HO CHI MINH UNIVERSITY OF TECHNOLOGY

COMPUTER SCIENCE AND ENGINEERING

Digital Design with the Verilog

CO2010

Laborary 2

Using RTL Design to implement a combinational logic

HCMC, 2017

Objective 3
Requirement 3
I. The combination circuit using RLT Method 3
1. Design Mux 2_to_1 module using “assign”. 3
2. RTL Viewer. 5
3. Implementation Mux 4_to_1 using Schematic block 6
II. Exercise: 10

Objective

- ***Practicing RTL Desing in the combinational circuit (with "assign").***
- ***How to design circuit by creating a schematic.***
- ***How to check the RTL schematic using Quartus II.***

Requirement

- ***Submit file code Verilog (.v), file block diagram/schematic (.bdf), and a waveform screen when checking circuit under .zip or .rar with name LAB2_MSSV.[zip,rar].***

I. The combination circuit using RLT Method

1. Design Mux 2_to_1 module using "assign".

Step 1. Create new project in Quartus II IDE.

<u>Note:</u> FPGA chip is Cyclone IV EP4CGX150DF31C7.!

Step 2. Create new Verilog HDL file.

Step 3. Describe Mux 2_to_1 module as below code:

```
module mux_2to1 (out, sel, in0, in1);
   output  out;
   input   sel;
   input   in0;
   input   in1;

   assign out = (sel == 1'b0) ? in0 : in1;

endmodule
```

The "`assign out = (sel == 1'b0) ? in0 : in1;`" means if any input signals such as `sel, in0, or in1` changes, the output signal will alter as followed rules:

- If `sel == 1'b0`, then `out = int0.`
- If `sel == 1'b1`, then `out = int1.`

Step 4. Save Verilog filename correctponse with module's name.

Step 5. Set mux_2to1 as Top level module.

Step 6. Processing → Start → Start Analysis & Synthesis. If succeed, there is a Flow Summary as the below picture.

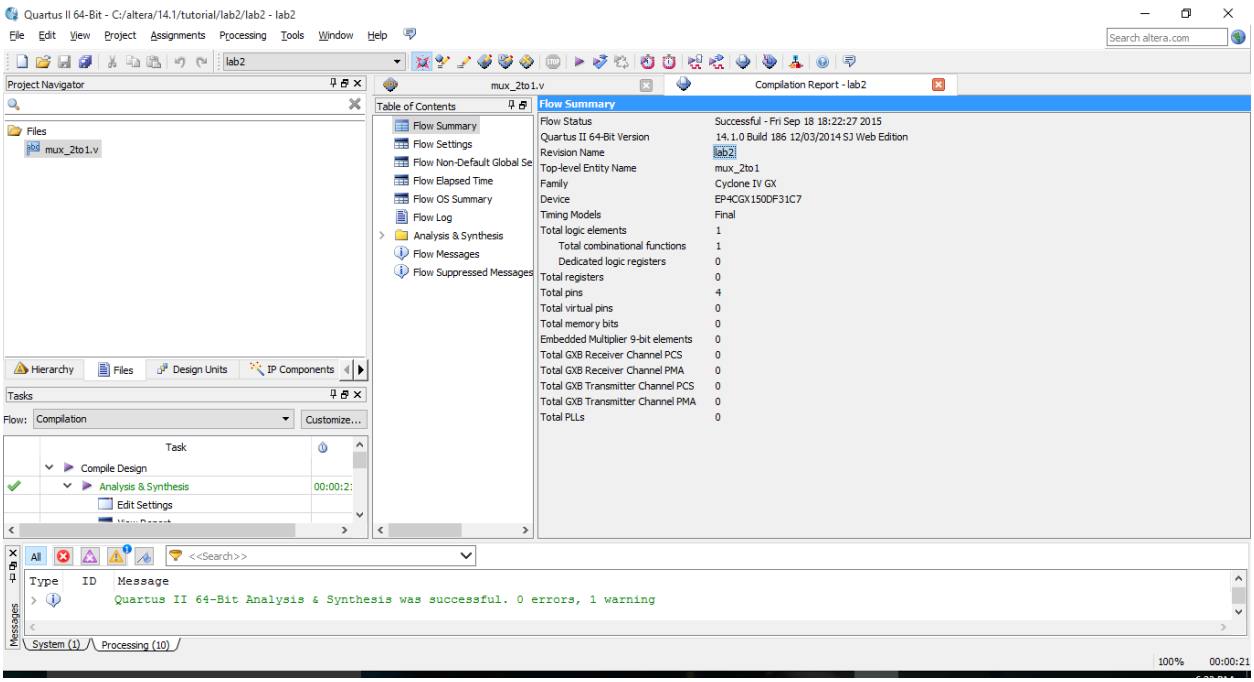


Figure Error! No text of specified style in document.**-1 Flow summary for Analysis and Synthesis process.**

Step 7. Pin assignment for INPUT, OUTPUT signals by choosing .As**signments → Pin Planner**. The Pin Planner will appear as the below image.

