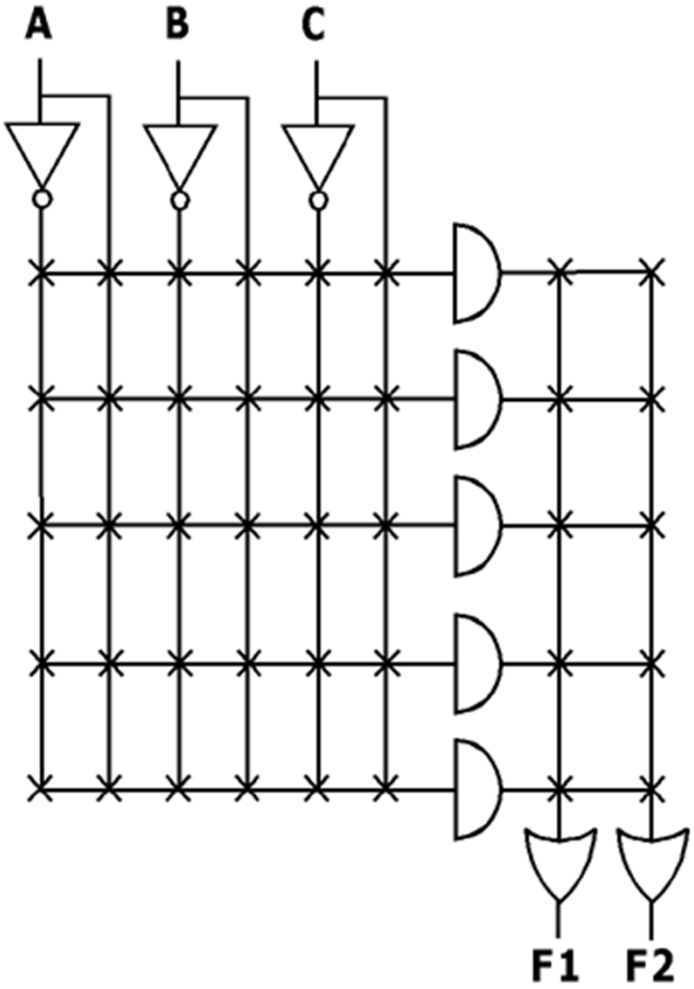
Digitalne strukture
7. Vaja

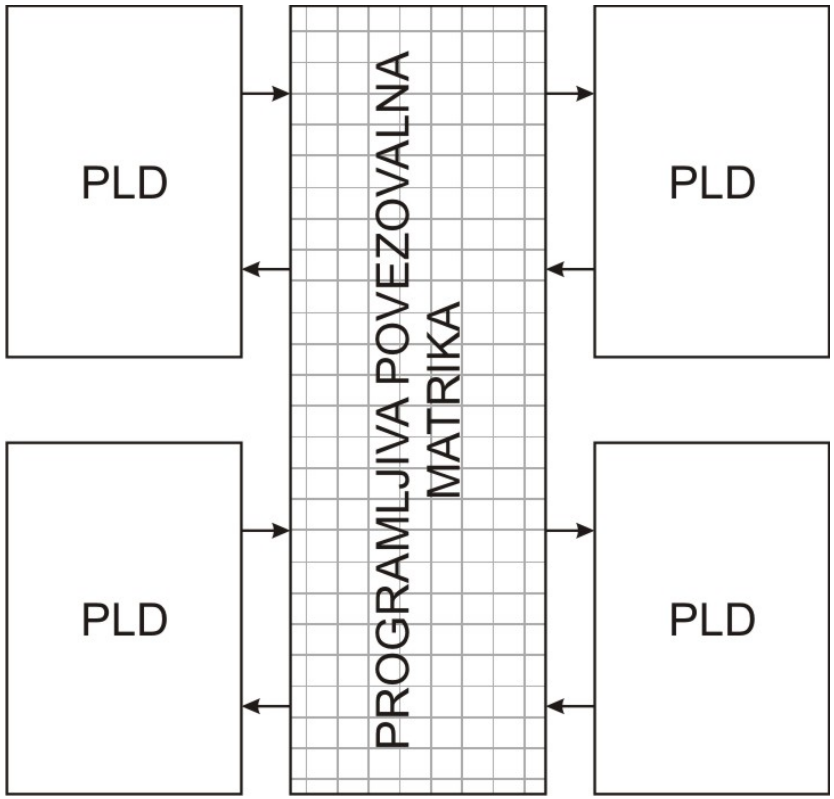
Realizacija sinhronskih sekvenčnih vezij s programirljivim vezjem FPGA

PLD – Programmable Logic Device

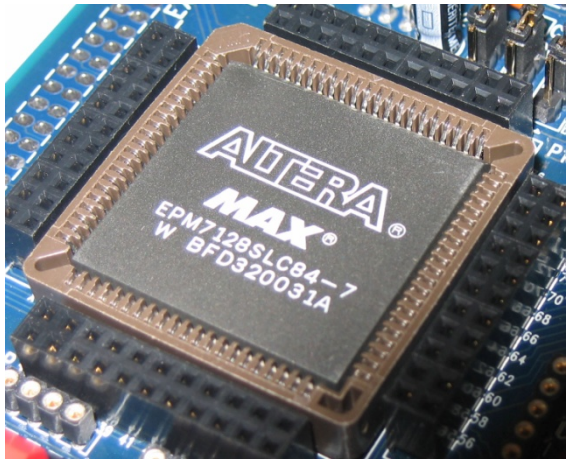


- PROM, EPROM, PLA, PAL, GAL
- Stiki povezav določajo končno funkcijo vezja
- Površina vezja se veča s kvadratom št. vhodov
- Nekaj 100 logičnih vrat

