

Klausur (25.02.2011) :
Technische Grundlagen der Informatik 1
Digitale Systeme
WS 2010/2011

Vorname	:	
Name	:	
Matrikelnummer	:	
Klausur-Code	:	001

Wichtige Hinweise:

- Mobiltelefone sind auszuschalten
- für die Lösung darf weder Bleistift noch Rotstift verwendet werden
- für die Lösungen sind die Aufgabenblätter zu verwenden
- der Lösungsweg muss nachvollziehbar sein und sich an der Aufgabenstellung orientieren
- Betrugsversuche werden mit einem Nichtbestehen der Klausur geahndet

Mit der Veröffentlichung meines Ergebnisses unter dem oben angegebenen Klausur-Code erkläre ich mich einverstanden (Bitte ankreuzen)

Hiermit erkläre ich, dass alle personenbezogenen Angaben richtig sind, ich die Hinweise gelesen und verstanden habe und ich mich gesundheitlich in der Lage fühle an der Klausur teilzunehmen.

Unterschrift: _____

Aufgabe	1	2	3	4	5	6	Σ
max. Punkte	12	10	13	7	7	11	60
erreichte Punkte							
Korrektor							

Aufgabe 1 (12 Punkte)

- (a) Nennen Sie ein Minimierungsverfahren, das auch *don't care* Fälle berücksichtigt!

- (b) Wenn man das *Shannon*-Verfahren gleichzeitig auf alle Variablen einer logischen Funktion anwendet (vollständig entwickelt), liegt diese in einer bestimmten Form vor. Wie wird diese genannt?

- (c) Nennen Sie die Sprachkonstrukte von VHDL, welche die Schnittstelle und Funktionalität kennzeichnen!

- (d) Wie werden die Bits für die Bereichsüberschreitungen bei arithmetischen Operation auf Dualzahlen genannt. Worin unterscheiden sich diese?

- (e) Nennen Sie drei Gesetze der Booleschen Algebra!

- (f) Nennen Sie ein Verfahren zur Ermittlung von Primimplikanten!

- (g) Geben Sie ein Verfahren zur Minimierung boolescher Funktionen an!

- (h) Benennen Sie die beiden aus der Vorlesung bekannten Automatentypen! Worin unterscheiden sich diese?

- (i) Was ist der Unterschied zwischen einem getakteten D-Flipflop und einem getaktetem Master-Slave-Flipflop?

- (j) Benennen Sie zwei Verzögerungsmodelle mit denen man in VHDL das Verhalten von Signalen modellieren kann!

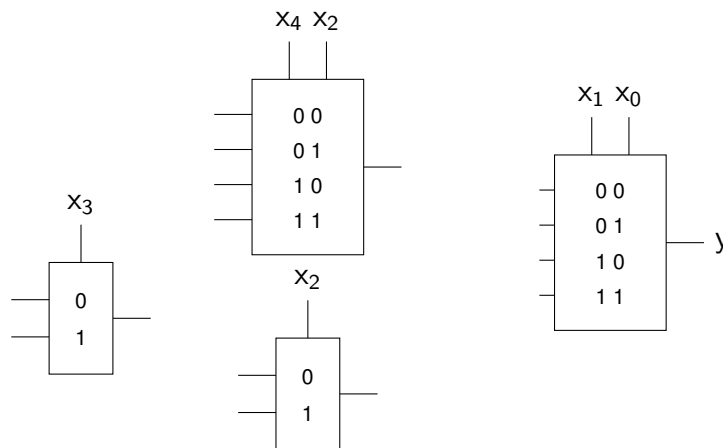
Aufgabe 2 (10 Punkte)

Gegeben ist die folgende Funktion

$$y = x_0 \bar{x}_1 \bar{x}_2 x_3 x_4 + x_0 x_1 \bar{x}_2 x_3 + x_0 \bar{x}_1 \bar{x}_2 \bar{x}_4 + \bar{x}_0 x_1 + x_0 x_1 x_2 \bar{x}_4 + x_0 \bar{x}_1 x_2 \bar{x}_4$$