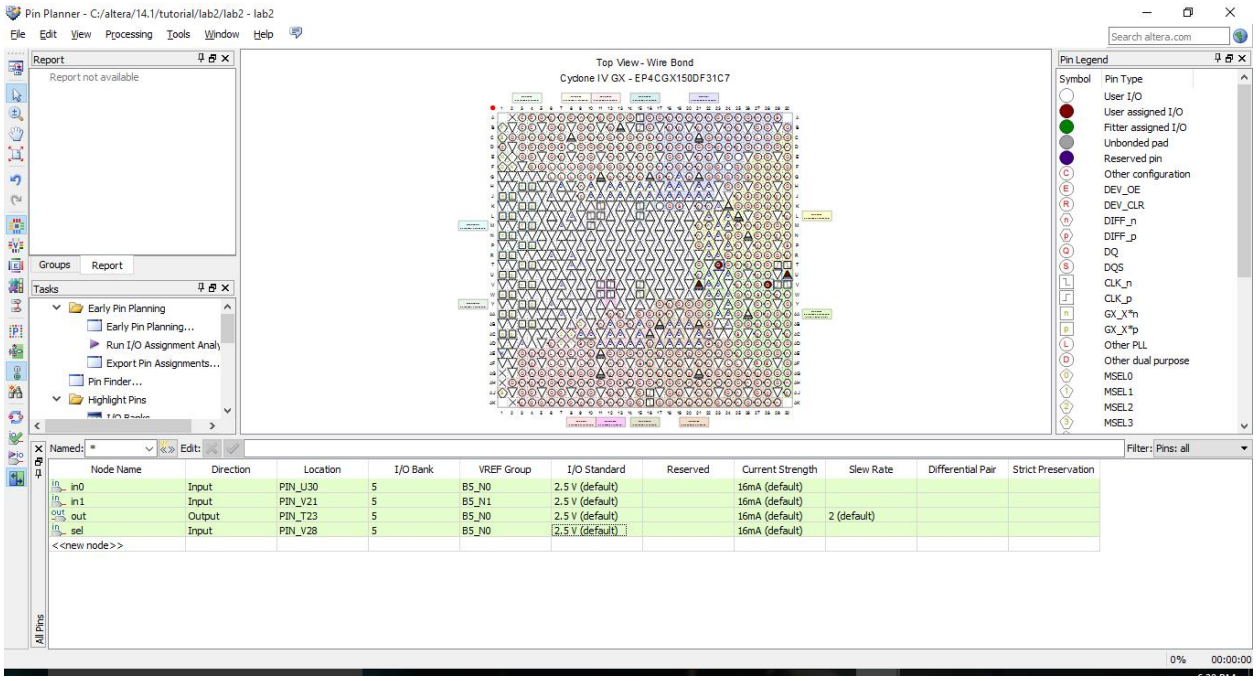


Figure Error! No text of specified style in document.**-2 Pin Planner.**

Assign pins as the Table 2-1:

Table Error! No text of specified style in document.**-1 Pin Assignment.**

Tín hiệu	Loại	Tên	Chân
out	output	LEDR[0]	PIN_T23
sel	input	SWITCH[0]	PIN_V28
in0	input	SWITCH[1]	PIN_U30
in1	input	SWITCH[2]	PIN_V21

Step 8. Compile, Install and Check the logic in DE2i-150 board by Processing → Start Compilation (or press Ctr + L)

2. RTL Viewer.

To see RTL view of a module, first, we have to Analysis and Synthesis. Then **Set top level** for the module which needs to see RTL design. And in this lab, it is **mux_2to1.v.**

Double Click to **RTL viewer** option in **Task** window.

