

Klausur zur Vorlesung

"Grundlagen der technischen Informatik" und "Grundlagen der Rechnerarchitektur" Sommersemester 2002

1. Teil: GTI

Der erste Teil (GTI) der Klausur umfasst 6 Aufgaben und hat 8 Seiten, die in maximal 1 Stunde zu bearbeiten sind. **Es sind keine Hilfsmittel zugelassen.** Schreiben Sie Ihre Lösungen nur auf die dafür vorgesehenen Blätter! Verwenden Sie **kein eigenes Konzeptpapier**; notfalls erhalten Sie welches bei der Aufsicht.

Lassen Sie die Aufgaben und ihre Lösungen zusammengeheftet!

Schreiben Sie auf jedes Blatt (auch auf das Konzeptpapier) in Blockschrift Ihren **Namen** und Ihre **Matrikelnummer**.

Bei mehreren präsentierten Lösungen wird die Aufgabe nicht gewertet! Streichen Sie daher bei Angabe mehrerer Lösungsansätze die nicht zu bewertenden Lösungen durch!

Abschreiben und abschreiben lassen oder Hilfe Dritter führt zum Nichtbestehen der Klausur.

Es sind in diesem ersten Teil insgesamt 60 Punkte erreichbar.

Viel Erfolg!

Vor- und Nachname: _____

Matrikelnummer: _____ Fachbereich: _____

Studienfach (ankreuzen):

Informatik	<input type="checkbox"/>
Ingenieur-Informatik (Elektrotechnik)	<input type="checkbox"/>
Ingenieur-Informatik (Maschinenbau)	<input type="checkbox"/>
Ingenieur-Informatik (Informatik)	<input type="checkbox"/>
Wirtschafts-Informatik	<input type="checkbox"/>
Informatik als Nebenfach	<input type="checkbox"/>
Lehramt Informatik	<input type="checkbox"/>

Aufgabe	1	2	3	4	5	6	Summe
Punkte	14	8	14	8	8	8	60
Erreicht							

Summe	GTI:		GRA:		Gesamt:		Note:	
-------	------	--	------	--	---------	--	-------	--

Name:

Matrikel-Nummer:

Aufgabe 1 (14 min, 3 + 7 + 2 + 2 Punkte): Karnaugh-Diagramm

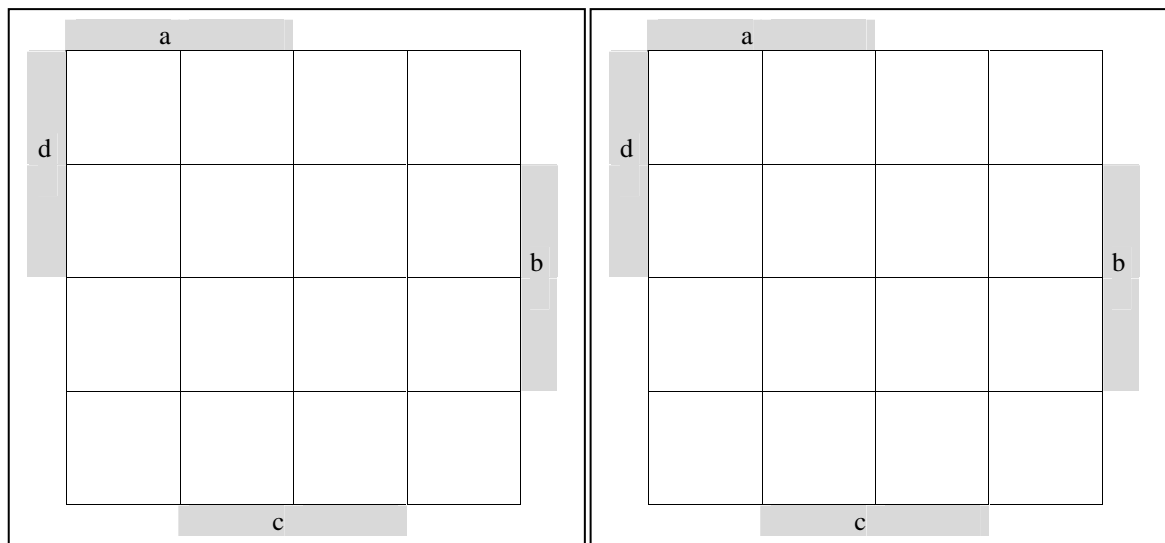
Gegeben sei die boolesche Funktion

$$F = a \bar{c} d + b c d + \bar{a} \bar{c} d + b c \bar{d} + \bar{a} \bar{b} \bar{d} + \bar{a} b \bar{c} \bar{d}$$

- (a) Bestimmen Sie eine minimale Darstellung von F mit Hilfe des Karnaugh-Diagramms: Füllen Sie dazu das Diagramm zunächst entsprechend mit Einsen aus. Markieren Sie dann alle gefundenen **Primimplikanten** und nummerieren Sie diese durch.

