

A thick dark blue vertical bar is positioned on the left side of the page. From its base, several thin, curved lines in shades of blue and grey sweep upwards and to the right, creating an abstract, organic shape.

# PROIECT CIRCUITE INTEGRATE DIGITALE

PROIECTUL CU CERINTELE:

7-F-IV

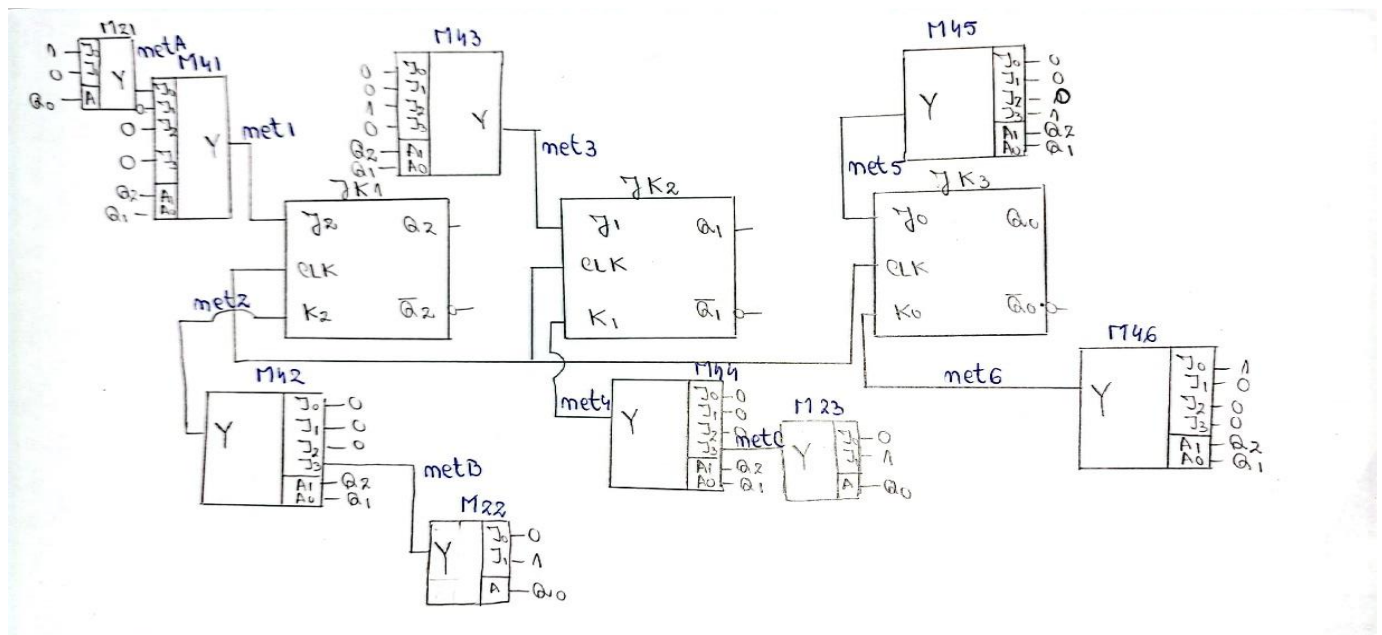
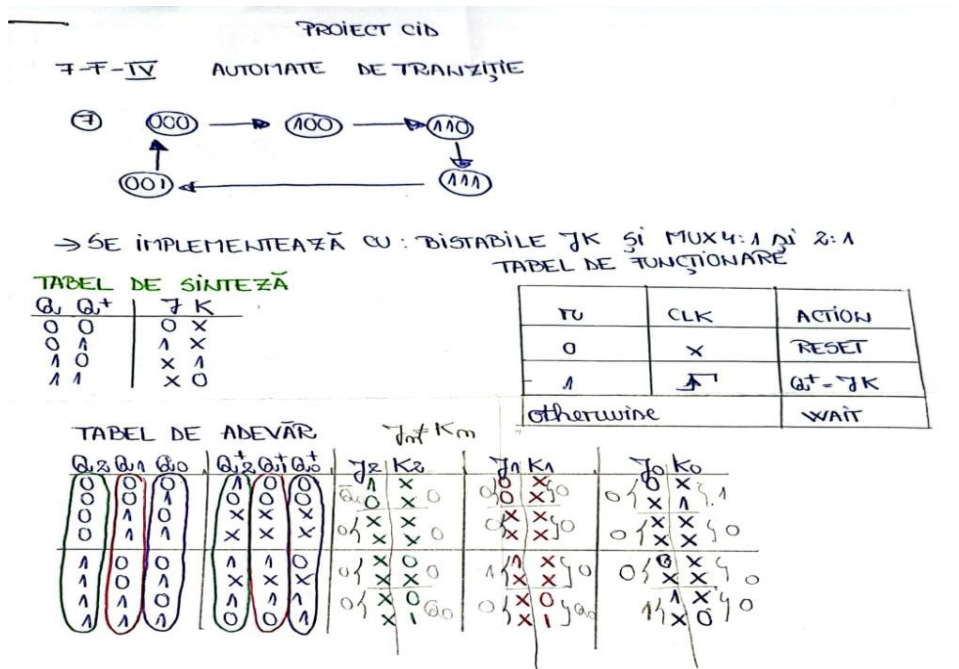
**Hentes Cristiana-Maria**

