

Professor Dr.-Ing. Stefan Kowalewski

Hilal Diab, M.Sc.

Kamal Barakat, M.Sc.

Dipl.-Inform. Dominik Franke

Aachen, 22. Januar 2010

SWS: V4/Ü2, ECTS: 7

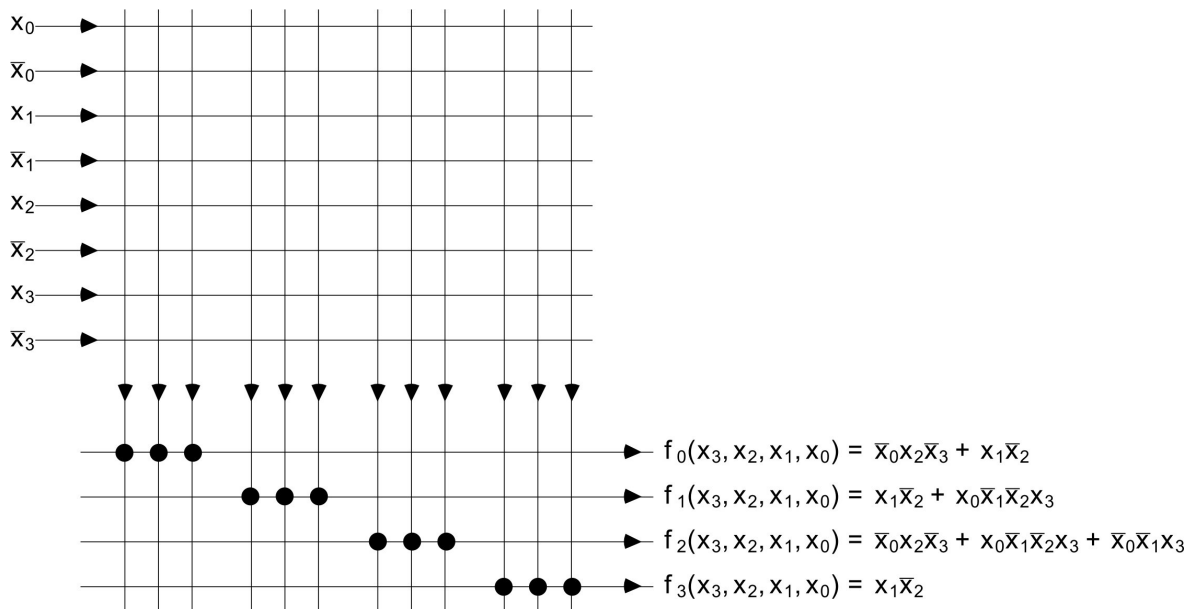
Einführung in die Technische Informatik

WS 2009/2010

Blatt 11: Lösungsvorschlag

Aufgabe 1: PAL

Skizzieren Sie, wie sich die unten angegebenen Booleschen Funktionen f_0, f_1, f_2, f_3 mittels eines PAL realisieren lassen, indem Sie die entsprechenden Punkte in das folgende PAL eintragen.



Aufgabe 2: (*)PLA

Gegeben sei das folgende PLA:

| | | | | | |
|-------|---|---|---|---|-----------|
| x_1 | → | 2 | 3 | 0 | 3 |
| x_2 | → | 2 | 3 | 0 | 0 |
| x_3 | → | 2 | 0 | 3 | 2 |
| <hr/> | | | | | |
| | | 1 | 0 | 0 | 1 → f_1 |
| | | 0 | 1 | 1 | 0 → f_2 |

Schreiben Sie die durch das PLA realisierten Funktionen f_1 und f_2 in einer Wahrheitstabelle auf.

Lösungsvorschlag

Die durch das PLA dargestellte Funktion kann wie folgt beschrieben werden:

| x_1 | x_2 | x_3 | f_1 | f_2 |
|-------|-------|-------|-------|-------|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 |

Die Funktionen können direkt am PLA abgelesen werden. Sie lauten:

$$f_1(x_3, x_2, x_1) = x_3 x_2 x_1 + x_3 \bar{x}_1$$

$$f_2(x_3, x_2, x_1) = \bar{x}_2 \bar{x}_1 + \bar{x}_3$$

Aufgabe 3: Duale Gleitkommazahlen

- a) Interpretieren Sie die folgenden Bit-Strings jeweils als IEEE-Gleitkommazahl und geben Sie die entsprechende Dezimalzahl an:

11000001111011010000000000000000
10111110010100000000000000000000

- b) Berechnen Sie im Dualsystem das Produkt der beiden dargestellten Zahlen. Stellen Sie das Ergebnis wieder im IEEE-Format dar.
- c) Berechnen Sie im Dualsystem die Summe der beiden dargestellten Zahlen. Stellen Sie das Ergebnis wieder im IEEE-Format dar.

Aufgabe 4: (★)Duale Gleitkommazahlen

Stellen Sie die Zahlen -10,625 und 0,3828125 im IEEE-754 Format dar.

