

Prof. Dr.-Ing. Jürgen Teich
Lehrstuhl für Informatik 12
(Hardware-Software-Co-Design)
Universität Erlangen-Nürnberg

Klausur Grundlagen der Technischen Informatik

21. Juli 2008

Name	
Matrikelnummer	
Studienrichtung	

Aufgabe	1	2	3	4	Σ
max. Punkte	25	25	25	25	100
erreichte Punkte					
Note					

Aufgabe 1 (Zahlensysteme)

(20 Punkte)

a) Konvertieren Sie die Zahlen 10001_{10} und 2046_{10} zunächst in das Binär- und dann in das Hexadezimalsystem. (3 Punkte)

b) Beschreiben Sie den allgemeinen Aufbau einer Zahl N in einem polyadischen Zahlensystem. (2 Punkte)

c) Konvertieren Sie $(0|1000\ 0101|100\ 1000\ 0000\ 0000\ 0000)$ aus der IEEE 754-Gleitkommadarstellung in das Dezimalsystem. Format: V : Vorzeichen, E : Exponent, M : Mantisse

V	E		M		
31	30	23	22	0	

(3 Punkte)

d) Wandeln Sie 17_{10} in ihre Binärdarstellung und multiplizieren Sie in der Binärdarstellung mit acht. Wandeln Sie das Ergebnis wieder in die Dezimaldarstellung um. (2 Punkte)

e) Subtrahieren Sie 56_{10} von 107_{10} im Dualsystem. Die Berechnung und das Ergebnis sind in der Zweierkomplement-Darstellung anzugeben. (4 Punkte)

f) Beantworten Sie folgende Auswahlfragen. Jede richtige Antwort ergibt einen Punkt, jede falsche Antwort führt zu einem Punktabzug, nicht beantwortete Fragen werden nicht gewertet, weniger als null Punkte sind nicht möglich. (6 Punkte)

1. In der Gleitkommadarstellung nach IEEE 754 gibt es nur eine Darstellung des Wertes null.

wahr falsch

2. Im Vergleich zu vorzeichenlosen Binärwerten können im Zweierkomplement mit der gleichen Anzahl an Bits nur halb so viele positive Werte dargestellt werden.

wahr falsch

3. Alle Zahlen zwischen Null und Eins lassen sich ohne Rundungsfehler in die Gleitkommadarstellung nach IEEE 754 umwandeln.

wahr falsch

4. Die Addition zweier Zahlen im Zweierkomplement mit 32 Bit ist aufwendiger als die Addition zweier vorzeichenloser Binärzahlen mit 32 Bits.

wahr falsch

5. Bei der Umwandlung von Zahlen im Hexadezimalsystem in Zahlen zur Basis drei kann es zu Rundungsfehlern kommen.

wahr falsch

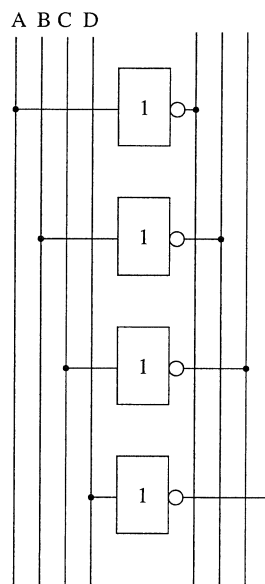
6. In einem polyadischen Zahlensystem ergibt sich die Wertigkeit einer Stelle als Produkt ihrer Position multipliziert mit der Basis.

wahr falsch

Aufgabe 2 (Minimierung von Schaltfunktionen)

(20 Punkte)

	A	B	C	D	y_1
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	-
5	0	1	0	1	1
6	0	1	1	0	-
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	-
13	1	1	0	1	0
14	1	1	1	0	1
15	1	1	1	1	0

(a) Funktionstabelle für die Schaltfunktion y_1 .(b) Vorlage für das Schaltnetz der Schaltfunktion y_2 .

- a) Beantworten Sie folgende Auswahlfragen. Jede richtige Antwort ergibt einen Punkt, jede falsche Antwort führt zu einem Punktabzug, nicht beantwortete Fragen werden nicht gewertet, weniger als null Punkte sind nicht möglich. (3 Punkte)

1. Eine disjunktive Normalform (DNF) besteht aus:

- Mintermen Medtermen Maxtermen

2. Die Primterme einer konjunktiven Minimalform (KMF) heißen auch:

- Primimplikanten Primimplikate Primimplikantoren

3. Kann man Schaltfunktionen in disjunktiver Normalform (DNF) in eine konjunktive Normalform (KNF) überführen?

- ja nein nicht jede

- b) Bestimmen Sie eine disjunktive Minimalform der als Funktionstabelle (Teilabbildung (a)) gegebenen Schaltfunktion y_1 unter Verwendung eines Symmetriediagramms.

