

Klausur zur „Technischen Informatik I“, WS 2004/05

3. Februar 2005

Hinweise:

- **Bearbeitungszeit:** 2 Stunden
Gesamtpunktzahl: 100 Punkte
 Zum Bestehen der Klausur sind **40 Punkte** erforderlich.
- Hilfsmittel sind nicht erlaubt.
- Jede Aufgabe ist auf einem eigenen Blatt zu bearbeiten.
- Alle ausgehändigten Blätter sind zurückzugeben und, soweit beschrieben, mit Namen zu versehen. Auch der Klausurtext ist abzugeben.

Viel Erfolg!

Name:

Matr.-Nr.: **Studienfach:**

Aufgabe	max. Punktzahl	erreichte Punktzahl	korrigiert von
1	8		
2	13		
3	10		
4	10		
5	12		
6	10		
7	10		
8	5		
9	12		
10	10		
Summe	100		

Note:

1. Amdahlsches Gesetz

8 Punkte

Sie haben den Auftrag einen Computer zu erweitern. Es stehen zwei Optimierungen zur Debatte: Multiplikationen können um den Faktor 4 und Speicherzugriffe um den Faktor 2 beschleunigt werden.

Auf dem Computer wird meistens ein Programm ausgeführt, das eine Laufzeit von 100 Sekunden hat und von dieser Zeit 20 % für Multiplikationen, 50% für Speicherzugriffe und 30% für sonstige Instruktionen benötigt.

- (a) Wie lautet das Amdahlsche Gesetz? / 2
- (b) Was folgt aus dem Amdahlschen Gesetz für die zur Debatte stehenden Optimierungen? / 2
- (c) Welche Optimierung führen Sie durch, wenn Ihr Budget nur eine der beiden Erweiterungen erlaubt? Begründen Sie Ihre Antwort. / 2
- (d) Welche Beschleunigung kann bei der Integration beider Optimierungen erzielt werden? / 2

2. Boolesche Algebra

13 Punkte

- (a) Zeigen Sie, dass in *jeder* Booleschen Algebra gilt: $xy = x \Leftrightarrow xy' = 0$. Benennen Sie jedes verwendete Gesetz. / 4
- (b) Gegeben sei folgende Boolesche Funktion:

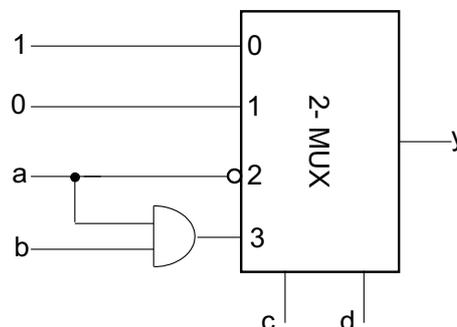
$$f(a, b, c) = ((a + b) a' + c)' (b' + c)$$

- i. Entwerfen Sie das Schaltnetz zu dieser Funktion mit den elementaren Gattern UND, ODER, NICHT. (Dabei haben UND und ODER-Gatter 2 Eingänge, NICHT hat einen Eingang.) / 2
- ii. Minimieren Sie diese Funktion mit Hilfe der Gesetze der Booleschen Algebra. Diese brauchen nicht benannt zu werden. (Hinweis: Es kann so minimiert werden, dass die Funktion von einer der Variablen unabhängig wird.) / 3
- (c) Bestimmen Sie zu der Schaltfunktion f mit $f(x, y, z, u) = (x + yz)(x'y + u)'$ die disjunktive und die konjunktive Normalform. / 4

3. Multiplexer

10 Punkte

- (a) Geben Sie die Boolesche Funktion $y = f(a, b, c, d)$ zu dem nebenstehenden Schaltnetz an. / 2



- (b) Implementieren Sie die folgende Boolesche Funktion durch einen Multiplexer mit 8 Dateneingängen. / 4

$$g(a, b, c, d) = abc' + acd + a'bcd' + a'b'd' + a'b'c'd$$

- (c) Kann die Funktion g auch mit einem 2-MUX ohne Verwendung zusätzlicher binärer Gatter realisiert werden? Negationen sind erlaubt. Begründen Sie Ihre Antwort! / 4

4. Bestimmung von Minimalpolynomen

10 Punkte

Gegeben sei nebenstehendes Karnaugh-Diagramm für eine Funktion f . Die mit 'D' gekennzeichneten Einträge seien „Don't care“-Einträge.

