

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

Klausur

Grundlagen der Technischen Informatik

04. Oktober 2013

Name	
Matrikelnummer	
Studienrichtung	

Aufgabe	1	2	3	4	5	Σ
max. Punkte	15	20	20	15	10	80
erreichte Punkte						
Note						

Organisatorische Hinweise

Bitte sorgfältig lesen und die Kenntnisnahme durch Unterschrift bestätigen

1. Bitte legen Sie Ihren Studentenausweis bereit.
 2. Als Hilfsmittel sind nur Schreibmaterialien und ein beidseitig handbeschriebenes DIN A4-Blatt zugelassen.
 3. Schmierpapier wird nicht abgegeben und auch nicht korrigiert.
 4. Sie können bei der Aufsicht zusätzliche Bearbeitungsblätter anfordern.
 5. Unleserliches wird nicht bewertet.
-

Erklärung

1. Im Falle einer während der Prüfung auftretenden Prüfungsunfähigkeit zeige ich dies sofort der Aufsicht an und befolge deren Anweisungen. Ich weiß, dass ich die volle Beweislast trage. Ich lasse mir das Formular des Prüfungsamts, das für diese Fälle vorgesehen ist, aushändigen und verfare nach den dort niedergelegten Richtlinien.
2. Ich weiß, dass im Falle des Täuschungsversuchs oder der Benutzung unerlaubter Hilfsmittel („Unterschleif“) der Prüfungsausschuss die Entscheidung treffen kann, die betroffene Prüfungsleistung als mit „nicht ausreichend“ bewertet gelten zu lassen.
3. Ich habe die obigen Hinweise zur Kenntnis genommen.

Erlangen, den 04. Oktober 2013

.....
Unterschrift

Einwilligung

Ich bin damit einverstanden, dass mein vorläufiges Ergebnis anonymisiert, jedoch unter Angabe der Matrikelnummer, am Mitteilungsbrett und auf der Webseite des Lehrstuhls für Informatik 12 veröffentlicht wird.

Die Bekanntgabe des vorläufigen Ergebnisses begründet keinen Rechtsanspruch.

Die Bekanntgabe des endgültigen Ergebnisses erfolgt ausschließlich durch das Prüfungsamt.

Erlangen, den 04. Oktober 2013

.....
Unterschrift

Aufgabe 1 (Zahlensysteme)

(15 Punkte)

- a) Wie lautet der Wertebereich einer mit n Bit darstellbaren vorzeichenlosen Binärzahl? (1 Punkt)
- b) Wie lautet der Wertebereich einer n -stelligen Binärzahl im Zweierkomplement? (1 Punkt)
- c) Welches ist die größte mit n Nibblen (Bezeichnung für Ziffern im Hexadezimalsystem) darstellbare Hexadezimalzahl? (1 Punkt)
- 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. (4 Punkte)
1. Die Zahl $(3,375)_{10}$ lässt sich ohne Rundungsfehler in die Gleitkommadarstellung nach IEEE 754 umwandeln. wahr falsch
 2. Die Zahl $(1,325)_{10}$ lässt sich ohne Rundungsfehler in die Gleitkommadarstellung nach IEEE 754 umwandeln. wahr falsch
 3. Bei der Umwandlung von Ganzzahlen im Hexadezimalsystem in Zahlen zur Basis drei kann es zu Rundungsfehlern kommen. wahr falsch
 4. Eine Multiplikationseinheit für zwei vorzeichenlose 4 Bit breite Binärzahlen a und b mit einem 4 Bit breitem Ergebnis $s = a \cdot b$ kann auch für die Multiplikation zweier 4-Bit-Zahlen im Zweierkomplement verwendet werden. wahr falsch

- h) Addieren Sie die beiden BCD-kodierten Zahlen $(0011\ 1000\ 0001)_{\text{BCD}}$ und $(0001\ 1001\ 1001)_{\text{BCD}}$. Die Addition ist im BCD-Format durchzuführen. Berücksichtigen Sie dabei die eventuell auftretenden Pseudotetraden. (4 Punkte)

- i) In dieser Aufgabe werden 8-Bit Gleitkommazahlen betrachtet. 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)

<i>V</i>	<i>E</i>	<i>M</i>
7	6 4 3	0

- Führen Sie nun die Multiplikation der beiden in diesem Format dargestellten Gleitkommazahlen $1\ 110\ 1000$ und $1\ 001\ 0010$ aus und geben Sie das Ergebnis im oben beschriebenen Format an! (4 Punkte)

Aufgabe 2 (Minimierung von Schaltfunktionen)

(20 Punkte)

- a) Realisieren Sie die gegebene Schaltfunktion $f_1(x_3, x_2, x_1, x_0) = \overline{x_1 x_2 + x_3 + x_4}$ als CMOS-Schaltung. (4 Punkte)

