

Christopher Yang
Professor Izidor Gertner
February 2015

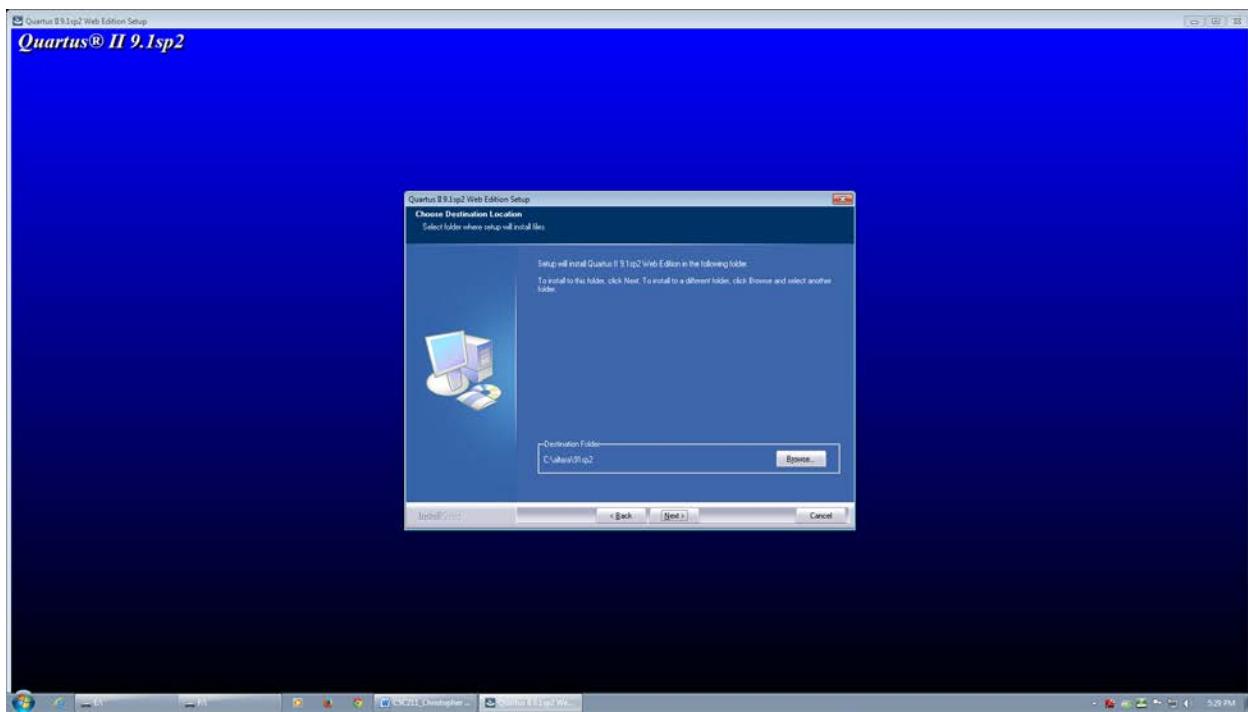
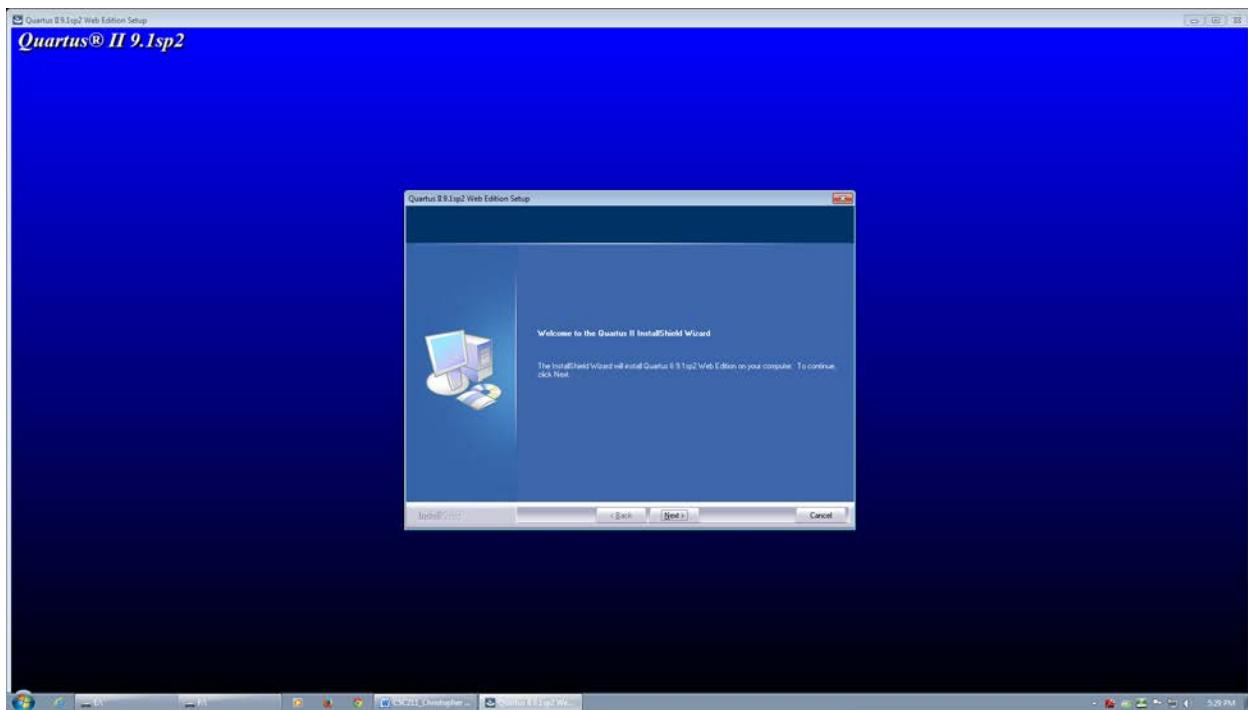
Quartus Tutorial

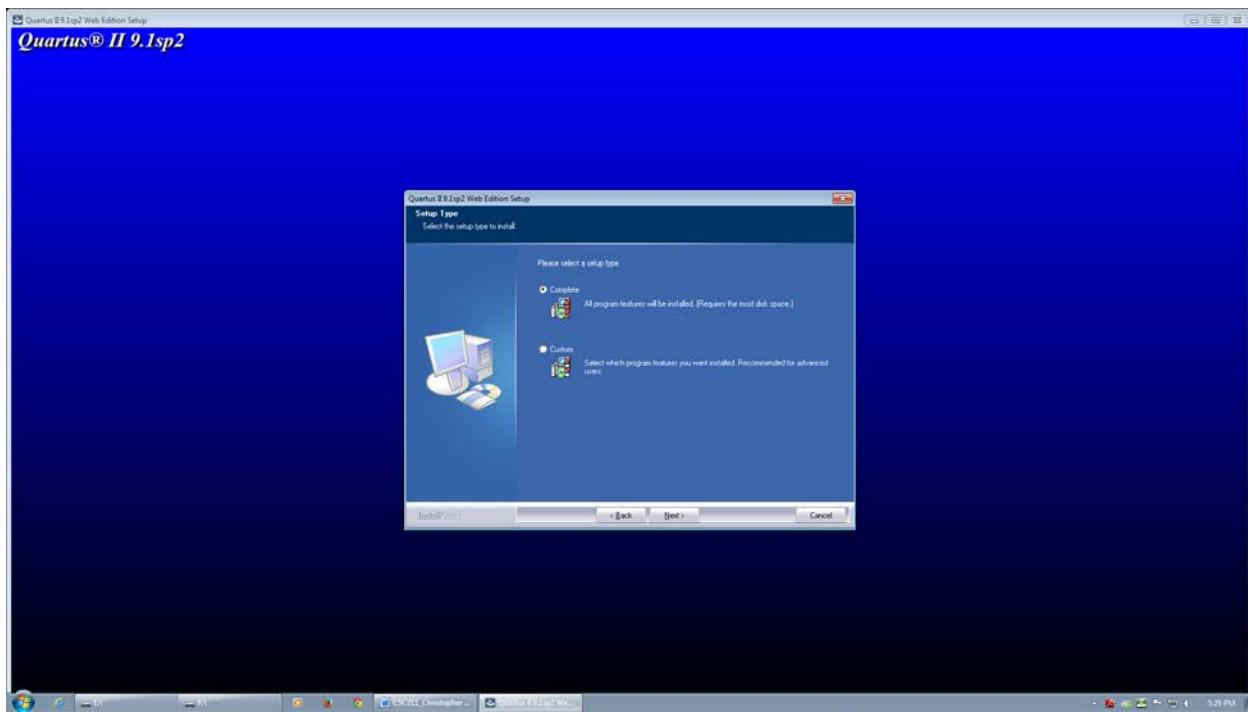
Installing Quartus II Web Edition 9.1sp2

1. Go to <http://www.altera.com/products/software/quartus-ii/web-edition/qts-we-index.html>.
2. Download Software Web Edition – Free.

The screenshot shows the Altera website for Quartus II Web Edition Software. The URL in the address bar is www.altera.com/products/software/quartus-ii/web-edition/qts-we-index.html. The page features the Altera logo and navigation links for Devices, Design Tools & Services, End Markets, Technology, Training, Support, About, and Buy. The main content area is titled "Quartus II Web Edition Software" and includes sections for "Download Software Web Edition – Free", "Compare Subscription Edition and Web Edition", and "No license required for Quartus® II Web Edition software". It lists supported FPGA families and features like ModelSim®-Altera Starter Edition software. The right sidebar contains sections for Next Steps (Download Software, License Software, Request Software DVD, Get Training), Buy Now (Buy Software, Purchase Development Kits, Buy Cables), Support (View Quartus II Help, Get Software Support, View Knowledge Base, Use Troubleshooter, Join the Altera Forum), and Documentation (Get Documentation, Get Handbook (PDF), Get Email Updates, Get Product Catalog (PDF)). The bottom of the page shows standard browser controls and a status bar indicating 10:21 PM, 12/12/2014.

3. Open the file you downloaded, 91sp2_quartus_free.exe
4. Go through the installation process.

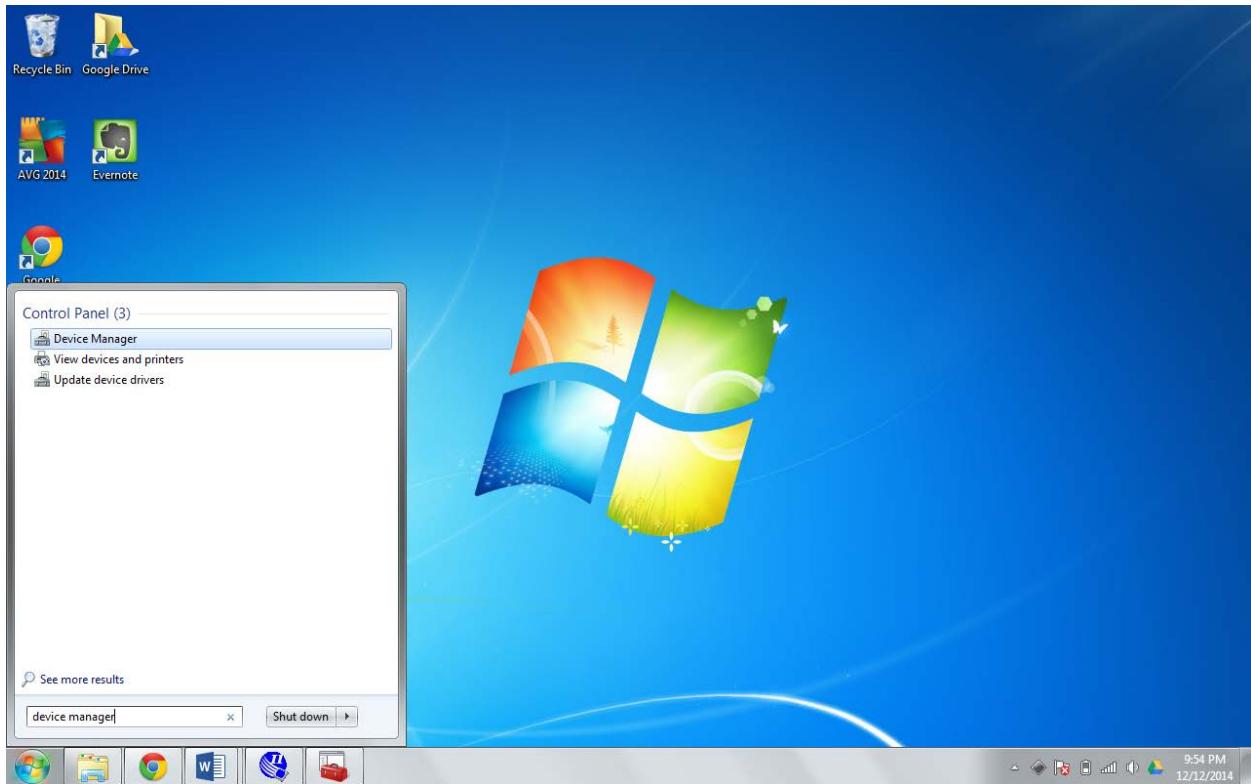




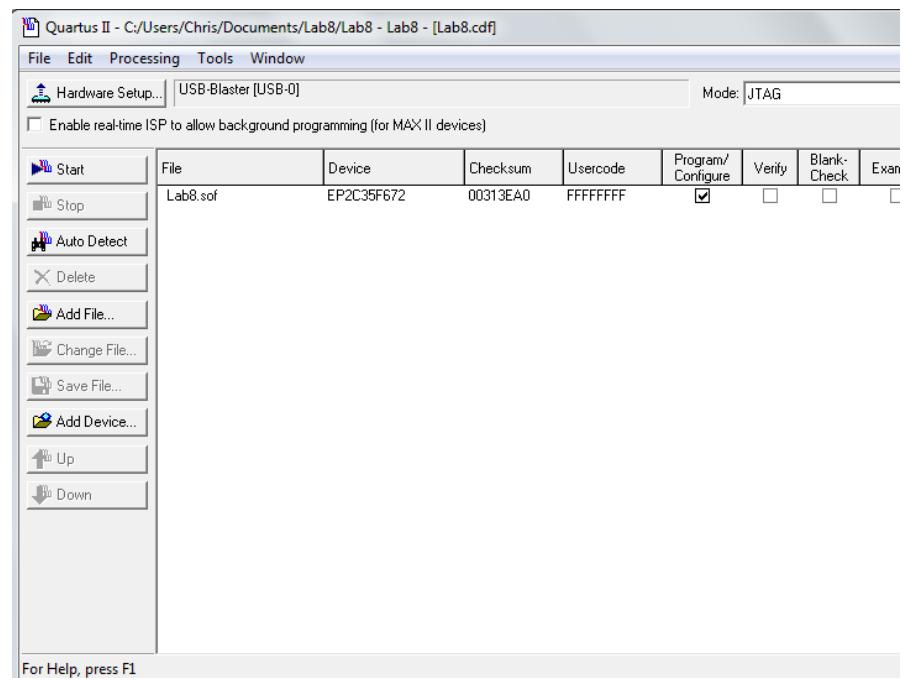
- Once finished, you will have Quartus Web II Edition Software installed on your computer.