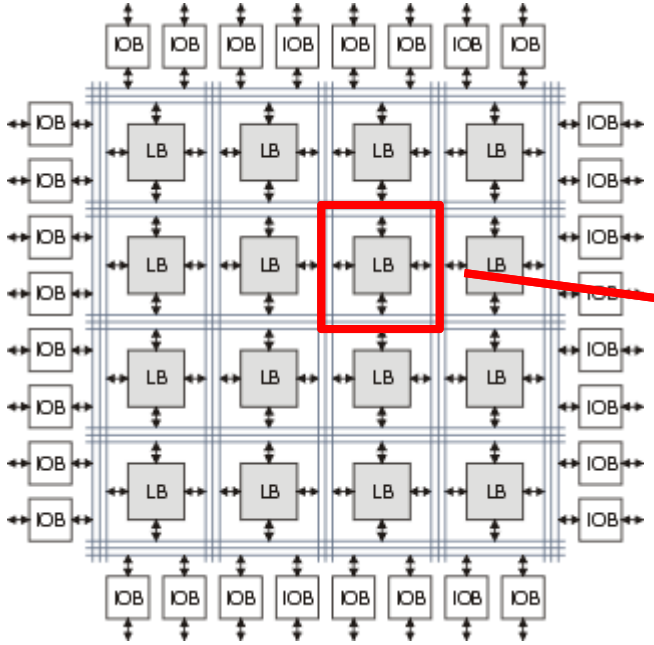
CPLD – Complex Programmable Logic Device



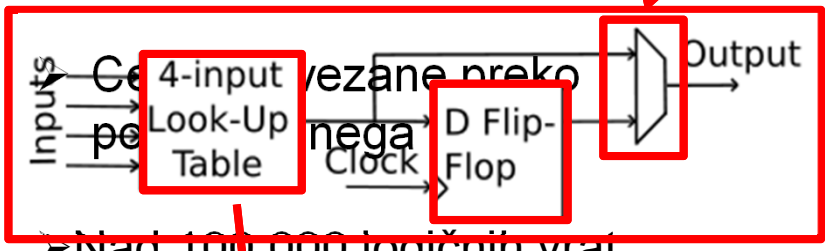
- Več blokov PLD povezanih s programljivo povezovalno matriko
- Izvedba kompleksnih logičnih funkcij
- Od 1000 do 100.000 logičnih vrat



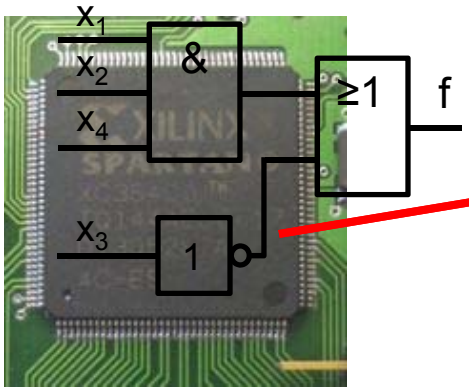
FPGA – Field Programmable Gate Array



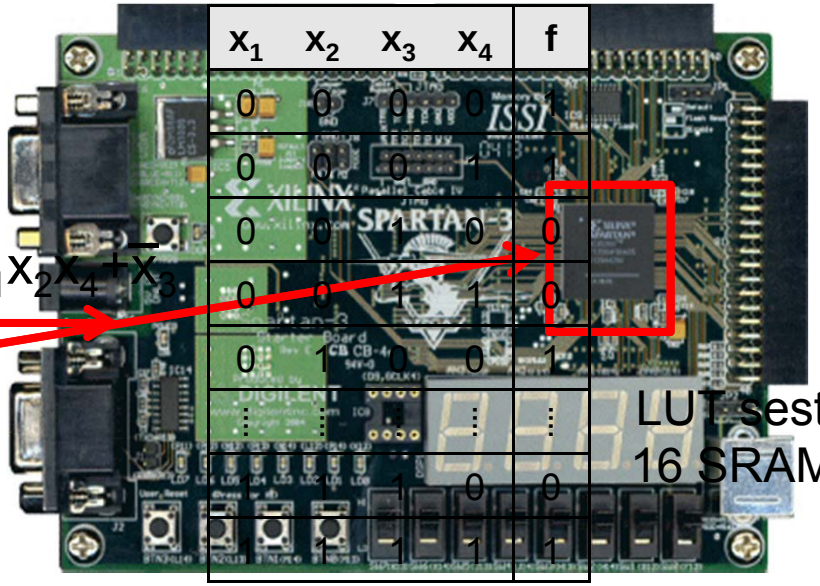
- Matrika logičnih celic (LB) obkroženih z vhodno/izhodnimi celicami