Auszufüllendes Karnaugh-Diagramm:

Ersatzdiagramm:



- (b) Bestimmen Sie die **minimalen** Überdeckungsindexmengen $M\ddot{U}(P_i, x)$, für alle Primimplikanten. (Lassen Sie die Menge leer, falls P_i nicht existiert.)

$$M\ddot{U}(P_1, x) = \{ \quad \quad \quad \}$$

$$M\ddot{U}(P_2, x) = \{ \quad \quad \quad \}$$

$$M\ddot{U}(P_3, x) = \{ \quad \quad \quad \}$$

$$M\ddot{U}(P_4, x) = \{ \quad \quad \quad \}$$

$$M\ddot{U}(P_5, x) = \{ \quad \quad \quad \}$$

$$M\ddot{U}(P_6, x) = \{ \quad \quad \quad \}$$

- (c) Geben Sie die sich aus (b) ergebende Überdeckungsfunktion $\ddot{U}F$ **ohne** weitere Vereinfachungen an.

$$\ddot{U}F = \langle \quad \quad \quad \rangle$$

- (d) Geben Sie eine minimale Darstellung als minimale boolesche Funktion hier an:

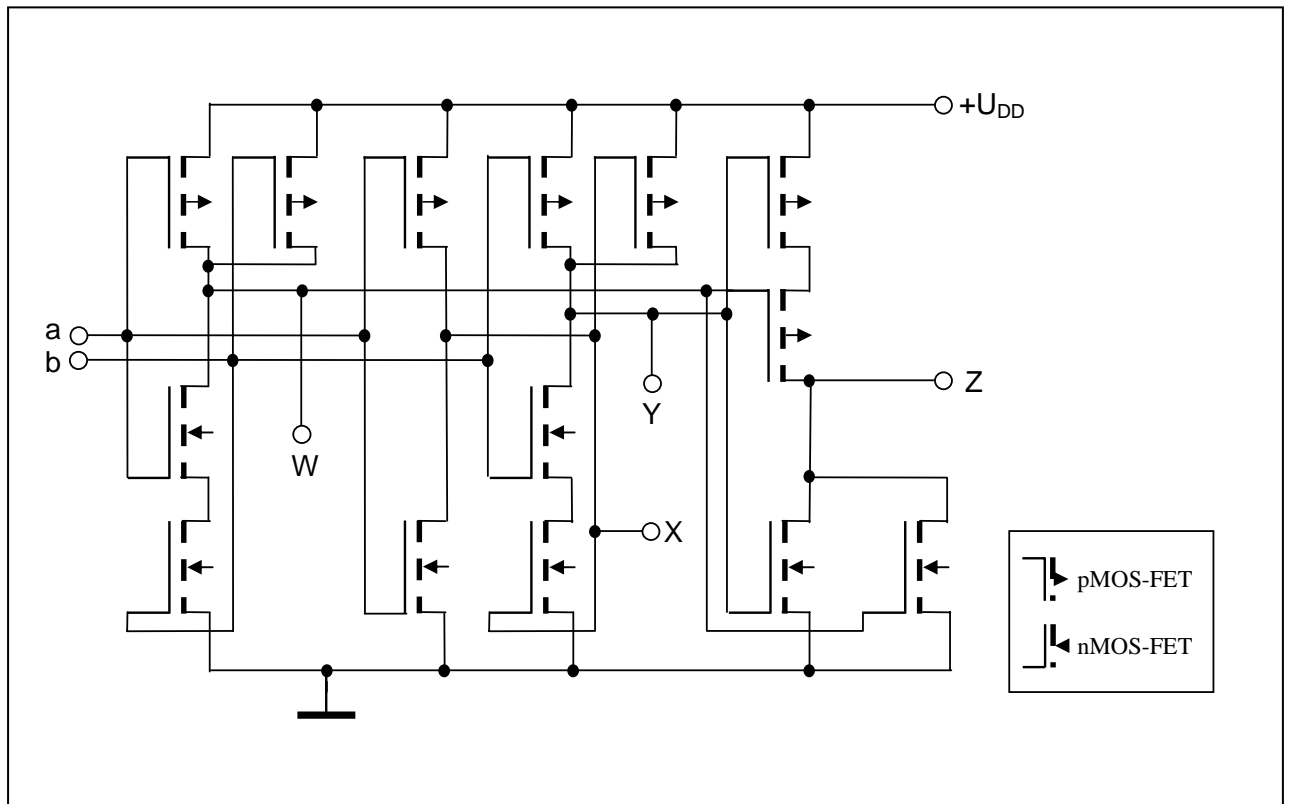
$F_{\min} =$

Name:

Matrikel-Nummer:

Aufgabe 2 (8 min, 8 Punkte): MOS-Technologie

Gegeben ist die folgende Schaltung mit nMOS- und pMOS-Feldeffekttransistoren.



a) Füllen Sie die folgende Wertetabelle aus:

a	b	W	X	Y	Z
Funktion: (nur 2 Variablen) ¹					
0	0				
0	1				
1	0				
1	1				

¹ Benutzen Sie ggf. auch W, X, Y, Z.

Name:

Matrikel-Nummer:

Aufgabe 3 (14 min, 4 + 10 Punkte): Automaten und VHDL

Gegeben ist die untenstehende VHDL-Beschreibung für einen Mealy-Automaten.

- (a) Geben Sie in dem dafür vorgesehenen Kasten den Zustandsübergangsgraphen an.
- (b) Füllen Sie für die gegebene Eingangsfolge den *Trace* des Automaten in der Tabelle auf der folgenden Seite aus.

```
Entity AUTOMAT is
  Port ( CLK : In std_logic;
        RES : In std_logic;
        X : In std_logic;
        Y : Out std_logic );
end AUTOMAT;

architecture BEHAVIORAL of AUTOMAT is

  type fsm_state is (C1, C2, C3, C4);
  signal CS, NS: fsm_state;

begin
  SEQ: process (CS, X)
  begin
    case CS is
      when C1 =>
        if X = '0' then
          Y <= "0" ;
          NS <= C3 ;
        else
          Y <= "1" ;
          NS <= C3 ;
        end if;
      when C2 =>
        if X = '0' then
          Y <= "0" ;
          NS <= C2 ;
        else
          Y <= "0" ;
          NS <= C1 ;
        end if;
      when C3 =>
        if X = '0' then
          Y <= "1" ;
          NS <= C1 ;
        else
          Y <= "0" ;
          NS <= C2 ;
        end if ;
    end case ;
  end process ;
```

```
MEM: process(CLK, RES)
begin
  if (RES = '1') then
    CS <= C1 ;
  else
    if (CLK'event and CLK = '1') then
      CS <= NS;
    end if;
  end if;
end process ;

end BEHAVIORAL ;
```

Lösung zu (a):

Name:

Matrikel-Nummer:

Fortsetzung zu Aufgabe 3:

Lösung zu Aufgabenteil (b):

Füllen Sie für die gegebene Eingangsfolge den **Trace** des Automaten in der Tabelle aus.

CLK	0	0	1	0	1	0	1	0	1	1
RES	1	0	0	0	0	0	0	0	0	0
X	0	1	1	0	0	0	0	1	1	0
Y										
NS	C1									
CS										

Ersatztable zu Aufgabenteil (b): *(Streichen Sie ggf. die obige Lsg. durch)*

CLK	0	0	1	0	1	0	1	0	1	1
RES	1	0	0	0	0	0	0	0	0	0
X	0	1	1	0	0	0	0	1	1	0
Y										
NS	C1									
CS										

Name:

Matrikel-Nummer:

Aufgabe 4 (8 min, 2 + 6 Punkte): Division

- (a) Rechnen Sie die Werte $X = 15$, $Y = 4$ und $-Y$ in das **2er-Komplement** unter Verwendung von 5 Bit inklusive Vorzeichen um. **Geben Sie jeweils die Werte aller 5 Bits an** (das höchstwertigste Bit steht links).