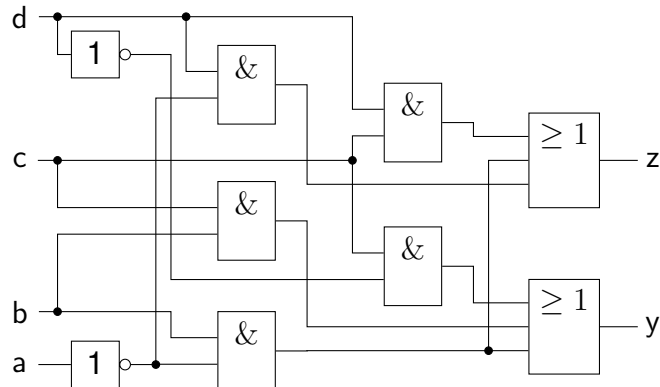
1. Formen Sie die gegebene Funktion mit Hilfe von Shannon so um, dass sie auf die vorgegebene Multiplexerstruktur abgebildet werden kann.
2. Bilden Sie die umgeformte Funktion unter Verwendung aller Multiplexer auf die Multiplexerstruktur ab.

Hinweis: Verwenden Sie dabei die Multiplexer so häufig wie möglich.



Aufgabe 3 (13 Punkte)

Gegeben ist das folgende Schaltnetz:



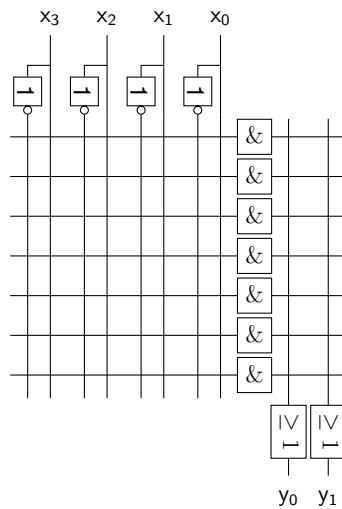
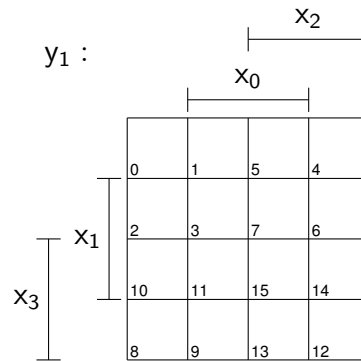
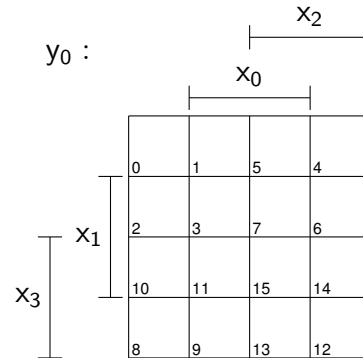
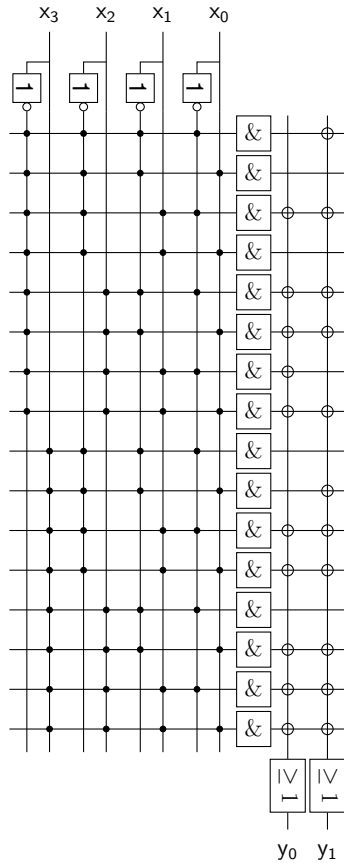
(a) Lesen Sie die Funktionen y und z aus!

(b) Geben Sie für die zuvor ermittelten Funktionen eine VHDL-Entity-Spezifikation `dummy` an und realisieren Sie anschließend eine Architektur `behaviour`. Benutzen Sie den Signaltypen `bit`. Verwenden Sie interne Signale um mehrfach genutzte Teilausdrücke separat zuzuweisen.

Aufgabe 4 (7 Punkte)

Übertragen Sie die Funktionen y_0, y_1 aus dem unten abgebildeten PROM in die angegebenen KV-Diagramme. Minimieren Sie die beiden Funktionen und tragen Sie diese anschließend möglichst effizient in das gegebene PLA ein.

Hinweis: Beim ersten Schritt ist es sinnvoll sich vorher zu überlegen, welche strukturellen Eigenschaften von PROM und KV-Tafeln zusammenhängen.



