

Klausur zur Vorlesung

"Grundlagen der technischen Informatik" und "Grundlagen der Rechnerarchitektur" Wintersemester 1999/2000

1. Teil: GTI

Der erste Teil (GTI) der Klausur umfaßt 7 Aufgaben und hat 17 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 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	7	Summe
Punkte	6	10	14	11	6	6	7	60
Erreicht								

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

Name:

Matrikel-Nummer:

Diese Seite ist leer und steht als Konzeptpapier zur Verfügung.

Name:

Matrikel-Nummer:

Aufgabe 1 (6 min, 4 + 2 Punkte): Gleitkommadarstellung/Arithmetik

Das 32-Bit-IEEE-Format für Gleitkommazahlen benutzt ein Bit für das Vorzeichen, 8 Bit für die Exponenten-Darstellung (Exzess = 127_{10}) und 23 Bit für die Mantissen-Darstellung.

- a) Wandeln Sie die Zahlen 21.0_{10} und 0.1875_{10} in das 32-Bit IEEE-Format um. Stellen Sie die Zahlen normalisiert dar.

S	EXP	FRACTION	
1	10000001	01100000000000000000000	-5.5_{10}
0	10000011	01010000000000000000000	21_{10}
0	01111100	10000000000000000000000	0.1875_{10}

- b) Addieren Sie die folgenden Zahlen binär gemäß der Additionsvorschrift für Gleitkommazahlen im IEEE-Format. Stellen Sie das Ergebnis wieder normalisiert dar.

S	EXP	HB	FRACTION	
0	10000011	1	01010000000000000000000	21_{10}
0	10000011	0	00000011000000000000000	$+0.1875_{10}$
0	10000011	1	01010011000000000000000	21.1875_{10}
0	10000011	1	01010011000000000000000	<i>normalisiert</i>

Hinweis:

- S = Vorzeichen
- EXP = Exponent + Exzess
- HB = Hidden Bit (nicht Bestandteil der Darstellung)
- FRACTION = Mantisse ohne führende 1.

Normalisierte Darstellung: Wert = $(-1)^S \times 1.FRACTION \times 2^{EXP-Exzess}$

Aufgabe 2 (10 min, 1 + 9 Punkte): Steuerwerk mit PLA

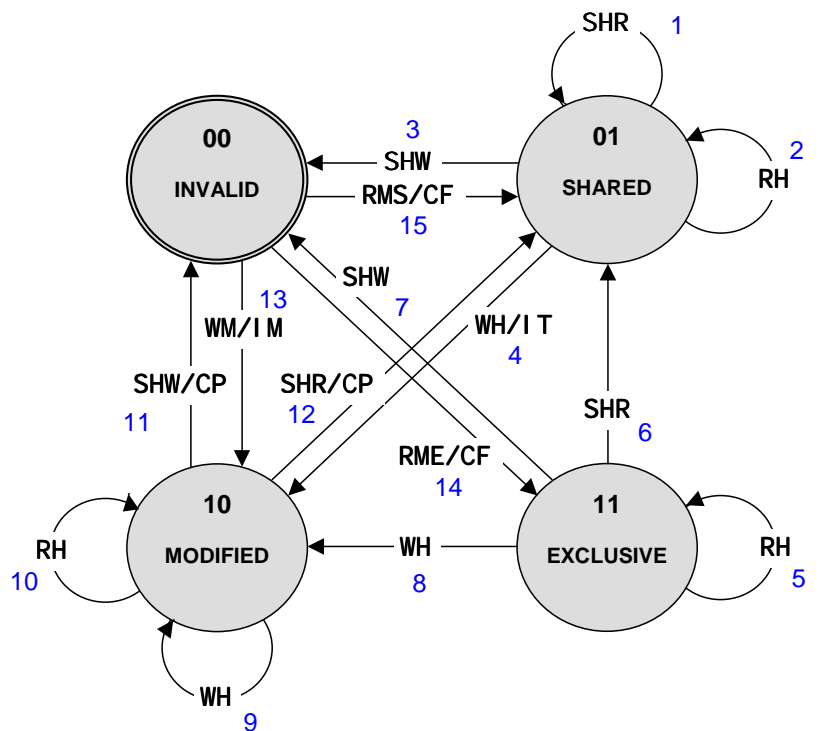
In der unten angegebenen Abbildung ist der Zustandsübergangsgraph für den Automaten zur Überwachung einer Cacheline zur Sicherstellung ihrer Konsistenz nach dem **MESI-Protokoll** eines PowerPC 60x angegeben.

