Start Project Navigator

1. Start new Project Navigator from "*Start Menu*."

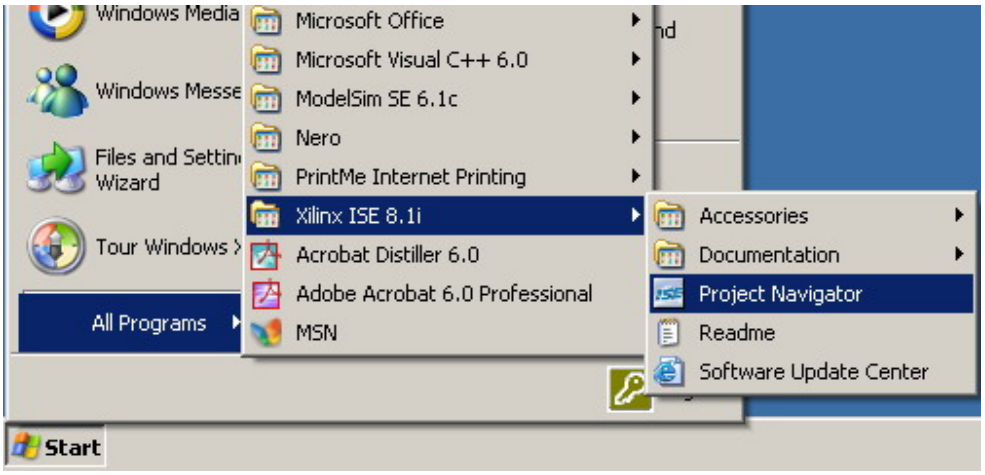

Fig (1)

Create New Project

1. File ->New Project

2. In the dialog page, enter project named and location (somewhere on your network drive, i.e. Z:\. The type of the top-level module should be **HDL**.

3. If you want to use schematic to implement your design (Lab1), choose your project type on the dialog shown as Fig (2).

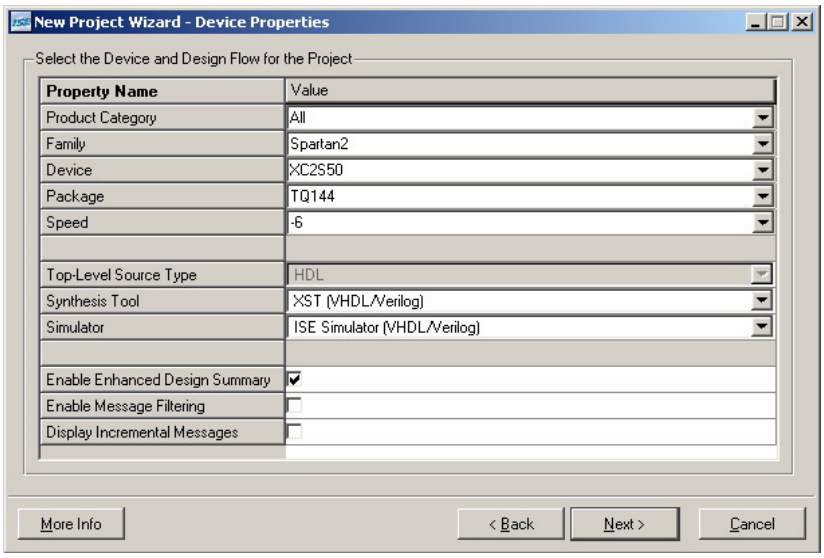

Fig (2)

4. If you want to use Verilog to implement your design (Lab2~Lab5), choose your simulator value as "**Modelsim-SE Verilog**", shown as Fig (3).