- Ce vezane preko po nega
- Nad 100.000 logičnih vrat



$f = x_1 x_2 x_4 + \overline{x_3}$

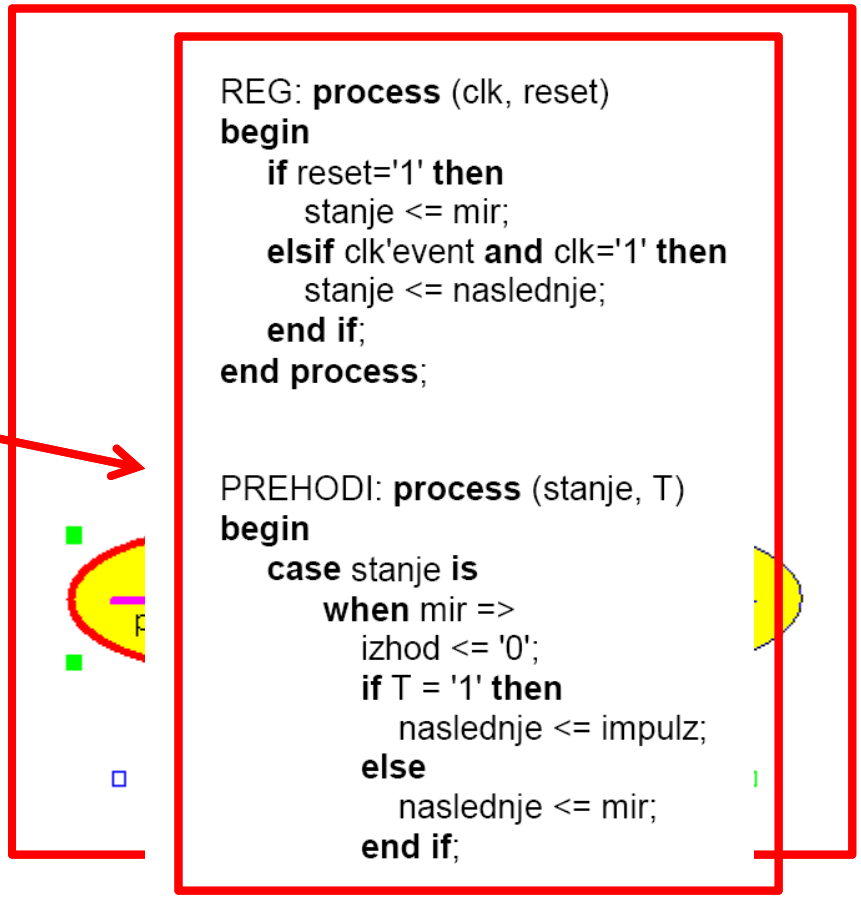


LUT sestavlja 16 SRAM celic

FPGA – Field Programmable Gate Array

NIVOJI MODELIRANJA

- Tranzistorji
- Logična vrata
- Nivo registrov
- Grafični modeli
- Obnašanje vezja
- Sistemski modeli



```
REG: process (clk, reset)
begin
    if reset='1' then
        stanje <= mir;
    elsif clk'event and clk='1' then
        stanje <= naslednje;
    end if;
end process;


PREHODI: process (stanje, T)
begin
    case stanje is
        when mir =>
            izhod <= '0';
            if T = '1' then
                naslednje <= impulz;
            else
                naslednje <= mir;
            end if;
```

VHDL koda

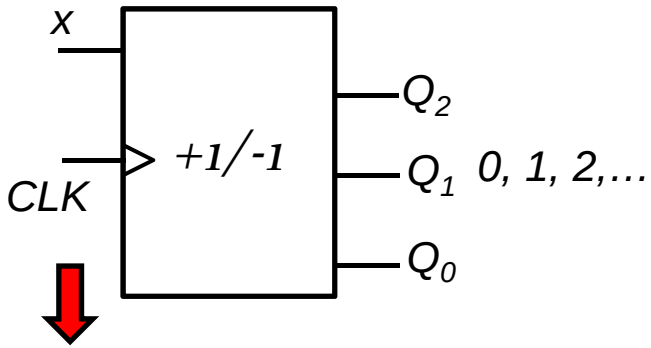
Naloge

7a: Z jezikom VHDL zapišite delovanje sinhronskega števca po modulu 5.

Uporabite stavka *process* in *if… then…else*

```
process (spremenljivke)
begin
:
sekvenčni stavki;
:
end process;
```

```
if izraz 1 then
stavki;
elsif izraz 2 then
stavki;
:
else izraz n;
end if;
```



Ukaz za proženje števca ob uri:

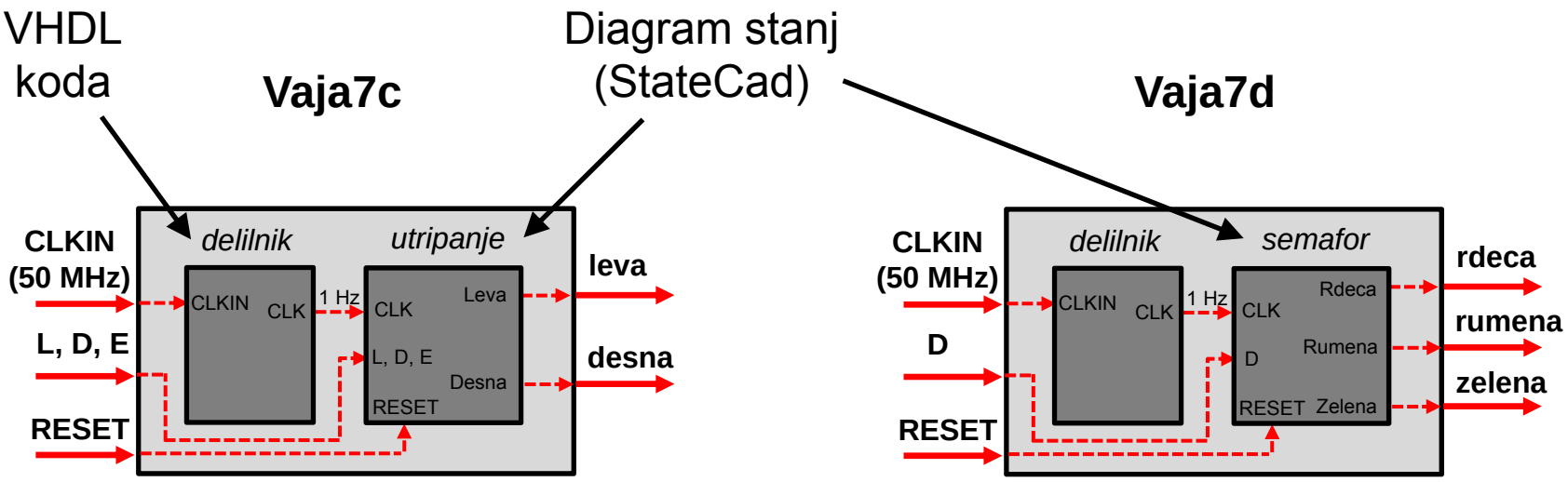
```
if (clk'event and clk='1') then
stavki;
end if;
```

Naloge

7b: Zapišite VHDL kodo za vezje, ki bo frekvenco CLK na plošči Xilinx zmanjšalo iz 50 MHz na 1 Hz.

7c: Zgradite sinhronsko sekvenčno vezje, ki bo vklapljalo smernike pri avtomobilu.

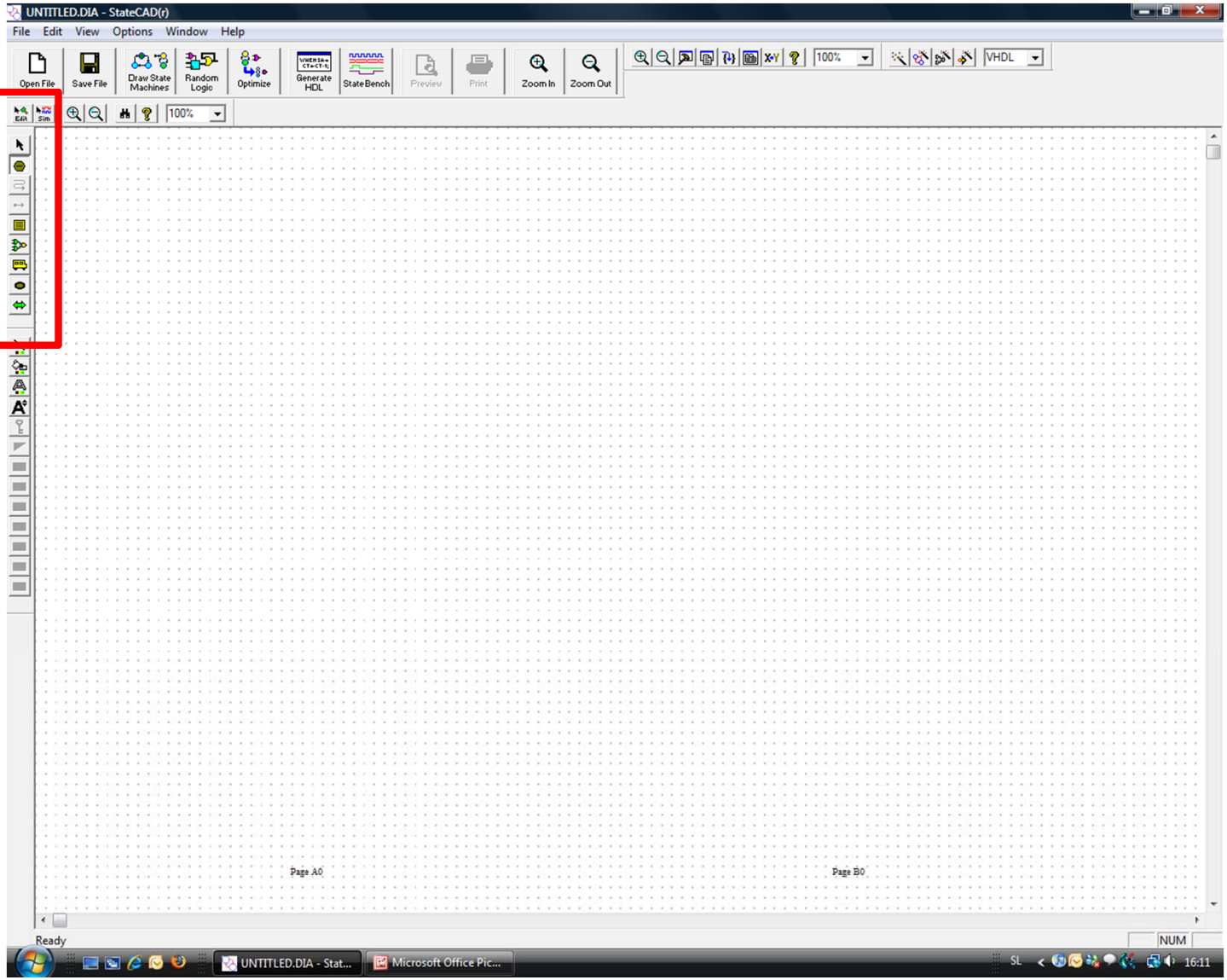
7d: Zgradite vezje, ki bo simuliralo delovanje semaforja.



