

Aufgabenblätter zur Prüfung

Digitaltechnik und Entwurfsverfahren & Rechnerorganisation
und

Technische Informatik I/II

am 27. Februar 2015, 14:00 – 16:00 Uhr

- Beschriften Sie bitte gleich zu Beginn jedes Lösungsblatt deutlich lesbar mit Ihrem Namen und Ihrer Matrikelnummer.
- Diese Aufgabenblätter werden nicht abgegeben. Tragen Sie Ihre Lösung deshalb ausschließlich in die für jede Aufgabe vorgesehenen Bereiche der Lösungsblätter ein. Lösungen auf separat abgegebenen Blättern werden nicht gewertet.
- Außer Schreibmaterial sind während der Klausur keine Hilfsmittel zugelassen. Täuschungsversuche durch Verwendung unzulässiger Hilfsmittel führen unmittelbar zum Ausschluss von der Klausur und zur Note „nicht bestanden“.
- Soweit in der Aufgabenstellung nichts anderes angegeben ist, tragen Sie in die Lösungsblätter bitte nur die Endergebnisse ein. Die Rückseiten der Aufgabenblätter können Sie als Konzeptpapier verwenden. Weiteres Konzeptpapier können Sie auf Anfrage während der Klausur erhalten.
- Halten Sie Begründungen oder Erklärungen bitte so kurz wie möglich. (Der auf den Lösungsblättern für eine Aufgabe vorgesehene Platz steht übrigens in keinem Zusammenhang mit dem Umfang einer korrekten Lösung!)
- Die Gesamtpunktzahl beträgt 90 Punkte. Zum Bestehen der Klausur sind mindestens 40 Punkte zu erreichen.

Viel Erfolg und viel Glück!

Aufgabe 1 *Schaltfunktionen* (10 Punkte)

Gegeben sei die vollständig definierte Schaltfunktion $f(w, x, y, z)$:

$$f(w, x, y, z) = (w \vee \bar{y}) (\bar{w} \vee x \vee y) (\bar{w} \vee \bar{x} \vee \bar{z})$$

1. Tragen Sie die Schaltfunktion $f(w, x, y, z)$ in das im Lösungsblatt vorbereitete KV-Diagramm ein. 3 P.
2. Geben Sie alle Primimplikanten von f an. Kennzeichnen Sie, bei welchen Primimplikanten es sich um Kernprimimplikanten handelt. 2 P.
3. Geben Sie eine disjunktive Minimalform (DMF) von $f(w, x, y, z)$ an. 1 P.
4. Überführen Sie die DMF in ein Schaltnetz, in dem alle Eingangsvariablen nur bejaht zur Verfügung stehen und nur NOR-Gatter verwendet werden. 4 P.

Aufgabe 2 *Laufzeiteffekte* (6 Punkte)

Gegeben ist das in Abbildung 1 dargestellte Schaltnetz. NAND-, NOR- und OR-Gatter haben eine Totzeit von 5 ns , das XOR-Gatter von 7 ns und der Inverter von 2 ns . Die Eingangsvariablen a , b und c wechseln zum Zeitpunkt $t = 0$ *gleichzeitig* von 0 auf 1.

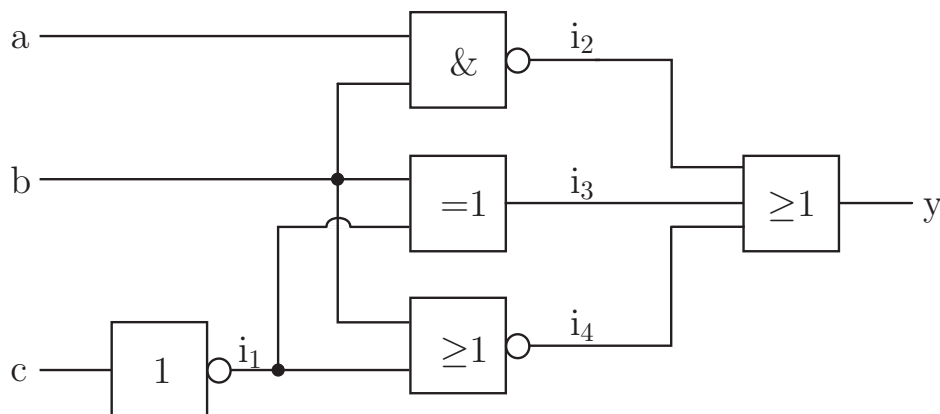


Abbildung 1: Schaltnetz

1. Geben Sie die Verläufe der Signale i_1 , i_2 , i_3 , i_4 und y an, indem Sie das im Lösungsblatt angegebene Zeitdiagramm vervollständigen. 3 P.
2. Treten im Zeitverlauf Hasardfehler auf? Falls ja, um welchen Typ handelt es sich bei dem zu Grunde liegenden Hasard? Begründen Sie Ihre Antwort. 3 P.