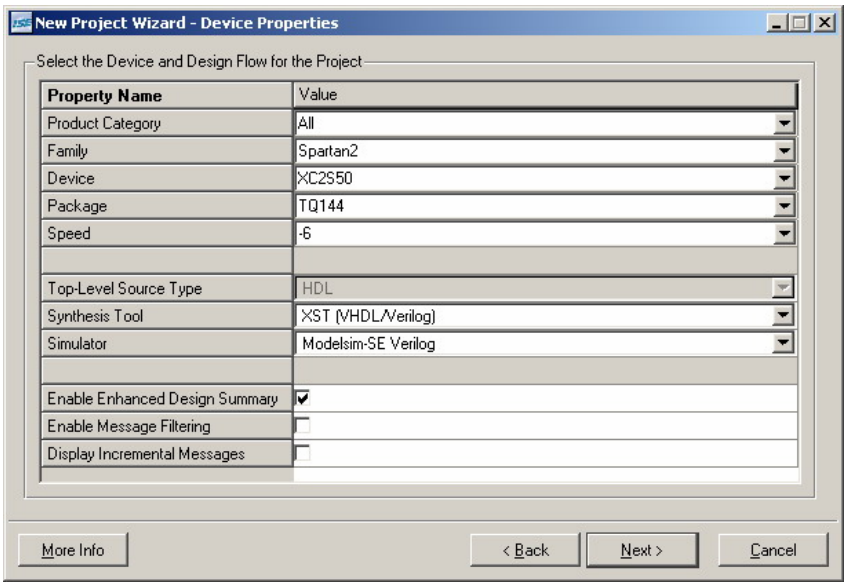

Fig (3)

Create New Schematic

1. Create New Source->Schematic.

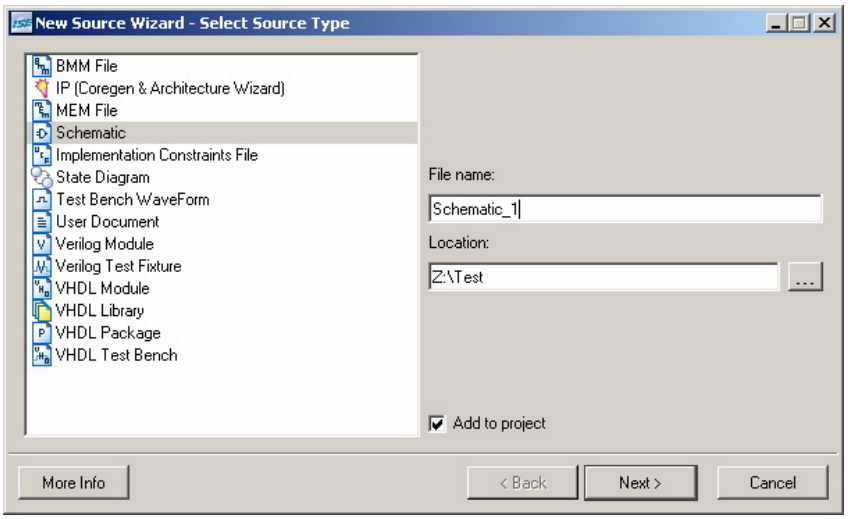

Fig (4)

2. Select the symbols for your schematic from the panel on the left.

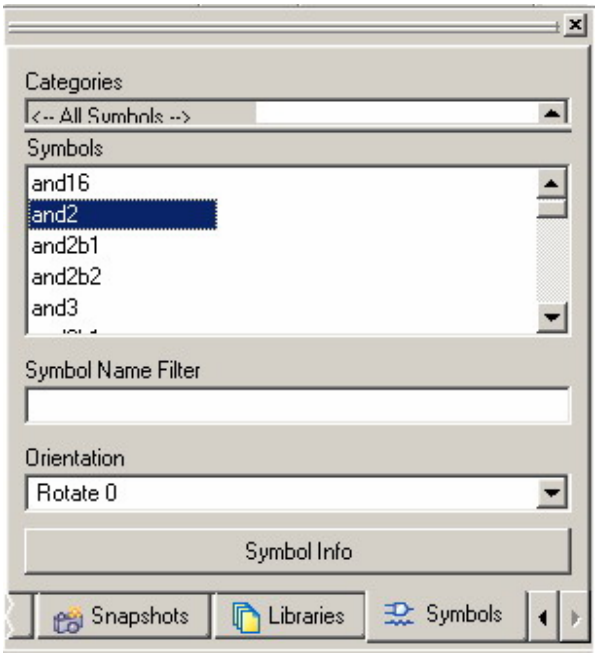

Fig (5)