Lösungsvorschlag

-10,625:

negative Zahl, also sign = 1

Vorkommastellen: $(10 = 8 + 2)_{10} = (1010)_2$

Nachkommastellen: $(0,625 = \frac{1}{2} + \frac{1}{8})_{10} = (0,101000...)_{2}$

Normalisierung: $(10,625)_{10} = (1010,101)_2 = (1,010101)_2 \cdot 2^3$

Exponent: $(3)_{Exponent} = (3 + 127)_{10} = (10000010)_2$

-10,625 = 1 10000010 010101000000000000000000

0,3828125:

positive Zahl, also sign = 0

Vorkommastelle: $(0)_{10} = (0)_2$

Nachkommastellen:

$$0,3828125 \cdot 2 = 0,765625 \Rightarrow 0$$

$$0,765625 \cdot 2 = 1,53125 \Rightarrow 1$$

$$0,53125 \cdot 2 = 1,0625 \Rightarrow 1$$

$$0,0625 \cdot 2 = 0,125 \Rightarrow 0$$

$$0,125 \cdot 2 = 0,25 \Rightarrow 0$$

$$0,25 \cdot 2 = 0,5 \Rightarrow 0$$

$$0,5 \cdot 2 = 1 \Rightarrow 1$$

Normalisierung: $(0 + 0,0110001)_2 = (1,10001)_2 \cdot 2^{-2}$

Exponent: $(-2 + 127)_{10} = (64 + 32 + 16 + 8 + 4 + 1)_{10} = (01111101)_2$

$0,3828125 = 0\ 01111101\ 100010000000000000000000$

Aufgabe 5: Zahlendarstellung

- a) Geben Sie für die folgenden Zahlen jeweils die Vorzeichen-, Einer- und Zweier-Komplement-Darstellung an. Gehen Sie von einer Wortlänge von $n = 16$ aus.

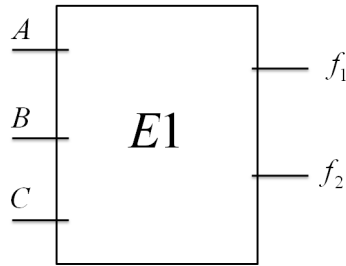
+66, -101, -204, +204, +198, -523, -1021

- b) Geben Sie eine vereinfachte Vorschrift zur Berechnung des Zweier-Komplements einer Dualzahl an, mit der das Zweier-Komplement in einem Schritt erzeugt wird und nicht mit den zwei Schritten Komplementbildung und Addition von 1. Begründen Sie die Korrektheit ihrer Vorschrift.
- c) Berechnen Sie im 4-Bit-Zweier-Komplement $4+3$, $-4+3$, $4+5$, $-4+(-5)$ und wandeln Sie das Ergebnis in das Dezimalsystem um.
- d) Bei Addition positiver Zahlen erkennt man einen Überlauf am Auftreten eines Übertrags an der höchstwertigen Bitposition. Bei der Addition im Zweier-Komplement wird dieser Übertrag normalerweise ignoriert. Warum? Woran kann man bei Addition im Zweier-Komplement einen Überlauf erkennen?
- e) Stellen Sie die Zahlen 328 und 470 im BCD-Code dar!
- f) Interpretieren Sie folgende Bitfolge als BCD-Code und berechnen Sie den dezimalen Wert:

1001011001010011

Aufgabe 6: (★)VHDL

- a) Es soll folgendes Modul mit den Eingängen A, B, C und den Ausgängen f_1 und f_2 in VHDL beschrieben werden:



Geben Sie eine Hardware Spezifikation für die Ein- und Ausgänge des Moduls an. Benutzen Sie dazu die **ENTITY** Umgebung.

- b) Das Verhalten der Ausgänge soll so spezifiziert werden, sodass die Funktionen $f_1 = A + \bar{B}$ und $f_2 = A + (C \cdot \bar{B})$ realisiert werden. Vervollständigen Sie folgenden VHDL Code:

```
architecture P1 of E1 is
begin
  process(A, B, C)
  begin
    if (A = '1') then
      f1 <= '1';
      f2 <= '1';
    else
      — insert code here
    end if;
  end process;
end P1;
```

- c) Erstellen Sie eine weitere Verhaltensbeschreibung mithilfe der **architecture** Umgebung. Verzichten Sie diesmal jedoch auf Prozesse.

Lösungsvorschlag

a)

```
entity E1 is
  port(A,B,C: in std_logic;
        f1, f2: out std_logic);
end E1;
```

b)

```
architecture P1 of E1 is
begin
  process(A, B, C)
  begin
    if (A = '1') then
```

```

        f1 <= '1';
        f2 <= '1';
    else
    — insert code here
        if (B = '0') then
            f1 <= '1';
            if (C = '1') then
                f2 <= '1';
            else
                f2 <= '0';
            end if;
        else
            f1 <= '0';
            f2 <= '0';
        end if;
    — end insert code
    end if;
end process;
end P1;

```

c)

```

architecture P2 of E1 is
begin
    f1 <= A or (not B);
    f2 <= A or (C and (not B));
end P2;

```