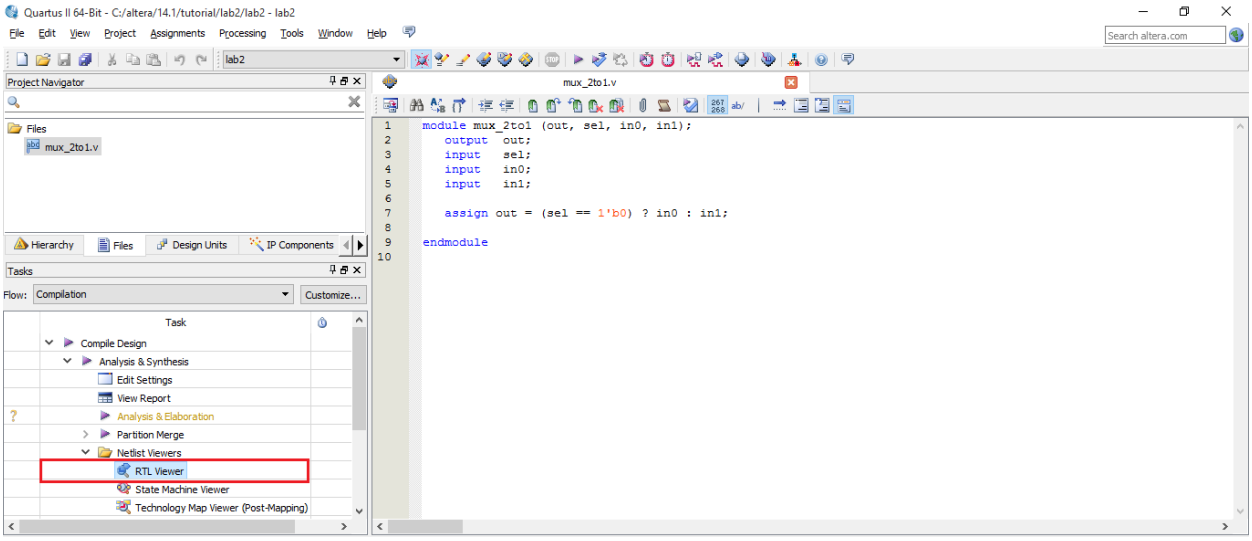


Figure Error! No text of specified style in document.**-3 RTL Viewer Option.**

Then we get the result in RTL Viewer window:

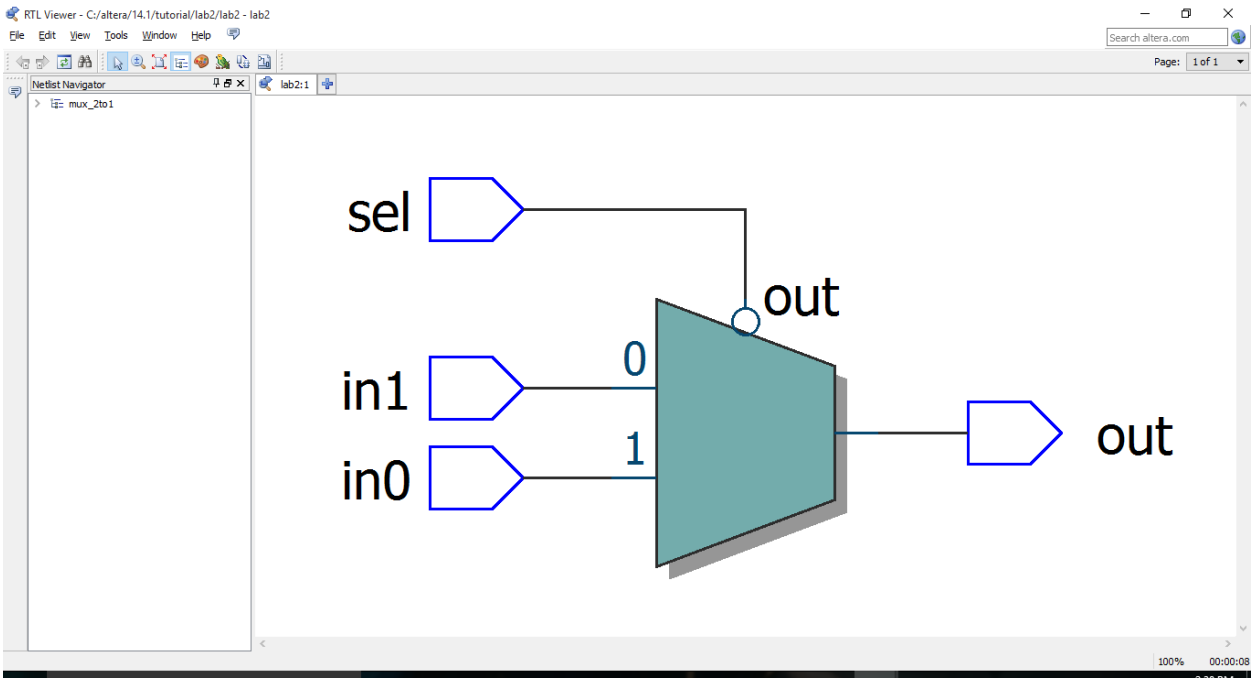


Figure Error! No text of specified style in document.**-4 mux 2to1 structure in RTL.**

3. Implementation Mux 4_to_1 using Schematic block

Step 9. Create mux 2to1 symbol to use in designing by schematic.