3. Add -> I/O Maker to add the input/output pins of your schematic.

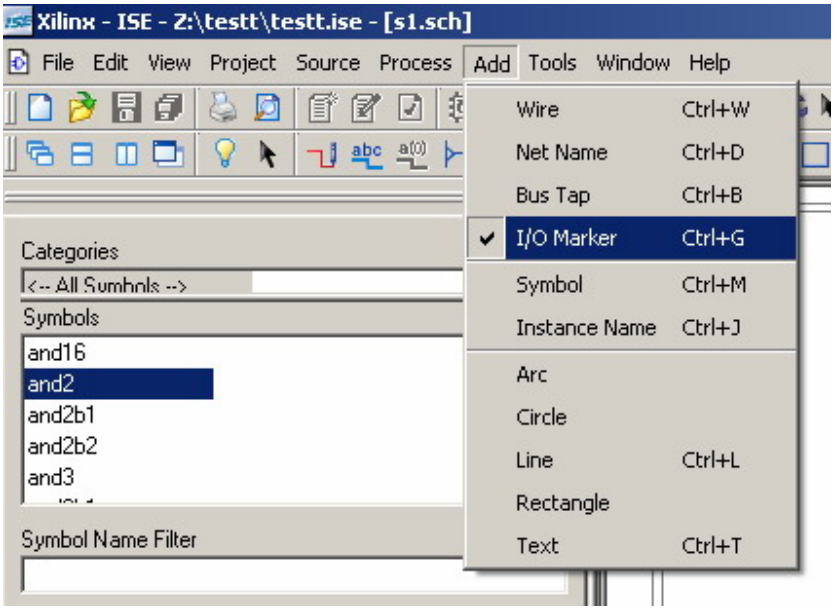

Fig (6)

4. Double clicking on the I/O label can edit the property of each I/O pin.

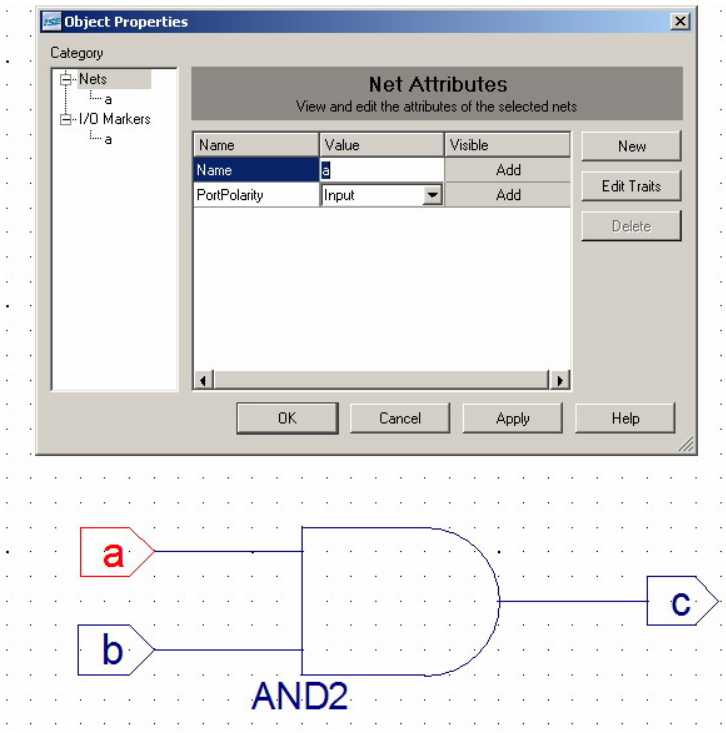

Fig (7)

