Visual Tutorial Xilinx ISE

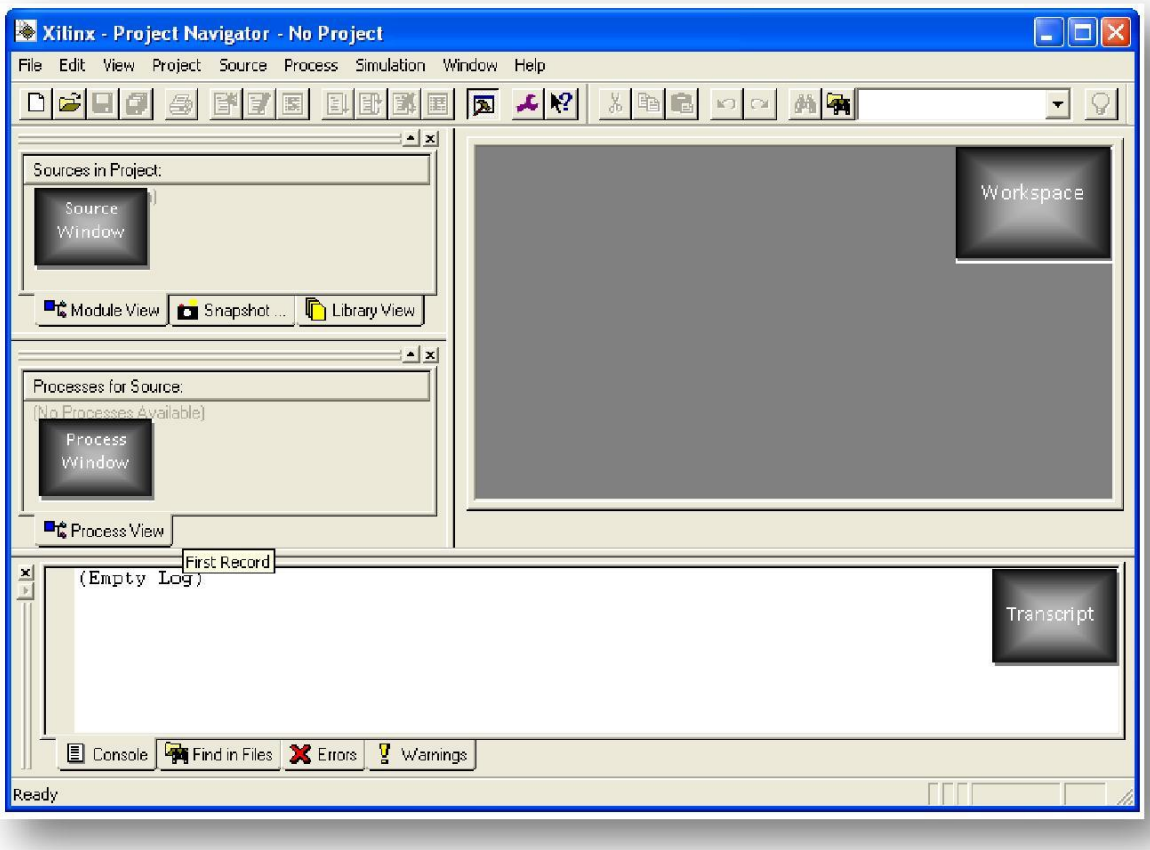
Department of Computer Science and IT, UOL
DLD / Fall 2016
Course Instructor: Rao Muhammad Umer
Visual Tutorial Xilinx ISE

STEP BY STEP DEMONSTARTION OF IMPLEMENTING A DESIGN ON FPGA USING XILINX ISE

Step 1: **Start the Xilinx ISE**

Start the Xilinx Project Navigator by desktop shortcuts or by using

Start → Programs → Xilinx ISE 7.1i



Step 2: **Create a New Project**

In the window go to *File → New Project*

Specify the project name and location.

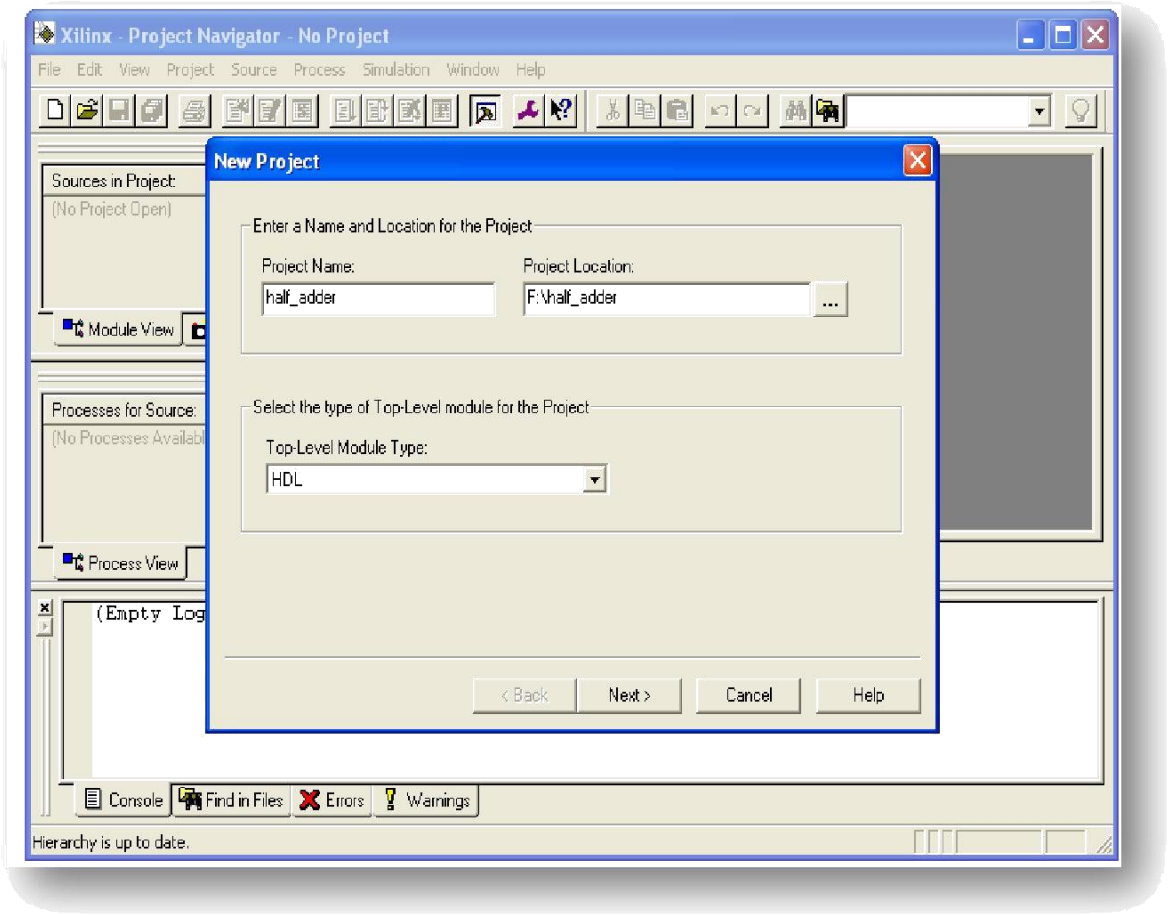
Verify that HDL is selected from the Top-Level Source Type List.

Click **Next** to move to device properties page.

Use the pull-down arrow to select the Value for each device according to the Proto Board

Device Family:	*Spartan3*
Device:	*xc3s400*
Package:	*pq208*
Speed Grade:	*-5*
Top-Level Module Type	*HDL*
Synthesis Tool:	*XST (VHDL / Verilog)*
Simulator:	*ISE Simulator*
Generated Simulation Language:	*Verilog*