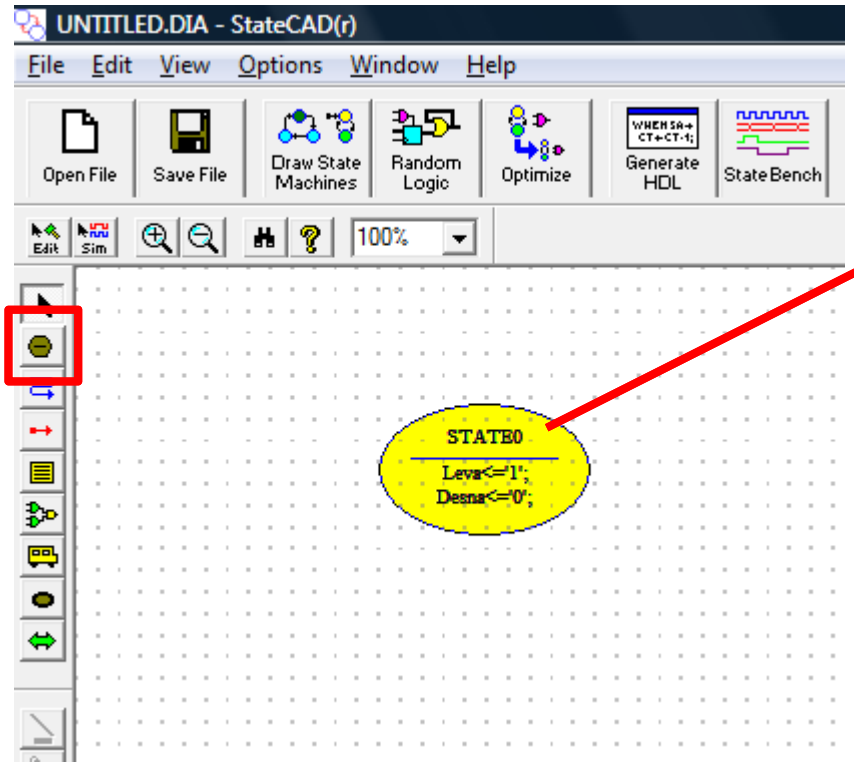
Vhodi: CLKIN, RESET, L, D, E
Izhodi: leva, desna

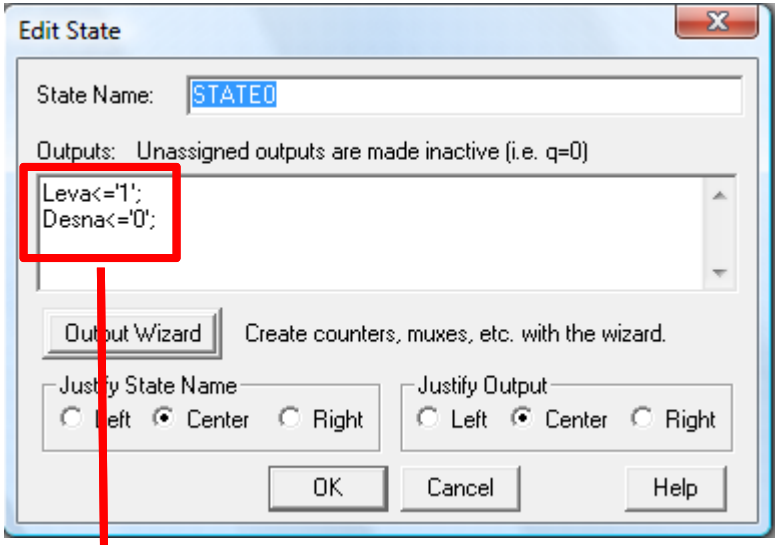
Vhodi: CLKIN, RESET, D
Izhodi: rdeca, rumena, zelena

StateCAD


UNTITLED.DIA - StateCAD(r)
File Edit View Options Window Help
Open File
Save File
Draw State Machines
Random Logic
Optimize
Generate HDL
State Bench
Preview
Print
Zoom In
Zoom Out
100%
VHDL
Page A0
Page B0
Ready
NUM
UNTITLED.DIA - Stat...
Microsoft Office Pic...
SL
16:11

