

Tutorial on
FPGA Design Flow
based on
Xilinx ISE WebPack
and ISim

ver. 2.1

Updated: Spring 2016

Software Requirements:

To use this tutorial, you need the following software:

GUI/Environment: Xilinx ISE ver. 14.6 or 14.7

VHDL Simulator: ISim

Synthesis: Xilinx XST 14.6 or 14.7 or Synplify Premier DP

Implementation: Xilinx ISE ver. 14.6 or 14.7

For instructions on how to set up Xilinx environment at school and home, please view the instructions posted at

<https://ece.gmu.edu/tutorials-and-lab-manuals>

Note: To set up Xilinx ISE Webpack at home, you will also need to acquire license for the webpack version you installed. To view detailed steps, please see the appendix of the tutorial “***Tutorial on Simulation with ISim***” posted at

<https://ece.gmu.edu/tutorials-and-lab-manuals>

Preparing the Input:

Download examples associated with this tutorial posted at

<http://ece.gmu.edu/tutorials-and-lab-manuals>

as **design_flow_examples.zip**

VHDL Source Files: Unzip the file “design_flow_examples.zip” and make sure that you obtain the folder **lab3_demo** with the following files:

RTL VHDL code:

clock_divider.vhd

counter.vhd

SSegCtrl.vhd

lab3_demo_package.vhd

lab3_demo.vhd

Testbench:

lab3_demo_tb.vhd

User Constraints File:

Select the ucf file based on the FPGA board you are using. For example, if you have Digilent Nexys 3 FPGA Board, then select **lab3_demo_nexys3.ucf**.

Throughout this tutorial we refer to the ucf file as **lab3_demo_ufc.ucf**. Make sure you use the file suitable for your FPGA board.

The current version of the tutorial was tested using the following tools:

Toolset

- Xilinx ISE WebPack Versions : 14.2, 14.6, 14.7

Synthesis Tool

- ISE Webpack Synthesis&Implementation Versions : 14.2, 14.6, 14.7
 - Synplify Premier DP

Implementation Tool

- Xilinx ISE/WebPack Version : 14.2, 14.6, 14.7

Simulation Tool

- ISim (Bundled with Xilinx ISE 14.2, 14.6, and 14.7)

Table of Contents

1.	Project Settings	5
2.	Behavioral Simulation	12
3.	Synthesis and Implementation	13
3.1	Synthesis with Xilinx XST	13
3.2	Synthesis with Synplify Premier DP	17
3.3	Translate	17
3.4	Post-Translate Simulation	18
3.5	Map	19
3.6	Place and Route	21
3.7	Post Place and Route Simulation	21
3.8	Implementation Reports	22
4.	Specifying Frequency/ Time period using User Constraints	24
5.	Optimization strategies	28
6.	Pin Assignment	32
7.	Bit Stream Generation	35
8.	Uploading Bitstream to FPGA Board	36
	Appendix	38

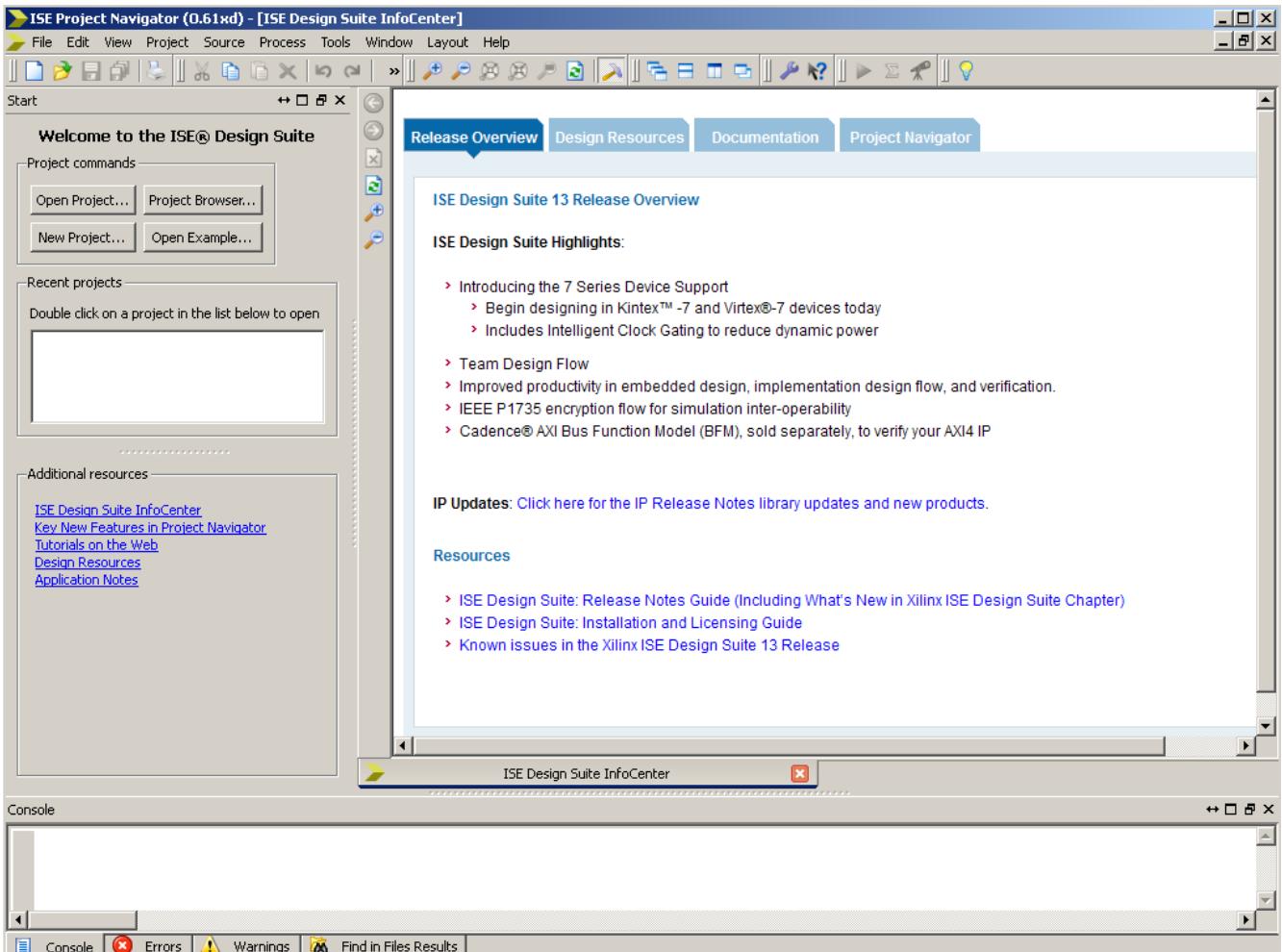
1. Project Settings

To Start Xilinx ISE in the ECE Labs go to:

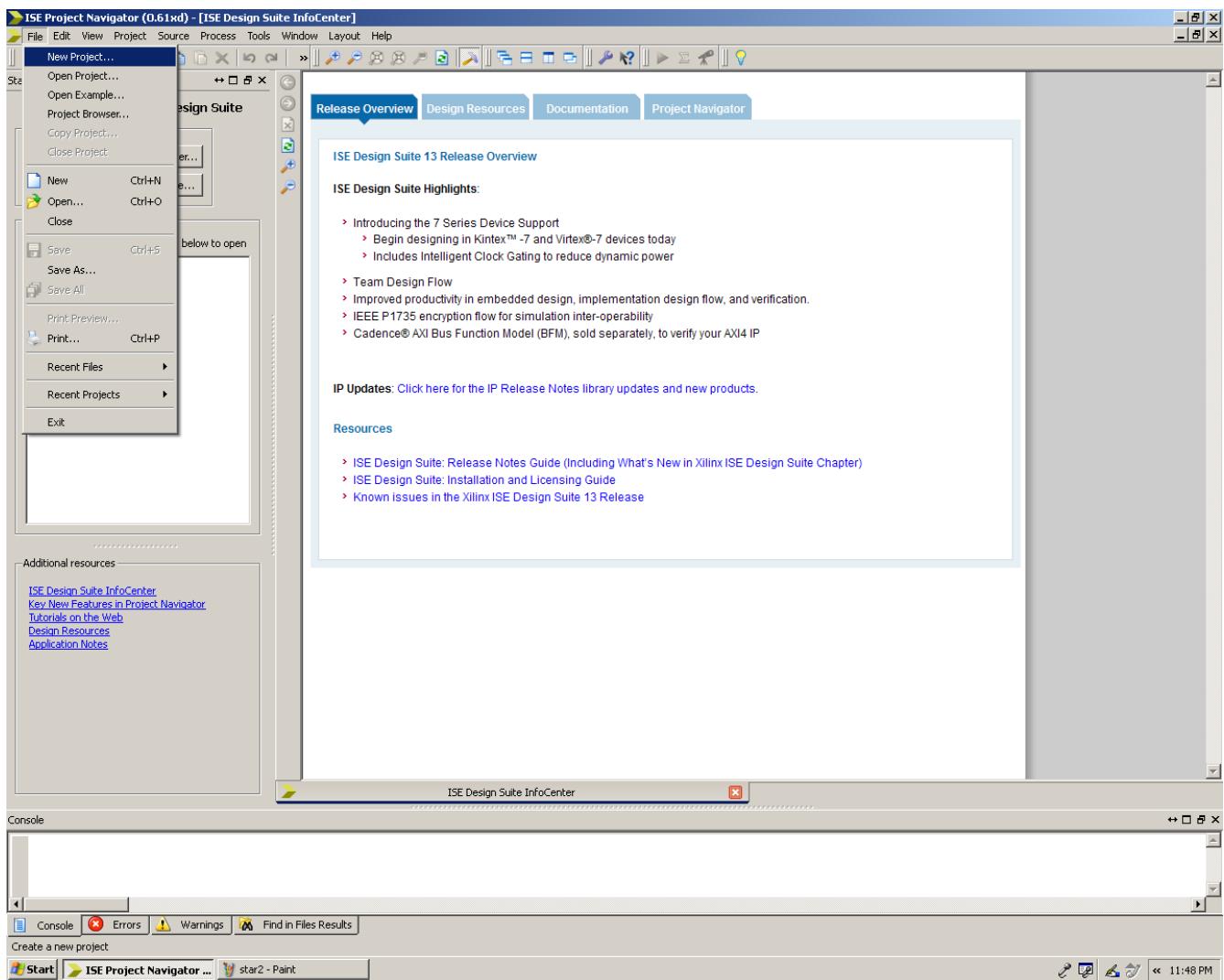
Start->All Programs-> VLSI Tools->Xilinx ISE <version-number>->ISE Design Tools ->Project

Navigator

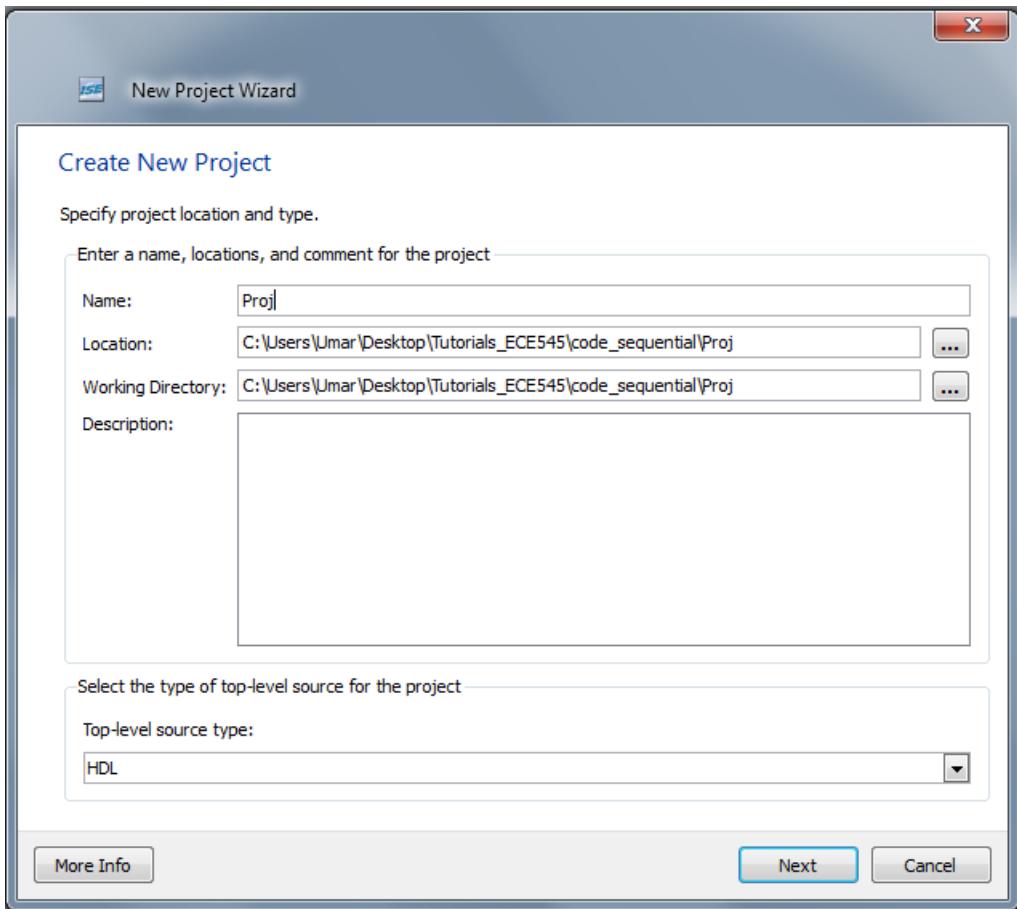
At home, adjust the path accordingly based on your installation.



To start a new Project go to the menu File->New Project



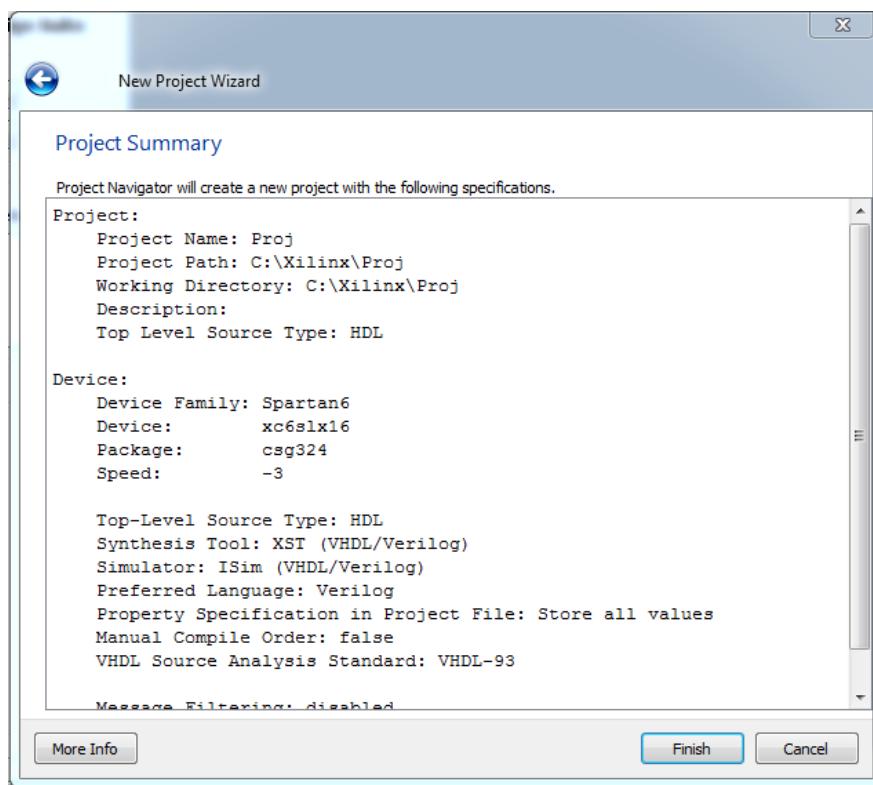
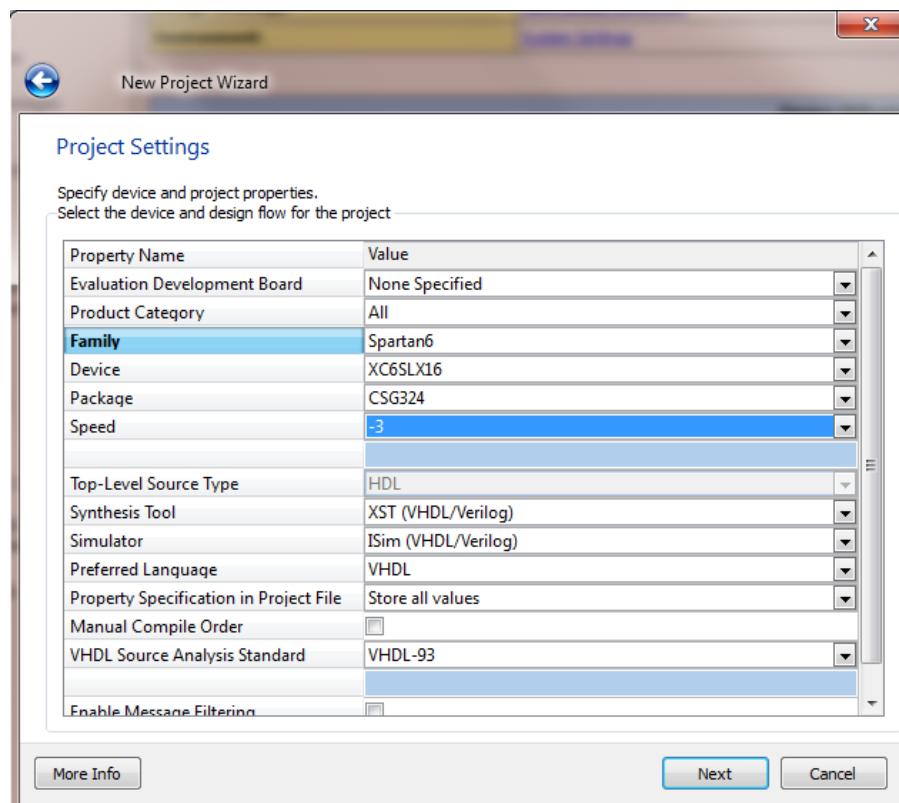
Specify Project Name and location for project files



Click **Next** and specify

For Nexys 3

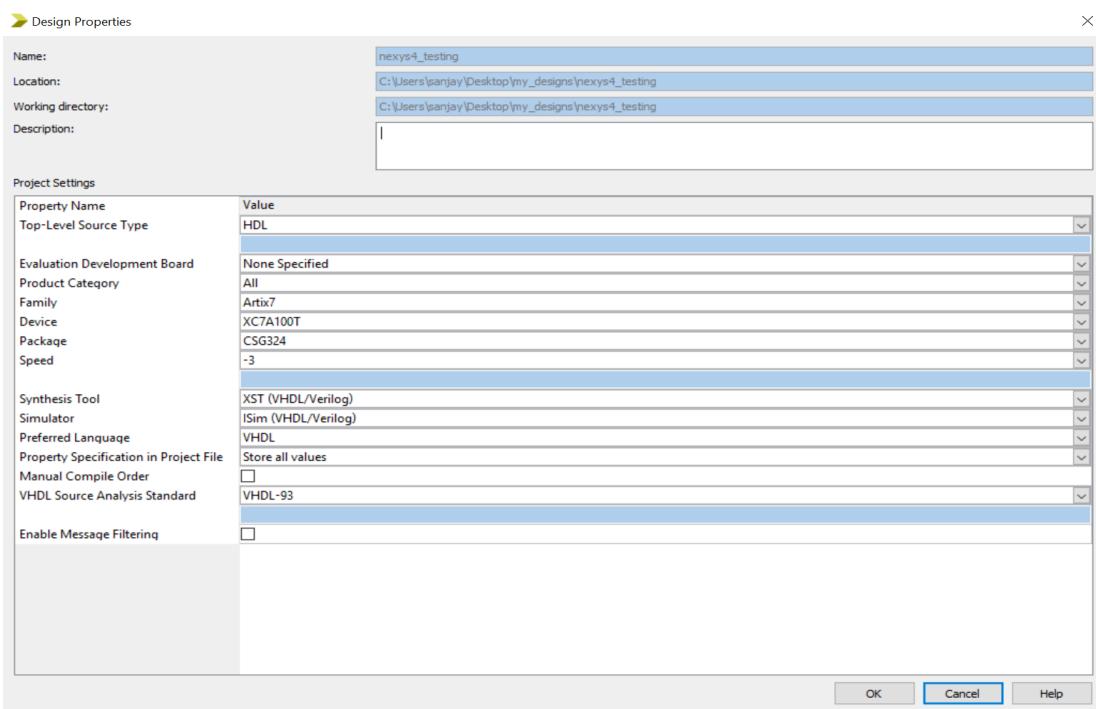
Family	:	Xilinx <version number>x Spartan6
Device	:	XC6SLX16
Package	:	CSG324C
Speed Grade	:	-3
Simulator	:	ISim



