

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!

Musterlösung

Vor- und Nachname: _____

(ohne Gewähr)

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:	
-------	------	--	------	--	---------	--	-------	--

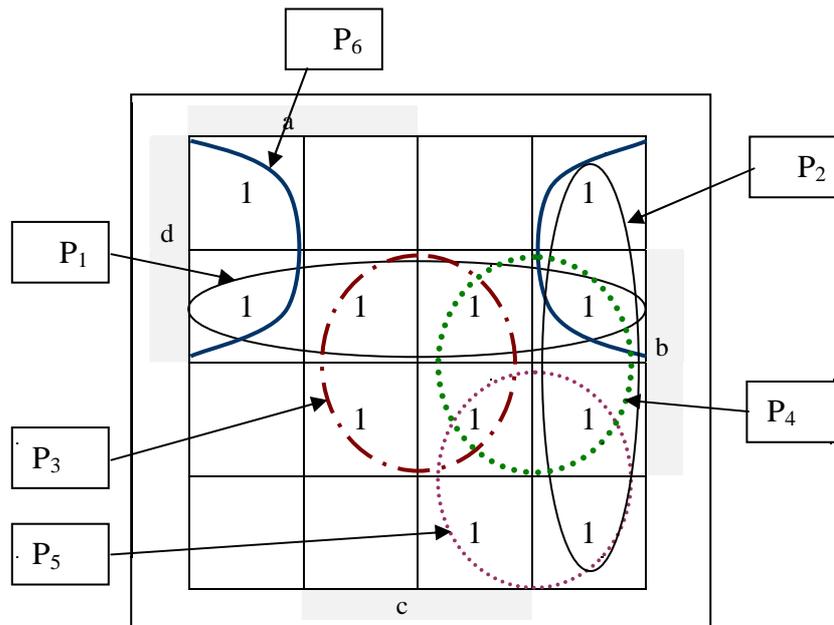
Aufgabe 1 (14 min, 3 + 7 + 2 + 2 Punkte): Karnaugh-Diagramm

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.

Lösung:



- (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.)

$$M\ddot{U}(P_1, x) = \{ \{1\}, \{3, 6\} \}$$

$$M\ddot{U}(P_2, x) = \{ \{2\}, \{5, 6\} \}$$

$$M\ddot{U}(P_3, x) = \{ \{3\} \}$$

$$M\ddot{U}(P_4, x) = \{ \{4\}, \{2, 3\}, \{1, 5\}, \{3, 5, 6\} \}$$

$$M\ddot{U}(P_5, x) = \{ \{5\} \}$$

$$M\ddot{U}(P_6, x) = \{ \{6\} \}$$

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

$$\ddot{U}F = \langle P_3 P_5 P_6 (P_1 + P_3 P_6) (P_2 + P_5 P_6) (P_4 + P_2 P_3 + P_1 P_5 + P_3 P_5 P_6) \rangle$$

