

Rechenbeispiele

Beispiel 1

Abbildung 1 zeigt die FET-Realisierung einer logischen Funktion $y(a,b,c)$.

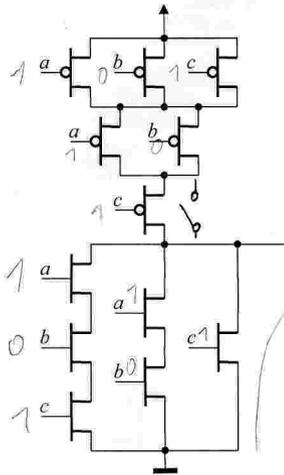
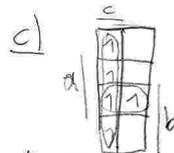


Abbildung 1

a) $Y = 0$

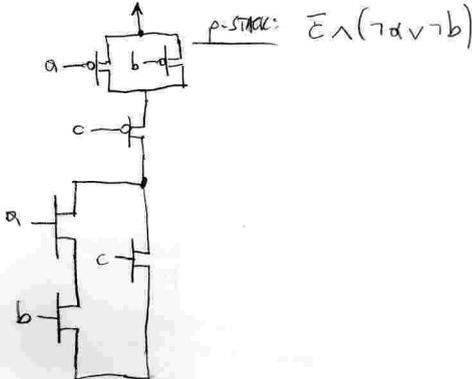
b) $\bar{Y} = (\bar{a}bnc) \vee (a\bar{b}nc) \vee c$
 $Y = (\bar{a} \vee \bar{b} \vee \bar{c}) \wedge (\bar{a} \vee \bar{b}) \wedge \bar{c}$



d) $\bar{Y} = c \vee (\bar{a} \wedge \bar{b}) = N\text{-STACK}$

- Leiten Sie anhand dieses Schaltbildes her, welchen Wert der Ausgang y für folgendes Bitmuster am Eingang annimmt: $(a,b,c) = (1,0,1)$!
- Geben Sie die Funktion y in Form einer Booleschen Gleichung an!
- Zeichnen Sie das KV-Diagramm dieser Funktion y !
- Leiten Sie aus dem KV Diagramm eine vereinfachte Darstellung der Funktion y her
- Zeichnen Sie die FET-Realisierung dieser vereinfachten Form. Was hat sich geändert? Warum?

e)



2c)

$$Y_3 = (b_0 \wedge b_2 \wedge \neg b_1) \vee (b_0 \vee b_1 \wedge \neg b_2)$$

$$Y_3(b_0=0) = (0 \wedge b_2 \wedge \neg b_1) \vee (1 \wedge b_1 \wedge \neg b_2) = (b_1 \wedge \neg b_2)$$

$$Y_3(b_0=1) = (1 \wedge b_2 \wedge \neg b_1) \vee (0 \wedge \neg b_1 \wedge \neg b_2) = (b_2 \wedge \neg b_1)$$

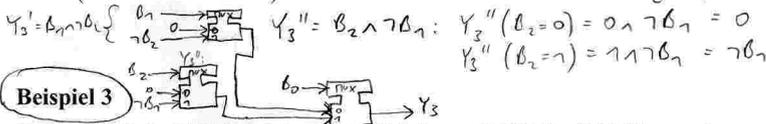
$$Y_3' = b_1 \wedge \neg b_2 : Y_3'(b_1=0) = 0 \wedge \neg b_2 = 0$$

$$Y_3'(b_1=1) = 1 \wedge \neg b_2 = \neg b_2$$

Beispiel 2

Zu realisieren ist eine Schaltung, die von einer Zahl b das Quadrat y berechnet. Der Eingangswert b ist als positive Binärzahl (b_2, b_1, b_0) dargestellt, ebenso das Ergebnis y (y_3, y_2, y_1, y_0).