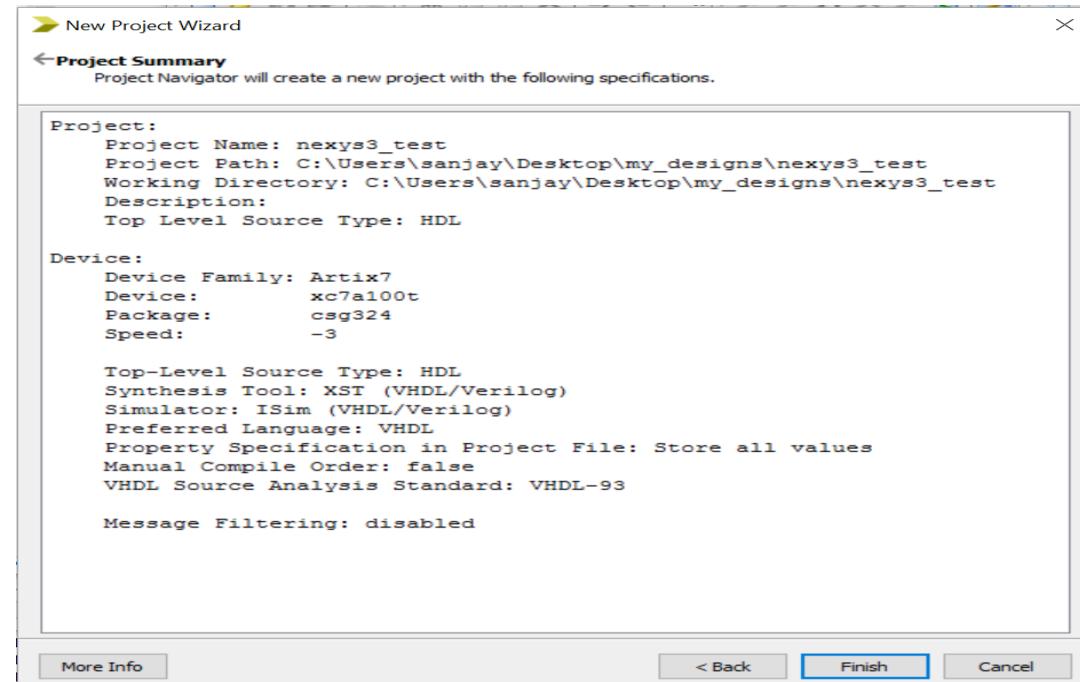
Click Finish

For Nexys 4

Family : Xilinx <version number>x Artix7
Device : XC7A100T
Package : CSG324AB
Speed Grade : -3
Simulator : ISim



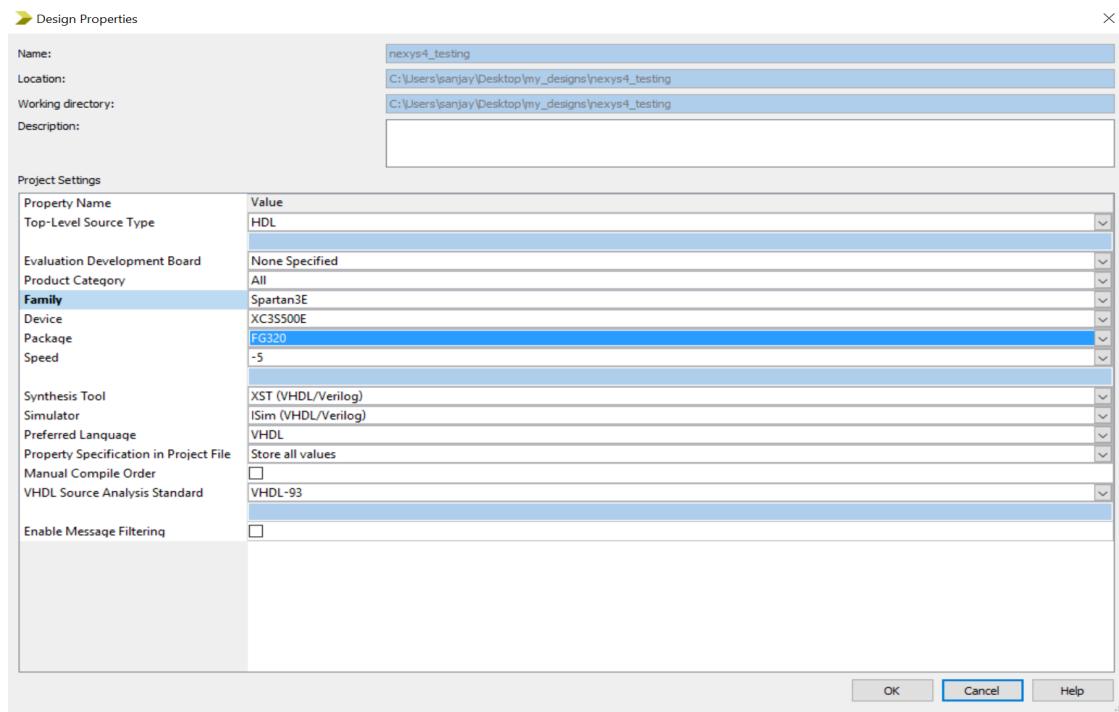
Click OK



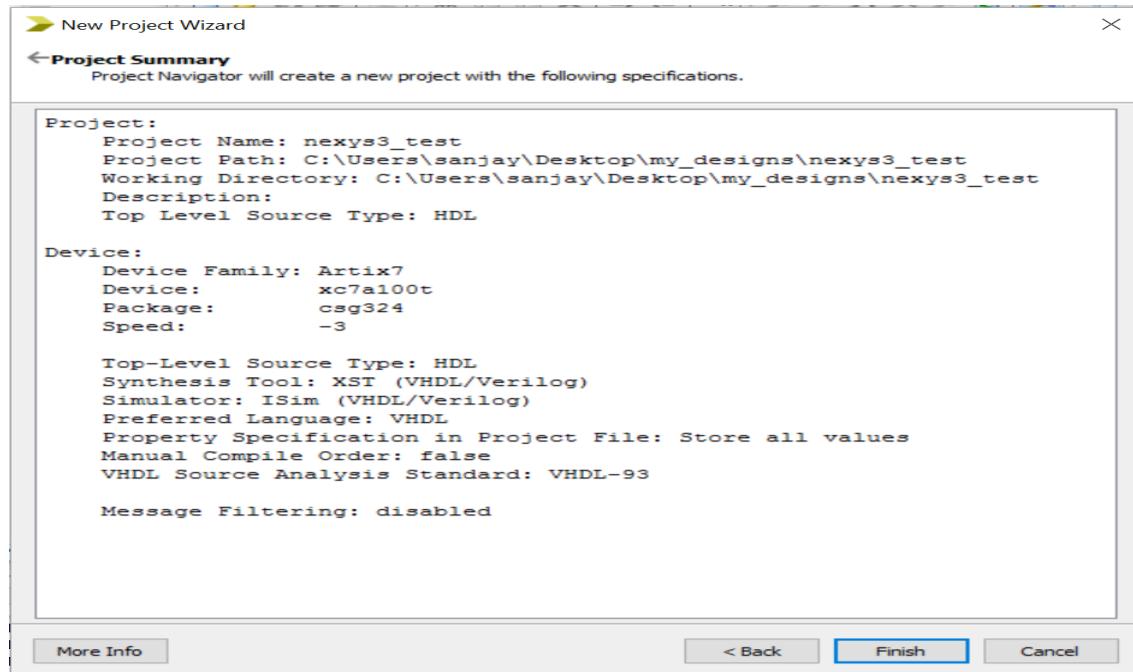
Click Finish

For Basys 2

Family : Xilinx <version number>x Spartan3E
Device : XC3S500E
Package : FG320
Speed Grade : -5
Simulator : ISim

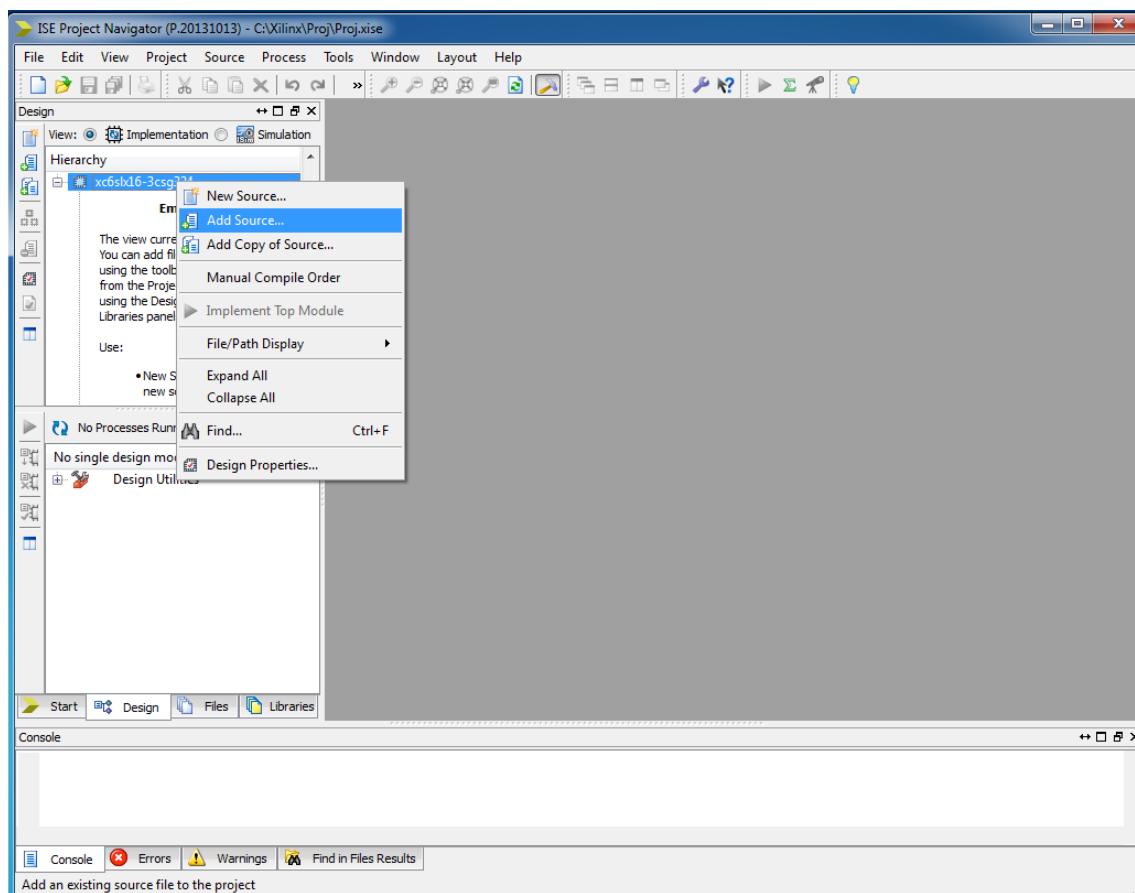


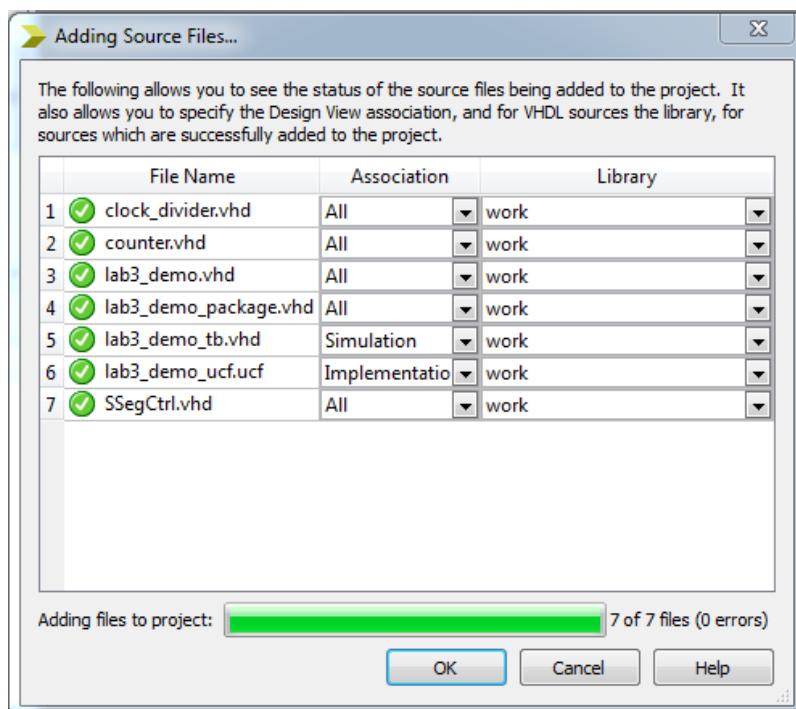
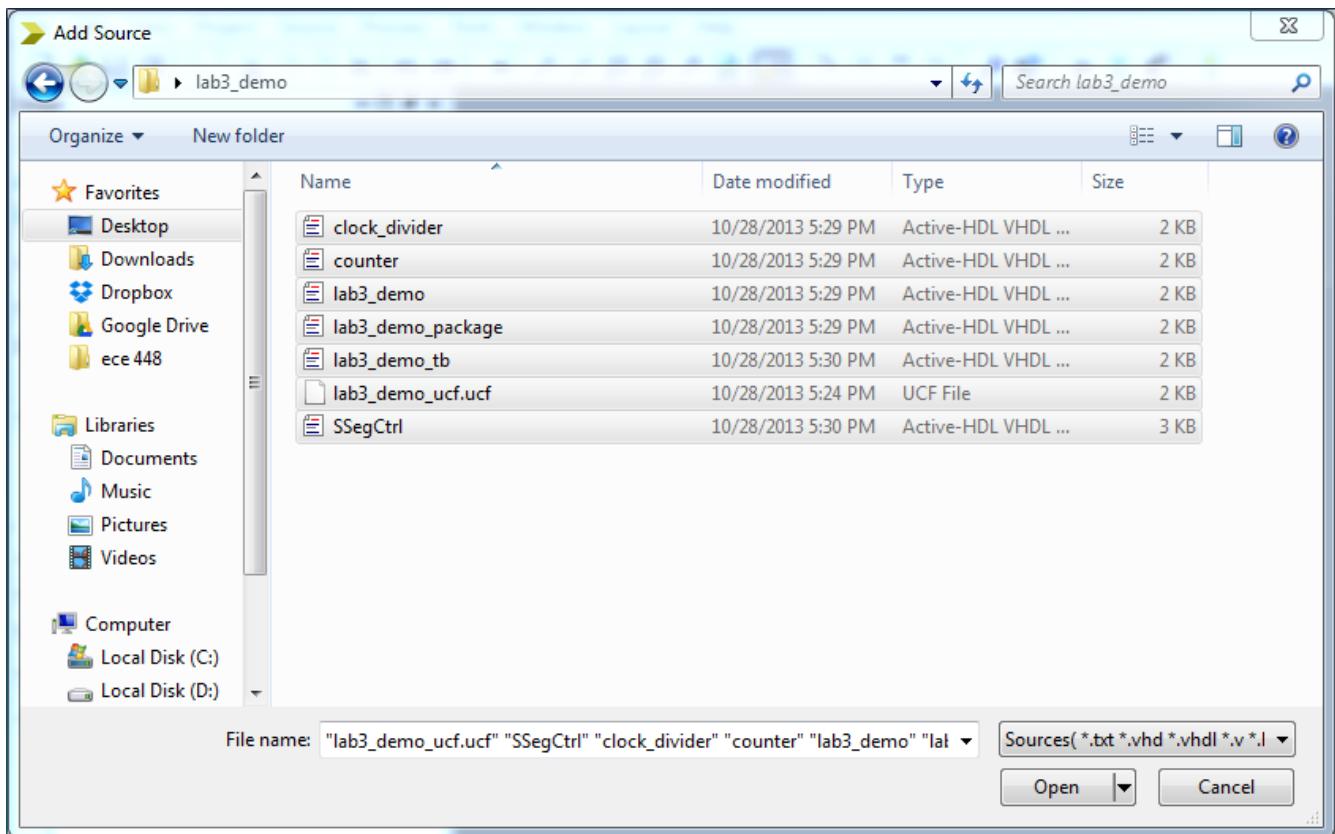
Click OK



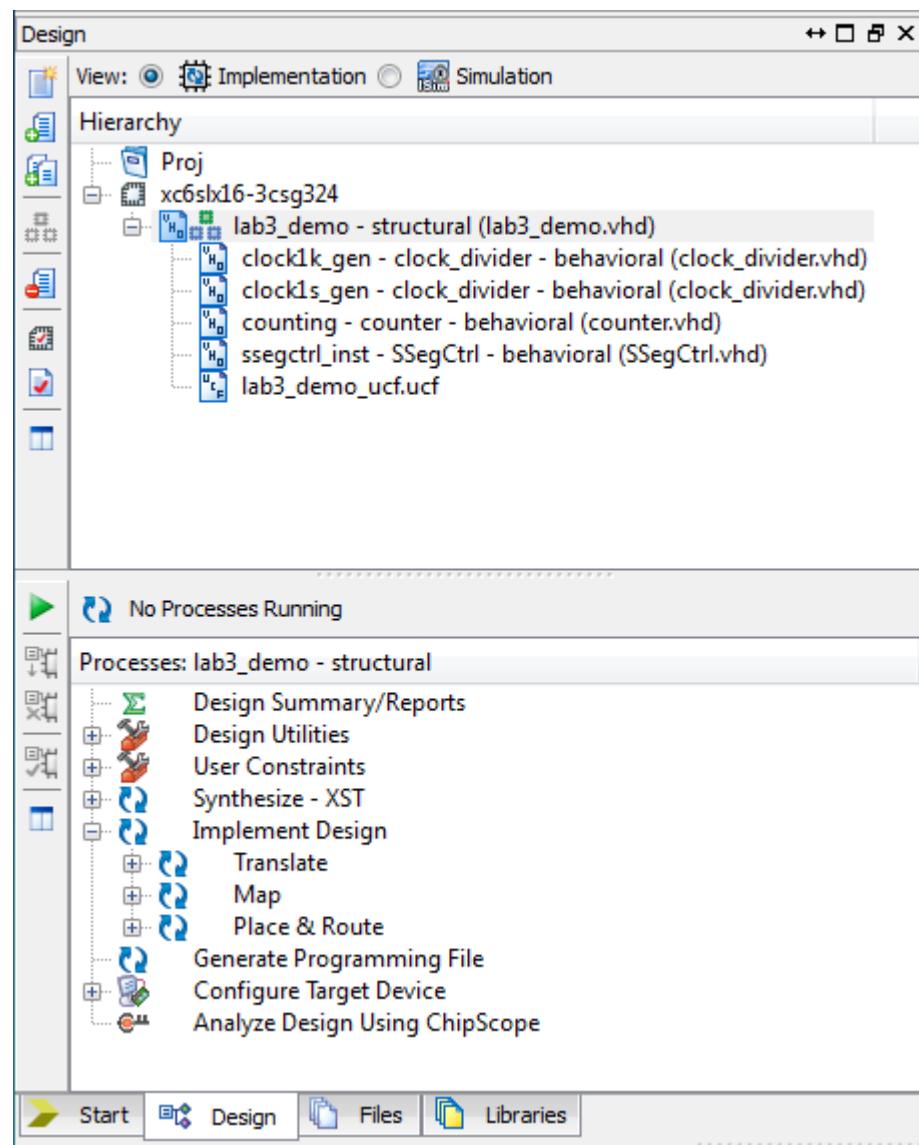
Click Finish

To add VHDL sources to the design, right click at the name of the FPGA device (e.g., “xc6slx16-3csg324” for Nexys 3) under Hierarchy and select **Add Source**. Choose files for the project.