Anmerkung: In der Funktionstabelle bezeichnet "–" ein "don't care", d.h. es ist egal, ob an dieser Stelle eine "1" oder eine "0" steht. (4 Punkte)

- c) Bestimmen Sie alle Primeinsblöcke für die Schaltfunktion $y_2 = f(A, B, C, D) = ABCD + ABC\bar{D} + \bar{A}BCD + \bar{A}\bar{B}CD$ mit Hilfe des Quine/McCluskey-Verfahrens. (6 Punkte)

- d) Welches Problem kann mit einer Überdeckungstabelle gelöst werden? Tritt dieses Problem im Aufgabenteil c) auf? Begründen Sie ihre Aussage. (2 Punkte)

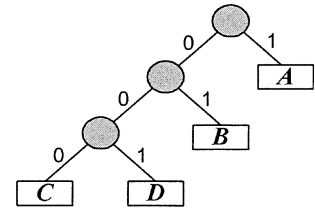
- e) Ermitteln Sie eine kostenminimale Implementierung für die Schaltfunktion y_2 aus Aufgabenteil c) unter Verwendung des Petrick-Verfahrens. Als Kostenmaß gilt die Anzahl der Literale der verwendeten Terme zuzüglich der Summe der verwendeten Terme. (3 Punkte)

- f) Zeichnen Sie die minimierte Schaltfunktion aus Aufgabenteil e) oder die Schaltfunktion y_2 aus Aufgabenteil c) als Schaltnetz. Sie können hierzu die Vorlage in der obigen Teilabbildung (b) verwenden. (2 Punkte)

Aufgabe 3 (Codierung und Automaten)

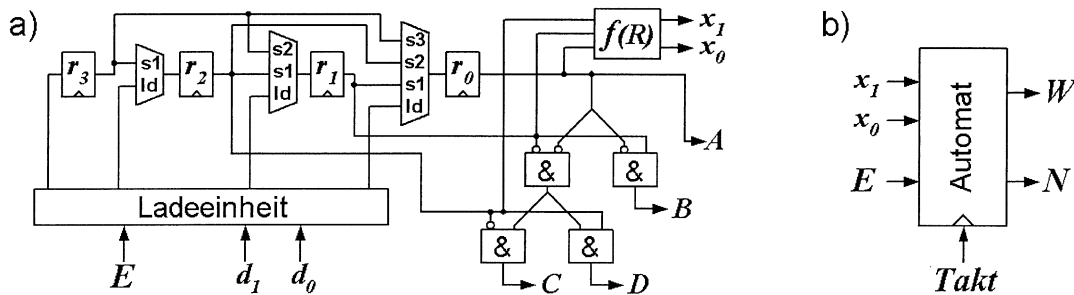
(20 Punkte)

Gegeben sei der rechts abgebildete Huffman-Codierungsbaum für ein Alphabet $\{A, B, C, D\}$.



- a) Geben Sie die Huffman-Codierung der Zeichenkette *DABAC* an. Wieviele Bits werden benötigt. (1 Punkt)
- b) Wieviele Bits würde die Codierung der Zeichenkette minimal benötigen, wenn das Alphabet mit fester Wortbreite codiert sei. (1 Punkt)
- c) Auf wieviel Prozent lässt sich die Codierung aus b) mit fester Wortbreite im besten Fall komprimieren, wenn die durch den Huffman-Baum angegebene Codierung verwendet wird. (1 Punkt)

Im folgenden soll ein Steuerungsautomat für den unten abgebildeten Huffman-Hardware-Dekomprimierer (Abb. a) entworfen werden. Dieser Automat enthält als internen Zustand den Füllstand des Auffangregisters bestehend aus den Flipflops r_0, r_1, r_2 und r_3 . Dieses Auffangregister hat die Aufgabe die in ihm befindenden Eingangsdaten nach dem Decodieren eines Zeichens so nach rechts zu schieben, dass das nächste Codewort im nächsten Taktzyklus korrekt an dem mit Logikgattern implementierten Huffman-Baum anliegt. Sollte hierdurch Platz für ein neues zwei Bit breites Eingangswort ($d_1 d_0$) in dem Auffangregister geschaffen werden, so wird dieses durch die Ladeeinheit rechtsbündig im gleichen Takt in den freiwerdenden Bereich eingefügt. Ein rechtsbündiges Auffüllen des Auffangregisters findet immer dann statt, wenn 1. Platz und 2. gültige Eingangsdaten vorhanden sind. Neue gültige Eingangsdaten werden mit $E = 1$ signalisiert.



