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

Klausur
Grundlagen der Technischen Informatik

27. Juli 2009

Name	
Matrikelnummer	
Studienrichtung	

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

Aufgabe 1 (Zahlensysteme) (20 Punkte)

a) Welches ist die größte mit n Bits darstellbare Dezimalzahl? (1 Punkt)

b) Welches ist die größte mit n Nibblen (Bezeichnung für Ziffern im Hexadezimalsystem) darstellbare Hexadezimalzahl? (1 Punkt)

c) Wie lautet die Zahl $(4096)_{10}$ in vorzeichenloser Binärdarstellung? (1 Punkt)

d) Wieviel Bits weniger werden für die Binärzahl aus Aufgabe c) gebraucht gegenüber einer Codierung der Zahl im BCD-Format? (1 Punkt)

e) Geben Sie die Anzahl der Bits an, welche für die Darstellung der Zahl aus Aufgabe c) im 2er-Komplement benötigt werden. (1 Punkt)

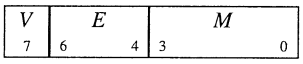
f) Geben Sie die Anzahl der Bits an, welche für die Darstellung der Dezimalzahl $(-4096)_{10}$ im 2er-Komplement benötigt werden. (1 Punkt)

g) Subtrahieren Sie $(4096)_{10}$ von $(39)_{10}$, d.h. $(39 - 4096)$. Die Berechnung und das Ergebnis sind in 2er-Komplement-Darstellung mit 16 Bits durchzuführen. (3 Punkte)

h) Addieren Sie die beiden BCD kodierten Zahlen $(0111\ 0011\ 1001)_{\text{BCD}}$ und $(0001\ 0110\ 1001)_{\text{BCD}}$. Die Addition ist im Binärformat durchzuführen. Berücksichtigen Sie dabei die eventuell entstehenden Pseudotetraden. (4 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 folgend aus:
Vorzeichen (1 Bit), Exponent (3 Bit), Mantisse (4 Bit)



Führen Sie nun die Multiplikation der beiden in diesem Format dargestellten Gleitkommazahlen $1\ 011\ 1000$ und $0\ 010\ 1100$ aus! (4 Punkte)

j) Geben Sie alle Varianten an, Fehler erkennen und korrigieren zu können, für einen Code mit minimaler Hammingdistanz von 5. (3 Punkte)

Aufgabe 2 (Automaten und Flipflops) (20 Punkte)

Im Folgenden soll eine Steuerung für eine Stoppuhr (siehe Abbildung 1) mit drei Tasten (Start/Stop, Split und Clear) entworfen werden. Diese besitzt die Eingänge g (Start/Stop) und z (Split), sowie die Ausgänge e, w und s. Falls $e = 0$, ist der Zähler nicht aktiv, und auf T wird die gestoppte Zeit ausgegeben. Falls $e = 1$, ist der Zähler aktiv, und auf T wird die aktuelle Zeit ausgegeben. Das Register für die Zwischenzeit übernimmt die an D anliegenden Daten genau dann, wenn $w = 1$. Die gespeicherte Zwischenzeit wird auf Q ausgegeben. Mit Hilfe des Multiplexers (Steuersignal s) kann die Zählerausgabe bzw. die Registerausgabe ausgewählt werden. Nehmen Sie an, dass sämtliche Flipflops der Stoppuhr einen asynchronen Reset-Eingang besitzen, der mit der Clear-Taste verbunden ist, so dass das Drücken dieser Taste im Folgenden nicht weiter betrachtet werden muss.