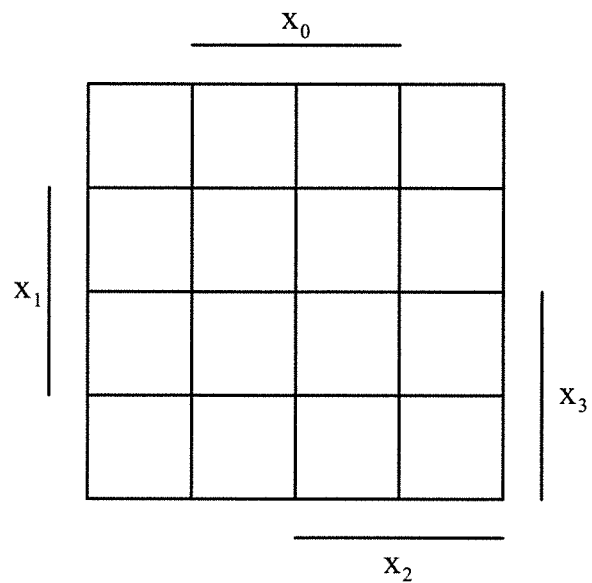
VDD

GND

- b) Realisieren Sie die Schaltfunktion $f_1(x_3, x_2, x_1, x_0)$ aus Aufgabe a) als Gatterschaltung, die nur aus NAND-Gattern besteht. Wie viele Transistoren werden für diese Schaltung benötigt? (4 Punkte)

- c) Gegeben sei folgende Funktionstabelle für die Schaltfunktion $f_2(x_3, x_2, x_1, x_0)$. Ermitteln Sie mit Hilfe eines Symmetriediagramms eine DMF dieser Schaltfunktion. (5 Punkte)

x_3	x_2	x_1	x_0	$f_2(x_3, x_2, x_1, x_0)$
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	-
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	-
1	1	1	0	0
1	1	1	1	0



Primimplikanten:

DMF:

- d) Gegeben seien die beiden Schaltfunktionen $f_3(x_2, x_1, x_0) = (\bar{x}_2 + x_1 + \bar{x}_0) \cdot (x_2 + \bar{x}_1)$ und $f_4(x_2, x_1, x_0) = \bar{x}_2 \bar{x}_1 + x_2 x_1 + \bar{x}_0$. Sind diese beiden Schaltfunktionen äquivalent? Begründen Sie Ihre Antwort. (3 Punkte)

- e) Gegeben sei folgende Überdeckungstabelle einer Schaltfunktion mit allen Primimplikanten, den von ihnen überdeckten Einsstellen sowie deren Kosten. Wenden Sie Vereinfachungsregeln für diese Tabelle an und ermitteln Sie anschließend mit Hilfe des Petrick-Verfahrens eine kostenminimale Überdeckung. (Teilschritte angeben!) (4 Punkte)

p_i	E_0	E_3	E_4	E_{17}	E_{20}	E_{25}	E_{30}	E_{34}	E_{36}	c_i
p_1			x					x		4
p_2							x	x		3
p_3		x		x						3
p_4	x				x	x				3
p_5							x	x	x	3
p_6						x		x	x	3
p_7				x	x	x		x		4
p_8	x	x						x		5

Aufgabe 3 (Automaten und Flipflops)

(20 Punkte)

Im Folgenden soll ein System für die Ansteuerung von Airbags im Automobil entworfen werden. Als Eingabedaten stehen vorverarbeitete Crash-Statusinformationen zur Verfügung, über welche die Ansteuerung entscheidet, ob und wenn ja welcher von drei zur Verfügung stehenden Airbags (*LAirbag*, *MAirbag*, *RAirbag*) gezündet werden muss. Beispielsweise sorgt der Signalwert *cLinks* oder *cRechts* dafür, dass der Airbag auf der jeweiligen Seite mittels der Ausgabe *LAirbag* bzw. *RAirbag* gezündet wird. Sollte direkt im Anschluss auch ein Crash auf der jeweils anderen Seite signalisiert werden, so wird statt der neu signalisierten Seite der Airbag *MAirbag* ausgelöst. Zusätzlich zu den elementaren Crash-Statusinformationen (*cLinks*, *cRechts*, *keinCrash*) steht noch ein Signalwert (*disabled*) zum Deaktivieren der Airbagansteuerung (Ausgabe *NoAction*) zur Verfügung (Re-Aktivierung kann nur mittels Eingabe *keinCrash* erfolgen). Sollte dieser Signalwert allerdings direkt nach einer Zündung eines Airbags auftreten, so wird dieser ignoriert, da es sich hierbei um eine Fehlabschaltung handeln könnte. Folglich ist die Reaktion auf diese Eingabe nur im Initialzustand *NoCrash* möglich, zu dem nach jedem Crash auch mittels *keinCrash* zurückgekehrt wird.

a) Codieren Sie die Ein- und Ausgaben mittels binärer Variablen.