**Seria:A**

**Grupa: 2123**

**Semigrupa: 1**

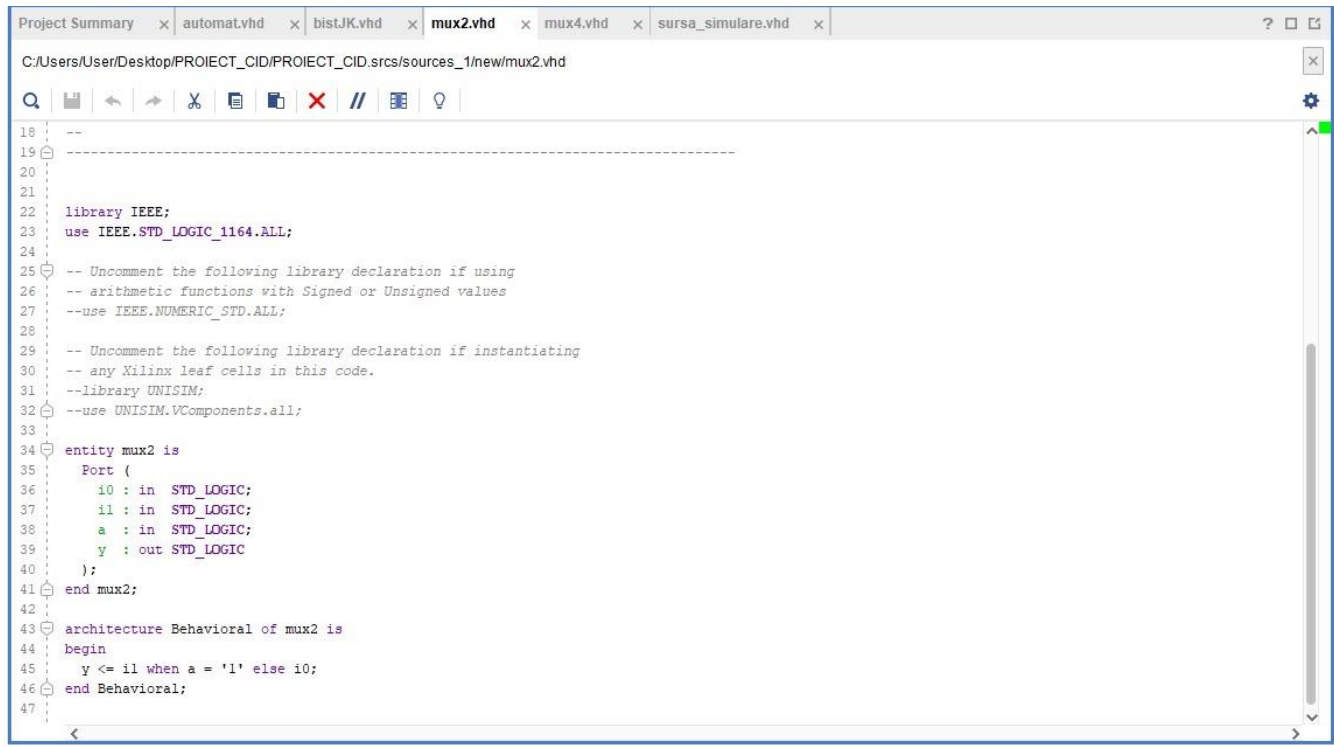
1. Rezolvarea pe hartie a automatului: bistabil JK cu urmatoarea diagrama de tranzitii:



Pentru rezolvarea automatului am folosit 3 bistabile JK, 6 MUX 4:1 si 3 MUX 2:1.

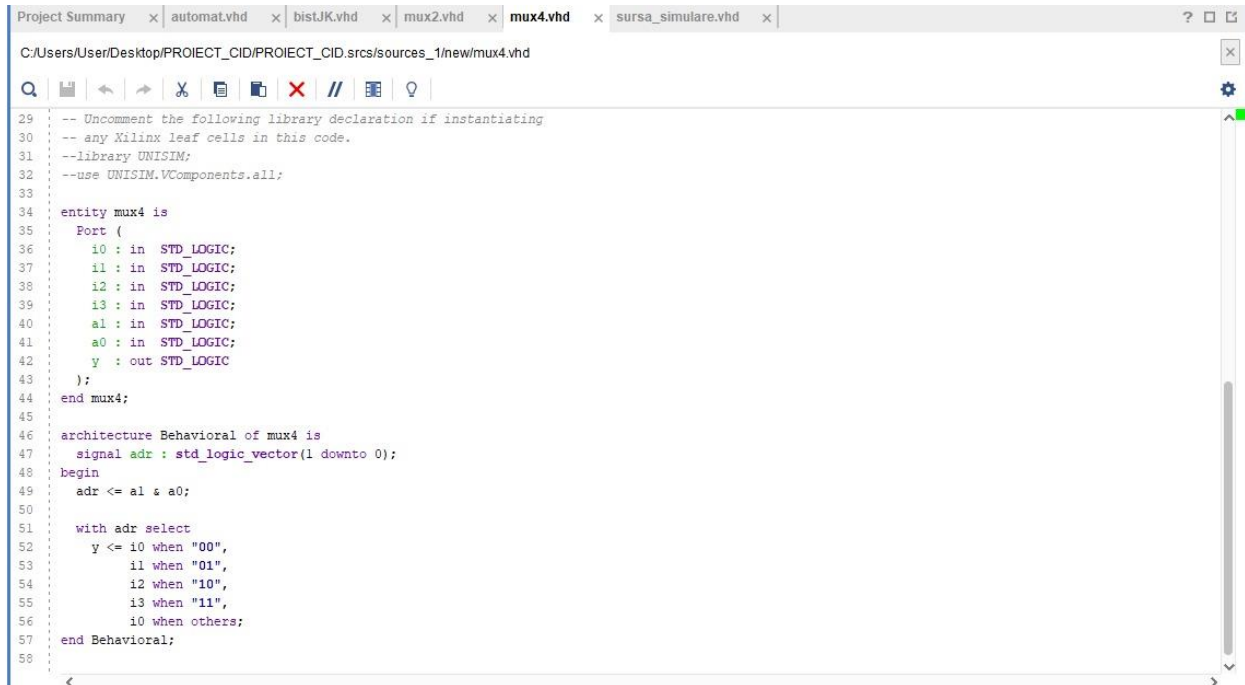
## 2. Circuitul combinational- format din MUX 4:1 SI MUX 2:1.