Installing the USB driver from the DE2 Board

1. Make sure the USB Blaster Cable is connected to the DE2 Board and your computer.
2. Turn the DE2 Board power on.
3. Go to Device Manager in Windows.

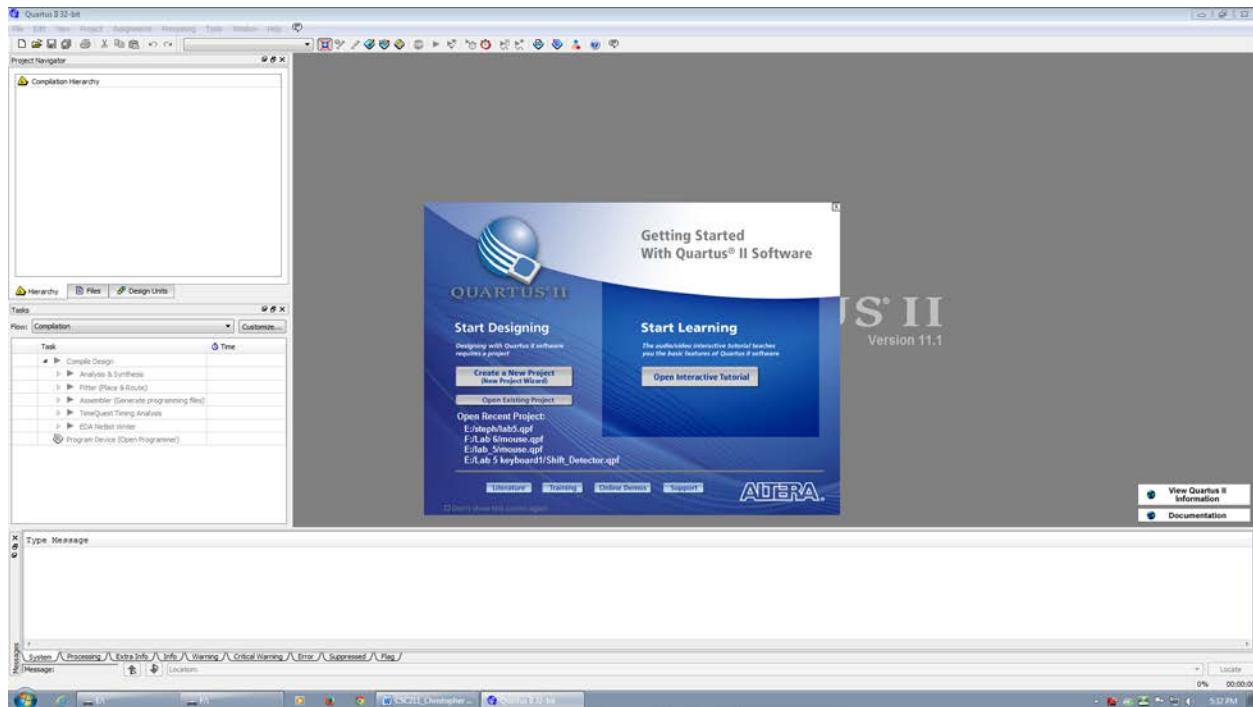


4. Right click the Unknown Device under Other.
5. Select Update Driver Software.
6. Select Browse and look for the file in your computer.
7. Look for the software under Computer(C):\altera\91sp2\quartus\drivers\usb-blaster
8. Press Ok.
9. Go to Quartus and select Tools -> Programmer.
10. At the top make sure it states “USB-Blaster” as shown below:)

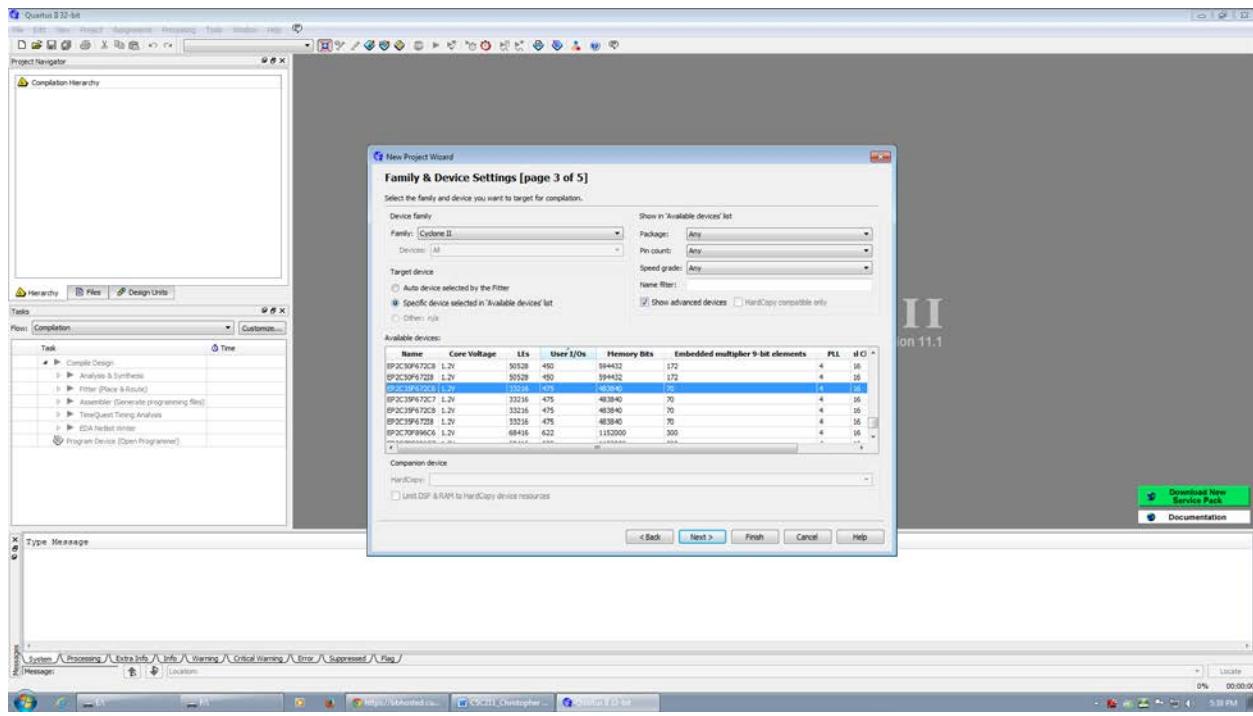


Creating a Project on Quartus

1. Open Quartus and click on Create a New Project.



2. Click Next.
3. Choose a working directory and name your project.
4. Continue pressing Next until you reach the device settings. Configure the setting to match everything in the picture below: Set Family to Cyclone II. Set device to EP2C35F672C6.



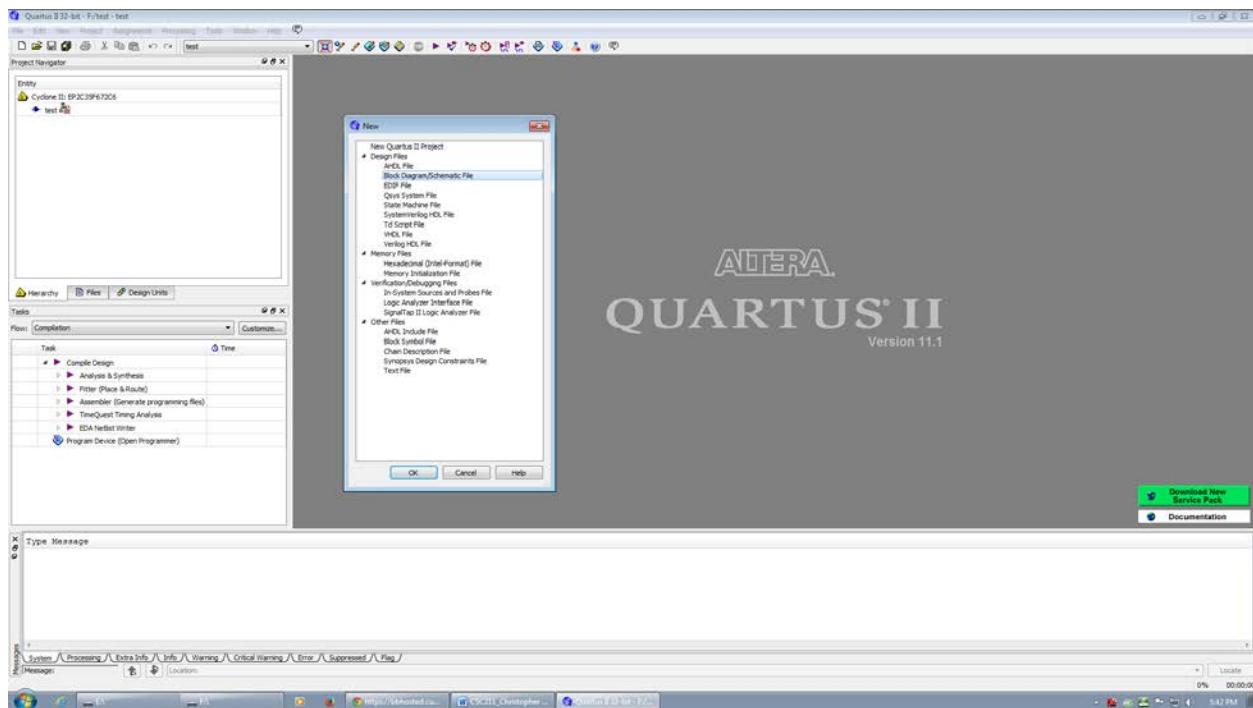
5. Click Finish.

Making a Block Diagram File

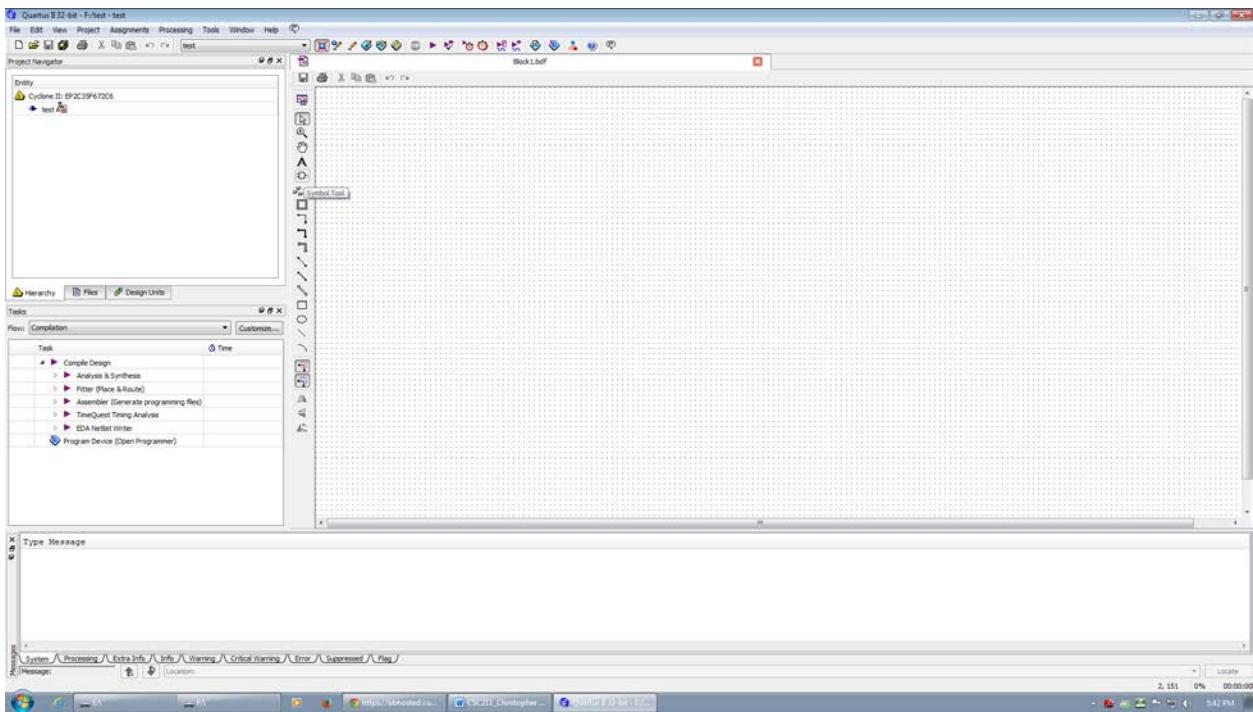
1. Go to File -> New.