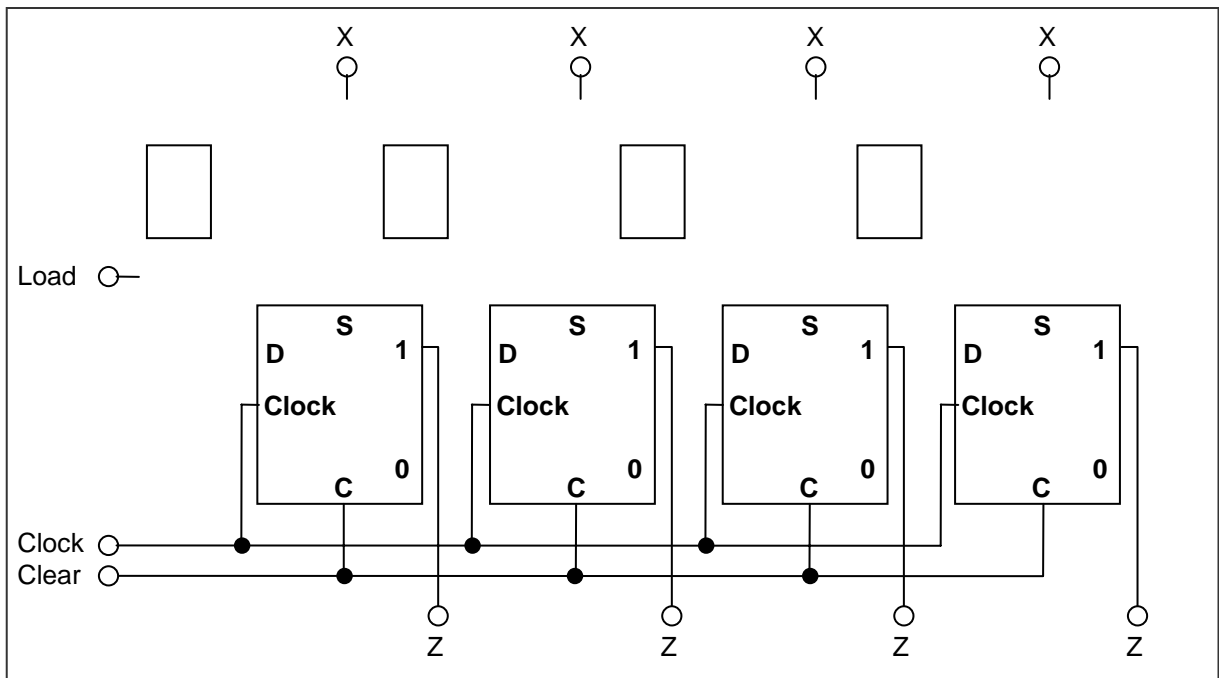
$X = 15_{10} =$, $Y = 4_{10} =$, $-Y = -4_{10} =$
--

- (b) Wenden Sie den aus der Vorlesung bekannten "Restoring Algorithmus" zur Festkomma-division auf die Werte $Q := X = 15$ und $Y = 4$ an. Das Register Q ist 4 Bit breit, AC besitzt ein 5tes Bit für das intern benötigte Vorzeichen. **Markieren Sie am Ende der Rechnung deutlich wo in den Registern der Quotient bzw. der Rest abzulesen ist!**

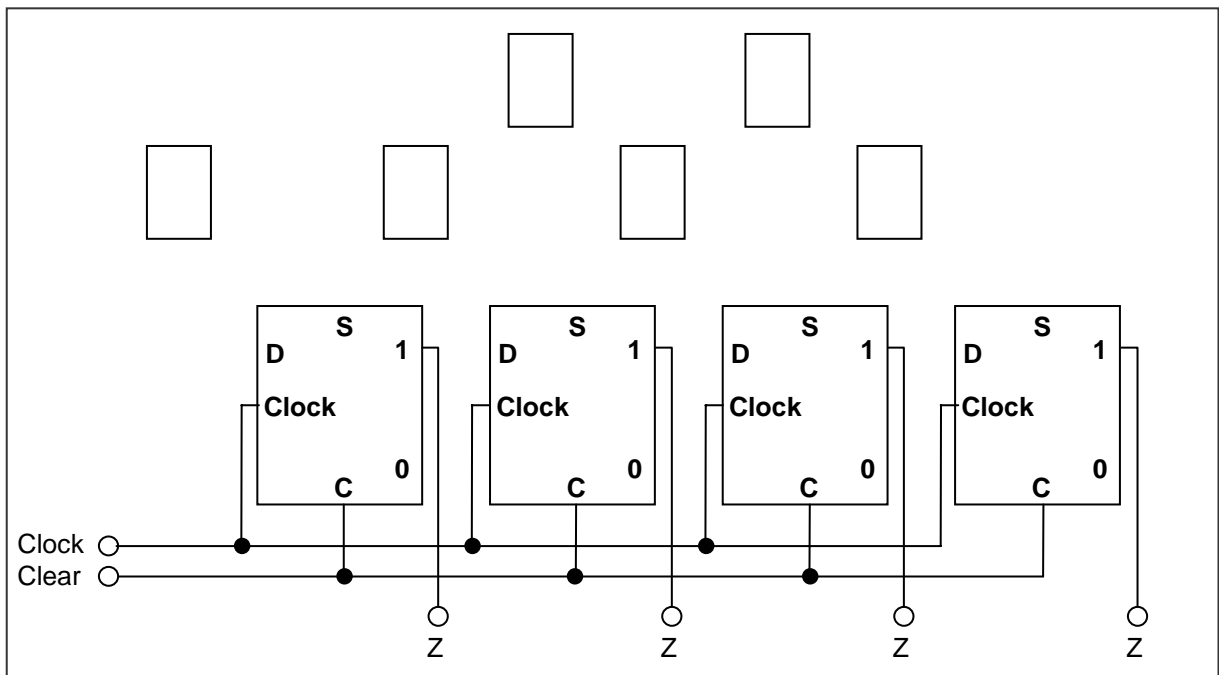
Zyklus	AC (5 Bit Darstellung)	Q (4 Bit Darstellung)	Nächste Operation (Kommentar)
1	00000	1111	
2			
3			
4			

