Hochschule Konstanz Fakultät Informatik

WS 10/11

Klausur: Rechnerarchitektur Datum: 15.02.11

Prüfer: Prof. Dr. J. Neuschwander Zeitdauer: 90 Min.

Name:
Matr.-Nr.:

Hinweise:

Beschriften Sie jedes weitere abgegebene Lösungsblatt deutlich lesbar mit Ihrem Namen und Ihrer Matrikelnummer.

Es sind für diese Klausur als Hilfsmittel nur Taschenrechner, ein DIN-A4 Blatt als Formelsammlung sowie der Befehlssatzes des Prozessors M68000 zugelassen.

Bei jeder angegebenen Lösung muss der Lösungsweg jeweils nachvollziehbar sein.

Maximal erreichbare Punktzahl der Klausur:	37 Punkte
Mindestpunktzahl zum Bestehen der Klausur:	17 Punkte
Aufgabe 1:	7 Punkte
Aufgabe 2:	6 Punkte
Aufgabe 3:	6 Punkte
Aufgabe 4:	7 Punkte
Aufgabe 5:	5 Punkte
Aufgabe 6:	6 Punkte

1. Aufgabe: (7 P)

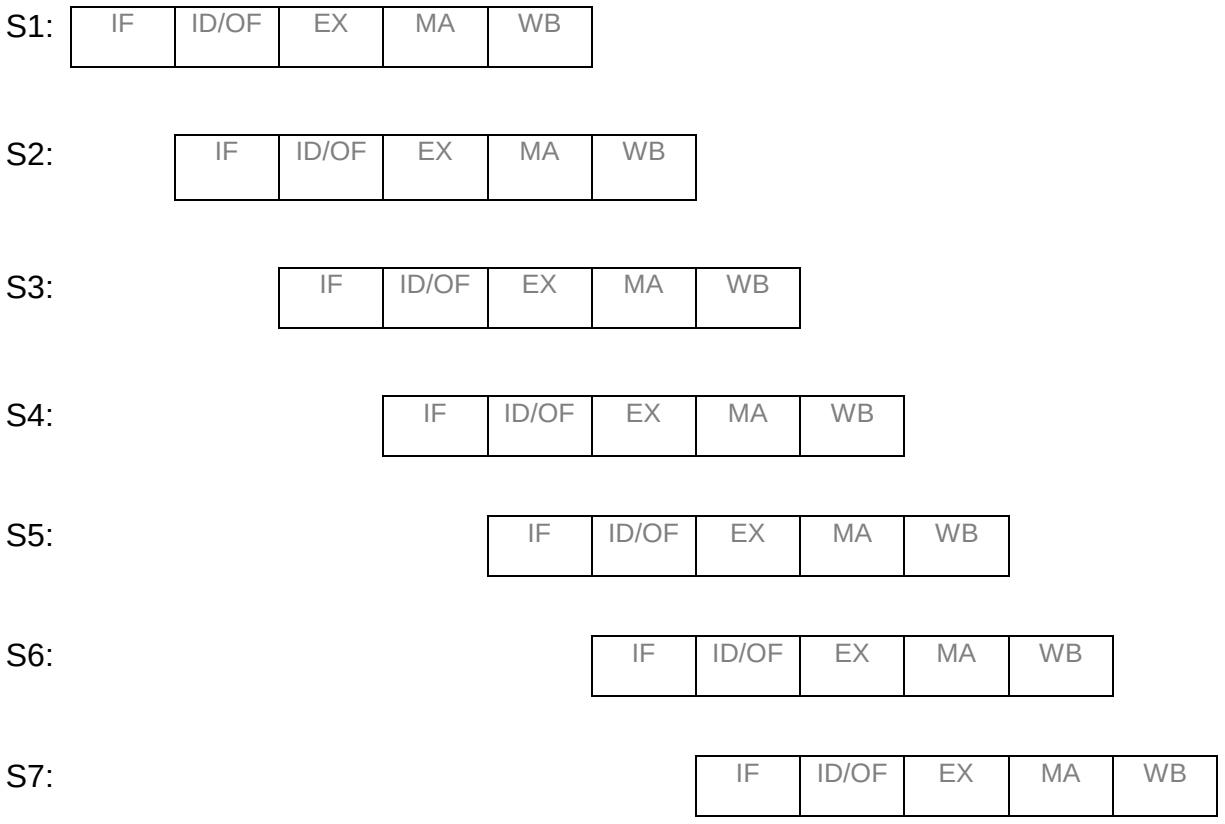
Gegeben sei folgende Programmsequenz für einen RISC-Prozessor:

```
S1:   LOAD    R1, (A1)      ; R1:= <(A1)>
S2:   LOAD    R2, (A2)      ; R2:= <(A2)>
S3:   LOAD    R3, (A3)      ; R3:= <(A3)>
S4:   ADD     R4, R3, R2    ; R4:= R3 + R2
S5:   SUB     R3, R2, R1    ; R3:= R2 – R1
S6:   ADD     R2, R3, R4    ; R2:= R3 + R4
S7:   STORE   (A3), R4      ; <(A3)>:= R4
```