2. Select Block Diagram/Schematic File under Design Files.



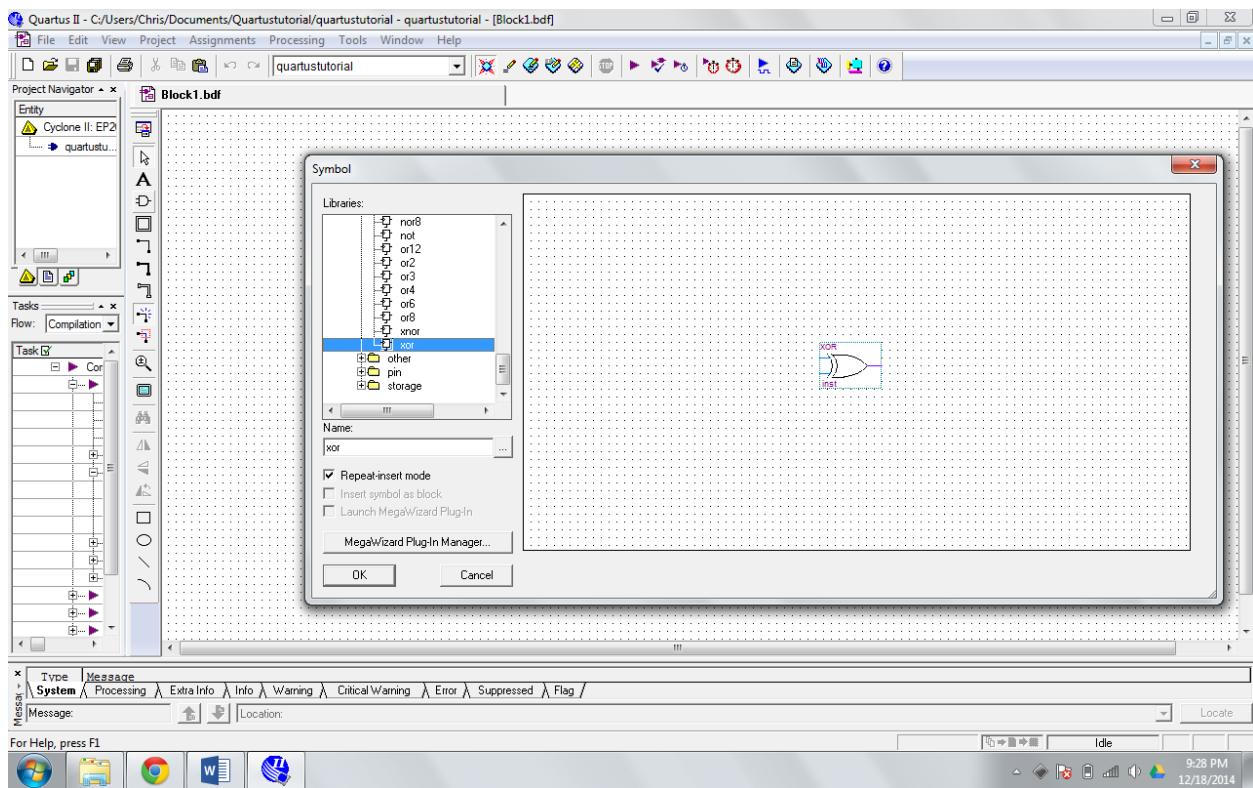
3. Select the symbol tool on the left side.



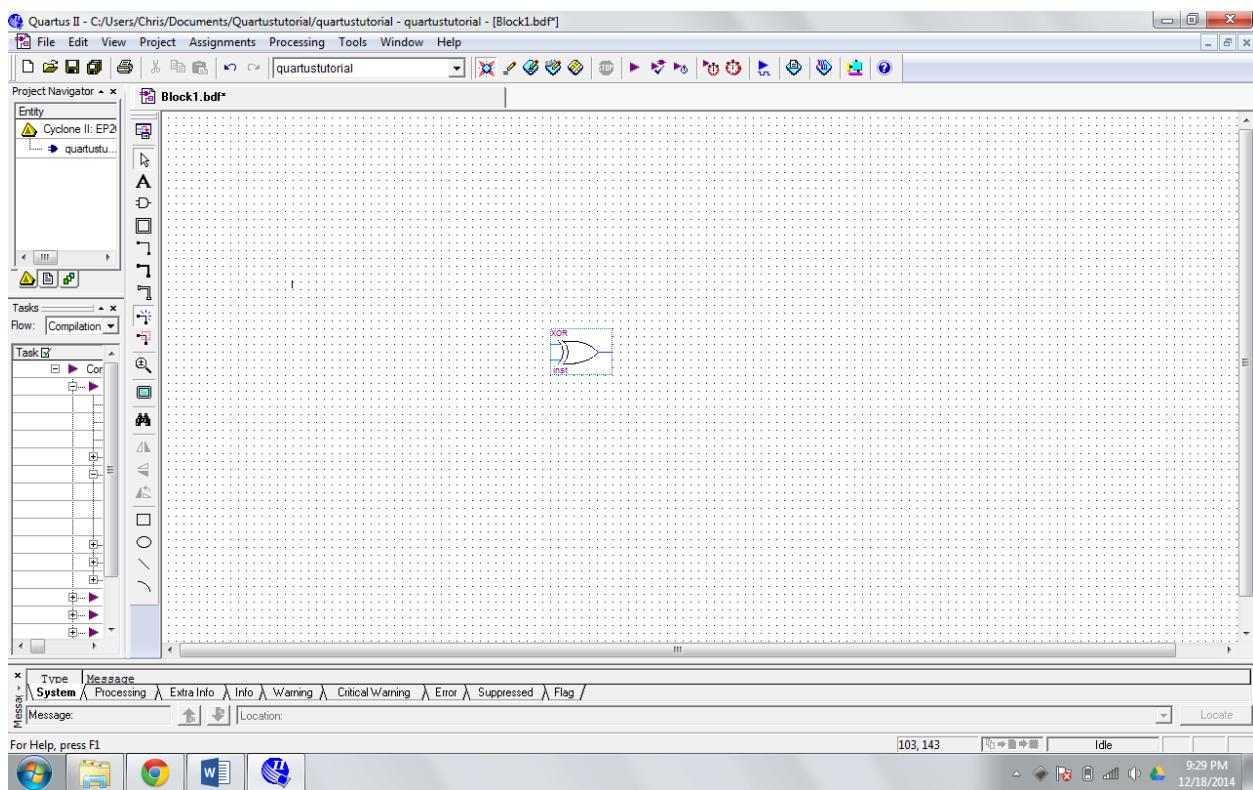
- Under primitives\logic you will find the various gates available. I will be creating a 1-bit adder, consisting of inputs A, B and outputs Sum, Cout (carry-out). These are the Sum of Product functions for the outputs:

$$\begin{aligned} \text{Sum} &= A \oplus B \\ \text{Cout} &= A \cdot B \end{aligned}$$

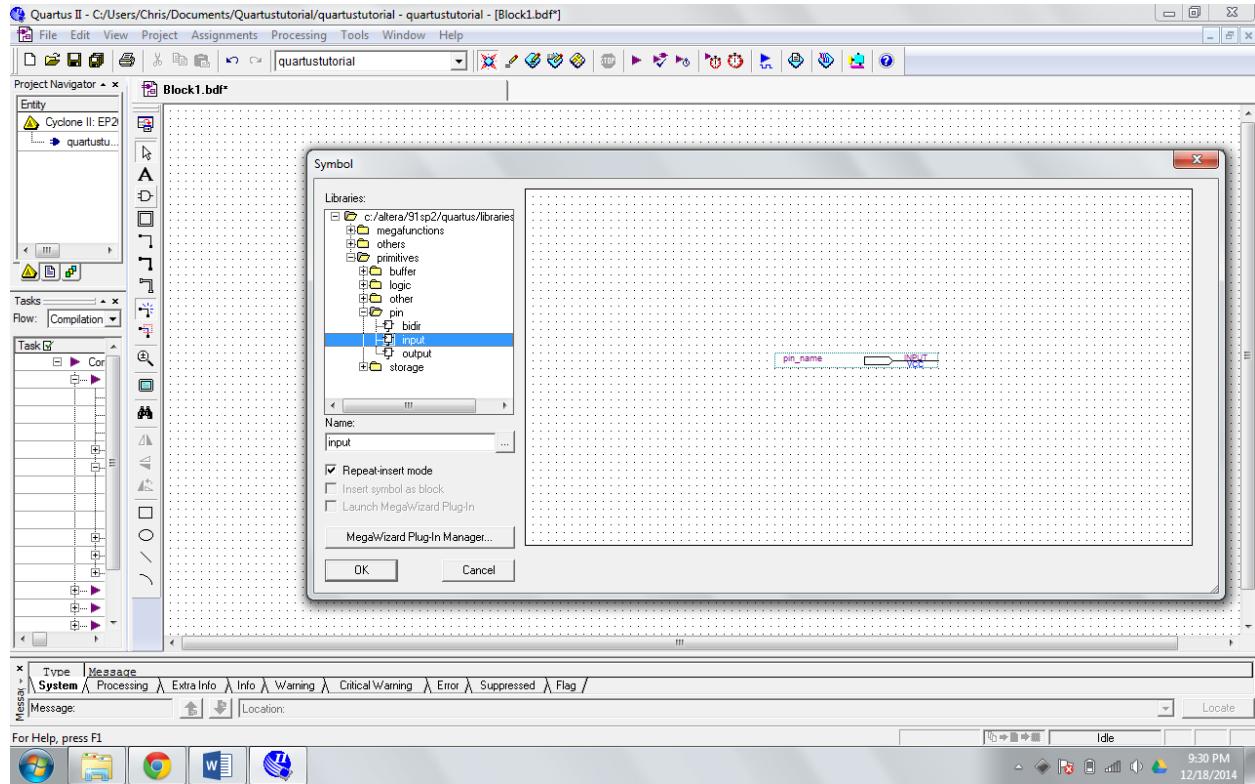
- Let's begin by selecting an XOR gate, labeled "xor", and press OK.



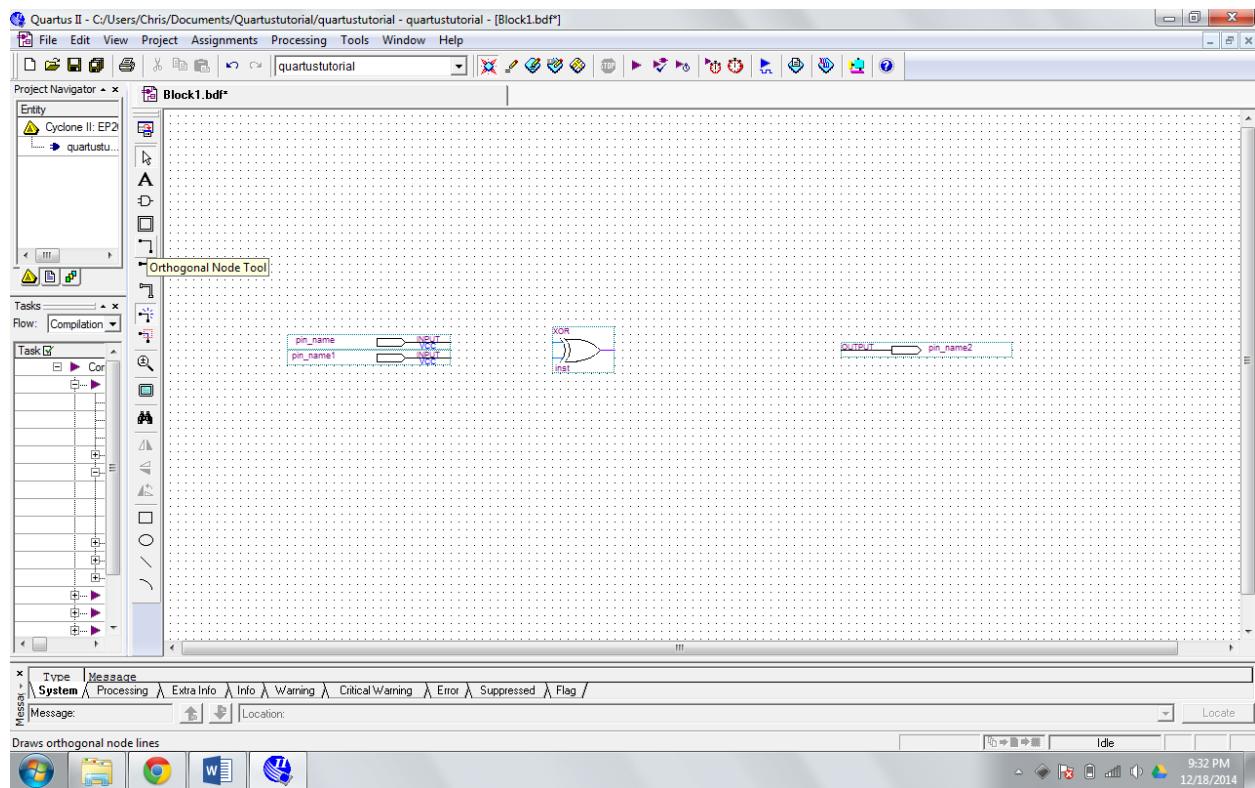
6. Click anywhere on the board to place the gate. To stop placing the gate, click the cursor symbol on the left.