Aufgabe 5 (8 min, 4 + 4 Punkte): Flip-Flops und Register

- (a) Vervollständigen Sie die nachfolgende Schaltung, so dass ein **4-Bit Parallel-Seriell-Wandler** entsteht, der die seriellen Daten am Ausgang Z_3 ausgibt.



- (b) Vervollständigen Sie die nachfolgende Schaltung, so dass ein synchroner **4-Bit Zähler modulo 16** entsteht.



Hinweis: Zeichnen Sie fehlende Signalleitungen (Verbindungen) in die Schaltung ein. Geben Sie außerdem an, welche Gatter-Funktionen die dargestellten Rechtecke darstellen. Füllen Sie dafür das Schaltsymbol entsprechend aus! **Die Gatter haben jeweils maximal 2 Eingänge!** (Sie können die Gatterfunktionen für AND, OR, NAND, NOR, XOR, XNOR und INVERTER benutzen.)

Name:

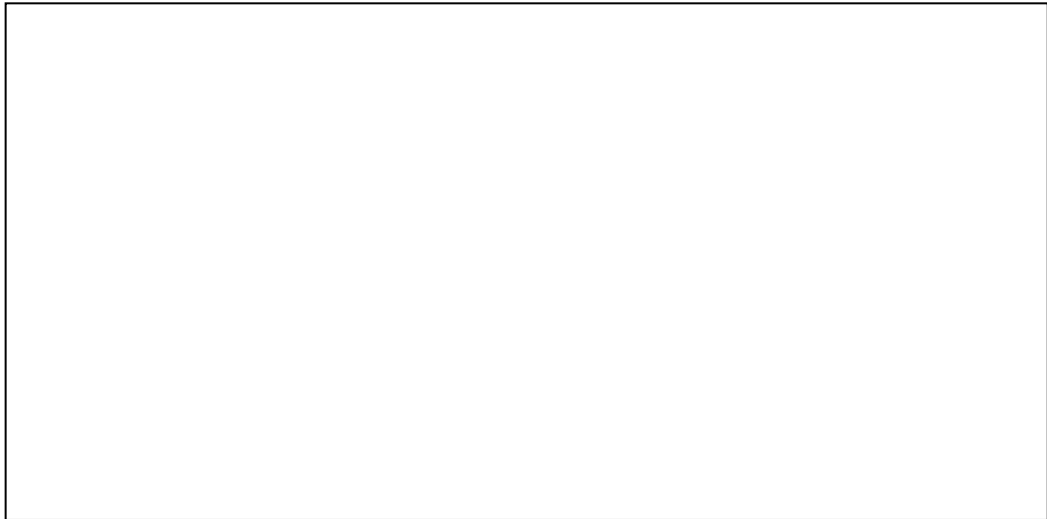
Matrikel-Nummer:

Aufgabe 6 (8 min, 4 + 4 Punkte): VHDL

Eine Steuerung eines Hochregalmotors soll durch den Baustein X_CONTROL realisiert werden. Das Signal X_ON darf nur auf '1' gesetzt werden, wenn die aktuelle X-Adresse X_IST(0:2) ungleich der anzufahrenden X-Adresse X_SOLL(0:2) ist und wenn der Eingabewert X_START gesetzt ist.

- a) Geben Sie die VHDL-Beschreibung der Schnittstelle dieses Steuerbausteins an. Benutzen Sie als Datentyp *std_logic* bzw. *std_logic_vector*.

entity
Port (



end

- b) Vervollständigen Sie die Verhaltensbeschreibung für den Baustein X_CONTROL.

architecture BEHAVIORAL of X_CONTROL is
begin
X_CONTROL: process(X_IST, X_SOLL, X_START)
begin



end process X_CONTROL;
end BEHAVIORAL;

configuration CFG_X_CONTROL_BEHAVIORAL of X_CONTROL is
for BEHAVIORAL
end for;
end CFG_X_CONTROL_BEHAVIORAL;