Create New Verilog Module

1. Create New Source->Verilog Module.

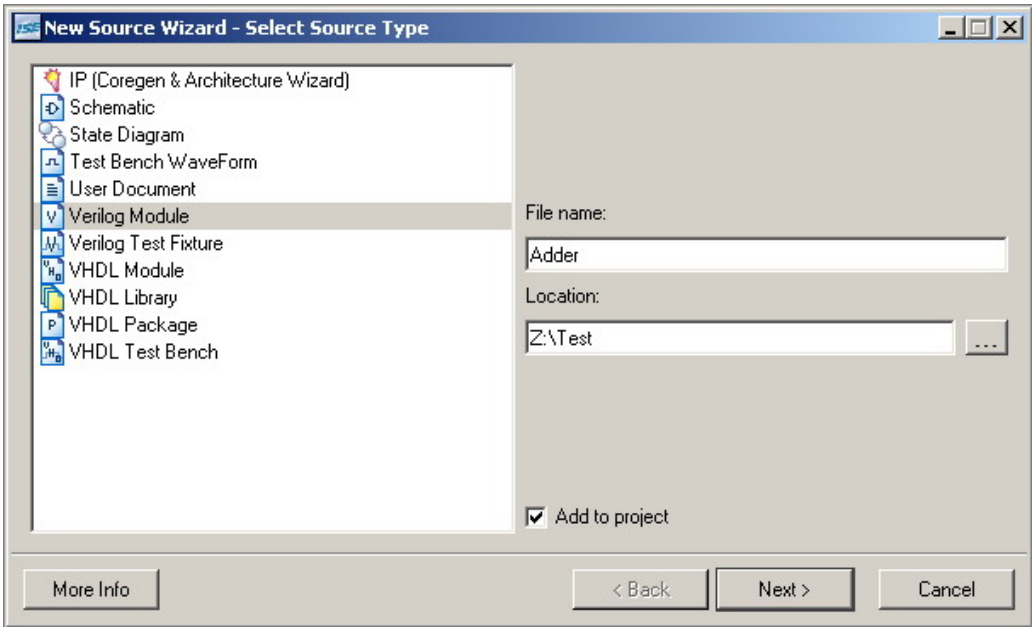

Fig (8)

2. Set Module I/O port.

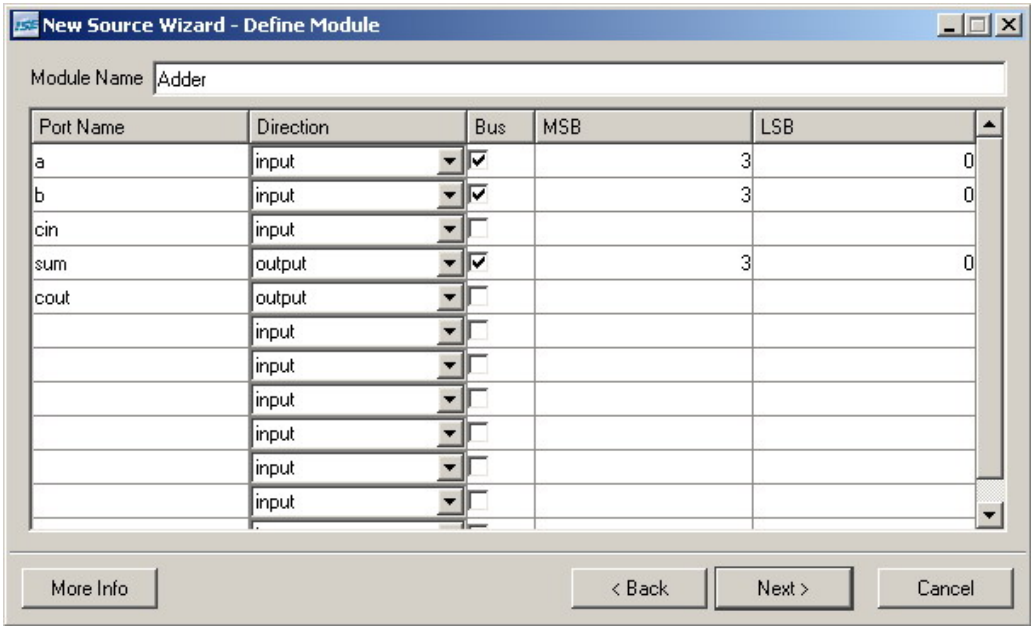

Fig (9)