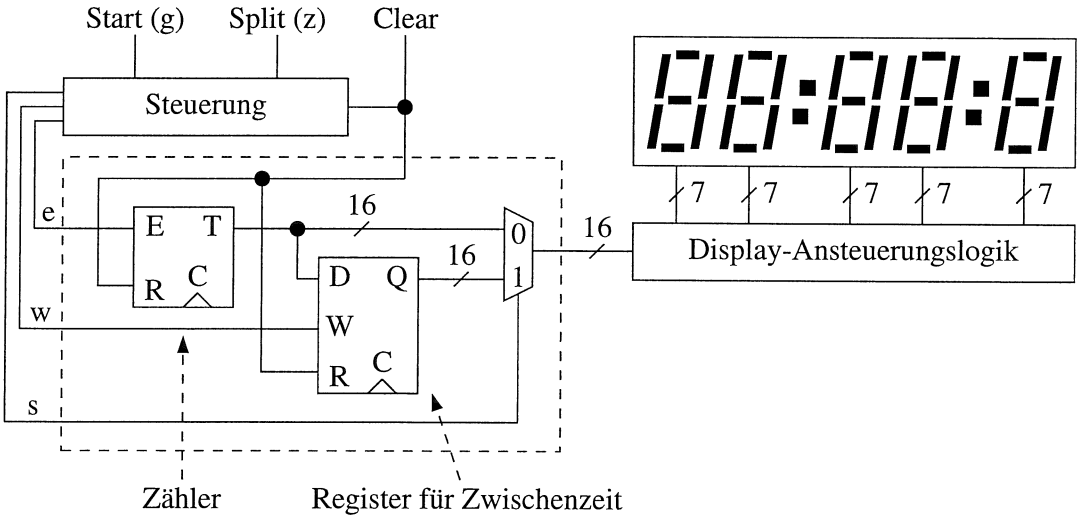


Abbildung 1: Stoppuhr

a) Ordnen Sie den folgenden Ausgabefunktionen jeweils den entsprechenden Automatentyp (Mealy, Moore, Medwedew) zu. Dabei ist $E(t)$ die aktuelle Eingabe, $Z(t)$ der aktuelle Zustand, und $A(t)$ die zu berechnende Ausgabe: (1 Punkt)

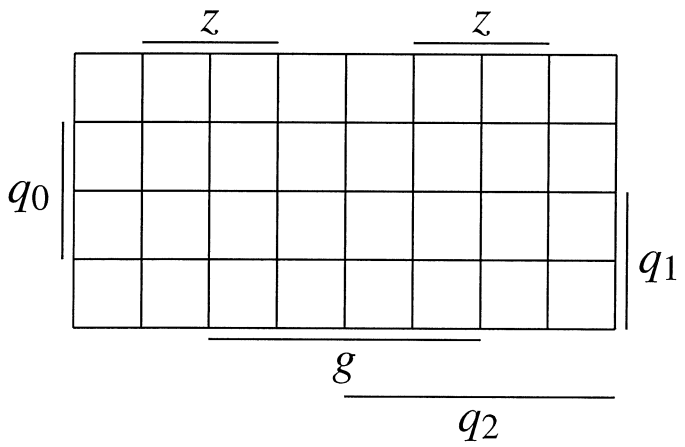
   - $A(t) = f(Z(t))$:
   - $A(t) = Z(t)$:
   - $A(t) = f(Z(t), E(t))$:

b) Im Folgenden soll die Steuerung durch einen Mealy-Automaten beschrieben werden. Dabei soll sich die Stoppuhr wie folgt verhalten: Ist die Uhr gestoppt, führt ein Druck auf die Start-Taste ($g = 1$) dazu, dass die Uhr von der zuletzt gestoppten Zeit an weiterläuft. Läuft die Uhr, führt ein erneuter Druck auf die Start-Taste dazu, dass die Uhr wieder anhält. Während die Uhr läuft, kann mit einem Druck auf die Split-Taste ($z = 1$) eine Zwischenzeit genommen werden, wobei eine bereits gespeicherte Zwischenzeit überschrieben werden soll. Wird die Split-Taste gedrückt während die Uhr gestoppt ist, soll abwechselnd die gestoppte Zeit und die Zwischenzeit ausgegeben werden (Wurde keine Zwischenzeit genommen während die Uhr lief, soll ein Druck auf die Split-Taste keine Auswirkungen auf die angezeigte Zeit haben, d.h. es wird in diesem Fall immer die gestoppte Zeit ausgegeben). Läuft die Uhr, soll immer die aktuelle Zeit ausgegeben werden. Falls $g = 1$, soll z ignoriert werden. Nehmen Sie weiterhin an, dass jeder Tastendruck nur einen Takt lang anliegt. Erstellen Sie den Automatengraphen für den spezifizierten Automaten in Form eines Mealy-Automaten mit maximal 5 Zuständen. (6 Punkte)

c) Im Folgenden soll die Zustandsübergangstabelle inklusive der drei zur Realisierung der Ansteuerfunktion verwendeten, taktflankengesteuerten T-FlipFlops für den von Ihnen entwickelten Automaten aufgestellt werden. Bestimmen Sie eine Kodierung der Zustände aus b), so dass der Startzustand durch "000" repräsentiert wird. Vervollständigen Sie die nachfolgende Zustandsübergangstabelle inklusive der Ansteuerfunktion. (5 Punkte)

