

Архитектура Xilinx Spartan FPGA развојног окружења

FPGA

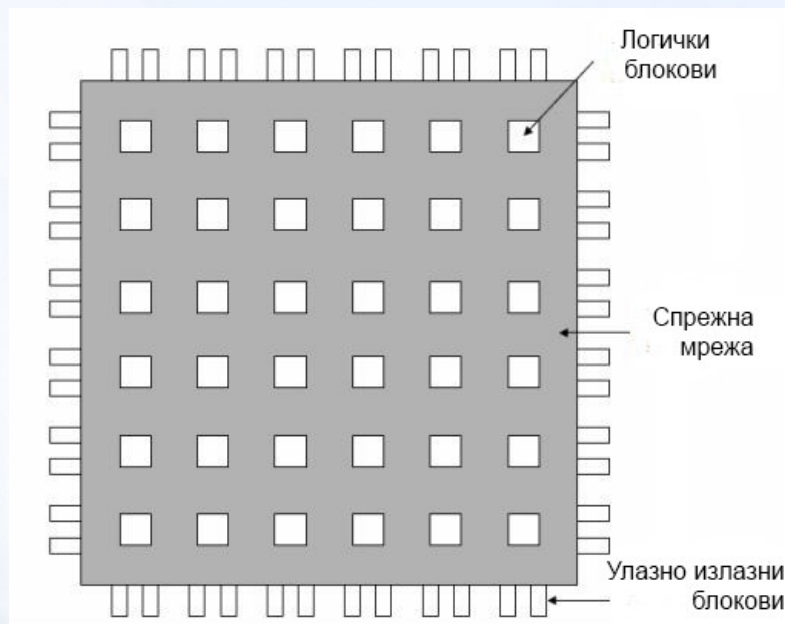
FPGA (Field-Programmable Gate Array) представља интегрисано коло чија се унутрашња структура може конфигурисати од стране крајњег корисника.

Дефинисање унутрашње структуре FPGA компоненте се врши уз помоћ HDL језика или шематских дијаграма. У зависности од начина на који су имплементирани, нека FPGA кола могу да се програмирају само једанпут, а друга могу бити репрограмирана изнова и изнова.

Основна архитектура

Основну архитектуру FPGA-ова чине следеће три компоненте:

- логички блокови
- спрежна мрежа
- улазно излазни блокови

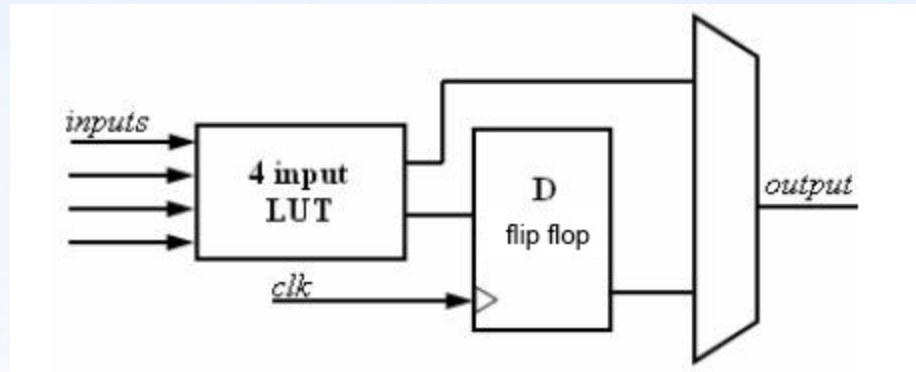


- Три основна типа програмабилних елемената код FPGA кола су RAM, анти-осигурачи (anti-fuses), и флеш EPROM програмска меморија.
- Логички блокови, често називани логички елементи (LE), или комбинациони логички блокови (CLB), чине логику FPGA кола.
- Спрега између CLB-ова се остварује коришћењем програмабилне спрежне мреже, тј програмабилне архитектуре за рутирање.

- Улазно-излазни блокови, названи U/I блокови, остварују везу са спољним светом, а могу се програмирати да буду улазни или излазни. FPGA кола се карактеришу ситно-зрнастом (fine-grain) архитектуром.
- То значи да се логичке операције обављају на нивоу бита или на нивоу речи малог капацитета (< 4 бита), што омогућава креирање веома флексибилних архитектура прилагођених специфичним захтевима апликације.

FPGA блок

Класичан FPGA блок има 4 улаза у логички део, по један Д флип-флоп и мултиплексер.



Постоји само један излаз, који може бити синхронизован са тактом или се може директно водити са излаза из логичког блока. Пошто је CLK сигнал од посебне важности и како је његова улога углавном стандардна за свако коло, он се блоку придружује као посебан сигнал (одвојен од осталих улаза).