### MUX 2:1



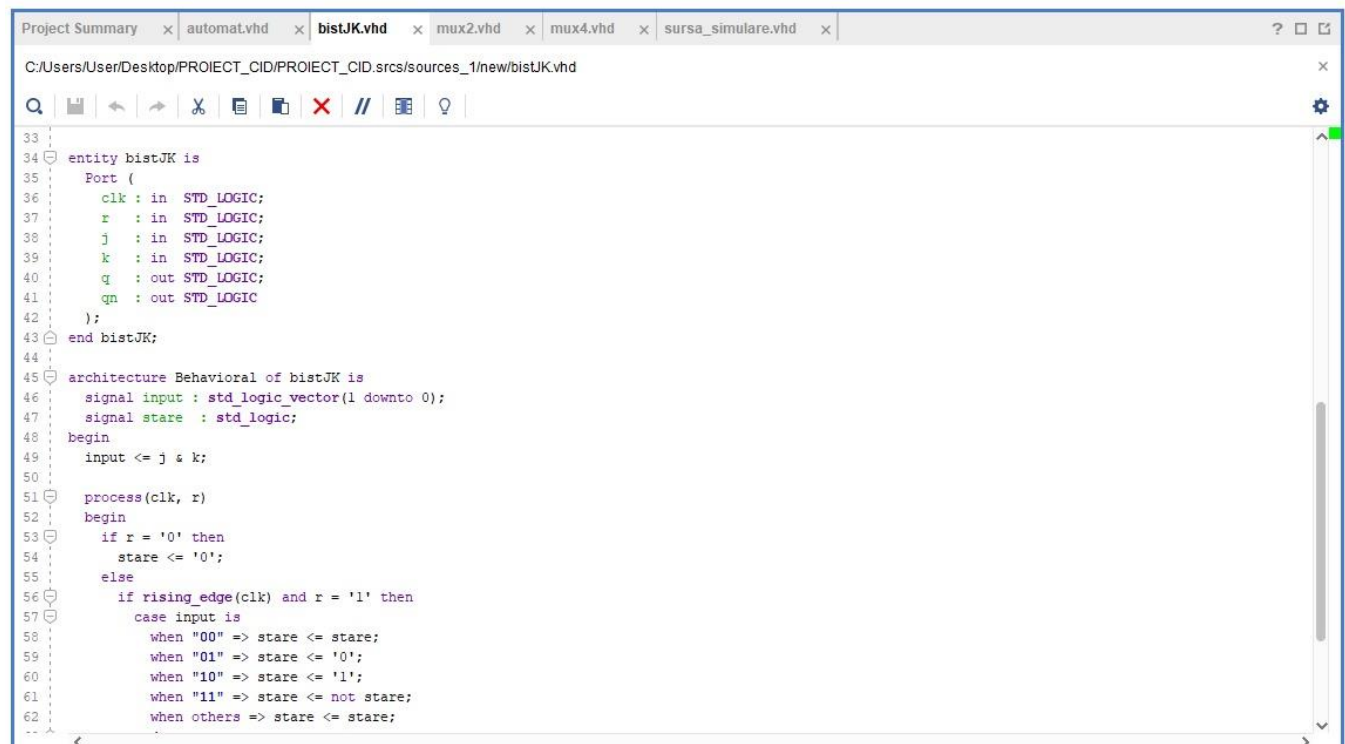
```
18  --
19  -----
20
21
22  library IEEE;
23  use IEEE.STD_LOGIC_1164.ALL;
24
25  -- Uncomment the following library declaration if using
26  -- arithmetic functions with Signed or Unsigned values
27  --use IEEE.NUMERIC_STD.ALL;
28
29  -- Uncomment the following library declaration if instantiating
30  -- any Xilinx leaf cells in this code.
31  --library UNISIM;
32  --use UNISIM.VComponents.all;
33
34  entity mux2 is
35      Port (
36          i0 : in  STD_LOGIC;
37          i1 : in  STD_LOGIC;
38          a  : in  STD_LOGIC;
39          y  : out STD_LOGIC
40      );
41  end mux2;
42
43  architecture Behavioral of mux2 is
44      begin
45          y <= i1 when a = '1' else i0;
46      end Behavioral;
47  
```