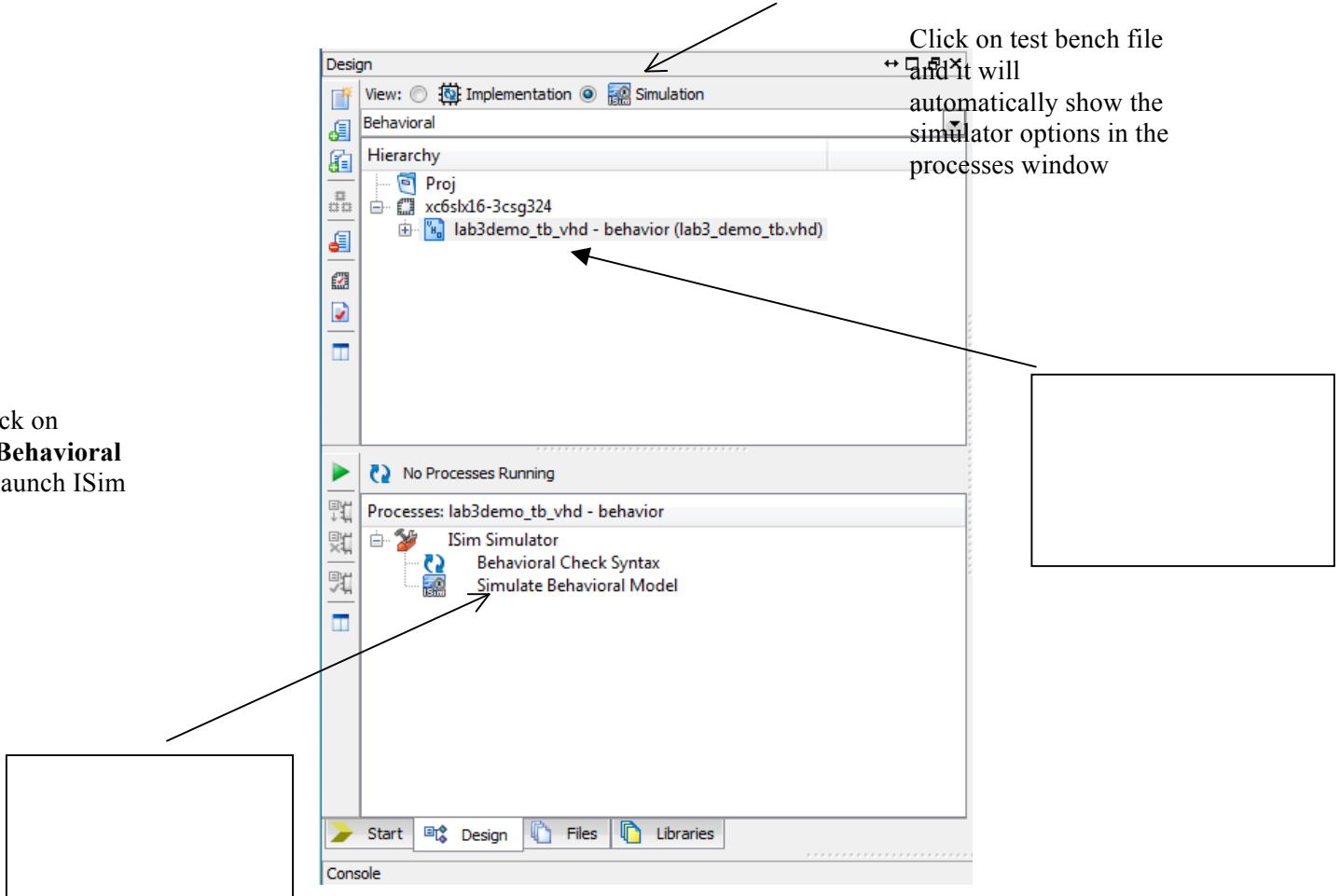
Then click **OK**. Your project has been defined.



2. Behavioral simulation

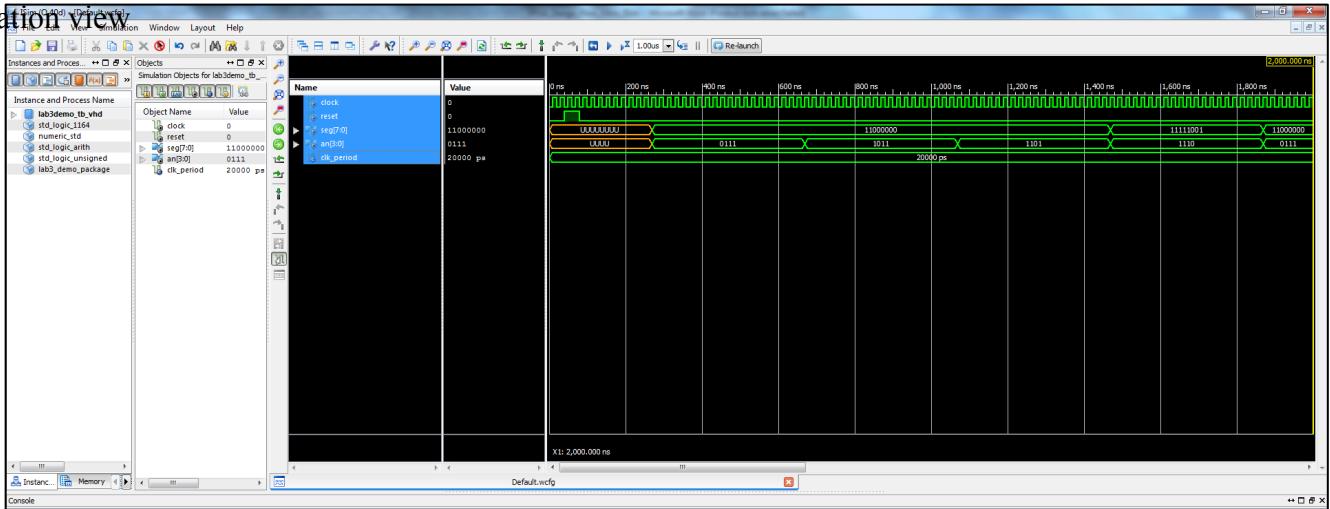
Change the view from Implementation to Simulation by clicking on the radio button next to **Simulation**. In the Design Menu we choose **Behavioral Simulation**. The testbench for this design was set as a top level. In the Processes menu choose **ISim Simulator**, **Simulate Behavioral Model**, right-click and choose **Run**.

Double click on
**Simulate Behavioral
Model** to launch ISim
simulator



It will launch ISim simulator automatically and waveform for behavioral simulation will appear.

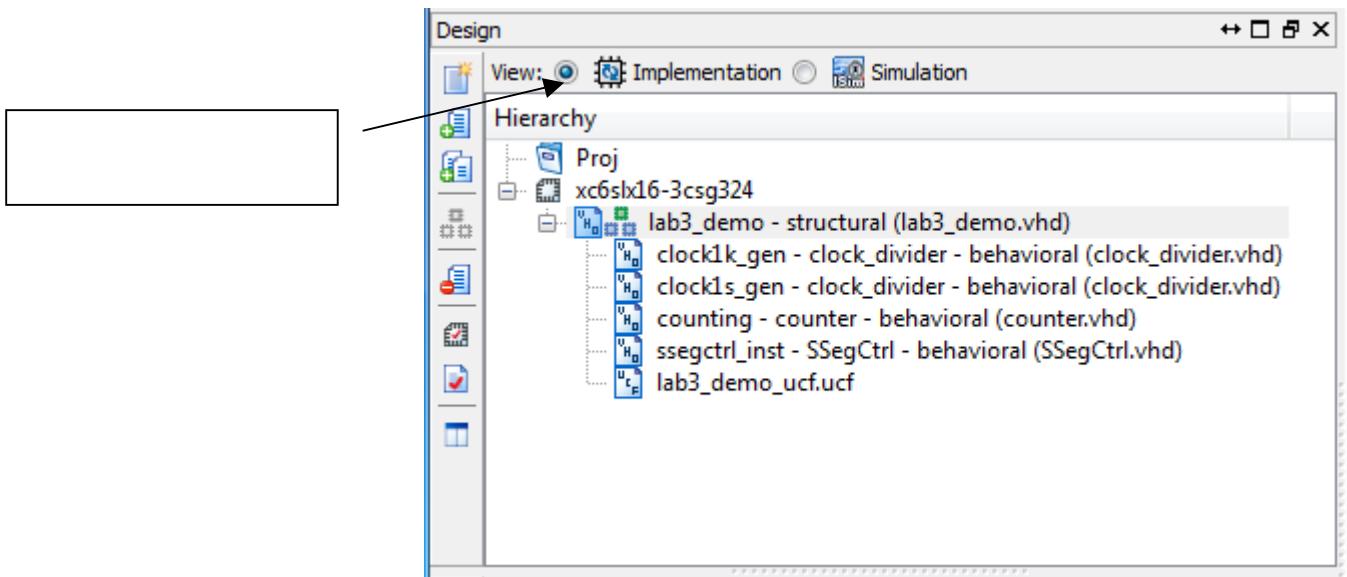
click here to switch to Implementation view.



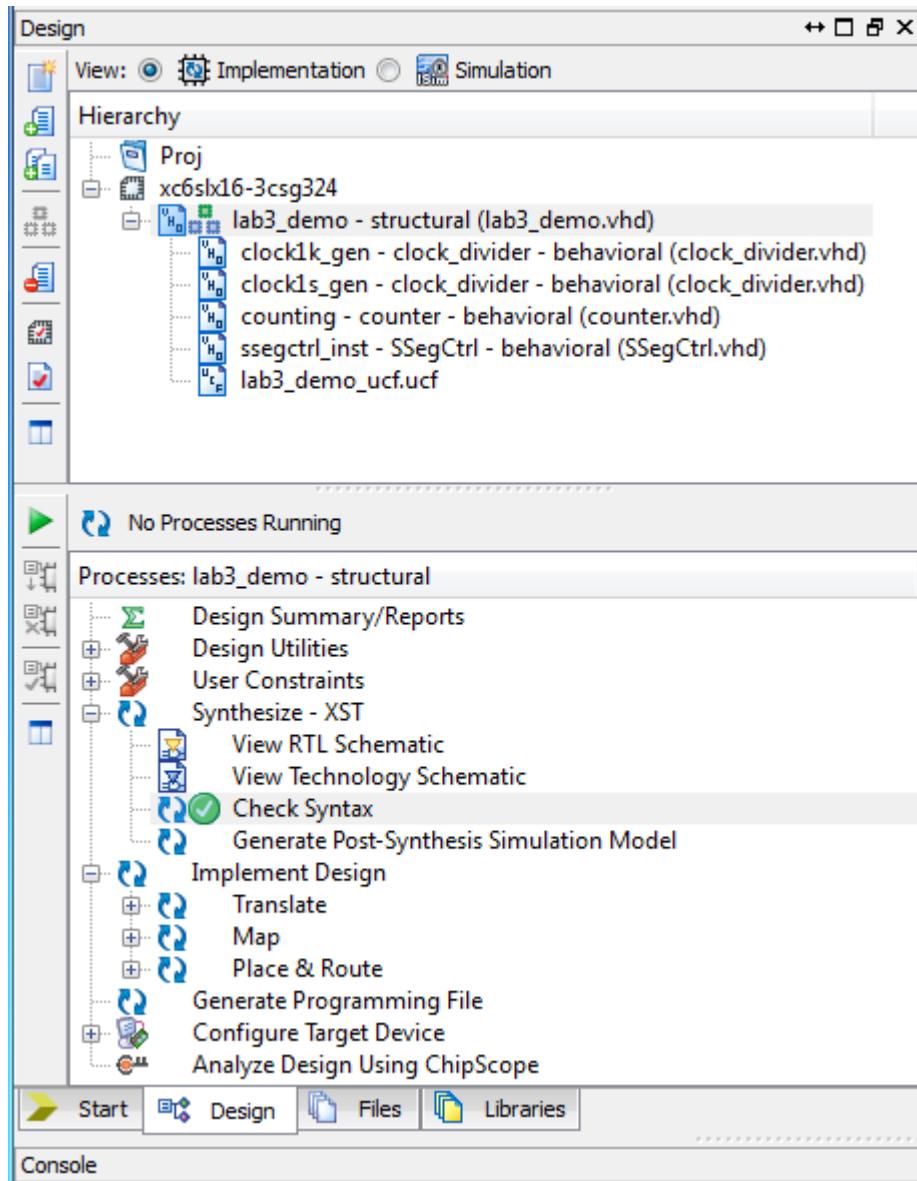
3. Synthesis

3.1 Synthesis with Xilinx XST

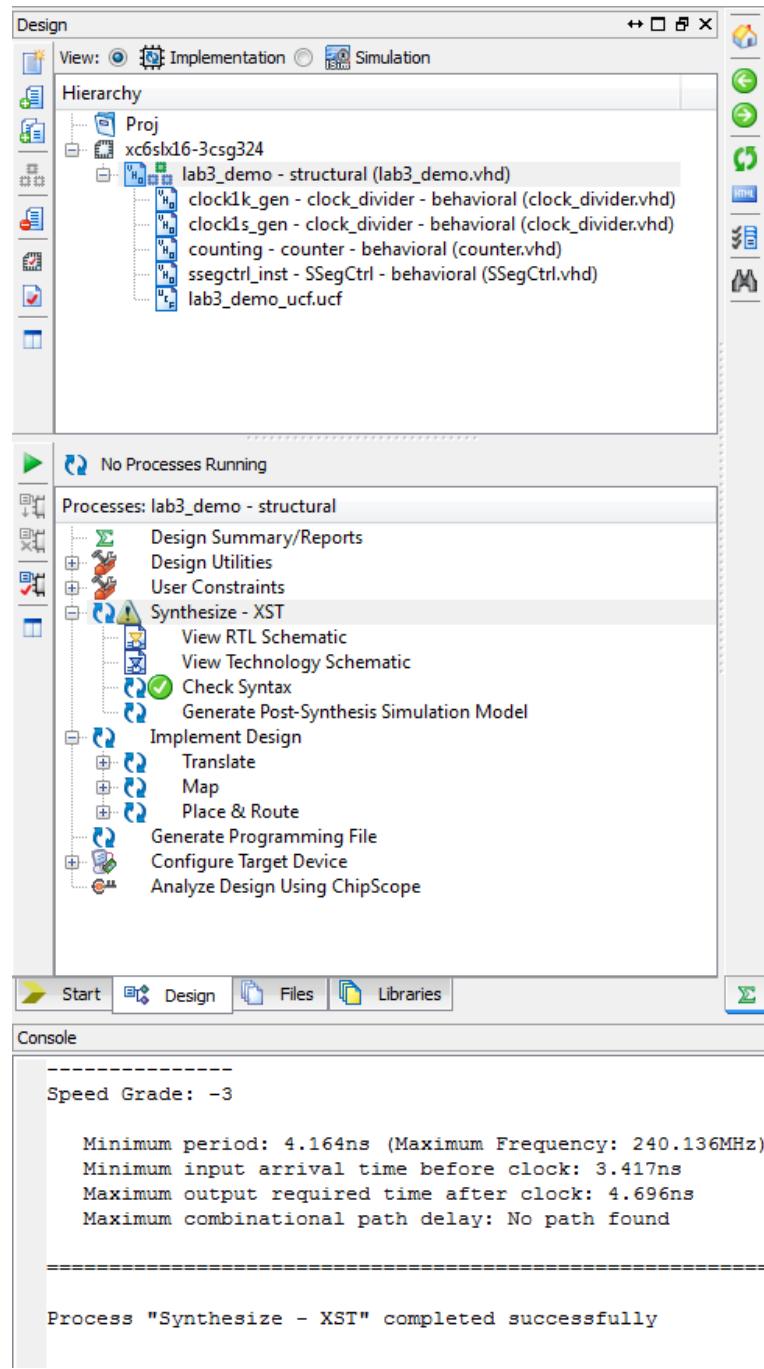
Change the view from Simulation to implementation by clicking on the radio button next to **Implementation**.



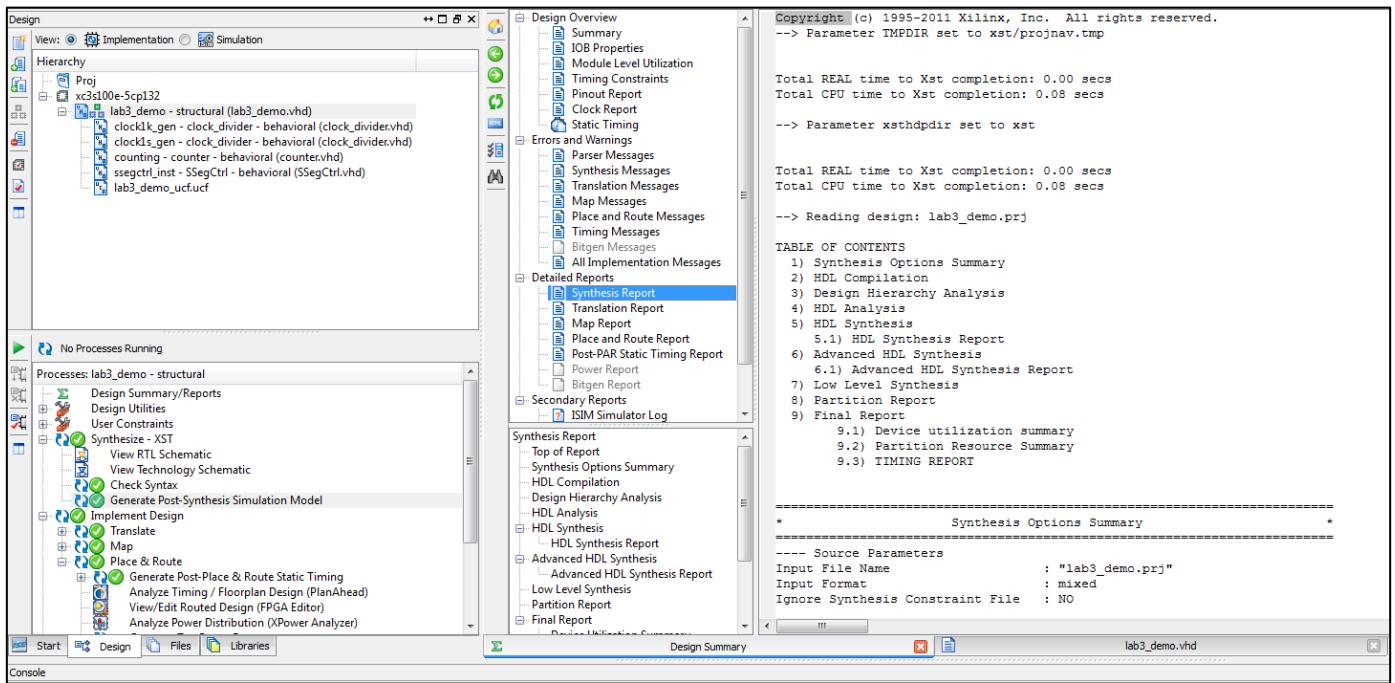
Click on your top-level file (lab3_demo in our case). It will show synthesis and implementation options in the processes window. Click **Check Syntax** under the synthesis option to check if VHDL sources are properly coded.