Произвођачи FPGA

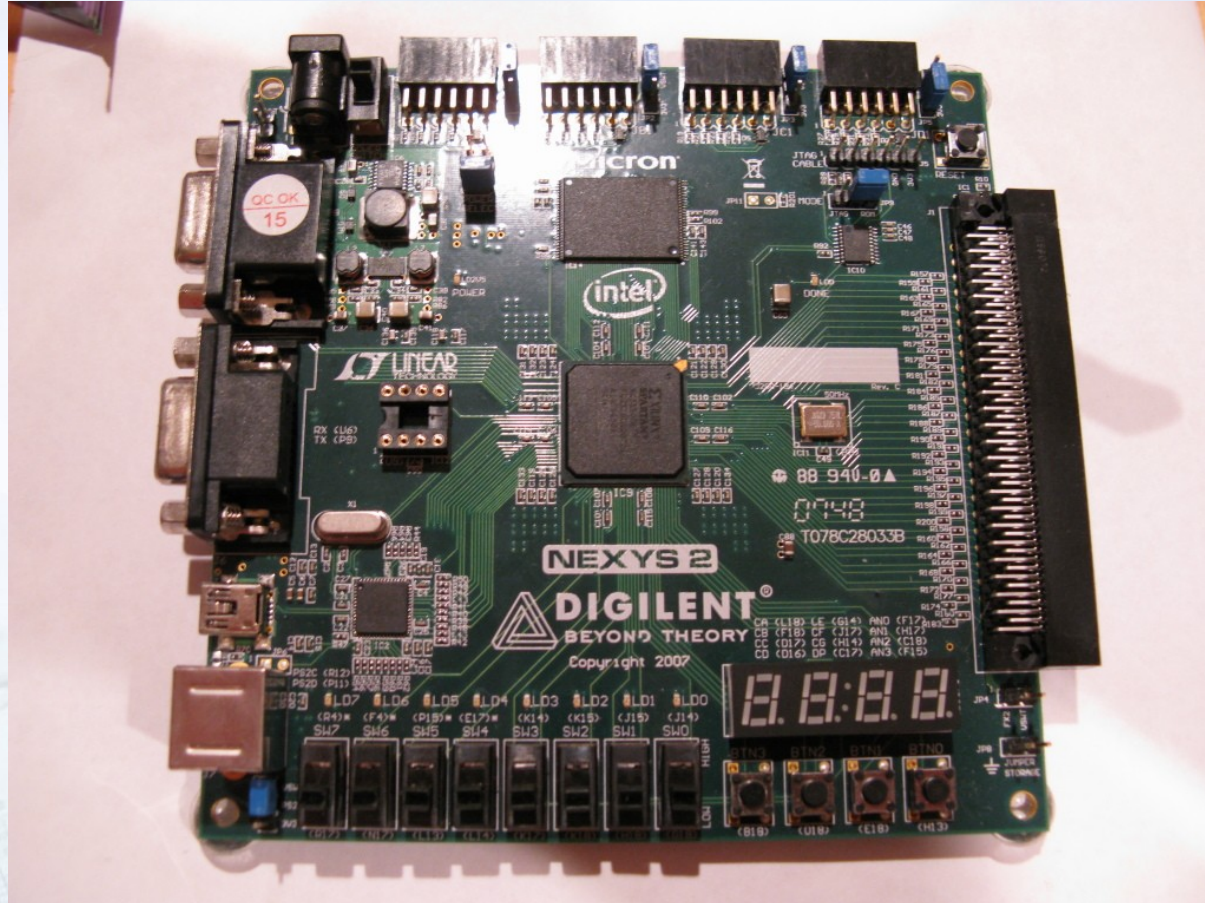
Тренутно главне произвођаче FPGA компоненти на тржишту чине компаније Xilinx и Altera. Заједно контролишу преко 80% тржишта, а сам Xilinx преко 50% читавог тржишта.

Компаније које такође производе FPGA компоненте су: Lattice Semiconductor, Actel, Atmel, SiliconBlue Technologies, QuickLogic и Achronix.

Nexys 2 плоча

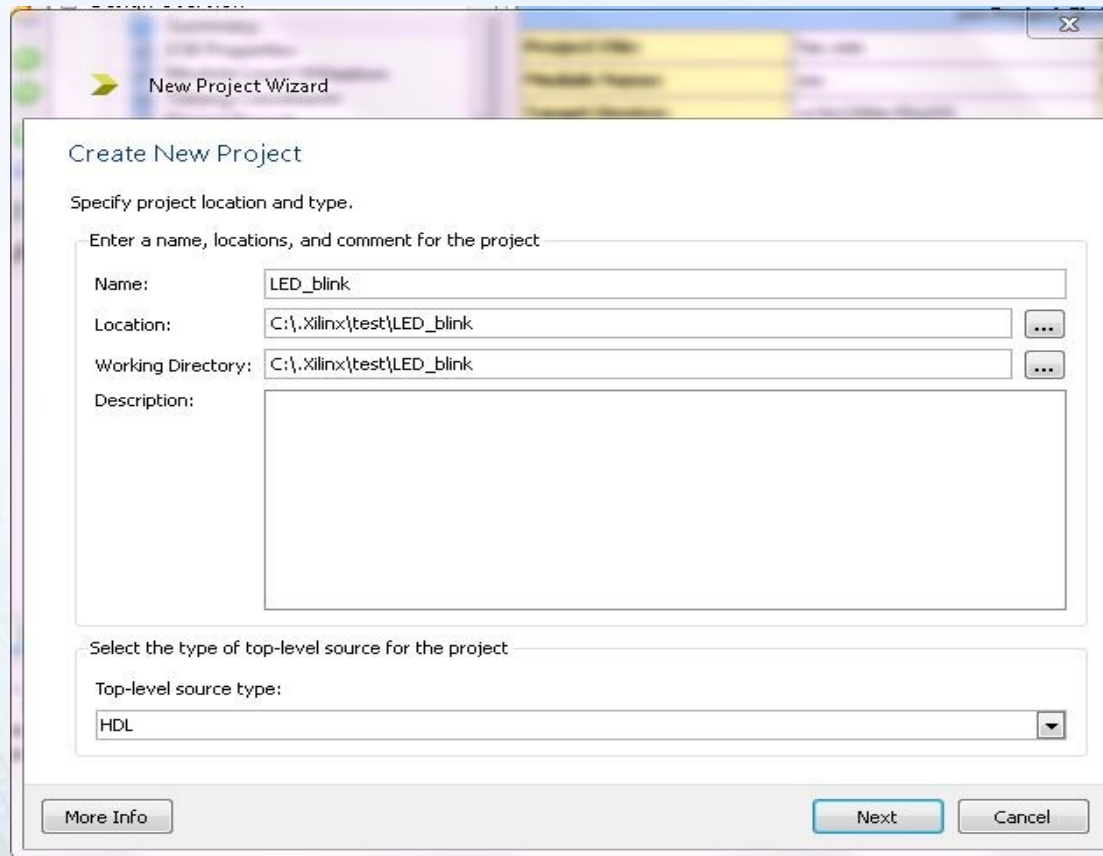
- USB2 (high speed)
- 16 Mbytes sd RAM
- 16 Mbytes flash ROM
- 50MHz осцилатор плус подножје за секундарни осцилатор
- 8 LED
- 4 тастера
- 8 прекидача
- PS/2 порт
- VGA порт
- Серијски порт
- 4 12-топинских pmod конектора
- Hirose FX2 socket connector
- Четвороцифарни седмо-сегментни дисплеј

Nexys 2 плата



Реализација пројекта на Nexys2 Xilinx Spartan 3 FPGA платформи

File => New Project



The screenshot shows the 'New Project Wizard' dialog box in Xilinx ISE. The title bar says 'New Project Wizard'. The main heading is 'Create New Project'. Below it, the instruction 'Specify project location and type.' is displayed. A sub-instruction reads 'Enter a name, locations, and comment for the project'. There are four input fields: 'Name:' with the text 'LED_blink', 'Location:' with 'C:\.Xilinx\test\LED_blink', 'Working Directory:' with 'C:\.Xilinx\test\LED_blink', and a larger 'Description:' text area. Below these fields, another instruction says 'Select the type of top-level source for the project'. A dropdown menu for 'Top-level source type:' is set to 'HDL'. At the bottom, there are three buttons: 'More Info', 'Next', and 'Cancel'.