## MUX 4:1

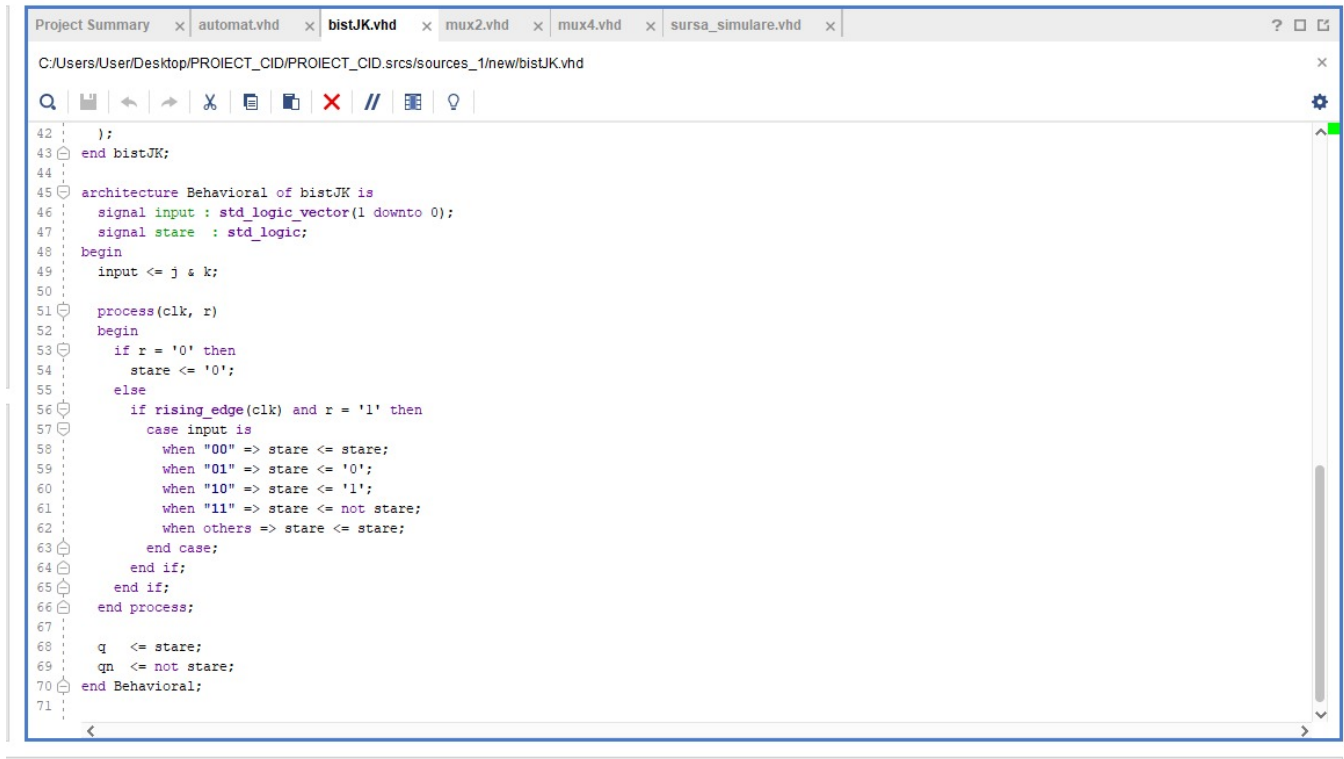


```
29 -- Uncomment the following library declaration if instantiating
30 -- any Xilinx leaf cells in this code.
31 --library UNISIM;
32 --use UNISIM.VComponents.all;
33
34 entity mux4 is
35     Port (
36         i0 : in  STD_LOGIC;
37         i1 : in  STD_LOGIC;
38         i2 : in  STD_LOGIC;
39         i3 : in  STD_LOGIC;
40         a1 : in  STD_LOGIC;
41         a0 : in  STD_LOGIC;
42         y  : out STD_LOGIC
43     );
44 end mux4;
45
46 architecture Behavioral of mux4 is
47     signal adr : std_logic_vector(1 downto 0);
48 begin
49     adr <= a1 & a0;
50
51     with adr select
52         y <= i0 when "00",
53             i1 when "01",
54             i2 when "10",
55             i3 when "11",
56             i0 when others;
57 end Behavioral;
58
```