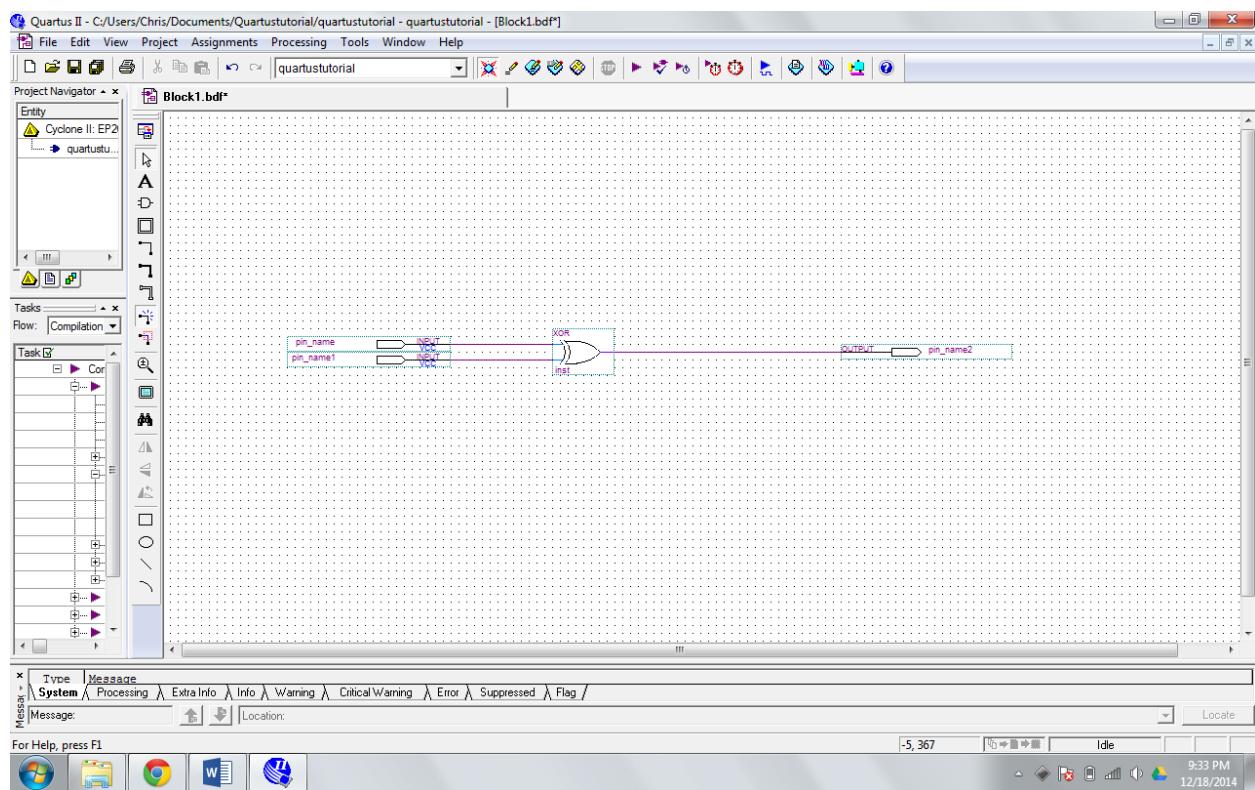
7. Now that we have an XOR gate, we need the inputs and outputs. This can be found in the symbols tools under primitives/pins.



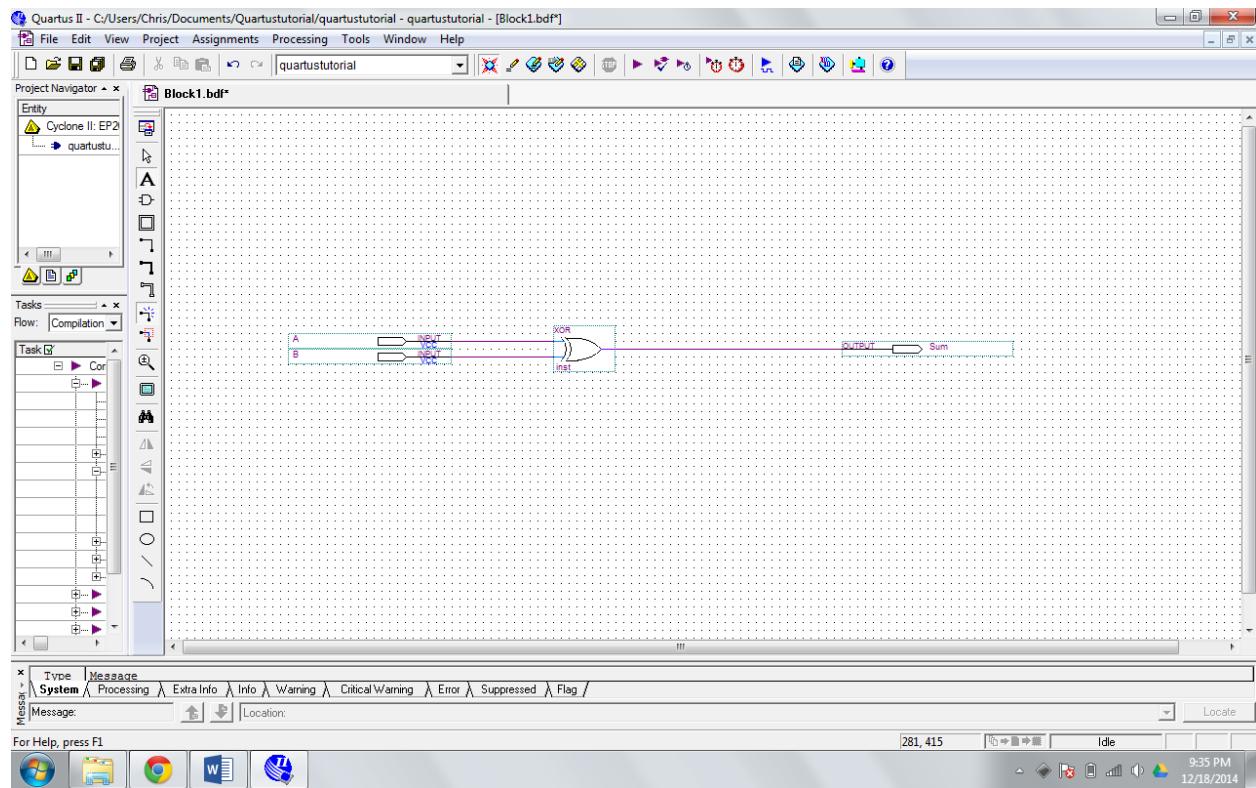
8. Place the inputs and outputs, and connect them using the Orthogonal Node Tool:



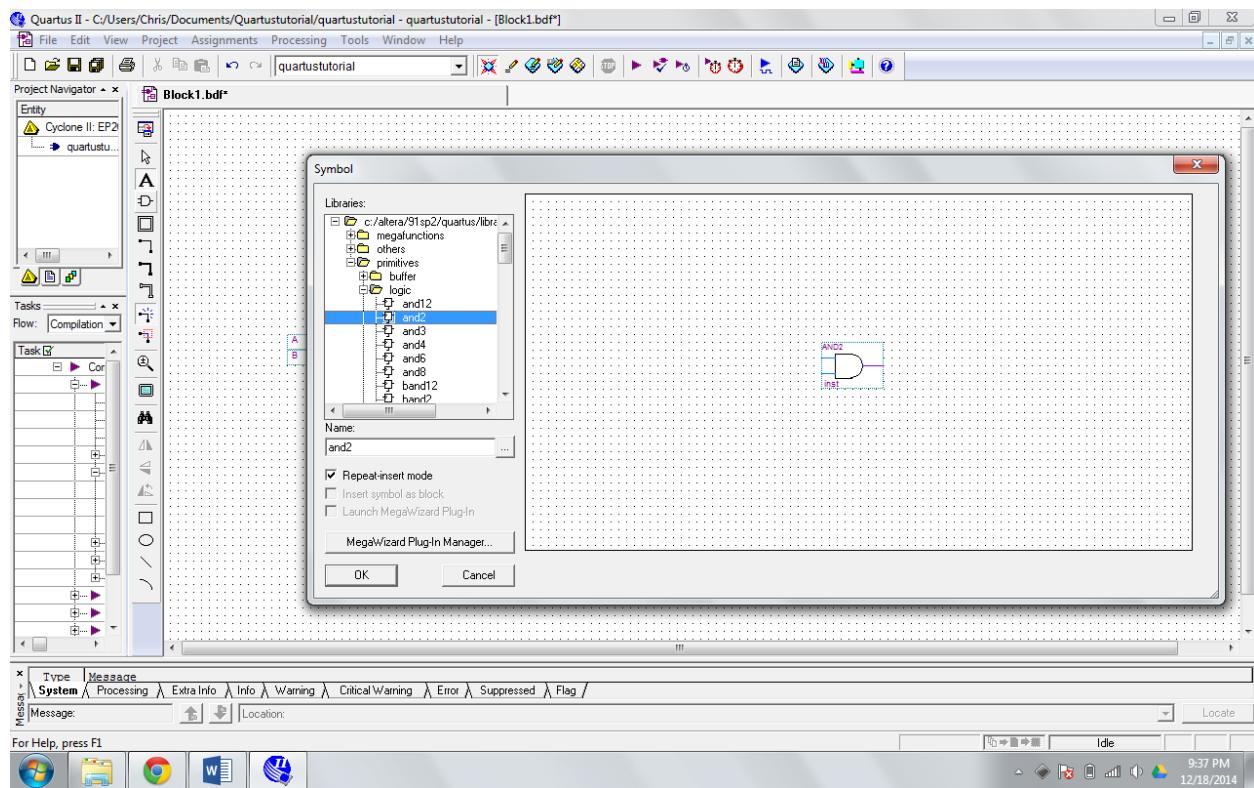
9. Hold the left mouse button at the end of the first input pin and drag the line to the top input of the XOR gate. Do this for the rest of the input and output.



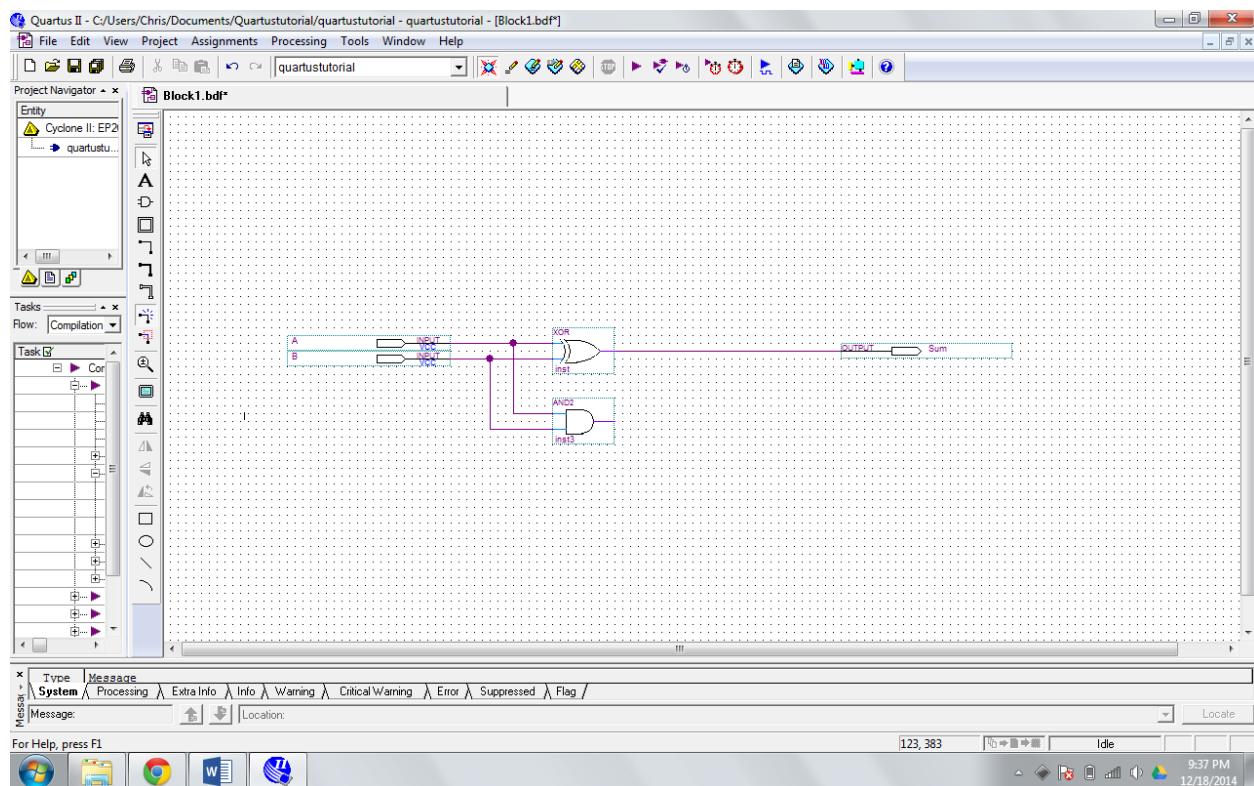
10. Rename the inputs and outputs by clicking on the “A” symbol on the left. Then click on the pins themselves. Rename the inputs to A and B, and the output to F.



