimmt, dass er um die Länge des aktuellen Befehls erhöht wurde, in anderen Fällen wurde der PC mit einem neuen Wert überschrieben. Bitte kreuzen Sie in der folgenden Tabelle an, welche Art der Bestimmung bei welcher Art von Maschinenbefehl vorkommen kann. Dabei könnten bei einem Maschinenbefehl auch beide Arten der Bestimmung vorkommen.

Art des Maschinenbefehls	Bestimmung des PC durch	
	Erhöhen um akt. Befehl	Überschreiben
Verknüpfung von Operanden		
Datentransfer Speicher → Prozessor (Lesen)		
Datentransfer Prozessor → Speicher (Schreiben)		
Sprung (unbedingte Verzweigung)		
bedingte Verzweigung		

Name:

Vorname:

Matr.-Nr.:

14

Aufgabe 8: Befehlsebenen-Parallelität (8 Punkte)

Ordnen Sie die folgenden Konzepte zur Befehlsebenen-Parallelität durch Ankreuzen **einer** der drei vorgegebenen Kategorien zu! (je 1 P.)

Konzept	basiert auf		
	Hardware	Software	Hard- und Software
Ausführung außerhalb der Programmreihenfolge			
Spekulation			
Dynamisches Scheduling			
Registerumbenennung			
Rückordnungspuffer			
Superskalarer Prozessor			
VLIW-Prozessor			
EPIC-Prozessor			

Aufgabe 9: Gleitkommadarstellung (9 Punkte)

Gegeben sei die Dezimalzahl $Z_{10} = -101,75$.

- a) Stellen Sie die Zahl Z_{10} als Dualzahl mit Vorzeichen und Betrag dar. (1 P.)

$$Z_2 = \dots\dots\dots, \dots\dots\dots_2$$

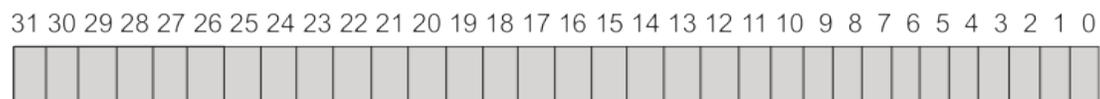
- b) Stellen Sie nun die Zahl Z_{10} als gebrochene, normalisierte Zahl Z_{32} im 32-bit-Format des IEEE-754-Standards dar und tragen Sie dazu die entsprechenden Werte für Vorzeichen, verschobenen Exponenten (auch Charakteristik oder biased exponent genannt) und Mantisse in das folgende Schema ein. (2 P.)

$$Z_{32} = (-1)^{\dots\dots\dots} \cdot 2^{(\dots\dots\dots)_{10}} \cdot (\dots\dots\dots, \dots\dots\dots 0 \dots 0)_2$$

- c) Geben Sie den verschobenen Exponenten als 8-Bit-Dualzahl an! (2 P.)

$$e = \dots\dots\dots_2$$

- d) Tragen Sie die Zahl Z_{32} in den folgenden Bitrahmen ein und bezeichnen Sie die unterscheidbaren Bitfelder. (3 P.)



- e) Geben Sie die Zahl Z_{32} als Hexadezimalzahl Z_{16} an. (1 P.)

$$Z_{16} = \dots\dots\dots$$

Hinweis:

Die Indizes \dots_2 , \dots_{10} , \dots_{16} und \dots_{32} kennzeichnen jeweils Zahlen im Binär-, Dezimal- sowie Hexadezimal-System bzw. im 32-bit-IEEE-Format.

Aufgabe 10: Datenkonflikte (8 Punkte)

- a) Welche Datenkonflikte müssen bei einem skalaren RISC-Prozessor beachtet werden? (1 P.)

- b) Welche Datenkonflikte müssen bei einem superskalaren RISC-Prozessor beachtet werden? (2 P.)

- c) Nennen Sie mindestens zwei Methoden zur Behebung von echten Datenabhängigkeiten beim Befehlspipelining! (2 P.)

- d) Wie viele Umbenennungsregister sind nötig, um alle unechten Datenkonflikte (Gegen- und Ausgabeabhängigkeiten) im folgenden Assembler-Programm aufzulösen? (3 P.)

1: LW R1, (R2)	; R1 ← Mem[R2]
2: ADD R3, R4, R1	; R3 ← R4 + R1
3: SUB R4, R5, R6	; R4 ← R5 - R6
4: MUL R7, R4, R8	; R7 ← R4 * R8
5: SUB R8, R9, R10	; R8 ← R9 - R10
6: ADD R11, R8, R12	; R11 ← R8 + R12
7: DIV R12, R13, R14	; R12 ← R13 / R14
8: SW (R15), R12	; Mem[R15] ← R12

Aufgabe 11: Sprungvorhersage (8 Punkte)

Eine Schleife in einem Programm enthält fünf bedingte Sprünge. In der nachfolgenden Liste sind die Sprungergebnisse für die einzelnen Sprünge aufgelistet.

Hierbei gilt (T: branch taken; N: branch not taken).