New Project Wizard

Create New Project

Specify project location and type.

Enter a name, locations, and comment for the project

Name: LED_blink

Location: C:\.Xilinx\test\LED_blink

Working Directory: C:\.Xilinx\test\LED_blink

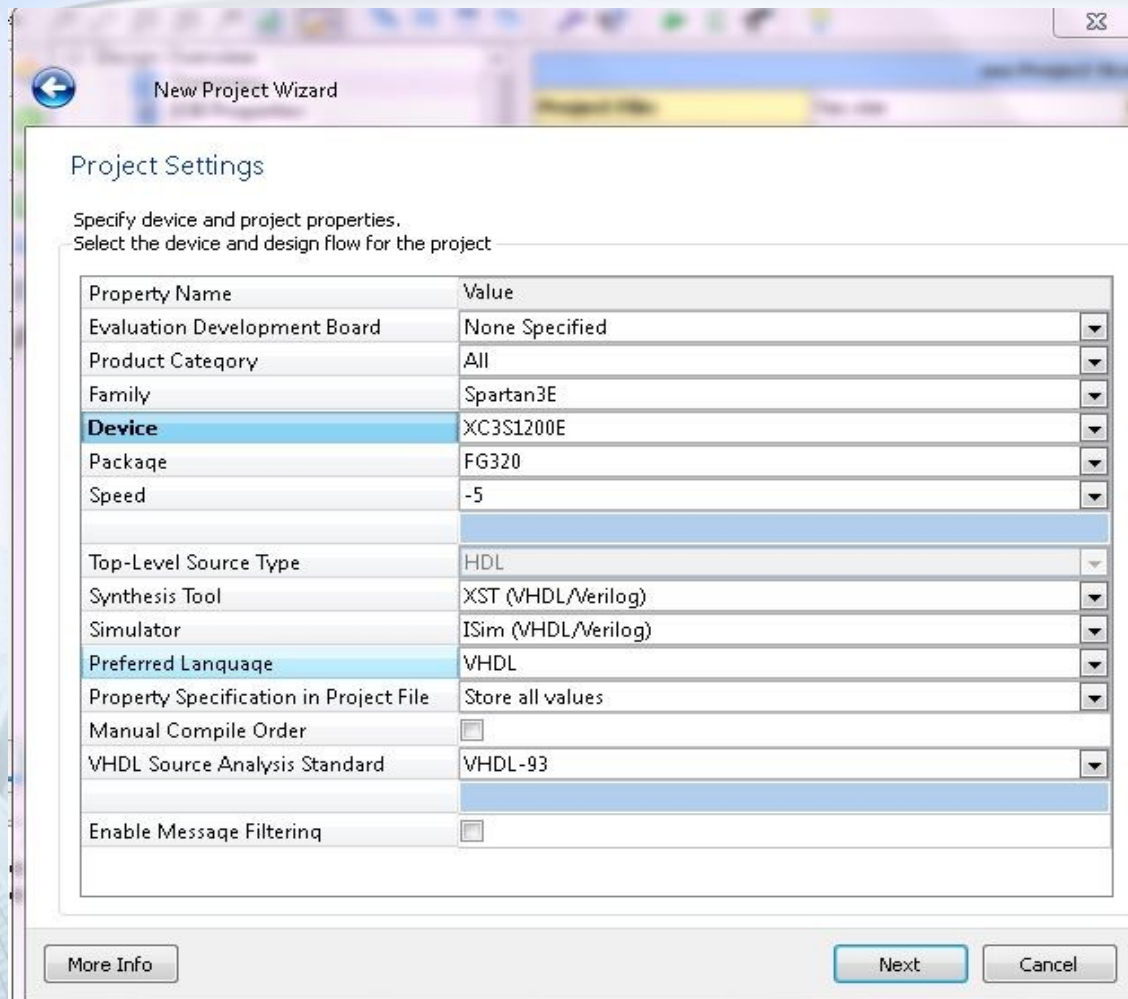
Description:

Select the type of top-level source for the project

Top-level source type: HDL

More Info Next Cancel

Одабир параметара



The image shows a screenshot of the 'New Project Wizard' dialog box in a software development environment. The 'Project Settings' tab is active, displaying a list of project properties. The 'Device' property is highlighted in blue. The background of the slide features a faint image of a computer monitor and keyboard.