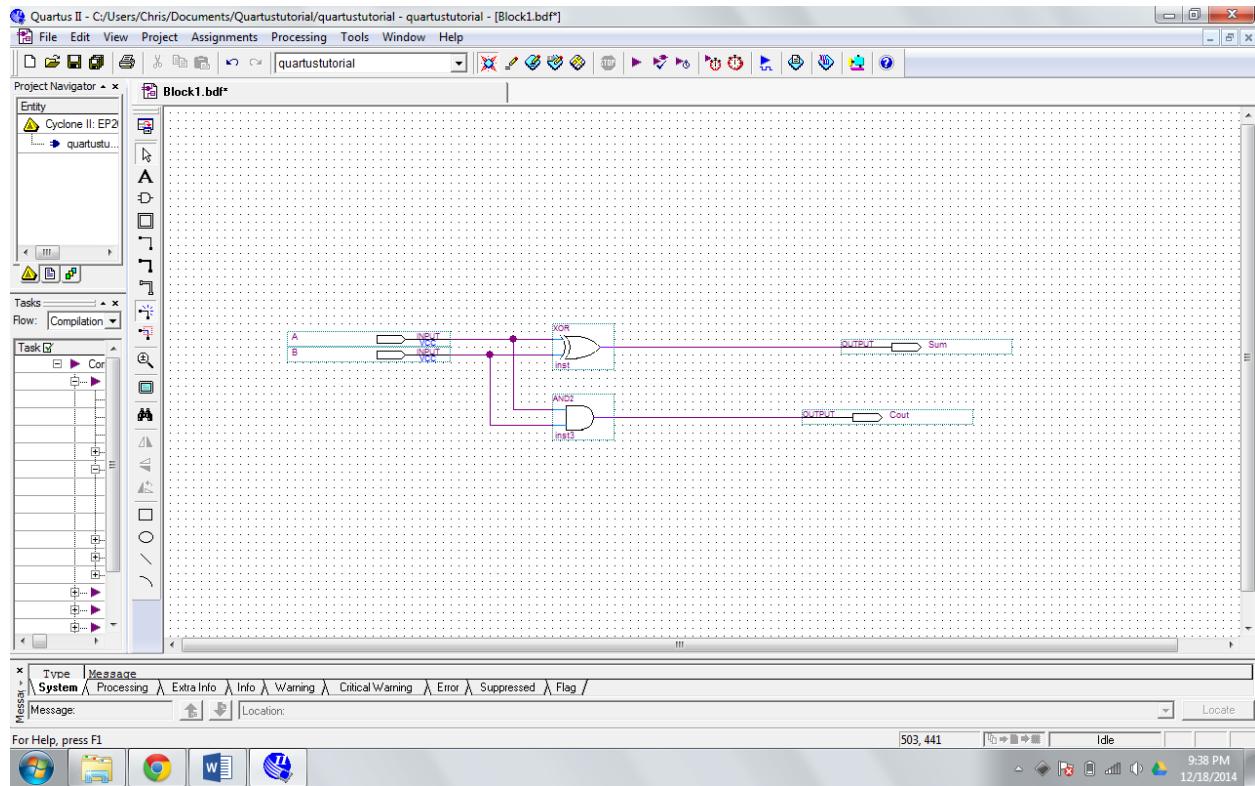
11. Now we need to draw Cout. Start with an AND gate using “and2”



12. Connect A and B to the new AND gate.



13. Create another output for Cout and connect it to the output for the AND gate.

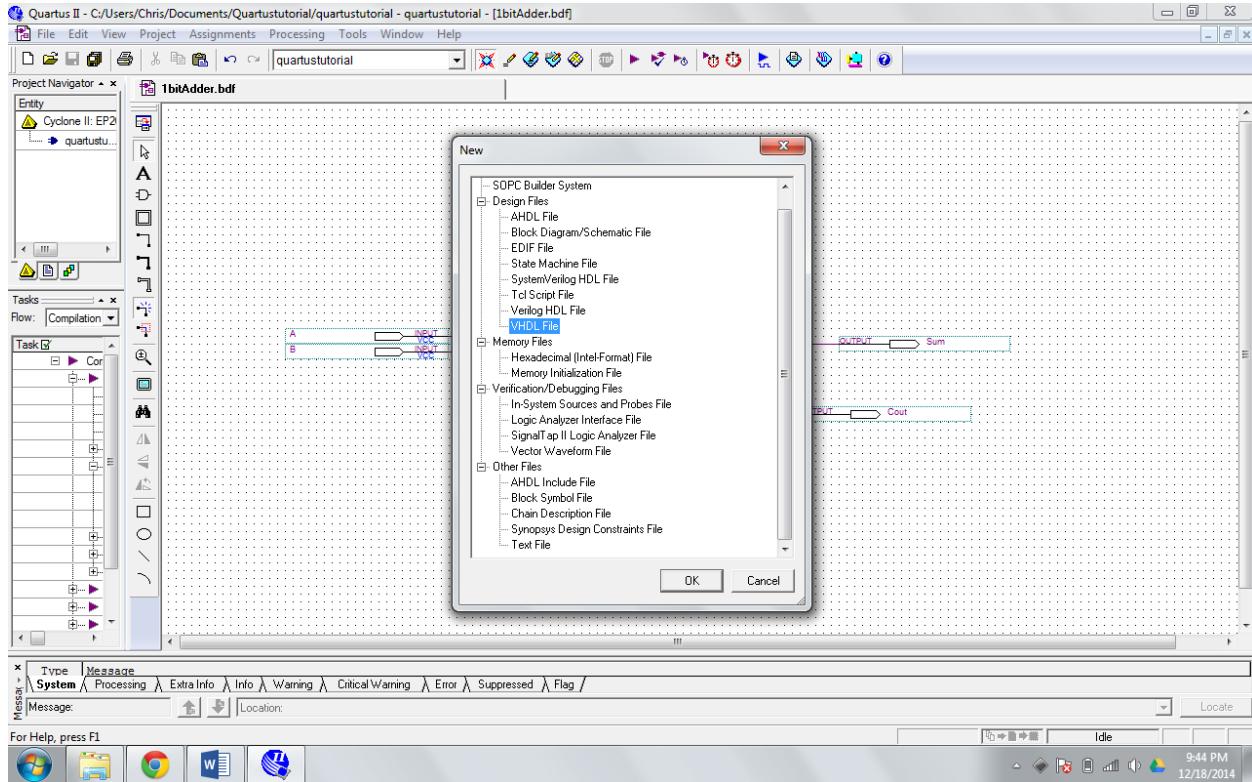


14. Save the file.

15. Now you have a block diagram file (.bdf) for a 1-bit adder.

Creating a VHDL file

1. An alternative to block diagram files is VHSIC Hardware Description Language (VHDL) files.
2. Begin by creating a new VHDL file:



3. Start by including library ieee and using std_logic as specified:

```

library ieee;
use ieee.std_logic_1164.all;

entity Adder is
    port (A, B: in std_logic;
          Sum, Cout: out std_logic);
end Adder;

```

4. Next, create an entity that specifies the name of your circuit and the inputs and outputs. It should look like this:

```

library ieee;
use ieee.std_logic_1164.all;

entity Adder is
    port (A, B: in std_logic;
          Sum, Cout: out std_logic);
end Adder;

```

- Finally, write the architecture that explains how the outputs will be receiving their inputs from their functions.

The screenshot shows the Quartus II software interface. The main window displays the VHDL code for a 1-bit adder:

```

library ieee;
use ieee.std_logic_1164.all;

entity Adder is
    port (A, B: in std_logic;
          Sum, Cout: out std_logic);
end Adder;

architecture Adder_behav of Adder is
begin
    Sum <= (A xor B);
    Cout <= (A and B);
end Adder_behav;

```

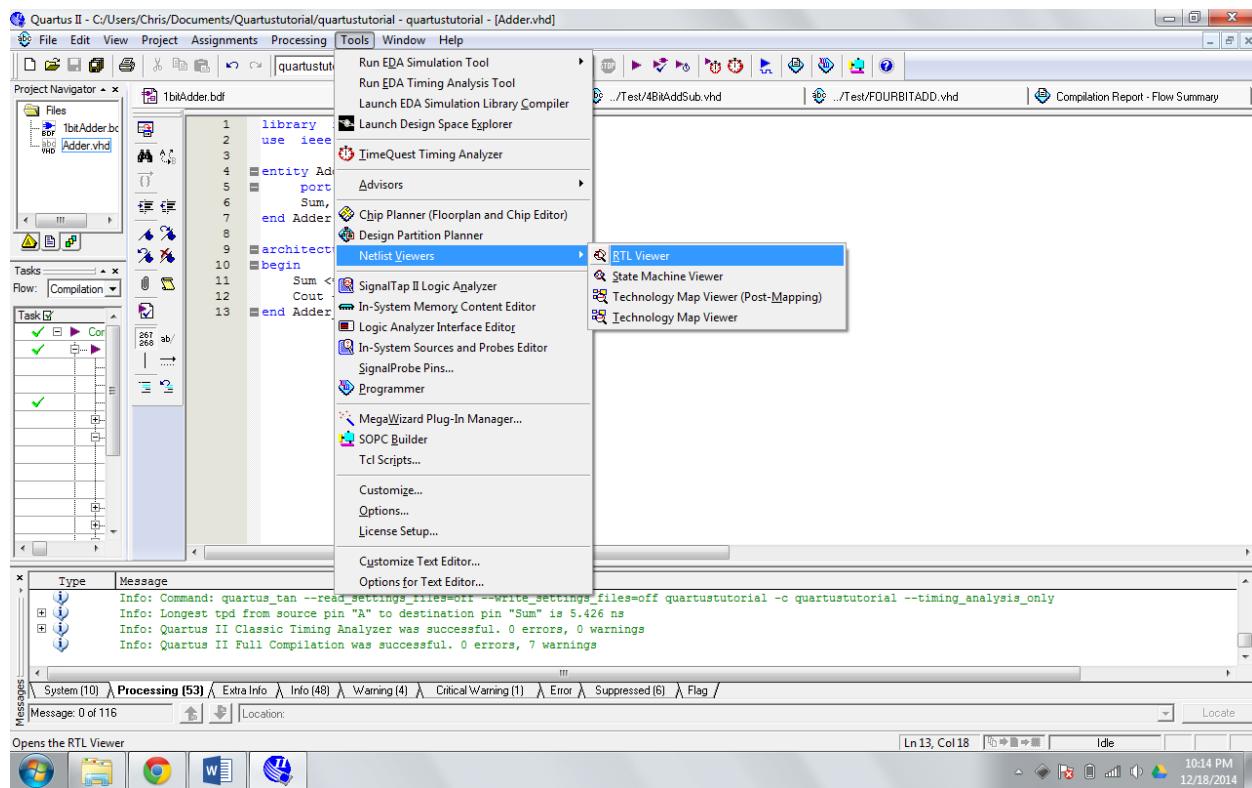
The Project Navigator on the left shows a file named "1bitAdder.bdf". The bottom pane shows a message window with compilation logs:

```

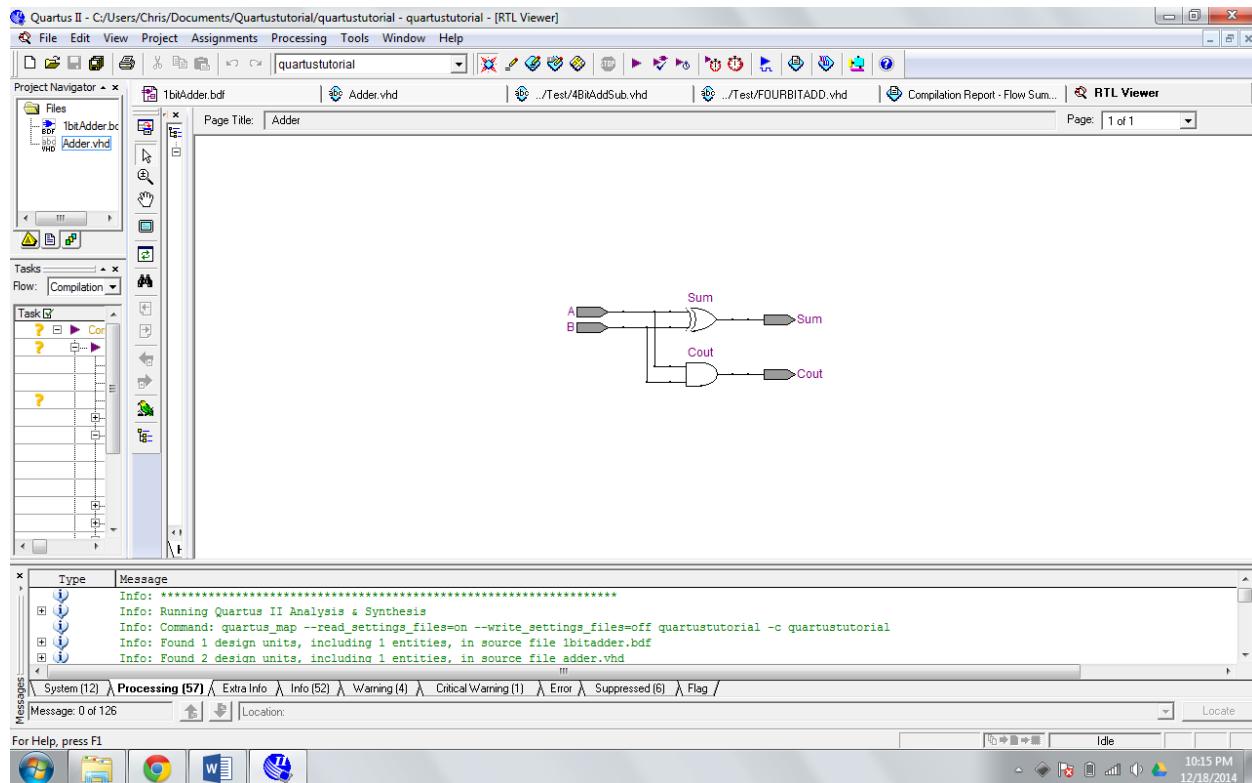
Info: Command: quartus_tan --read_settings_files=off --write_settings_files=off quartustutorial -c quartustutorial --timing_analysis_only
Info: Longest tpd from source pin "A" to destination pin "Sum" is 5.426 ns
Info: Quartus II Classic Timing Analyzer was successful. 0 errors, 0 warnings
Info: Quartus II Full Compilation was successful. 0 errors, 7 warnings

```

- Save the file. Make sure the name you save it as is the same as the entity.
- After compiling, you can check the circuit by going to Tools -> Netlist Viewers -> RTL Viewer