Aufgabe 3 *Schaltwerke*

(7 Punkte)

Gegeben sei die nachstehende Ablauftabelle eines Automaten mit den symbolischen Zuständen A , B , C und D (Tabelle 1).

Zustand	Eingabe		Folgezustand	Ausgabe	
Z^t	e_1^t	e_0^t	Z^{t+1}	y_1	y_0
A	1	–	D	0	0
A	0	0	B	0	1
A	0	1	A	1	1
B	–	0	C	1	1
B	–	1	A	0	1
C	–	0	A	1	0
C	–	1	C	0	1
D	–	–	A	0	0

Tabelle 1: Ablauftabelle des Automaten

1. Um welchen Automatentyp handelt es sich beim angegebenen Automaten? 1 P.
2. Geben Sie den Automatengraphen des Automaten an. 2 P.

Ein Mealy-Schaltwerk für ein Codeschloss soll einen Türöffner nur dann betätigen, wenn der Code 1 2 3 auf einer Zehnertastatur in dieser Reihenfolge eingegeben wird. Folgende Eingangsvariablen stehen zur Verfügung:

$T_1 = 1$ Taste 1 gedrückt (sonst $T_1 = 0$)

$T_2 = 1$ Taste 2 gedrückt (sonst $T_2 = 0$)

$T_3 = 1$ Taste 3 gedrückt (sonst $T_3 = 0$)

$T_f = 1$ Taste 4-9 bzw. 0 gedrückt (sonst $T_f = 0$)

Das Ausgangssignal $\ddot{O}$ (Öffnen) muss zum Öffnen der Tür nur kurzzeitig $\ddot{O} = 1$ sein.

3. Erstellen Sie den Automatengraphen für das Schaltwerk. 4 P.

Aufgabe 4 *CMOS*

(10 Punkte)

1. Die Schaltfunktion

5 P.

$$y = (a, b, c) = \bar{a} \vee \bar{b}c$$

soll in der CMOS-Technologie realisiert werden. Es stehen Ihnen zwei NAND-Gatter und ein Inverter-Gatter zur Verfügung. Formen Sie die Schaltfunktion entsprechend um und geben Sie das resultierende Transistor-Schaltbild an.

2. Was ist der Unterschied zwischen einem n-Kanal- und einem p-Kanal-MOSFET? Warum werden pMOS-Transistoren in CMOS-Schaltungen so angeordnet, dass sie eine logische Eins an den Ausgang durchschalten, während nMOS-Transistoren eine logische Null durchschalten?

2 P.

3. Skizzieren Sie den Aufbau eines pMOS-Transistors. Beschriften Sie alle Bestandteile. Aus Ihrer Zeichnung müssen die unterschiedlich dotierten Zonen und die Anschlüsse des Transistors eindeutig erkennbar sein.

3 P.

Aufgabe 5 *Rechnerarithmetik & Codes* (12 Punkte)

1. Addieren Sie die beiden BCD-Zahlen:

2 P.

$$\begin{array}{r} 0100 \ 1001 \\ + \ 0101 \ 0111 \\ \hline \end{array}$$

Der Rechenweg soll klar ersichtlich dargestellt werden.

2. Gegeben sei das folgende 12-Bit-Fließkommaformat:

11	10	6	5	0
Vz	Charakteristik	Mantisse		

Dabei gilt:

- Basis: $b = 2$
 - $V_z = 0$ für positive Zahlen, $V_z = 1$ für negative Zahlen.
 - Der Exponent Exp ist in Exzess-15-Darstellung (Charakteristik = $Exp + 15$). Die Werte der Charakteristik 0 und 31 dienen zur Darstellung der Null und ∞ .
 - Die Mantisse ist, wie beim IEEE-Format, in normalisierter Form mit einer impliziten 1 links vom Komma repräsentiert.
- (a) Stellen Sie die Zahl $-1,875_{10}$ im obigen Format dar. 1 P.
- (b) Geben Sie die kleinste darstellbare normalisierte positive Zahl in diesem Format an. Welchen Dezimalwert hat sie? (Sie brauchen die Zweierpotenzen nicht explizit auszurechnen.) 1 P.
- (c) Geben Sie die größte darstellbare normalisierte positive Zahl in diesem Format an. Welchen Dezimalwert hat sie? (Sie brauchen die Zweierpotenzen nicht explizit auszurechnen.) 1 P.
- (d) Wie viele verschiedene Darstellungen der Zahl 0 gibt es in diesem Format? 1 P.
- (e) Addieren und multiplizieren Sie die folgenden, im obigen Format angegebenen Zahlen: 4 P.

$$A = \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|} \hline 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ \hline \end{array}$$

$$B = \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|} \hline 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ \hline \end{array}$$

Stellen Sie die Ergebnisse in normalisierter Form dar und tragen Sie sie in die vorbereiteten Felder auf dem Lösungsblatt ein. Geben Sie den Rechenweg an.

Hinweis: Bei einem Überlauf der Mantisse wird nicht gerundet, sondern nur abgeschnitten.

3. Welche Vor- und Nachteile hat die BCD-Arithmetik gegenüber der Dual-Arithmetik? 1 P.
4. Wie viele Prüfbits müssen bei Verwendung des Hamming-Codes mindestens zu einem 32-Bit-Datenwort hinzugefügt werden, um einen Ein-Bit-Fehler korrigieren zu können? 1 P.

Aufgabe 6 *MIMA-Architektur*

(8 Punkte)

Die MIMA ist die Ihnen aus der Vorlesung bekannte mikroprogrammierte Minimalmaschine (**siehe Beiblatt: Architektur der MIMA**), die nach dem von-Neumann-Prinzip aufgebaut ist, d. h. Maschinenbefehle werden sequentiell abgearbeitet. In der Lese-Phase wird ein über IAR adressierter Befehl aus dem Speicher gelesen und im IR abgelegt. Die Lese-Phase dauert 5 Taktzyklen. Im 6. Taktzyklus wird der Befehl dekodiert (Dekodier-Phase). Die Ausführungsphase beginnt im 7. Taktzyklus. Nach der Ausführung des Befehls folgt ein Zugriff auf den nächsten Befehl.

Nehmen Sie an, dass ein Hauptspeicherzugriff (Lesen und Schreiben) drei Takte dauert und währenddessen $R = 1$ bzw. $W = 1$ sein muss. Eine ALU-Operation sei nach einem Takt abgeschlossen.

1. Geben Sie das Mikroprogramm für die Lese-Phase (Fetch-Phase) in Register-Transfer-Schreibweise an, d. h. in der Form

5 P.

1. Takt: $IR \rightarrow SAR; \quad R = 1$
2. Takt:
:
:

2. Wie viele verschiedene Operationen kann die ALU der MIMA ausführen?

1 P.

3. Wie viele verschiedene Befehle können für die MIMA maximal implementiert werden?

1 P.

4. Wie werden aus dem 24 Bit breiten Datenbus die 20 Bit breiten Adressen extrahiert?

1 P.

Aufgabe 7 MIPS-Assembler

(9 Punkte)

1. Das folgende MIPS-Programm sollte die Länge eines Strings `str` berechnen. Leider haben sich bei der Implementierung einige Fehler eingeschlichen.

3 P.

```

                .data
str:            .asciiz "TI ist spitze"

                .text
                .globl main
main:           la $a0, str
                jal strlen
                move $v0, $a0
                li $v0, 1           # print integer
                syscall
                li $v0, 10          # exit
                syscall

strlen:         li $v0, 0
label_0:        lb $t0, 0($a0)
                bnez $t0, label_1
                addi $v0, $v0, 1
                addi $a0, $a0, 1
                b label_0
label_1:        jr $ra

```

Finden Sie diese Fehler und korrigieren Sie das Programm so, dass es korrekt arbeitet, wenn man Ihnen verraten würde, dass

- das Argument beim Systemaufruf `print_int` im Register `$a0` stehen muss,
- die Nummern der Systemaufrufe im Programm richtig sind.

2. Führen Sie den folgenden MIPS-Code aus und geben Sie die Änderungen in den Register- und Speicherinhalten an. Verwenden Sie die im Lösungsblatt angegebenen Tabellen.