Create Test Bench Waveform

1. Create New Source->Test Bench Waveform.

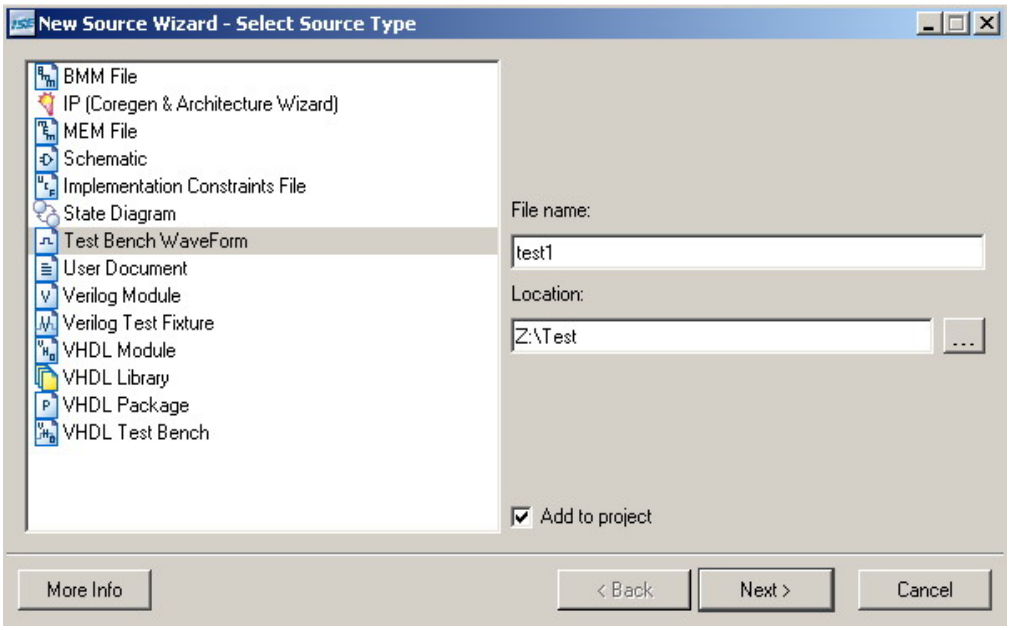

Fig (10)

2. Select a source you want to associate with the test bench.

3. Enter the test pattern and the expected output waveforms.

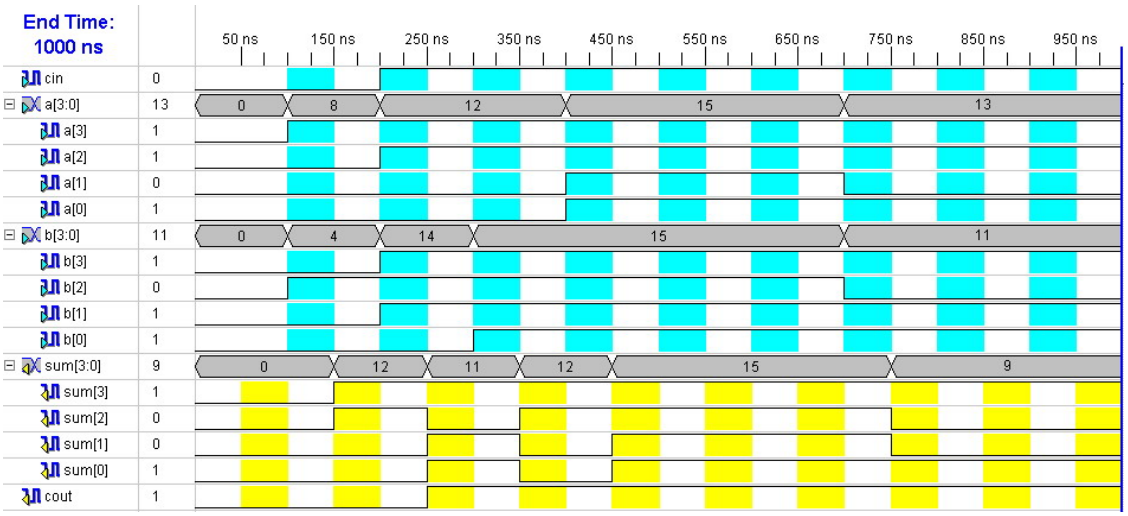

Fig (11)

Run Simulation

1. Change the mode to "*Behavioral Simulation*."

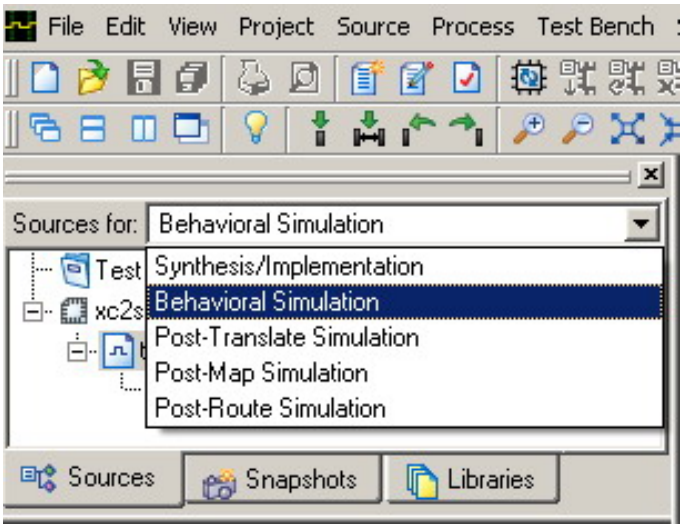

Fig (12)