Der Automat besitzt zwei Ausgänge. Mit $W = 1$ wird signalisiert dass ein Eingangswort ($d_1 d_0$) eingelesen worden ist und $N = 1$ zeigt an, dass ein neues gültiges Zeichen decodiert worden ist. Als Beispiel sei angenommen dass der Füllstand aktuell 3 betrage und der Inhalt von $r_0 = 0$ und von $r_1 = 1$ betrage. Es kann somit das nächste Zeichen (hier *B*) decodiert werden was unmittelbar mit $N = 1$ angezeigt wird. Sind neue Eingangsdaten vorhanden ($E = 1$), so werden diese in r_1 und r_2 geladen, was durch $W = 1$ signalisiert wird. Sind keine neuen Daten verfügbar ($E = 0$), so ist $W = 0$ und der Füllstand verringert sich auf 1. Vor dem Eingang des Automaten (Abb. b) befindet sich ein Schaltnetz $F(R) = F(r_0, r_1, r_2) = \{x_1, x_0\}$, welches anhand der im Auffangregister befindlichen Daten die freiwerdenden Stellen des Auffangregisters als Binärzahl angibt.

- d) Vieviele Zustände besitzt der beschriebene Automat? (1 Punkt)
- e) Um welchen Automatentypen handelt es sich? (1 Punkt)
- f) Geben Sie die Funktion des Schaltnetzes $F(R) = \{x_1, x_0\}$ als Funktionstabelle und in disjunktiver Minimalform (DMF) an. (4 Punkte)
- g) Zeichnen Sie den Automatengraphen und geben Sie dabei sämtliche Ein- und Ausgaben an. (8 Punkte)
- h) Geben Sie die Ausgabefunktion für W als disjunktiver Minimalform (DMF) an. (3 Punkte)

ß ä des Huffman

Für eine Anwendung aus dem Bereich der Bioinformatik soll ein Automat erstellt werden, der die Basenpaare der DNA erkennen und einordnen kann. Die Basen der DNA sind Adenin (A), Thymin (T), Guanin (G) und Cytosin (C). Der Automat soll zwei aufeinanderfolgende Basen (ein Basenpaar) auf ihre Gültigkeit hin überprüfen. Die Basenpaare AT und GC sind gültige Basenpaare.

Die Reihenfolge, in der die Basen eingelesen werden, spielt hierbei keine Rolle. Der Automat soll ferner *bereit*, *gültig*, *ungültig* und *identisch* ausgeben können.

Die Kodierung der Basen durch die binären Eingabevariablen i_0 und i_1 , sowie der Ausgabevariablen o_0 und o_1 entnehmen Sie bitte den oben dargestellten Tabellen.

- a) Im Folgenden soll der Automat einmal als Mealy- und einmal als Moore-Automat entwickelt werden. Der Automat ist wie folgt spezifiziert: Der Anfangszustand des Automaten ist *ready*, die Ausgabe lautet *bereit*. Daraufhin werden zwei Basen *nacheinander* eingelesen und die Gültigkeit (*gültig*, *ungültig* oder *identisch*) ausgegeben. Anschließend kehrt der Automat in den Zustand *ready* zurück und ist bereit, das nächste Basenpaar auszuwerten. Hinweis: Nach dem Einlesen der ersten Base soll die Ausgabe des Automaten weiterhin *bereit* bleiben.
 - a.1) Erstellen Sie den Automatengraphen für den spezifizierten Automaten in Form eines Moore-Automaten mit maximal 8 Zuständen. (4 Punkte)
 - a.2) Erstellen Sie den Automatengraphen für den spezifizierten Automaten in Form eines Mealy-Automaten mit maximal 5 Zuständen. (5 Punkte)
- b) Im Folgenden soll die Zustandsübergangstabelle für den von Ihnen entwickelten Mealy-Automaten aufgestellt werden. Hierfür sollen Sie die 5 Zustände durch die binären Variablen q_0 , q_1 und q_2 kodieren. Vervollständigen Sie die Zustandsübergangstabelle. (4 Punkte)