- Beschreiben Sie eine Realisierung als ROM: Welches ROM benötigen Sie (Speicherzellen, Wortbreite), wie beschalten Sie die Adressen, wie die Daten? Geben Sie eine Liste der Speicherinhalte an!
- Gehen Sie von der Liste der Speicherinhalte aus (a) aus und beschreiben Sie die logischen Verknüpfungen, die Sie für y_0, y_1, y_2 und y_3 realisieren müssten, wenn Sie die Funktion mittels logischer Grundfunktionen realisieren wollten.
- Leiten Sie eine Realisierung für die Funktion $y_3(b_2, b_1, b_0)$ mittels Multiplexer-basierter Logik (wie im ACTEL-FPGA) her! Zeichnen Sie das Schaltbild Ihrer Lösung!



Beispiel 3

Gegeben ist die in Abbildung 2 dargestellte Schaltung mit 100MHz Takt. Ein asynchrones Eingangssignal mit 2MHz liegt am Eingang von Flip-Flop FF1. Der Ausgang von FF1 geht über kombinatorische Logik COMB an den Eingang von Flip-Flop FF2. Die Verzögerungszeit der kombinatorischen Logik beträgt 2ns. Laut Datenblatt haben die beiden Flip-Flops folgende Parameter:

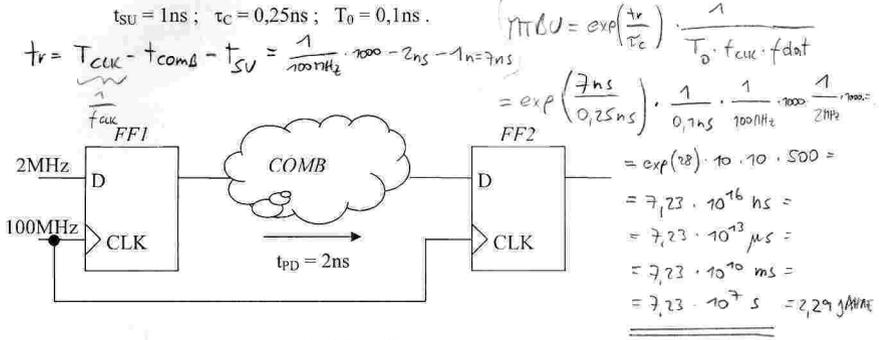


Abbildung 2

- Welche MTBU ist zu erwarten?
- Die Testabteilung schreibt nun vor, dass vor die D-Eingänge der beiden Flipflops jeweils ein Multiplexer geschaltet werden muss. Welchen Grund wird die Testabteilung dafür haben?
Für den Schrittzähler werden Multiplexer für die Flipflops zum gemeinsamen Schrittzähler verbunden.
- Die Multiplexer verursachen im Datenpfad ein Delay von 2ns. Welche MTBU ergibt sich nun?
- Auf welchen Wert müssten Sie die Frequenz des asynchronen Eingangssignals verringern, um wieder die ursprüngliche MTBU zu erreichen?

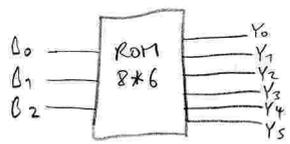
c) $t_r = 7ns - 2ns = 5ns$ *Schleifen, da Flip-Flop weniger Zeit hat zu empfangen.*

$MTBU = \exp\left(\frac{5ns}{0,25ns}\right) \cdot \frac{1}{0,1ns} \cdot \frac{1}{100MHz} \cdot 1000 \cdot \frac{1}{2MHz} \cdot 1000 = 2,42 \cdot 10^{13} ns = 6,74 h$

d) $MTBU = 7,23 \cdot 10^{16} ns = \exp\left(\frac{5ns}{0,25ns}\right) \cdot \frac{1}{0,1ns} \cdot \frac{1}{100MHz} \cdot 1000 \cdot \frac{1}{x MHz} \cdot 1000 \Rightarrow x = \frac{0}{0}$

2) a)

	(MSB)		(LSB)		(MSB)			(LSB)			
	B_2	B_1	B_0	B_0	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0	
0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	1	1
2	0	1	0	0	0	0	1	0	0	0	4
3	0	1	1	0	0	1	0	0	0	1	9
4	1	0	0	0	0	1	0	0	0	0	16
5	1	0	1	0	0	1	1	0	0	1	25
6	1	1	0	1	1	0	0	1	0	0	36
7	1	1	1	1	1	1	0	0	0	1	49