4 P.

	Registersatz		Hauptspeicher	
	Register	Inhalt	Adresse	Inhalt
<code>addi \$t3, \$t0, 0x20</code>	<code>\$t0</code>	0x10	<code>\$0x20</code>	0x22
<code>lw \$t1, 0(\$t3)</code>	<code>\$t1</code>	0x14	<code>\$0x24</code>	0x30
<code>and \$t4, \$t1, \$t0</code>	<code>\$t2</code>	0x16	<code>\$0x28</code>	0x40
<code>sw \$t4, 0x04(\$t2)</code>	<code>\$t3</code>	0x28	<code>\$0x2C</code>	0x50
	<code>\$t4</code>	0x1234	<code>\$0x30</code>	0x60

3. Der MIPS-Prozessor besitzt bekannterweise einen Fließkomma-Arithmetik-Prozessor als Coprozessor. Warum dürfen bei Arithmetik-Operationen mit doppelter Genauigkeit nur die Register des Coprozessors mit gerader Registernummer verwendet werden?

2 P.

Aufgabe 8 *Pipelining*

(8 Punkte)

Das folgende Programmstück soll in der DLX-Pipeline abgearbeitet werden.

```
S1:      add   $t1, $zero, $zero
S2:      lw    $t3, 0x1500($zero)
S3:      lw    $t4, 0x5000($t1)
S4:      add   $t5, $t4, $t3
S5:      ori   $t5, $t5, 0x10
S6:      sw    $t5, 0x400($t1)
S7:      addi  $t1, $t1, 4
S8:      bnez  $t2, exit
S9:      srli  $t1, $t1, 2
S10:     sw    $t1, 0x2000($zero)
        ...
exit:    ...
```

1. Bestimmen Sie alle echten Datenabhängigkeiten und alle Steuerflussabhängigkeiten im Programmstück. 3 P.
2. Gehen Sie davon aus, dass *Forwarding*-Techniken implementiert sind. Zur Behebung von Pipelinekonflikten sollen NOP-Befehle (*No Operation*) in den Befehlsstrom eingefügt werden.
Ergänzen Sie das Programmstück durch das Einfügen von möglichst wenigen NOP-Befehlen, sodass alle Pipelinekonflikte behoben werden. 3 P.
3. Warum können Ausgabeabhängigkeiten (*output dependence*) und Gegenabhängigkeiten (*anti-dependence*) in der DLX-Pipeline nicht zu Konflikten führen? 2 P.

Aufgabe 9 *Cache-Speicher*

(12 Punkte)

1. Bei einem Cache-Speicher mit einer Speicherkapazität von 128 KByte ist die Hauptspeicheradresse in ein 16 Bit Tag-Feld, ein 11 Bit Index-Feld und ein 5 Bit Byte-Offset unterteilt. Geben Sie bei der Beantwortung der folgenden Fragen den Lösungsweg an.

(a) Bestimmen Sie die Blockgröße in Bytes.

1 P.

(b) Wie viele Einträge besitzt der Cache-Speicher?

1 P.

(c) Wie ist der Cache-Speicher organisiert?

2 P.

2. Es soll ein 5-fach-assoziativer Cache-Speicher (*5-way set associative cache*) mit 4 Sätzen und einer Blockgröße von 8 Byte realisiert werden. Nehmen Sie an, dass die Hauptspeicheradresse 32 Bit breit ist. Zur Verwaltung eines Cacheblocks wird nur ein Statusbit (*Valid*-Bit: V) verwendet.

3 P.

Bestimmen Sie den insgesamt erforderlichen Speicherbedarf zur Realisierung dieses Cache-Speichers.

3. Gegeben sei ein direkt-abgebildeter Cache-Speicher (*direct mapped cache*) mit einer Speicherkapazität von 32 Byte und einer Blockgröße von 8 Byte. Als Aktualisierungsstrategie wird ein Durchschreibverfahren (*write through policy*) mit *write no-allocate* verwendet. Bei dieser Aktualisierungsstrategie wird ein CPU-Datum bei einem *write miss* nur in den Speicher geschrieben. Bei einem *write hit* wird ein CPU-Datum sowohl in den Cache als auch in den Speicher geschrieben.

5 P.

Betrachten Sie die folgenden Lese- und Schreibzugriffe auf die in dezimaler Schreibweise angegebenen Adressen:

Adresse	0	16	48	48	56	8	8	56	32	0
read/write	r	r	w	r	r	r	r	w	w	r
Index	0	2								
Tag	0	0								
Hit/Miss	Miss									

Vervollständigen Sie die obige Tabelle im Lösungsblatt. Verwenden Sie dabei **Miss** für Cache-Miss und **Hit** für Cache-Hit.