New Project Wizard

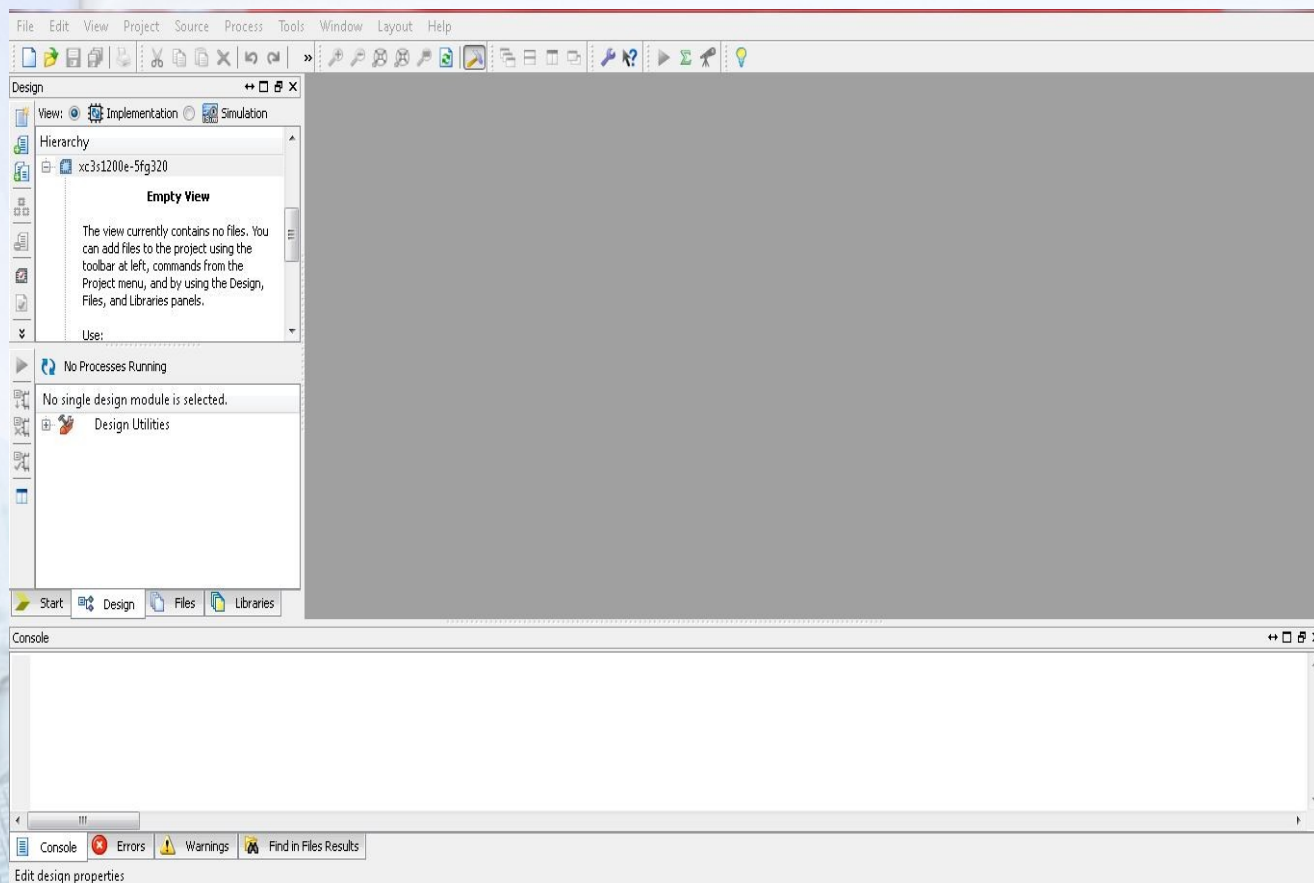
Project Settings

Specify device and project properties.
Select the device and design flow for the project

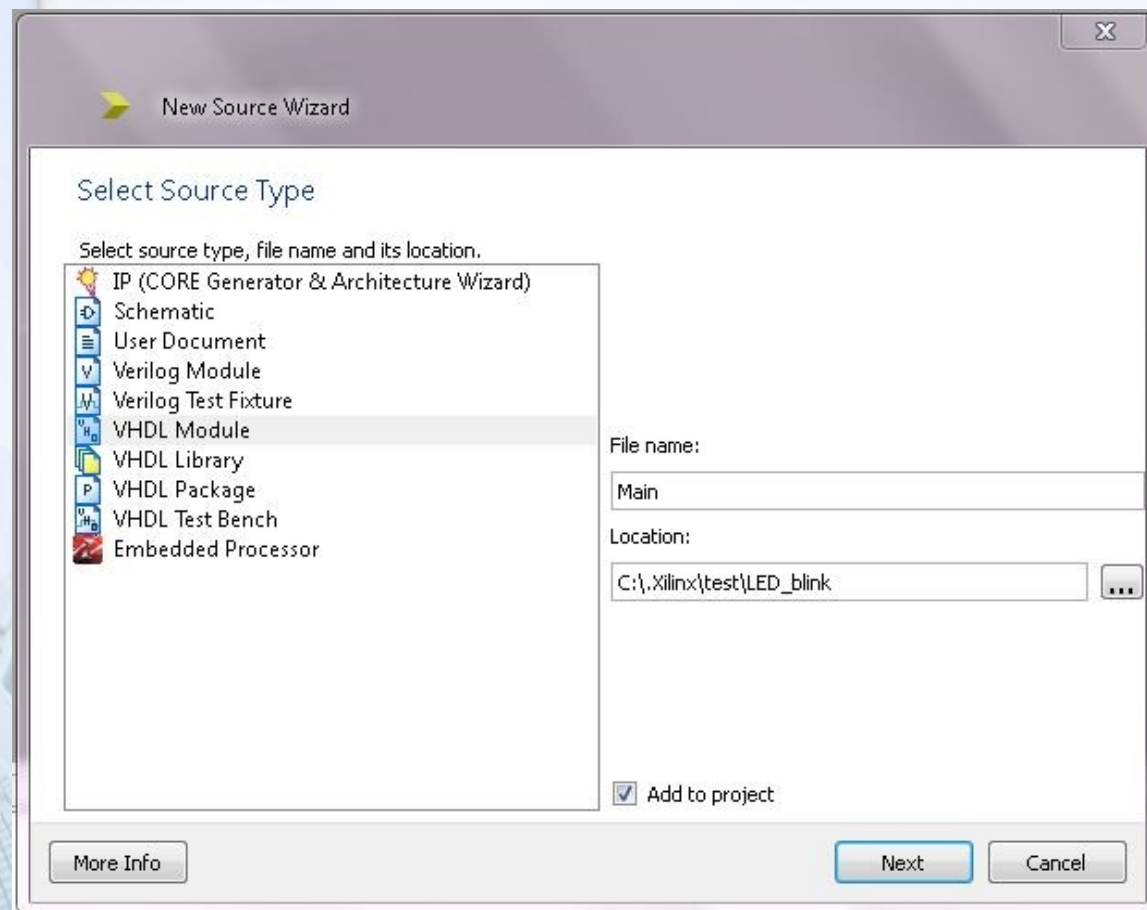
Property Name	Value
Evaluation Development Board	None Specified
Product Category	All
Family	Spartan3E
Device	XC3S1200E
Package	FG320
Speed	-5
Top-Level Source Type	HDL
Synthesis Tool	XST (VHDL/Verilog)
Simulator	ISim (VHDL/Verilog)
Preferred Language	VHDL
Property Specification in Project File	Store all values
Manual Compile Order	☐
VHDL Source Analysis Standard	VHDL-93
Enable Message Filtering	☐