48 SPEICHERZELLE
6 BIT WORTBREITE

b)

$Y_0:$

0	0
1	1
1	1
0	0

(LSB) B_0 | B_1

$Y_1:$

0	0
0	0
0	0
0	0

(LSB) B_0 | B_1

$Y_1 = 0$

$Y_3:$

0	0
1	0
0	1
0	0

(LSB) B_0 | B_1

$Y_3 = (B_0 \wedge B_2 \wedge \neg B_1) \vee (B_0 \wedge B_1 \wedge \neg B_2)$

$Y_5:$

0	0
0	0
1	0
1	0

(LSB) B_0 | B_1

$Y_5 = (B_2 \wedge B_1)$

c)

$Y_2 = (B_1 \wedge B_0 \wedge \neg B_2) \vee (\neg B_1 \wedge B_0 \wedge B_2)$

$Y_3 = [B_1 \wedge (B_0 \wedge \neg B_2)] \vee [\neg B_1 \wedge (B_0 \wedge B_2)]$

$B_1:$

$$\text{MTBU} = 7,23 \cdot 10^{-16} \text{ ns} = 4,85 \cdot 10^{-13} \text{ ns} \cdot \frac{1}{X} \quad / \cdot X \quad / : 7,23 \cdot 10^{-16} \text{ ns}$$

$$X = \frac{4,85 \cdot 10^{-13} \text{ ns}}{7,23 \cdot 10^{-16} \text{ ns}} = \underline{\underline{0,000670925 \text{ MHz}}} = \underline{\underline{670,816 \text{ Hz}}}$$

Philipp Metzler - phil@goli.at
Goli.at - www.goli.at

AUF DIESE FREQUENZ VERSTIMMEN,
UM URSPRÜNGLICHE MTBU ZU ERREICHEN.

trick : process (yes, no)
begin
if yes'event and yes = 1 then
if no = '0' then
one <= "00";
else
one <= two;
end if;
end if;
end process trick;

Beispiel 4

Gegeben ist folgender VHDL-Code einer State-Machine:

```
library IEEE;
use IEEE.std_logic_1164.all;

entity silence is
port(
yes : in std_logic;
no : in std_logic;
upset : in std_logic
alpha: out std_logic_vector(0 to 1);
end silence;
```

```
architecture sound of silence is
signal one, two:
std_logic_vector(0 to 1);
begin
tick: process(one)
begin
state next_state
case one is
when "00" => two <= "01";
when "01" => two <= "10";
when "10" => two <= "11";
when others => two <= "00";
end case;
end process tick;
```

NEXT-STATE LOGIC:
ASYNCRONOM: BEWECHNUNG DER NÄCHSTEN ZUSTÄNDE

SYNCHRONER TEIL:
SYNCHRONER ZUSTANDSWECHEL

```
trick: process(yes, no)
begin
if (no) = '0' then
one <= "00";
else
if (yes) event and yes = '1' then
one <= two;
end if;
end if;
end process trick;
```

OUTPUT LOGIC:
ASYNCRONOM! BEWECHNUNG DER AUSGÄNDE

```
track: process(one)
begin
if upset = "1" then
alpha <= "111";
else
case one is
when "00" => alpha <= "101";
when "01" => alpha <= "001";
when "10" => alpha <= "110";
when others => alpha <= "010";
end case;
end if;
end process track;
end sound;
```

YES " NO "

- (a) Identifizieren Sie Taktsignal und Reset!
- (b) Nennen Sie die 3 Funktionsblöcke, die eine State-Machine stets umfassen muß? Welcher Process im obigen Code entspricht welchem Funktionsblock?
- (c) Um welchen Typ von State-Machine (Moore oder Mealy) handelt es sich? Begründen Sie!
- (d) Verändern Sie den Code so, dass der Reset nun synchron anstelle von asynchron wird (Schreiben Sie den Code für den entsprechend veränderten Process vollständig an)!

NEBU, DA SICH FUNKTION UPSET UM RISTEL BEM AUF AUSGÄNDE AUSWIRKT.