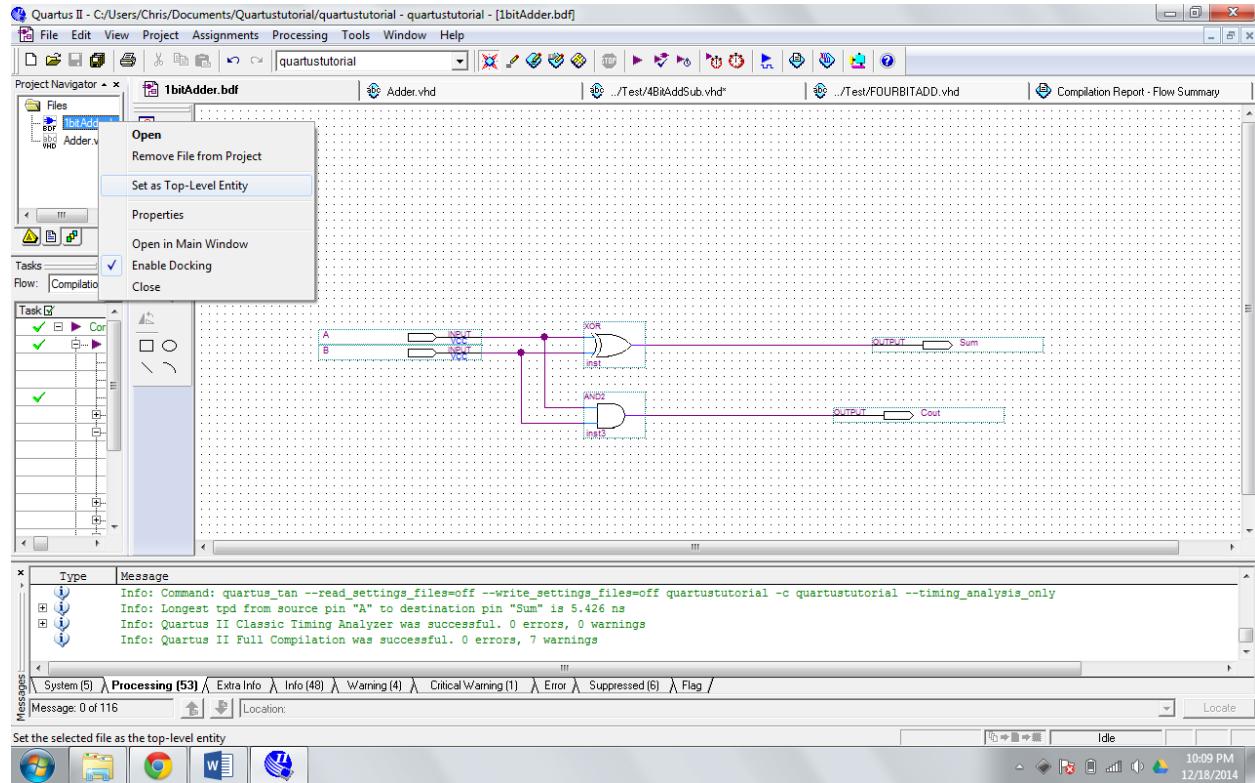


8. Compare the circuit with the block diagram file and verify they are the same.

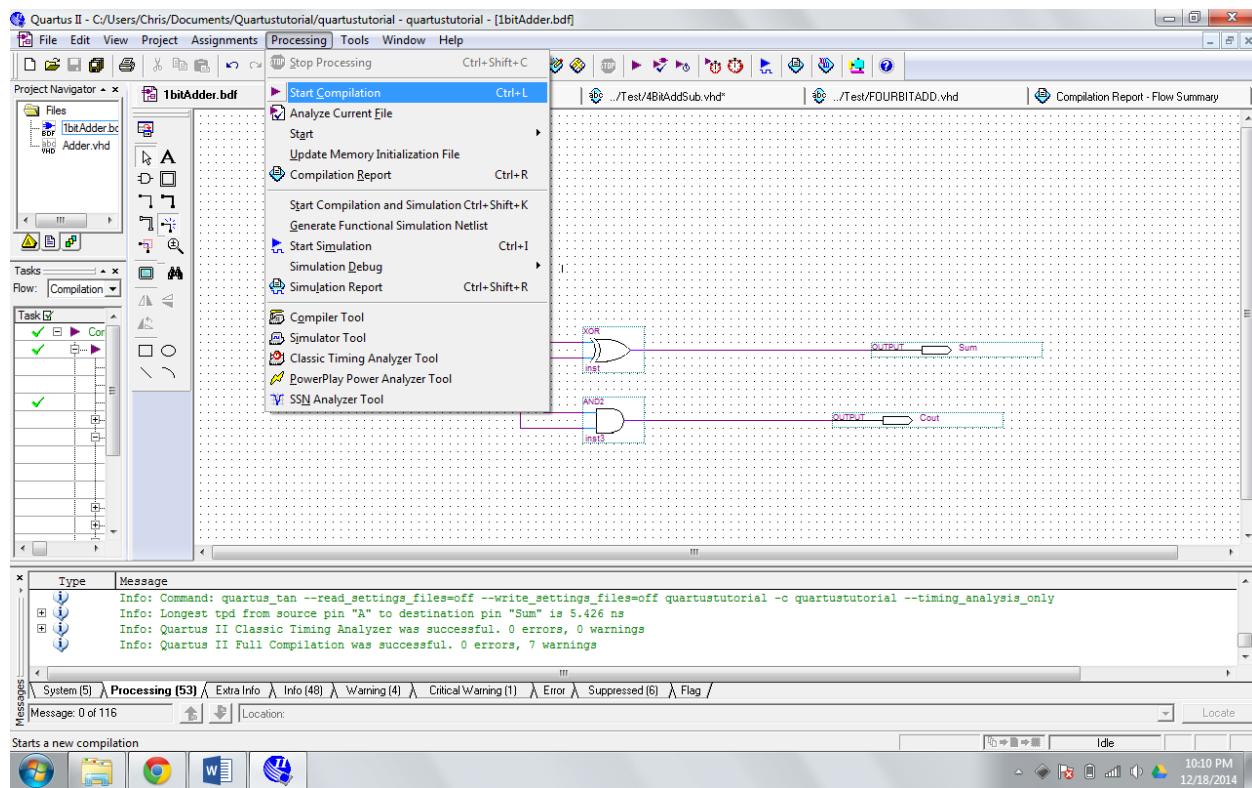


Compiling your file

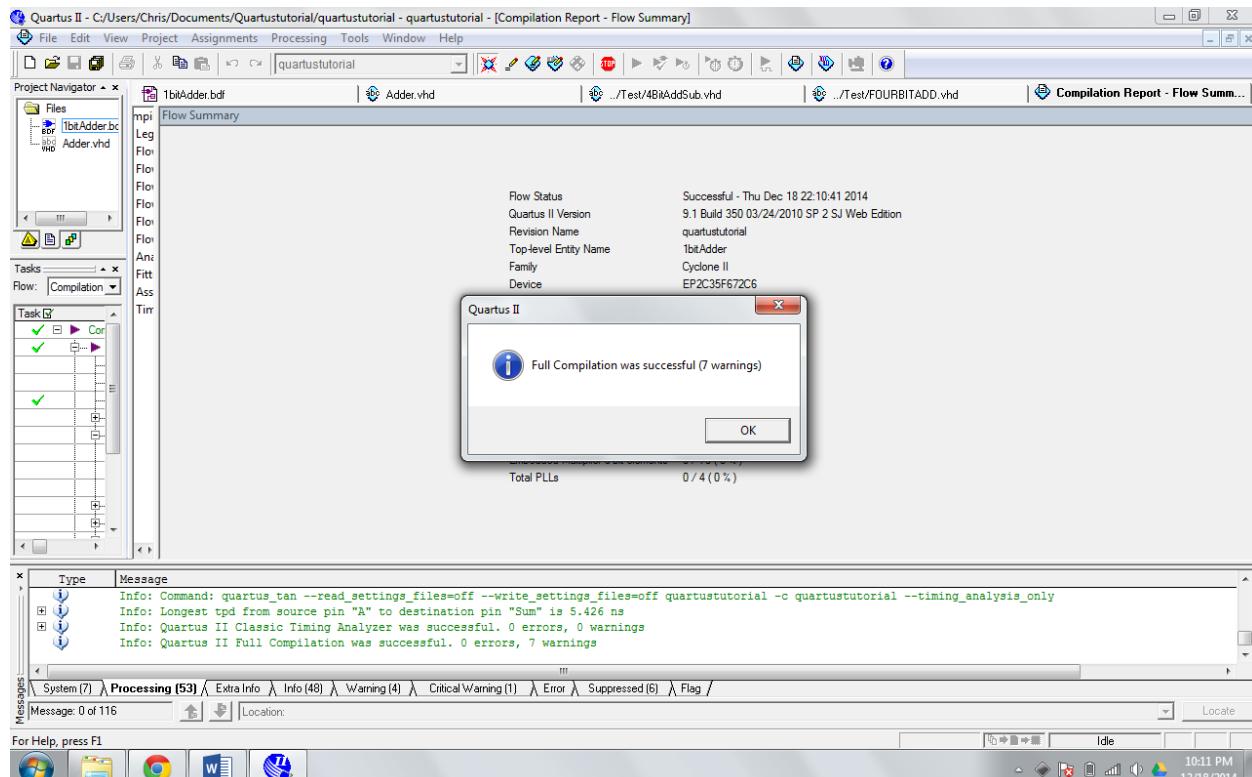
1. Before validating that your file has been built and works correctly, it needs to be compiled.
2. Select the file that you want to compile by right clicking on it in the Files tab of Project Navigator.
3. Select Set as Top-Level Entity.



4. Go to Processing -> Start Compilation.



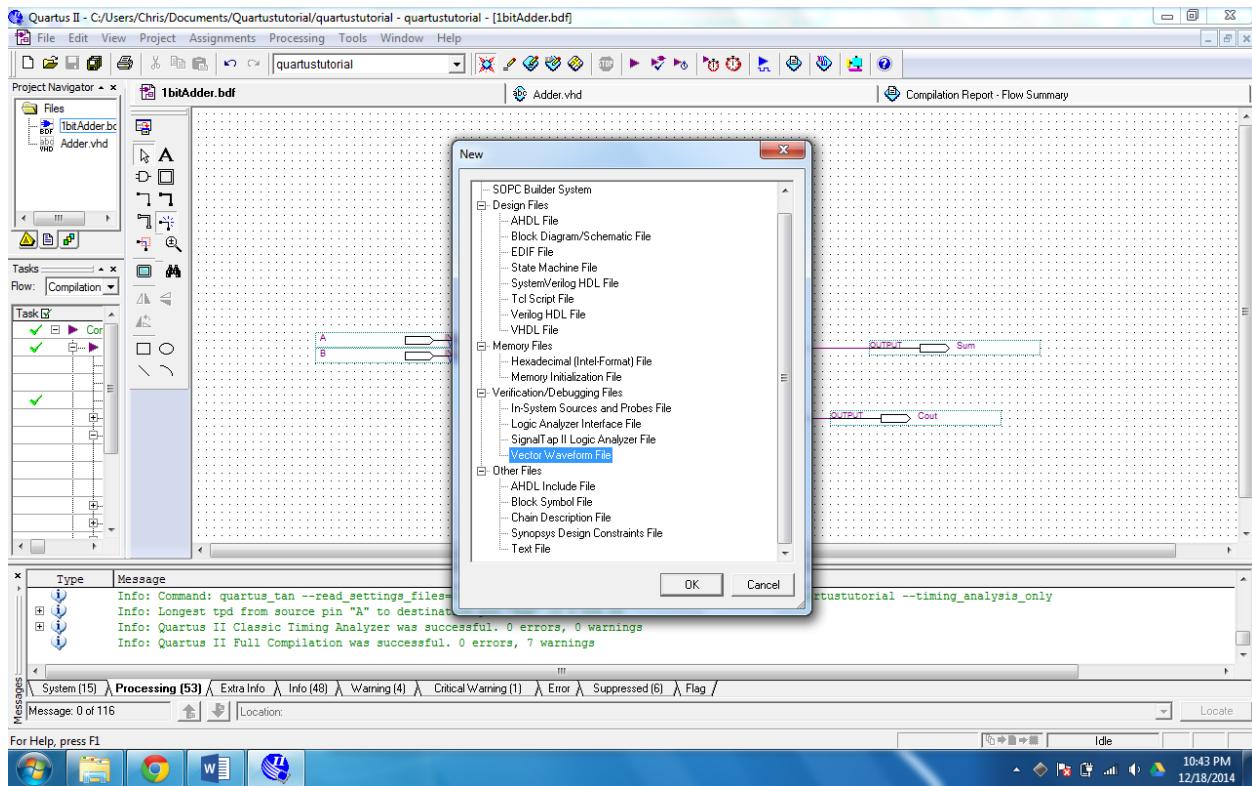
5. Once it is finished, a pop-up will say “Full Compilation was successful.”



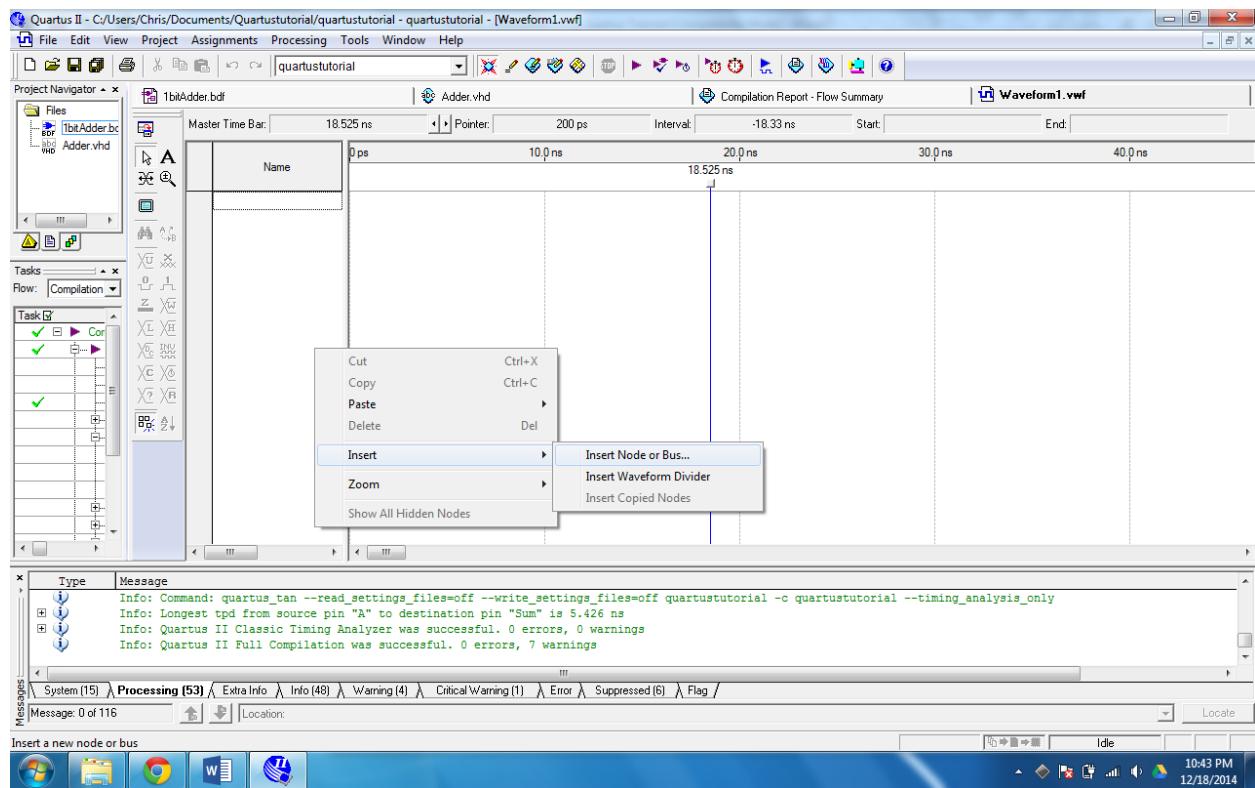
6. Press Ok and your file is now compiled. You can also repeat this step for the VHDL file.