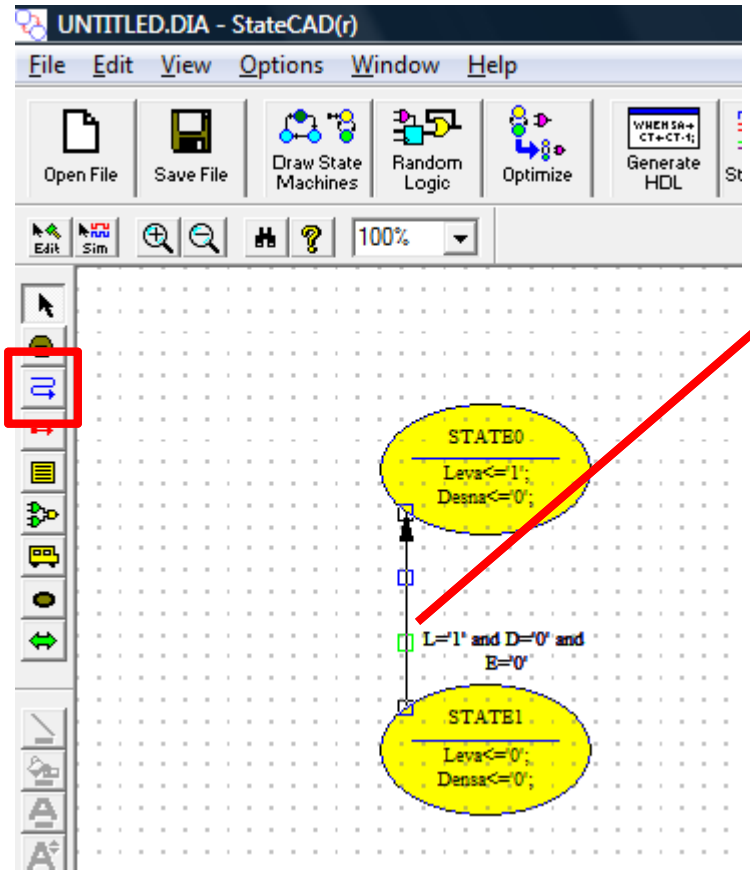
StateCAD – Definiranje stanj

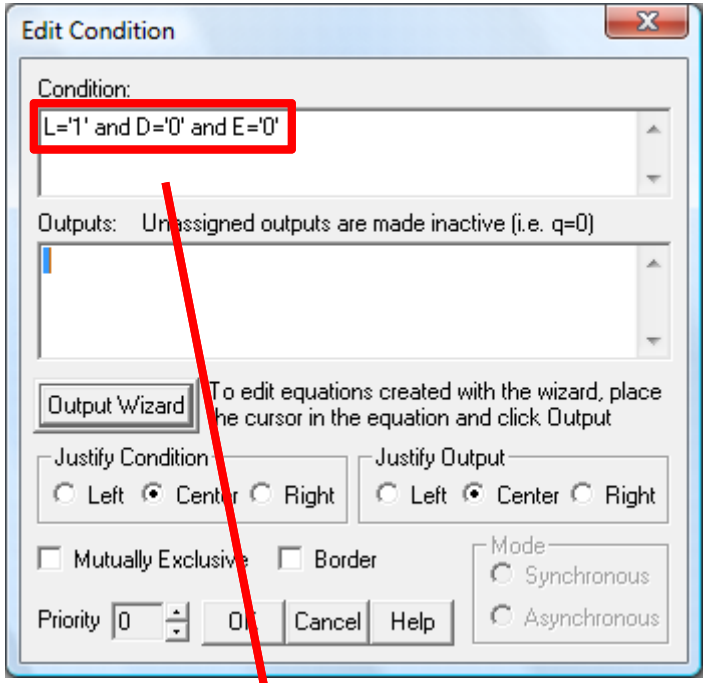




Leva <= ‘1’;
Desna <= ‘0’;