Sprung 1: T-T-T
Sprung 2: N-N-N-N
Sprung 3: T-T-N-T-N-T-N
Sprung 4: T-T-T-N-T
Sprung 5: T-N-N-N-T-T-N-N-T-T

Es sei angenommen, dass die Sprungverläufe bei jedem Durchlauf diesselben sind. Analysieren Sie die prozentuale Vorhersagegenauigkeit für die folgenden Prädiktoren:

- a) Statische Sprungvorhersage mit der Annahme `branch taken` (1 P.)
- b) Statische Sprungvorhersage mit der Annahme `branch not taken` (1 P.)
- c) Ein-Bit-Prädiktor, initialisiert mit `branch taken` (2 P.)
- d) Zwei-Bit-Prädiktor mit Sättigungszähler, initialisiert mit `weakly taken` (4 P.)

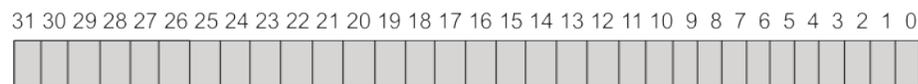
Wichtig ist die Herleitung der Ergebnisse. Die exakten numerischen Werte für die Vorhersagegenauigkeit sind nicht zum Erreichen der max. Punktzahl erforderlich!

Aufgabe 12: Cache-Organisation (12 Punkte)

Ein Mikroprozessor besitze einen 32-bit-Adressbus und einen 16-bit-Datenbus. Er verfüge über einen *Direct Mapped Cache*, dessen Datenspeicher eine Kapazität von 64 kbyte hat und Einträge (*Cache Lines*) der Länge 16 byte enthält. Unter der Organisation eines Speichers versteht man die Angabe der Anzahl m der Speicherzellen und ihrer Länge l (in Bits oder Bytes), meist in der Form $m \times l$.

- a) Geben Sie die Anzahl der Cache-Einträge sowie die Organisation, d.h. die Anzahl der Speicherzellen und die Länge (in Bits oder Bytes) jeder Speicherzelle, des Datenspeichers an. (Rechnung erforderlich!) (2 P.)

- b) Tragen Sie in das folgende Bild die unterscheidbaren Bitfelder einer Adresse für die Auswahl eines Bytes, eines Cache-Eintrags und die im Cache gespeicherte Teiladresse ein und benennen Sie diese Bitfelder. (2 P.)



Bitfelder:.....

- c) Geben Sie für die gefundene Adreßaufteilung die Kapazität und die Organisation des Adressspeichers (*Tag-RAM*) im Cache an. (Rechnung erforderlich!) (2 P.)

- d) Das Datenregister DR enthalte den Wert $DR = \$ABCD$, das Adressregister AR den Wert $AR = \$FACD\ 3A6A$. Mit diesen Registern werde der Schreibbefehl $ST(AR), DR$ „Schreibe DR in die Speicherzelle, deren Adresse in AR steht.“ ausgeführt. Geben Sie an, welcher Eintrag im Cache verändert wird, wenn dieser
- nach dem Rückschreibverfahren verwaltet wird und ein *Write Hit* vorliegt,
 Index: $\$. \dots = \dots_{10}$
 („ $\$$ “, „ $_{10}$ “ kennzeichnen einen Hexadezimal- bzw- Dezimalwert!)
 Adressen der veränderten Bytes im Eintrag: $\$. \dots$ und $\$. \dots$
 - nach dem Durchschreibverfahren mit *Write Around* verwaltet wird, bei dem nach einem *Write Miss* das Datum nur im Hauptspeicher abgelegt wird. Im Cache liege unter dem Index $\$3A6$ der Tag $\$FA00$. (2 P.)
- e) Der Cache sei nun als 4-fach satzassoziativer Cache (*n-Way Set Associative Cache*) organisiert, besitze aber die gleiche Gesamtkapazität von 64 kbyte für die Datenspeicher, aber mit einer Länge der Einträge von 32 byte. Jeder Eintrag im Tag-RAM der Teil-Caches enthalte zwei Verwaltungsbits V (*Valid*) und D (*Dirty*).
 Bestimmen Sie wiederum die Anzahl der Einträge pro Teil-Cache, die Organisation der Tag-RAMs und ihre Gesamtkapazität und die unterscheidbaren Bits einer Speicheradresse. (Rechnung erforderlich !) (4 P.)

Kapazität der Teil-Caches:

Einträge pro Teil-Cache:

Länge der Tag-RAM-Einträge:

Organisation der TAG-RAMs:

Gesamtkapazität:

Bitfelder:

Aufgabe 13: Parallelverarbeitung (5 Punkte)

Eine sequentielle Anwendung benötige t_{seq} Sekunden. Davon lassen sich 80% parallelisieren, d.h. sie können gleichmäßig auf p Prozessoren verteilt werden. Die Rechenleistung der p Prozessoren sei jeweils gleich, so dass alle parallelen Anteile in der gleichen Zeit berechnet werden.

- a) Es sei zunächst vorausgesetzt, dass die Parallelisierung keinen zusätzlichen Aufwand (wie z.B. Kommunikationskosten) verursacht. Bestimmen Sie Speedup und Effizienz für einen Parallelrechner mit 2,4,8 und 16 Prozessoren! (3 P.)

- b) Nun soll angenommen werden, dass für jeden parallelen Prozessor ein zusätzlicher sequentieller Anteil 1% anzusetzen ist. Berechnen Sie Speedup und Effizienz für ein 16-Prozessor-System! (2 P.)

Wichtig ist die Herleitung der Ergebnisse. Die exakten numerischen Werte sind nicht zum Erreichen der max. Punktzahl erforderlich!