More Info Next Cancel

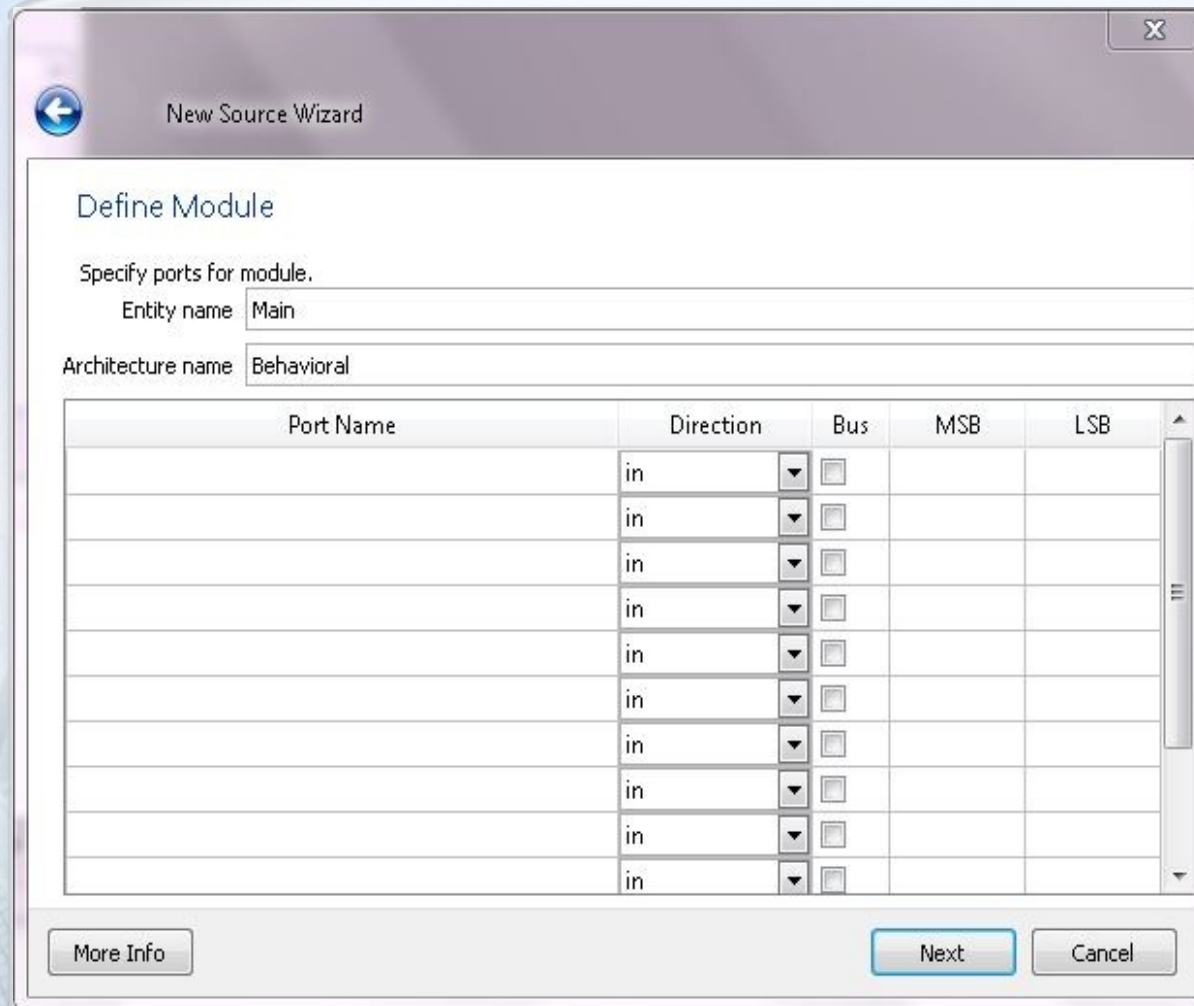
Нови празан пројекат



Додавање VHDLмодула



Дефинисање портова



The image shows a 'New Source Wizard' dialog box with a title bar containing a back arrow and a close button. The main area is titled 'Define Module'. Below this, it says 'Specify ports for module.' There are two input fields: 'Entity name' with the value 'Main' and 'Architecture name' with the value 'Behavioral'. Below these is a table for defining ports. The table has five columns: 'Port Name', 'Direction', 'Bus', 'MSB', and 'LSB'. The 'Port Name' column is empty for all rows. The 'Direction' column contains the value 'in' for all rows, with a dropdown arrow next to each. The 'Bus' column contains a checkbox for each row, all of which are currently unchecked. The 'MSB' and 'LSB' columns are empty for all rows. At the bottom of the dialog, there are three buttons: 'More Info', 'Next', and 'Cancel'.