+ In Project Navigator window → choose File tab → next choose mux_2to1.v file → Create Symbol Files for Current File

+ Or File → Create/Upload → Create Symbel Files for Current File.

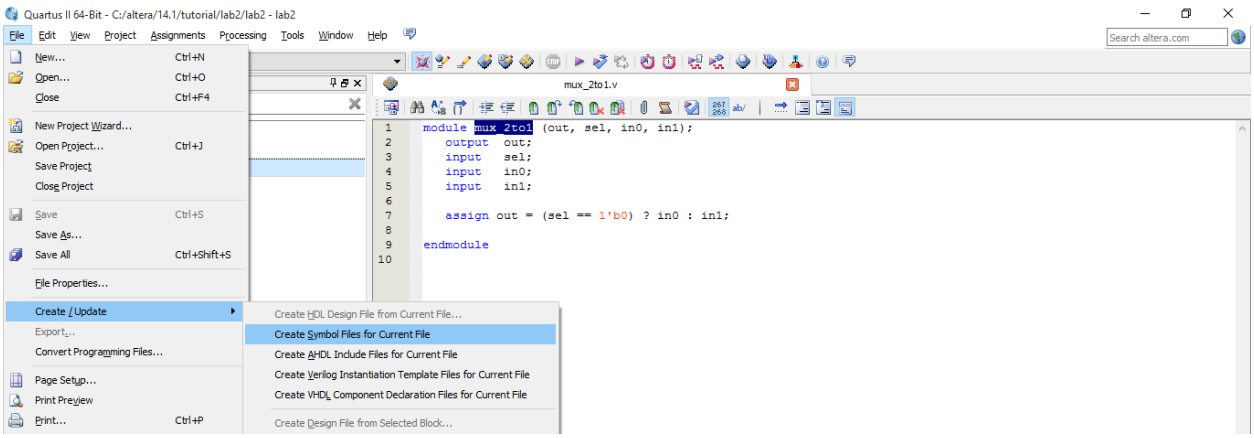


Figure Error! No text of specified style in document.**-5 Create symbol for mux_2to_1 file.**

Step 10. Create Schematic file.

File → New or 🗋 symbol in an accessory bar. Then select **Block Diagram/Schematic File** type and **OK**. The window to design schematic is display as below:

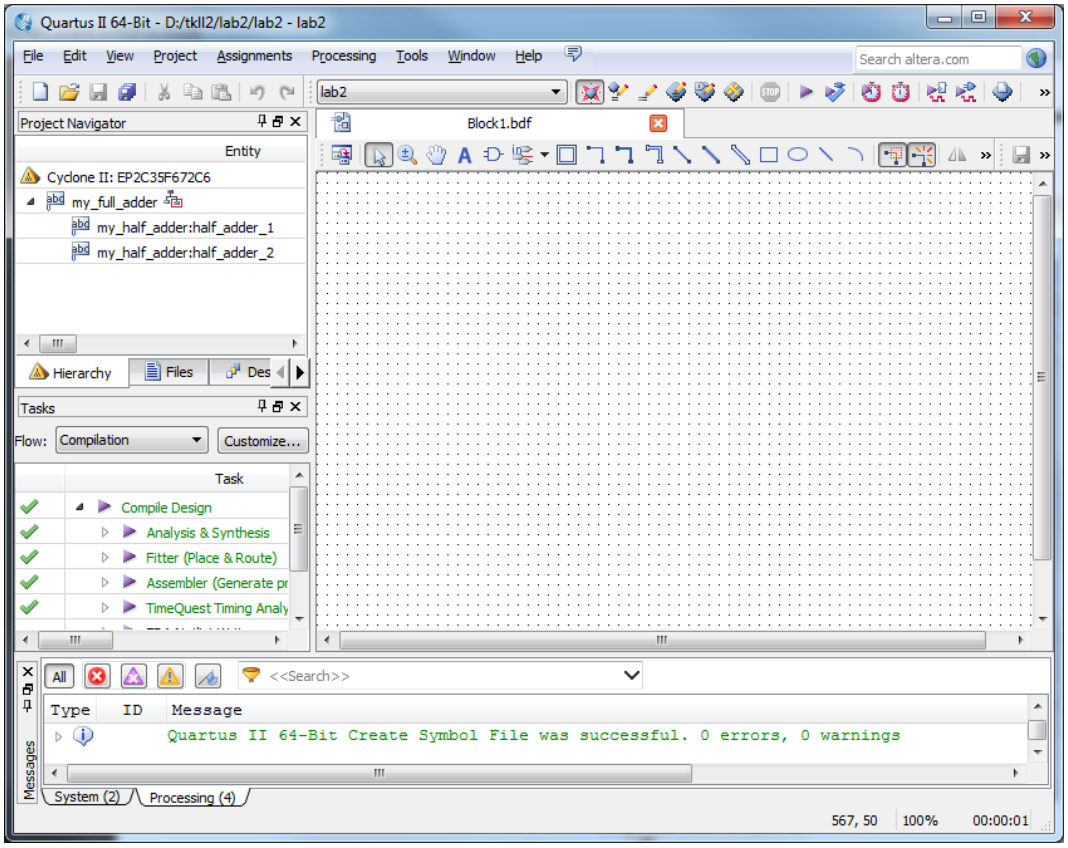


Figure Error! No text of specified style in document.**-6 Schematic Design window.**

Step 11. Some tools for designing in the Schematic window.

- (*Select tool*): select a component in the schematic window.
- (*Text tool*) : create a text component in the schematic window.
- (*Symbol tool*): Logic components to design such as logic gate, or megafunction.
- (pin tool): create intput, output, and bidir pin.
- (*Block tool*): create a function block which makes the design hierarchy.
- (*orthogonal node tool*): connector
- (*orthogonal bus tool*): bus connector
- (*Zoom tool*): zoom in and zoom out.
- (*Full Screen*): choose to work in Full Screen option or not.
- (Find): find a component.

Step 12. Next choose the (Symbol tool)

In the Libraries window, choose **Project ->** mux_2to1 symbol -> OK.

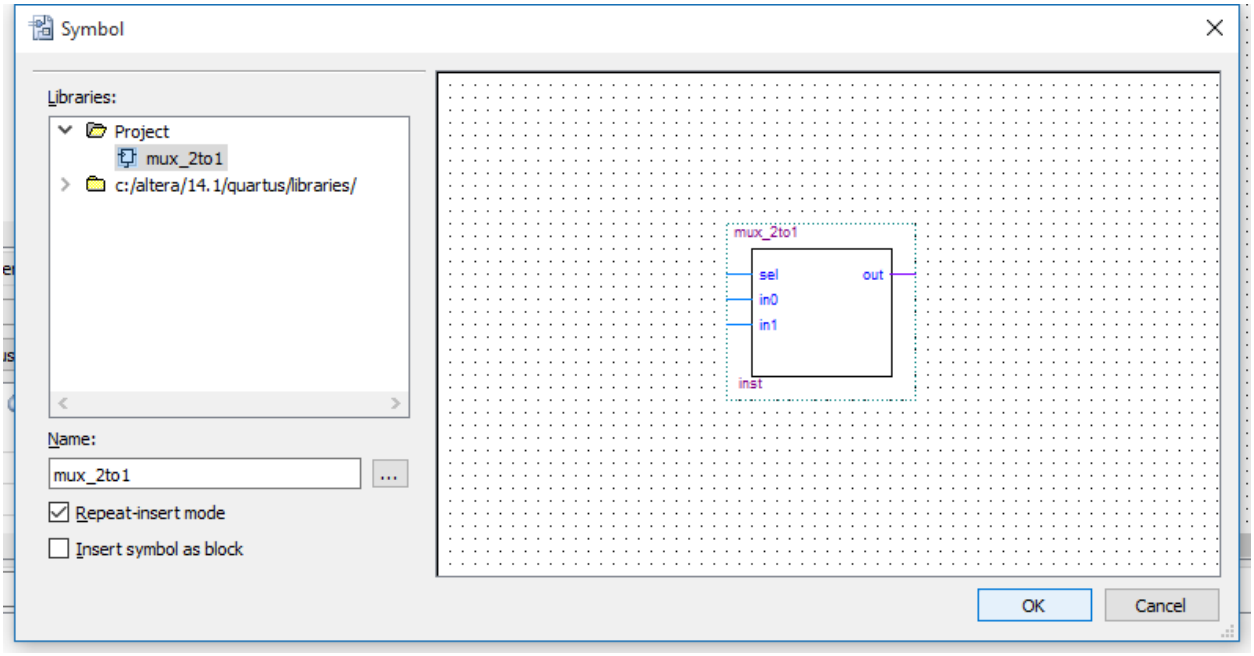


Figure Error! No text of specified style in document.**-7 Symbol window.**

Step 13. Add two more mux_2to1 symbol as below

Step 14. Add input, output pin.