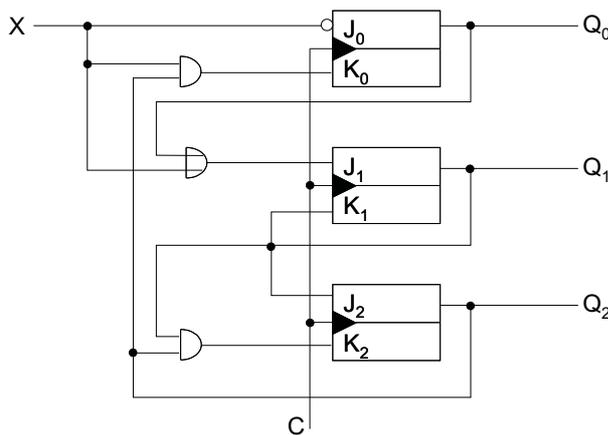
$x_3x_4 \backslash x_1x_2$	00	01	11	10
00	D	0	1	1
01	0	D	1	0
11	0	D	1	1
10	1	0	1	D

- (a) Geben Sie ein Minimalpolynom für f an. Wählen sie dafür passende 'D'-Einträge. / 4
- (b) Geben Sie alle Primimplikanten für die Komplementärfunktion $\bar{f}(x_1, x_2, x_3, x_4) = f(x_1, x_2, x_3, x_4)'$ an. Die 'D'-Einträge für f seien dabei alle gleich Null. / 4
- (c) Welche Bedingungen muss die Anordnung der Felder eines Karnaugh-Diagramms erfüllen? / 1
- (d) Welche Regel wird bei der Überdeckung implizit ausgenutzt? / 1

5. Schaltwerkanalyse

12 Punkte

Gegeben sei folgendes Schaltwerk:



Die Funktionstabelle eines JK-Flipflops lautet:

J	K	Q_{n+1}	Funktion
0	0	Q_n	Speichern
0	1	0	Rücksetzen
1	0	1	Setzen
1	1	Q'_n	Invertieren

- (a) Bestimmen Sie die Funktionsterme für die Eingänge J_0, K_0, J_1, K_1, J_2 und K_2 der drei JK-Flipflops in Abhängigkeit von Q_0, Q_1, Q_2 und X . / 3
- (b) Erstellen Sie die Zustandsfolgetabelle nach folgendem Muster ausgehend vom Startzustand 000 für $X=0$ und $X=1$. Der Wert für X sei dabei konstant gleich Null oder konstant gleich Eins. Es müssen nur die Zustände berücksichtigt werden, die das Schaltwerk unter diesen Voraussetzungen annimmt. Die Zustandsübergangstabelle muss also nicht vollständig sein. / 5

X	Q_2^n	Q_1^n	Q_0^n	J_2	K_2	J_1	K_1	J_0	K_0	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
0	0	0	0	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	0	0	0	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

- (c) Zeichnen Sie den Zustandsübergangsgraphen. / 2
- (d) Beschreiben Sie verbal die Funktion des Schaltwerks. / 2

6. Zahlendarstellung

10 Punkte

(a) Zweierkomplementdarstellung

/ 4

- i. Stellen Sie die Zahlen $43_{(10)}$ und $-22_{(10)}$ als 8-Bit-Zweierkomplementzahlen dar.
- ii. Addieren Sie die beiden Zahlen $(43 + (-22))$ im Zweierkomplement.

(b) Gleitkommadarstellung

/ 6

Nach dem IEEE 754 Standard ist die Darstellung einer Gleitkommazahl bei einfacher Genauigkeit wie folgt: $(-1)^V \times (1 + \text{Mantisse}) \times 2^{\text{Exponent}-127}$

- i. Die Zahl 127 wird als Bias bezeichnet. Warum wird die Biased Notation verwendet?
- ii. Wandeln Sie folgende Zahl, die in der Gleitkommadarstellung (IEEE 754) gegeben ist, in Dezimaldarstellung um:

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	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
V	Exponent								Mantisse																						

iii. Addieren sie die obige Zahl mit der folgenden Gleitkommazahl:

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	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
V	Exponent								Mantisse																						