New Source Wizard

Define Module

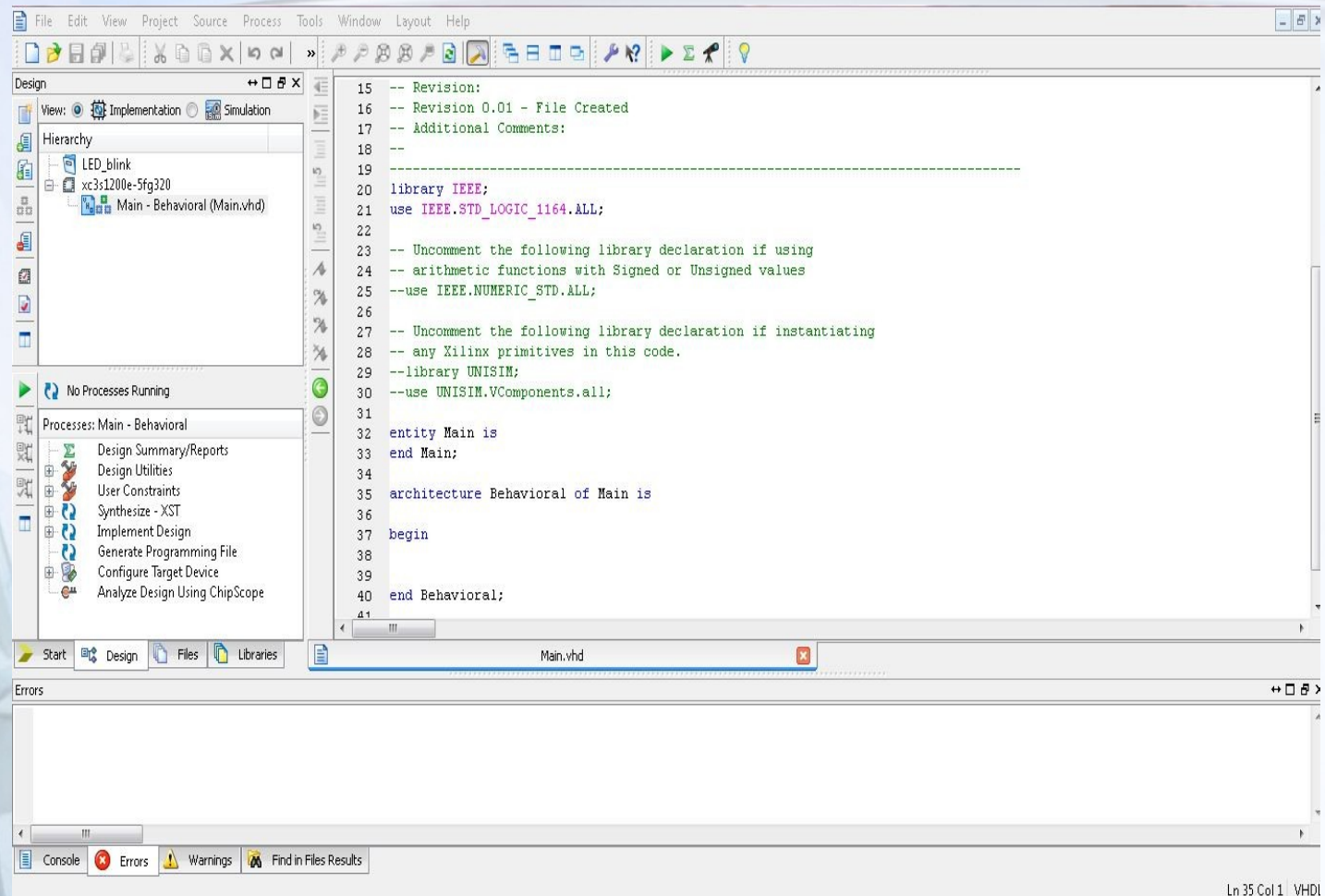
Specify ports for module.

Entity name

Architecture name

Port Name	Direction	Bus	MSB	LSB
	in	☐		
	in	☐		
	in	☐		
	in	☐		
	in	☐		
	in	☐		
	in	☐		
	in	☐		
	in	☐		
	in	☐		
	in	☐		

Главни прозор



Додавање библиотека и дефинисње портова

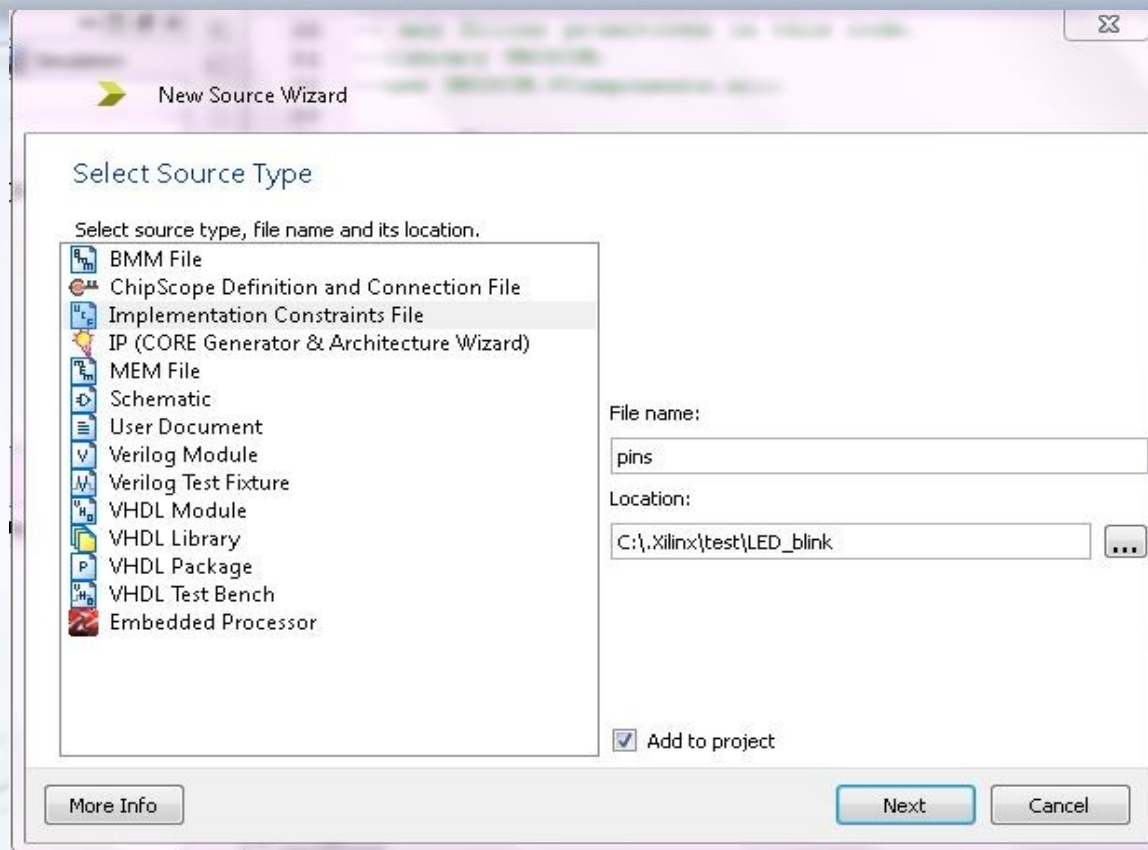
```
20 library IEEE;
21 use IEEE.STD_LOGIC_1164.ALL;
22 use IEEE.STD_LOGIC_ARITH.ALL;
23 use IEEE.STD_LOGIC_UNSIGNED.ALL;
24
25 -- Uncomment the following library declaration if using
26 -- arithmetic functions with Signed or Unsigned values
27 --use IEEE.NUMERIC_STD.ALL;
28
29 -- Uncomment the following library declaration if instantiating
30 -- any Xilinx primitives in this code.
31 --library UNISIM;
32 --use UNISIM.VComponents.all;
33
34 entity Main is
35 port (
36     CLK: in std_logic;
37     LED: out std_logic
38 );
39 end Main;
40
```