Choose  (pin tool), add pins, and change name by double click on the pin. The result is as below:

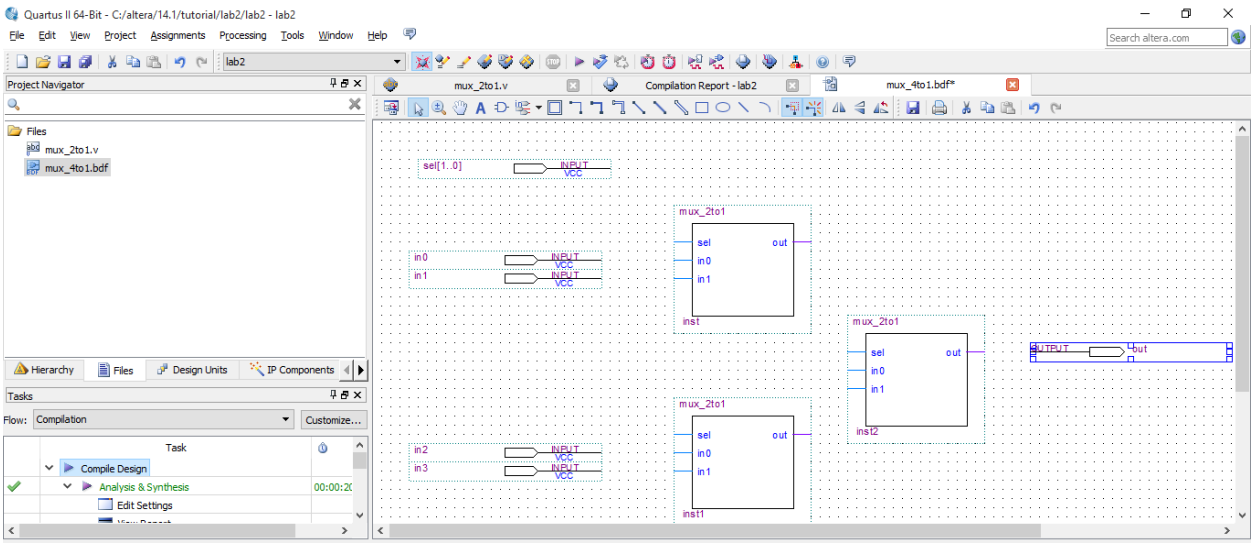


Figure Error! No text of specified style in document.**-8 Add input and output pin for the design.**

Step 15. Connect the pin by click (orthogonal node tool) or the bus by click (*orthogonal bus tool*), then drag the mouse from a source location to destination location.