Aufgabe 10 *Speicherbausteine & -verwaltung* (8 Punkte)

1. Wie unterscheiden sich statische und dynamische RAM-Bausteine in Bezug auf die Zugriffszeit und die Integrationsdichte? 2 P.
2. Wie viele Adressleitungen sind erforderlich bei einem Speicherbaustein mit einer Kapazität von 2048 Bits und einer 256×8 -Organisation? Begründen Sie Ihre Antwort. 1 P.
3. Wie ist ein ROM-Baustein mit der Speicherkapazität von 4096 Bits und 10 Adressleitungen organisiert? Begründen Sie Ihre Antwort. 1 P.

Gegeben sei eine Speicherverwaltungseinheit (MMU) mit einer Seitengröße von 1 Kbyte, 8 virtuellen Seiten und 4 physikalischen Seiten (Frames). Die Seitentabelle ist in Tabelle 2 angegeben.

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

Tabelle 2: Seitentabelle

4. Ermitteln Sie die physikalischen Adressen zu den folgenden virtuellen Adressen: 4 P.

3072, 1524, 2500, 1023

Lösungsblätter zur Klausur

Digitaltechnik und Entwurfsverfahren & Rechnerorganisation
und

Technische Informatik I/II

am 27. Februar 2015, 14:00 – 16:00 Uhr

Name:	Vorname:	Matrikelnummer:
-------	----------	-----------------

Digitaltechnik und Entwurfsverfahren/TI-1	
Aufgabe 1	von 10 Punkten
Aufgabe 2	von 6 Punkten
Aufgabe 3	von 7 Punkten
Aufgabe 4	von 10 Punkten
Aufgabe 5	von 12 Punkten

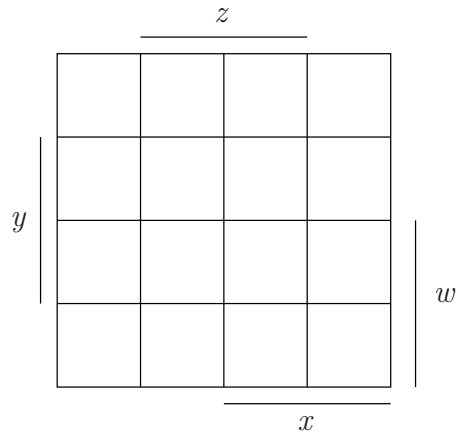
Rechnerorganisation/TI-2	
Aufgabe 6	von 8 Punkten
Aufgabe 7	von 9 Punkten
Aufgabe 8	von 8 Punkten
Aufgabe 9	von 12 Punkten
Aufgabe 10	von 8 Punkten