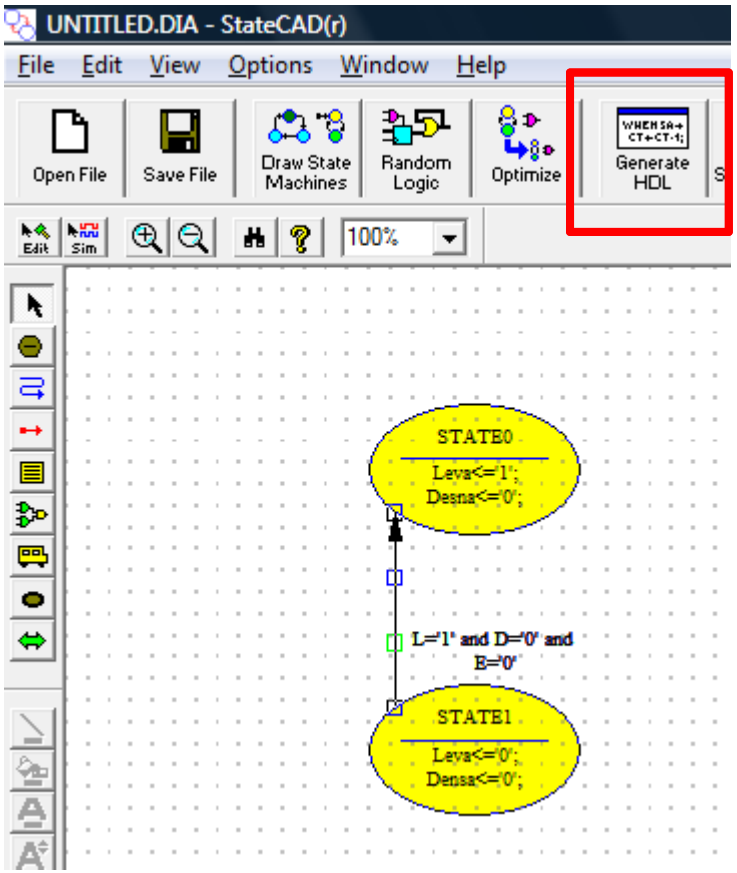
StateCAD – Definiranje prehodov





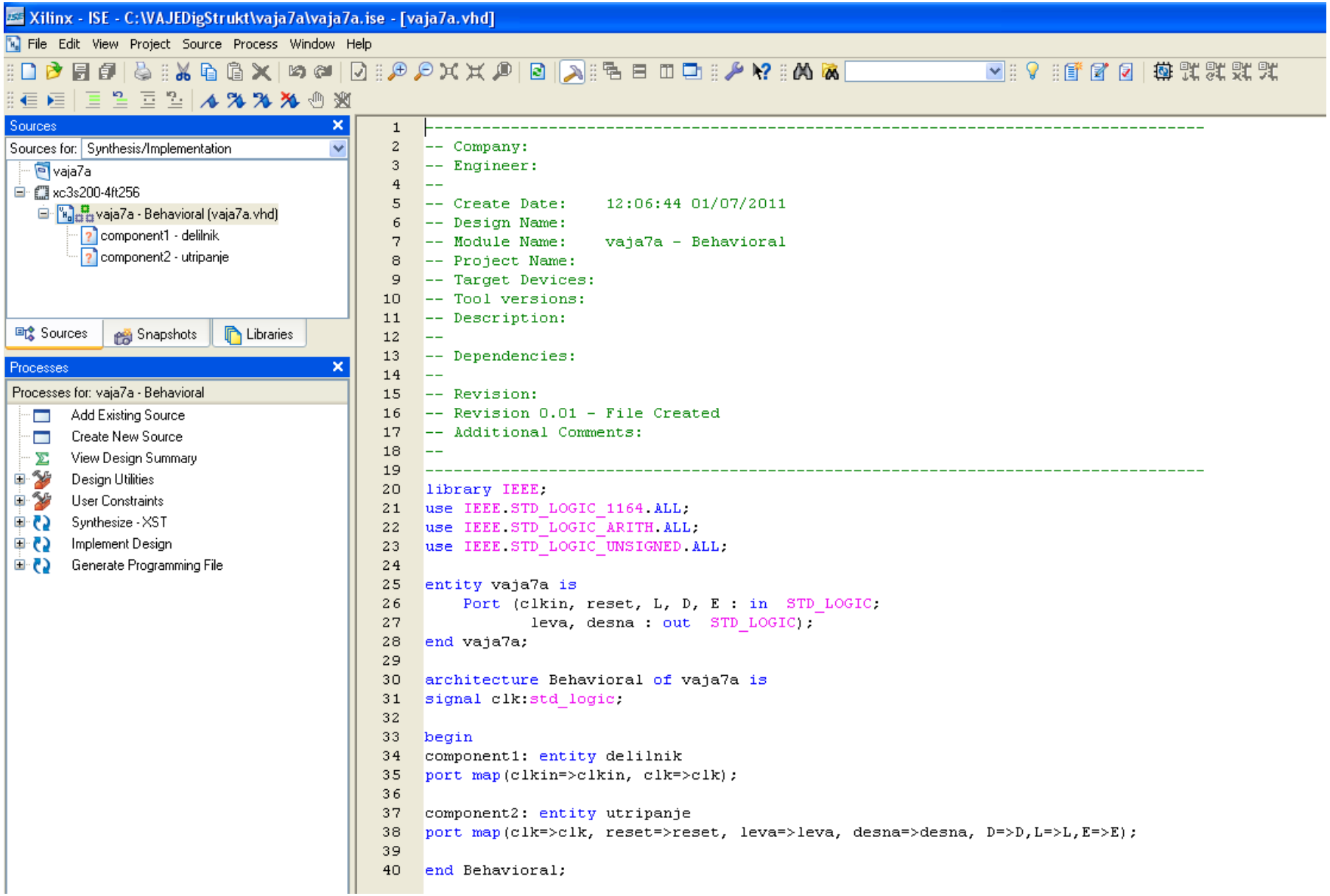
L=‘1’ and D=‘0’ and E=‘0’