Die einzelnen Befehle werden in der fünfstufigen DLX-Pipeline verarbeitet:

Befehl holen:	IF	(Latenzzeit: 5 ns)
Befehl decodieren/ Operanden holen	ID/OF	(Latenzzeit: 6 ns)
Operation ausführen	EX	(Latenzzeit: 5 ns)
Speicherzugriff durchführen	MA	(Latenzzeit: 6 ns)
Ergebnis speichern	WB	(Latenzzeit: 6 ns)

a) Markieren Sie mit Pfeilen nur die Abhängigkeiten der Pipelinestufen, die im gegebenen Programm zu Pipeline-Konflikten führen.



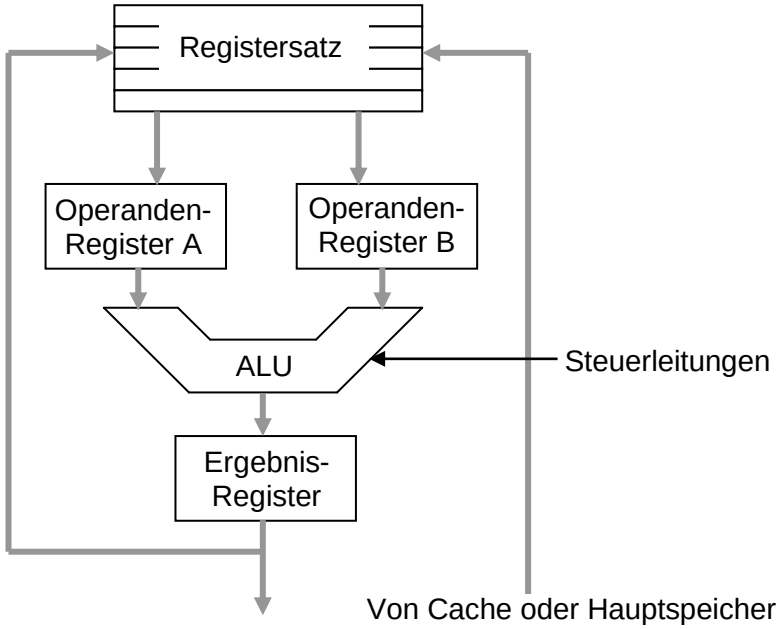
b) Die auftretenden Pipeline-Konflikte sollen vom Compiler behandelt werden. Ergänzen Sie obiges Programmstück so, dass die auftretenden Pipeline-Konflikte beseitigt werden und das Ergebnis dem der normalen, sequenziellen Bearbeitung entspricht.

c) Wie viele Takte werden benötigt, um dieses Programm aus b) abzuarbeiten?

d) Die Latenzzeiten der einzelnen Pipelinestufen sind am Anfang der Aufgabe angegeben. Die Latenzzeit der Pipelineregister beträgt 3 ns. Mit welcher maximalen Taktfrequenz kann die Pipeline betrieben werden?

f) Wie viele Befehle eines Programms sind (mindestens) erforderlich, wenn bei der verwendeten Pipeline ein **Speed-up** von **4,99** erreicht werden soll?

g) Zeichnen Sie die Modifikation des Rechenwerks, die notwendig ist, um die unter a) festgestellten Pipeline-Konflikte hardwaremäßig zu lösen in das folgende Bild ein.



i) Geben Sie nun alle Datenabhängigkeiten im Programmstück an.

```
S1:   LOAD    R1, (A1)        ; R1:= <(A1)>
S2:   LOAD    R2, (A2)        ; R2:= <(A2)>
S3:   LOAD    R3, (A3)        ; R3:= <(A3)>
S4:   ADD     R4, R3, R2      ; R4:= R3 + R2
S5:   SUB     R3, R2, R1      ; R3:= R2 – R1
S6:   ADD     R2, R3, R4      ; R2:= R3 + R4
S7:   STORE   (A3), R4        ; <(A3)>:= R4
```

Stellen Sie die jeweiligen Abhängigkeiten durch einen Pfeil dar und benennen sie diese Abhängigkeit.