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

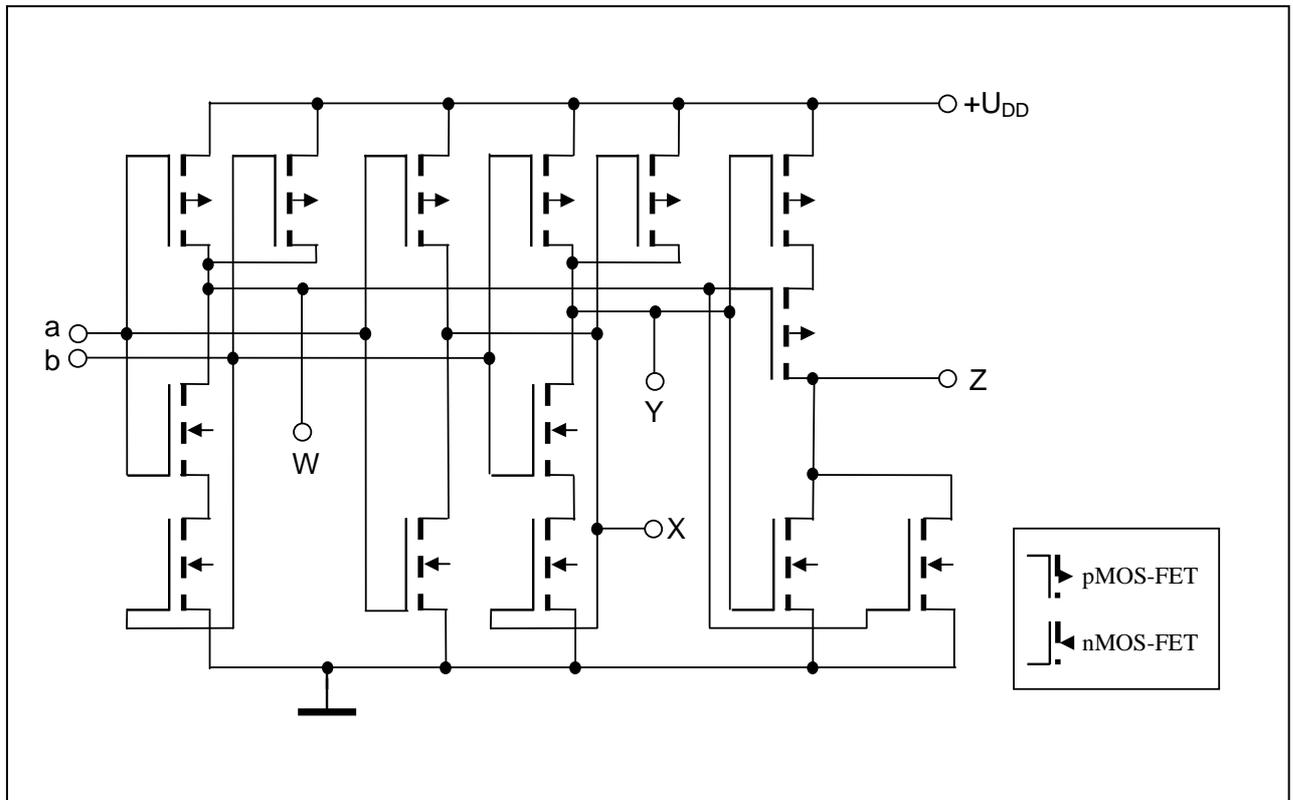
$$F_{\text{Min}} = P_3 + P_5 + P_6 = b c + \bar{a} \bar{d} + \bar{c} d$$

Name:

Matrikel-Nummer:

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

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



Füllen Sie die folgende Wertetabelle aus und geben Sie die Funktion an:

a	b	W	X	Y	Z
Funktion: (nur 2 Variablen) ¹		$\overline{a \cdot b}$	\overline{a}	$\overline{X \cdot b}$	$\overline{W + Y}$
0	0				
0	1				
1	0				
1	1				

¹ Benutzen Sie ggf. auch W, X, Y, Z.

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);
  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 ;
end

```

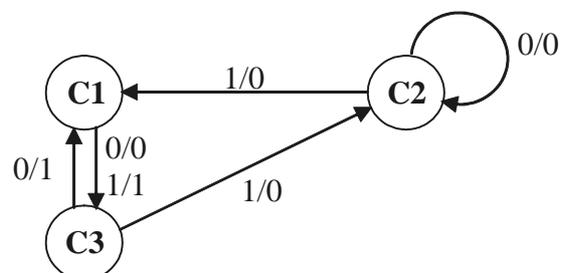
```

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	0	1	0	1	0	0	1	0	0	0
NS	C1	C3	C2	C1	C3	C3	C1	C2	C1	C2
CS	1	C1	C3	C3	C1	C1	C3	C3	C2	C2

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										
CS										

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

- (a) Rechnen Sie die Werte $X = 18$, $Y = 5$ und $-Y$ in das **2er-Komplement** unter Verwendung von 6 Bit inklusive Vorzeichen um. **Geben Sie jeweils die Werte aller 6 Bits an** (das höchstwertigste Bit steht links).