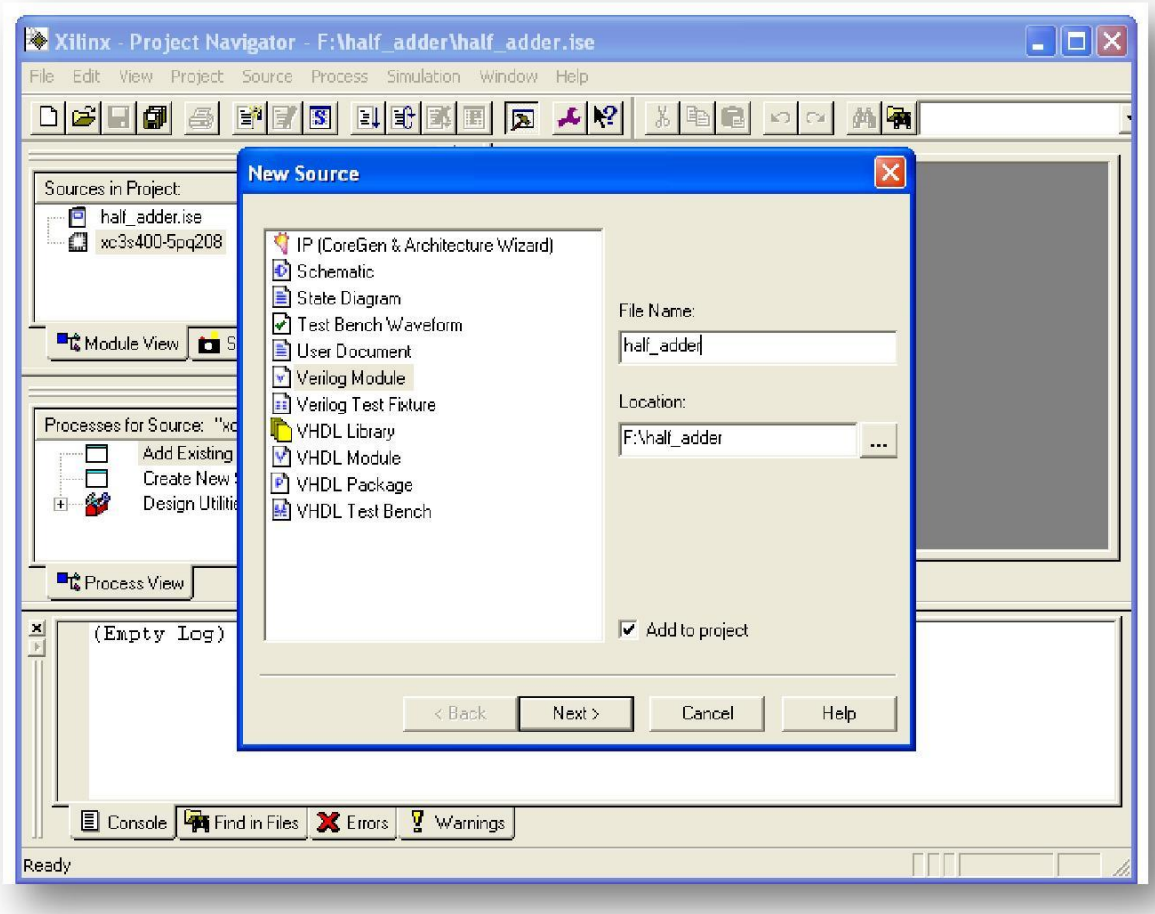
Click **Finish**. Project summary is seen.

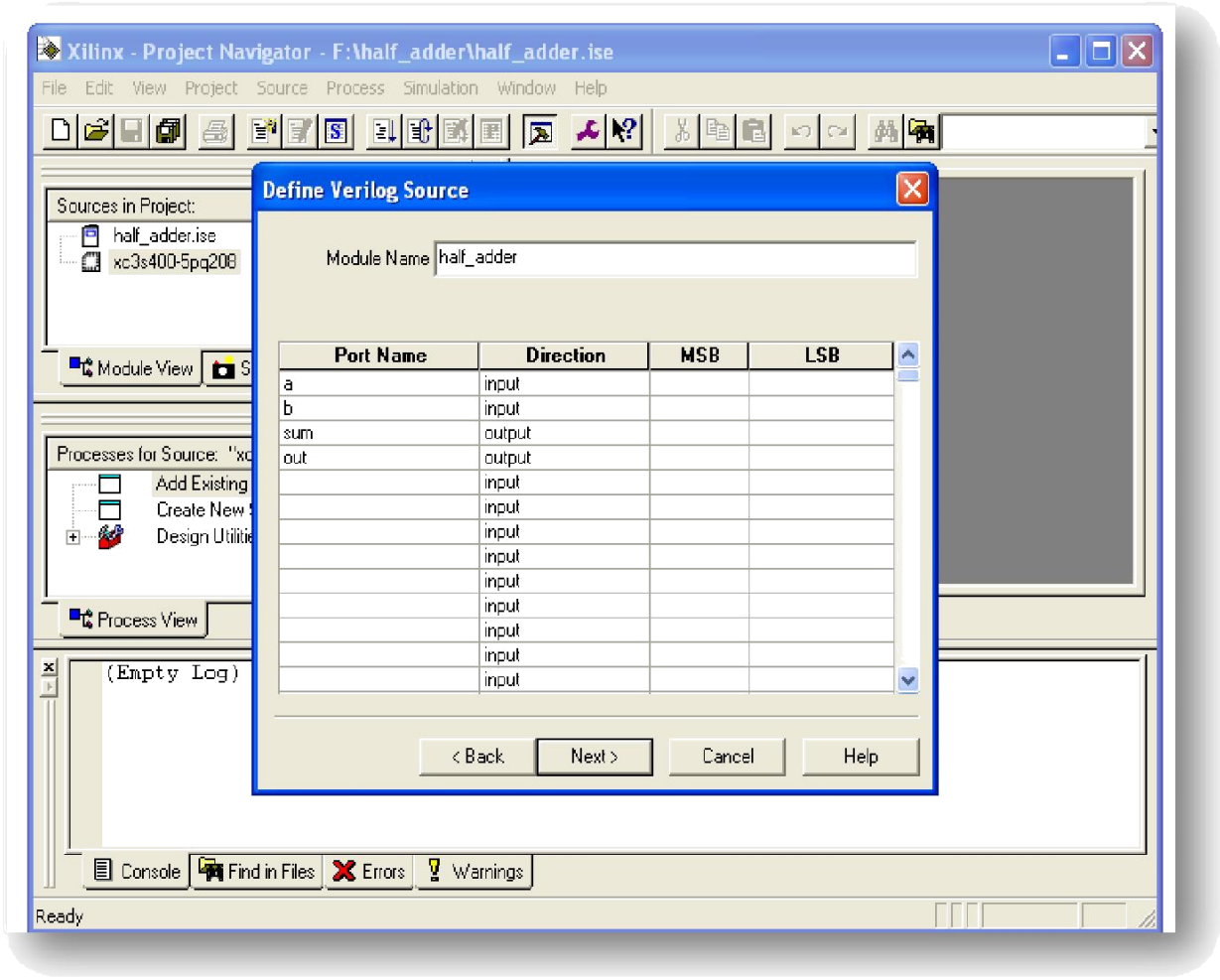

Xilinx - Project Navigator - No Project
File Edit View Project Source Process Simulation Window Help
Sources in Project:
(No Project Open)
Module View
Processes for Source:
Process View
(Empty Log
Console
Find in Files
Errors
Warnings
Hierarchy is up to date.
New Project
Enter a Name and Location for the Project
Project Name:
half_adder
Project Location:
F:\half_adder
...
Select the type of Top-Level module for the Project
Top-Level Module Type:
HDL
< Back
Next >
Cancel
Help

Step 3: **Create a New Verilog Source File**

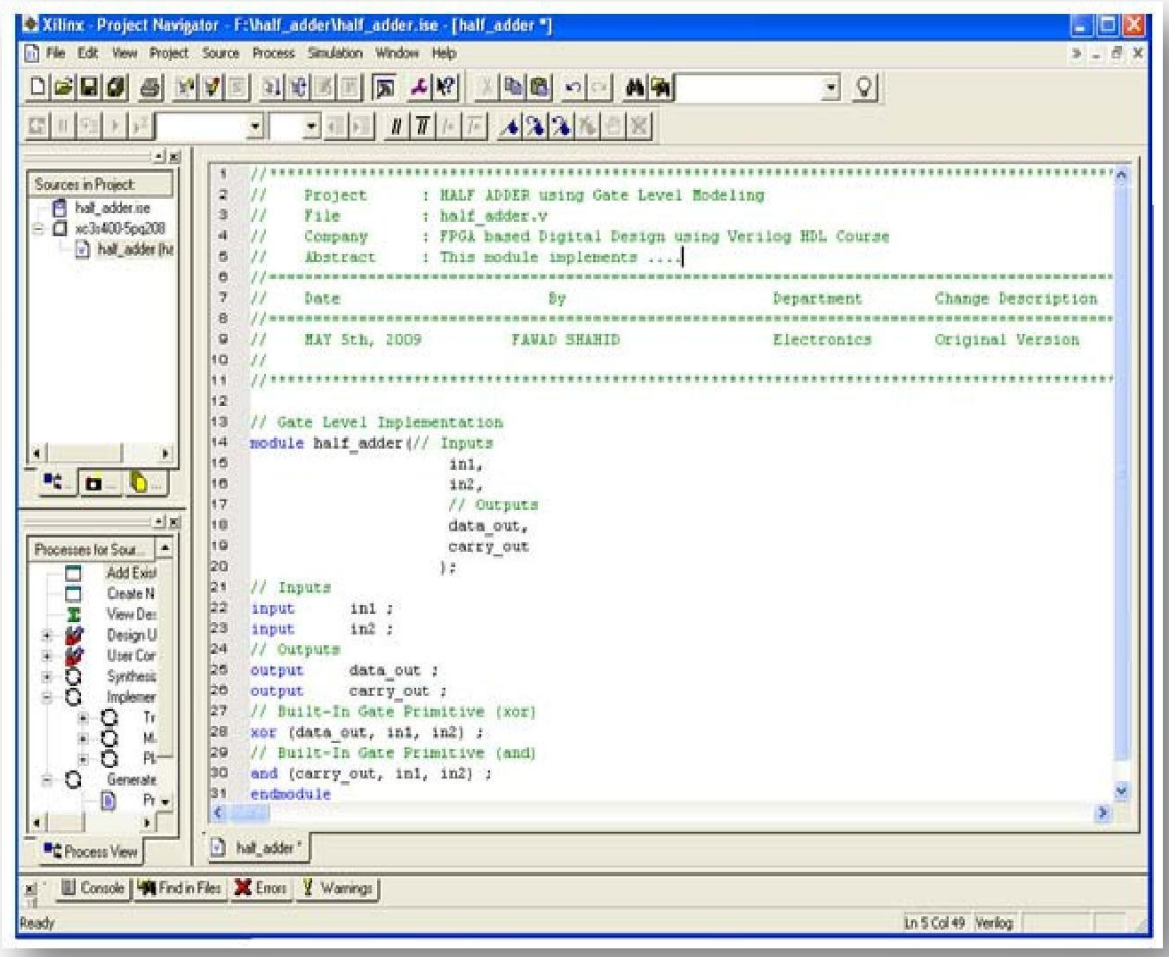
Click on the symbol of FPGA device in the left pane and then *right click* → *Click on new source* → *Verilog module* and give the file name. *Click Next* → *Define Ports* in this case.