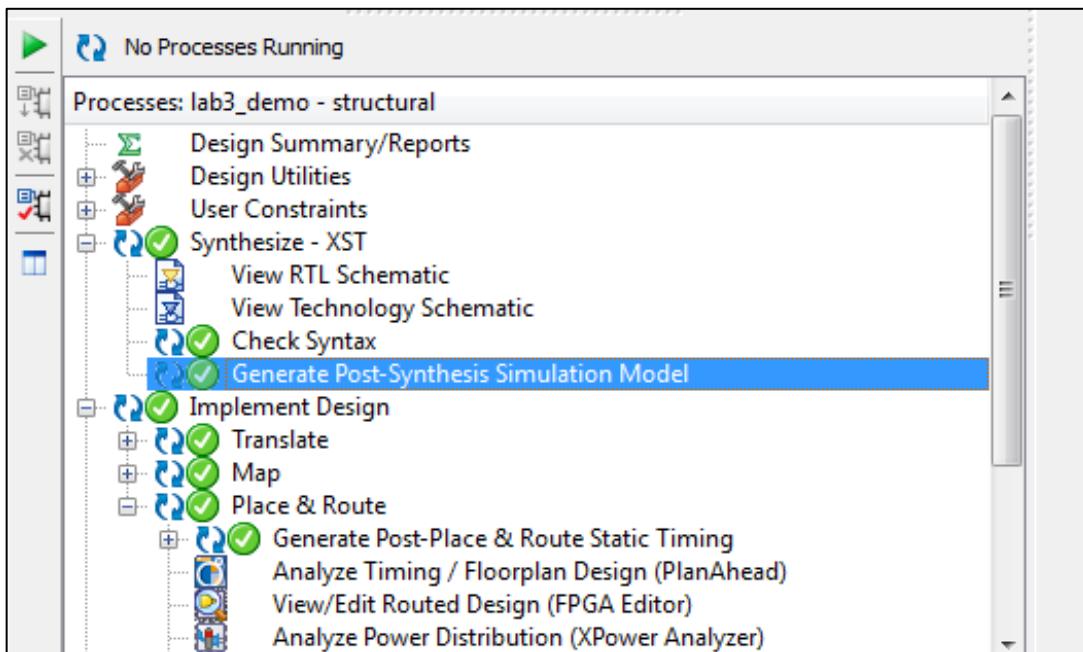
After the syntax check is done, double click on **Synthesize - XST** option to synthesize your design. A green tick will appear showing that the synthesis is successfully completed.



When Synthesis process is completed, the report from synthesis becomes available. Some of the most important features of the design are the minimum clock period and the maximum clock frequency. We can find these two parameters in the report file from Synthesis. Please remember that the values of these parameters after synthesis are different than the values of the same parameters after implementation. The other important information we can find in this report is the amount of FPGA resources your design requires.



Generate post-synthesis simulation model by clicking on the option available under synthesis.

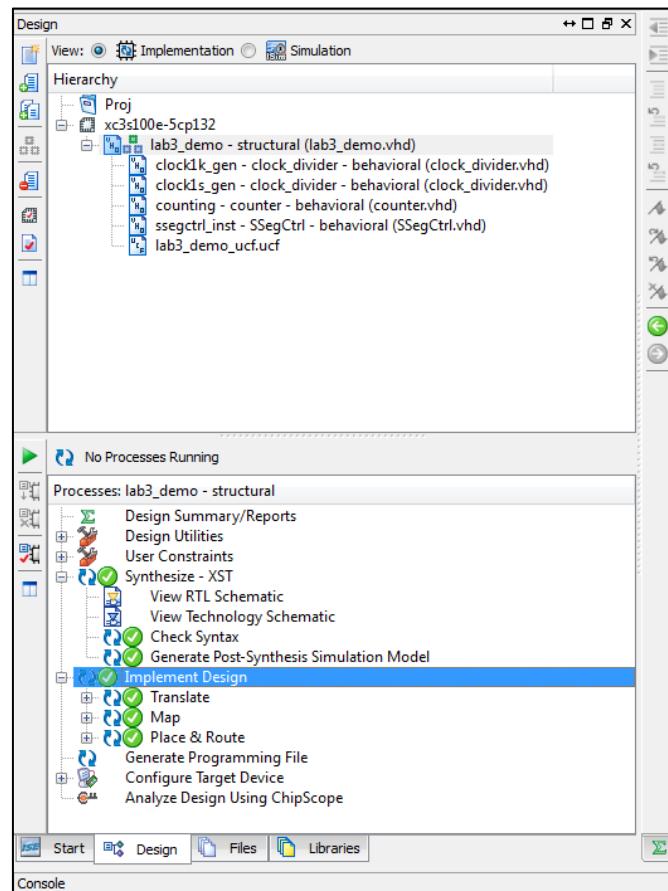


After Synthesis and generation of Post-Synthesis Simulation Model were completed successfully, we can start Implementation part.

3.2 Synthesis with Synplify Premier DP

Synthesis can also be performed using Synplify Premier DP. For detailed instructions, please see the appendix at the end.

Implementation: Double click on the implementation menu to start the implementation. It will automatically do translate, map, and place & route one after the other. Make sure to see the green tick accompanying all tasks to show that they are successfully completed.



You can individually run each task (translate, map and place & route) one by one to better understand the design flow.

3.3 Translate

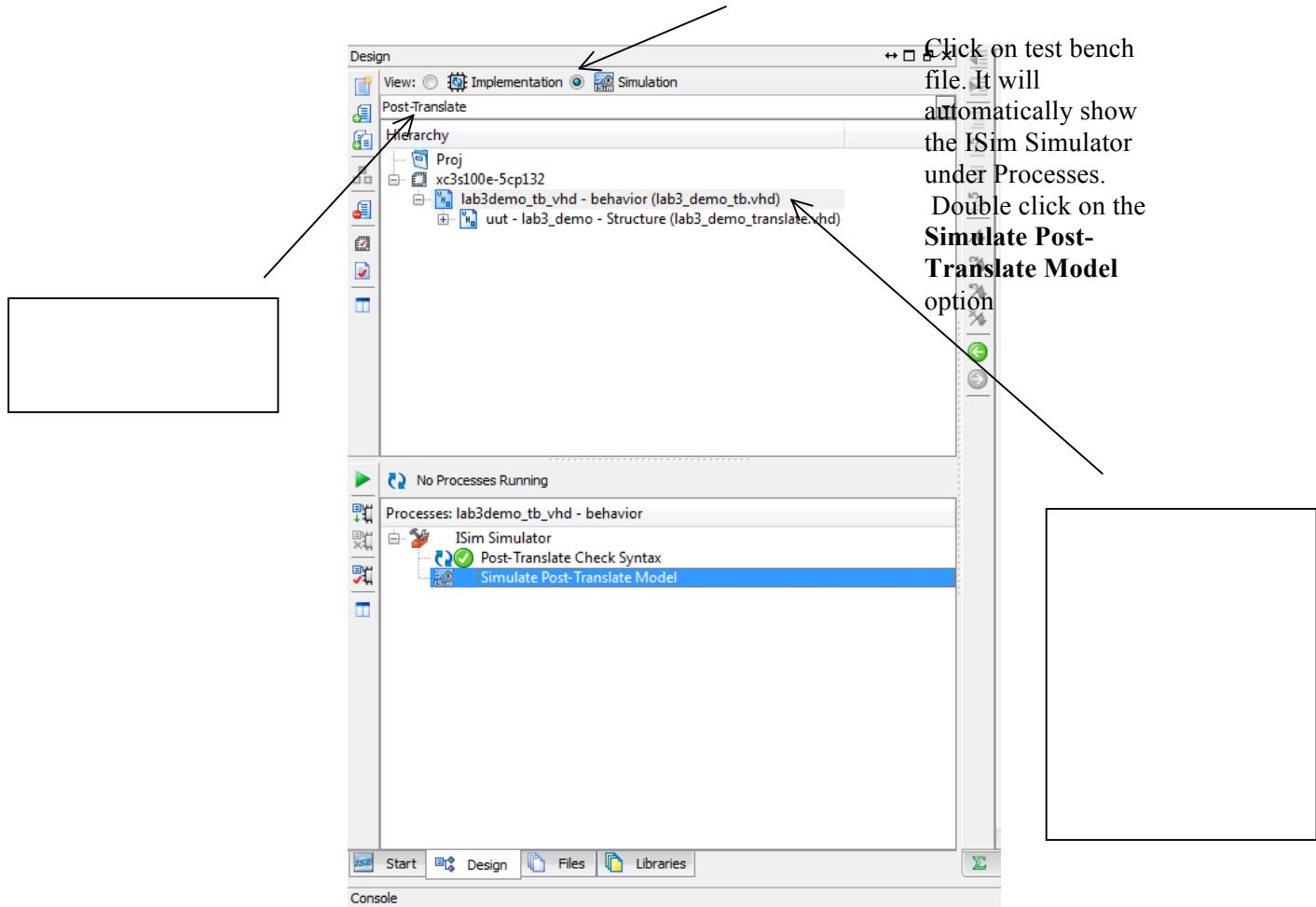
The first step in the Xilinx Design Flow for implementation is Translate. Under **Implement Design** option, choose **Translate**, and then **Run**.

If you are successful with this part you should generate Post Translate Simulation Model.

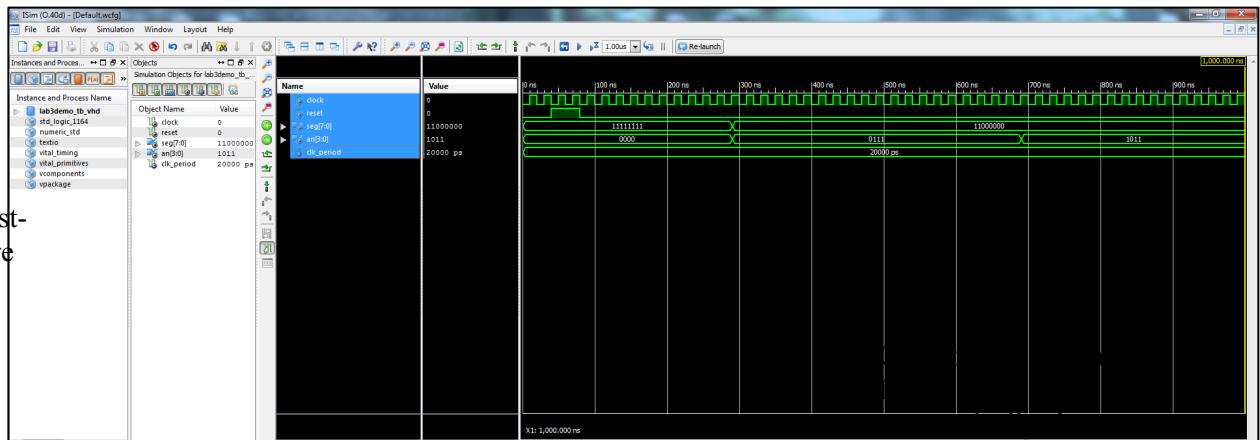
Change Behavioral Post-Translate Simulation

Post-Translate

option here Double-click on **Generate Post-Translate Simulation Model** under Translate option. Change the view from **Implementation to Simulation** by clicking on the radio button next to **Simulation**.



Double click on the Simulate Post-Translate Model option. ISim simulator will be launched and waveforms for the post-translate simulation will appear.

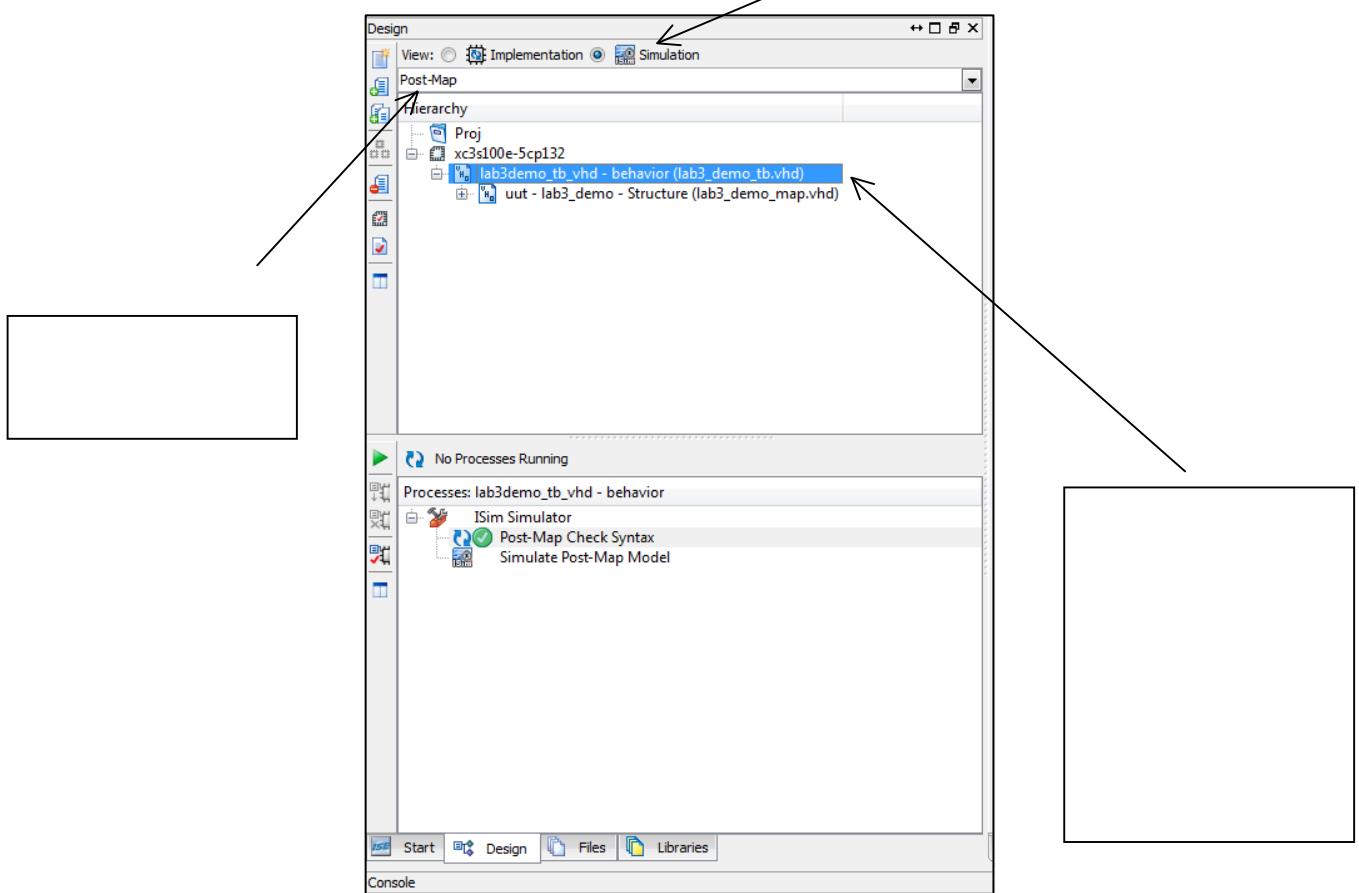


3.5 Map

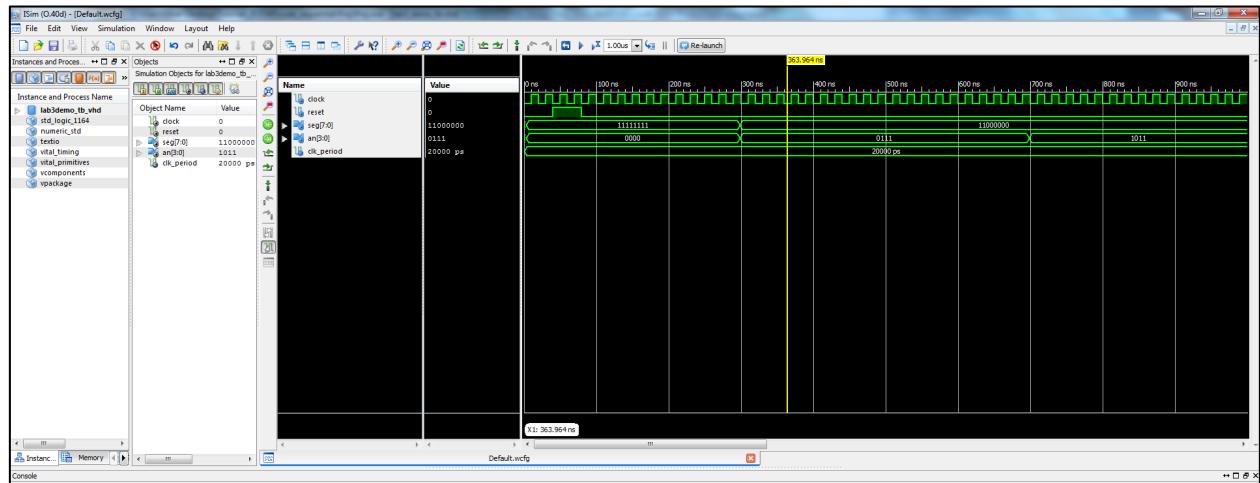
Change the view from **Simulation** to **Implementation** by clicking on the radio button beside **Implementation**. Under **Implement Design** option, double click on **Map** to start mapping process.

Post-Map Simulation

When you right-click on **Generate Post-Map Simulation Model** then pop-up menu appears. Choose **Run** to start. Change the view from Implementation to Simulation by clicking on the radio button beside **Simulation**.



Double click on the Simulate Post-Map Model option. ISim simulator will be launched and waveforms for post-map simulation will appear.

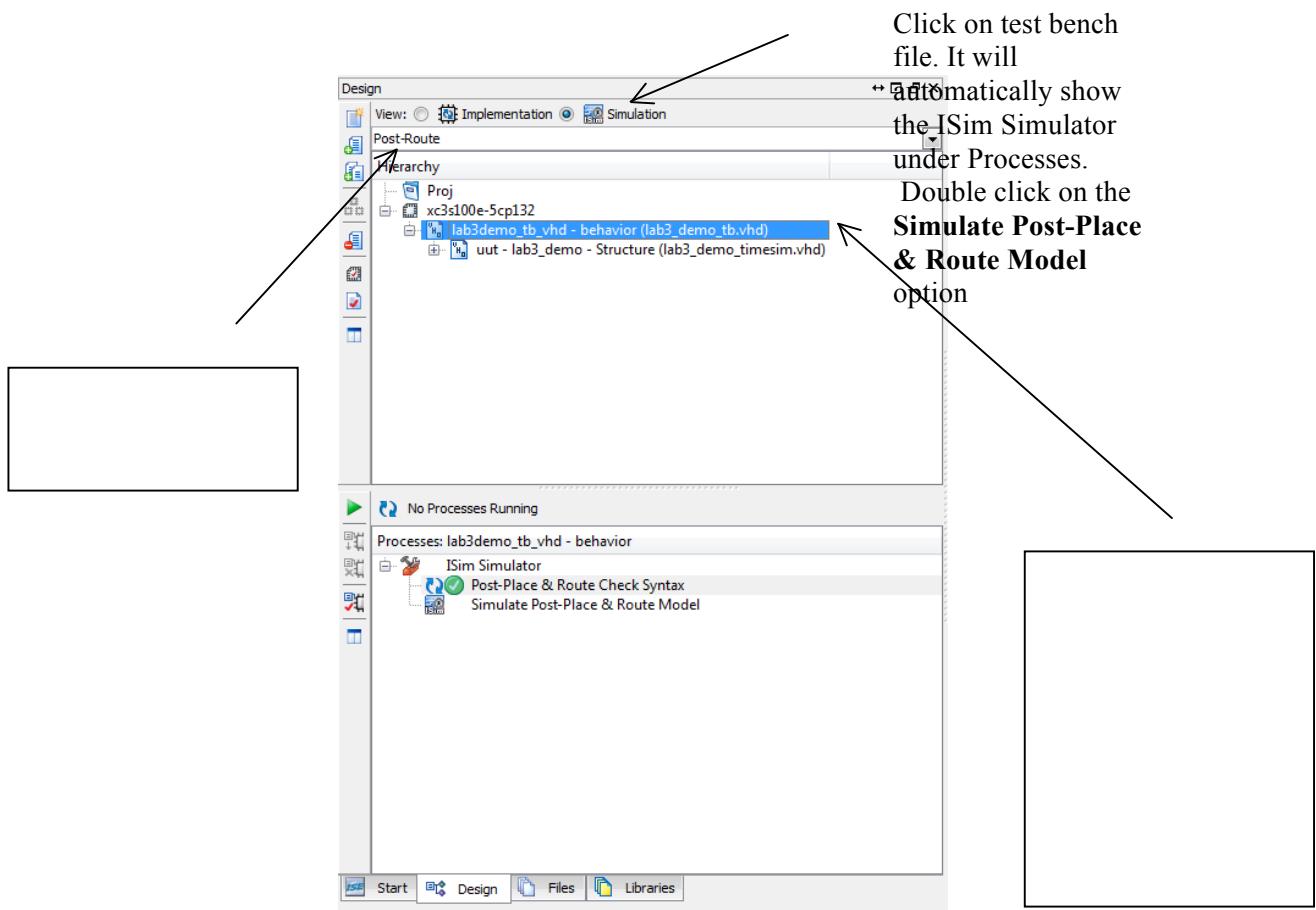


3.6 Place and Route

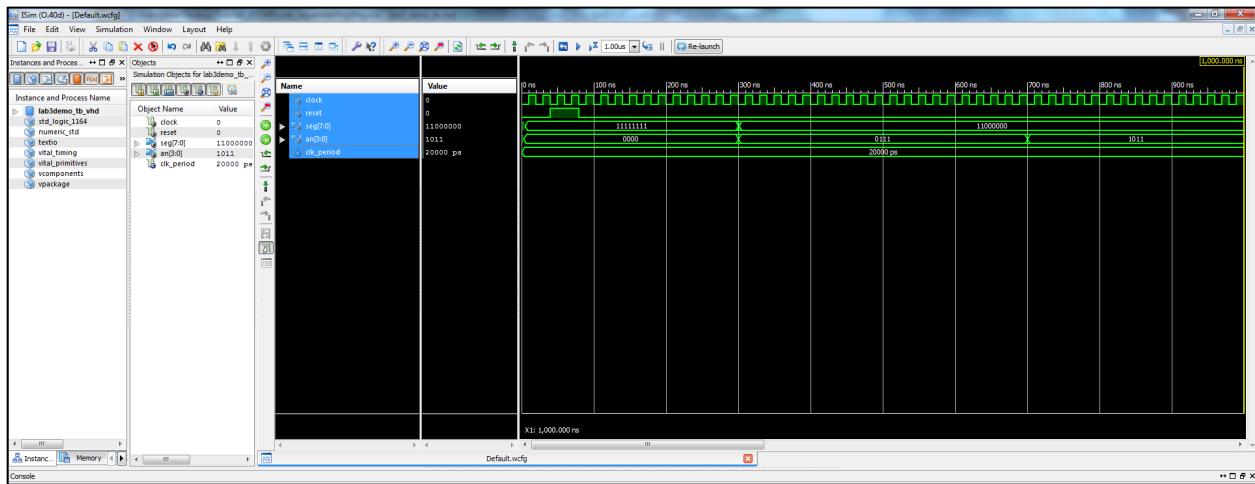
Click on radio button beside **Implementation** to switch to implementation mode. From the **Implement Design** menu, choose **Place and Route**. Double click to start place and route process. After that, double click on **Generate Post Place and Route Simulation Model** to run it.

Change Post-Map to
Post-Route option

here Double click on Generate Post Place & Route Model, then pop-up menu appears to run it. Change the view from Implementation to Simulation by clicking on the radio button beside Simulation.

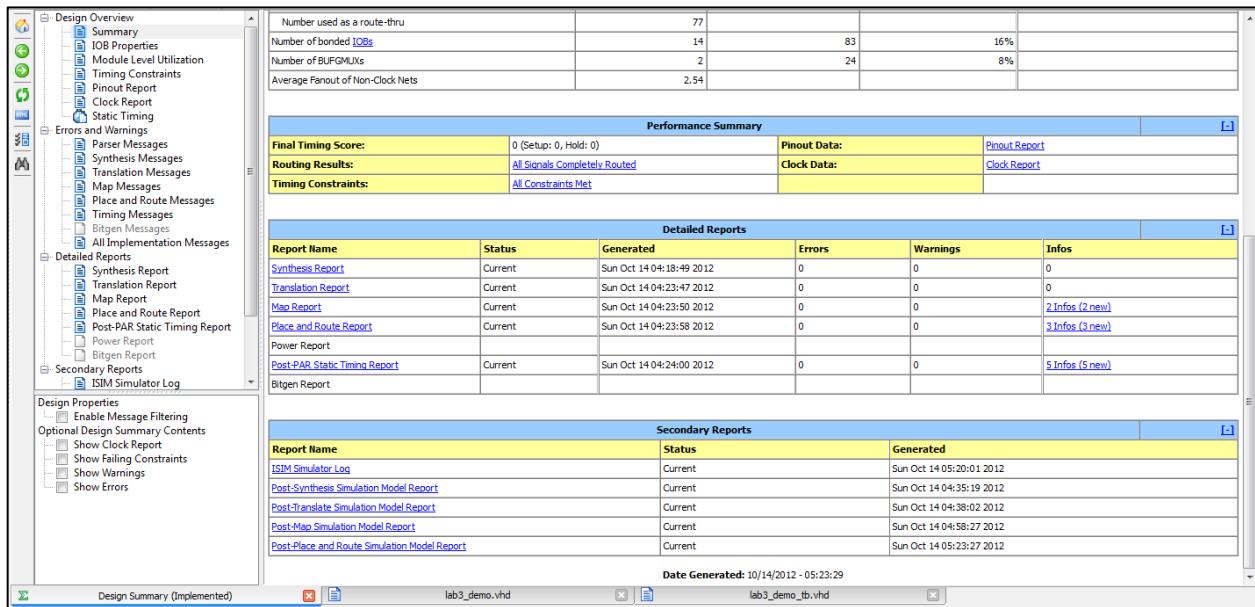


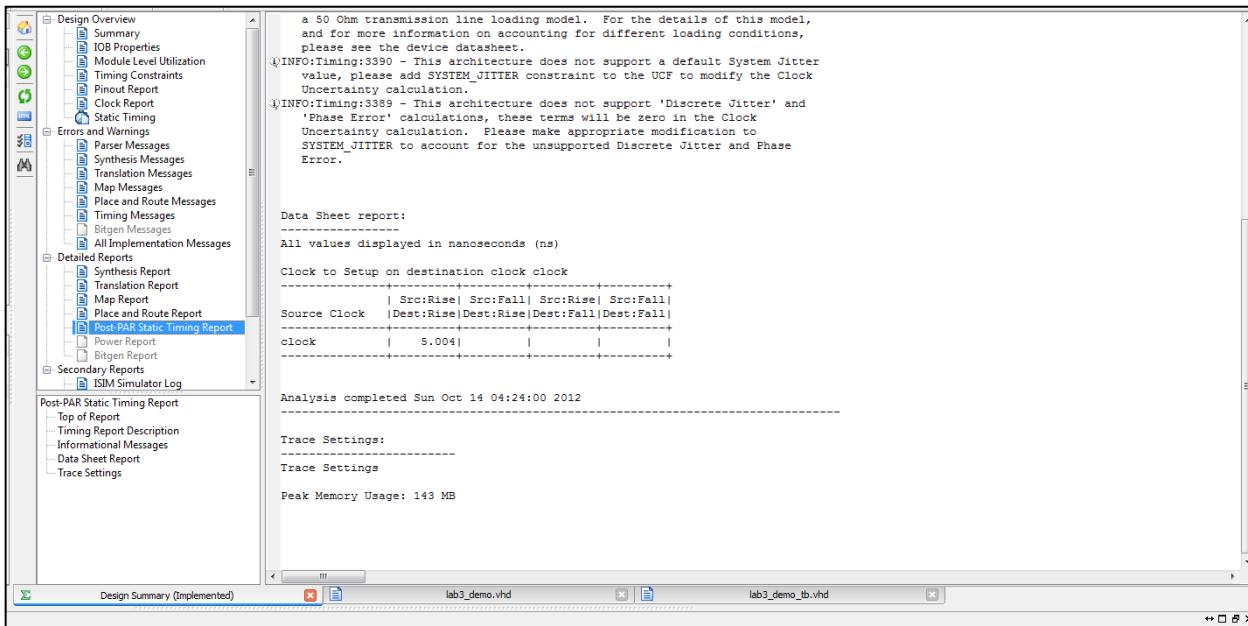
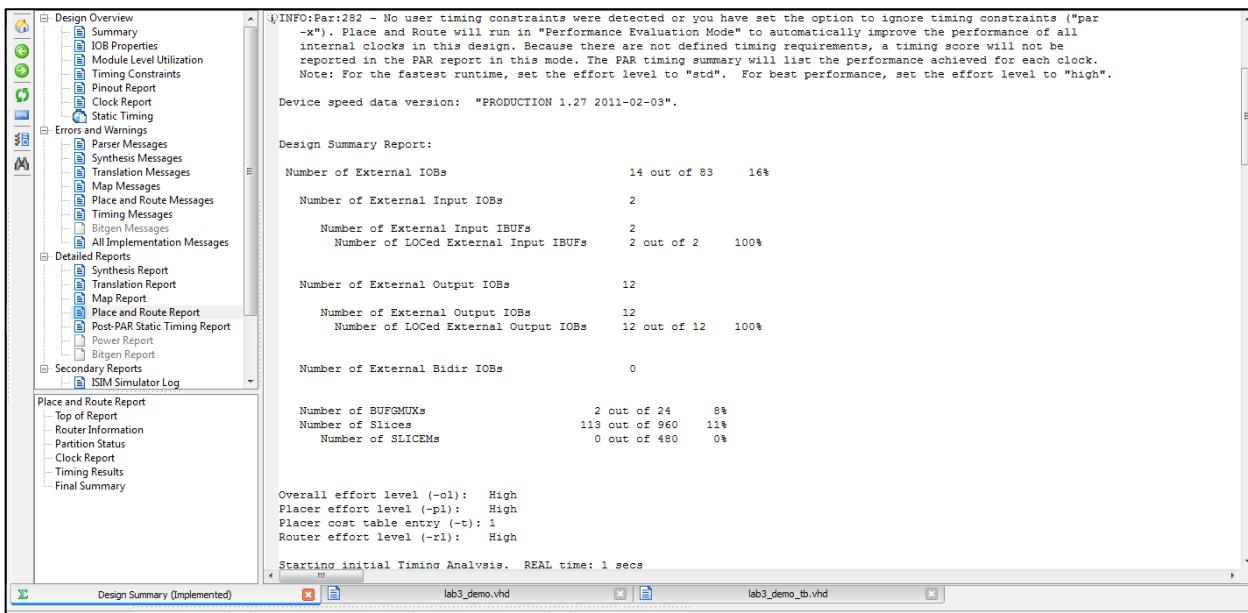
Double click on the Simulate Post-Place & Route Model option. The ISim simulator will be launched and waveforms for post-place & route simulation will appear.



3.8 Implementation Reports

Review the Implementation Reports shown on the following pages. They are available under **Design Summary**.





Note 1: The next section talks about the use of user constraint file (.ucf). All the configurations related to clock and I/O pin assignment are done in the *.ucf file. If you want to do any of the following tasks (synthesis, post-synthesis simulation, implementation, post-translate simulation, post-map simulation or post place and route simulation), and you are not trying to upload the bit stream to FPGA then you can skip the next sections (4-7).

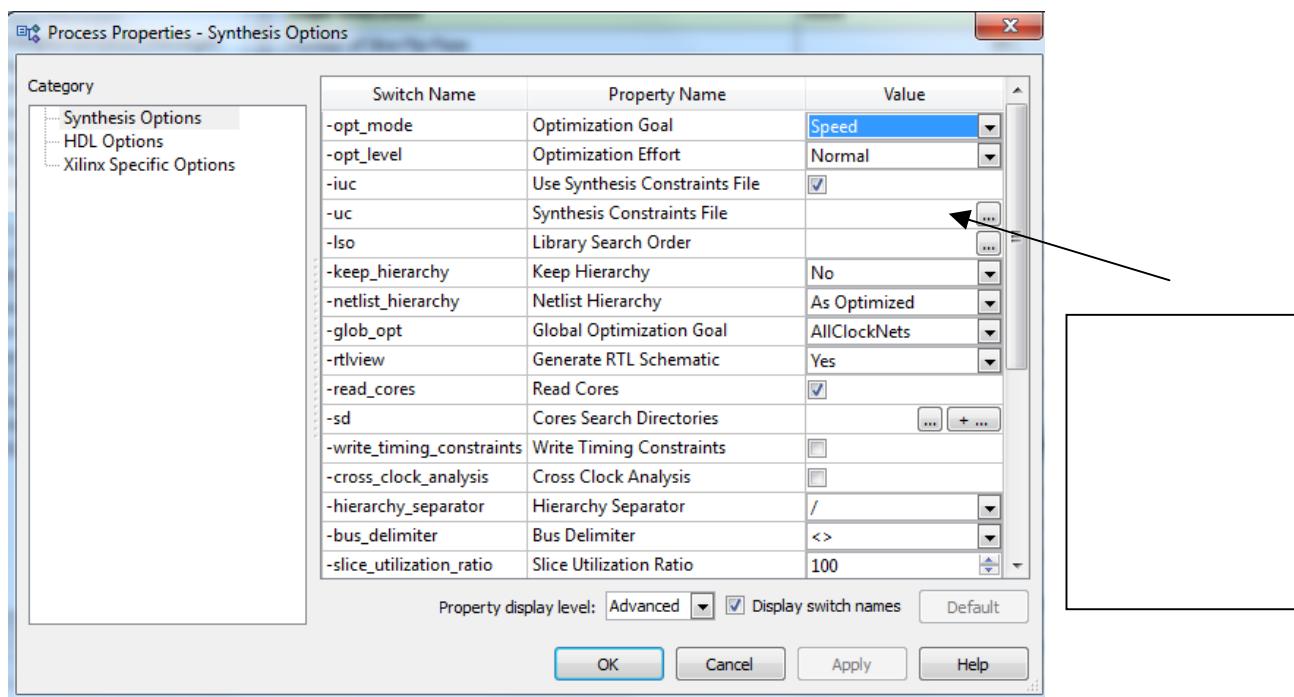
Note 2: Use sections (4-5) and skip sections (6-7) if you want to do Timing analysis. In that particular case, you will choose auto-generated ucf file explained in section 4.3.1.

4. Specifying Frequency/ Time period using User Constraints: Click on radio button beside **Implementation** to switch to implementation mode again.

4.1 Requesting Synthesis clock frequency: In order to apply timing constraints to the design during the synthesis phase, we specify constraints in .xcf file. Create a new file in any text editor and save it with the extension (.xcf) with the clock period settings as per the design requirements, e.g.,

NET "clk" PERIOD = 3.82 ns;

To add xcf file to your design, right click **Synthesize-XST** under the processes menu and click **Process Properties**. Choose **Synthesis Options** under **Category**, and point to the path of your xcf file under **Switch Name = -uc**.



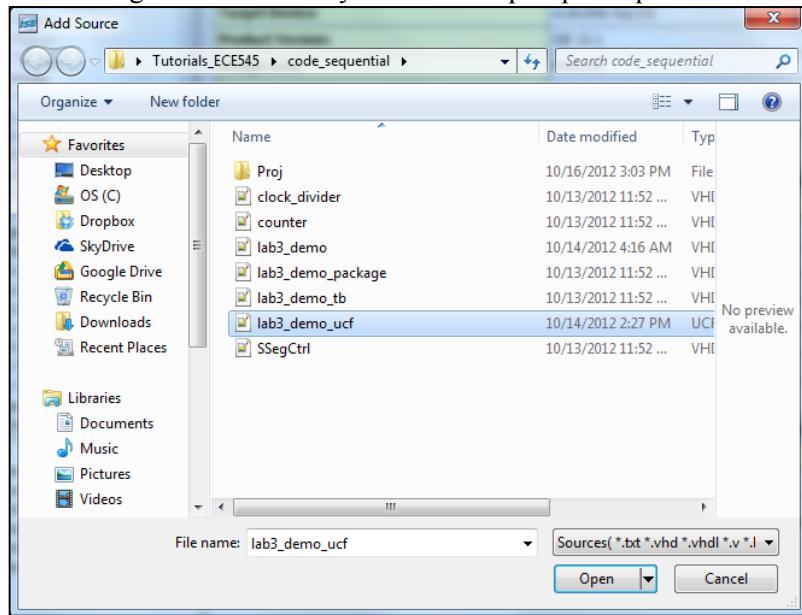
4.2 Requesting Implementation clock frequency: In order to apply timing constraints on the design during the implementation phase, we specify constraints in .ucf file. Create a new file in any text editor and save it with the extension (.ucf) with the clock period settings as per the design requirements, e.g.,

NET "clk" PERIOD = 3.82 ns;

Go to Project and click on Add source. Browse and select the .ucf file by giving path to the specific ucf file. Click OK to add the ucf file to the project.

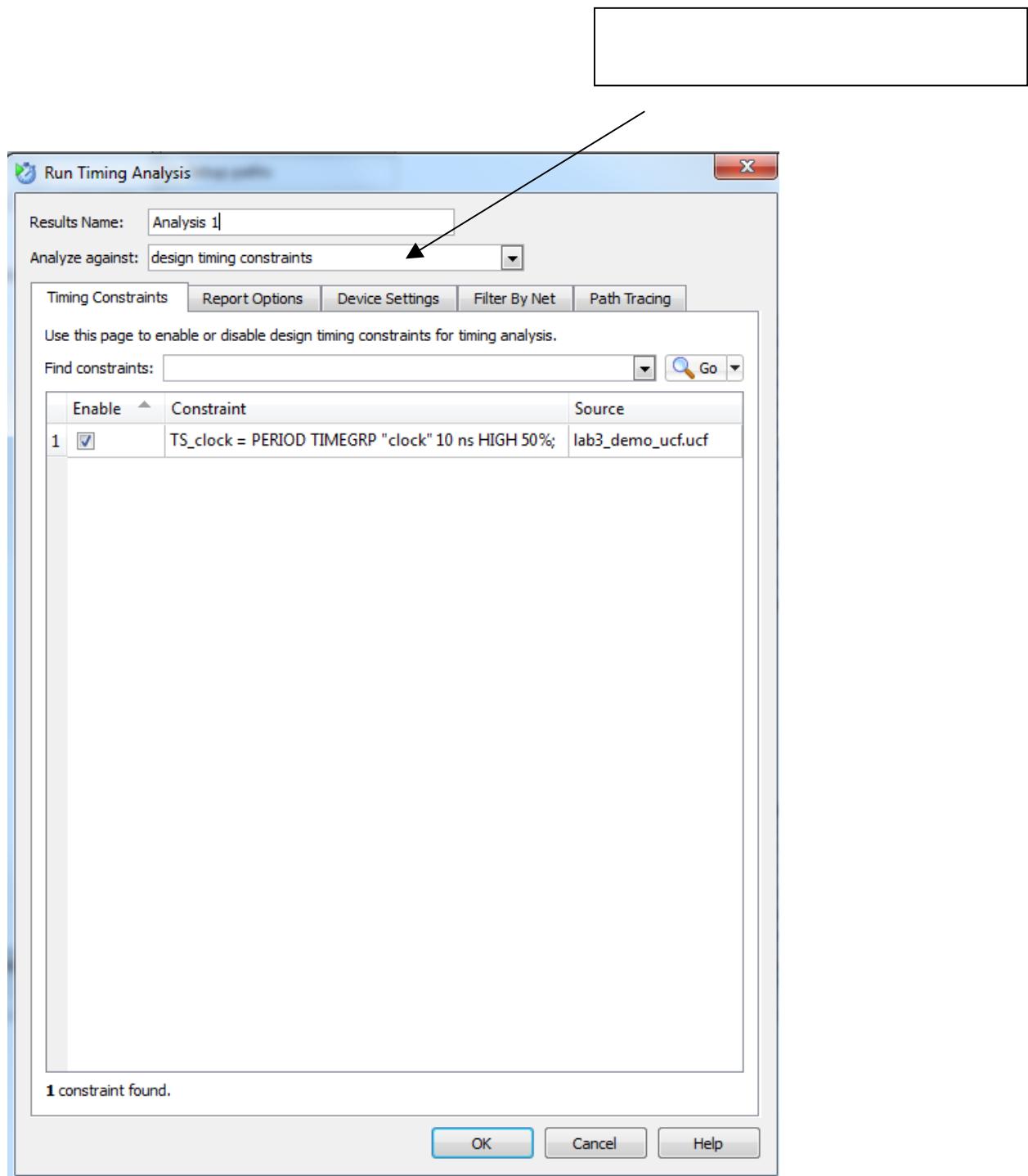
Note: The user constraints file (UCF) and Xilinx constraints file (XCF) are the most commonly used methods of entering constraints. UCF files are used during the implementation process. These files can be used to enter

timing, placement, and pinout constraints. XCF files are used when Xilinx XST is used as your synthesis tool. XCF files are used to write timing constraints for synthesis to help improve performance.



4.3 Obtaining textual representation of critical path using Timing analyzer (after place & route): Go to Tools menu and then click on **Timing Analyzer** and select **Post Place and Route** option. It will automatically include one more tab **Timing** in the menu. Click on **Timing** menu and then select **Run Analysis**.

4.3.1 Choosing between default (auto-generated) or user-defined constraints: The second setting in the Run Timing Analysis is Analyze again where you can specify through a drop down menu whether you want to specify any constraint on the design or you want to use default (auto-generated) constraints of the tool. As we have already specified the timing constraint in the ucf file for the sample design here, tool will show the constraints that are applied in the ucf file (lab_demo_<board_name>.ucf). Click OK to see the text report about the timing results after place route. If you have already specified the timing information in the ucf file, choose the default (auto-generated) timing constraints from the tool.



```

Report Navigation
└─ Timing report description
└─ Timing summary
└─ Informational messages
   └─ INFO:Timing:2752 ...s) of the report.
   └─ INFO:Timing:3339...evice datasheet.
   └─ INFO:Timing:3390...nty calculation.
   └─ INFO:Timing:3389...and Phase Error.
└─ Timing constraints
   └─ TS_clock = PERIOD..." 10 ns HIGH 50%
      └─ Setup paths
      └─ Hold paths
      └─ Component switching limits
└─ Constraint compliance
└─ Data sheet report
└─ Timing Analyzer settings

Info Message
1 INFO:Timing:3389 - This architecture does not support 'Discrete Jitter' and 'Phase Error' calculations, these terms will be zero in the Clock Uncertainty analysis.

-----
13.1
Copyright (c) 1995-2011 Xilinx, Inc. All rights reserved.

Design file: C:\Users\Umar\Dropbox\ECE545_F12\Tutorials_ECE545\code_sequential\Proj\lab3_demo.ncd
Physical constraint file: C:\Users\Umar\Dropbox\ECE545_F12\Tutorials_ECE545\code_sequential\Proj\lab3_demo.pcf
Device,package,speed: xc3s100e,cp132,-5 (PRODUCTION 1.27 2011-02-03)
Report level: verbose report

Environment Variable Effect
-----
NONE No environment variables were set
-----
```

Scroll down till the end of the file (Analysis 1.twx) to see the timing summary that includes minimum period and maximum frequency of the design after place and route.

```

Timing summary:
-----
Timing errors: 0 Score: 0 (Setup/Max: 0, Hold: 0)
Constraints cover 1737 paths, 0 nets, and 207 connections
Design statistics:
  Minimum period: 5.751ns{1} (Maximum frequency: 173.883MHz)
```

As we did not change the **Report options**, the tool will automatically show the timing information of the paths that have the biggest slack (worst critical path). Search for minimum period in the .twx file from all three paths and choose the one that matches the maximum period shown in the timing summary. It will also have the corresponding critical path information of your design after place and route along with it in the text format.

```

=====
Timing constraint: TS_clock = PERIOD TIMEGRP "clock" 10 ns HIGH 50%;
1737 paths analyzed, 120 endpoints analyzed, 0 failing endpoints
0 timing errors detected. (0 setup errors, 0 hold errors, 0 component switching limit errors)
Minimum period is 5.751ns.
-----

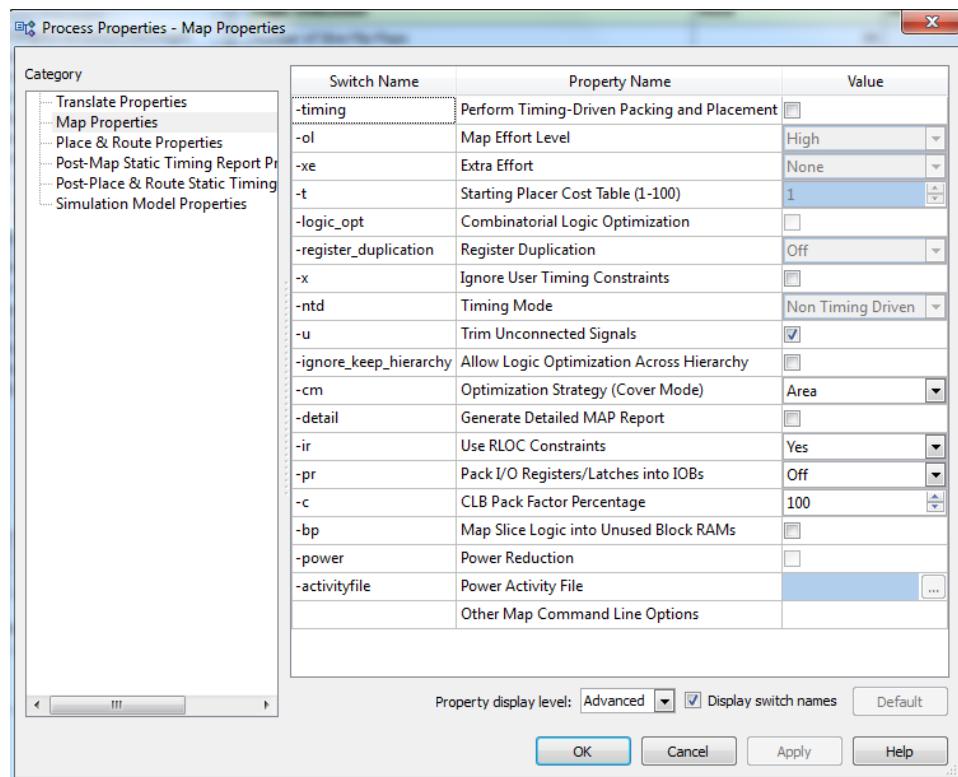
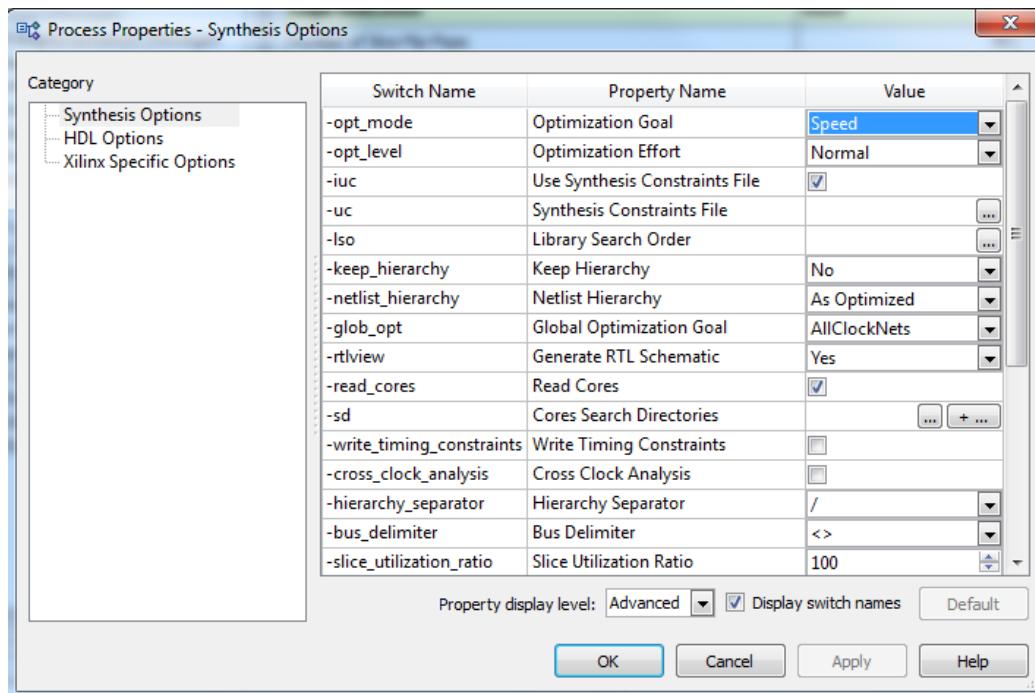
Paths for end point clock1k_gen/counter_3 (SLICE_X24Y22.F2), 32 paths
-----
Slack (setup path): 4.249ns (requirement - (data path - clock path skew + uncertainty))
Source: clock1k\_gen/counter\_1 (FF)
Destination: clock1k\_gen/counter\_3 (FF)
Requirement: 10.000ns
Data Path Delay: 5.751ns (Levels of Logic = 2)
Clock Path Skew: 0.000ns
Source Clock: clock\_BUFGP rising at 0.000ns
Destination Clock: clock\_BUFGP rising at 10.000ns
Clock Uncertainty: 0.000ns

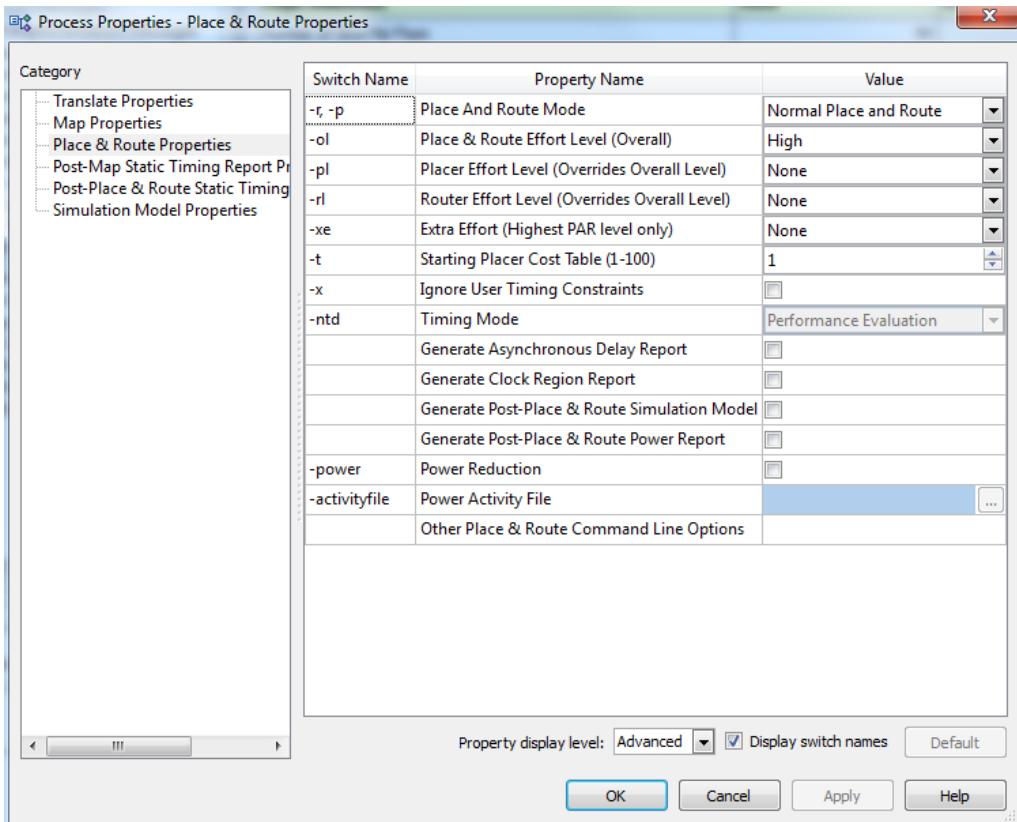
Maximum Data Path: clock1k_gen/counter_1 to clock1k_gen/counter_3

| Location          | Delay type             | Delay(ns)                              | Physical Resource                                                                                                                                                                                                                          |
|-------------------|------------------------|----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                   |                        |                                        | Logical Resource(s)                                                                                                                                                                                                                        |
| SLICE_X24Y23.XQ   | <a href="#">Tcko</a>   | 0.515                                  | <a href="#">clock1k_gen/counter&lt;1&gt;</a><br><a href="#">clock1k_gen/counter_1</a>                                                                                                                                                      |
| SLICE_X23Y32.F3   | net (fanout=2)         | 1.388                                  | <a href="#">clock1k_gen/counter&lt;1&gt;</a>                                                                                                                                                                                               |
| SLICE_X23Y32.COUT | <a href="#">Topcyf</a> | 1.011                                  | <a href="#">clock1k_gen/clock_cmp_eq0000</a><br><a href="#">clock1k_gen/clock_cmp_eq0000_wq_lut&lt;6&gt;</a><br><a href="#">clock1k_gen/clock_cmp_eq0000_wq_cv&lt;6&gt;</a><br><a href="#">clock1k_gen/clock_cmp_eq0000_wq_cv&lt;7&gt;</a> |
| SLICE_X24Y22.F2   | net (fanout=33)        | 2.061                                  | <a href="#">clock1k_gen/clock_cmp_eq0000</a>                                                                                                                                                                                               |
| SLICE_X24Y22.CLK  | <a href="#">Tfck</a>   | 0.776                                  | <a href="#">clock1k_gen/counter&lt;3&gt;</a><br><a href="#">clock1k_gen/Mcount_counter_eqn_32</a><br><a href="#">clock1k_gen/counter_3</a>                                                                                                 |
| Total             |                        | 5.751ns (2.302ns logic, 3.449ns route) | (40.0% logic, 60.0% route)                                                                                                                                                                                                                 |


```

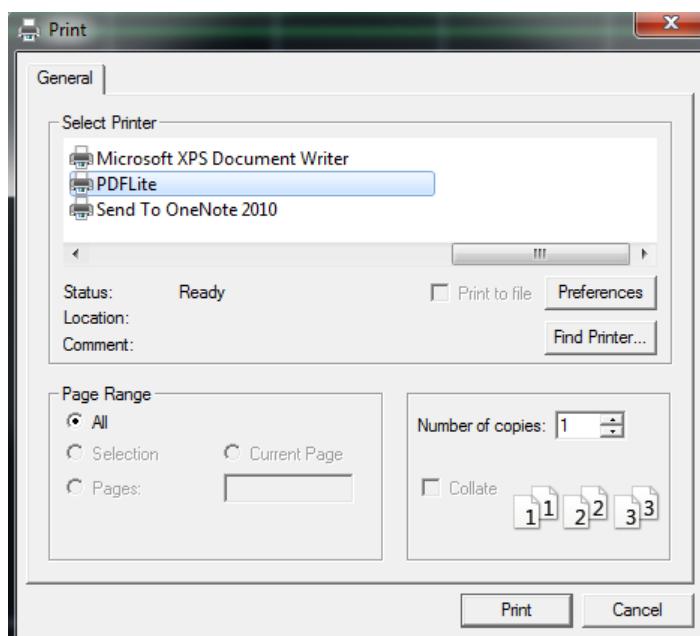
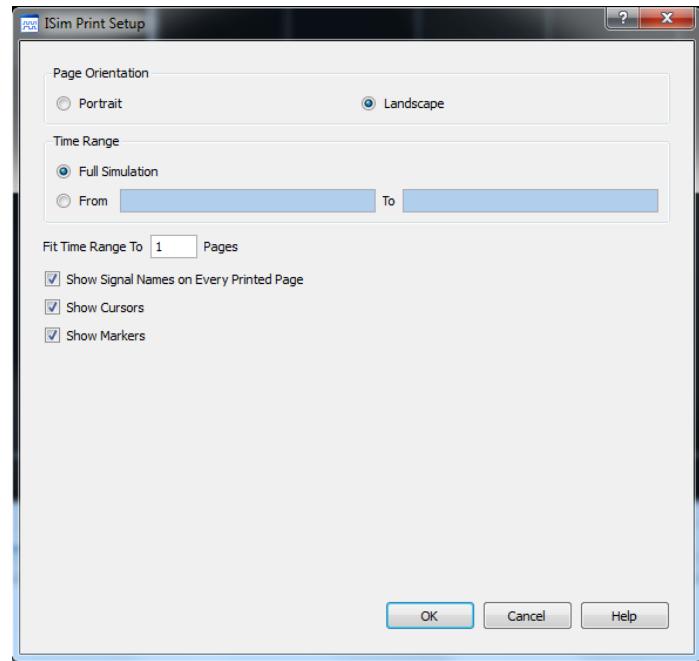
5. **Optimization strategies:** Right click on the synthesis and implementation options in processes menu and then choose process properties to select a complete list of optimization strategies.



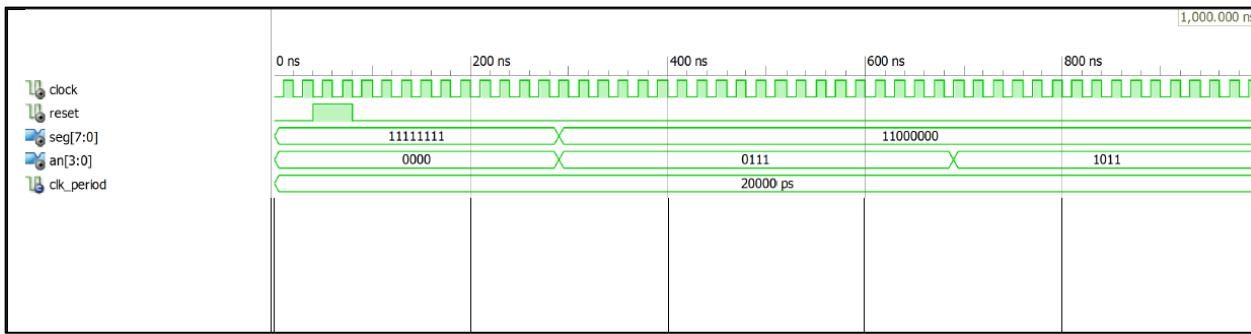


Print the output waveform in PDF format: In order to view all the important details of the behavioral, post-synthesis, post-translate, post-map and post-place & route simulations in waveform in PDF format, use 3rd part tools (PDF creator, PDF Lite etc).

Go to file menu and click on the print menu. A pop up will appear that will give you the option to select time range (full range or for a specific range) and an option to fit time range into 1 or multiple pages. Click OK and chose PDF Lite or any other tool to do PDF conversion. After completing the print setup, a new window pops up and ask you to save the file in PDF format.

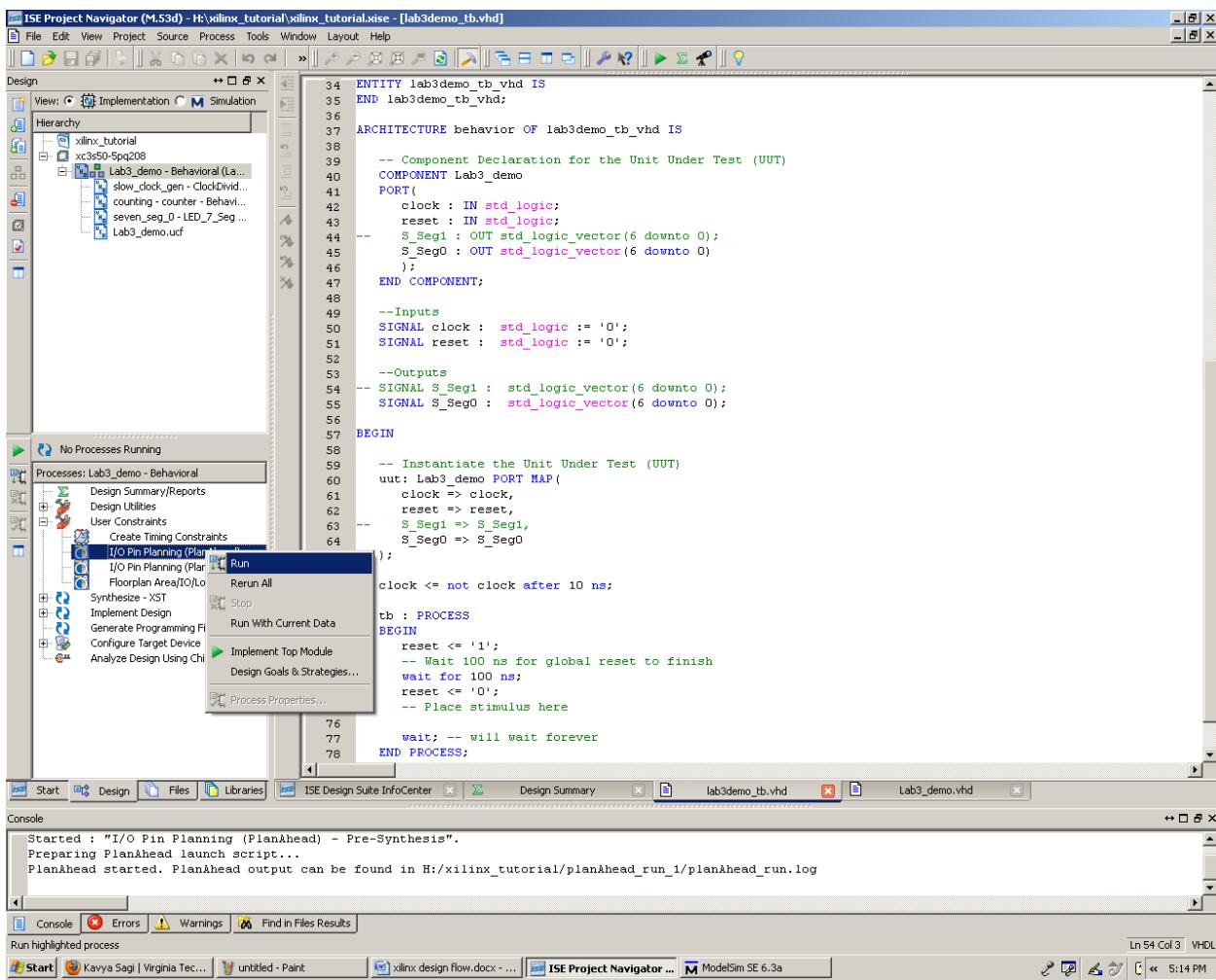


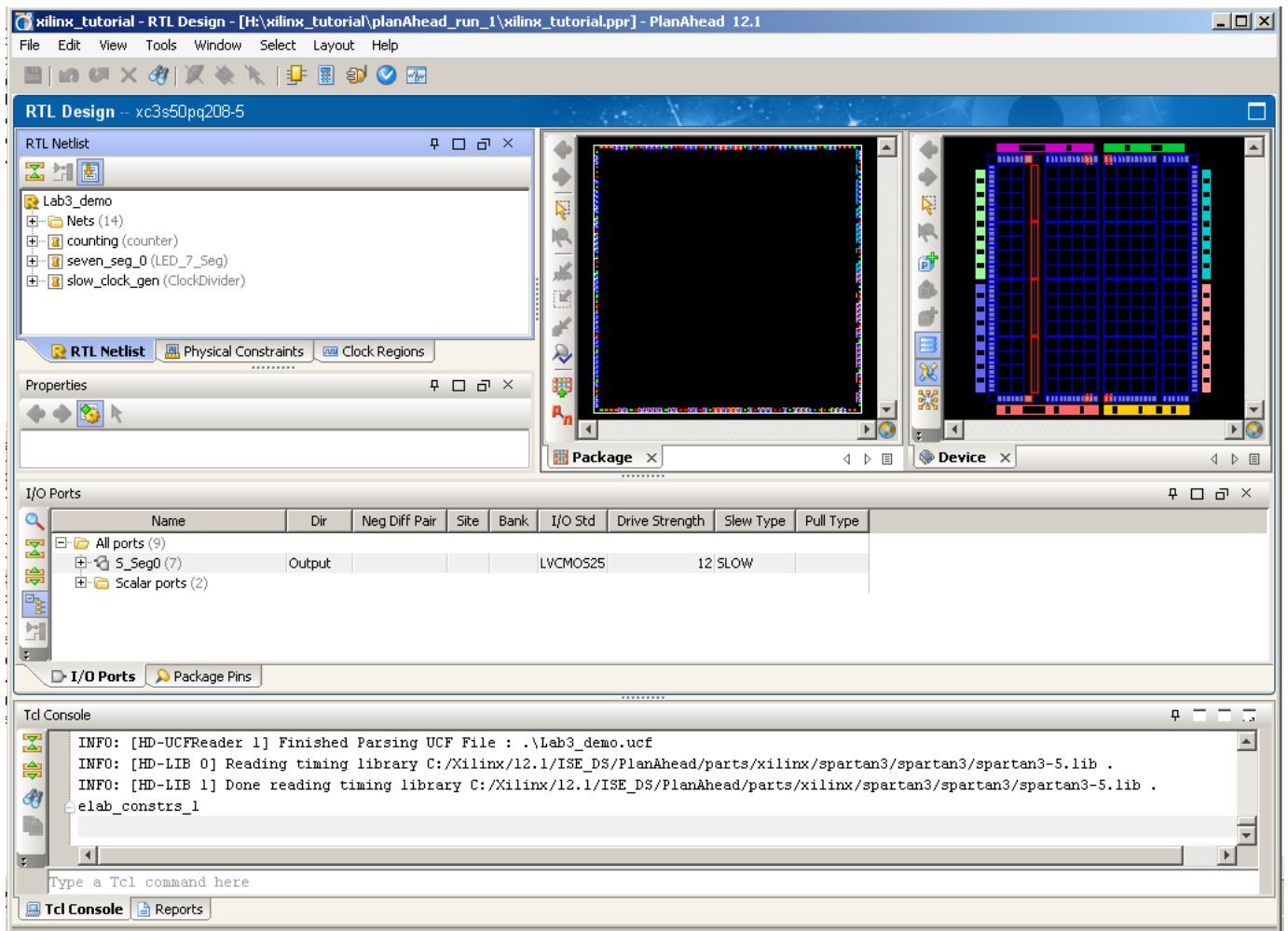
Here is the output waveform of post-translate simulation in PDF format.

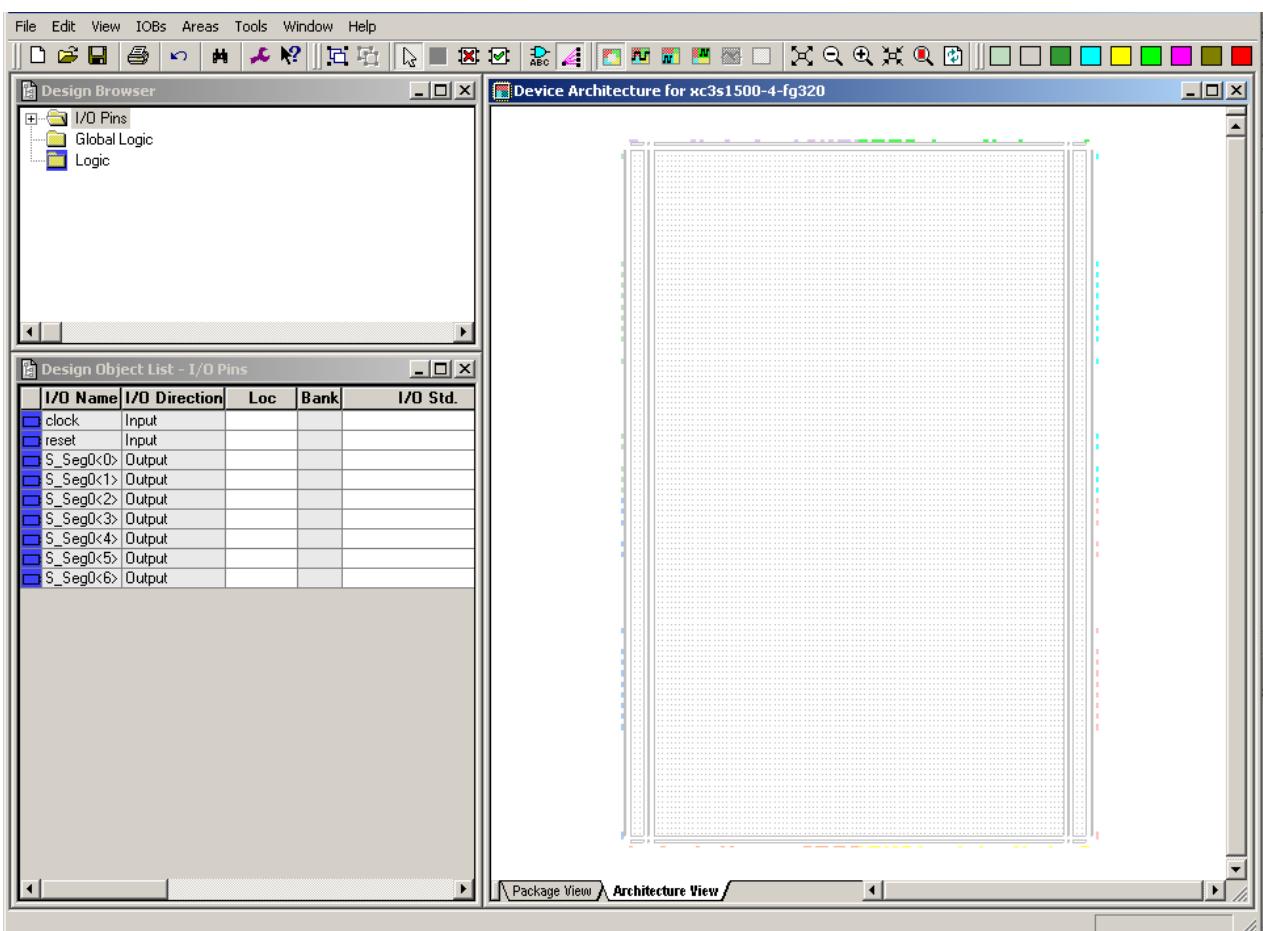


Note: Remaining two sections apply only to users that target specific FPGA boards (e.g., ECE-448 students)

- 6. Pin Assignment:** Go to menu Sources for and change this option to **Synthesis/Implementation**. Click on **Assign Package Pins**. The User Constraint File (UCF) will be created if you decide to assign design port names to the physical pins. Columns I/O Names and I/O Descriptions represent ports from your design. The Loc column should be used to input the location of the corresponding pin in the FPGA device.



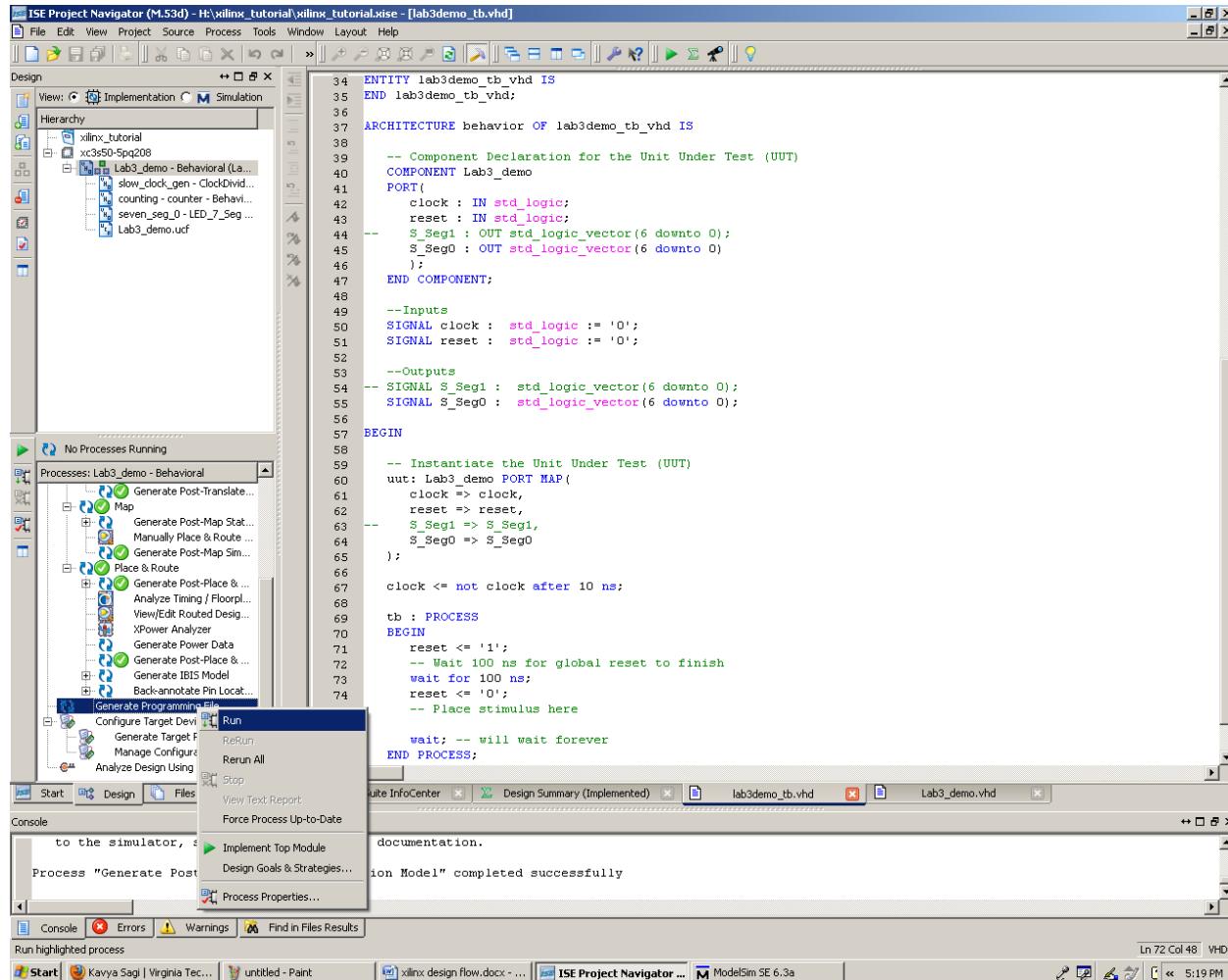




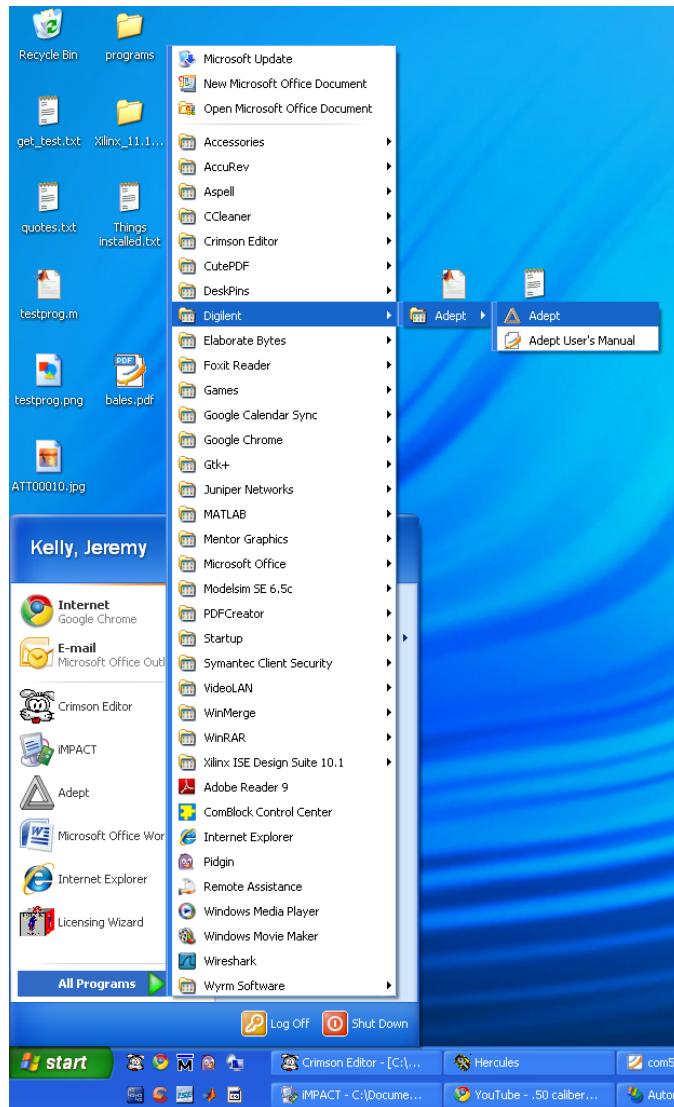
We can specify Pin Assignments by going to **User Constraint** menu. We can assign design port names to the physical pins of a chosen device by option **Edit Constraints** (Text). Keyword **NET** is for a port name assignment and keyword **LOC** is for a physical pin assignment.

7. Bit Stream Generation

Go to menu **Sources for** and change this option to **Synthesis/Implementation**. Choose **Generate Programming File**, do right-click and pop-up menu should appear. Choose **Run** to start bit generation process.



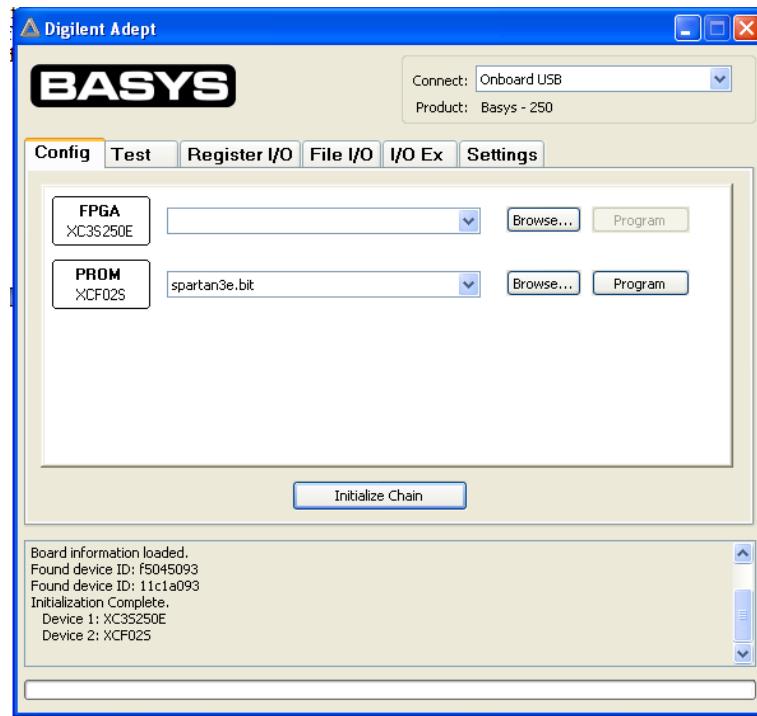
8. Uploading Bitstream to FPGA Board



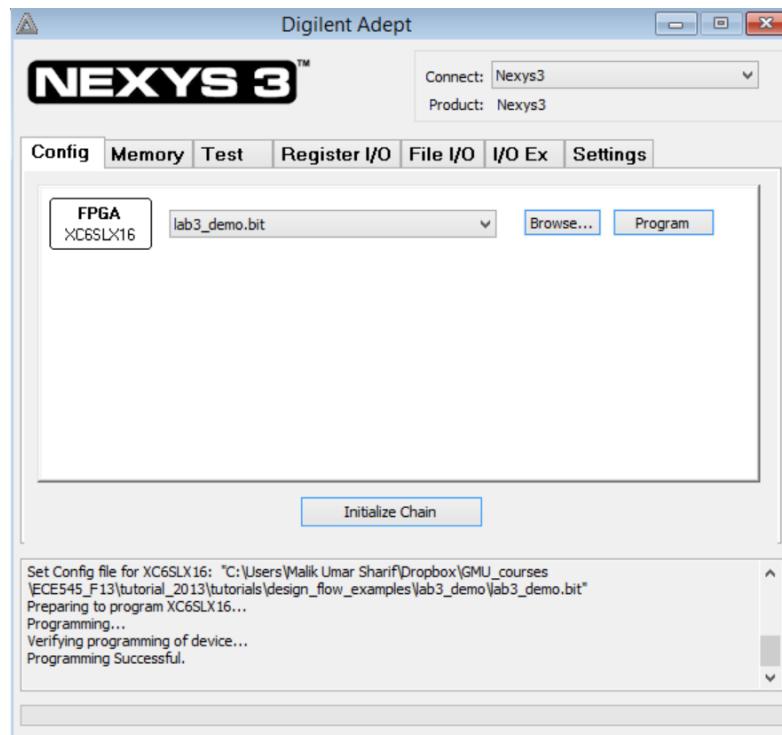
Before uploading Bit file, make sure that you change your **constant values** in all your files to proper values, and re-synthesize/re-implement all the files. In particular, in our example, please change the value of the constant slow_clock_period in the Lab3Demo_package.vhd.

Select the Adept program as shown in the picture above. When the program is opened, a device will be shown if it is connected and recognized. Select the bit file by clicking **Browse** and finding the appropriate file. Click **Program** to program the file device.

a. Uploading bit stream on Basys2 board:



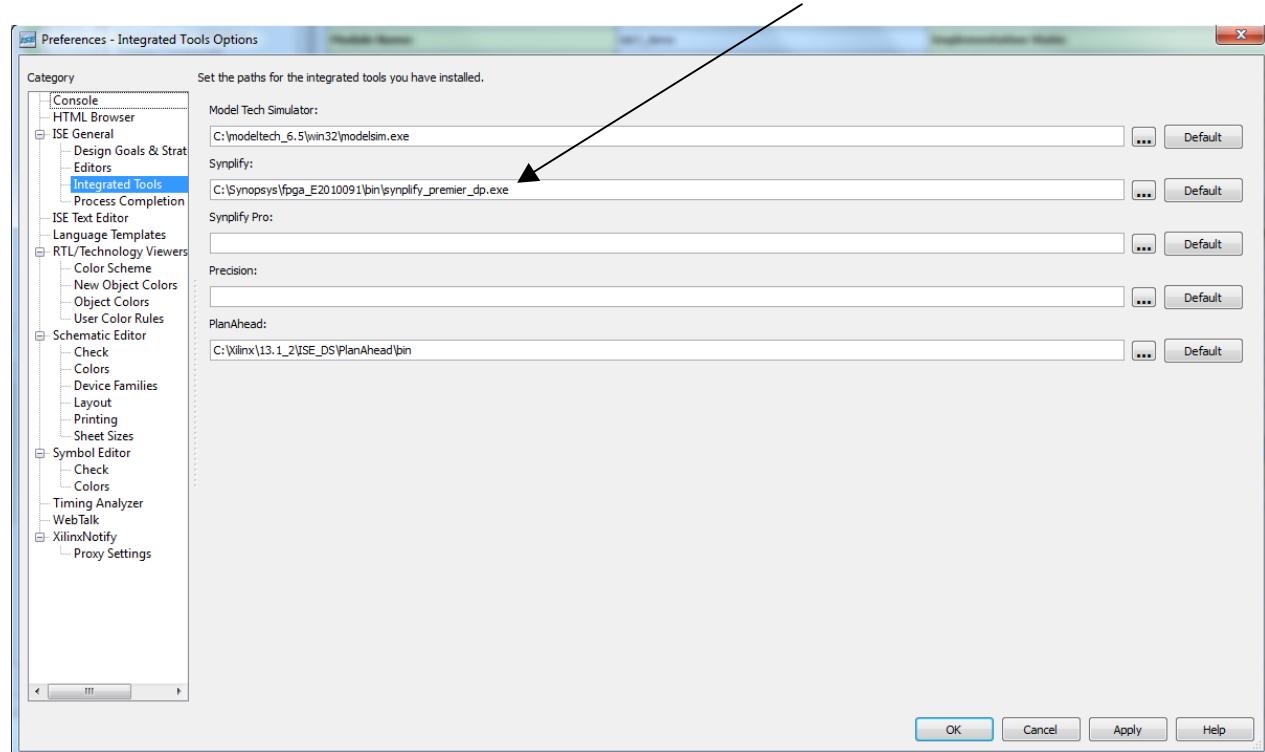
b. Uploading bit stream on Nexys3 board:



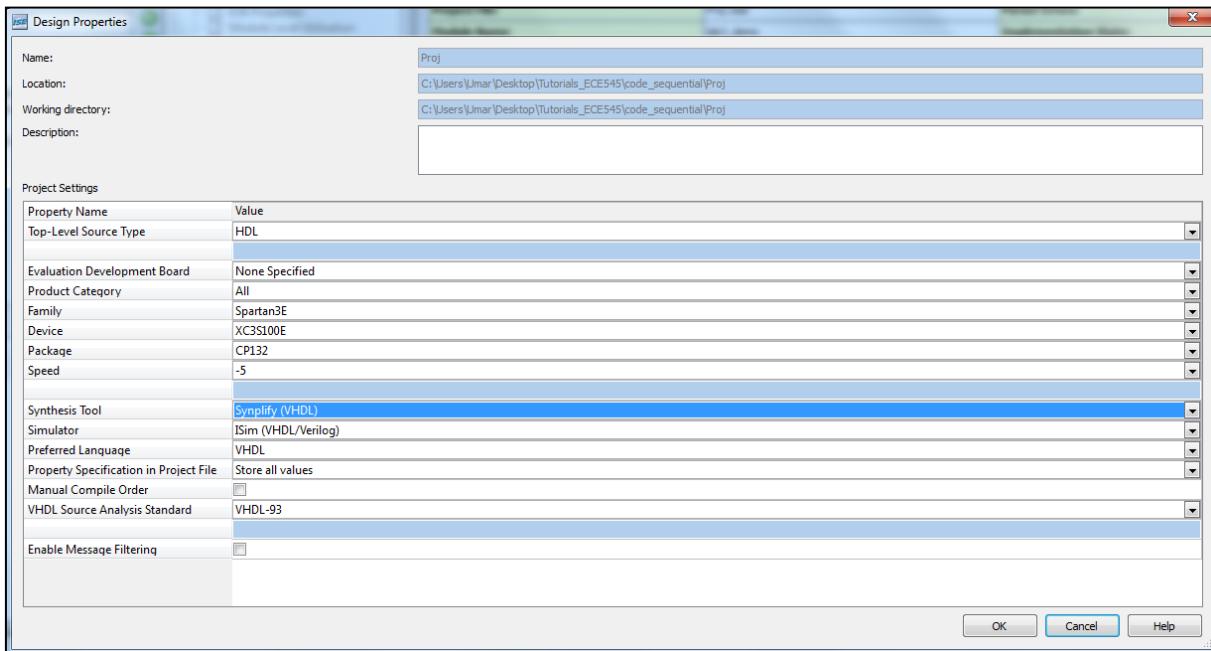
Appendix

Synthesis with Synplify Premier DP

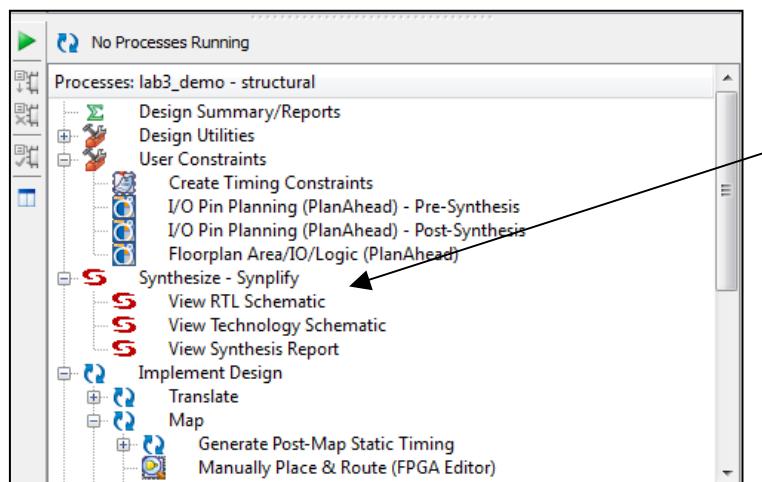
Synplify Premier DP is available only in the ECE Labs. In order to use it, please go to edit menu and click on Preferences. Once a new Preferences window pops up, select Integrated Tools under the Category ISE General. Set the path for Synplify Premier DP under the synthesis tool "Synplify". Click Apply to save the settings, and then click OK.



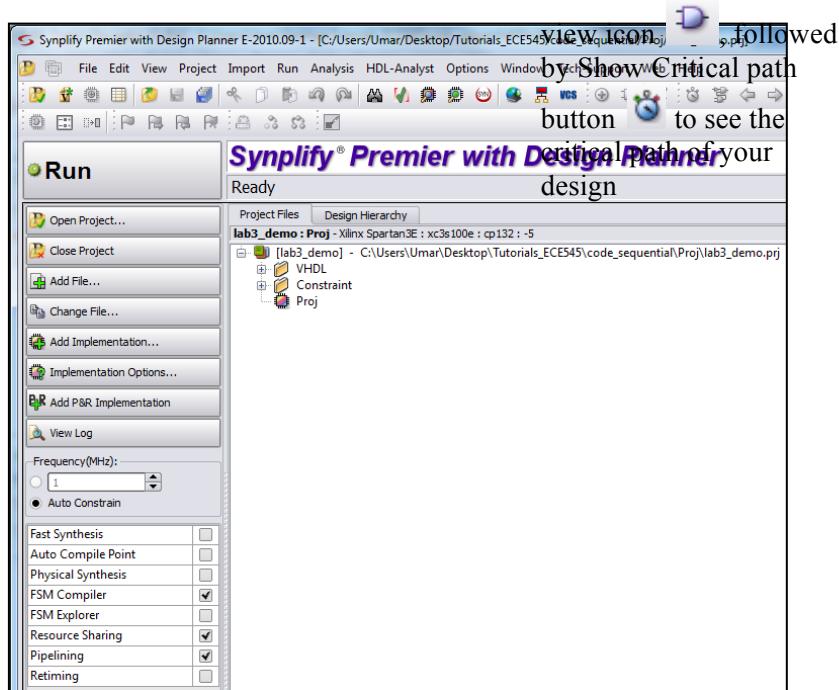
Go to Project menu and click on Design Properties. A new window pops up for Design properties. Go to synthesis tool and click on drop down menu to select **Synplify (VHDL)**. Click OK to save the settings.



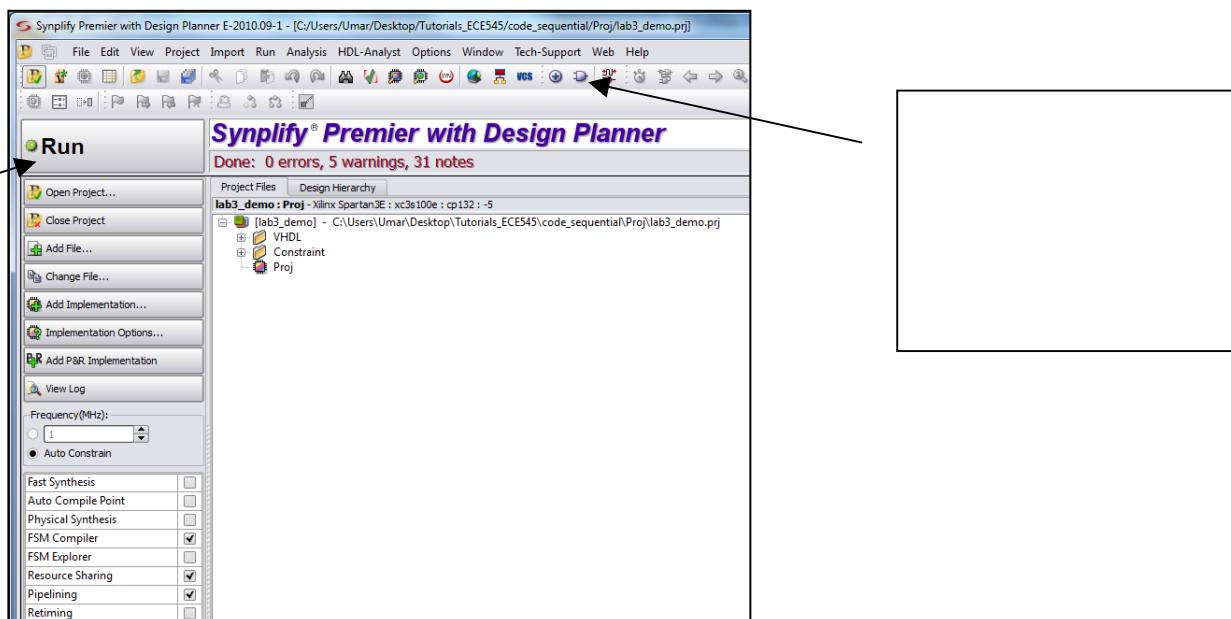
Now you can observe that the synthesis tool changes automatically in the processes menu.

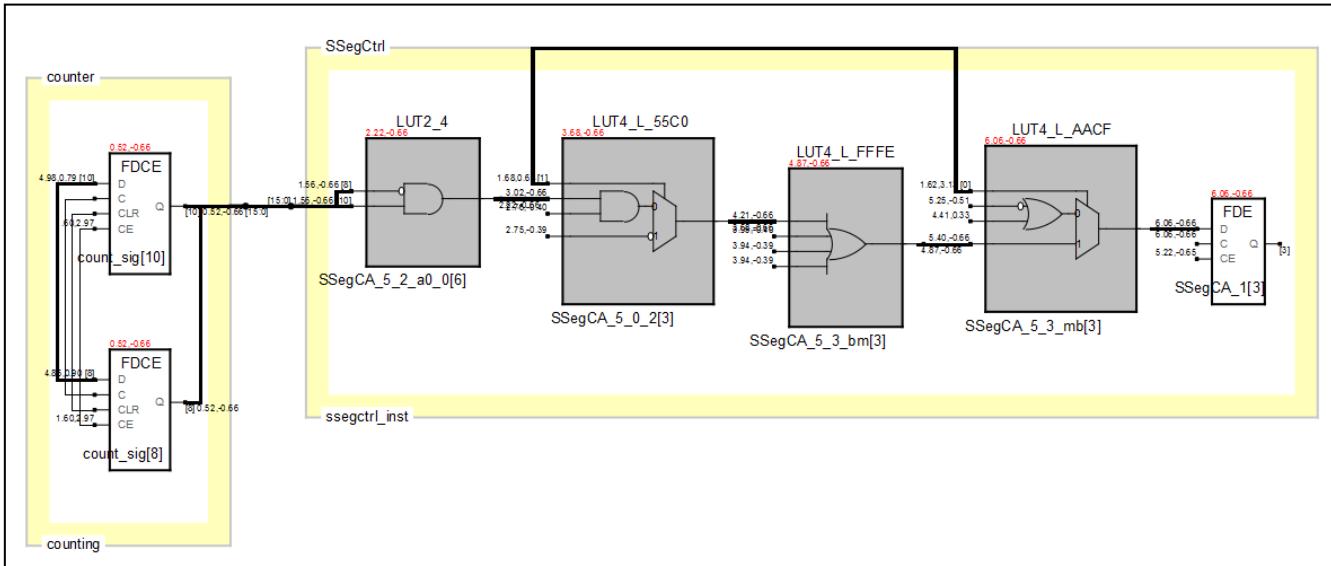


Double click on the **Synthesize - Synplify** to run synthesis using Synplify Premier DP. You can also view RTL and technology view of the schematic. Once you double click on **View RTL Schematic**, it will automatically invoke the tool "Synplify Premier DP".

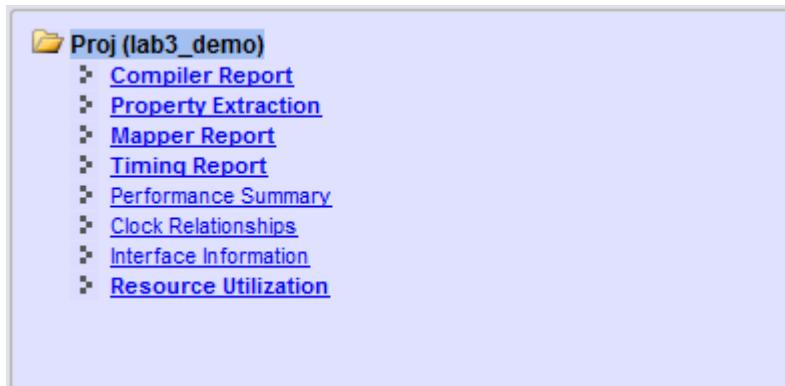


Finding the Critical path of the design: Click on **Run** to start the synthesis inside Synplify Premier DP. Click on Technology view icon followed by Show Critical path button to see the critical path of your design





Go to View menu and click on View Log File. It will open a window and will show you following options:



Select **Compiler Report** and go down to see the performance summary for the maximum frequency and minimum clock period of the design, estimated after synthesis.

Performance Summary							
<hr/>							
Starting Clock	Requested Frequency	Estimated Frequency	Requested Period	Estimated Period	Slack	Clock Type	Clock Group
lab3_demo clock	197.8 MHz	175.0 MHz	5.057	5.713	-0.656	inferred	Autoconstr_clkgroup_1

Now click on **Resource Utilization** to see the summary of device utilization in terms of different resources utilized by your design.

```
-----  
Resource Usage Report for lab3_demo  
  
Mapping to part: xc3s100ecp132-5  
Cell usage:  
FDC          35 uses  
FDCE         48 uses  
FDE          13 uses  
GND          5 uses  
MUXCY        3 uses  
MUXCY_L      88 uses  
MUXF5         2 uses  
VCC          5 uses  
XORCY        78 uses  
LUT1         84 uses  
LUT2         25 uses  
LUT3         16 uses  
LUT4         75 uses  
  
I/O ports: 14  
I/O primitives: 14  
IBUF          1 use  
IBUFG         1 use  
OBUF         12 uses  
  
BUFG          1 use  
  
I/O Register bits:           0  
Register bits not including I/Os: 96 (5%)  
  
Global Clock Buffers: 1 of 24 (4%)  
  
Total load per clock:  
lab3_demo|clock: 51  
  
Mapping Summary:  
Total LUTs: 200 (10%)  
  
Mapper successful!  
Process took 0h:00m:04s realtime, 0h:00m:03s cputime
```