Tragen Sie Ihre Lösung(en) hier ein:

q_2	q_1	q_0	i_1	i_0	o_1	o_0	q_2	q_1	q_0
0	0	0	0	0					
0	0	0	0	1					
0	0	0	1	0					
0	0	0	1	1					
0	0	1	0	0					
0	0	1	0	1					
0	0	1	1	0					
0	0	1	1	1					
0	1	0	0	0					
0	1	0	0	1					
0	1	0	1	0					
0	1	0	1	1					
0	1	1	0	0					
0	1	1	0	1					
0	1	1	1	0					
0	1	1	1	1					
1	0	0	0	0					
1	0	0	0	1					
1	0	0	1	0					
1	0	0	1	1					

(d) Symmetriediagramm

(c) Zustandsübergangstabelle

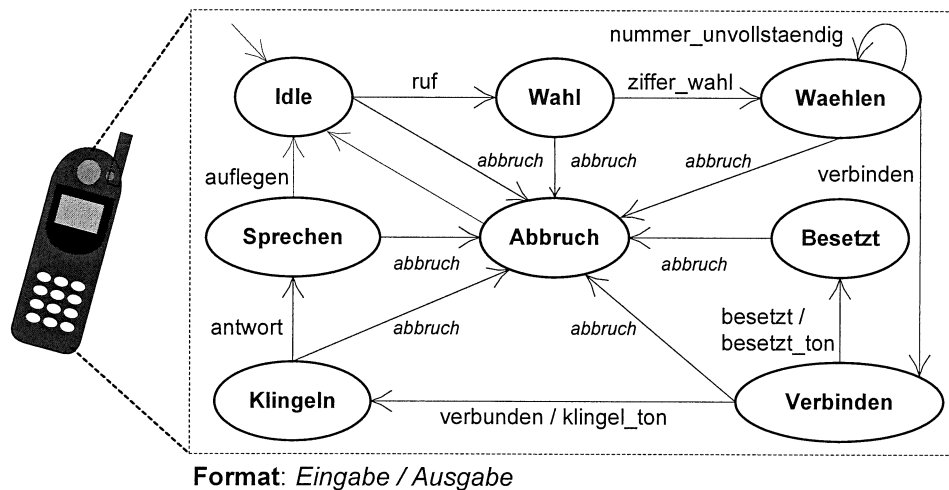
- c) Entwickeln Sie eine disjunktive Minimalform (DMF) für den Ausgang o_0 des Mealy-Automaten unter Verwendung des oben gegebenen Symmetriediagrammes.
Achten Sie auf Don't-Cares! (5 Punkte)
- d) Bestimmen Sie ein Schaltwerk des Mealy-Automaten ausgehend von der Zustandsübergangstabelle unter *ausschließlicher Verwendung von AND- und OR-Gattern, sowie D-Flipflops.* (2 Punkte)

Aufgabe 4 (VHDL)

(20 Punkte)

In dieser Aufgabe soll ein synchron getakteter Steuerungsautomat für z.B. ein Mobiltelefon in VHDL beschrieben werden:

- Wenn das Mobiltelefon eingeschaltet wird, geht der Steuerautomat in den Idle-Zustand über
- Ansonsten funktioniert die Steuerung gemäß folgendem Zustandsdiagramm:



- a) Geben Sie die Schnittstellenbeschreibung des abgebildeten Automaten in Form einer Entity in VHDL an. Überlegen Sie zuerst, ob Signale Ein- und Ausgabesignale, bzw. intern oder extern sind. (6 Punkte)
- b) Geben Sie eine Implementierung des Automaten (siehe Abbildung) in Form einer VHDL Architecture-Beschreibung an. (Die Verwendung von IEEE-Libraries kann weggelassen werden.) (12 Punkte)
- c) Welche zusätzliche spezielle VHDL-Beschreibung wird benötigt, um den Automaten in einem Simulationsprogramm zu simulieren? (1 Punkt)
- d) Welche Signale müssen in der speziellen Beschreibung aus Teilaufgabe c) unbedingt gesetzt/-initialisiert werden? (1 Punkt)