Der Automat besitzt **4 Zustände** (binär kodiert in S_1S_0):

- 00 : Cacheline invalid
- 01 : Cacheline shared
- 10 : Cacheline modified
- 11 : Cacheline exclusive

Der Automat besitzt **7 Eingangssignale** von der Bus-Logik, die den Zustand einer Cacheline beeinflussen. Der Automat kann **4 Ausgangssignale** generieren, die den Prozessor und seine Bus-Logik dann entsprechend steuern:

- a) Handelt es sich bei dem Automaten (bitte ankreuzen) um einen
- Mealy-Automaten oder
 - Moore-Automaten ?

**Inputs:**

SHW = Read Hit
 RMS = Read Miss, Shared
 RME = Read Miss, Exclusive
 WH = Write Hit
 WM = Write Miss
 SHR = Snoop Hit on a Read
 SHW = Snoop Hit on a Write or Read-with-Intend-to-Modify

Outputs:

CP = Cacheline Push
 IT = Invalidate Transaction
 IM = Read-with-Intend-to-Modify
 CF = Cacheline Fill

Hinweise: Die Zahlen an den Zustandsübergängen bestimmen eine Zeile des PLA!

Zustand 01 bedeutet $S_1 = 0$ und $S_0 = 1$!

Bei Zustandsübergängen ohne Ausgabe soll der PLA keine Ausgabe erzeugen!

Eine Automatenumwandlung ist nicht erforderlich.

Name:

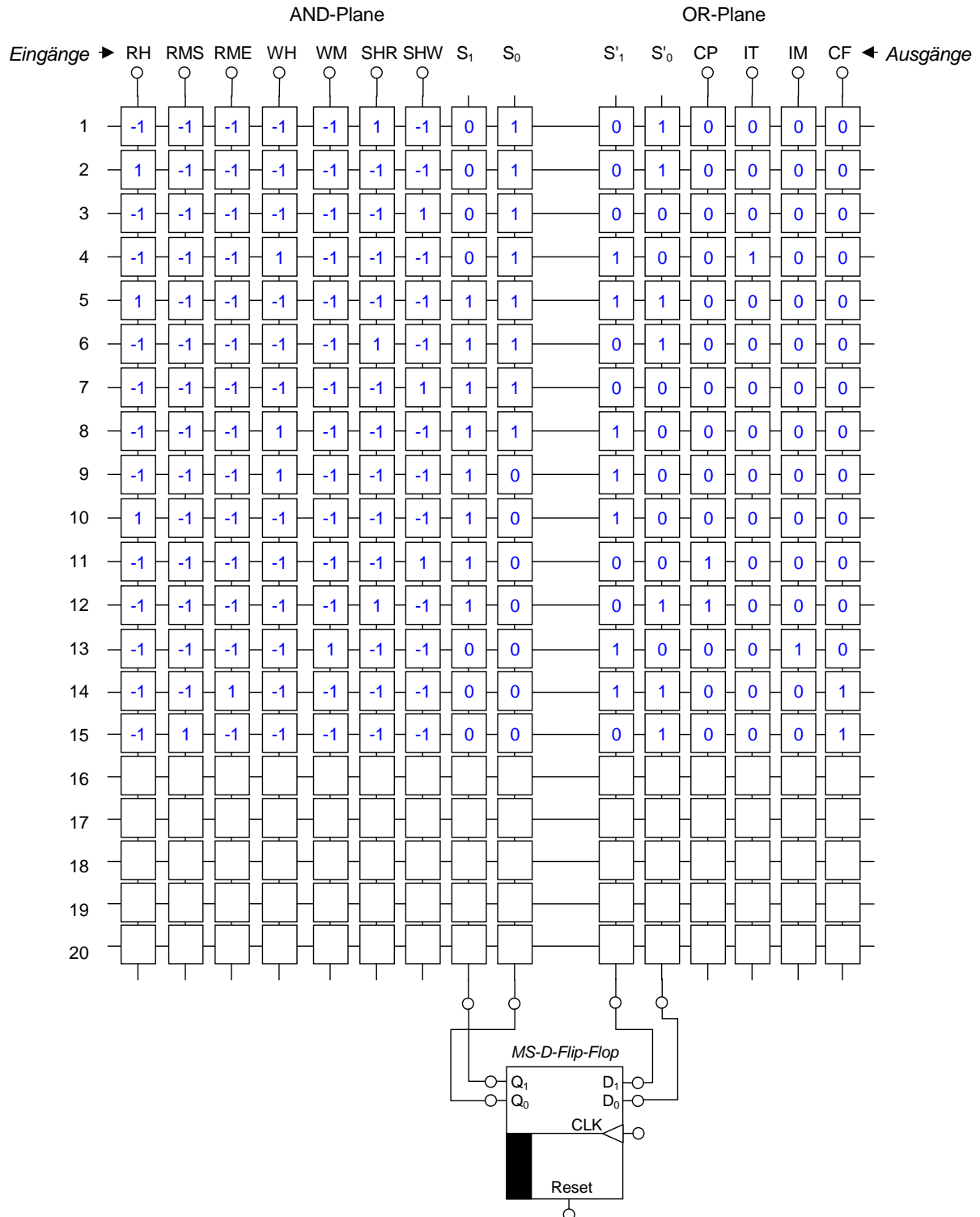
Matrikel-Nummer:

zu Aufgabe 2):

- b) Entwerfen Sie ein Steuerwerk für den angegebenen Automaten. Tragen Sie entsprechend -1, 1 und 0 in die Kästchen des PLA ein. Kästchen nicht benötigter Zeilen lassen Sie frei.

Hinweis: Der Automat erhält nur dann einen vollen Taktzyklus an seinem CLK-Eingang, wenn er ein Eingangssignal verarbeiten soll. Es kann zu einem Zeitpunkt nur ein Eingangssignal logisch '1' sein.

Zu 3): Steuerwerk mit PLA



Name:

Matrikel-Nummer:

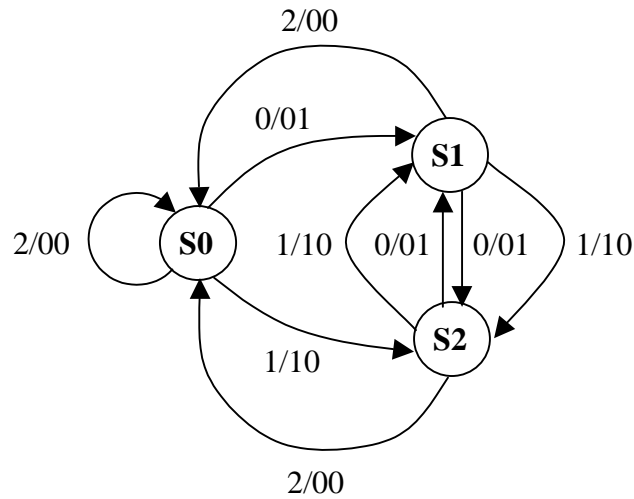
Diese Seite ist leer und steht als Konzeptpapier zur Verfügung.

Name:

Matrikel-Nummer:

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

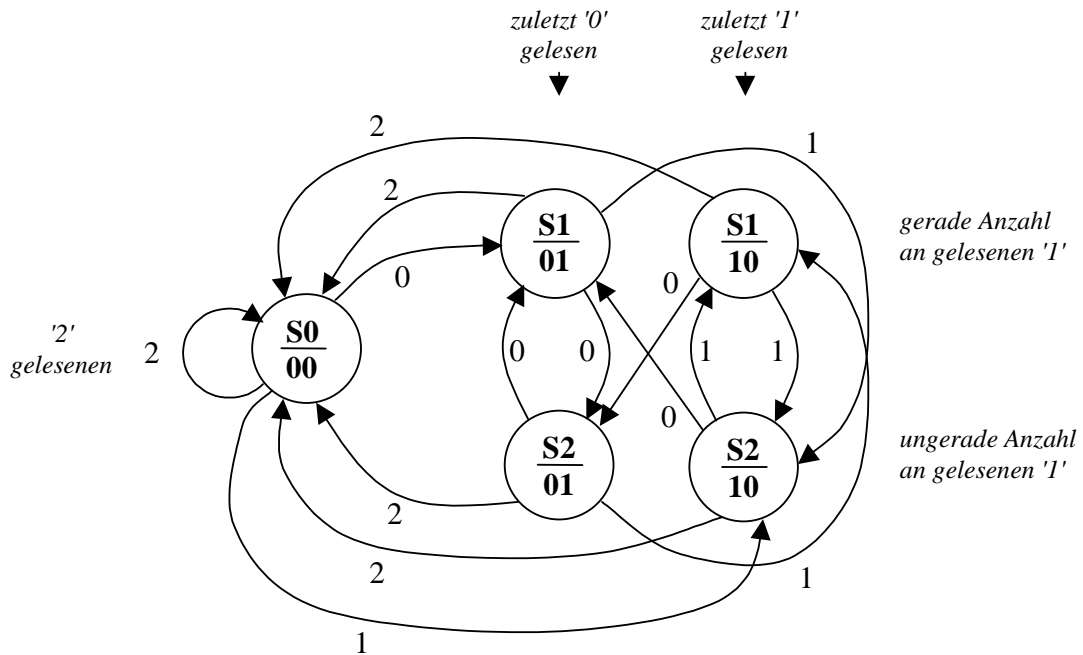
- a) Wandeln Sie den gegebenen Mealy-Automaten in einen äquivalenten Zustands-minimalen (*reduzierten*) Moore-Automaten um und geben Sie das Ergebnis auch als Zustandsübergangsgraphen an.



S0, S1, S2 sind Zustände;
1/01 steht für Eingabe X = 1, Ausgabe: Y(1) = 0, Y(0) = 1.

Ergänzen Sie bitte das vorgegebene Schema:

Lösung:



Name:

Matrikel-Nummer:

Diese Seite ist leer und steht als Konzeptpapier zur Verfügung.

Name:

Matrikel-Nummer:

Fortsetzung zu Aufgabe 3:

b) Geben Sie zu dem gegebenen VHDL-Programm den **Zustandsübergangsgraphen** an.

```
Entity AUTOMAT is
  Port ( CLK : In std_logic;
        RES : In std_logic;
        X : In std_logic;
        Y : Out std_logic_vector (1 downto 0) );
end AUTOMAT;
```

```
architecture BEHAVIORAL of AUTOMAT is
```

```
  type fsm_state is (S0, S1, S2);
  signal CURRENT_STATE, NEXT_STATE: fsm_state;
```

```
  begin
```

```
  SEQ: process (CURRENT_STATE, X)
```

```
  begin
```

```
  case CURRENT_STATE is
```

```
  when S0 =>
```

```
    if X = '0' then
```

```
      Y <= "00";
```

```
      NEXT_STATE <= S0 ;
```

```
    else
```

```
      Y <= "01";
```

```
      NEXT_STATE <= S1 ;
```

```
    end if;
```

```
  when S1 =>
```

```
    Y <= "10";
```

```
    NEXT_STATE <= S2 ;
```

```
  when S2 =>
```

```
    if X = '0' then
```

```
      Y <= "10";
```

```
      NEXT_STATE <= S0;
```

```
    else
```

```
      Y <= "01";
```

```
      NEXT_STATE <= S1 ;
```

```
    end if ;
```

```
  end case ;
```

```
end process ;
```

```
MEM: process(CLK, RES)
```

```
begin
```

```
if (RES = '0') then
```

```
  CURRENT_STATE <= S0 ;
```

```
else
```

```
  if (CLK'event and CLK = '1') then
```

```
    CURRENT_STATE <= NEXT_STATE;
```

```
  end if;
```

```
end if;
```

```
end process ;
```

```
end BEHAVIORAL ;
```

Lösung:

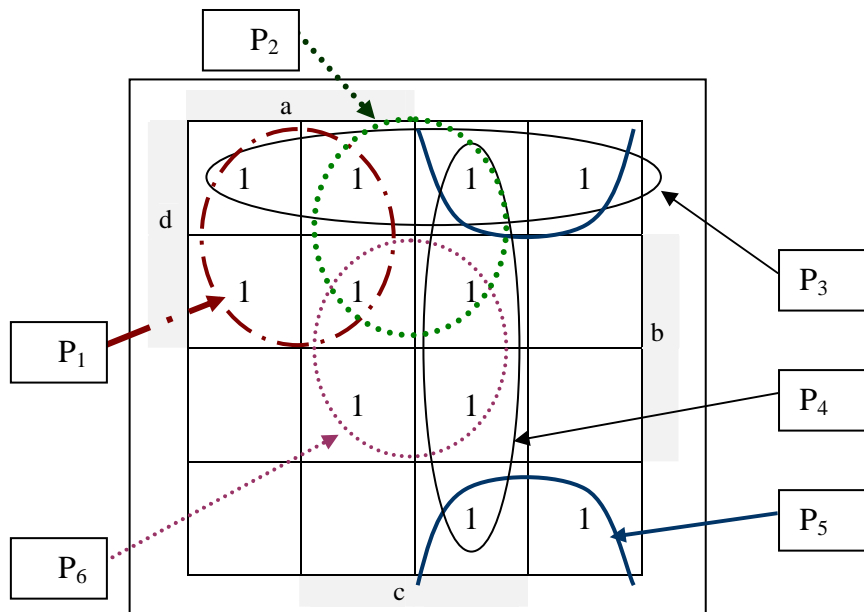
Name:

Matrikel-Nummer:

Diese Seite ist leer und steht als Konzeptpapier zur Verfügung.

Aufgabe 4 (11 min, 9 + 2 Punkte): Karnaugh Diagramme

In dem bereits ausgefüllten Karnaugh Diagramm sind 6 Primimplikanten P_1, \dots, P_6 eingetragen.
Sei $x = P_1 + P_2 + P_3 + P_4 + P_5 + P_6$.



- a) Bestimmen Sie die **minimalen** Überdeckungsindexmengen $M\ddot{U}(P_i, x)$, für alle Primimplikanten.

Lösung:

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

$$M\ddot{U}(P_2, x) = \{ \{2\}, \{3, 6\}, \{1, 3, 4\} \}$$

$$M\ddot{U}(P_3, x) = \{ \{3\}, \{1, 5\} \}$$

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

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

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

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

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

Name:

Matrikel-Nummer:

Diese Seite ist leer und steht als Konzeptpapier zur Verfügung.

Name:

Matrikel-Nummer:

Aufgabe 5 (6 min, 6 Punkte): Vereinfachung Bool'scher Funktionen

Minimieren Sie die Bool'sche Funktion

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

mit Hilfe des **aus der GTI-Vorlesung bekannten** Quine/McClusky Verfahrens. Das Verfahren benötigt hier 2 Iterationen. Geben Sie die Zwischenergebnisse nach den angegebenen Aktionen des Verfahrens in beiden Iterationen an. Bei der Summe der Konsensusse sind **alle** (d.h. **auch eventuell doppelt auftretende**) anzugeben!

Nr.	Aktion	Zwischenergebnisse
	Ausgangsfunktion F	$\bar{a} b \bar{c} d + a b \bar{c} \bar{d} + \bar{a} b \bar{c} \bar{d} + a \bar{b} \bar{c} \bar{d} + \bar{a} \bar{b} \bar{c} \bar{d}$
1	Summe aller „Simplen Konsensusse“	$\bar{a} b \bar{c} + b \bar{c} \bar{d} + a \bar{c} \bar{d} + \bar{a} \bar{c} \bar{d}$
	Nach „Streichen von Verlängerungen“ und „Vereinfachen“	$\bar{a} b \bar{c} + b \bar{c} \bar{d} + a \bar{c} \bar{d} + \bar{a} \bar{c} \bar{d}$
2	Summe aller „Simplen Konsensusse“	$\bar{c} \bar{d}$
	Nach „Streichen von Verlängerungen“ und „Vereinfachen“	$\bar{a} b \bar{c} + b \bar{c} \bar{d} + \bar{c} \bar{d}$

Ersatztable:

Wenn Sie diese Tabelle anstelle der oberen bewertet haben wollen, so streichen Sie die obere Tabelle durch!

Nr.	Aktion	Zwischenergebnisse
	Ausgangsfunktion F	$\bar{a} b \bar{c} d + a b \bar{c} \bar{d} + \bar{a} b \bar{c} \bar{d} + a \bar{b} \bar{c} \bar{d} + \bar{a} \bar{b} \bar{c} \bar{d}$
1	Summe aller „Simplen Konsensusse“	$\bar{a} b \bar{c} + b \bar{c} \bar{d} + a \bar{c} \bar{d} + \bar{a} \bar{c} \bar{d}$
	Nach „Streichen von Verlängerungen“ und „Vereinfachen“	$\bar{a} b \bar{c} + b \bar{c} \bar{d} + a \bar{c} \bar{d} + \bar{a} \bar{c} \bar{d}$
2	Summe aller „Simplen Konsensusse“	$\bar{c} \bar{d}$
	Nach „Streichen von Verlängerungen“ und „Vereinfachen“	$\bar{a} b \bar{c} + b \bar{c} \bar{d} + \bar{c} \bar{d}$

Name:

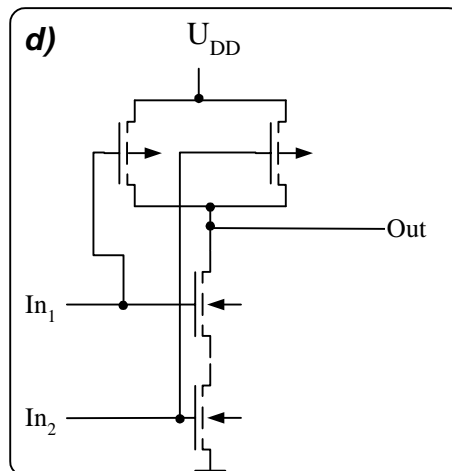
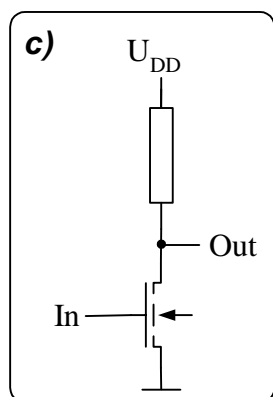
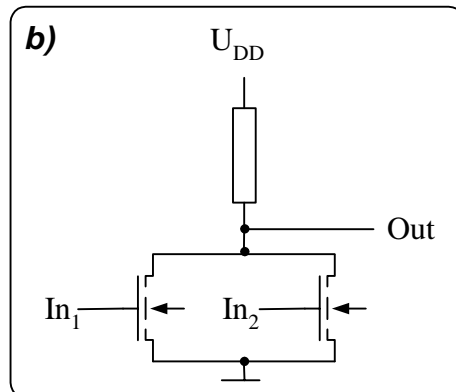
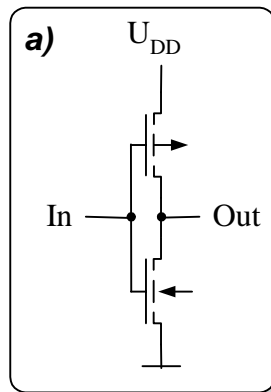
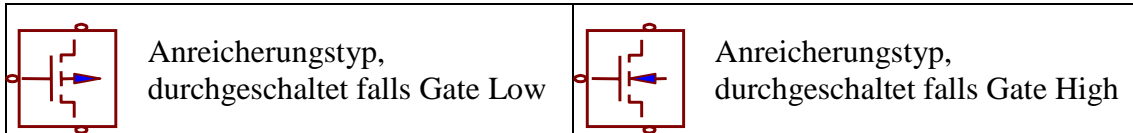
Matrikel-Nummer:

Aufgabe 6 (6 min, 1 + 2 + 1 + 2 Punkte): MOS Schaltungen

Identifizieren Sie die dargestellten Schaltungen nach verwendeter Technologie, der Art des Aufbaus sowie nach der implementierten logischen Funktion. Kreuzen Sie für jede Schaltung die entsprechenden Eigenschaften in der Tabelle an.

Bemerkung: Kreuzen Sie beim Aufbau beide Möglichkeiten an, wenn es sich bei der Schaltung um eine Mischform aus Pull-up und Pull-down handelt.

Zur Erinnerung: Funktionsweise der benutzten Transistoren:



Name:

Matrikel-Nummer:

Fortsetzung zu Aufgabe 6:

Lösungstabelle

	Technologie			Aufbau		Funktion					
	PMOS	NMOS	CMOS	Pull-up	Pull-Down	Not	And	Nand	Or	Nor	exor
a)			X	X	X	X					
b)		X			X					X	
c)		X			X	X					
d)			X	X	X			X			

Ersatztable 1

Wenn Sie diese Tabelle anstelle der oberen bewertet haben wollen, so streichen Sie die obere Tabelle durch!

	Technologie			Aufbau		Funktion					
	PMOS	NMOS	CMOS	Pull-up	Pull-Down	Not	And	Nand	Or	Nor	exor
a)											
b)											
c)											
d)											

Ersatztable 2

Wenn Sie diese Tabelle anstelle der oberen bewertet haben wollen, so streichen Sie die oberen Tabellen durch!

	Technologie			Aufbau		Funktion					
	PMOS	NMOS	CMOS	Pull-up	Pull-Down	Not	And	Nand	Or	Nor	exor
a)											
b)											
c)											
d)											

Aufgabe 7 (7 min, 1 + 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).

$$X = 15_{10} = 01111_2,$$

$$Y = 4_{10} = 00100_2, \quad -Y = -4_{10} = 11100_2$$

- 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
1	00000	1111	Shift
	00001	1110	AC \leq AC - Y
	+11100		
	11101		AC \leq AC + Y
	+00100		
	00001	1110	Shift
2	00011	1100	AC \leq AC - Y
	+11100		
	11111		AC \leq AC + Y
	+00100		
	00011	1100	Shift
3	00111	1000	AC \leq AC - Y
	+11100		
	00011		Q(0) \leq 1
	00011	1001	Shift
4	00111	0010	AC \leq AC - Y
	+11100		
	00011		Q(0) \leq 1
	Rest = 00011	0011 = Quotient	Fertig

Name:

Matrikel-Nummer:

Fortsetzung zu Aufgabe 7:

Ersatztablelle

Wenn Sie diese Tabelle anstelle der vorhergehenden bewertet haben wollen, so streichen Sie die vorhergehende Tabelle durch!

Q := X = 15 Y = 4

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
1	00000	1111	Shift
	00001	1110	AC <= AC - Y
	+11100		
	11101		AC <= AC + Y
	+00100		
	00001	1110	Shift
2	00011	1100	AC <= AC - Y
	+11100		
	11111		AC <= AC + Y
	+00100		
	00011	1100	Shift
3	00111	1000	AC <= AC - Y
	+11100		
	00011		Q(0) <= 1
	00011	1001	Shift
4	00111	0010	AC <= AC - Y
	+11100		
	00011		Q(0) <= 1
	Rest = 00011	0011 = Quotient	Fertig