(1 Punkt)

Eingabe	i_1	i_0
<i>cLinks</i>		
<i>cRechts</i>		
<i>keinCrash</i>		
<i>disabled</i>		

Eingabe des Automaten

Ausgabe	o_2	o_1	o_0
<i>NoCrash</i>			
<i>LAirbag</i>			
<i>RAirbag</i>			
<i>MAirbag</i>			
<i>NoAction</i>			

Ausgabe des Automaten

b) Entwickeln Sie die beschriebene Airbagansteuerung als Medwedev-Automat und geben Sie den Automatengraphen an.

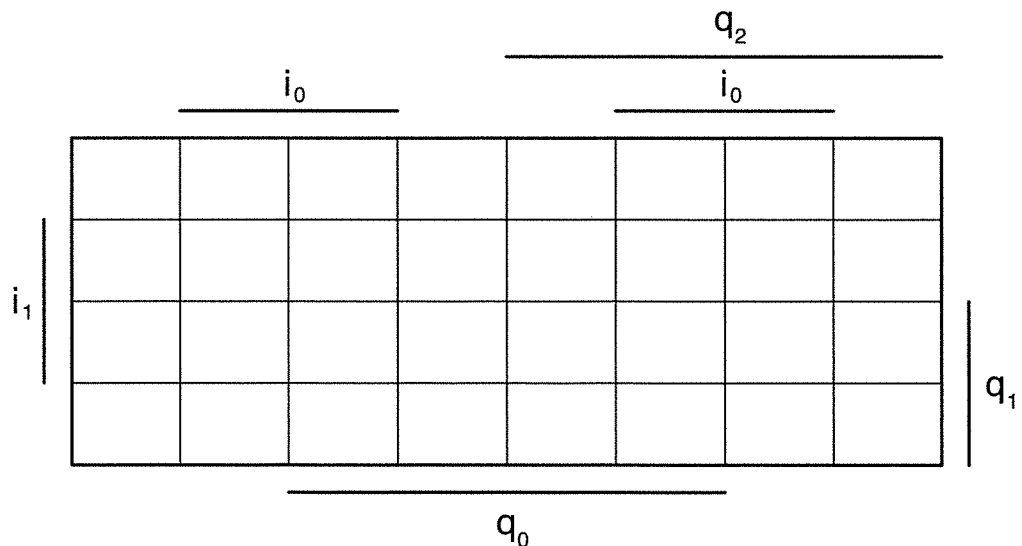
(4 Punkte)

c) Vervollständigen Sie die nachfolgend gegebene Automatentafel der Airbagsteuerung unter Verwendung von taktflankengesteuerten T/D/JK-Flipflops. (6 Punkte)

Zustandsname	Aktueller Zustand			Eingabe		Nachfolgezustand			Ansteuerfunktion				Ausgabe		
	q_2	q_1	q_0	i_1	i_0	q'_2	q'_1	q'_0	T_2	D_1	J_0	K_0	o_2	o_1	o_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) Entwickeln Sie eine disjunktive Minimalform (DMF) der Ansteuerfunktion des D-FlipFlops D_1 unter Verwendung des gegebenen Symmetriediagramms. Geben Sie den resultierenden Schaltalgebraischen Ausdruck an. (3 Punkte)

Achten Sie auf Don't-Cares sowie die vorgegebene Variablenordnung!



- e) Zeichnen Sie das vollständige Schaltwerk des in Teilaufgabe c) spezifizierten Medwedev-Automaten unter Zuhilfenahme der Ergebnisse aus Teilaufgabe d). Zur Vereinfachung der Aufgabe sind die Schaltnetze für die Signale J_0 , K_0 und T_2 nicht zu zeichnen, sondern stehen als bereits existierende Signale bereit. (4 Punkte)

- f) Geben Sie die Ausgabefunktionen an, wenn der oben erstellte Medwedev-Automat unverändert als Moore-Automat realisiert werden soll. Begründen Sie Ihre Antwort! (2 Punkte)

Aufgabe 4 (Arithmetik und Codierung)

(15 Punkte)

- a) Gegeben sind die beiden im Zweierkomplement dargestellten Binärzahlen $a = 01000111100_2$ und $b = 011011_2$. Wieviele Divisionsschritte müssen bei der Division der Zahl a durch die Zahl b anhand des Restoring-Divisionsverfahrens mindestens durchgeführt werden, um den Quotienten Q und den Rest R zu berechnen? Demonstrieren Sie das Restoring-Divisionsverfahren, indem Sie die Zahl a durch die Zahl b teilen und die einzelnen Schritte sowie R und Q explizit angeben. (7 Punkte)