## 3.Circuitul secvential- bistabil JK.



```
33
34 entity bistJK is
35     Port (
36         clk : in  STD_LOGIC;
37         r   : in  STD_LOGIC;
38         j   : in  STD_LOGIC;
39         k   : in  STD_LOGIC;
40         q   : out STD_LOGIC;
41         qn  : out STD_LOGIC
42     );
43 end bistJK;
44
45 architecture Behavioral of bistJK is
46     signal input : std_logic_vector(1 downto 0);
47     signal stare : std_logic;
48 begin
49     input <= j & k;
50
51     process(clk, r)
52     begin
53         if r = '0' then
54             stare <= '0';
55         else
56             if rising_edge(clk) and r = '1' then
57                 case input is
58                     when "00" => stare <= stare;
59                     when "01" => stare <= '0';
60                     when "10" => stare <= '1';
61                     when "11" => stare <= not stare;
62                     when others => stare <= stare;
63                 end case;
64             end if;
65         end if;
66     end process;
67
68     q <= stare;
69     qn <= not stare;
70 end Behavioral;
71
```



```
42     );
43 end bistJK;
44
45 architecture Behavioral of bistJK is
46     signal input : std_logic_vector(1 downto 0);
47     signal stare : std_logic;
48 begin
49     input <= j & k;
50
51     process(clk, r)
52     begin
53         if r = '0' then
54             stare <= '0';
55         else
56             if rising_edge(clk) and r = '1' then
57                 case input is
58                     when "00" => stare <= stare;
59                     when "01" => stare <= '0';
60                     when "10" => stare <= '1';
61                     when "11" => stare <= not stare;
62                     when others => stare <= stare;
63                 end case;
64             end if;
65         end if;
66     end process;
67
68     q <= stare;
69     qn <= not stare;
70 end Behavioral;
71
```

#### 4. Automatul de stare:

- am declarat in cadrul automatului de stare structurile de tip *entity* din cadrul MUX 4:1, MUX 2:1 si bistabilului JK ca si structuri de tip *component*;
- ca si porturi de intrare in cadrul automatului avem: semnalul de clk si r, iar ca port de iesire avem q;
- am definit cele 6 componente ale circuitului de tip MUX 4:1 si cele 3 de tip MUX 2:1;
- am declarat semnalele care leaga componentele intre ele (net-urile);

```

31 entity automat is
32   Port (
33     -- Inputs
34     clk : in  STD_LOGIC;
35     x   : in  STD_LOGIC;
36
37     -- Outputs
38     q   : out STD_LOGIC_VECTOR (2 downto 0)
39   );
40 end automat;
41
42 architecture Behavioral of automat is
43   -- Declare components
44   component bistJK is
45     Port (
46       -- Inputs
47       clk : in  STD_LOGIC;
48       x   : in  STD_LOGIC;
49       j   : in  STD_LOGIC;
50       k   : in  STD_LOGIC;
51
52       -- Outputs
53       q   : out STD_LOGIC;
54       qn  : out STD_LOGIC
55     );
56   end component bistJK;
57
58   component mux4 is
59     Port (
60       -- Inputs

```

```

57
58 component mux4 is
59   Port (
60     -- Inputs
61     i0 : in  STD_LOGIC;
62     i1 : in  STD_LOGIC;
63     i2 : in  STD_LOGIC;
64     i3 : in  STD_LOGIC;
65     a1 : in  STD_LOGIC;
66     a0 : in  STD_LOGIC;
67
68     -- Outputs
69     y : out STD_LOGIC
70   );
71 end component mux4;
72
73 component mux2 is
74   Port (
75     -- Inputs
76     i0 : in  STD_LOGIC;
77     i1 : in  STD_LOGIC;
78     a  : in  STD_LOGIC;
79
80     -- Outputs
81     y : out STD_LOGIC
82   );
83 end component mux2;
84
85 -- Declare internal signals
86 signal netA, netB, netC, net1, net2, net3, net4, net5, net6: std_logic;

