

PROIECT CID

- Rezolvarea temei de proiect pe hartie :

Lupu Miruna
Grupa 2121
12-D-II

~ Proiect ~

1) Automatul de tranziții:

2) Tabelul de adv. pt. fct. f:

a	b	c	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

3) $f = a\bar{c} + \bar{a}b + bc$

4) $f = b\bar{c} + ab + \bar{a}c$

5) Tabelul de adv. MUX 2-1:

Select line	Input	Output	
A_0	I_0	I_1	Y
0	0	0	0
0	0	1	1
1	1	0	1
1	1	1	1

6) Circuit logic diagram for $f = a\bar{c} + \bar{a}b + bc$.

7) Circuit logic diagram for $f = b\bar{c} + ab + \bar{a}c$.

8) Truth table for the 3-bit counter:

Q_2	Q_1	Q_0	Q_2^+	Q_1^+	Q_0^+
0	0	0	1	0	0
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	0	1	1

9) starea viitoare

c	b	\bar{b}	a
0	0	1	0
0	1	0	1
1	0	1	1
1	1	0	0

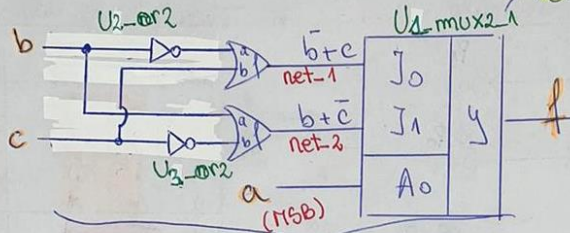
c	b	\bar{b}	a
0	0	1	0
0	1	0	1
1	0	1	1
1	1	0	0

a	f
0	$b+c$
1	$b+\bar{c}$

$$\Rightarrow J_0 = \bar{b} + c$$

$$\Rightarrow J_1 = b + \bar{c}$$

Circuitul combinational



MUX2:1 + parti logice

$Q_2 Q_1 Q_0$	00	01	11	10
Q_0	1	X	1	1
1				X

$$\Rightarrow D_2 = \bar{Q}_0$$

$Q_2 Q_1 Q_0$	00	01	11	10
Q_0		X	1	1
1				X

$$\Rightarrow D_1 = Q_2$$

D_1
0
0
X
0
1
X
1
0

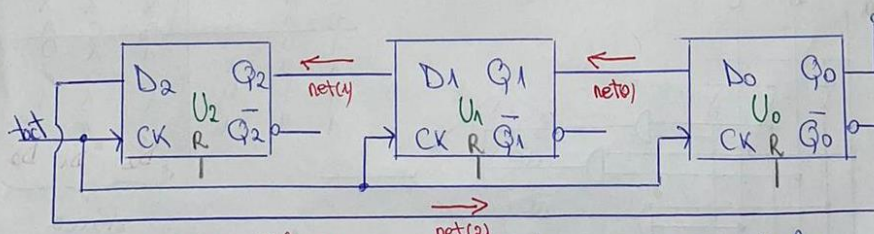
$Q_2 Q_1 Q_0$	00	01	11	10
Q_0		X	1	
1			1	X

$$\Rightarrow D_0 = Q_1$$

D_0
0
0
X
1
0
X
1
1

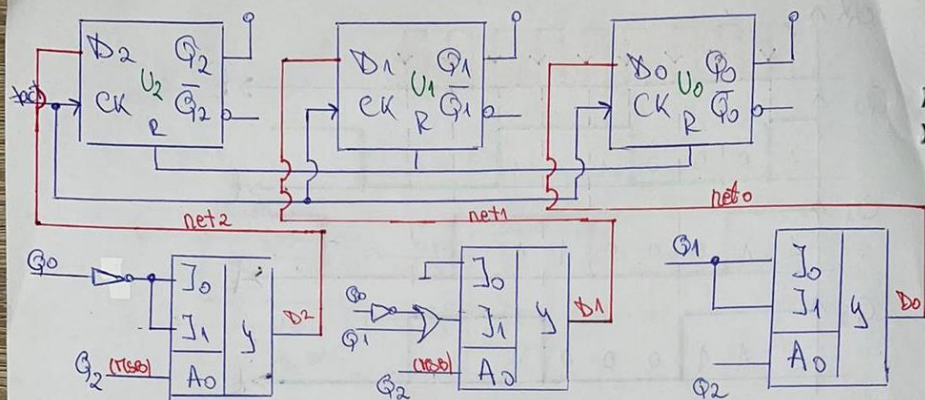
o1 Bistabilul D:

r	clk	Action
1	X	Reset
0	\downarrow	$Q^+ = D$
otherwise		Wait



numărător Johnson
2 bucle posibile în funcție de starea inițială

Circuitul secvențial



- Capturi de ecran :
- Circuitul combinational: **MUX2:1**

C:/Miruna/LupuMiruna_2111_Proiect_CID/srcs/sources_1/new/mux2_1.vhd



```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4
5  entity mux2_1 is
6      Port ( I0 : in STD_LOGIC;
7            I1 : in STD_LOGIC;
8            a : in STD_LOGIC;
9            y : out STD_LOGIC);
10 end mux2_1;
11
12 architecture Behavioral of mux2_1 is
13 begin
14
15     process (I0,I1,a)
16     begin
17         if a='0' then
18             y <= I0;
19         else
20             y <= I1 ;
21         end if;
22     end process;
23
24 end Behavioral;
25
```

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity testmux2_1 is
-- Port ( );
end testmux2_1;

architecture Behavioral of testmux2_1 is

component mux2_1 is
Port ( I0 : in STD_LOGIC;
      I1 : in STD_LOGIC;
      a : in STD_LOGIC;
      y : out STD_LOGIC);
end component mux2_1;

signal in0, in1, outy, a, b, c: std_logic;

begin

☐ in0 <= (not b) or c;
☐ in1 <= b or (not c);

MUX: mux2_1 port map ( I0 => in0, I1 => in1, a => a, y => outy);

generate_ina: process
begin
☐ a <= '0'; wait for 5 ns;
☐ a <= '1'; wait for 5 ns;
end process;

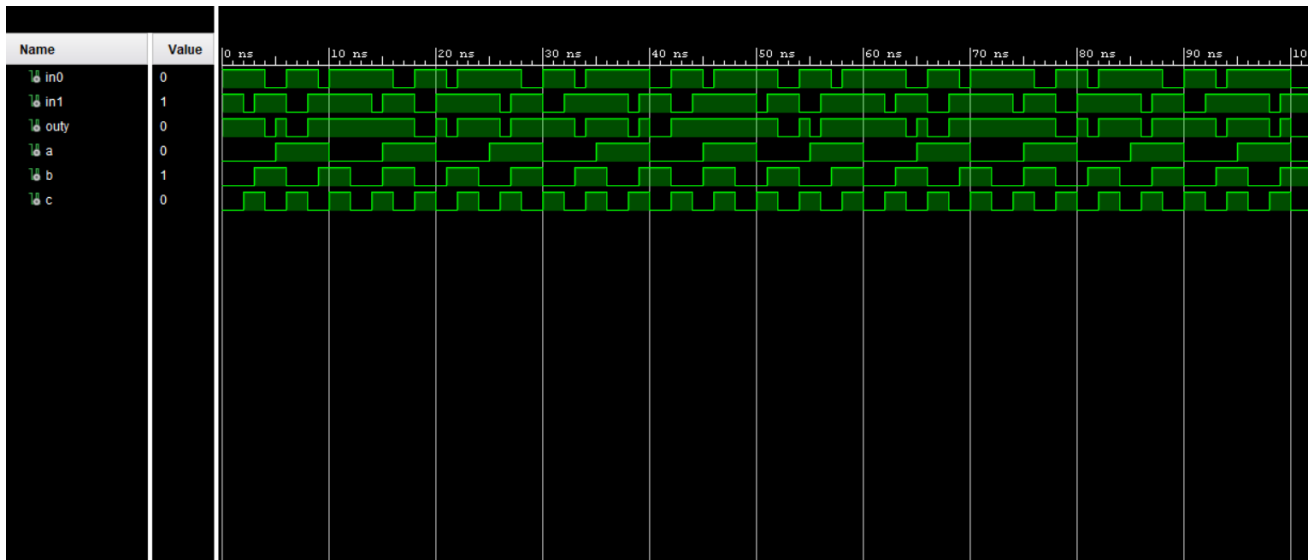
generate_ina: process
begin
a <= '0'; wait for 5 ns;
a <= '1'; wait for 5 ns;
end process;

generate_in0: process
begin
b <= '0'; wait for 3 ns;
b <= '1'; wait for 3 ns;
end process;

generate_in1: process
begin
c <= '0'; wait for 2 ns;
c <= '1'; wait for 2 ns;
end process;

end Behavioral;

```



- Circuitul secvential : **Bistabilul D**

C:/Miruna/LupuMiruna_2111_Proiect_CID/srcs/sources_1/new/bistabil.vhd