2. If you are using schematic to design, double click on "*Simulate Behavioral Model*" to run the simulation, shown as Fig (13)

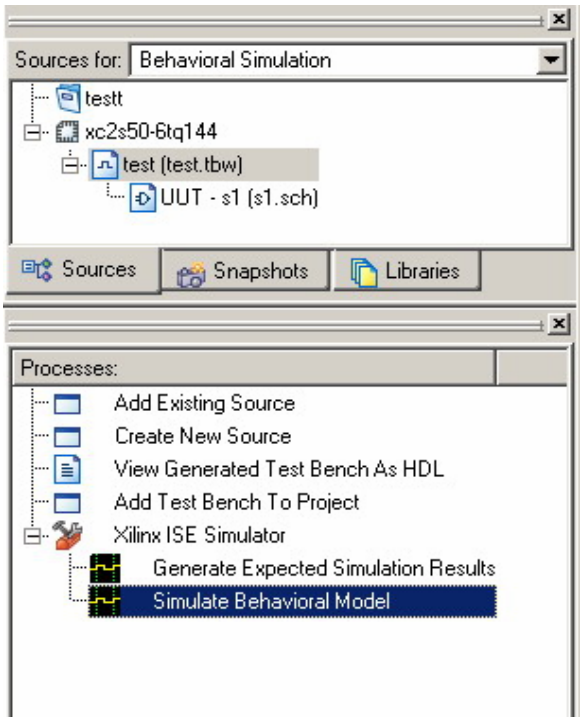

Fig (13)

3. If you are using Verilog to design, double clicking on "*Simulate Behavioral Model*" will call *ModelSim* to run the simulation, shown as Fig (14).

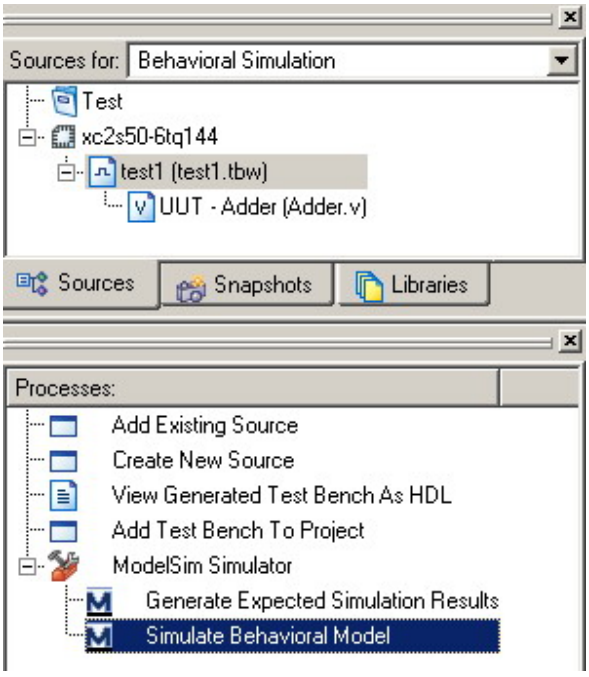

Fig (14)

Download Design to FPGA Board

1. Change the mode back to "*Synthesis/Implementation*."

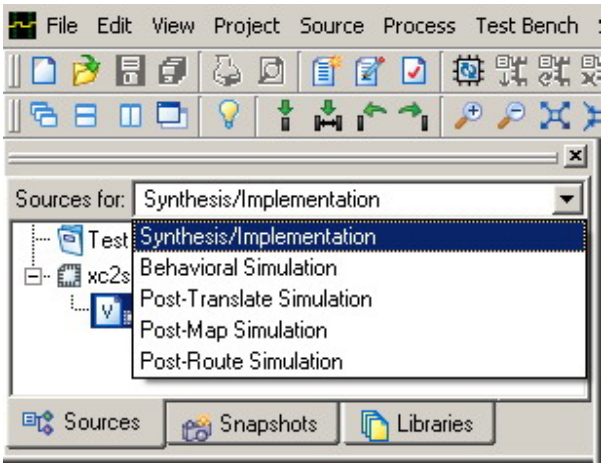

Fig (15)

