

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

Klausur

Grundlagen der Technischen Informatik

15. April 2009

Name	
Matrikelnummer	
Studienrichtung	

Aufgabe	1	2	3	4	Σ
max. Punkte	20	20	20	20	80
erreichte Punkte					
Note					

Aufgabe 1 (Zahlensysteme) (20 Punkte)

- a) Welches ist die größte mit n Bit darstellbare Dezimalzahl? (1 Punkt)
- b) Wieviele verschiedene Zahlen lassen sich mit einer n -stelligen Oktalzahl darstellen? (1 Punkt)
- c) Wie lautet die Zahl 1265_{10} in vorzeichenloser Binärdarstellung? (2 Punkte)
- d) Wieviel Bits weniger werden für die Binärzahl aus Aufgabe c) gebraucht gegenüber einer Codierung der Zahl im BCD-Format? (1 Punkt)
- e) Wie lautet die Zahl aus Aufgabe c) im Hexadezimalsystem? (1 Punkt)
- f) Wie lautet der Zahlenbereich des Zweierkomplements für eine n -Bit Zahl? (1 Punkt)
- g) Subtrahieren Sie 26 von 102. Die Berechnung und das Ergebnis sind in der 2er-Komplement-Darstellung anzugeben. (4 Punkte)
- h) Geben Sie die Zahl $0,375_{10}$ im Binärformat an! (2 Punkte)
- i) In dieser Aufgabe soll mit 8-Bit Gleitkommazahlen gearbeitet werden. Diese werden analog zum IEEE-Format gebildet!
 Das Format der Gleitkommazahl sieht dabei wie folgt aus:
Vorzeichen (1 Bit), Exponent (3 Bit), Mantisse (4 Bit)
- | | | | | | | | |
|-----|-----|---|---|-----|--|--|---|
| V | E | | | M | | | |
| 7 | 6 | 4 | 3 | | | | 0 |
- Führen Sie nun die Addition der beiden in diesem Format dargestellten Gleitkommazahlen 00011000 und 00101100 aus! (4 Punkte)
- j) Geben Sie eine maximale Menge von 4-Bit Codewörtern an, für die eine Korrektur von 1-Bit Fehlern möglich ist. Diese Menge muss auch das Wort 0001 enthalten. (3 Punkte)

Aufgabe 2 (Minimierung von Schaltfunktionen)

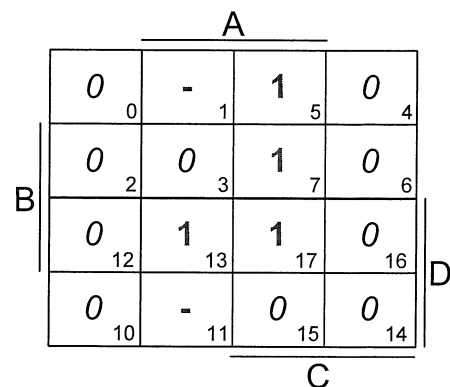
(20 Punkte)

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 konjunktive Normalform (KNF) besteht aus:
 - Mintermen
 - Modtermen
 - Maxtermen
2. Die Primterme einer disjunktiven Minimalform (DMF) heißen auch:
 - Primimplikanten
 - Plebiszite
 - Primimplikate
3. Kann man Schaltfunktionen in disjunktiver Normalform (DNF) in eine konjunktive Normalform (KNF) überführen?
 - ja
 - nein
 - nicht jede

b) Bestimmen Sie unter ausschließlicher Verwendung des Nelson-Verfahrens alle Primimplikanten für die in der nebenstehenden Abbildung bereits in ein Symmetriediagramm eingetragene Schaltfunktion. (7 Punkte)

c) Bestimmen Sie alle kostenminimalen Lösungen des Überdeckungsproblems aus der Teilaufgabe b) mit Hilfe des Petrick-Verfahrens. (10 Punkte)



Aufgabe 3 (Automaten und Flipflops)

(20 Punkte)

Ein klassischer, portabler Kassettenspieler (*Walkman*), soll durch einen von Ihnen entwickelten Automaten gesteuert werden. Der Walkman soll dabei über die Funktionen *Abspielen* (play), *Vorlauf* (forward), *schneller Vorlauf* (fast forward), einen *Rücklauf* (rewind) sowie *Stop* (stop) verfügen, die an die mechanische Steuerung weitergegeben werden. Der Walkman verfügt weiterhin über die vier Tasten *P* zum Abspielen, *FF* zur Auswahl des Vorlaufs, *RW* zur Auswahl des Rücklaufs sowie *S* zum Stoppen des Geräts. Das gleichzeitige Drücken mehrerer Tasten ist nicht möglich.

a) Vervollständigen Sie geeignet die Tabelle mit den fünf Funktionen, die durch die binären Zustandsvariablen q_0 , q_1 und q_2 kodiert werden. (1 Punkt)