Главни програм

```
41 architecture Behavioral of Main is
42   signal Counter: std_logic_vector(24 downto 0);
43   signal CLK_1Hz: std_logic;
44   begin

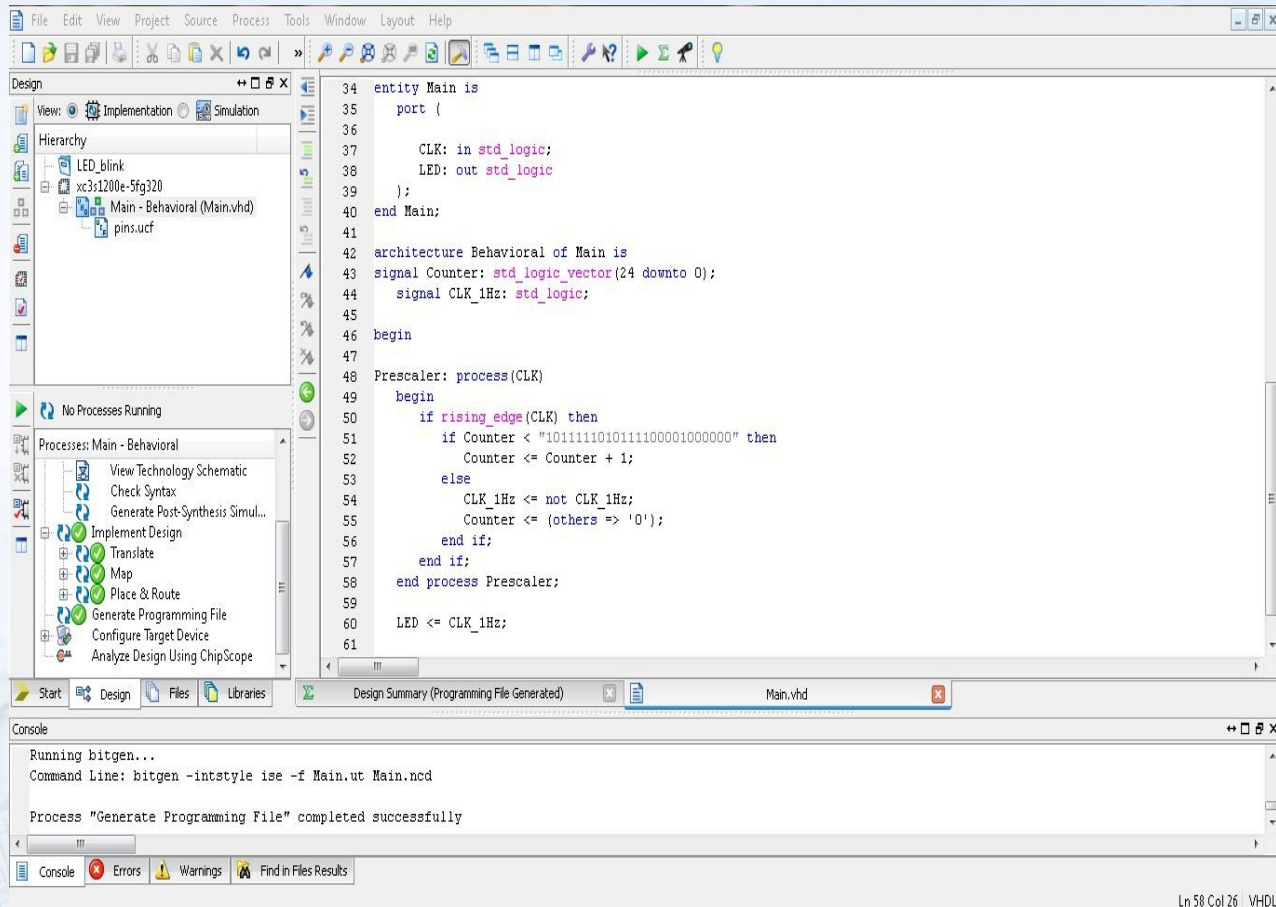
44   begin
45       if rising_edge(CLK) then
46           if Counter < "1011111010111100001000000" then
47               Counter <= Counter + 1;
48           else
49               CLK_1Hz <= not CLK_1Hz;
50               Counter <= (others => '0');
51           end if;
52       end if;
53   end process Prescaler;
54
55   LED <= CLK_1Hz;
56
57
58 end Behavioral;
```