2. Create New Source->Implementation Constraints File.

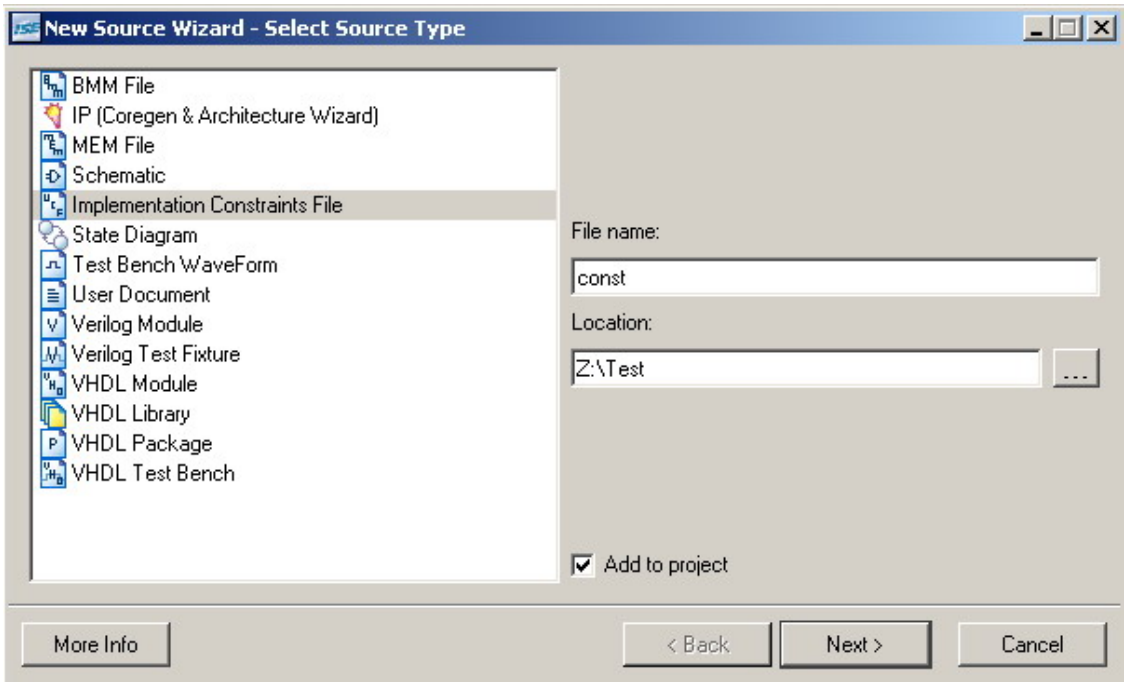

Fig (16)

3. Select the Verilog file you want to associate with the constraints file.

4. Double click on "*Assign Package Pins*".

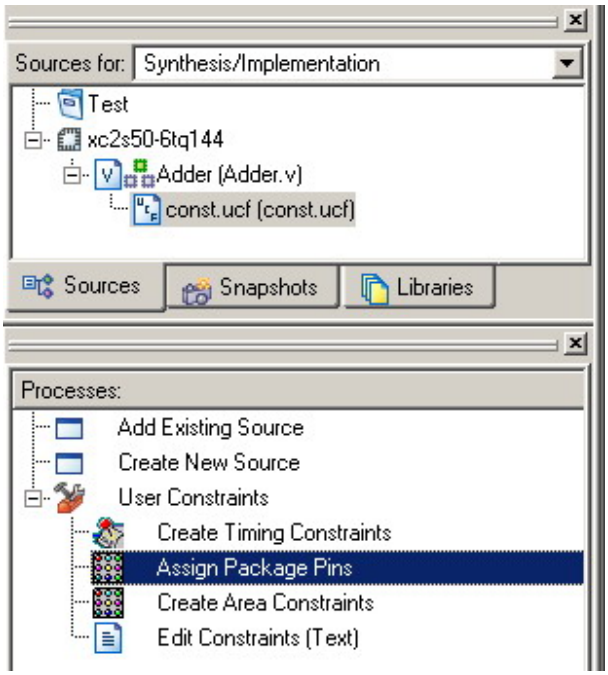

Fig (17)

5. Enter the pin number for each FPGA pin on the left side of the window.

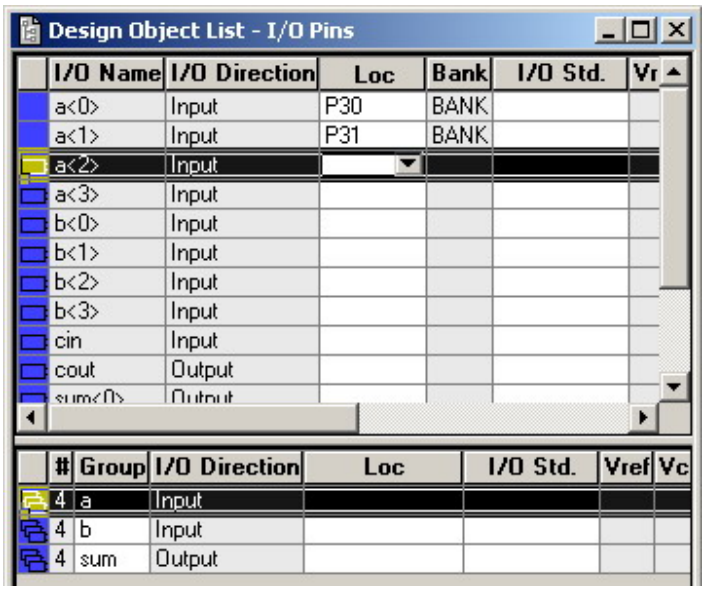

Fig (18)

6. To find out which pins of the board you can use for external logic refer to the XSA50 manual. As you can see below, some of these pins can be used as general-purpose I/O with the bitstream.

Configuration Pins (30*, 31*, 37, 38*, 39*, 44*, 46*, 49*, 57*, 60*, 62*, 67*, 68*, 69, 72, 106, 109, 111): These pins are used to load the SpartanII FPGA with a configuration bitstream. Some of these pins are dedicated to the configuration process and cannot be used as general-purpose I/O (37, 69, 72, 106, 109, 111). The rest can be used as general-purpose I/O after the FPGA is configured. If external logic is connected to these pins, you may have to disable it during the configuration process. The DONE pin (72) can be used for this purpose since it goes to a logic high only after the configuration process is completed.

Free Pins (77*, 78*, 79*, 80*, 83*, 84*, 85*, 86*, 87*): These pins are not connected to any other devices on the XSA Board so they can be used without restrictions as general-purpose I/O through the prototyping header.

7. When entering a pin number always insert a “P” before the number.

8. Synthesize and implement the design. Perform all the steps that are check-marked in Fig (19).

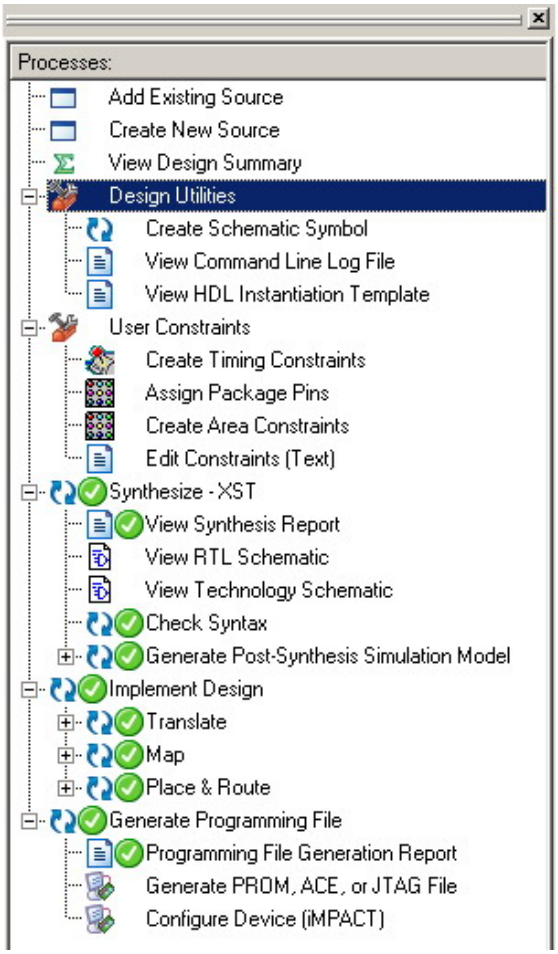

Fig (19)

9. The “.bit” file for FPGA board configuration is ready.