Funktion	q_2	q_1	q_0
play	0	0	0
forward	0	0	1
fast forward	0	1	0
rewind			
stop			

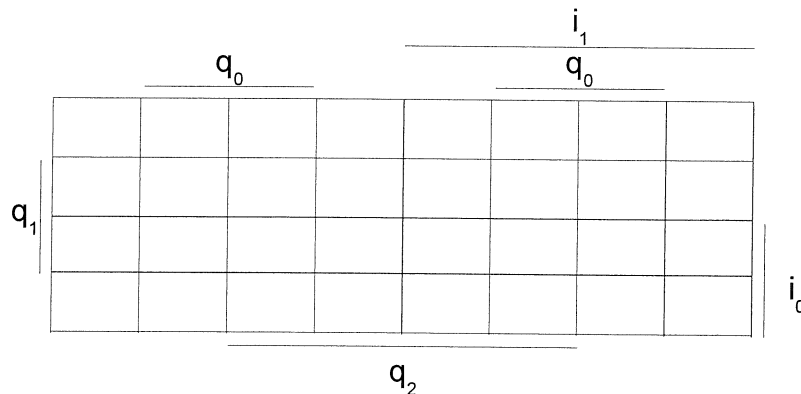
b) Vervollständigen Sie geeignet die Tabelle mit den vier Tasten, die durch die binären Eingabevariablen i_0 und i_1 kodiert werden. (1 Punkt)

Taste	i_1	i_0
P	0	0
FF	0	1
RW		
S		

- c) Im Folgenden soll der Automat als Medwedew-Automat entwickelt werden. Der Automat funktioniert hierbei wie folgt: Der Anfangszustand des Automaten ist *stop*. Durch Drücken der Taste *S* gelangt der Automat in den Zustand *stop*. Durch Drücken der Taste *P* gelangt der Automat in den Zustand *play*. Durch Drücken der Taste *FF* gelangt der Automat in den Zustand *forward*. Befindet sich der Automat im Zustand *forward*, führt das Drücken der Taste *FF* zum Wechsel in den Zustand *fast forward*. Befindet sich der Automat im Zustand *fast forward*, führt das Drücken der Taste *FF* zum Wechsel in den Zustand *forward*. Durch Drücken der Taste *RW* gelangt der Automat in den Zustand *rewind*. Erstellen Sie den Automatengraphen für den spezifizierten Automaten in Form eines Medwedew-Automaten mit maximal 5 Zuständen. (5 Punkte)
- d) Im Folgenden soll die Zustandsübergangstabelle inklusive der drei zur Realisierung der Ansteuerfunktion verwendeten, taktflankengesteuerten *T-FlipFlops* für den von Ihnen entwickelten Automaten aufgestellt werden. Vervollständigen Sie die nachfolgende Zustandsübergangstabelle inklusive der Ansteuerfunktion. (5 Punkte)

Aktueller Zustand			Eingabe		Nachfolgezustand			Ansteuerfunktion		
q_2	q_1	q_0	i_1	i_0	q'_2	q'_1	q'_0	T_2	T_1	T_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						

- e) Entwickeln Sie eine disjunktive Minimalform (DMF) der Ansteuerfunktion des T-FlipFlops T_2 unter Verwendung des gegebenen Symmetriediagrammes und tragen Sie ihre Lösung hier ein: (Achten Sie auf Don't-Cares sowie die vorgegebene Variablenordnung!) (4 Punkte)



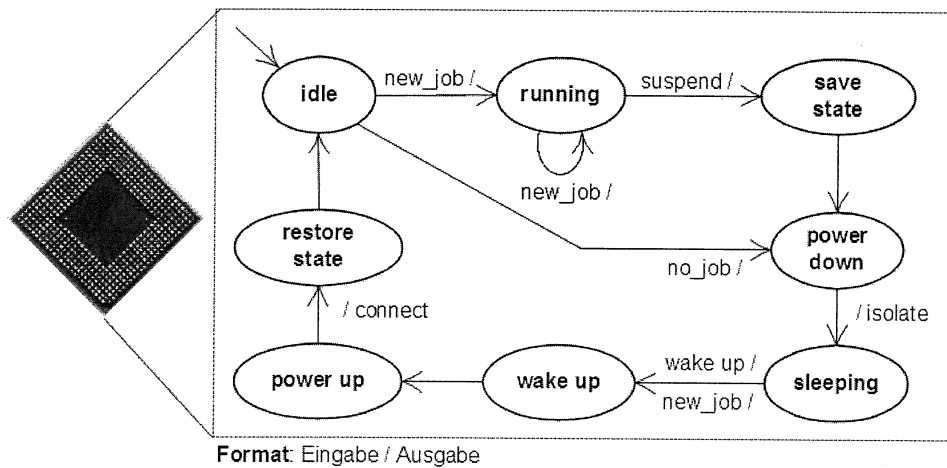
- f) Erstellen Sie eine minimierte Implementierung der Zustandsübergangsfunktion für die Zustandsvariable q_2 unter Verwendung der in Teilaufgabe e) ermittelten DMF. Verwenden Sie dabei ausschließlich AND- und OR-Gatter, Inverter sowie T-FlipFlops. (4 Punkte)

Aufgabe 4 (VHDL)

(20 Punkte)

In dieser Aufgabe soll ein synchron getakteter Steuerungsautomat für z.B. einen Prozessor-Chip mit geringem Energieverbrauch in VHDL beschrieben werden:

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



- 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.
Hinweis: alle Ein- bzw. Ausgabesignale nehmen nur Binärwerte 0 oder 1 an (6 Punkte)
- 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)
- Welche zusätzliche spezielle VHDL-Beschreibung wird benötigt, um den Automaten in einem Simulationsprogramm zu simulieren? (1 Punkt)
- Welche Signale müssen in der speziellen Beschreibung aus Teilaufgabe c) unbedingt gesetzt/-initialisiert werden? (1 Punkt)