```

1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4
5  entity bistabil is
6      Port ( D : in STD_LOGIC;
7            CLK : in STD_LOGIC;
8            R : in STD_LOGIC;
9            Q : out STD_LOGIC);
10 end bistabil;
11
12 architecture Behavioral of bistabil is
13
14     signal qint: std_logic;
15
16     begin
17
18     bistabil: process(R, CLK)
19     begin
20         if (R ='1') then
21             qint<= '1';
22         elsif (CLK'event and falling_edge (CLK)) then
23             qint <= d;
24         else
25             qint <= qint;
26         end if;
27     end process;
28
29     Q <= qint;
30
31 end Behavioral;

```

SIMULAREA:

C:/Miruna/LupuMiruna_2111_Proiect_CID/srcs/sim_1/new/simulation.vhd



```
1 library IEEE;
2 use IEEE.STD_LOGIC_1164.ALL;
3
4
5 entity testbench_dff is
6 end testbench_dff;
7
8 architecture Behavior of testbench_dff is
9
10     component bistabil is -- componenta bistabilului
11     Port ( D : in STD_LOGIC;
12           CLK : in STD_LOGIC;
13           R : in STD_LOGIC;
14           Q : out STD_LOGIC);
15     end component;
16
17     component mux2_1 is -- componenta mux2:1
18     Port ( I0 : in STD_LOGIC;
19           I1 : in STD_LOGIC;
20           a : in STD_LOGIC;
21           y : out STD_LOGIC);
22     end component;
23
24
25     constant clk_p : time := 10 ns; -- perioada de timp
26     signal clock : STD_LOGIC := '0';
27     signal reset : STD_LOGIC := '0';
28     signal Q : STD_LOGIC_VECTOR (2 downto 0) := "111"; --setam starea initiala ca fiind "111" de la automatul de tranzitii
29     signal D0, D1, D2 : STD_LOGIC := '1';
30     signal not_Q0 : STD_LOGIC := '1'; -- poarta Q0 negat intrucat o folosim la doar mux-uri
31     signal mux1_I1 : STD_LOGIC := '1'; -- poarta or intre Q0 negat si Q1
```

C:/Miruna/LupuMiruna_2111_Proiect_CID/srcs/sim_1/new/simulation.vhd



```
31     signal mux1_I1 : STD_LOGIC := '1'; -- poarta or intre Q0 negat si Q1
32
33 begin
34
35     not_Q0 <= not Q(0);
36     mux1_I1 <= (not Q(0)) or Q(1);
37
38     BST0: bistabil port map( D => D0, CLK => clock, R => reset, Q => Q(0)); -- 3 bistabile realizate de starea viitoare din tabelul de adevar
39     BST1: bistabil port map( D => D1, CLK => clock, R => reset, Q => Q(1));
40     BST2: bistabil port map( D => D2, CLK => clock, R => reset, Q => Q(2));
41
42     MUX0: mux2_1 port map(I0 => Q(1), I1 => Q(1), a => Q(2), y => D0); -- 3 mux-uri realizate de starea viitoare din tabelul de adevar
43     MUX1: mux2_1 port map(I0 => '0', I1 => mux1_I1, a => Q(2), y => D1);
44     MUX2: mux2_1 port map(I0 => not_Q0, I1 => not_Q0, a => Q(2), y => D2);
45
46     clk: process -- procesul de clock
47     begin
48         clock <= '0';
49         wait for clk_p/2;
50         clock <= '1';
51         wait for clk_p/2;
52     end process;
53
54
55     stim: process -- procesul de reset
56     begin
57         reset <= '1';
58         wait for 10 ns;
59         reset <= '0';
60         wait;
```

```
55
56 stim: process -- procesul de reset
57 begin
58     reset <= '1';
59     wait for 10 ns;
60     reset <= '0';
61     wait;
62 end process;
63
64 end;
```

REZULTATUL SIMULARII:

The screenshot shows a logic analyzer interface with a timing diagram. The top bar indicates the current file is 'Untitled 114'. The left sidebar shows the 'Scope' tab with a list of signals: clock, reset, Q[2:0], [2], [1], [0], D0, D1, D2, not_Q0, mux1_1, and clk_p. The main area displays a timing diagram with a time scale of 10000 ps. A yellow vertical line marks a point at 4.950 ns. The output signal Q[2:0] is highlighted in blue, showing a value of 111 at this time. The diagram also shows a clock signal and several input signals, all represented by green waveforms.

Signal	Value at 4.950 ns
clock	0
reset	1
Q[2:0]	111
[2]	1
[1]	1
[0]	1
D0	1
D1	1
D2	0
not_Q0	0
mux1_1	1
clk_p	10000 ps