$$X = 18_{10} = 010010_2,$$

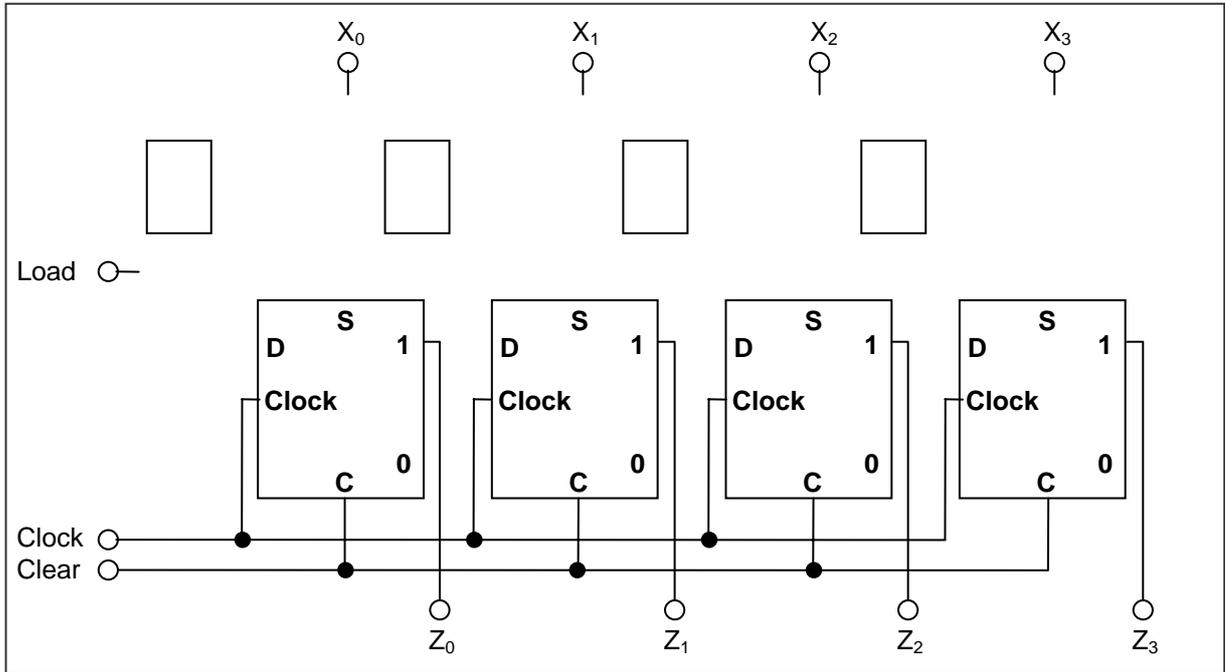
$$Y = 5_{10} = 000101_2, \quad -Y = -5_{10} = 111011_2$$

- (b) Wenden Sie den aus der Vorlesung bekannten "Restoring Algorithmus" zur Festkomma-division auf die Werte $Q := X = 18$ und $Y = 5$ an. Das Register Q ist 5 Bit breit, AC besitzt ein 6tes Bit für das intern benötigte Vorzeichen. **Markieren Sie am Ende der Rechnung deutlich wo in den Registern der Quotient bzw. der Rest abzulesen ist!**