- b) Ein vereinfachtes System zur Erkennung von Fahrspurübertretungen in einem Automobil kodiert das Überfahren der linken Spurmarkierung durch den Wert 100 und das Überfahren der rechten Spurmarkierung durch den Wert 001. Befindet sich das Fahrzeug hingegen innerhalb der Spurmarkierung, wird der Wert 010 verwendet. Welcher Hammingabstand wird durch diese Kodierung erreicht? Geben Sie zusätzlich an, wieviele Fehler erkannt und korrigiert werden können. (2 Punkte)

- c) Die Kodierung soll nun durch Verwendung eines Hammingcodes so erweitert werden, dass ein einfacher Fehler noch korrigiert werden kann. Wieviele zusätzliche Prüfstellen sind hierfür mindestens erforderlich? Geben Sie ein Schaltnetz zur Berechnung der Prüfstellen an und erweitern Sie dieses zur Korrektur einer beliebigen Informationsstelle anhand der Prüfstellen. Gehen Sie zur Vereinfachung davon aus, dass höchstens eine Binärstelle pro übertragenem Datenwort von einem Fehler betroffen sein kann. (6 Punkte)

Aufgabe 5 (VHDL)

(10 Punkte)

Implementieren Sie anhand des in Abbildung 1 dargestellten Moore-Automaten einen einfachen, synchron getakteten Schaltungsblock *CNTR_GumBallMachine* zur Steuerung eines Kaugummiautomaten. Ein Kaugummi kostet 20 Eurocent und der Automat akzeptiert ausschließlich 10- und 20-Eurocentmünzen. Wird eine 10- oder eine 20-Eurocentmünze eingeworfen, so ist das Signal *in_10* bzw. das Signal *in_20* für einen Takt auf '1' gesetzt. Alle Ein- und Ausgangssignale nehmen nur die Binärwerte '0' und '1' an. Der Automat befindet sich nach dem Einschalten im Initialzustand *S0*. Für die Notation eines Zustands gilt folgendes Format: Zustand/Ausgangssignal. Bei allen nicht dargestellten Kombinationen von Eingangssignalen sollen keine Zustandsänderungen erfolgen.

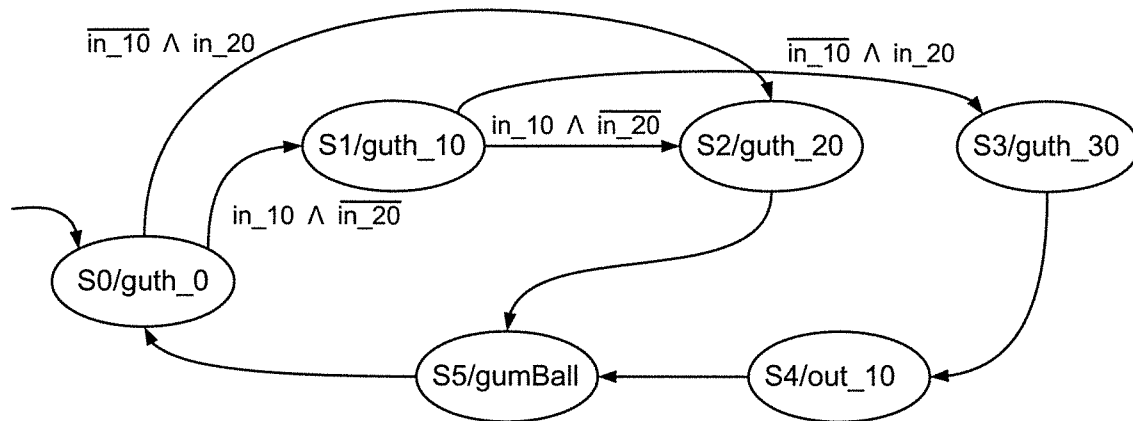


Abbildung 1: Dargestellt ist das Zustandsdiagramm des Moore-Automaten. Es sind nur diejenigen Eingangssignalkombinationen annotiert, die einen Zustandswechsel bewirken. Die Ausgangssignale sind nur annotiert, wenn sie den Wert '1' aufweisen.

- a) Geben Sie eine Schnittstellenbeschreibung des beschriebenen Automaten in Form einer Entity in VHDL an. Verwenden Sie hierzu das vorgegebene Code-Skelett. (Auf die Angabe der benötigten Bibliotheken der IEEE kann hierbei verzichtet werden.) (3 Punkte)

```
entity CNTR_GumBallMachine is
```

```
end entity;
```

- b) Geben Sie eine Implementierung des Schaltungsblockes *CNTR_GumBallMachine* in Form einer VHDL Architecture-Beschreibung an. Verwenden Sie hierzu das vorgegebene Code-Skelett. (Auf die Angabe der benötigten Bibliotheken der IEEE kann hierbei verzichtet werden.) (7 Punkte)

```
architecture behave of CNTR_GumBallMachine is
```

```
end architecture;
```