Click **Next** twice and then **Finish**.



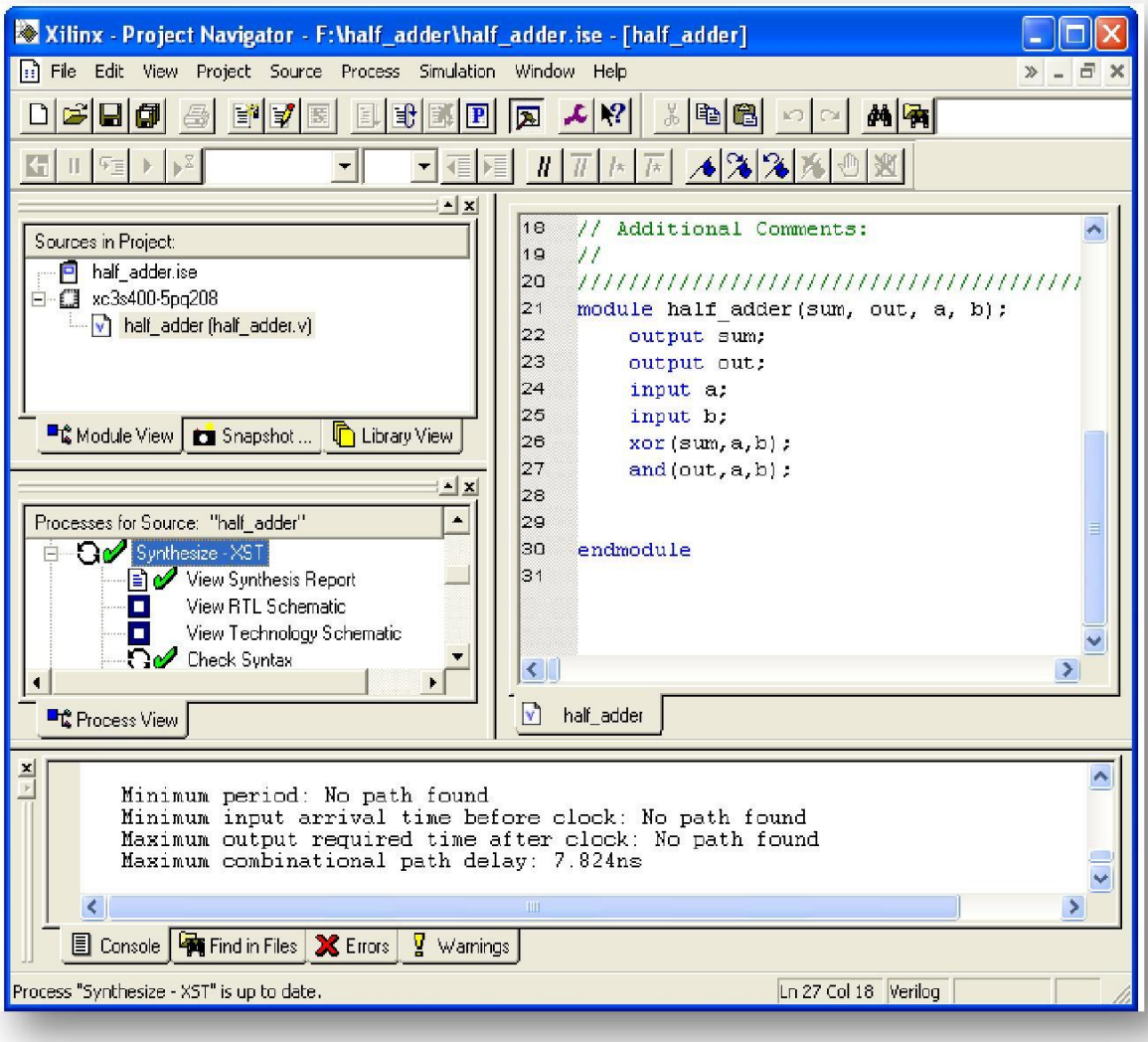

Xilinx - Project Navigator - F:\half_adder\half_adder.ise
File Edit View Project Source Process Simulation Window Help
Sources in Project:
half_adder.ise
xc3s400-5pq208
Module View
Processes for Source:
Add Existing
Create New
Design Utilities
Process View
(Empty Log)
Console
Find in Files
Errors
Warnings
Ready
Define Verilog Source
Module Name half_adder
Port Name
Direction
MSB
LSB
a input
b input
sum output
out output
< Back
Next >
Cancel
Help

Step 4: **Write the Code for the Design in Verilog Editor**



Step 5: **Check Syntax**

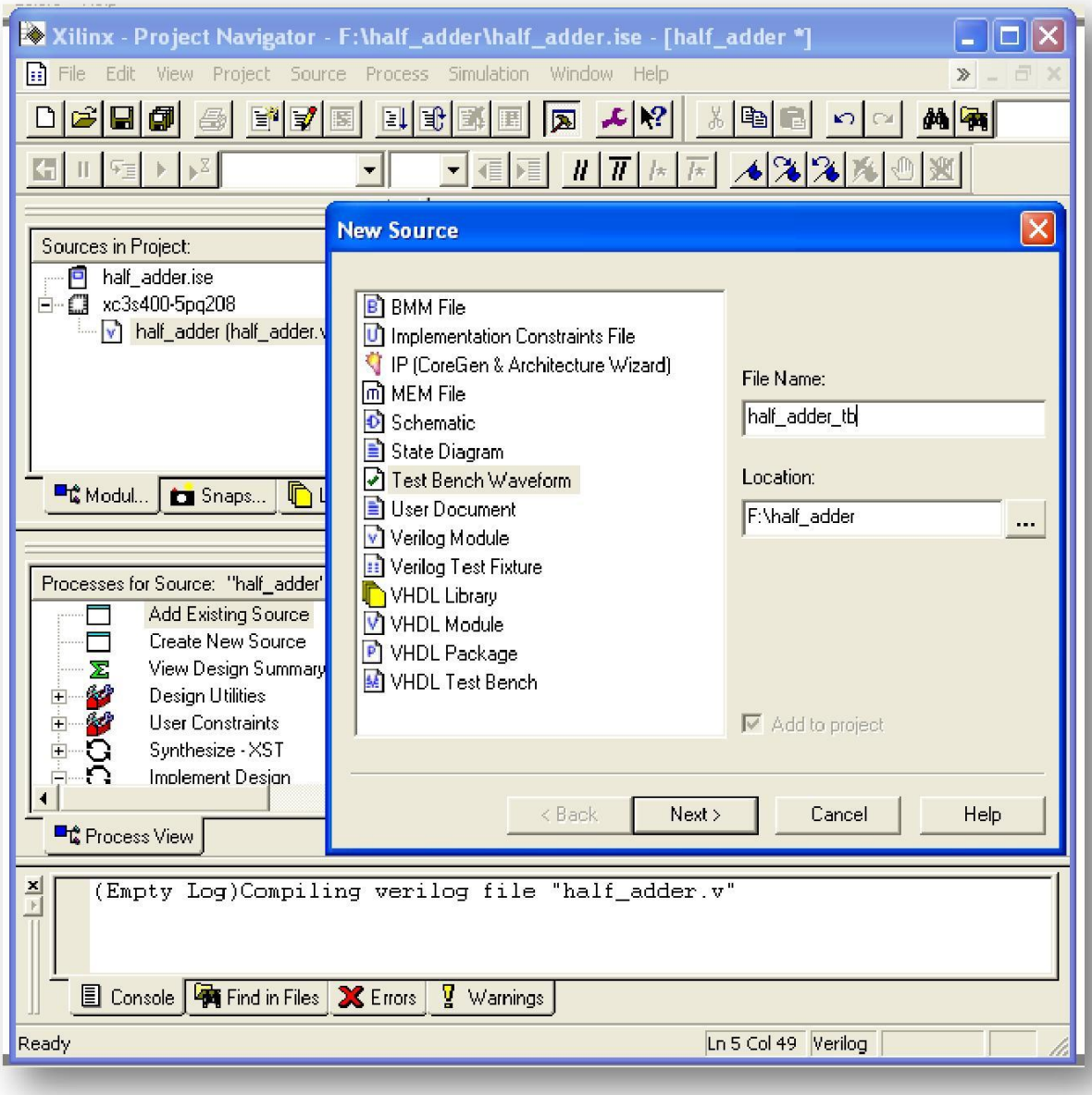
Run the *Check syntax → Process window → synthesize → check syntax* and remove errors if present.