2. Aufgabe: (6 P)

Skizzieren Sie für die unten angegebenen Architekturklassen jeweils eine entsprechende Codierung für die Ausführung der beiden Befehle

C = A * B; und **D = C – A.**

Die Variablen A, B, C und D befinden sich im Arbeitsspeicher und alle Ergebnisse sollen ebenfalls wieder im Arbeitsspeicher abgelegt werden. Verwenden Sie zur Bezeichnung von Registern R1, R2,…

Als Befehle stehen Ihnen zur Verfügung: **{ load, store, add, sub, mul, push, pull }**

Programm: **C = A * B; D = C – A;**
In den 4 Befehlsformaten codiert

Register-Register	Register-Speicher	Akkumulator	Stack-Maschine

3. Aufgabe: (6 P)

(Zur Ermittlung der Punktzahl in dieser Teilaufgabe werden von den richtig angekreuzten Aussagen die falsch angekreuzten Aussagen abgezogen; Nicht angekreuzte Aussagen zählen nicht und gehen somit nicht in die Bewertung ein.)

	richtig	falsch
Ein sogenannter „look-through-cache“ wird bevorzugt bei Multi-Prozessorsystemen eingesetzt.		
Bei einem semi-synchronen Systembus kann die Anzahl der wait states des Prozessors variieren, abhängig vom adressierten Baustein.		
Beim Rückschreibverfahren (Copy Back) wird beim Schreiben des Prozessors bei einem Cache-Hit sowohl der Cache als auch der Hauptspeicher aktualisiert.		
Unter Data-Alignment versteht man die Verschachtelung von Hauptspeicherzugriffen auf verschiedene Speicherbänke.		
Bei SDRAM-Speicherbausteinen kann die Zykluszeit durch Interleaving verbessert werden.		
Beim sogenannten „Bus Snooping“ wertet der Prozessor zur Konsistenzsicherung die Speicheradressen anderer Busmaster beim Zugriff auf den Hauptspeicher aus.		
Zur Implementierung einer dynamischen Sprungvorhersage kann ein voll-assoziativer Cache verwendet werden.		
Die mittlere Zugriffszeit in einem Speichersystem, das aus einem Verbund von Cachespeicher und Arbeitsspeicher besteht, steigt mit der Abnahme der Hitrate.		
Bei der virtuellen Adressierung ist die Anzahl der Seiten und der Rahmen immer gleich groß.		
Der *TAS*-Befehl wird auf der Hardwareebene der Prozessor-realisierung als ein sogenannter Read-Modify-Write-Zyklus ausgeführt.		
Bei VLIW-Prozessoren übernimmt der Compiler die dynamische Befehlszuteilung auf die Ausführungseinheiten.		
Bei einem Mehrwege Satz-assoziativen Cachespeicher mit 2^n Zeilen sind 2^n Vergleicher erforderlich.		

4. Aufgabe: (7 P)

Gegeben seien ein Direct-Mapped-Cache (DM), ein 2-Way-Set-Associative Cache (A4) und ein Full-Associative Cache (VA). Die drei Cache-Speicher haben jeweils eine Speicherkapazität von 64 Byte und werden in Blöcken von je 16 Byte geladen. Die Hauptspeicheradresse umfasst 32 Bit. Falls notwendig, wird die „Least Recently Used“-Ersetzungsstrategie verwendet.

a) Welche Bits der 32-Bit Adresse bilden Offset, Index und Tag? Skizzieren Sie hierzu die Unterteilung der Hauptspeicheradresse für die drei Cache-Speicher.

DM:

A2:

VA:

b) Der Zustand eines Cache-Blocks wird durch das Valid-Bit, das Dirty-Bit und 2 Bit für die Codierung der Alterungsinformation gekennzeichnet. Wie viel Speicherplatz und wie viele Vergleicher mit welcher Bitanzahl werden für die Realisierung des Tag-Speichers des A2-Cache benötigt?

	Speicherplatz	Anzahl der Vergleicher	Bitzahl des Vergleichers
A2			

c) Betrachten Sie die Folge der Lesezugriffe auf die folgenden in hexadezimaler Schreibweise angegebenen Hauptspeicheradressen:

$29, $34, $9A, $D0, $15, $25, $3A, $99, $C6

Nehmen Sie an, die Cache-Speicher seien zu Beginn leer. Kennzeichnen Sie in der folgenden Tabelle für jeden Cache-Speicher, ob es sich beim Lesezugriff auf die jeweiligen Adressen um einen Cache-Miss (Kennzeichnung „–„) oder um einen Cache-Hit (Kennzeichnung „x „) handelt.

Adresse	$29	$34	$9A	$D0	$15	$25	$3A	$99	$C6
DM									
A4									
VA									

Cache-Miss (Kennzeichnung „–„) Cache-Hit (Kennzeichnung „**x** „)

5. Aufgabe: (5 P)

5.1: Der in einem 32-Bit-Rechnersystem eingesetzte TLB (translation look aside buffer) dient der Beschleunigung der Adressumsetzung. Der Adressspeicher des betrachteten TLB ist 20 Bit breit und teilt sich auf in eine Segmentnummer (7 Bit) und eine Seitennummer (13 Bit). Der Datenspeicher, der die Framenummer enthält, ist 12 Bit breit.

a) Wie groß kann ein Segment des virtuellen Speichers maximal sein?

b) Wieviele Seiten passen maximal in ein Segment?

c) Mit wieviel Byte muss der Hauptspeicher minimal ausgebaut sein, damit alle Frames darin Platz finden?

d) Wie groß ist der Verschnitt eines Segments (in Byte), wenn die Daten in diesem Segment genau 1024 Seiten belegen?

e) Wieviele Frames belegt ein Programm im Arbeitsspeicher, wenn dessen Größe 12 MB beträgt?