Aktueller Zustand			Eingabe		Ausgabe			Nachfolgezustand			Ansteuerfunktion		
q_2	q_1	q_0	g	z	e	w	s	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									

d) Entwickeln Sie eine disjunktive Minimalform (DMF) der Ansteuerfunktion des T-FlipFlops T_0 unter Verwendung des gegebenen Symmetriediagrammes und tragen Sie ihre Lösung hier ein: (Achten Sie auf Don't-Cares sowie die vorgegebene Variablenordnung!) (4 Punkte)



e) Erstellen Sie eine minimierte Implementierung der Zustandsübergangsfunktion für die Zustandsvariable q_0 unter Verwendung der in Teilaufgabe d) ermittelten DMF. Verwenden Sie dabei ausschließlich AND- und OR-Gatter, Inverter sowie T-FlipFlops. (4 Punkte)

Aufgabe 3 (Minimierung von Schaltfunktionen, NAND-Technik) (20 Punkte)

a) Bestimmen Sie unter ausschließlicher Verwendung des Quine/McCluskey-Verfahrens alle Primimplikanten für die in der nebenstehenden Tabelle angegebene Schaltfunktion. (10 Punkte)

b) Bestimmen Sie alle kostenminimalen Lösungen (bezüglich der Anzahl der Literale zuzüglich der Anzahl der Terme) des Überdeckungsproblems aus der Teilaufgabe a) mit Hilfe des Petrick-Verfahrens. (5 Punkte)

c) Implementieren Sie die in Teilaufgabe b) erhaltene minimierte Schaltfunktion $f(A,B,C,D)$ als Schaltnetz unter ausschließlicher Verwendung von NAND-Gattern mit zwei Eingängen. (5 Punkte)

(dez.)	A	B	C	D	$f(A,B,C,D)$
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	-
10	1	0	1	0	-
11	1	0	1	1	-
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	0

Aufgabe 4 (VHDL) (10 Punkte)

In dieser Aufgabe soll eine ALU (*arithmetic logic unit*) mit 16-bit breiten Operanden für einen kleinen Mikroprozessors in VHDL beschrieben werden.

Auf diesem Mikroprozessor sollen folgende arithmetisch-logische Instruktionen ausführbar sein:

- ADD: Addition zweier vorzeichenloser Operanden
- SUB: Subtraktion zweier vorzeichenloser Operanden
- MUL: Multiplikation zweier vorzeichenloser Operanden
- AND: bitweise Verundung zweier Operanden,
- OR: bitweise Veroderung zweier Operanden,
- XOR: bitweises XOR zweier Operanden

a) Geben Sie die Schnittstellenbeschreibung der ALU in Form einer Entity in VHDL an. Überlegen Sie zuerst welche Eingabesignale und Ausgabesignale gebraucht werden.

Hinweis: Alle Ein- bzw. Ausgabesignale sollen vom Typ *std_logic_vector* sein. (3 Punkte)

b) Geben Sie eine Implementierung der ALU in Form einer VHDL Architecture-Beschreibung an. (*IEEE-Bibliotheken erlauben auch arithmetisch-logische Verknüpfungen auf Signalen vom Typ* std_logic_vector.) (5 Punkte)

c) Welche sprachlichen Konstrukte stehen in VHDL zur Beschreibung paralleler und sequenzieller Berechnungsabläufe zur Verfügung? (2 Punkte)

Aufgabe 5 (Rechnerarithmetik) (10 Punkte)

a) Geben Sie ein Schaltnetz für den in Abbildung 2 dargestellten 1-Bit Komparator an. (2 Punkt)

b) Geben Sie ein Schaltnetz für ein baumartiges Komparatornetzwerk, bestehend aus den 1-Bit Komparatoren aus Aufgabe a), für vorzeichenlose 4-Bit Zahlen an. (3 Punkte)

c) Geben Sie die Länge des kritischen Pfades für das Komparatornetzwerk aus Aufgabe b) an. Nehmen Sie dabei an, dass jedes Gatter außerhalb des 1-Bit Komparators als auch der 1-Bit Komparator selbst eine Verzögerungszeit von τ ns aufweist. (1 Punkte)

d) Werden für eine Division (a/b) bzw. Multiplikation $(a \cdot b)$ mit einer Zahl $b = 2^n, n \in \mathbb{N}$ immer ein allgemeiner Multiplizierer bzw. Dividierer benötigt oder reicht eine andere Schaltung mit geringerem Flächenverbrauch aus? Begründen Sie Ihre Antwort. (2 Punkte)

e) Geben Sie den Quotienten sowie den Rest als Binärzahl an, welche sich bei der Division der beiden im Binärsystem gegebenen Zahlen $(101\ 1101)_2 : (1000)_2$ ergeben. (2 Punkte)

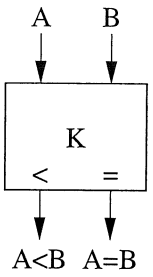


Abbildung 2: 1-Bit Komparator