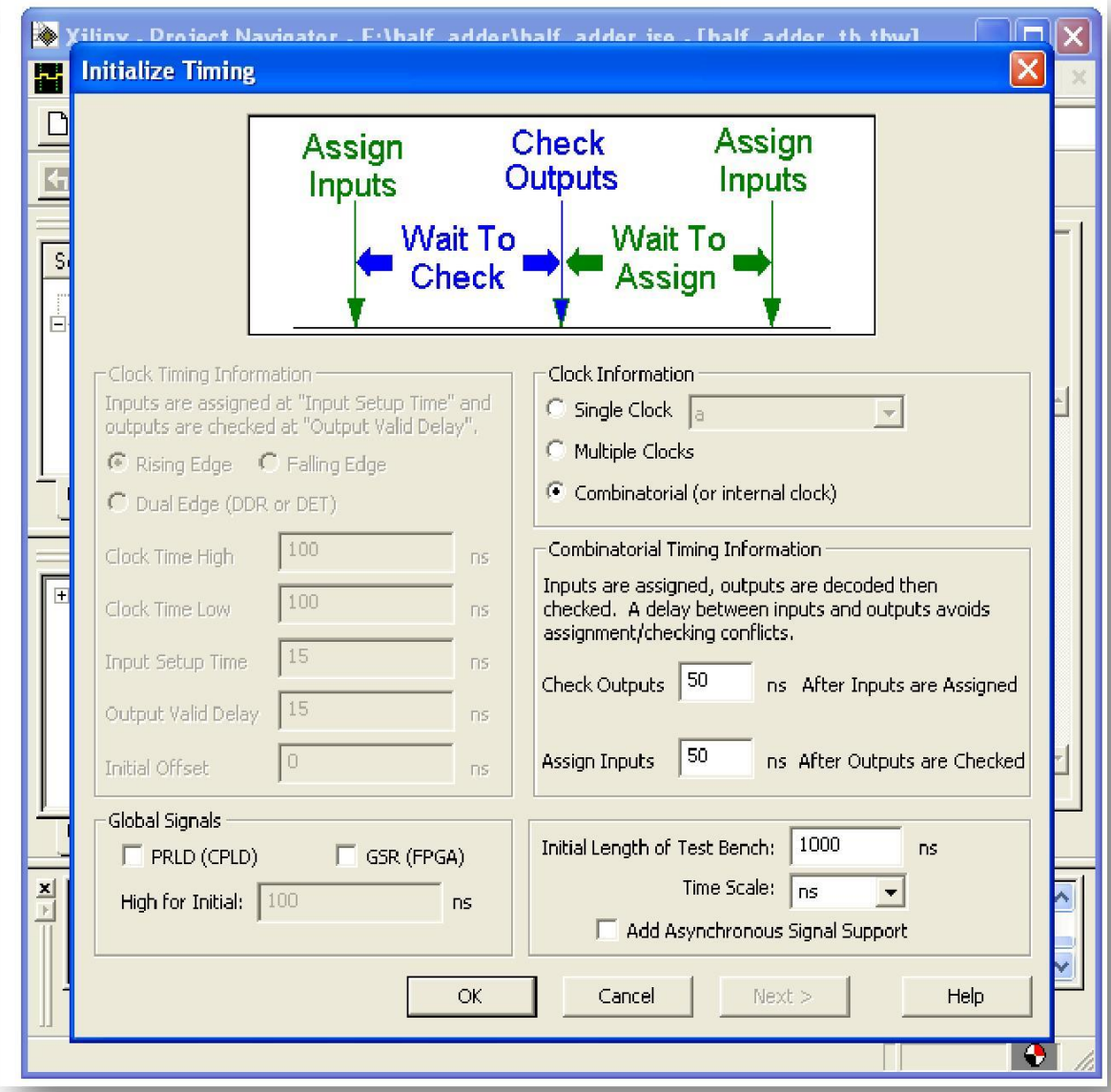
Step 6: **Create a Test Bench File**

Verify the operation of your design before implementing it as hardware. Simulation can be done using ISE Simulator.

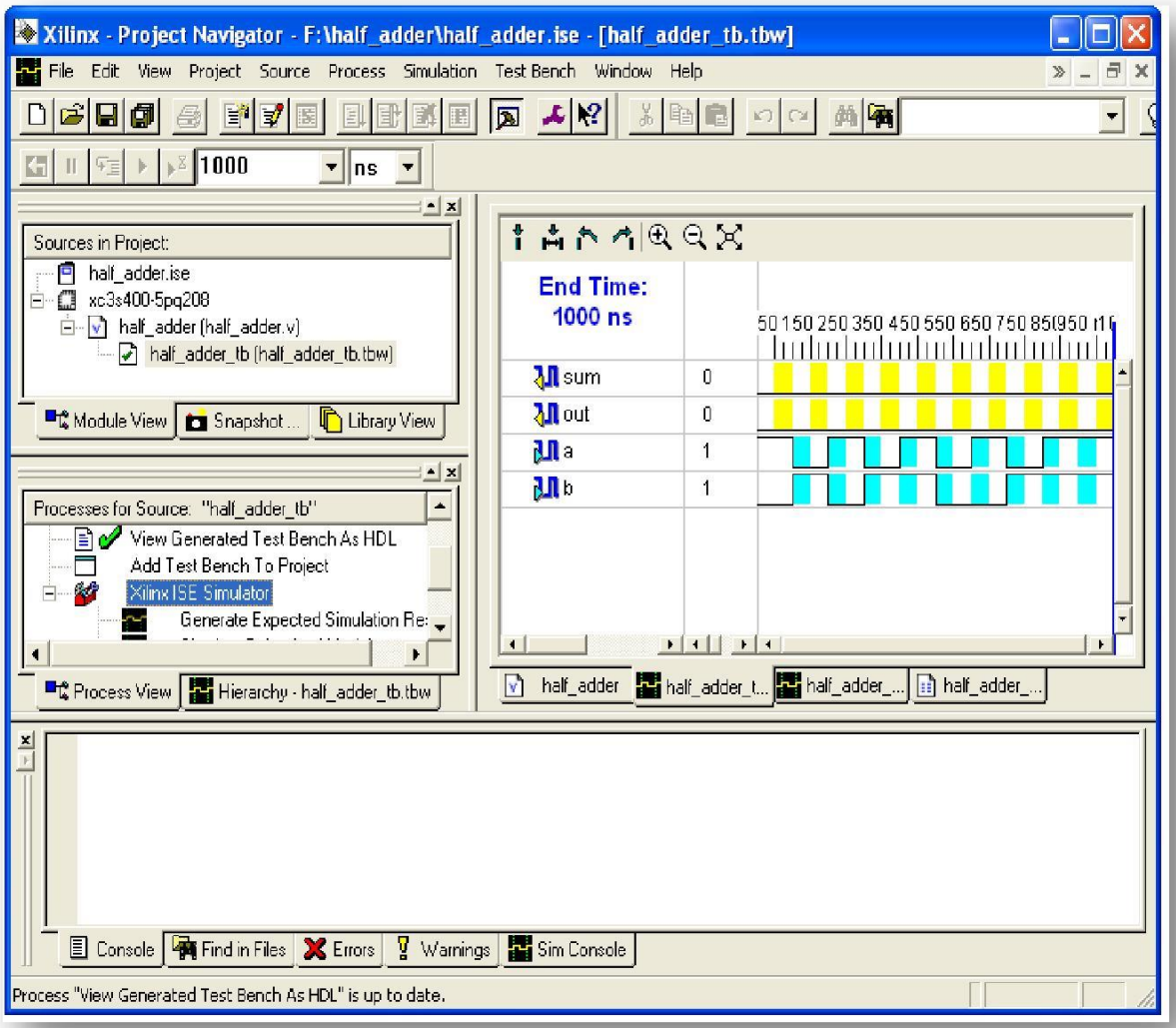
For this click on symbol of FPGA device and then *right click → Click on new source → Test Bench Waveform and give the name → Select entity → Finish*



Select the desired parameters for simulating your design. In this case select combinational circuit. Click **OK**.