5.2: Gegeben sei nun eine Speicherverwaltungseinheit (MMU). Der virtuelle Speicher in diesem System ist in 8 Seiten mit je 1 kByte unterteilt. Der physische Speicher hat eine Kapazität von 4 kByte. Der aktuelle Ausschnitt der Seitentabelle ist in folgender Tabelle angegeben.

Virtuelle Seitennummer	Physische Seitennummer
0	-
1	-
2	3
3	2
4	-
5	0
6	1
7	-

f) Skizzieren Sie die Unterteilung der 32-Bit breiten virtuellen Adresse.

g) Ermitteln Sie die physischen Adressen (Dezimaldarstellung) zu den folgenden virtuellen Adressen (Dezimaldarstellung) :

2200, 6630, 1021

6. Aufgabe: (6 P)

In der folgenden Aufgabe soll für alle Teilaufgaben jeweils die in Tabelle 8.1 dargestellte Ausgangssituation herrschen. Das Zeichen $ kennzeichnet die hexadezimale Darstellung von Zahlen. Beachten Sie jeweils das Datenformat.

Prozessor-Register	
Datenregister	**Inhalt**
D0	$ 00 00 11 11
D1	$ 22 22 33 33
D2	$ 44 44 55 55
D3	$ 66 66 77 77
D4	$ 88 88 99 99
D5	$ AA AA BB BB
D6	$ CC CC DD DD
D7	$ EE EE FF FF

Adressregister	Inhalt
A0	$ 00 01 00 02
A1	$ 00 01 00 0A
A2	$ 00 01 00 08
A3	$ 00 01 00 00
A4	$ 00 01 00 00
A5	$ 00 01 00 00
A6	$ 00 01 00 00
A7	$ 00 01 00 00

Arbeitsspeicher (16 Bit)	
Adresse	**Inhalt**
$ 10000	$ 00 01
$ 10002	$ 02 03
$ 10004	$ 04 05
$ 10006	$ 06 07
$ 10008	$ 08 09
$ 1000A	$ 0A 0B
$ 1000C	$ 0C 0D
$ 1000E	$ 0E 0F
$ 10010	$ 10 11
$ 10012	$ 12 13
.	
.	
.	
$ FFFFA	$ 01 02
$ FFFFC	$ 03 04
$ FFFFE	$ 05 06

Tabelle 8.1 : Ausgangssituation Register und Hauptspeicher

a) Tragen Sie den Inhalt der Datenregister D0 und D2 (je 32 Bit) in Tabelle 8.2 ein, nachdem mit der Ausgangssituation entsprechend Tabelle 8.1 alle folgenden Befehle nacheinander ausgeführt wurden :

```
K1      DC.W    $FFFFA
K2      DS.L    $1
START   MOVE.L  #K1, D0
        MOVE.W  K1, D1
        ADD.W   D1, D2
        MOVE.W  D2, K2
```

Prozessor-Register	
Datenregister	**Inhalt**
D0	$
D2	$

Tabelle 8.2 : Lösung zu Teilaufgabe a)

b) Tragen Sie den Inhalt der Datenregister D5, D6 und D7 sowie der Adressregister A0, A1 und A2 (je 32 Bit) in Tabelle 8.3 ein, nachdem mit der Ausgangssituation entsprechend Tabelle 8.1 die folgenden Befehle nacheinander ausgeführt wurden:

```
ANF   EQU      $1FF00
      MOVE.B   -(A2), D7
      MOVE.L   (A1)+, D6
      MOVE.W   (A1)+, D5
      MOVE.W   #ANF, A1
```

D5	$
D6	$
D7	$

Adressregister	Inhalt
A0	$
A1	$
A2	$

Tabelle 8.3 : Lösung zu Teilaufgabe b)

c) Tragen Sie den Inhalt der Datenregister D3 und D4 (je 32 Bit) in Tabelle 8.4 ein, nachdem mit der Ausgangssituation entsprechend Tabelle 8.1 die folgenden Befehle nacheinander ausgeführt wurden :

```
SWAP    D3
SUB.L  -(A1),D4
```

Prozessor-Register	
Datenregister	**Inhalt**
D3	$
D4	$

Tabelle 8.4 : Lösung zu Teilaufgabe c)