7. Division von ganzen Zahlen

10 Punkte

- (a) Zeigen Sie, dass bei der ganzzahligen Division einer $2n$ -Bit-Binärzahl durch eine n -Bit-Zahl ein Überlauf stattfinden kann, wenn für den Quotienten n Bits zur Verfügung stehen. / 1
- (b) Führen Sie die iterative Division in einer Konfiguration mit einem 4-Bit-Addierer / Subtrahierer für die folgenden vorzeichenlosen Binärzahlen durch. Der Dividend sei 00101110, der Divisor 0110. Erläutern Sie Ihre Vorgehensweise und das Ergebnis. / 6
- (c) Zeigen Sie: Unter der Annahme, dass kein Überlauf stattfindet, wird in der ersten Iteration des in der Vorlesung vorgestellten iterativen Verfahrens zur Division vorzeichenloser Binärzahlen immer nur das Quotientenbit 0 ermittelt. / 3

8. Ausführungsmodelle

5 Punkte

Übersetzen Sie die Anweisungsfolge

$$a := b+c; b := a+c; d := a-b;$$

wobei a, b und c Speicheradressen bezeichnen, in möglichst kurze Instruktionsfolgen für

- (a) eine Stackmaschine (Kellermaschine) mit den Befehlen / 2
 PUSH adr, POP adr, ADD und SUB
- (b) eine load/store-Architektur mit drei Registern r1, r2, r3 und den Befehlen / 3
 LOAD reg,adr, STORE reg, adr und OP reg, reg,
 wobei OP ADD oder SUB sein kann.

9. Assemblerprogrammierung

12 Punkte

- (a) Die folgende Assemblerprozedur `copy` soll Worte von der Adresse in Register `$a0` zu der Adresse in Register `$a1` kopieren und dabei die Anzahl der kopierten Worte in Register `$v0` berechnen. Der Kopiervorgang soll beendet werden, wenn das Wort 0 gefunden wird. Dieses Abschlusswort soll kopiert, aber nicht gezählt werden. Die Inhalte der Register `$a0` und `$a1` sollen nach Beendigung der Prozedur unverändert sein.

/ 8

```

copy:   addi $sp, $sp, -8
        sw   $a0, 4($sp)
        addi $v0, $zero, 0
loop:   lw   $t0, 0($a0)
        sw   $t0, 0($a1)
        addi $a0, $a0, 4
        addi $a1, $a1, 4
        beq  $v1, $zero, loop
        lw   $a0, 4($sp)
        lw   $a1, 0($sp)
        jr   $ra

```

Erläutern und korrigieren Sie die in der Prozedur enthaltenen Fehler.

Korrigieren Sie das Programm und versehen Sie es mit Kommentaren.

- (b) Schreiben Sie, *ohne* den Multiplikationsbefehl zu verwenden, eine MIPS-Assemblerprozedur `mul7` mit weniger als 5 Instruktionen, die im Ergebnisregister `$v0` das Siebenfache des Argumentwertes in Register `$a0` bestimmt.

/ 4

10. Fließbandverarbeitung

10 Punkte

- (a) i. Identifizieren Sie alle Datenhazards, die bei der Verarbeitung der folgenden Instruktionsfolge in der MIPS-Pipeline (IF-ID-EX-MEM-WB) auftreten.

/ 5

```

add $3, $4, $2
sub $5, $3, $1
lw  $7, 100($5)
add $8, $7, $2

```

- ii. Welche Hazards können mittels Forwarding aufgelöst werden, wo sind Leertakte (stalls) unumgänglich?

- iii. Skizzieren Sie die verzahnte Verarbeitung der Instruktionsfolge und markieren Sie dabei eventuell notwendiges Forwarding. Wieviele Takte werden für die Ausführung der Instruktionsfolge insgesamt benötigt?

- (b) Ordnen Sie die nebenstehende Instruktionsfolge so um, dass die Fließbandverarbeitung (mit forwarding) eine minimale Anzahl von Takten benötigt. Dabei soll die Bedeutung des Programms nicht verändert werden.

```

lw  $2, 100($6)
lw  $3, 100($7)
add  $4, $2, $3
add  $6, $3, $5
sub  $8, $4, $6
addi $8, $8, 4
lw  $7, 300($8)
slt  $1, $8, $7

```

/ 5

Skizzieren Sie die verzahnte Verarbeitung der Instruktionsfolge und markieren Sie dabei eventuell notwendiges Forwarding. Wieviele Takte werden für die Ausführung der Instruktionsfolge insgesamt benötigt?