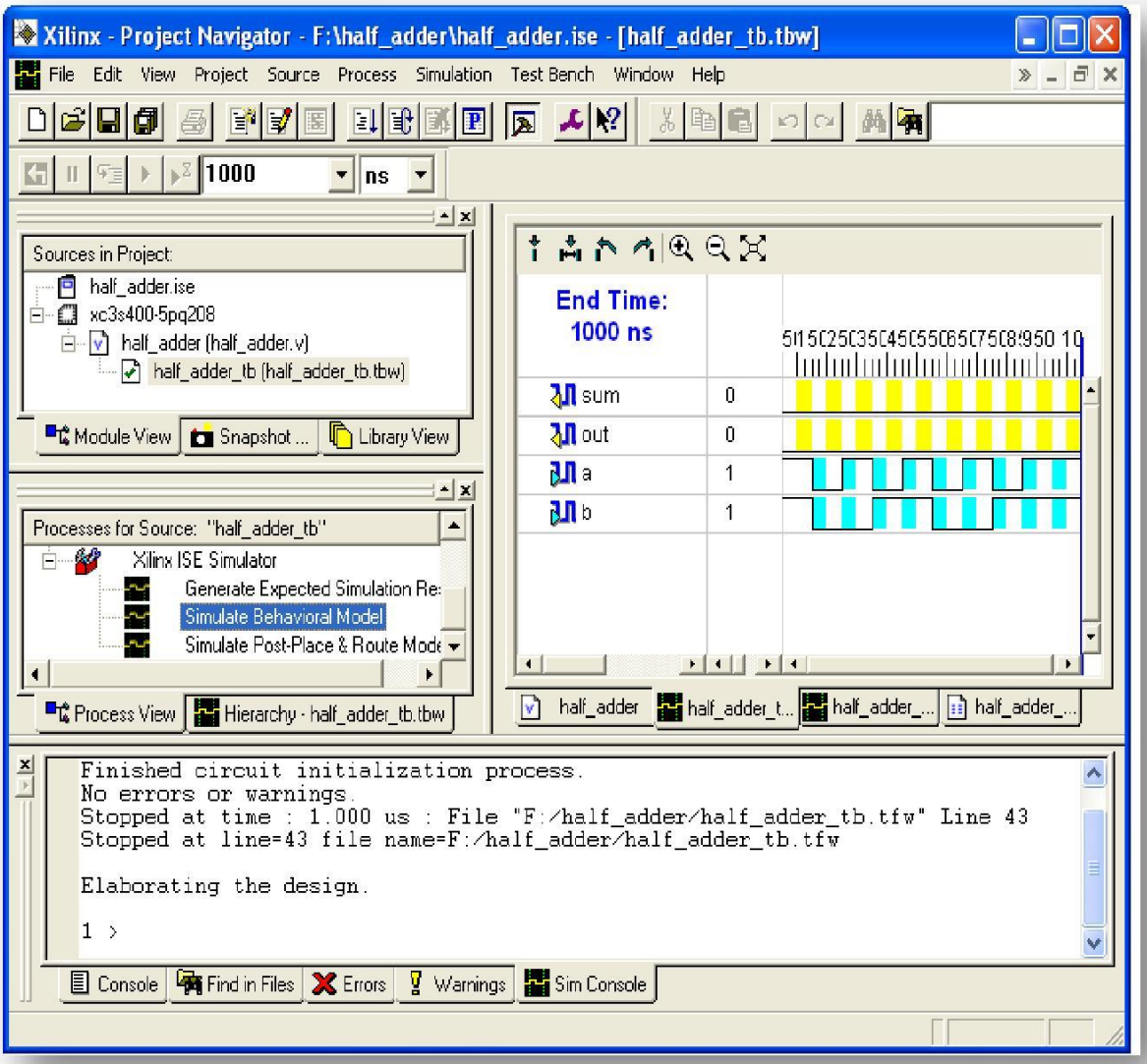
Test Bench file is created with extension ***".tbw"***



Step 7: **Simulate the Code**

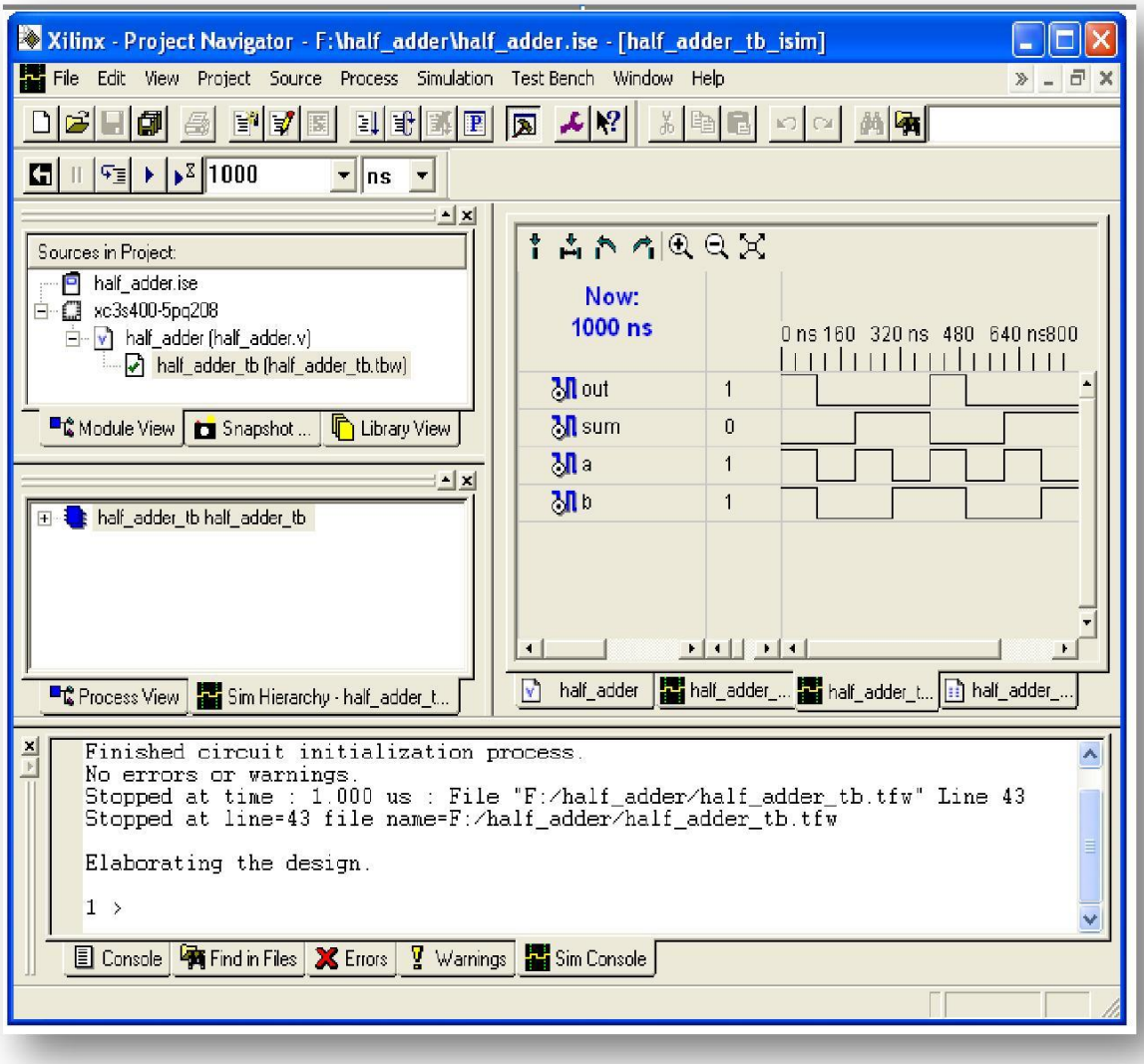
Click on test bench file. Test bench file will open in main window. Assign all the signals and save five.

From the source of process window, click on ***Simulate Behavioral Model*** in Process window.



Verify the design in wave window by seeing behavior of output signal with respect to input signal.

Close the ISE Simulator.