Aufgabe 5 (7 Punkte)

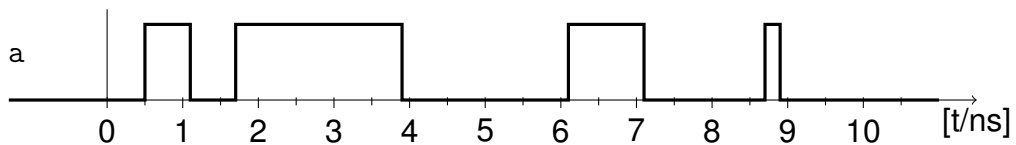
Das nachfolgende VHDL-Quellcodefragment enthält Signalzuweisungen:

```

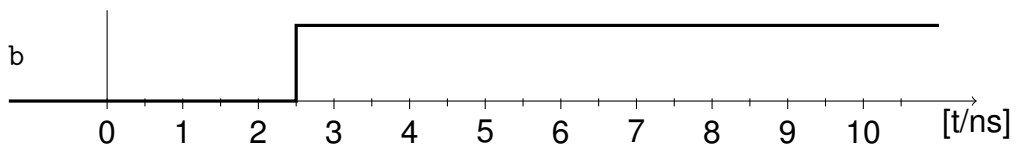
...
    signal a, b, o1, o2, o3 : bit;
begin
    o1 <= inertial a after 1.5 ns;
    o2 <= transport not (a and b) after 1.5 ns;
    o3 <= reject 0.5 ns inertial a and b after 1 ns;
...

```

Gegeben ist außerdem der Signalverlauf des Signals a:



sowie der des Signals b:



Ordnen Sie den folgenden gegebenen Signalverläufen die passenden Signale zu und tragen Sie jeweils den Namen (o1, o2, o3) in das entsprechende graue Kästchen ein. Kennzeichnen Sie die Signalverläufe, die nicht das Ergebnis einer der aufgeführten Signalzuweisungen sein können, durch ein x im grauen Kästchen:

<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	

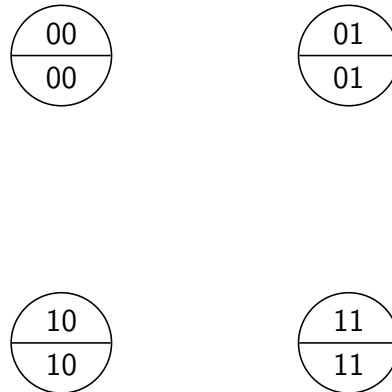
Aufgabe 6 (11 Punkte)

Es ist ein zweistelliger Gray-Code-Zähler als Moore-Automat zu entwerfen, dessen Zählrichtung durch die Steuerleitung x gesteuert werden kann:

- für $x = 1$ wird die Zählfolge 00, 01, 11 und 10 zyklisch durchlaufen (vorwärts zählen),
- für $x = 0$ wird die Zählfolge 00, 10, 11 und 01 zyklisch durchlaufen (rückwärts zählen).

Die Umschaltung der Zählrichtung ist in jedem Zustand erlaubt. Die vier Zustände des Schaltwerkes werden durch zwei Master-Slave-Flipflops q_0^t und q_1^t kodiert, wobei die Zustandskodierung mit dem Zählzustand identisch ist. Die Ausgänge des Automaten entsprechen also direkt den Zustandsvariablen.

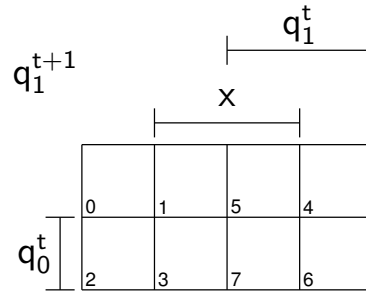
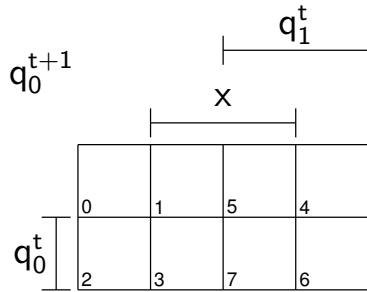
(a) Vervollständigen Sie das Zustandsdiagramm!



(b) Stellen Sie die Zustandsübergangstabelle des Automaten auf!

$\delta(q_1^t, q_0^t, x)$	q_1^t	q_0^t	x	q_1^{t+1}	q_0^{t+1}
0					
1					
2					
3					
4					
5					
6					
7					

- (c) Bestimmen Sie die minimalen Übergangsfunktionen mit Hilfe der KV-Diagramme und notieren Sie diese!



- (d) Vervollständigen Sie den Schaltplan des Automaten! Benutzen Sie ausschließlich Elementargatter!