StateCAD – Generiranje VHDL kode



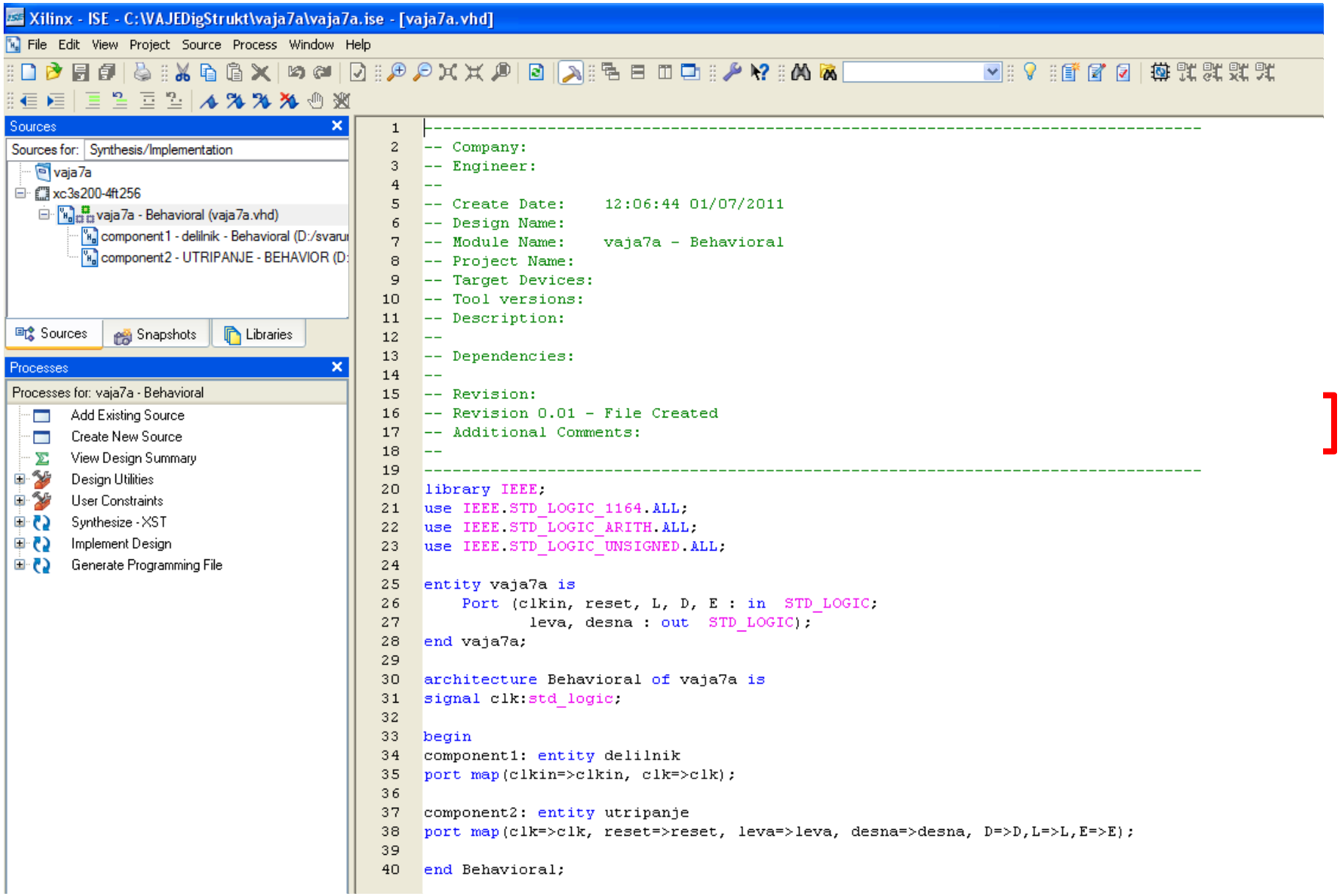
Ko končamo z risanjem diagrama stanj, ga pretvorimo v VHDL kodo.

Xilinx ISE-predloga (Vaja7c in Vaja7d)




```
1  ----------------------------------------------------------------------------------
2  -- Company:
3  -- Engineer:
4  --
5  -- Create Date:    12:06:44 01/07/2011
6  -- Design Name:
7  -- Module Name:    vaja7a - Behavioral
8  -- Project Name:
9  -- Target Devices:
10 -- Tool versions:
11 -- Description:
12 --
13 -- Dependencies:
14 --
15 -- Revision:
16 -- Revision 0.01 - File Created
17 -- Additional Comments:
18 --
19 ----------------------------------------------------------------------------------
20 library IEEE;
21 use IEEE.STD_LOGIC_1164.ALL;
22 use IEEE.STD_LOGIC_ARITH.ALL;
23 use IEEE.STD_LOGIC_UNSIGNED.ALL;
24
25 entity vaja7a is
26     Port (clkin, reset, L, D, E : in  STD_LOGIC;
27           leva, desna : out  STD_LOGIC);
28 end vaja7a;
29
30 architecture Behavioral of vaja7a is
31 signal clk:std_logic;
32
33 begin
34 component1: entity delilnik
35 port map(clkin=>clkin, clk=>clk);
36
37 component2: entity utripanje
38 port map(clk=>clk, reset=>reset, leva=>leva, desna=>desna, D=>D,L=>L,E=>E);
39
40 end Behavioral;
```

Xilinx ISE-predloga (Vaja7c in Vaja7d)



Za to vajo ni potrebno izdelati poročila.