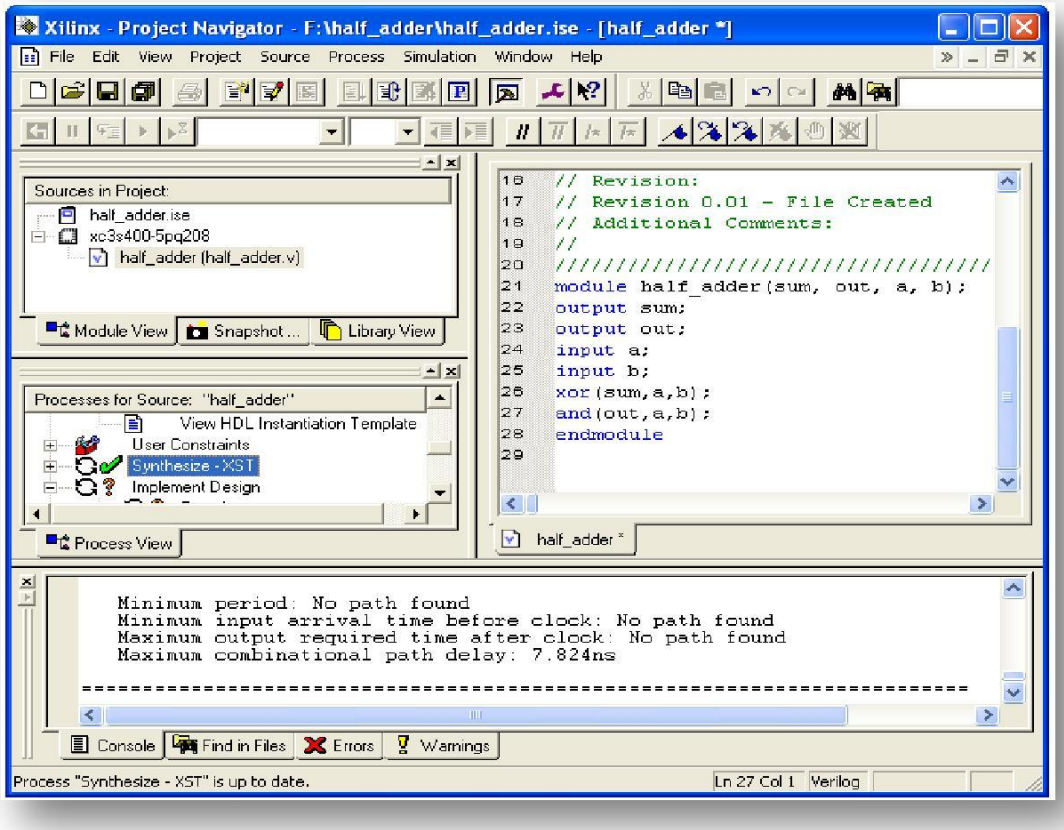
Step 8: **Synthesize the Design using XST**

Translate the design into gates and optimize it for the target architecture. This is the synthesis phase.

Highlight the Verilog file for the design in the Sources in Project window.

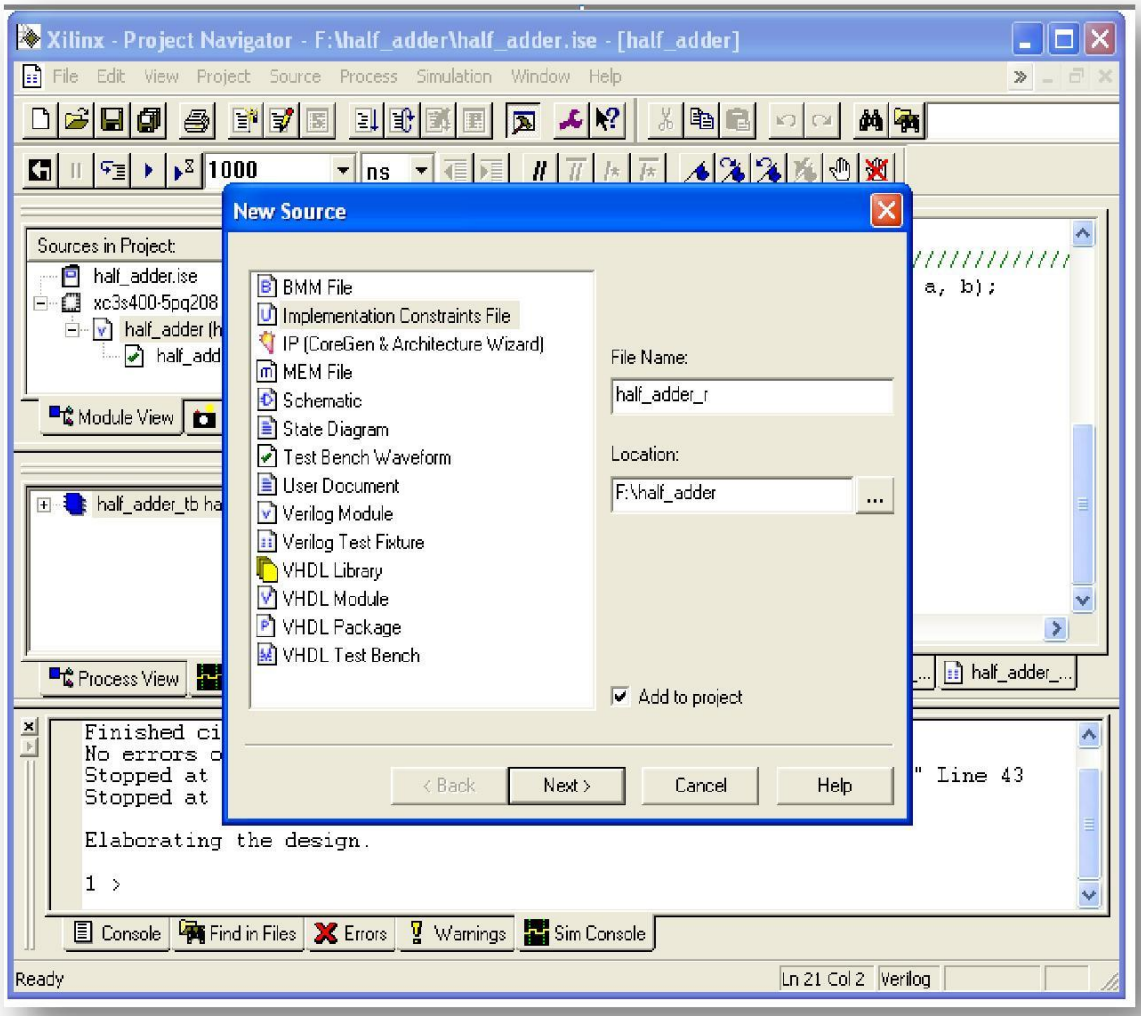
To run synthesis, *right click on Synthesize* and choose **Run** option, or double click on Synthesize in the processes for current source window. Synthesize will run and green check √ will appear next to Synthesize, when it is successfully completed. A yellow exclamation mark **!** indicates warnings and a red cross **X** indicates errors. Warnings are OK.

Check the synthesis report, If there are errors correct it and rerun synthesis.