Креирање Constraint фајла

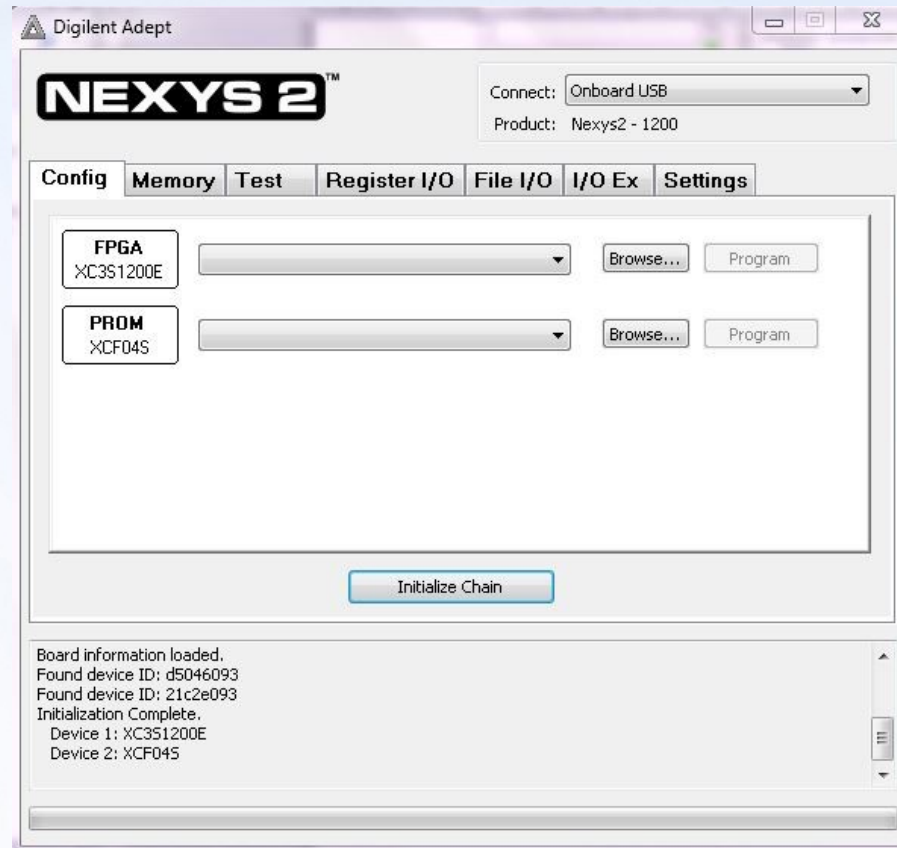


```
net "CLK" loc = "B8";  
net "LED" loc = "k15";
```

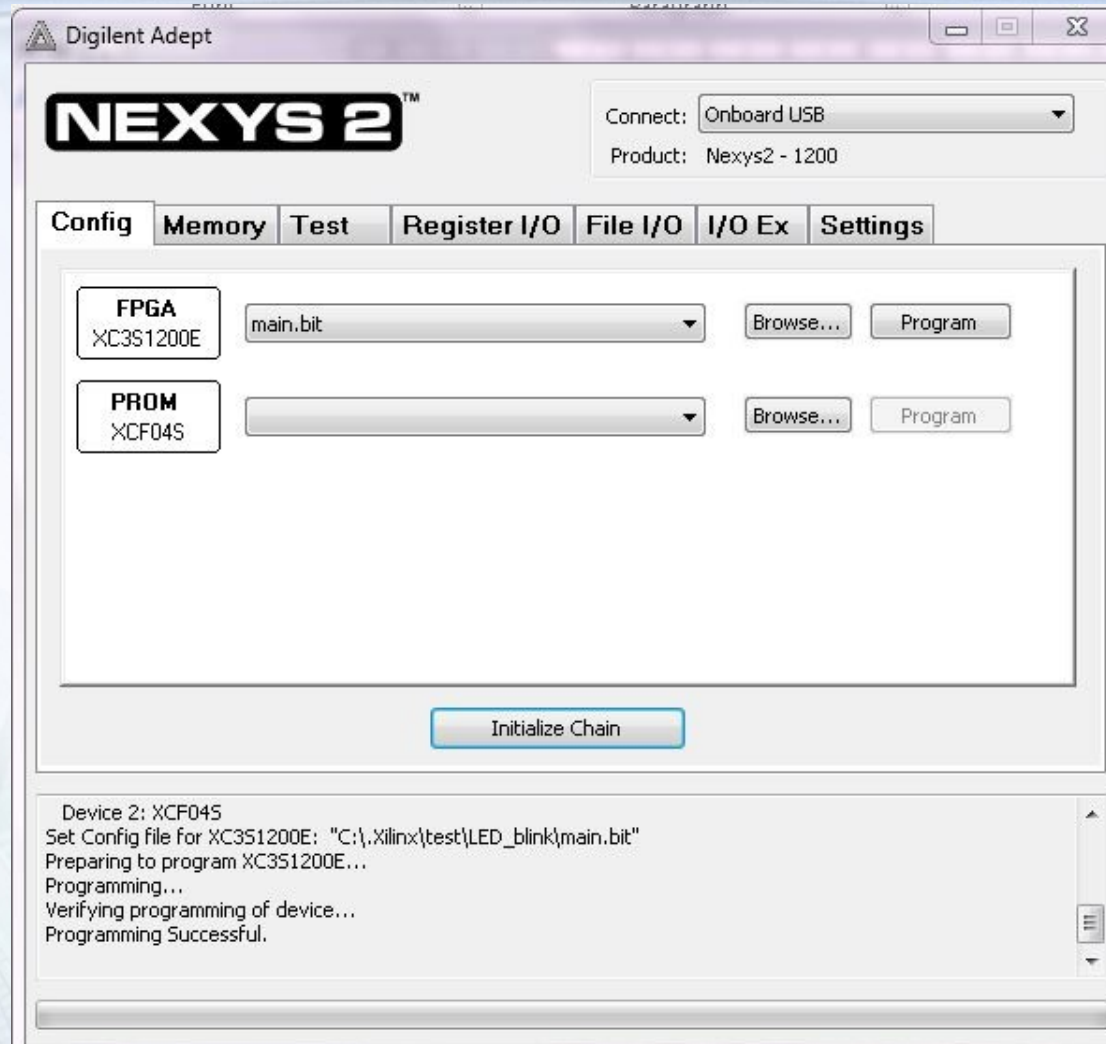
Генерисање програмског фајла



Повезивање плоче



Програмирање плоче



Блинкање LED-а