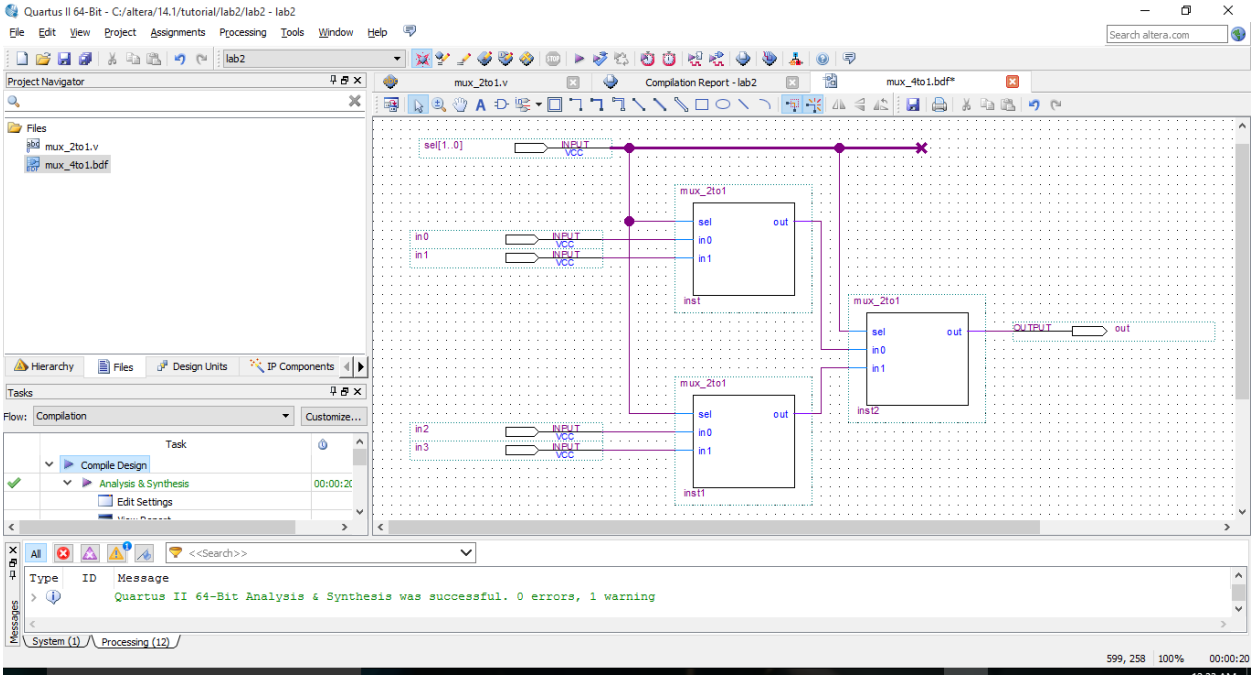


Figure Error! No text of specified style in document.**-9 After connecting pin and bus.**

Step 16. Save schematic design file in .bdf format.

Step 17. Set Top level module

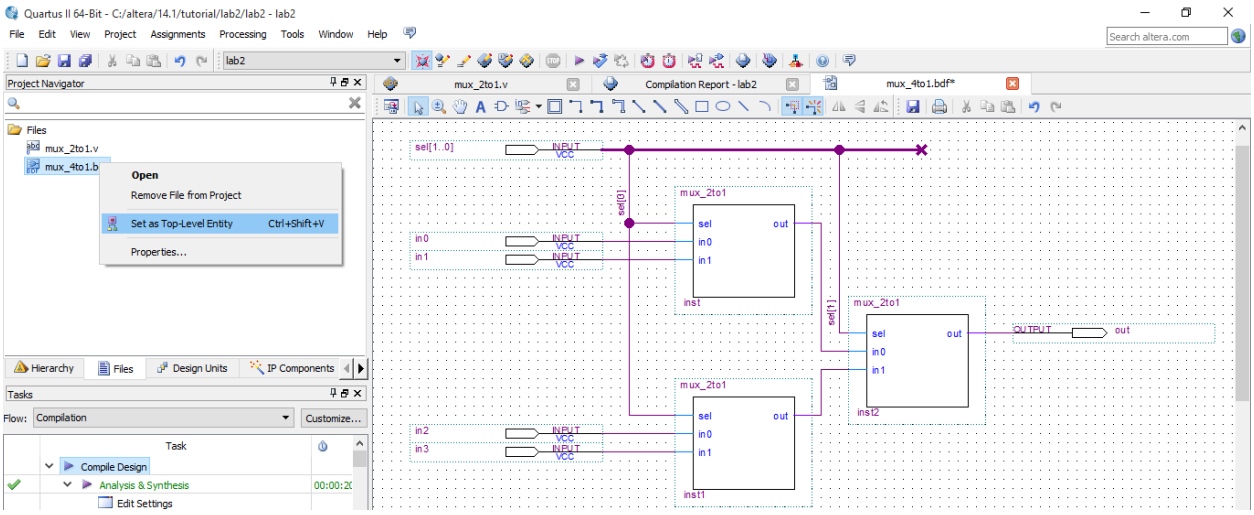


Figure Error! No text of specified style in document.**-10 Set Top level module**

Step 18. The Analysis & Synthesis.

Step 19. Pin Assignment. Note: delete the old pin from the previous design for mux_2to1 module.

Assign pin as the below table:

Signal	Type	Name	Pin
out	output	LEDR[0]	PIN_T23
sel[0]	input	SWITCH[0]	PIN_V28
sel[1]	input	SWITCH[1]	PIN_V21
in0	input	SWITCH[2]	PIN_C2
in1	input	SWITCH[3]	PIN_AB30
in2	input	SWITCH[4]	PIN_U21
in3	input	SWITCH[5]	PIN_U30

Table Error! No text of specified style in document.**-2 Pin table for mux_4to2 module.**

Step 20. Compile file to get the result as follow:

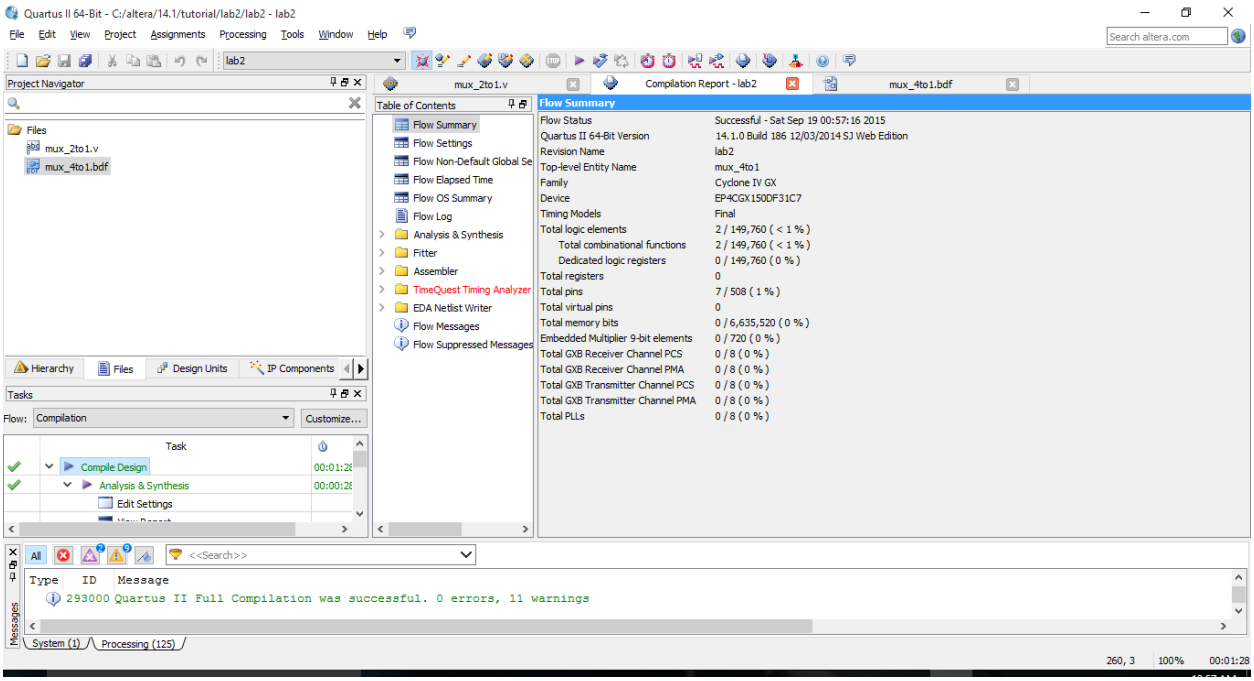


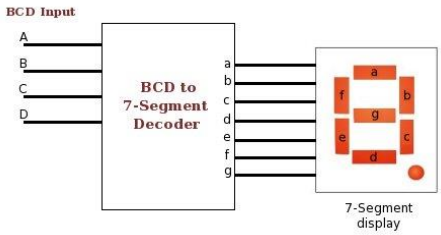
Figure Error! No text of specified style in document.**-11 Compilation result.**

Step 21. Install and test on DE2i-150 board.

II. Exercise:

1. [4 points] Design a BCD Decoder to 7-segment LED with the module name is `BCD_2_Led7seg (led7seg, enable, BCDin)`. Detail:
   - If `enable = 0`, then the system is disabled with 7-segment LEDs turn off.
   - If `enable = 0`, then decode a BCD number from 4-bits to 7-segment LEDs.
   - Note: 7-segment LEDs active HIGH.

Requirment:



a) Code using the RTL method (with assign) (2 points).

b) Create file testbench to test the module (2 points).

c) Take picture of RTL schematic (is a must for the report).

2. [8 points] Design a comparison circuit with 2 input [3:0] A, [3:0] B and 1 output [2:0] result. Detail:

- If A > B then result [0] = 1, otherwise result [0] = 0.
- If A = B then result [1] = 1, otheriwise result [1] = 0.
- If A < B then result [2] = 1, otherwise result [2] = 0.

The module's name is `comparision (result, A, B)`. Input A, B are assigned to 8 SWITCHs, and output result is connected to 3 LEDRs.

a) Code using the RTL method (with assign) (2 points).

b) Create file testbench to test the module (2 points).

c) Take picture of RTL schematic (is a must for the report).

d) Design with Schematic: Display input A to HEX6 (a 7-seg LED), and input B to HEX4, then output signal to LEDR (2 points).

e) Extend: Compare two 5-bits number and output to 7-segment LED using RLT design, input A is out to HEX6 and HEX7, input B is out to HEX4 and HEX5. The result is out to HEX0 with the rule: if A>B then result = 1, A=B then result = 0, A<B then result = 2 (2 points).