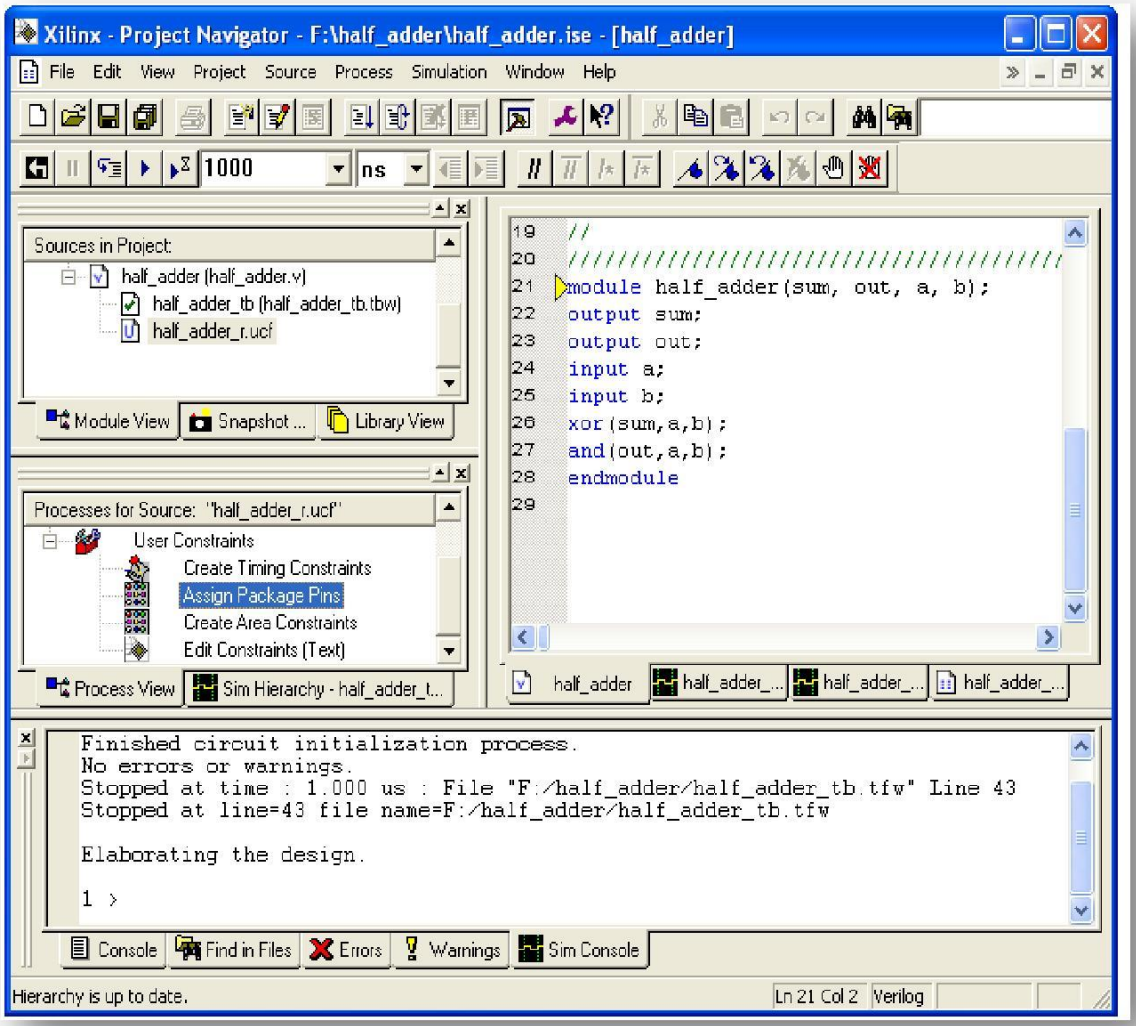


Step 9: **Create Constraints File (UCF)**

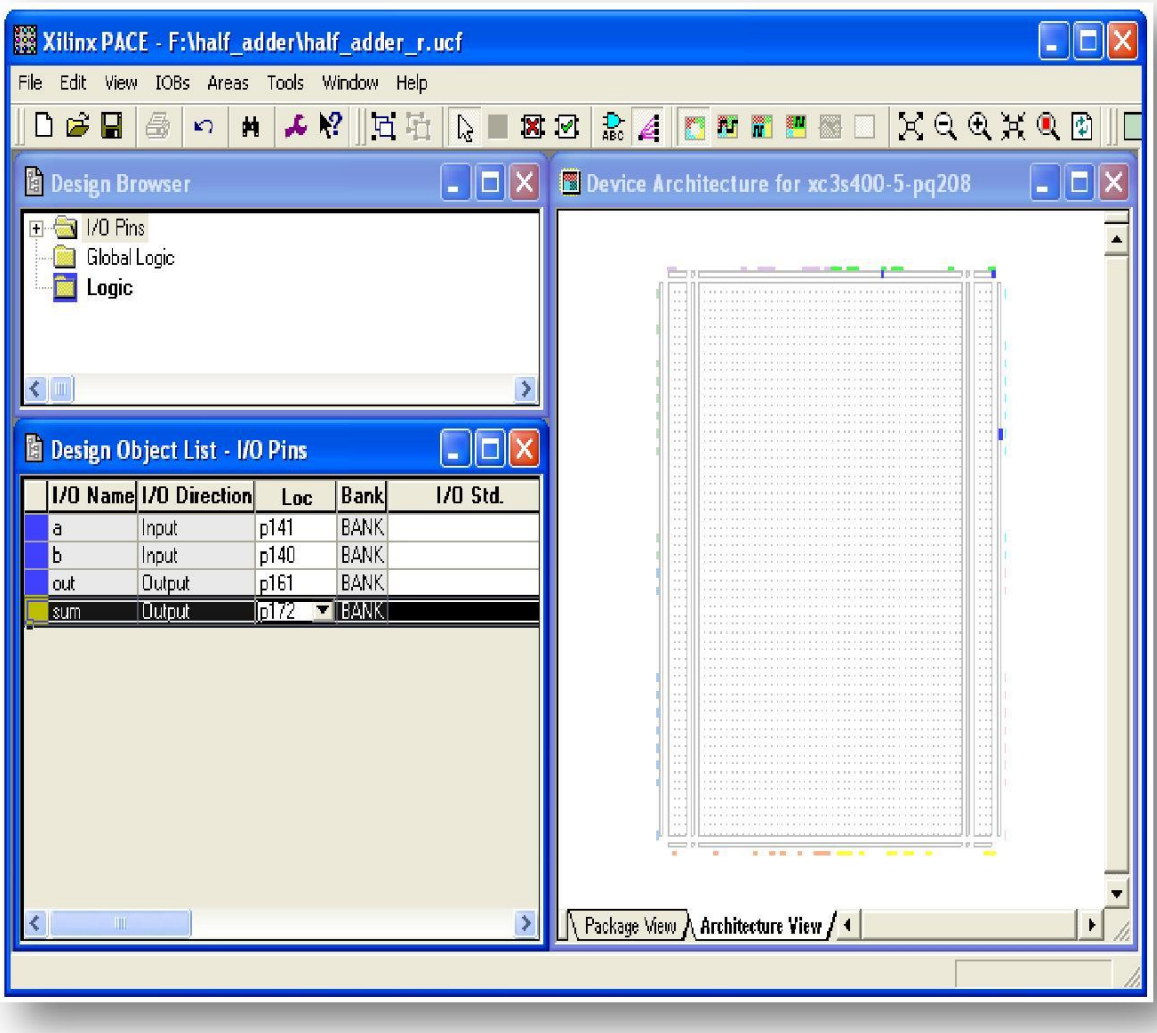
Click on the symbol of FPGA device and the *right click → Click on new source → Implementation Constraints File and give the name → Select entity → Finish.*



Click on User Constraint in processes for current source window and in that Double Click on *Assign Package Pins* option.



Xilinx PACE window opens. Enter all the pin assignments in PACE, depending upon the target device and number of input and outputs used in the design.



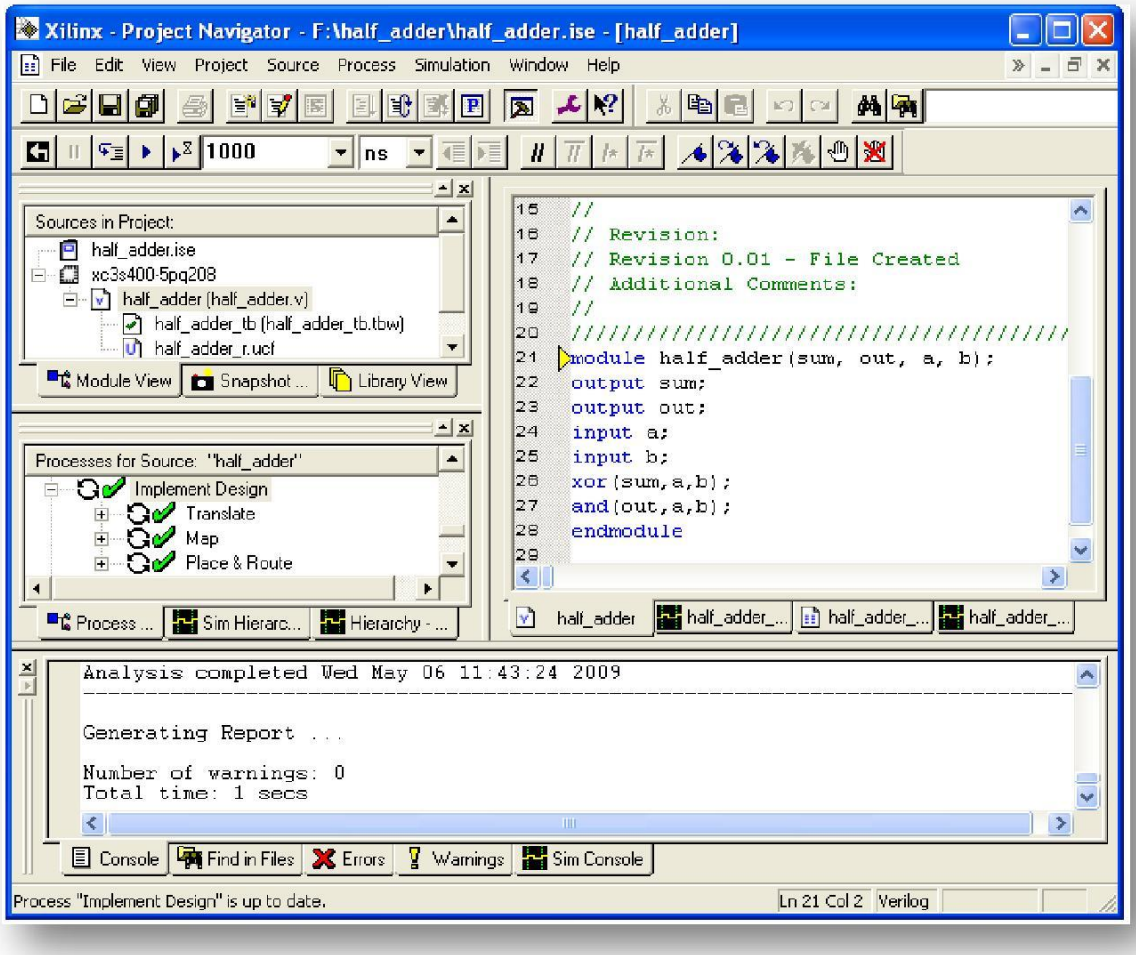
Step 10: **Implement the Design**

Once the synthesis is completed, the implementation stage consists of taking the synthesized netlist through translation, mapping, and place and route.

To check the design as it is implemented, reports are available for each stage in the implementation process.

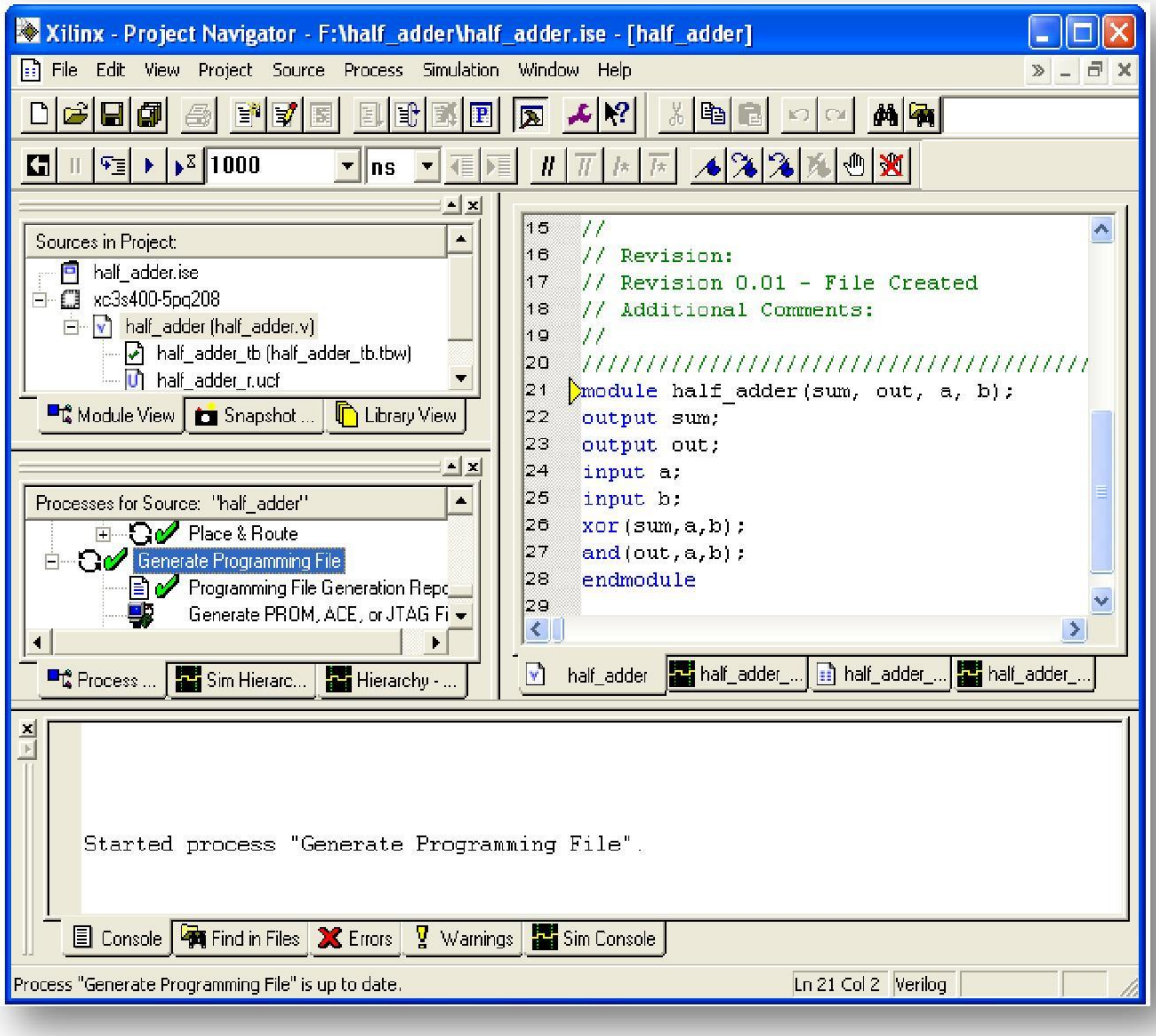
Use Xilinx constraints editor to add timing and location constraints for the implementation.

Right click on *Implement Design*, and choose Run option, *or* double left-click on *Implement Design.*



Step 11: **Generate the Programming File**

Right-click on *Generate Programming File* and choose the Run option, *or* double left-click on *Generate Programming File.* This will generate the Bit Stream.



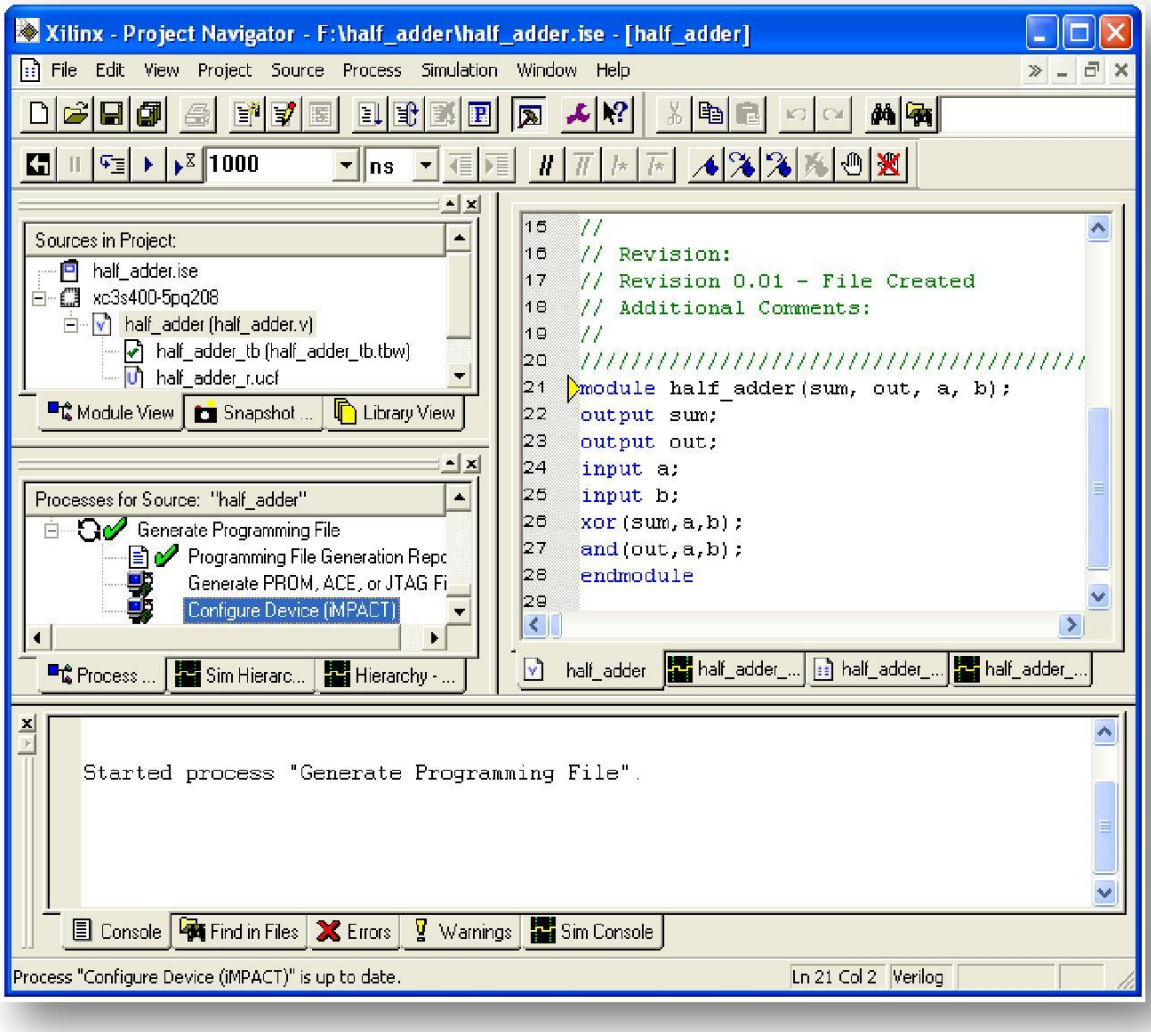
Step 12: **Download the Program in Boundary Scan Mode**

Double click on *"Configure Device (iMPACT)"*.

Right click on the device and Click on ***Program***.

If the device is programmed properly, it says *Programming Succeeded* or else *Programming Failed.* The DONE Led glows green if programming succeeds.

Note: Before downloading make sure that Protoboard is connected to PC's parallel port with the cable and power to the Protoboard is ON.



Step 13: **Test the Design**

Apply Input through DIP switches, output is displayed on LEDs.

Verify the behavior.