```

```
Project Summary x automat.vhd x bistJK.vhd x mux2.vhd x mux4.vhd x sursa_simulare.vhd x
C:/Users/User/Desktop/PROIECT_CID/PROIECT_CID/srcs/sources_1/new/automat.vhd

85 -- Declare internal signals
86 signal netA, netB, netC, net1, net2, net3, net4, net5, net6: std_logic;
87 signal qint: std_logic_vector(2 downto 0);
88 begin
89 -- Map output to internal signal
90 q <= qint;
91
92 -- Instantiate bistJK components
93 Bistjk1: bistJK port map (
94   clk => clk,
95   r  => r,
96   j  => net1,
97   k  => net2,
98   q  => qint(2)
99 );
100
101 Bistjk2: bistJK port map (
102   clk => clk,
103   r  => r,
104   j  => net3,
105   k  => net4,
106   q  => qint(1)
107 );
108
109 Bistjk3: bistJK port map (
110   clk => clk,
111   r  => r,
112   j  => net5,
113   k  => net6,
114   q  => qint(0)
115 );
116
```

```
Project Summary x automat.vhd x bistJK.vhd x mux2.vhd x mux4.vhd x sursa_simulare.vhd x
C:/Users/User/Desktop/PROIECT_CID/PROIECT_CID/srcs/sources_1/new/automat.vhd

117
118 M1: mux4 port map (
119   i0 => netA,
120   i1 => '0',
121   i2 => '0',
122   i3 => '0',
123   a1 => qint(2),
124   a0 => qint(1),
125   y  => net1
126 );
127
128 M2: mux4 port map (
129   i0 => '0',
130   i1 => '0',
131   i2 => '0',
132   i3 => netB,
133   a1 => qint(2),
134   a0 => qint(1),
135   y  => net2
136 );
137
138 M3: mux4 port map (
139   i0 => '0',
140   i1 => '0',
141   i2 => '1',
142   i3 => '0',
143   a1 => qint(2),
144   a0 => qint(1),
145   y  => net3
146 );
147
```

```
147
148 M4: mux4 port map (
149     10 => '0',
150     11 => '0',
151     12 => '0',
152     13 => netC,
153     a1 => qint(2),
154     a0 => qint(1),
155     y => net4
156 );
157
158 M5: mux4 port map (
159     10 => '0',
160     11 => '0',
161     12 => '0',
162     13 => '1',
163     a1 => qint(2),
164     a0 => qint(1),
165     y => net5
166 );
167
168 M6: mux4 port map (
169     10 => '1',
170     11 => '0',
171     12 => '0',
172     13 => '0',
173     a1 => qint(2),
174     a0 => qint(1),
175     y => net6
176 );
---
```

```
167
168 M6: mux4 port map (
169     10 => '1',
170     11 => '0',
171     12 => '0',
172     13 => '0',
173     a1 => qint(2),
174     a0 => qint(1),
175     y => net6
176 );
177
178 muxx21: mux2 port map (
179     10 => '1',
180     11 => '0',
181     a => qint(0),
182     y => netA
183 );
184
185 muxx22: mux2 port map (
186     10 => '0',
187     11 => '1',
188     a => qint(0),
189     y => netB
190 );
191
192 muxx23: mux2 port map (
193     10 => '0',
194     11 => '1',
195     a => qint(0),
196     y => netC
---
```



## 5.Sursa de simulare

```
Project Summary x automat.vhd x bistJK.vhd x mux2.vhd x mux4.vhd x sursa_simulare.vhd x
C:/Users/User/Desktop/PROIECT_CID/PROIECT_CID.srscs/sim_1/new/sursa_simulare.vhd

33
34 entity sursa_simulare is
35     -- Port ( );
36 end sursa_simulare;
37
38 architecture Behavioral of sursa_simulare is
39     component automat is
40     Port (
41         clk : in  STD_LOGIC;
42         r   : in  STD_LOGIC;
43         q   : out STD_LOGIC_VECTOR (2 downto 0)
44     );
45 end component automat;
46
47 signal clk, r : std_logic;
48 signal q      : std_logic_vector(2 downto 0);
49 begin
50     UUT: automat port map (clk, r, q);
51
52     process
53     begin
54
55         clk <= '0';
56         wait for 1.2 ns;
57         clk <= '1';
58         wait for 1.2 ns;
59     end process;
60
61     r <= '0' after 0 ns, '1' after 2.2 ns;
62 end Behavioral;
--
```

## Functionalitate:

