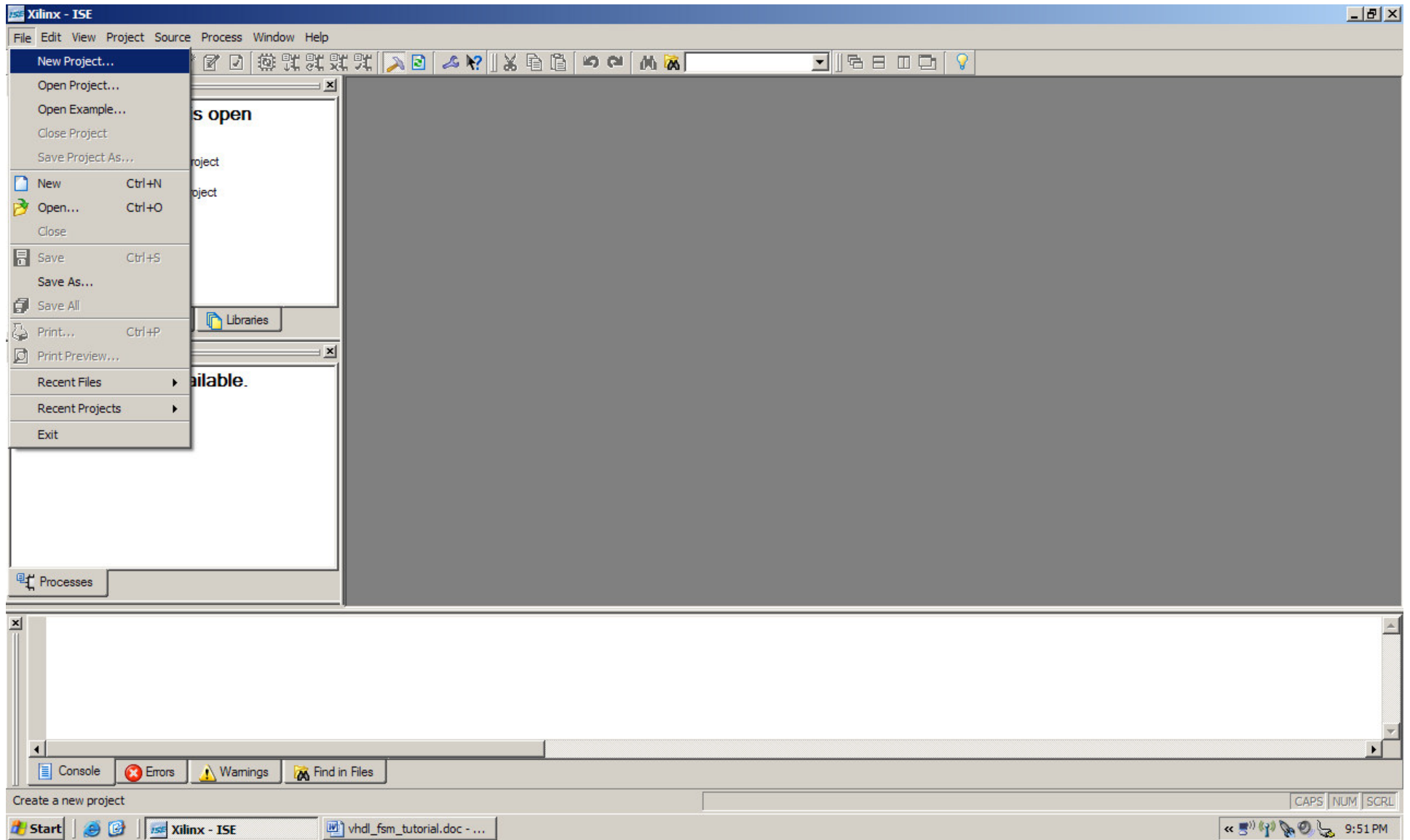
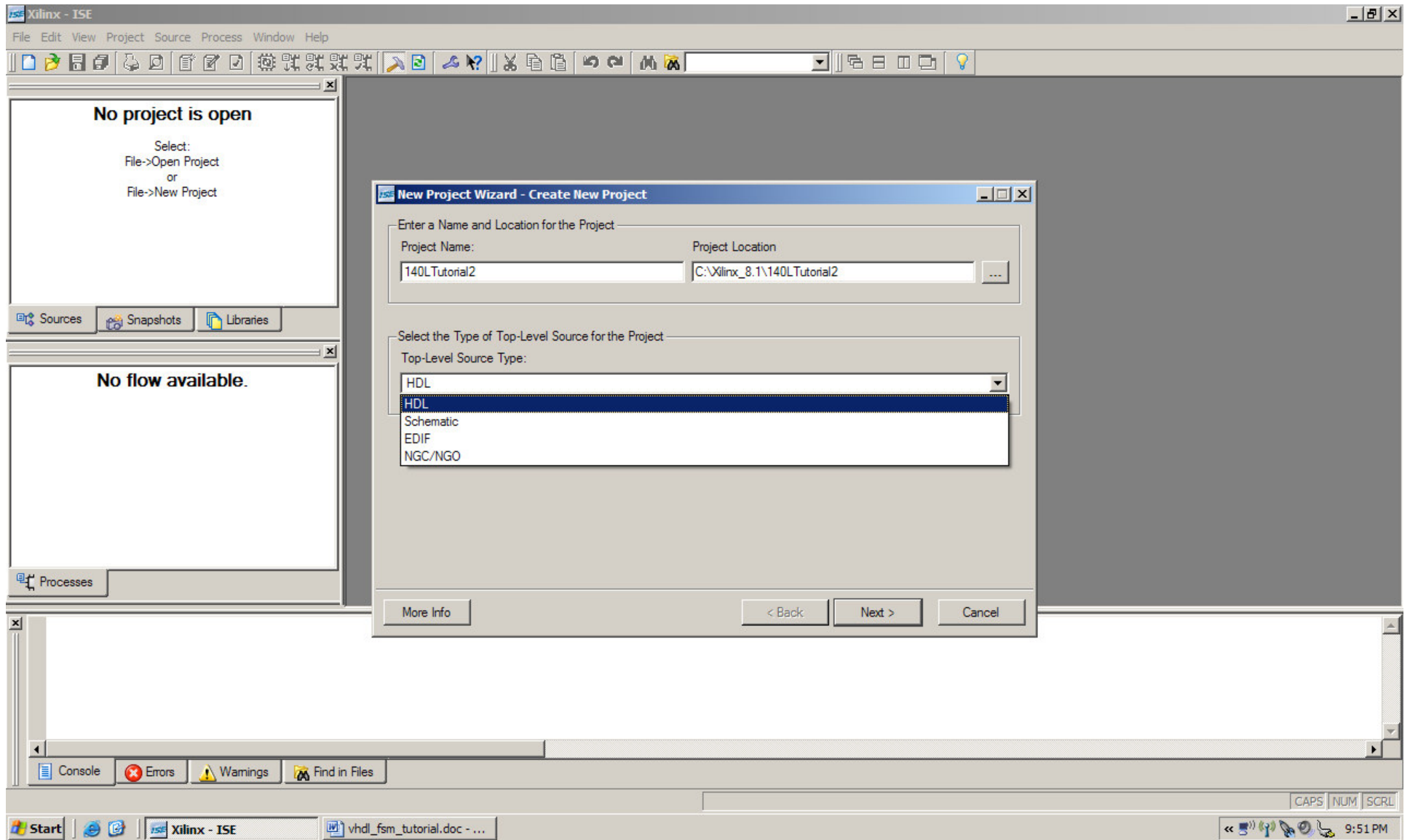
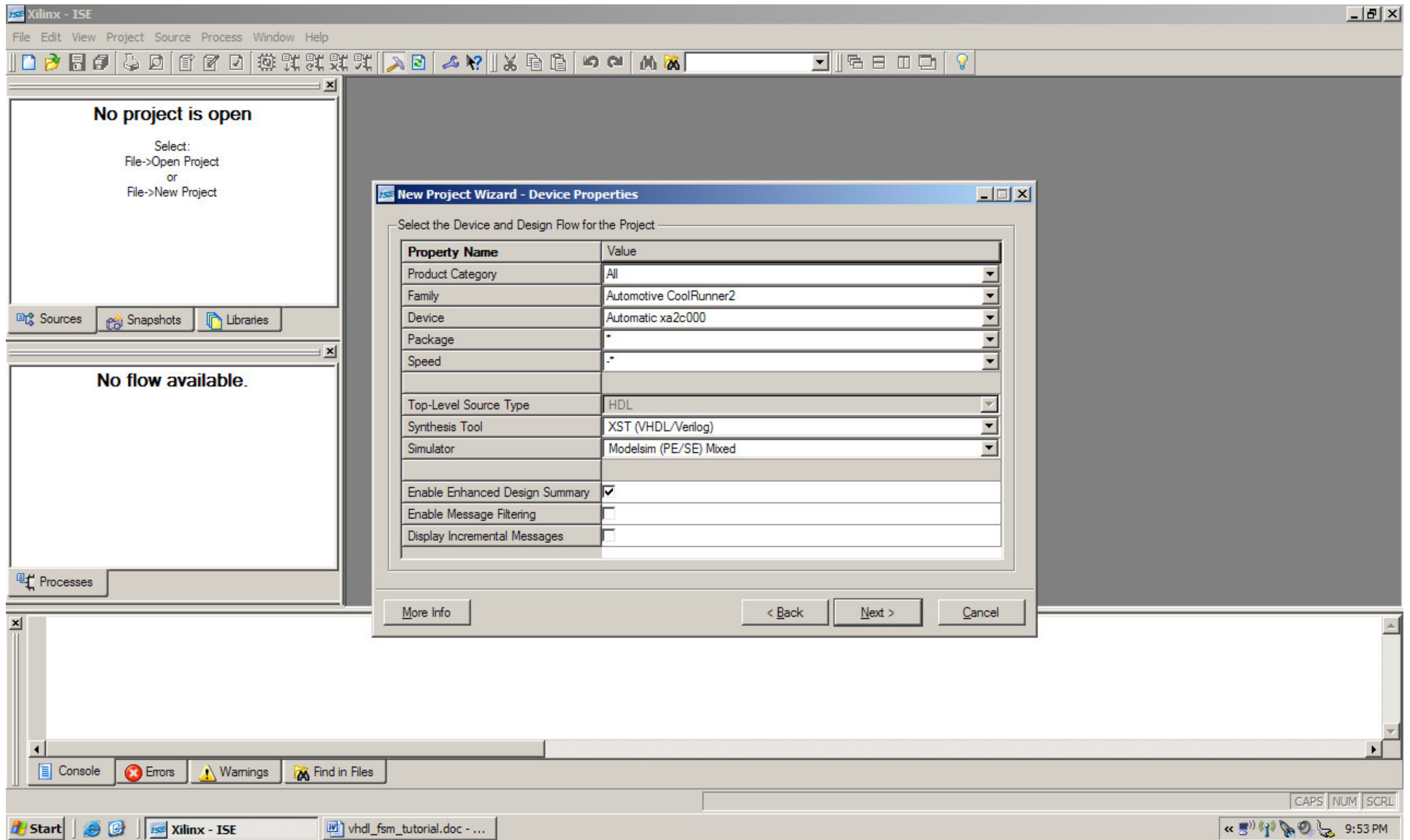




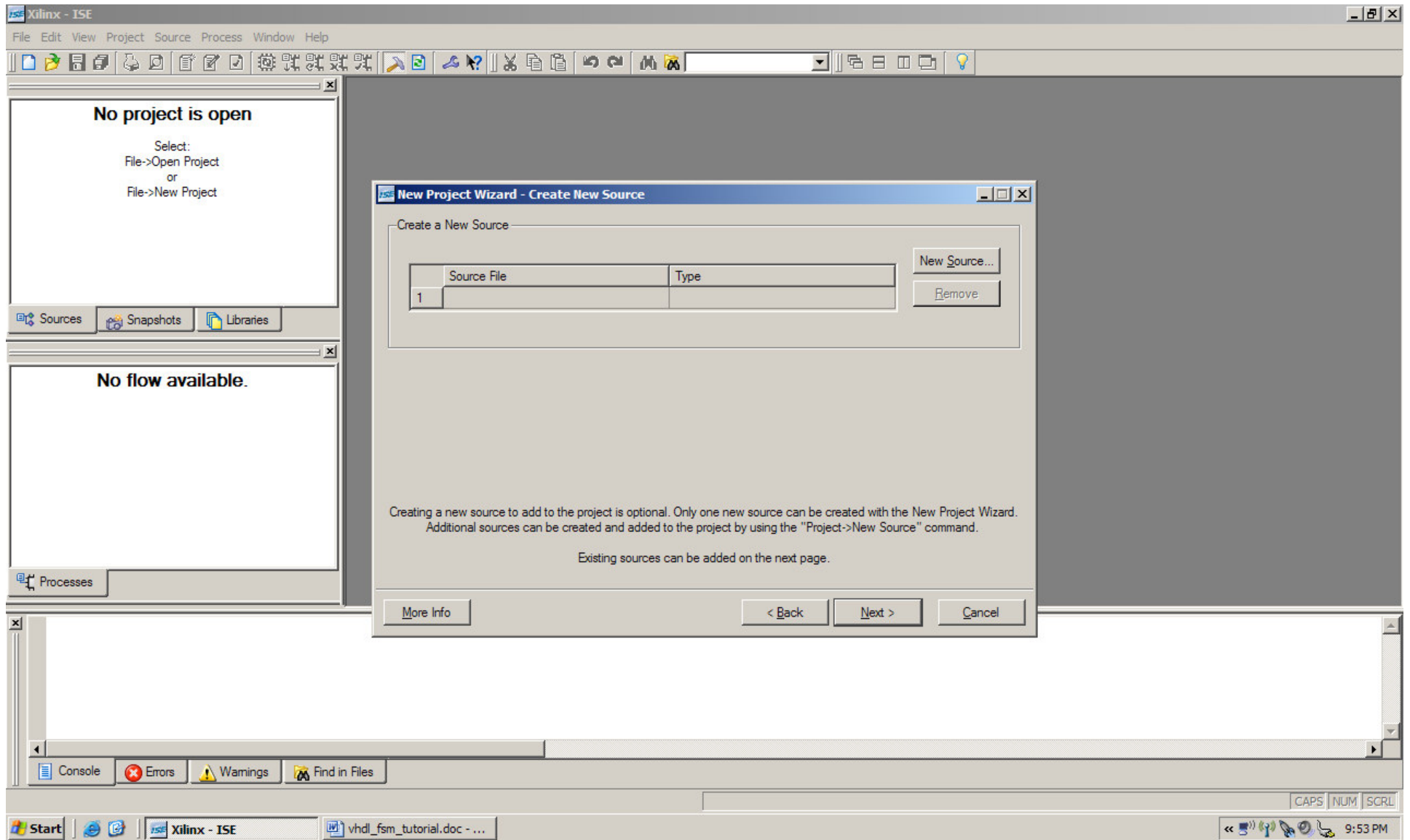
Start a new project called 140LTutorial2. In this tutorial, we will learn how to create a simple finite state machine using VHDL. The finite state machine will detect odd number of ones. There will be 3 inputs (data_in, rst, clk) and one output (odd_ind).



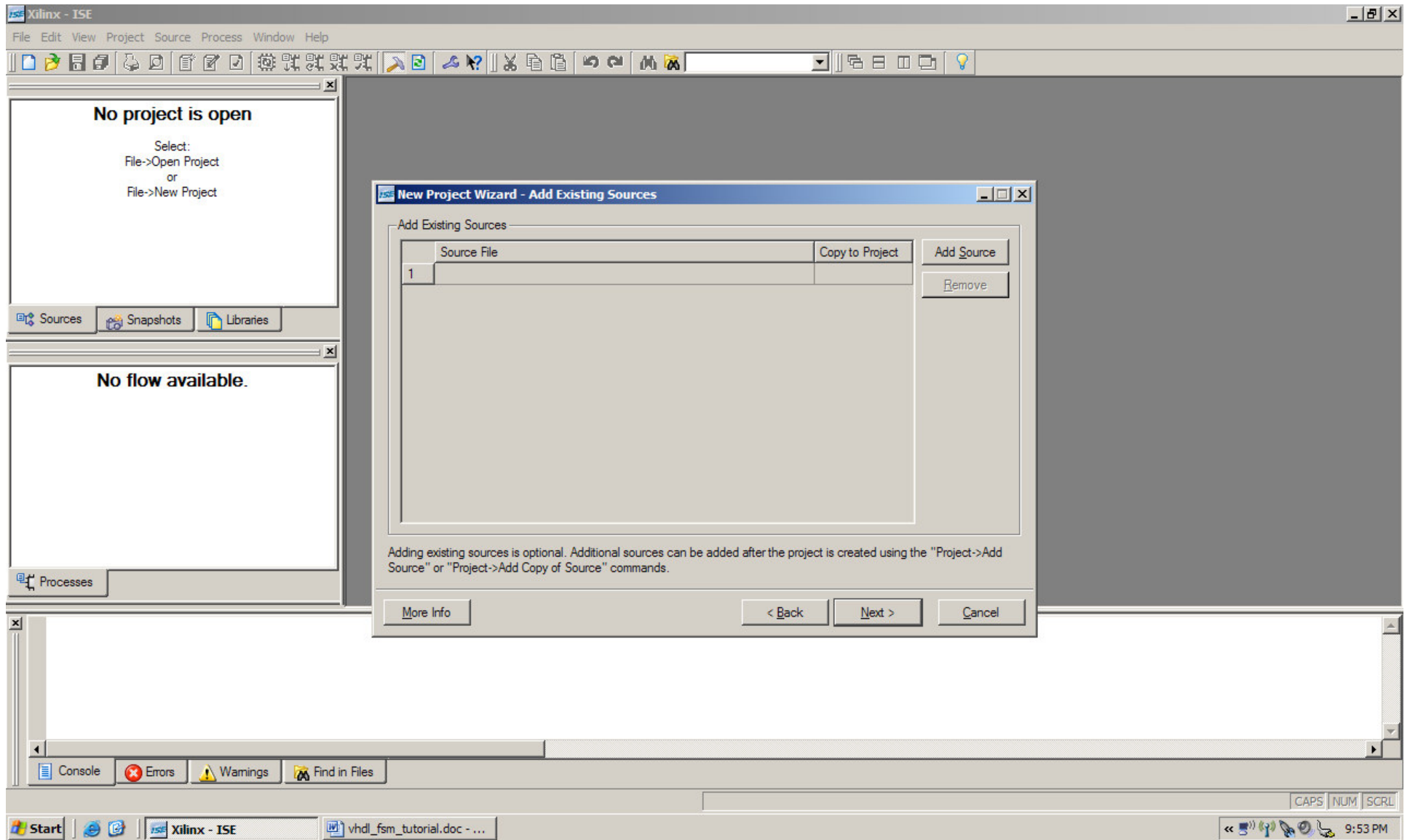
Enter the project name 140LTutorial2. Be sure to select HDL as the Top-Level Source Type. Click Next to continue.



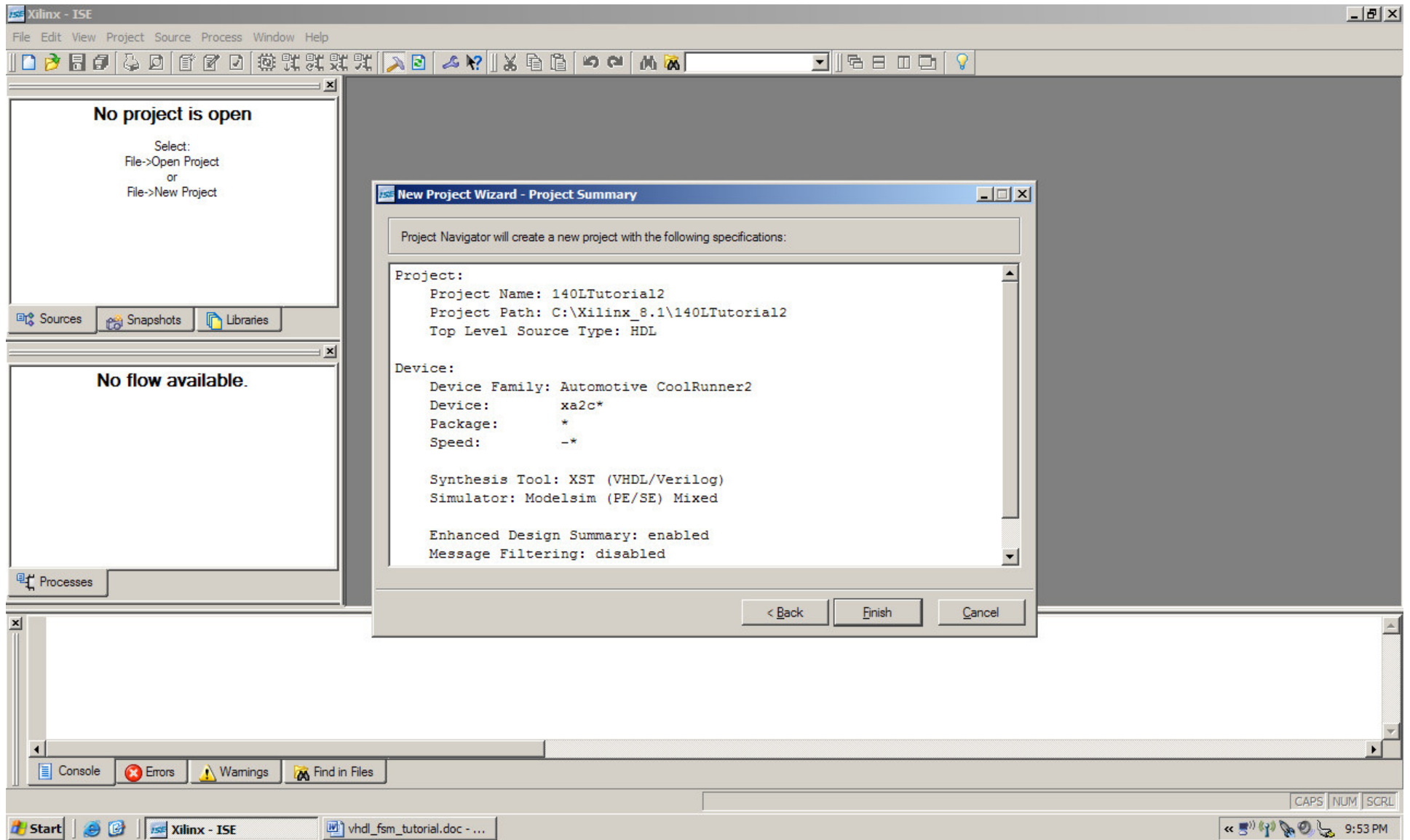
Leave everything as default... Your simulator selection should have been selected as Modelsim (PE/SE) Mixed in the previous labs already. Click Next to continue.



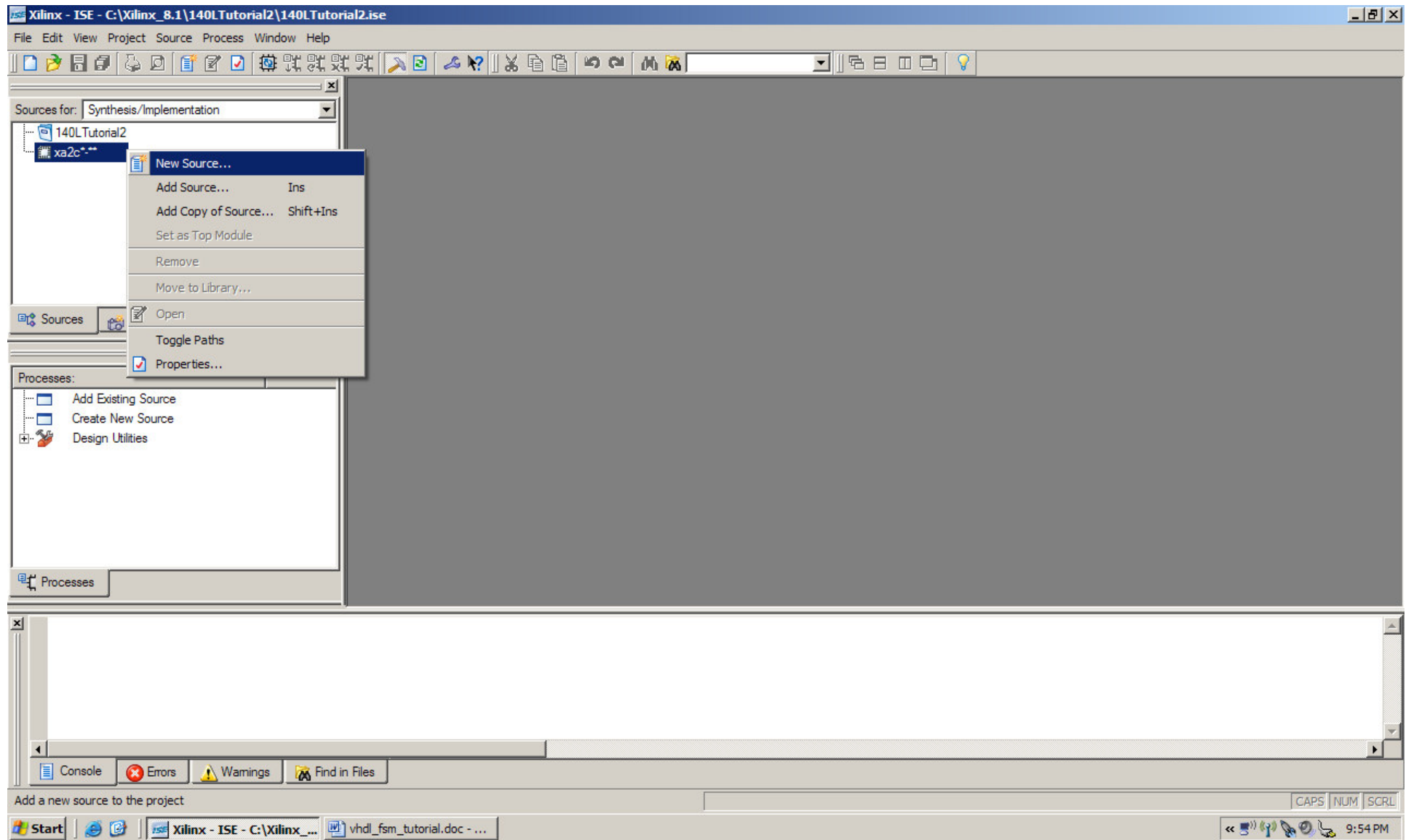
Leave everything as default. You can always add new source files later.
Click Next to continue.



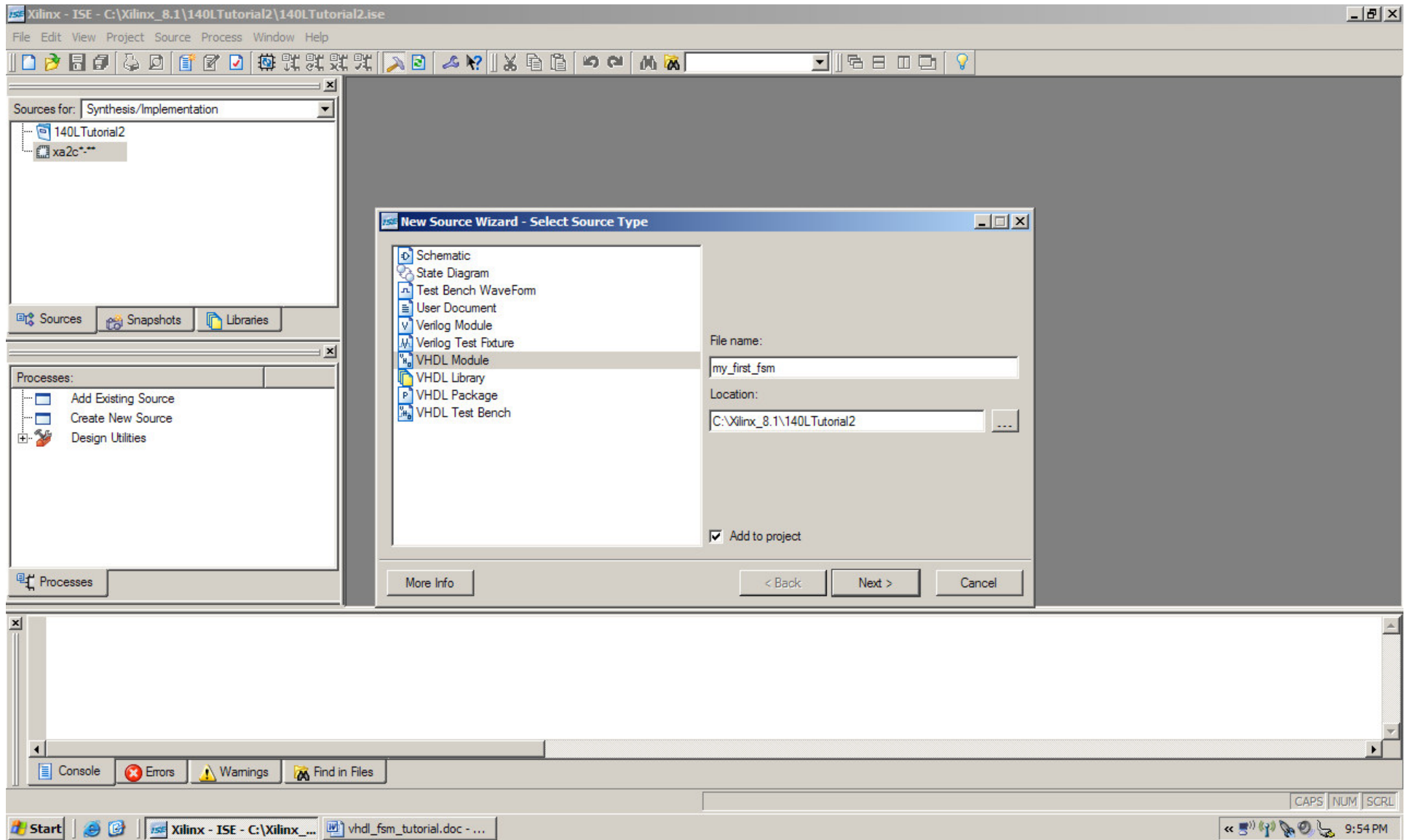
Since you don't have existing sources to add, you can skip this step also. Click Next to continue.



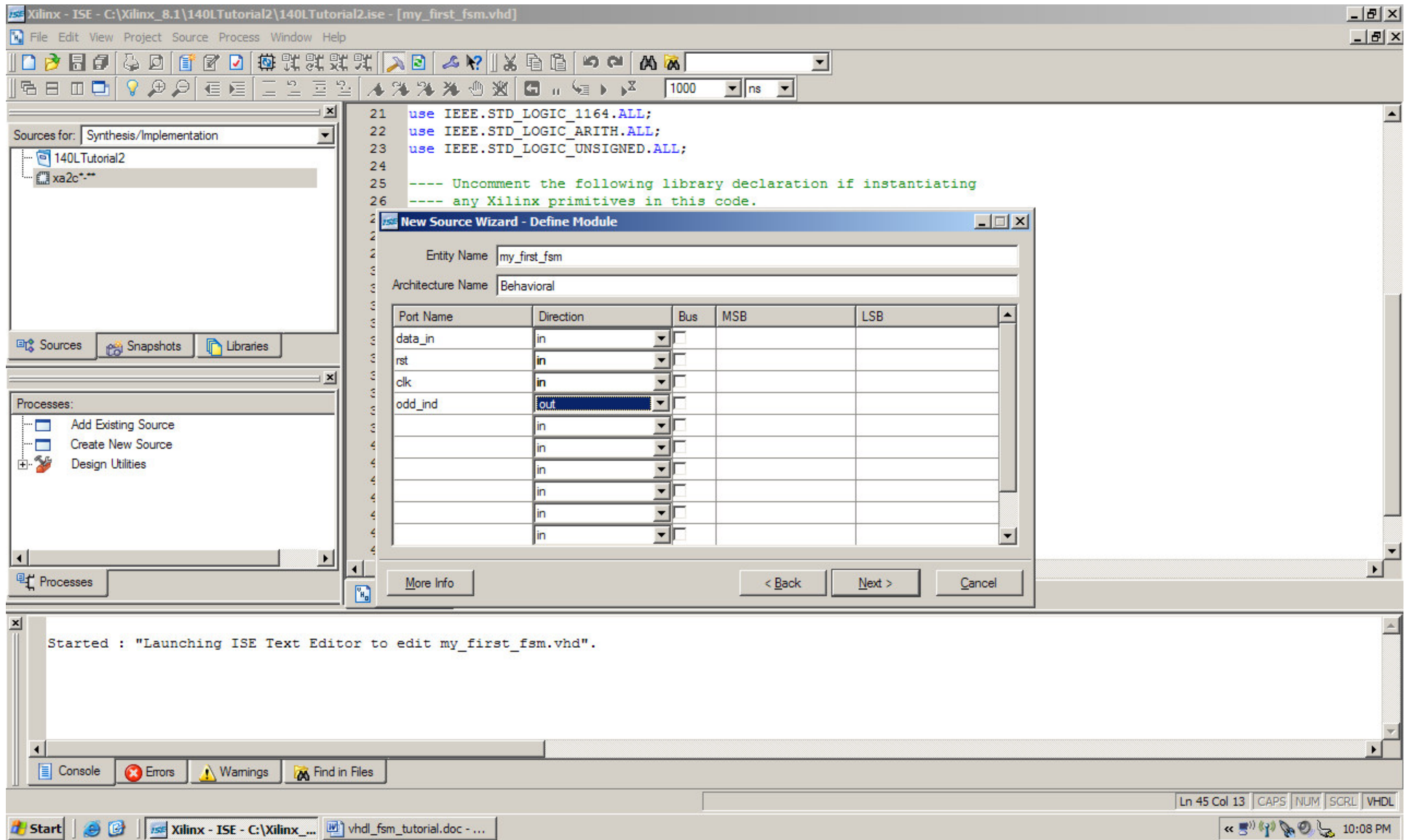
A quick summary... click Finish



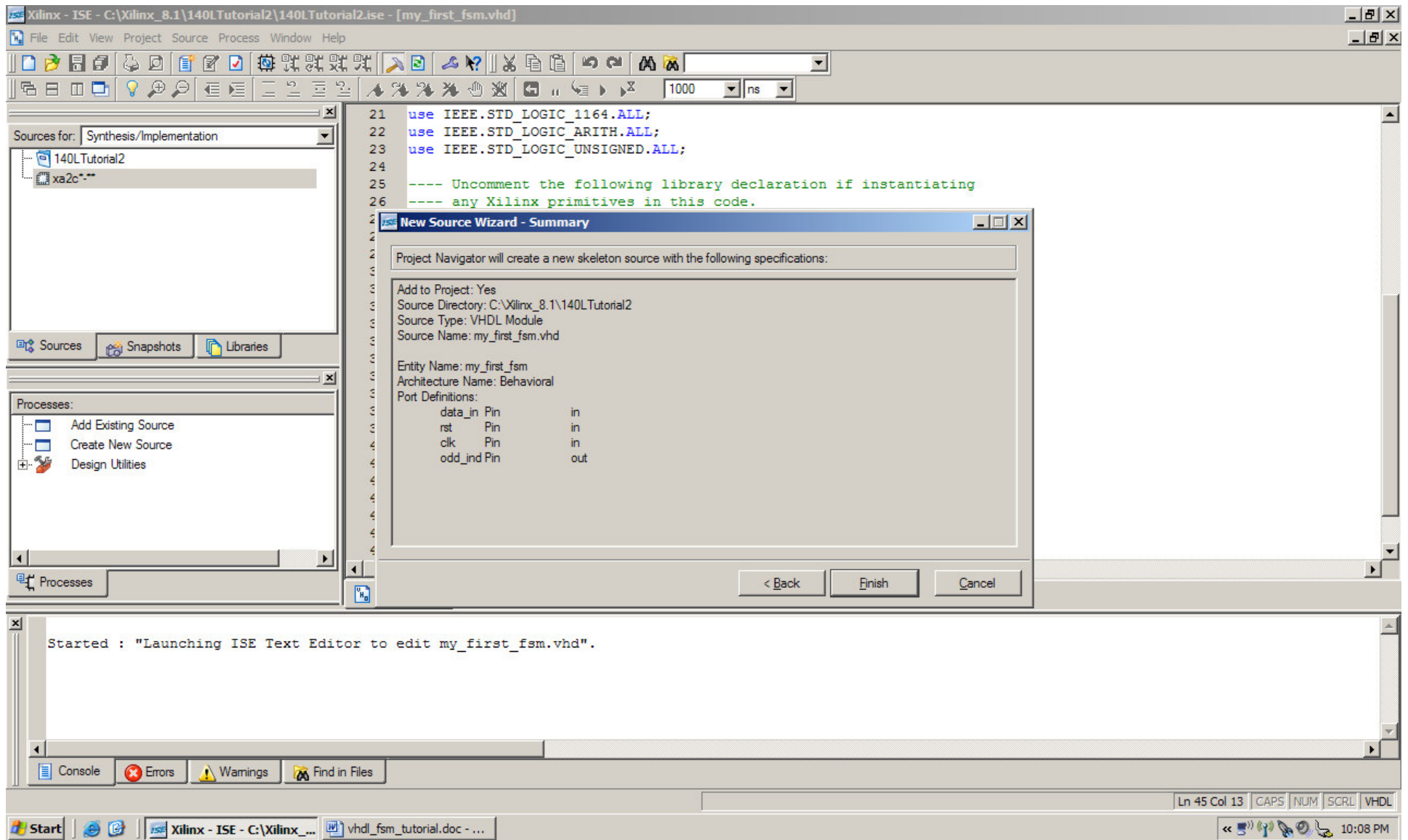
Now, let's create a new VHDL source.



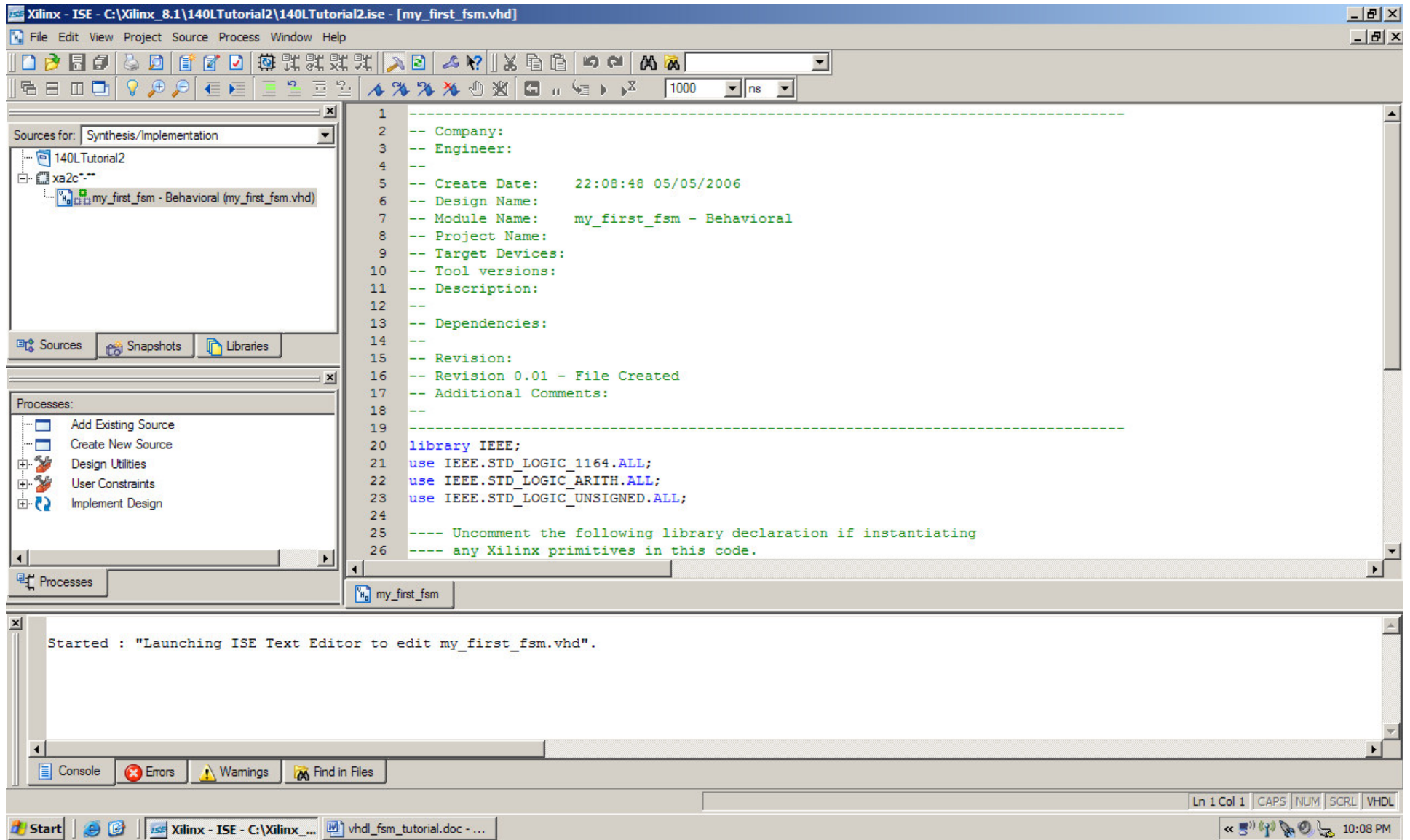
Make sure you select VHDL Module on the left pane and enter the name “my_first_fsm”
Click Next to continue.



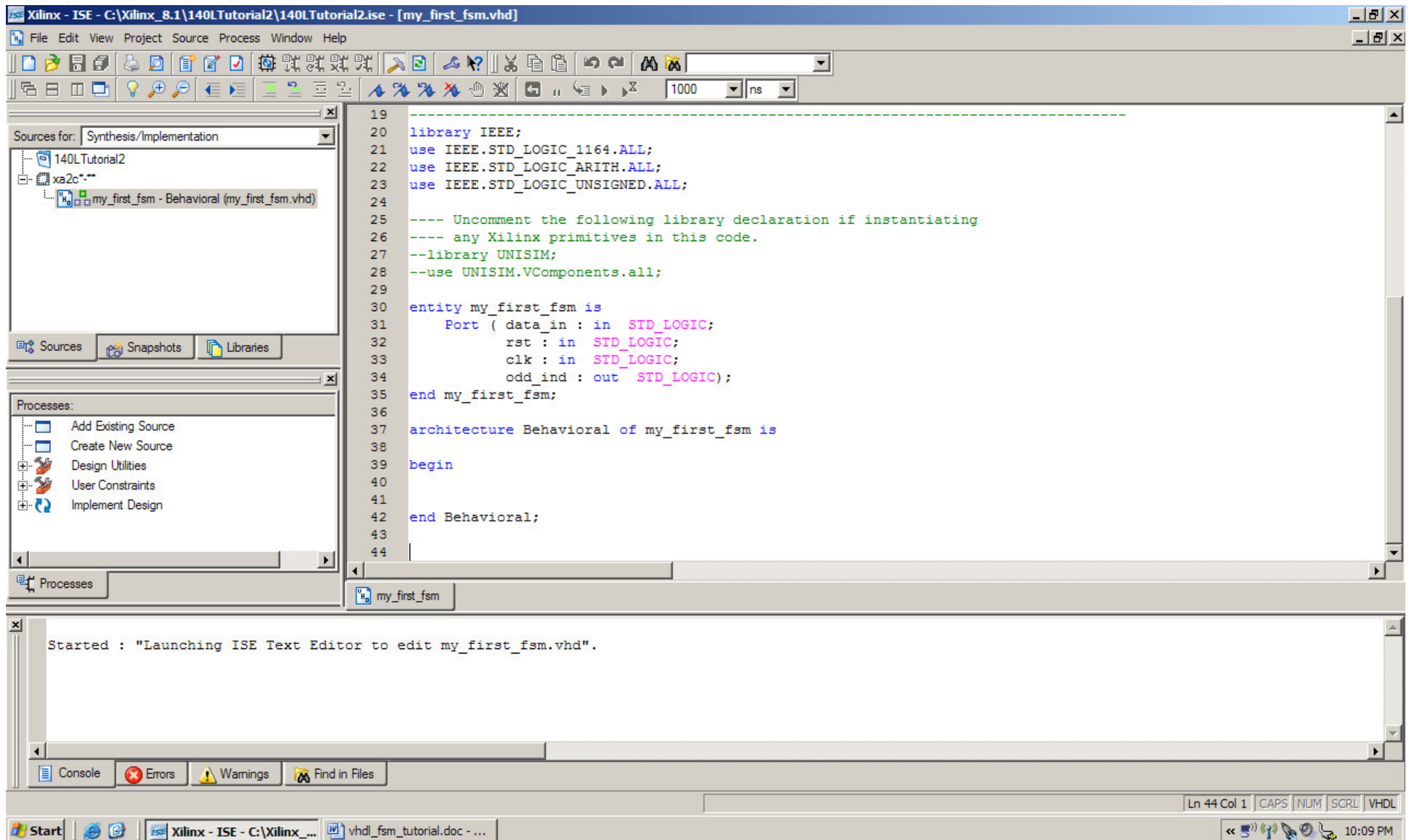
Enter three inputs (data_in, rst, clk) and one output (odd_ind) to indicate an even or odd number of ones detected from the data_in in every clock cycle. Click Next to continue.



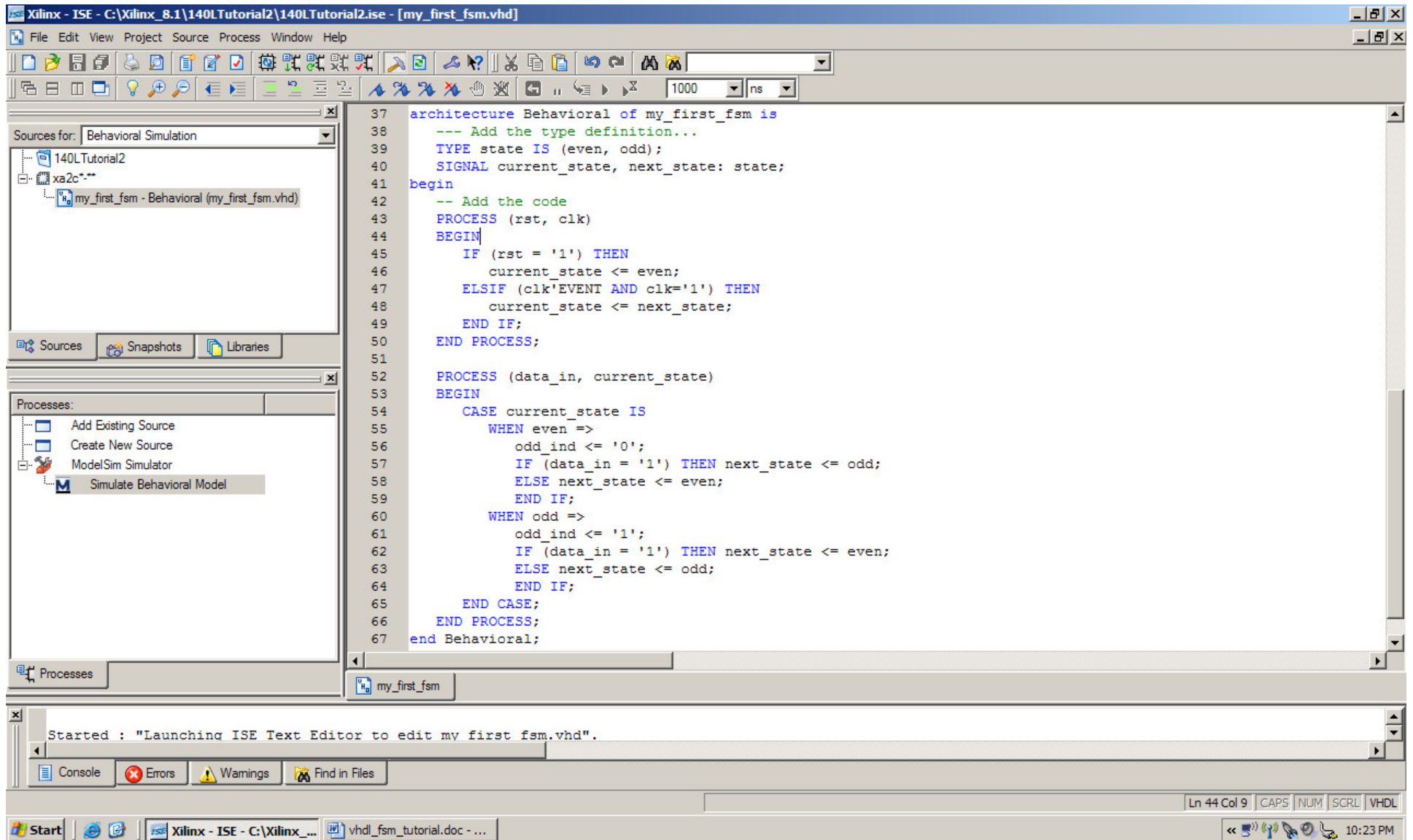
A quick summary... click Finish.



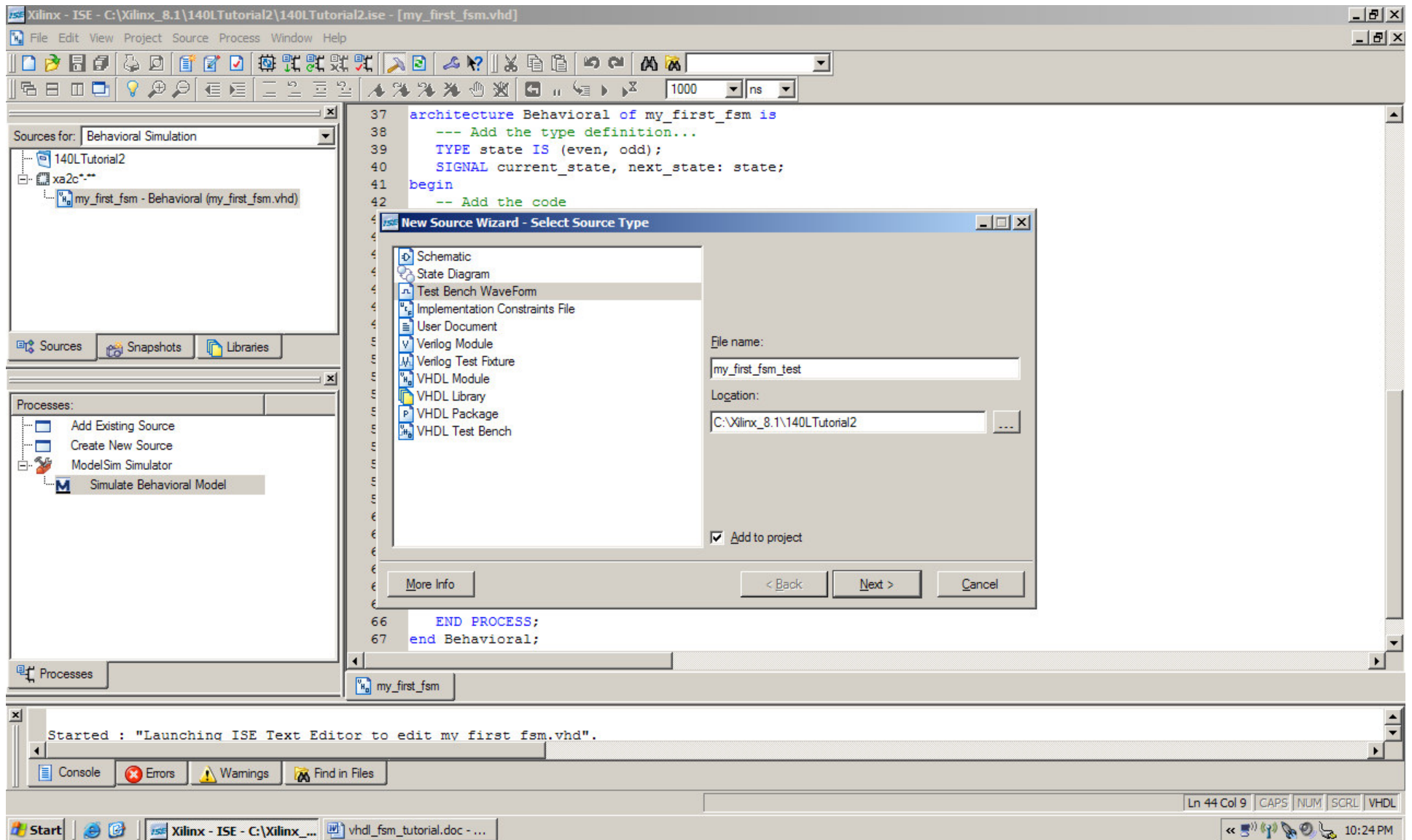
Notice the 3 libraries already included as default.



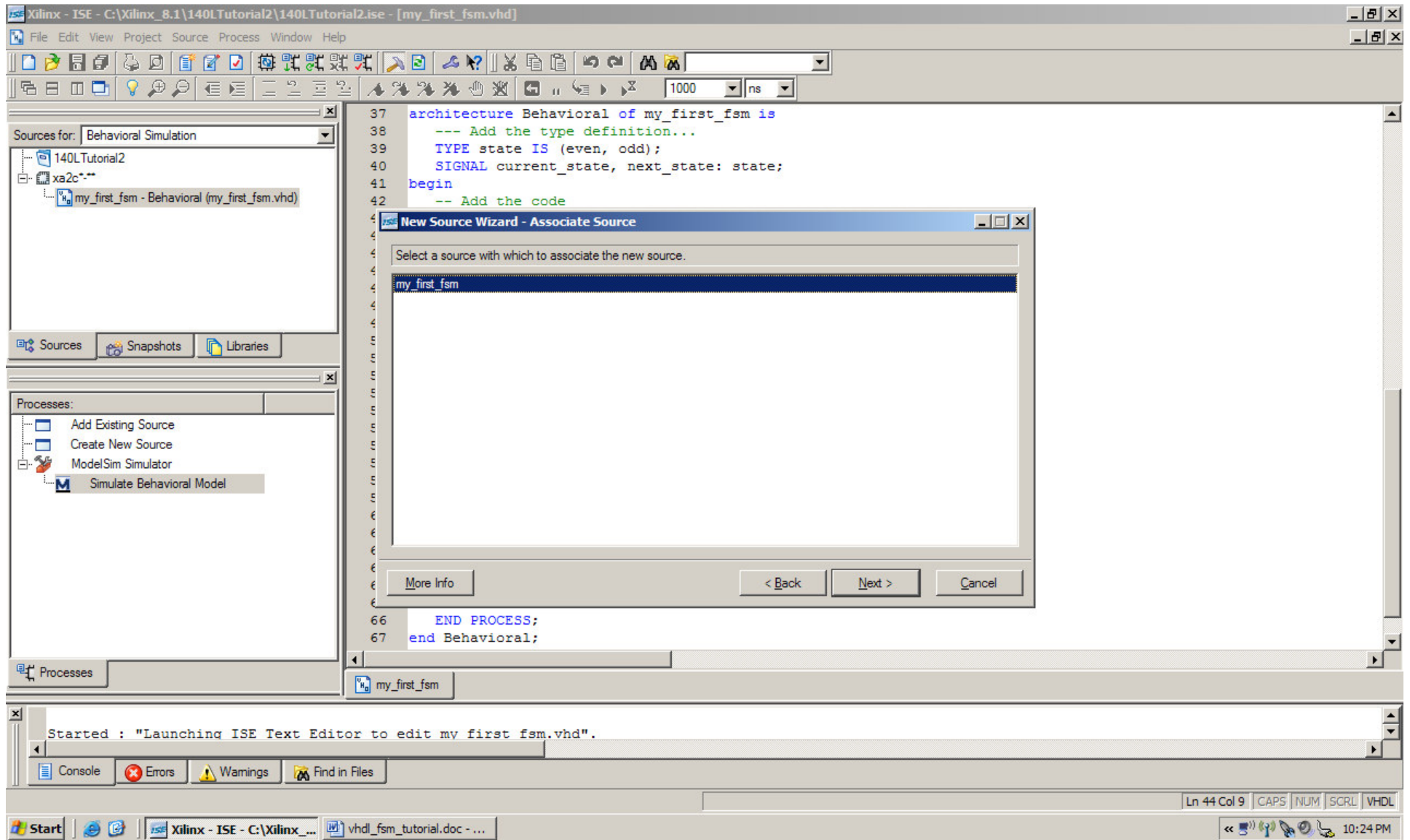
Scroll down and you'll see the skeleton code built for you. Entity is the definition and architecture is the actual logic implementation.



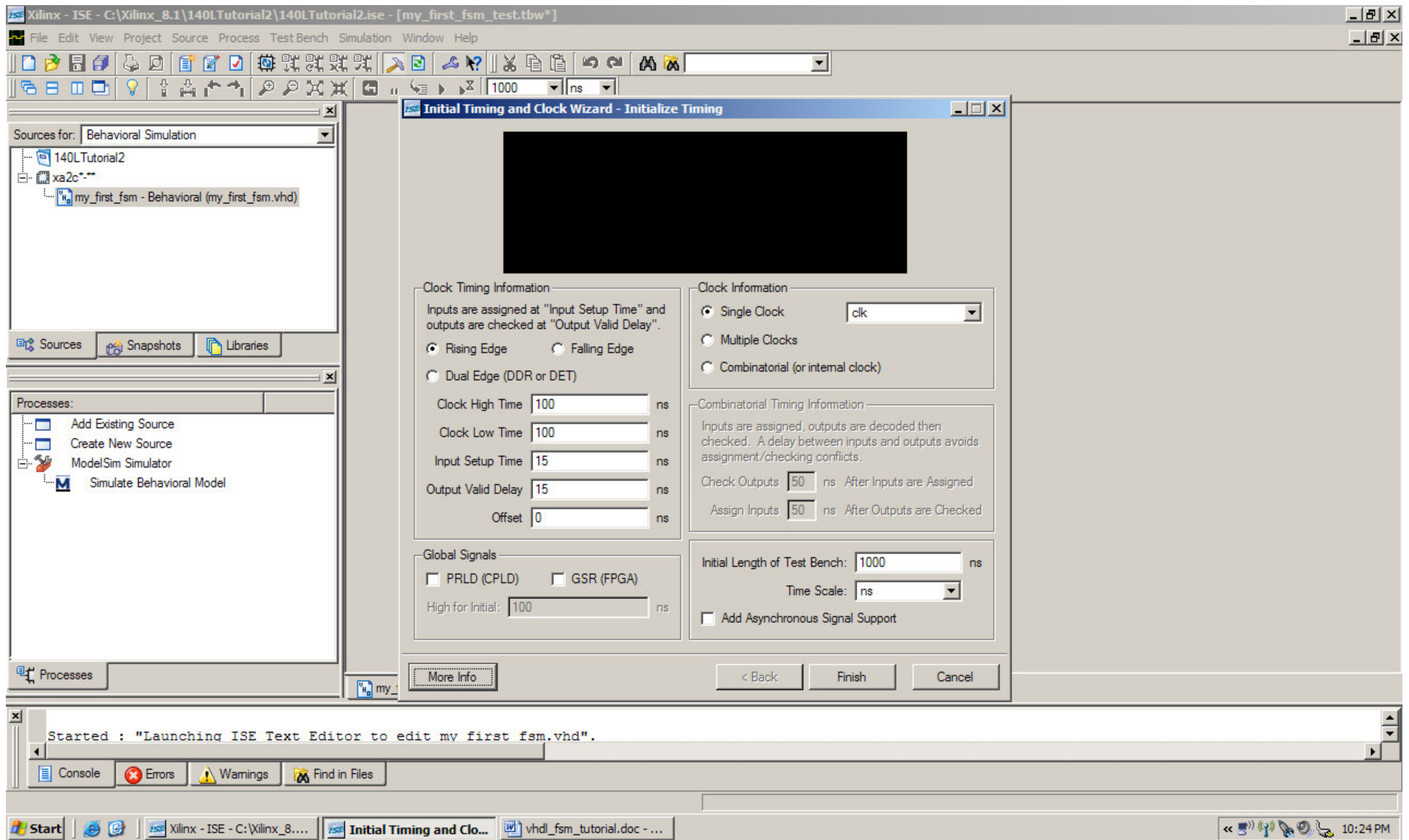
Add the code for the finite state machine.



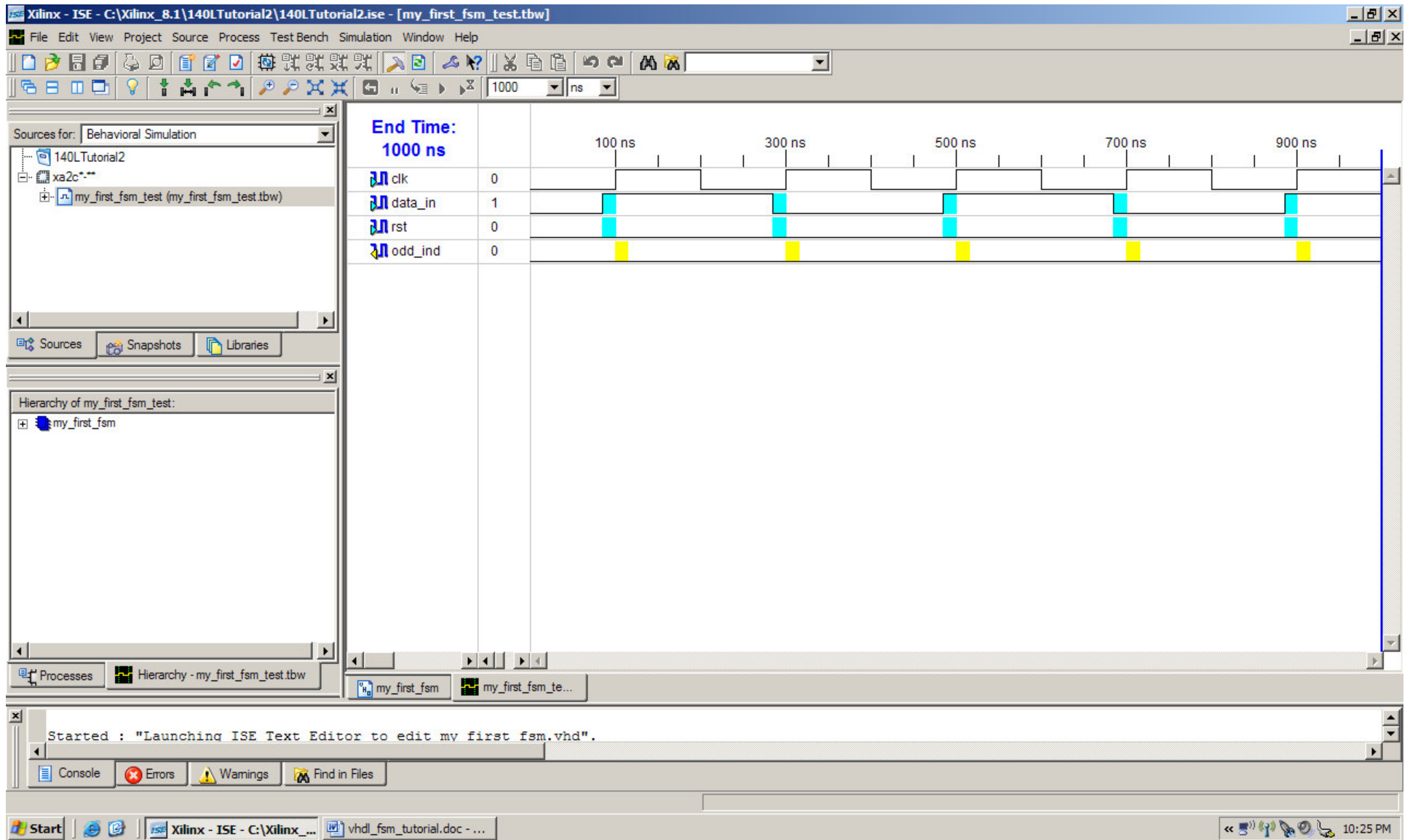
Now, add the test case for the state machine by right clicking on the file and choose “New Source” as you did in previous tutorial.



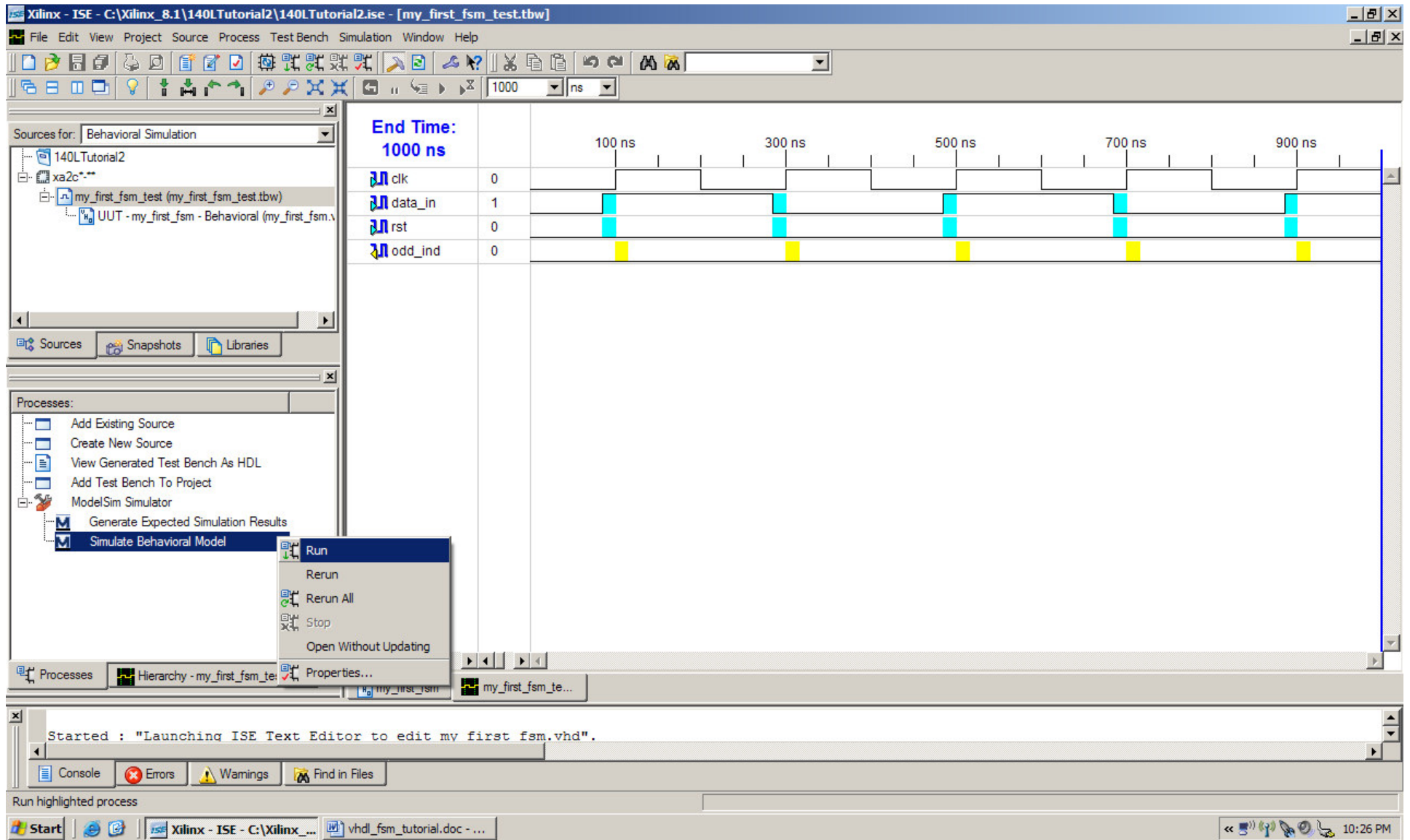
Choose the file you just created to be tested.
Click Next to continue.



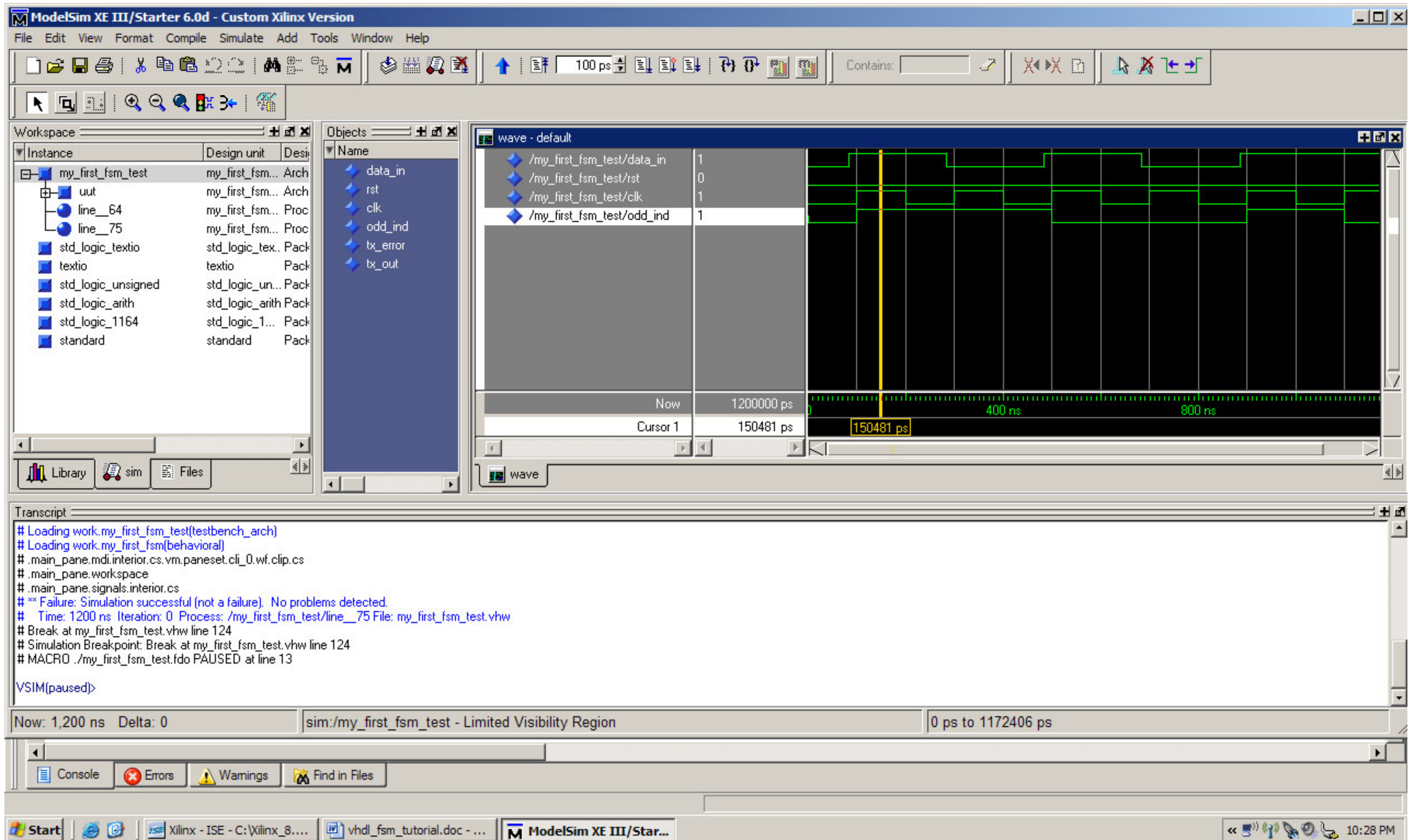
A new window will pop up. Be sure to select the Single Clock and select the “clk” as the clock line. Leave everything else as default.



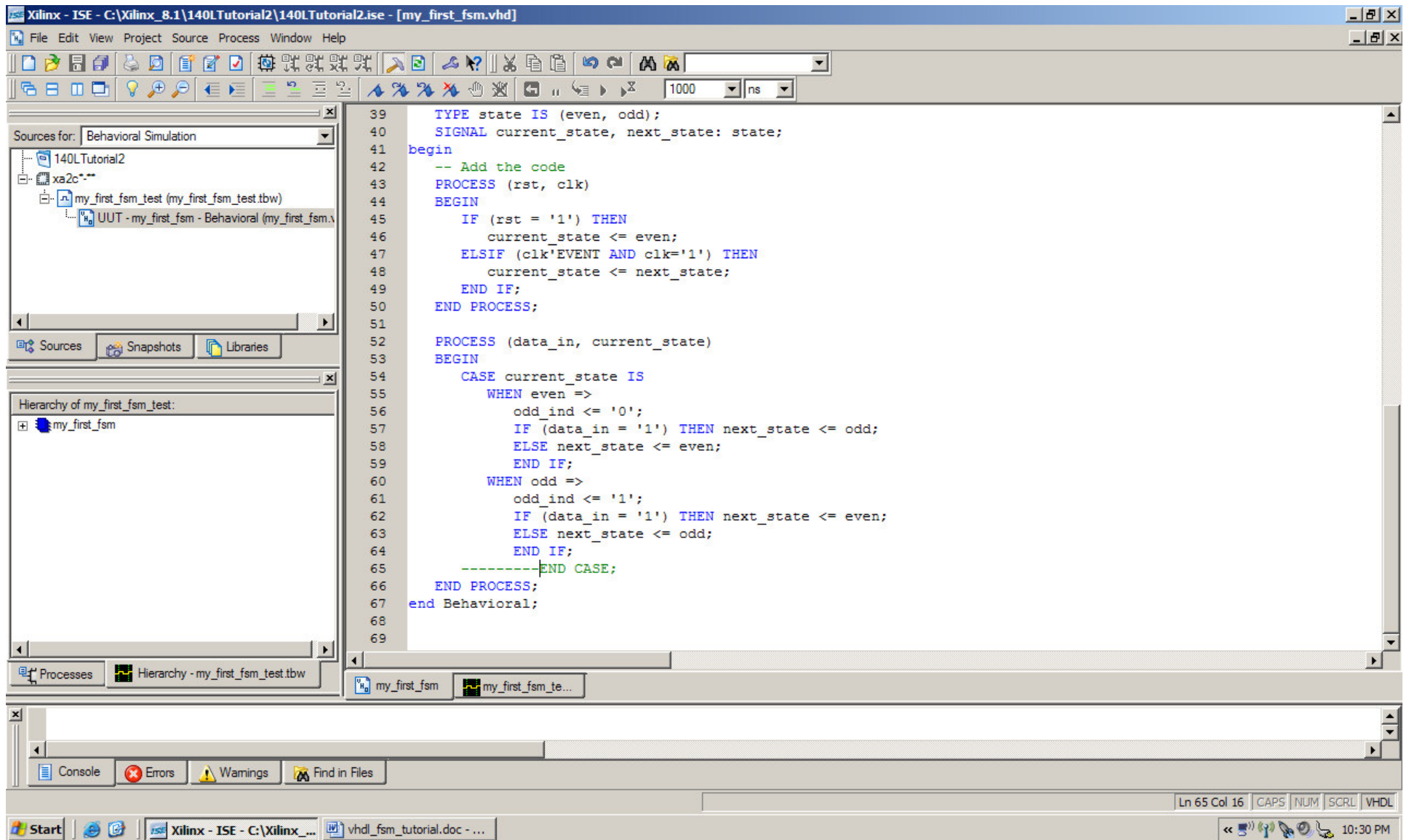
Toggle the `data_in` line and set `rst='0'`.



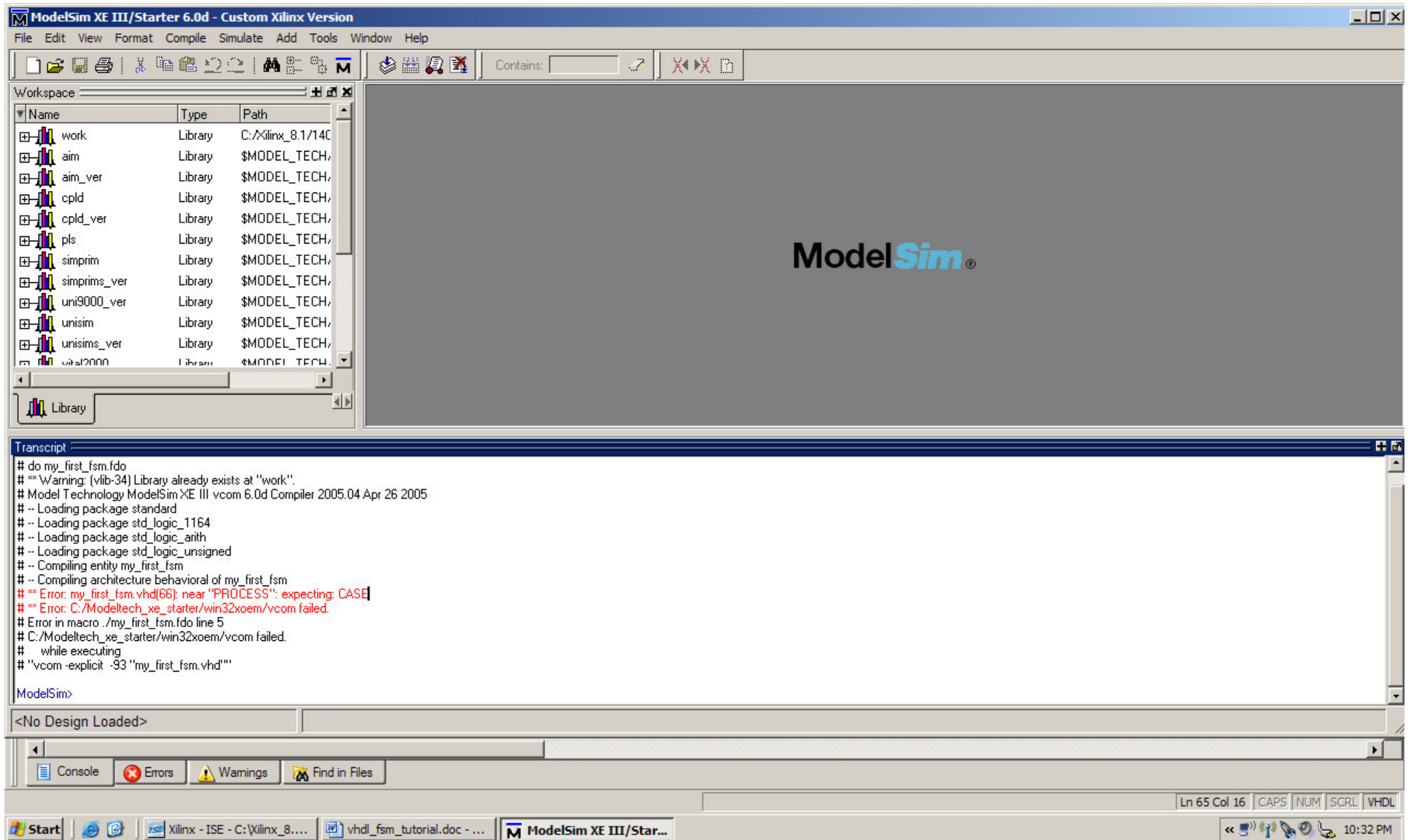
Now, run the simulation.



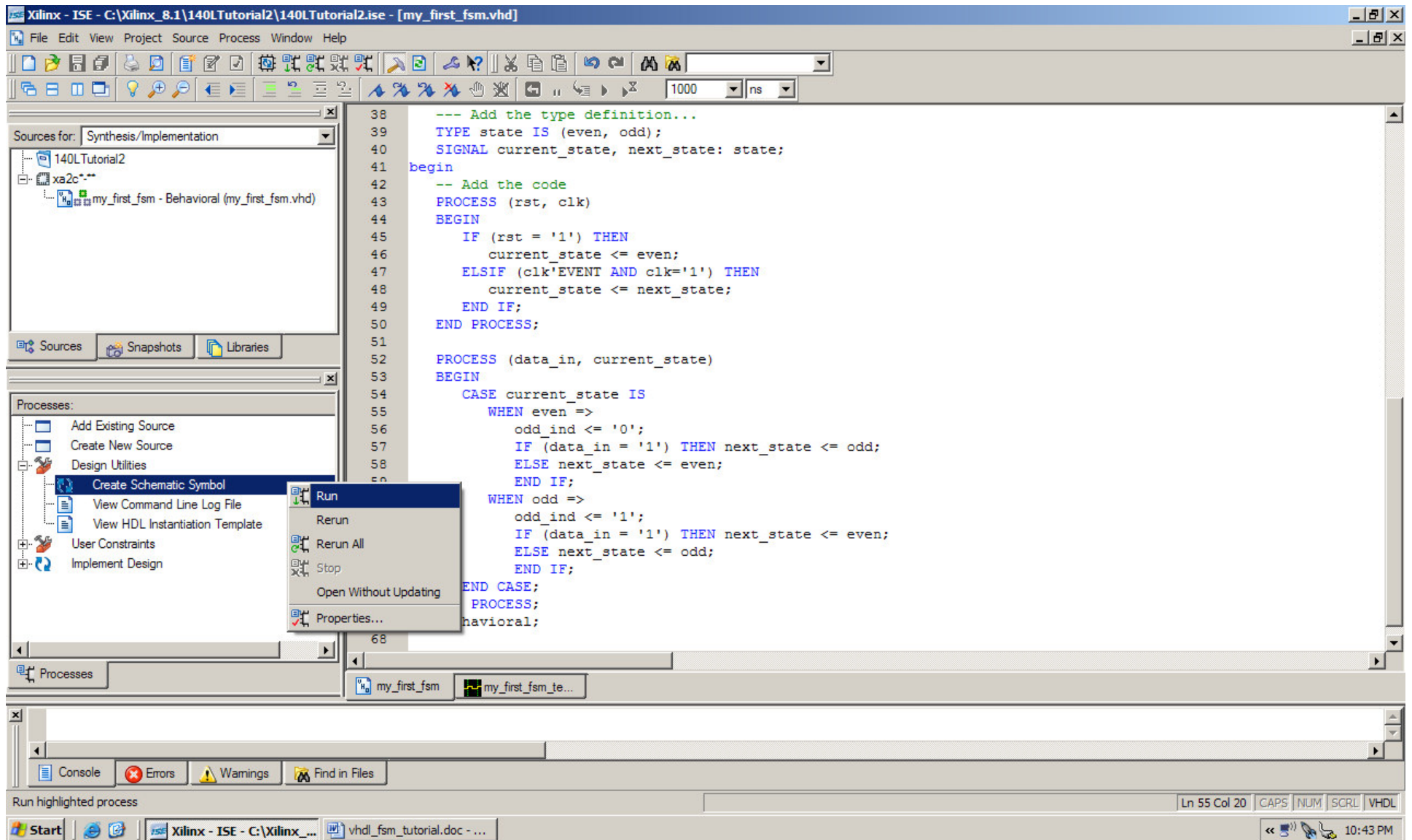
If you typed everything in correctly, you will see the simulation result as expected.



Let's force a mistake by commenting out the 'END CASE' statement and see what happens...



Notice the Modelsim will not run if there is an error.



After making the correction, let's see if we can see what the circuit will look like when we generate it. Select the Sources for "Synthesis/Implementation" in the source upper left window. In the Process window, drill down the "Design Utilities" and run the "Create Schematic Symbol". Also, right click on the "Implement Design" and click "Run". This will translate your code to actual hardware.

Xilinx - ISE - C:\Xilinx_8.1\140LTutorial2\140LTutorial2.ise - [C:\Xilinx_8.1\140LTutorial2\my_first_fsm_html\fit\appletref.htm]

File Edit View Project Source Process Window Help

Sources for: Synthesis/Implementation

140LTutorial2

xa2c...

my_first_fsm - Behavioral (my_first_fsm.vhd)

Sources Snapshots Libraries

Processes:

View HDL Instantiation Template

User Constraints

Implement Design

Synthesize - XST

View Synthesis Report

View RTL Schematic

View Technology Schematic

Check Syntax

Translate

Fit

Generate Programming File

Optional Implementation Tools

Processes

my_first_fsm my_first_fsm_te... C:\Xilinx_8.1\1...

Process "Generate Programming File" completed successfully

Console Errors Warnings Find in Files

Done

Start Xilinx - ISE - C:\Xilinx_... vhdl_fsm_tutorial.doc - ... 10:47 PM

XILINX CPLD Reports

CoolRunner-II

Filter Report | Timing Report

Fitter Report

Summary
Errors/Warnings
Logic
Inputs
Function Blocks
Equations
Pin List
Compiler Options
Text Report
Help

Summary

Design Name	my_first_fsm
Fitting Status	Successful
Software Version	I.27
Device Used	XA2C32A-6-VQ44
Date	5- 5-2006, 10:46PM

RESOURCES SUMMARY

Macrocells Used	Pterms Used	Registers Used	Pins Used	Function Block Inputs Used
1/32 (4%)	1/112 (1%)	1/32 (4%)	4/33 (13%)	1/80 (2%)

PIN RESOURCES

Signal Type	Required	Mapped	Pin Type	Used	Total
Input	1	1	I	0	1
Output	1	1	I/O	2	24

Equation Display Style: VHDL

If all goes well, you can see all green check marks when you drill down the “Implementation Design”.

Xilinx - ISE - C:\Xilinx_8.1\140LTutorial2\140LTutorial2.ise - [C:\Xilinx_8.1\140LTutorial2\my_first_fsm_html\fit\appletref.htm]

File Edit View Project Source Process Window Help

Sources for: Synthesis/Implementation

- 140LTutorial2
 - xa2c*-**
 - my_first_fsm - Behavioral (my_first_fsm.vhd)

Processes:

- View HDL Instantiation Template
- User Constraints
- Implement Design
 - Synthesize - XST
 - View Synthesis Report
 - View RTL Schematic
 - View Technology Schematic
 - Check Syntax
 - Translate
 - Fit
 - Generate Programming File
 - Optional Implementation Tools

Equation

Run

Rerun

Rerun All

Stop

Open Without Updating

Properties...

CPLD Reports

Filter Report | Timing Report

Fitter Report

Summary

Errors/Warnings

Logic

Inputs

Function Blocks

Equations

Pin List

Compiler Options

Text Report

Help

Summary

Design Name	my_first_fsm
Fitting Status	Successful
Software Version	I.27
Device Used	XA2C32A-6-VQ44
Date	5- 5-2006, 10:46PM

RESOURCES SUMMARY

Macrocells Used	Pterms Used	Registers Used	Pins Used	Function Block Inputs Used
1/32 (4%)	1/112 (1%)	1/32 (4%)	4/33 (13%)	1/80 (2%)

PIN RESOURCES

Signal Type	Required	Mapped	Pin Type	Used	Total
Input	1	1	I	0	1
Output	1	1	I/O	2	24

Started : "Launching RTL Schematic Viewer for my_first_fsm.ngr".

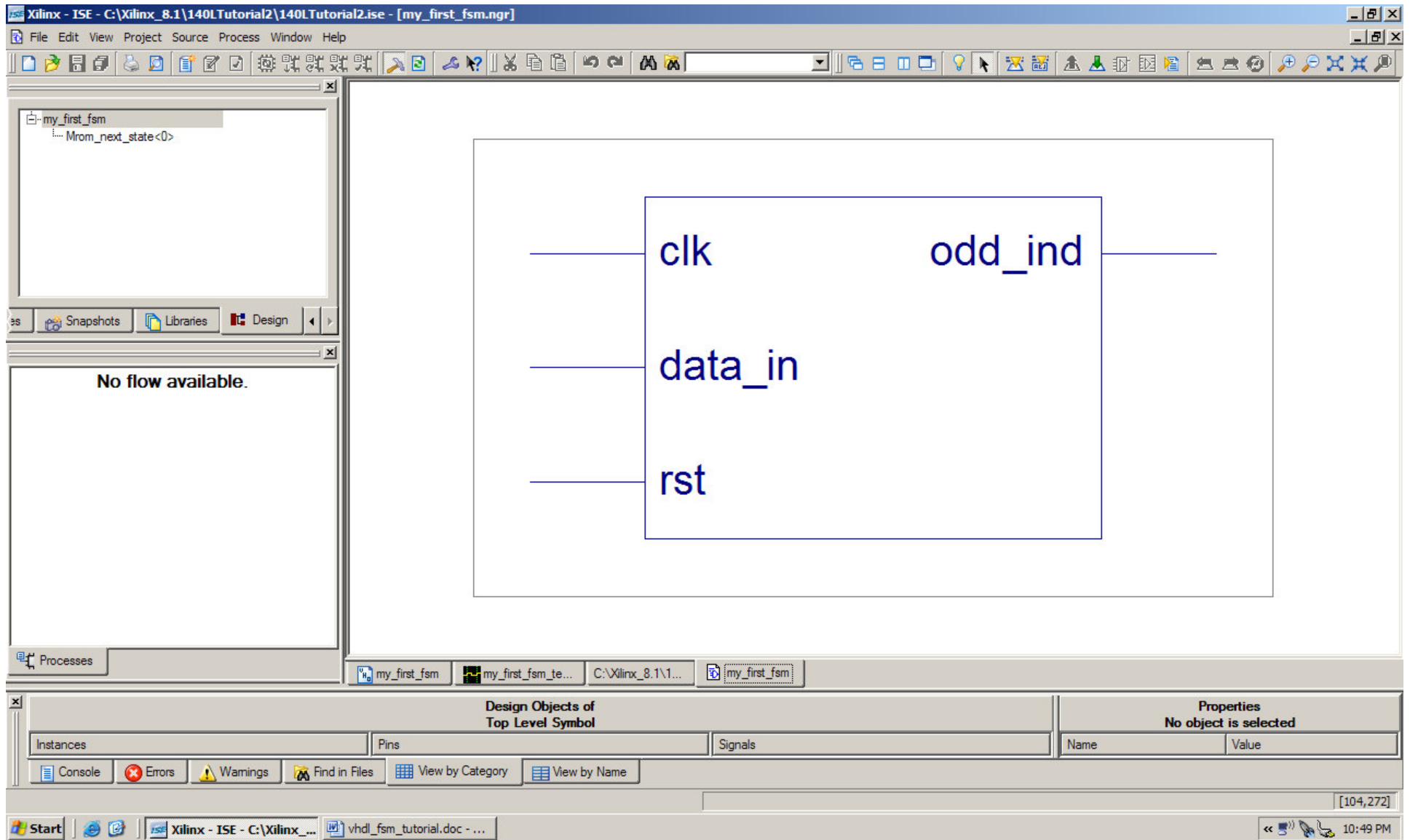
Console Errors Warnings Find in Files

Run highlighted process

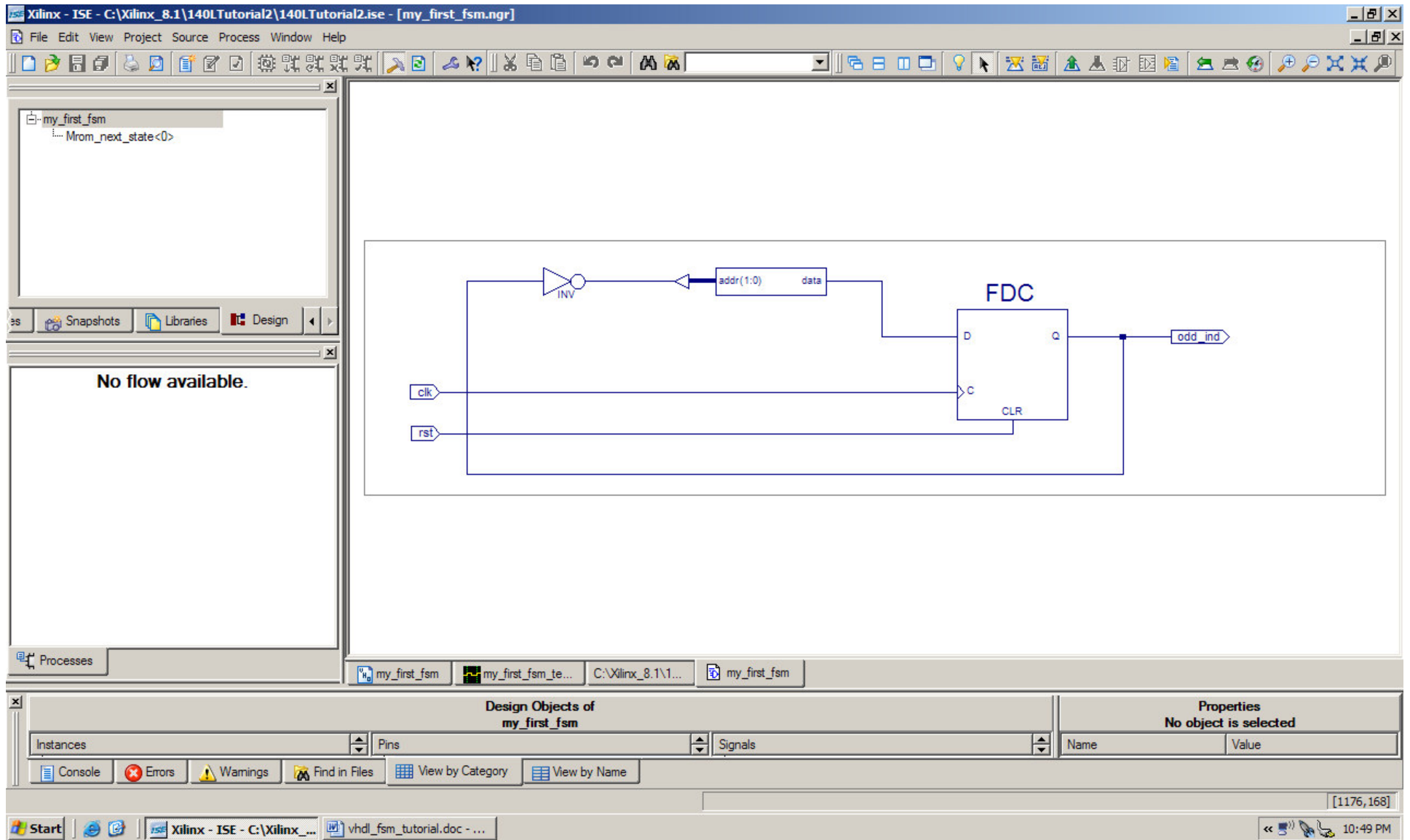
Start | Xilinx - ISE - C:\Xilinx_... | vhdl_fsm_tutorial.doc - ...

10:48 PM

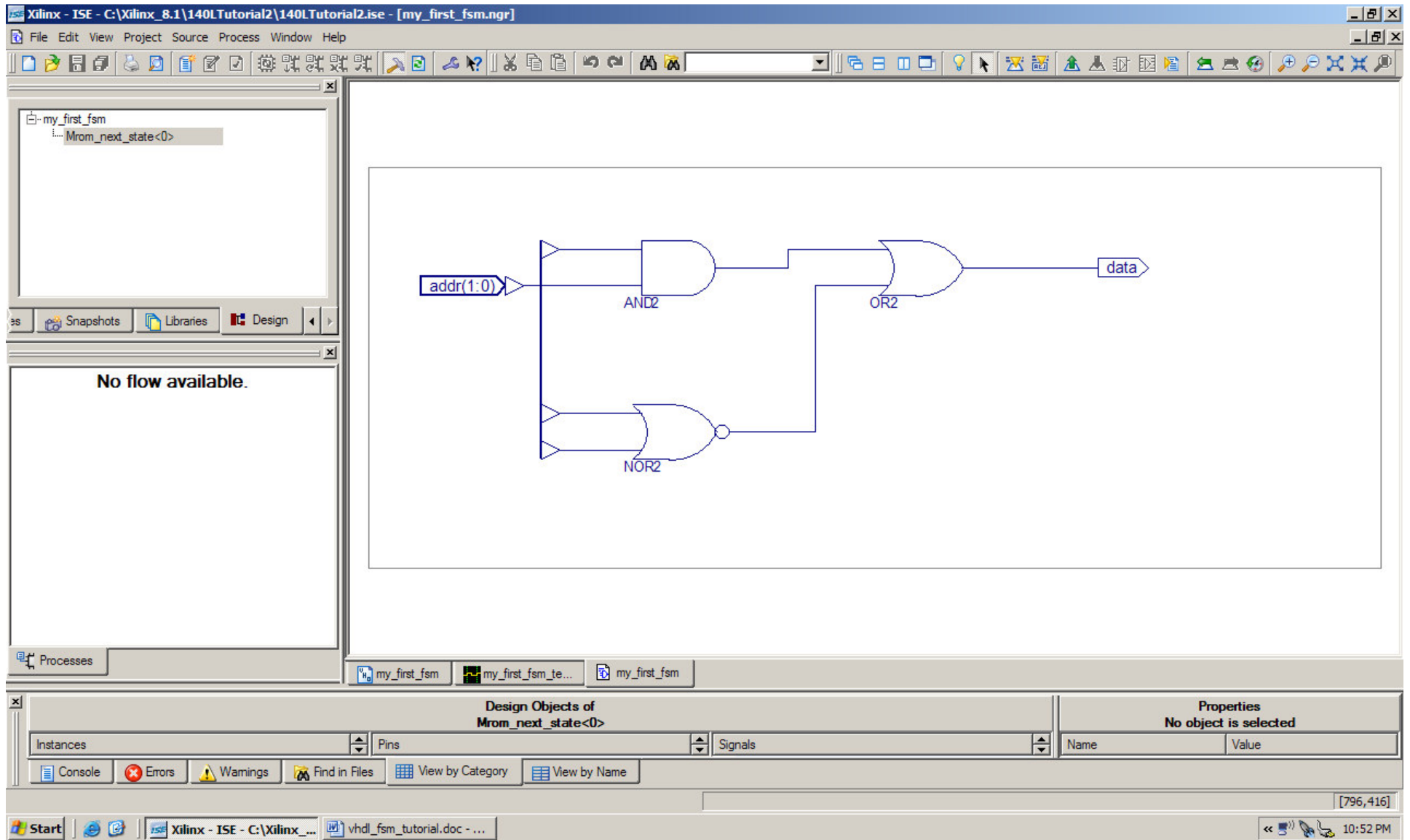
You can see the actual logic circuit by running the “View RTL Schematic”.



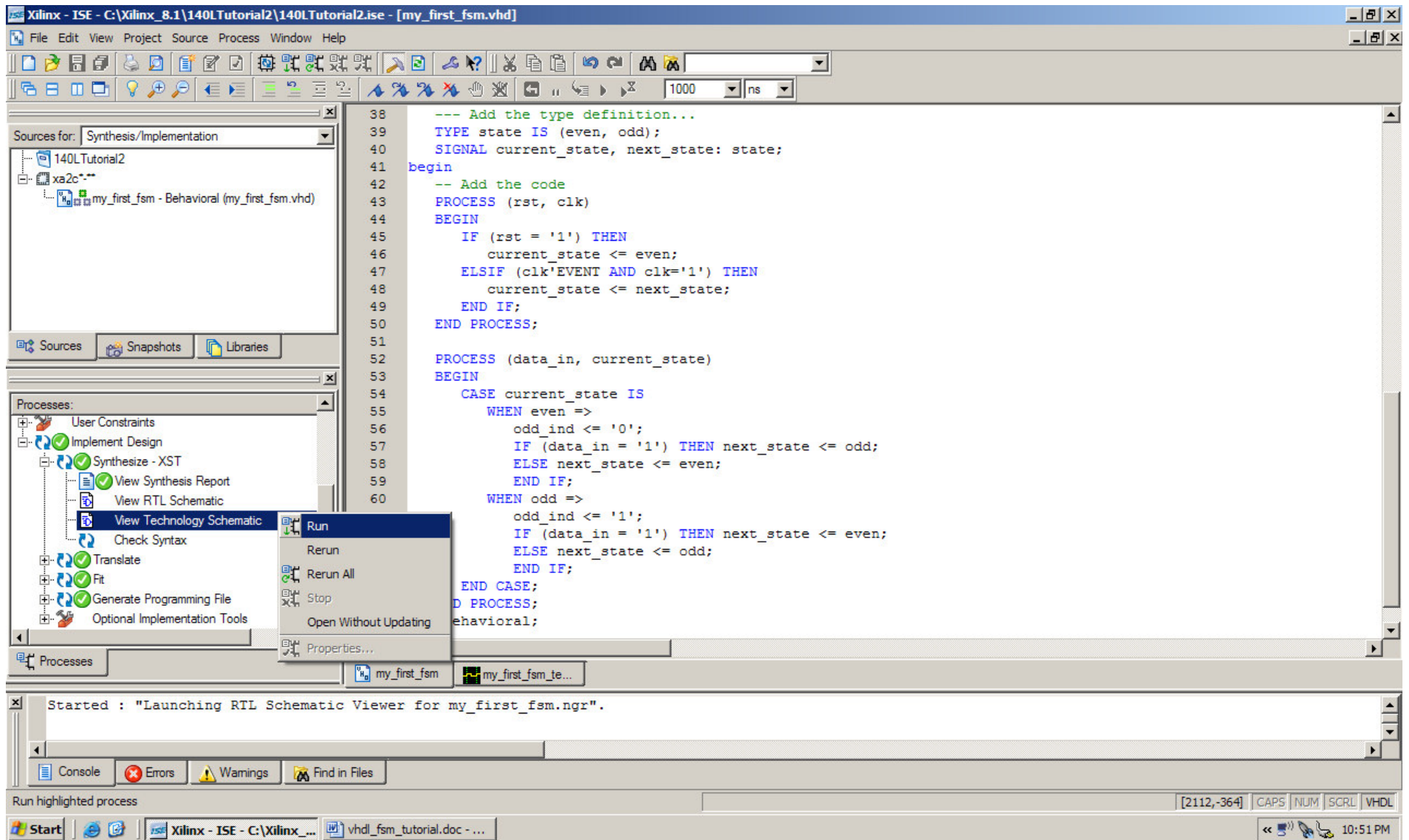
This is the top level view. Double click on the circuit itself to drill down the detail.



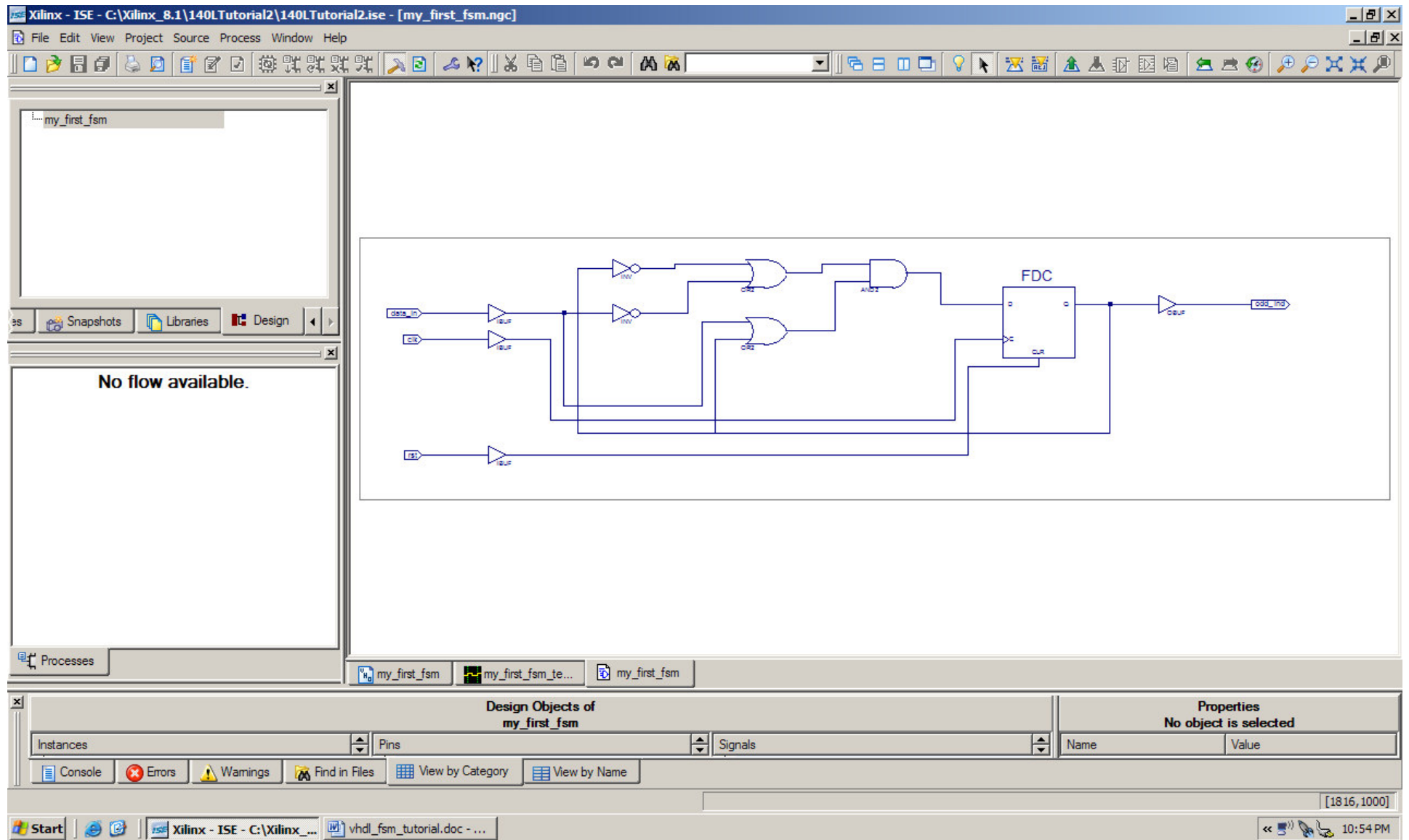
You can see the circuit actually created. Double click on the block next to the inverter.



You now see the logic gates which implemented the logic of this circuit.
Close this window...



You can also see the entire design by running the “View Technology Schematic”.



After double click on the top level circuit, you can see the entire design.
Now you are ready to do lab3.part3. Good luck!