

2. Klausur (29.03.2010):  
 Technische Grundlagen der Informatik 1  
 Digitale Systeme  
 WS2009/2010

<b>Vorname</b>	: .....
<b>Name</b>	: .....
<b>Matrikelnummer</b>	: .....
<b>Studiengang</b>	: .....

<b>Aufgabe</b>	1	2	3	4	5	6	7	$\Sigma$
<b>max. Punkte</b>	12	15	16	12	12	15	18	100
<b>erreichte Punkte</b>								
<b>Korrektor</b>								

**Wichtige Hinweise:**

- Mobiltelefone ausschalten
- Deckblatt ausfüllen
- Kopf aller abgegebenen Seiten mit Namen und Matrikelnummer versehen
- für die Lösung darf weder Bleistift noch Rotstift verwendet werden
- für die Lösungen die Aufgabenblätter verwenden
- der Lösungsweg muss nachvollziehbar sein und sich an der Aufgabenstellung orientieren
- Benutzung des Vorlesungsskripts ist erlaubt
- Betrugsversuche werden mit einem Nichtbestehen der Klausur geahndet

## 1. Aufgabe (12 Punkte)

Gegeben ist folgende Funktion  $y$ :

$$y = (a \rightarrow (\bar{c} e)) + a b \bar{c} + d$$

- a) Formen Sie die Funktion algebraisch in eine minimale Konjunktive Normalform um. Dabei sind die Regeln und Gesetze zur Umformung logischer Ausdrücke anzuwenden.

Tipp: Bei Umformungen erleichtern häufig Substitutionen die Schreibearbeit.

- b) Formen Sie mit Hilfe der DeMorgan'schen Gesetze die minimierte Funktion aus a) in eine NOR-Gleichungsstruktur um. Es stehen die negierten und nichtnegierten Literale sowie NOR-Gatter mit maximal vier Eingängen zur Verfügung.
- c) Zeichnen Sie das zugehörige Logikdiagramm aus NOR-Gatter.

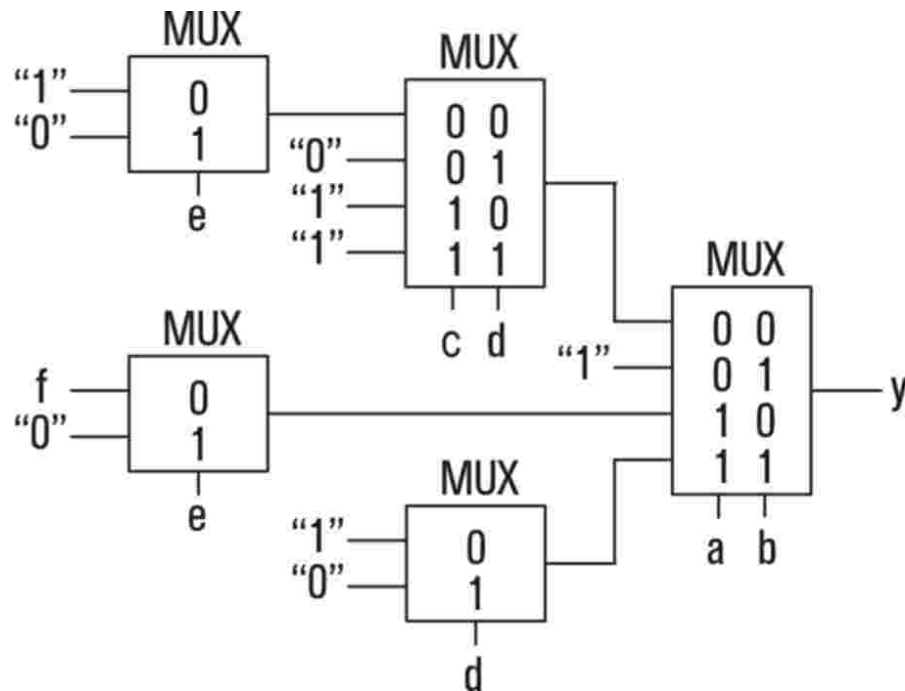
## 2. Aufgabe (15 Punkte)

Ermitteln Sie für die folgende Funktion sämtliche Primimplikanten mit Hilfe der Tison-Methode.

$$y = (\overline{a} + b \overline{c}) + (a \overline{b} + c d) (a d + \overline{a} \overline{c}) + \overline{m} l$$

### 3. Aufgabe (16 Punkte)

Gegeben ist nachfolgend abgebildetes Multiplexer-Schaltnetz.



- Stellen Sie die Funktion  $y$  entsprechend der Struktur des Multiplexer-Schaltnetzes dar, ohne zu vereinfachen.
- Vereinfachen Sie algebraisch die in a) aufgestellte Funktion  $y$ .

#### 4. Aufgabe (12 Punkte)

Bestimmen Sie aus den angegebenen Funktion  $y$  mit ihren Primimplikanten die minimale Überdeckungen mit den in der Vorlesung dazu vorgestellten Verfahren. Dokumentieren Sie bei der Lösung die Vorgehensweise.

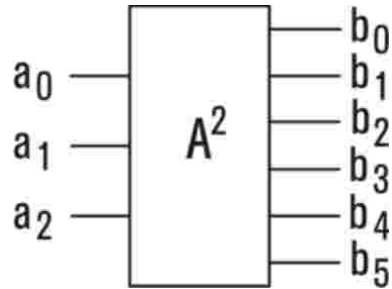
$$y = P_1 + P_2 + P_3 + P_4 + P_5 + P_6 + P_7$$

- Vervollständigen Sie die vorgegebene Mintermtabelle.
- Bestimmen Sie mit Hilfe der Mintermtabelle die minimale Überdeckung. Bitte alle möglichen Lösungsalternativen angeben!

Primimplikanten	Minterme									
$P_1: \overline{x_2} \overline{x_3}$										
$P_2: \overline{x_0} \overline{x_2}$										
$P_3: \overline{x_0} \overline{x_1} \overline{x_3}$										
$P_4: \overline{x_1} \overline{x_2} \overline{x_3}$										
$P_5: \overline{x_0} \overline{x_2} \overline{x_3}$										
$P_6: \overline{x_0} \overline{x_1} \overline{x_2}$										
$P_7: \overline{x_0} \overline{x_1} \overline{x_3}$										

## 5. Aufgabe (12 Punkte)

Für die dreistellige Dualzahl  $A = a_2 a_1 a_0$  ist ein Quadrierschaltnetz nach angegebenen Blockschaltbild zu entwerfen. Das Ergebnis wird mit der sechsstelligen Dualzahl  $B = b_5 b_4 b_3 b_2 b_1 b_0$  ausgegeben.



- Erstellen Sie die Wertetabelle für das Quadrierschaltnetz.
- Stellen Sie die minimalen Funktionen für die einzelnen Ergebnisbits  $b_i$  auf.

## 6. Aufgabe (15 Punkte)

Entwerfen Sie auf der Basis eines Moore-Automaten einen synchronen Untersetzer, dessen Ausgangssignal A während jedes 5. Taktzyklus eine „1“ zeigt. Der Startzustand  $z_2 z_1 z_0$  soll „100“ mit A = „1“ sein. Zum Entwurf stehen D-Master-Slave-Flipflops, NAND-Gatter und Inverter zur Verfügung.

- a) Erstellen Sie den Zustandsgraphen und die Zustandstabelle.
- b) Entwerfen Sie die Ansteuergleichungen für die D-Master-Slave-Flipflops.
- c) Zeichnen Sie das Logikdiagramm für den Untersetzer.

## 7. Aufgabe (18 Punkte)

Entwerfen Sie für das nachfolgend gegebene Funktionsbündel  $f_1, f_2$  ein vollständiges VHDL-Modell. Die Modelle für die zu instanzierenden Komponenten sind gegeben. Von den Eingangssignalen  $e_i$  stehen nur die nichtnegierten Literale zur Verfügung. Alle Signale und Ports sind vom Typ `bit`.

$$f_1 = e_1 \overline{e_3} + \overline{e_2}$$

$$f_2 = \overline{e_2} + \overline{e_3}$$

```
entity INV is
  port (a : in bit; b : out bit);
end INV;
```

```
architecture GATE of INV is
begin
  b <= not a;
end GATE;
```

```
entity NAND is
  port (x,y : in bit; z : out bit);
end NAND;
```

```
architecture GATE of NAND is
begin
  z <= x nand y;
end GATE;
```

- Zeichnen Sie das Schaltbild der entity( $f_1, f_2$ ) mit allen für das zu entwerfende VHDL-Modell erforderlichen Bezeichnern.
- Entwerfen Sie das VHDL-Modell für die entity( $f_1, f_2$ ).