Gesamtpunktzahl:	
------------------	--

	Note:
--	-------

Aufgabe 1

1. KV-Diagramm:



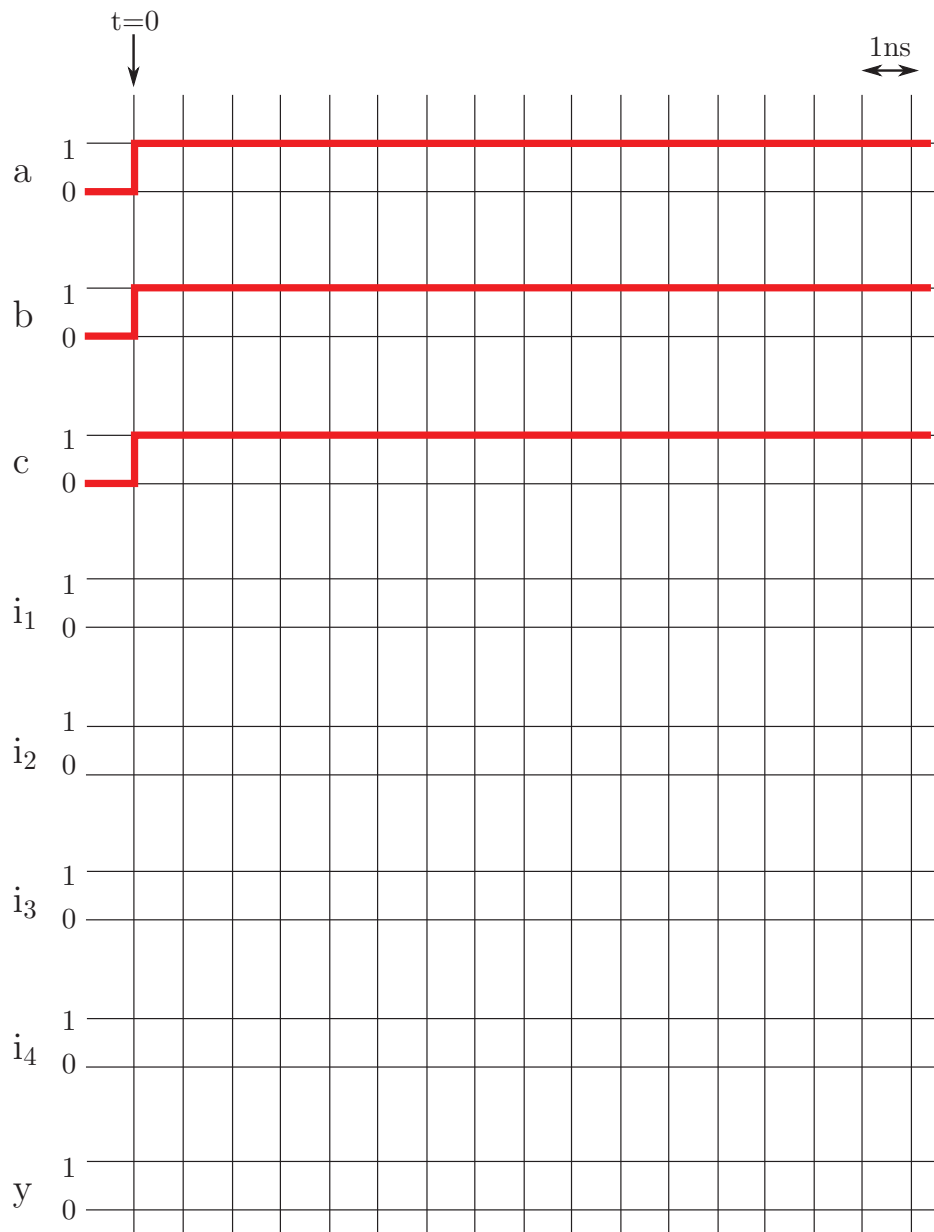
2. Primimplikanten und Kernprimimplikanten:

3. Disjunktive Minimalform:

4. Schaltnetz mit NOR-Gattern:

Aufgabe 2

1. Zeitdiagramm:



2. Hasardfehler (falls ja, Analyse):

Name:

Vorname:

Matr.-Nr.:

4

Aufgabe 3

1. Automatentyp:

2. Automatengraph:

3. Automatengraph:

Aufgabe 4

1. Umgeformte Schaltfunktion und Transistor-Schaltbild:
2. Unterschied zwischen n-Kanal- und einem p-Kanal-MOSFET:
3. Aufbau eines pMOS-Transistors:

Name:

Vorname:

Matr.-Nr.:

7

2 c) Größte darstellbare normalisierte positive Zahl:

--	--	--

Dezimalwert:

2 d) Verschiedene Darstellungen der Null:

2 e)

Aufgabe 6

1. Mikroprogramm für die Lese-Phase:

1. Takt:

2. Takt:

:

2. Anzahl der ALU-Operationen:

3. Maximale Anzahl der MIMA-Befehle:

4. Adressen bei MIMA:

Aufgabe 7

1. Fehler im Programm: (Geben Sie bitte nur die fehlerhaften Zeilen an!)

2. Register- und Speicherinhalte nach der Ausführung:

Registersatz		Hauptspeicher	
Register	Inhalt	Adresse	Inhalt
\$t0		\$0x20	
\$t1		\$0x24	
\$t2		\$0x28	
\$t3		\$0x2C	
\$t4		\$0x30	

3. Nur gerade Registernummern bei doppelter Genauigkeit:

Aufgabe 8

1. Echte Datenabhängigkeiten und Steuerflussabhängigkeiten:
2. Behebung der Pipelinekonflikte:
3. Ausgabeabhängigkeiten und Gegenabhängigkeiten in der DLX-Pipeline:

Aufgabe 10

1. Statische und dynamische RAM-Bausteine:

2. Anzahl der Adressleitungen:

3. Organisation des ROM-Bausteins:

4. Physikalische Adressen:

Virtuelle Adresse	Physikalische Adresse
3072	
1524	
2500	
1023	