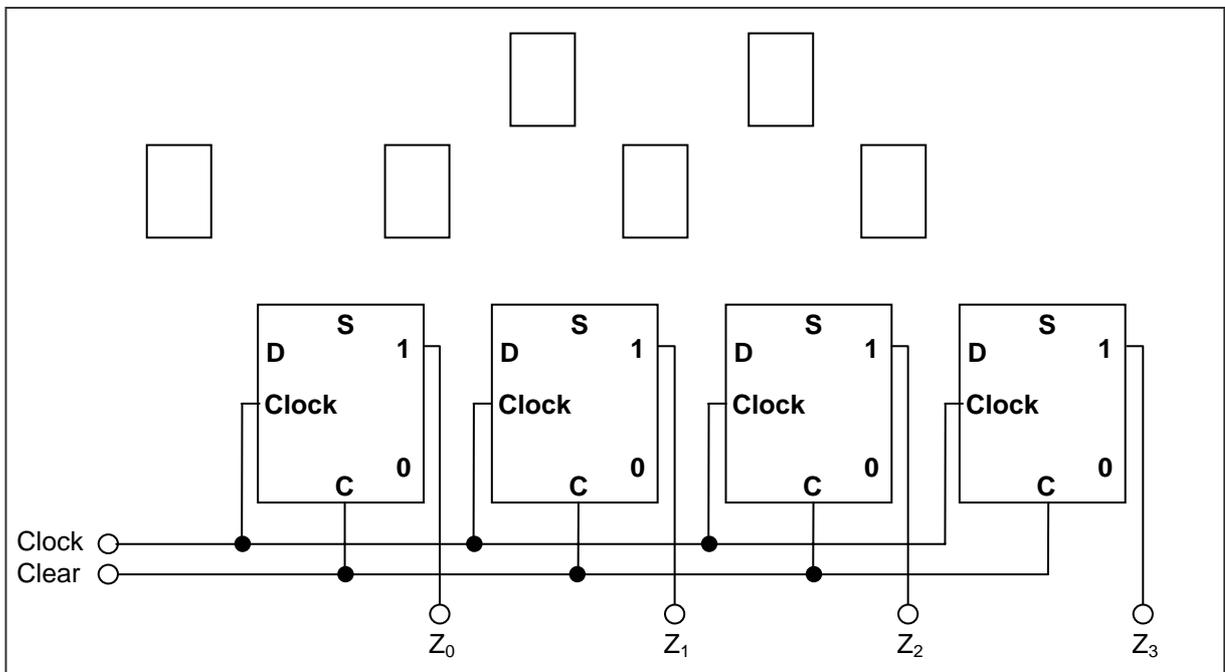
Zyklus	AC (6 Bit Darstellung)	Q (5 Bit Darstellung)	Nächste Operation (Kommentar)
1	000000 000001 +111011 111101 +000101 000001	10010 00100 00100	Shift AC <= AC - Y AC <= AC + Y Shift
2	000010 +111011 111101 +000101 000010	01000 01000	AC <= AC - Y AC <= AC + Y Shift
3	000100 +111011 111111 +000101 000100	10000 10000	AC <= AC - Y AC <= AC - Y Shift
4	001001 +111011 000100 000100	00000 00000 00001	AC <= AC - Y Q(0) <= 1 Shift
5	001000 +111011 000011 000011	00010 00011	AC <= AC - Y Q(0) <= 1
	Rest = 000011	00011 = Quotient	Fertig

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.)

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

Gegeben ist der folgende Ausschnitt aus einem VHDL Programm:

```
Entity INC is
  Port (x: In Integer; u,z: Out Integer);
End INC;
Architecture ARCH of INC is
  Variable y: Integer := 0;
  begin
  test: process(x)
  begin
    while ((x < 7 + y) or (y < 10 * x)) loop
      x <= x + 1;
      y := y + 2;
    end loop;
    z <= y + x;
    u <= y;
    y := 0;
  end process test;
end ARCH;
```

- (a) Nehmen Sie an, dass das Eingangssignal X zu einem beliebigen Zeitpunkt auf "1" gesetzt und das Programm aktiviert wird. Welchen Wert hat die Ausgangsvariable U und Z nach Beendigung des Programms?

u = Z =

- (b) Im gleichen Programm lautet der Process „test“ jetzt folgendermaßen:

```
test: process(x)
begin
  for y in 1 to 7 loop
    x <= x + 1 after 2ns;
  end loop;
  z <= y + x;
  u <= y;
end process test;
```

Welchen Wert hat U und Z zum Zeitpunkt $t + 1ns$ nachdem X zum Zeitpunkt t auf "1" gesetzt wurde?

u = Z =

- (c) Nach wie viel ns ist der **Prozess** „test“ aus Aufgabenteil (b) nach dem Zeitpunkt t (Aktivierung) beendet?

Ende = t + ns