Simulating through vector waveforms

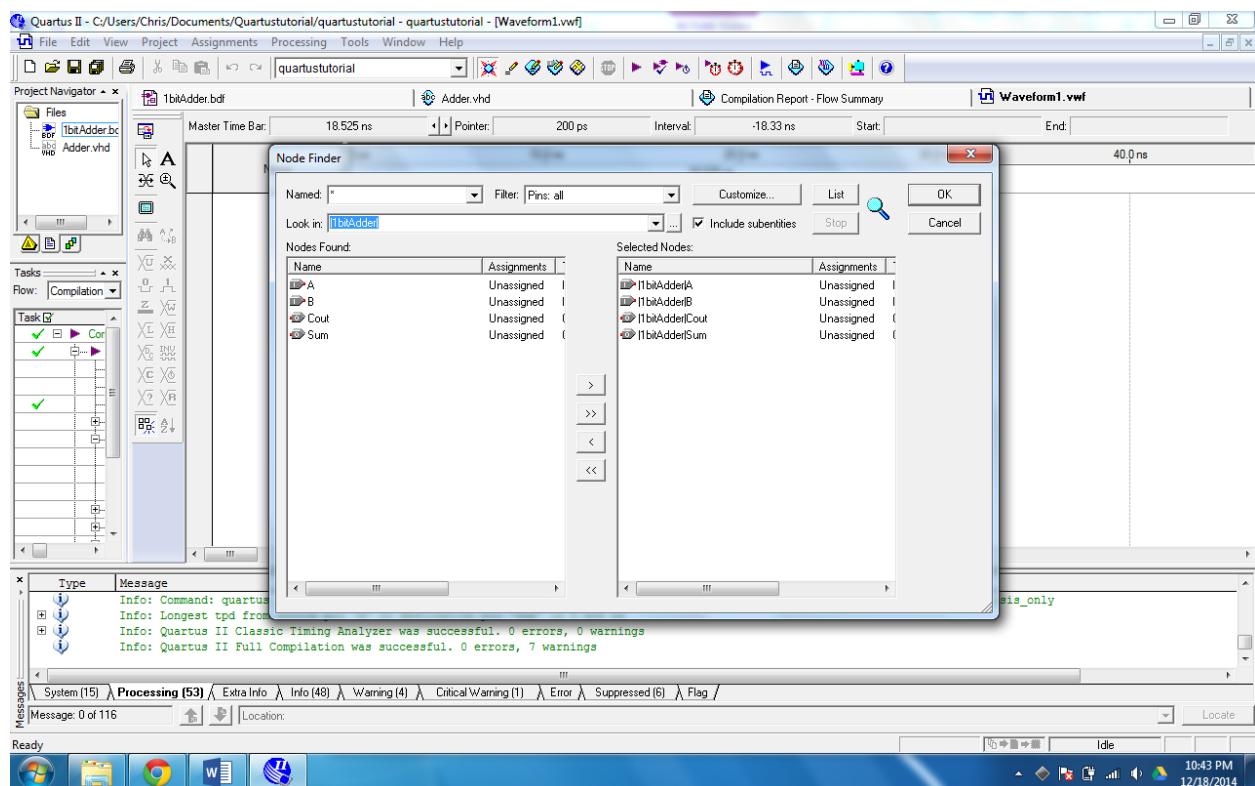
1. Go to File -> New -> Vector Waveform File (under Verification/Debugging Files).



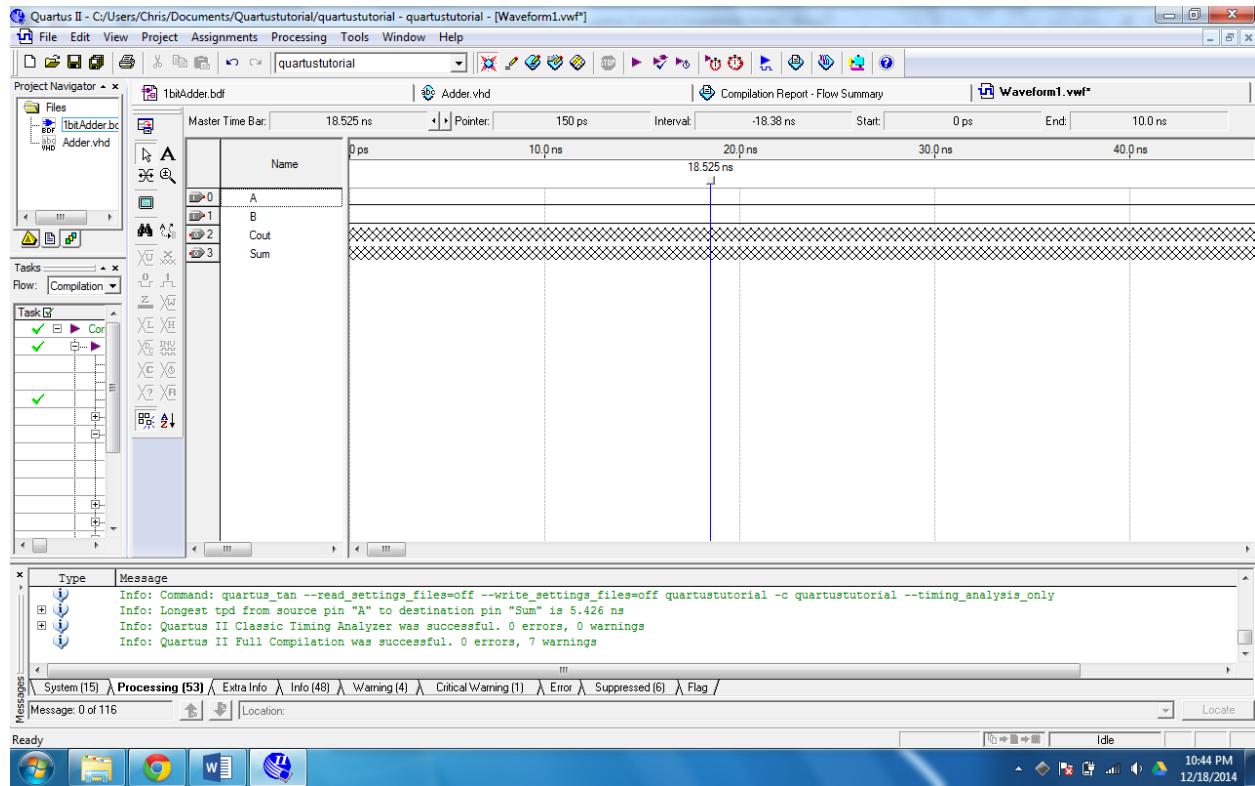
2. Under Name, right-click and select Insert -> Insert Node of Bus.



3. Click on Node Finder.
4. Select List and click the button that has “>>” as a symbol.



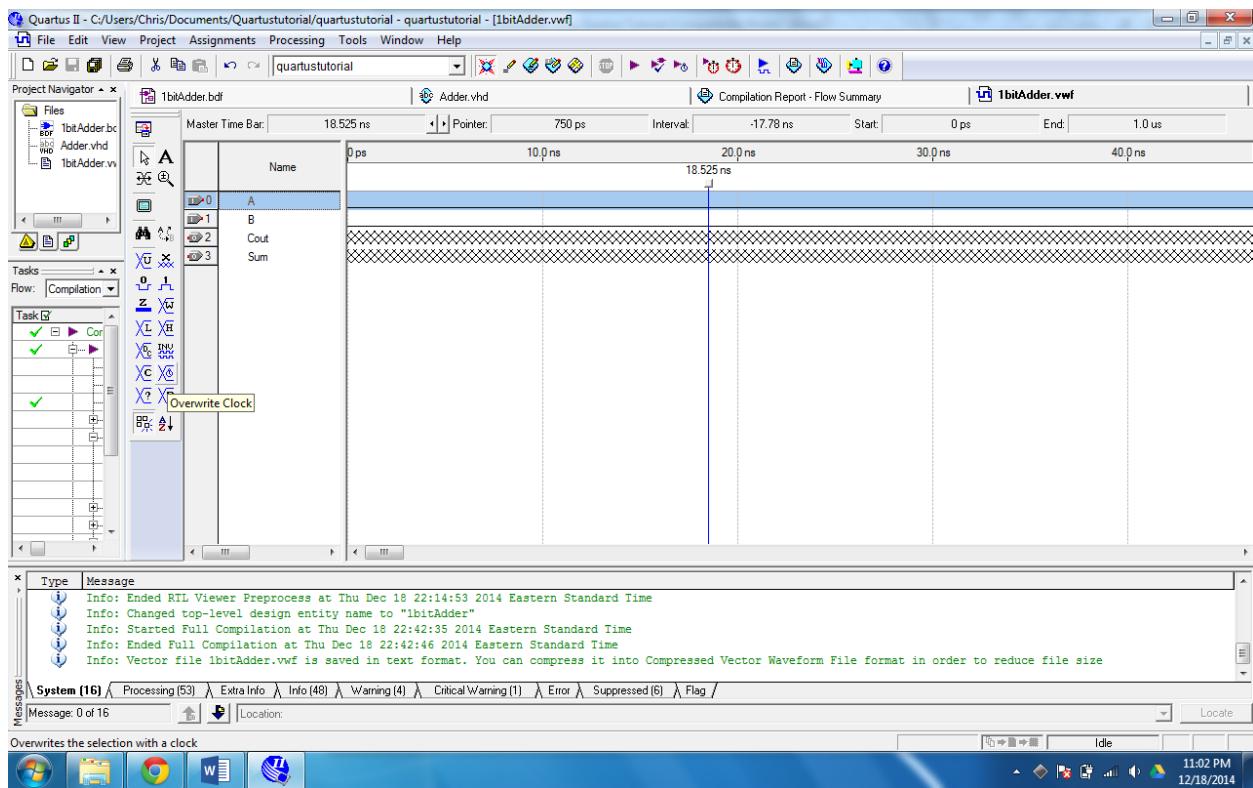
- Click Ok and Ok again. This is what you should have:



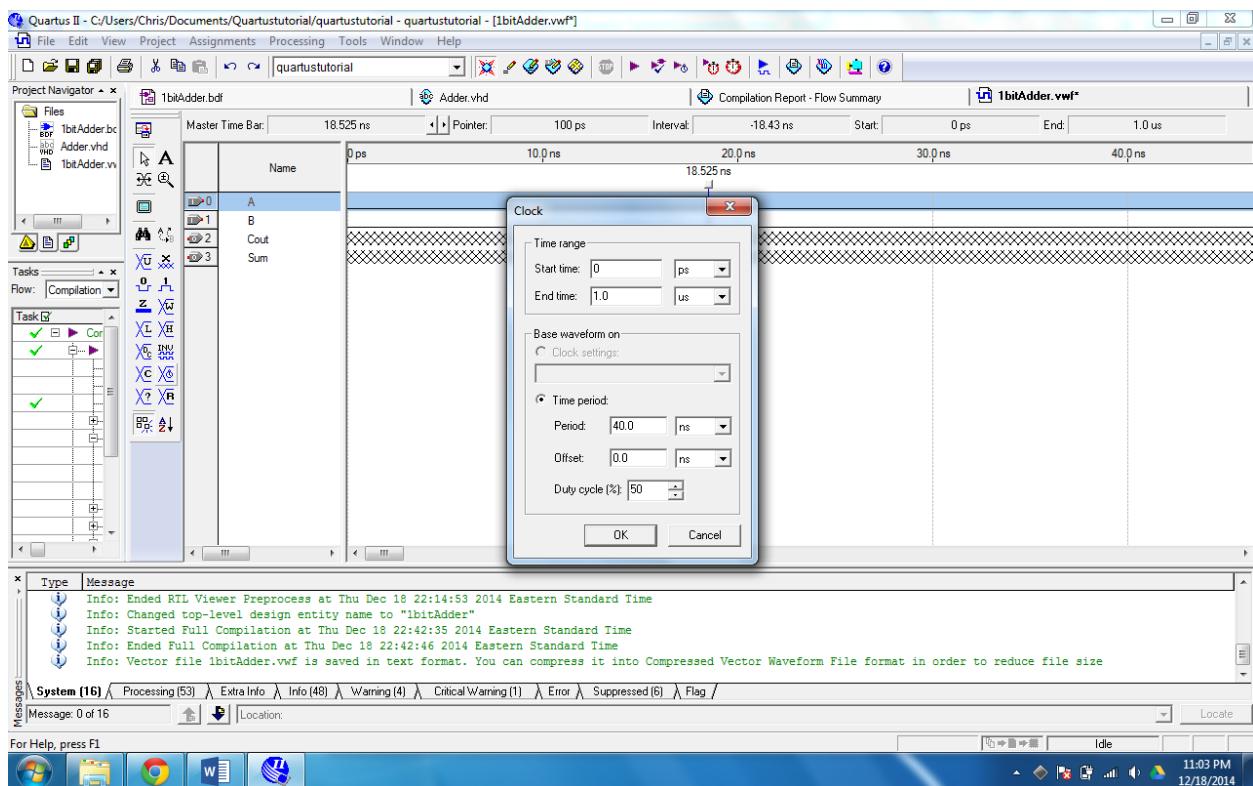
- Select the input that you want to create a waveform with by selecting it under Name.
- Let's test all the possible combination of values of A and B with the corresponding solutions. Here is the truth table that we will simulate and verify:

A	B	Sum	Cout
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

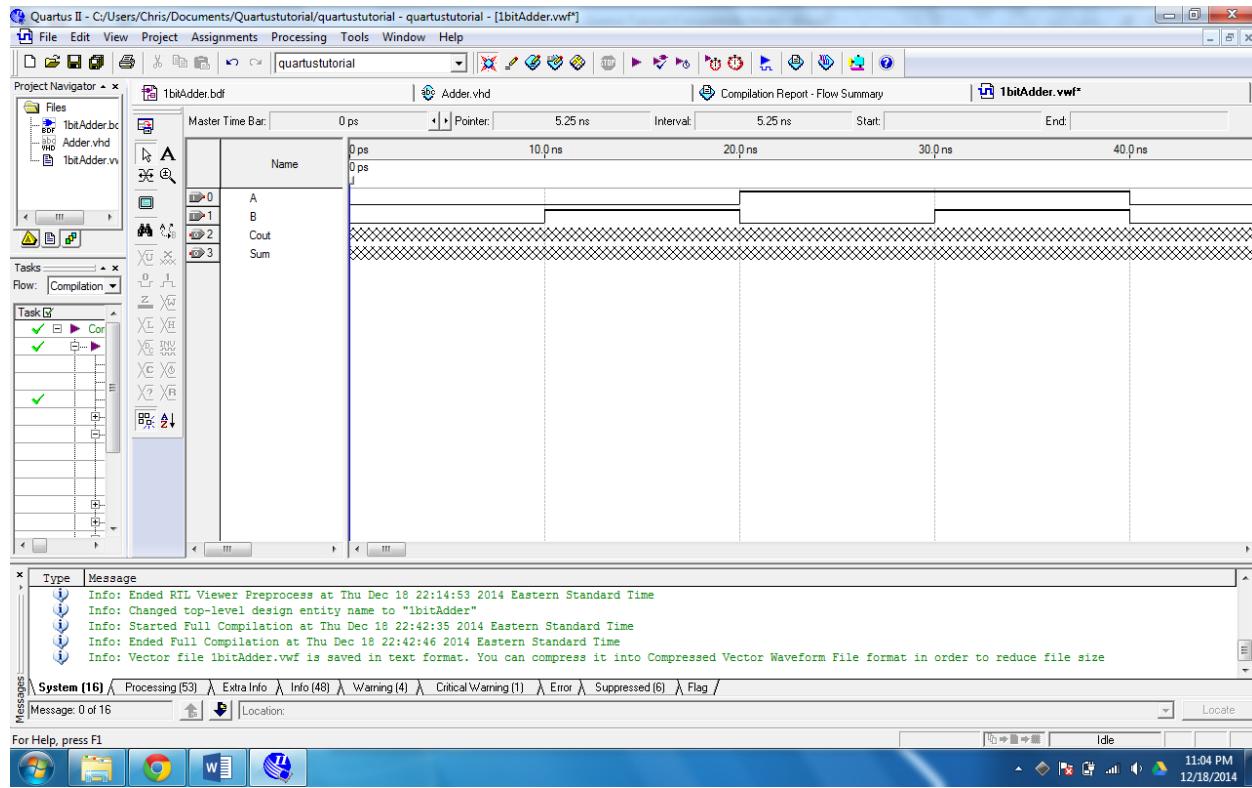
- Create the desired waveform through the buttons in the Waveform Edition on the left. For example: Forcing high/low, Random value, or Overwrite clock.
- Begin by overwriting the clock for A:



10. Set the period to 40ns.



11. Do the same for B, but with 20ns instead.



12. When you have created the waveforms, it is the time to simulate.

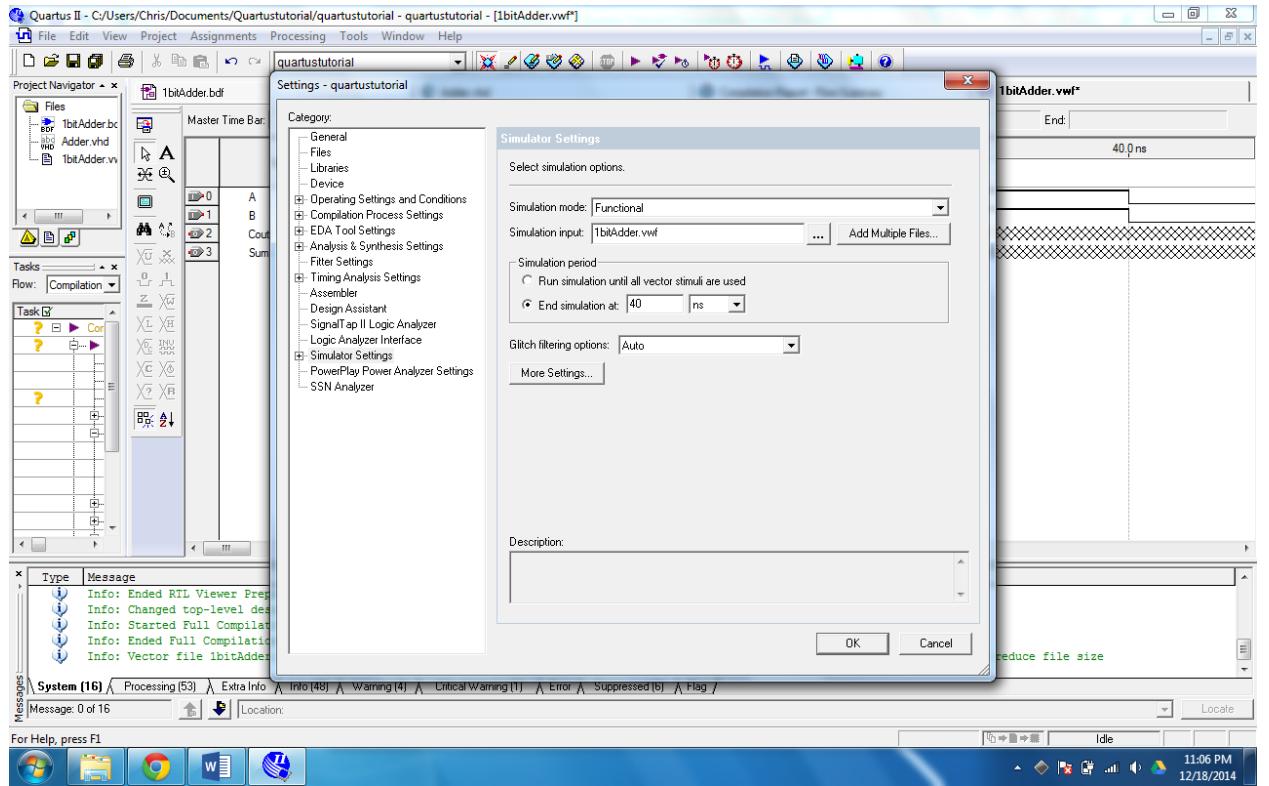
13. Save the vector waveform file (.vwf).

14. Go to Assignments -> Settings.

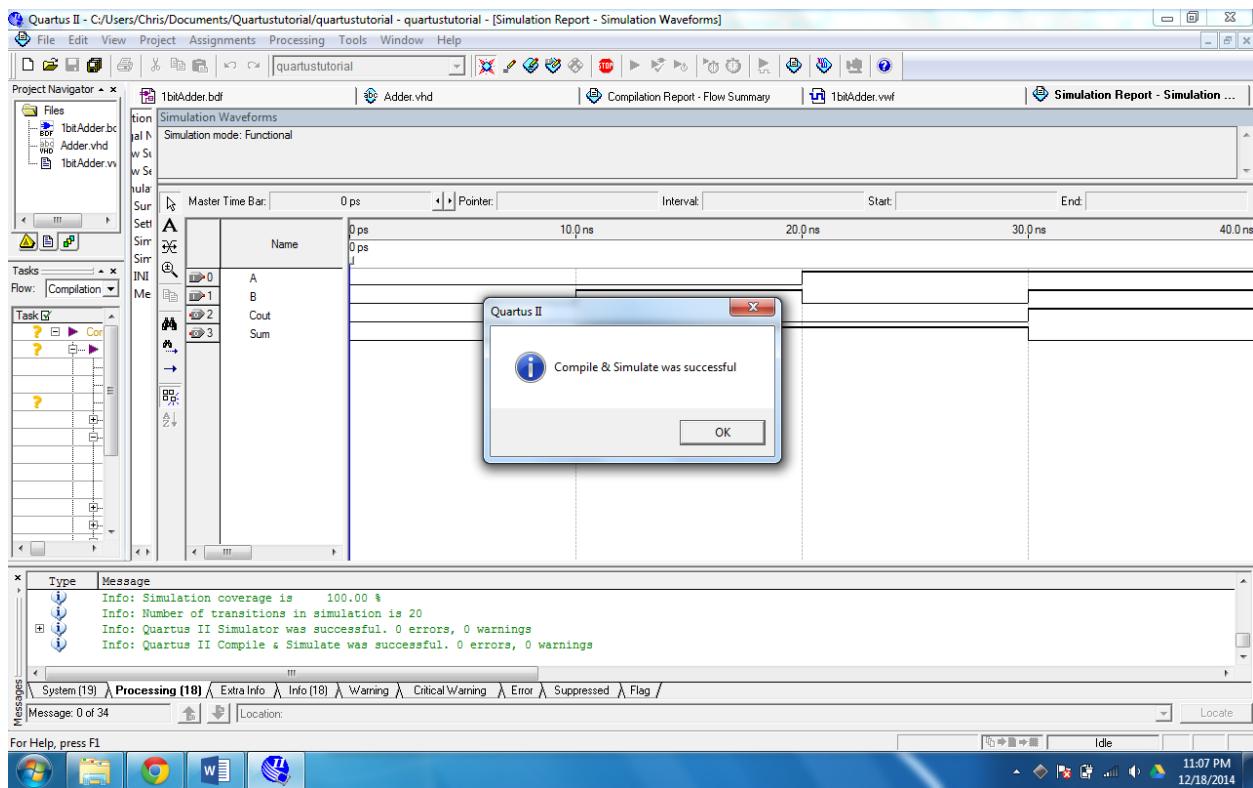
15. For Simulation mode, choose Functional.

16. For Simulation input, click on the button “...” and look for the .vwf file you just saved and select it.

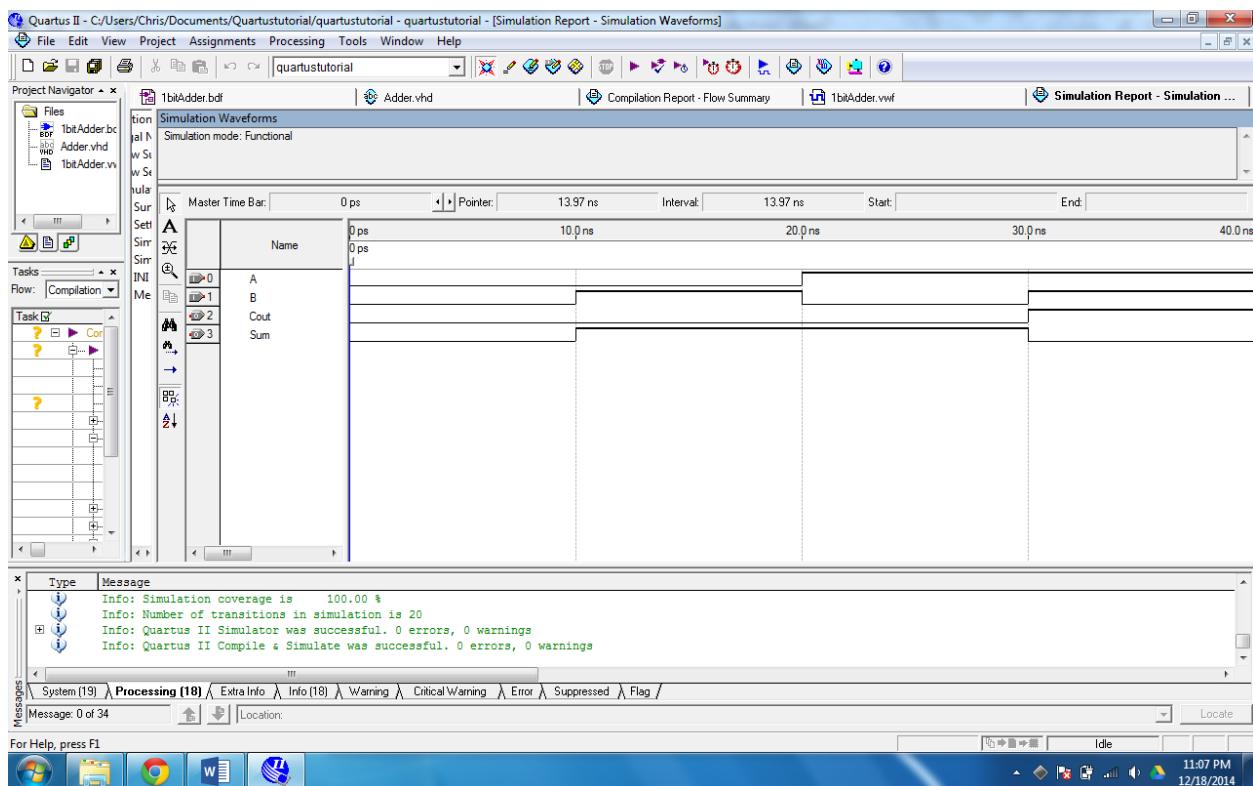
17. Under Simulation period, make sure to select “End Simulation at:” and pick a desired ending time so that the simulation doesn’t run forever. We can pick 40ns.



18. Press Ok.
19. Go to Processing -> Start Compilation and Simulation
20. Wait for the popup and make sure the message says “Compilate & Simulate was successful.”



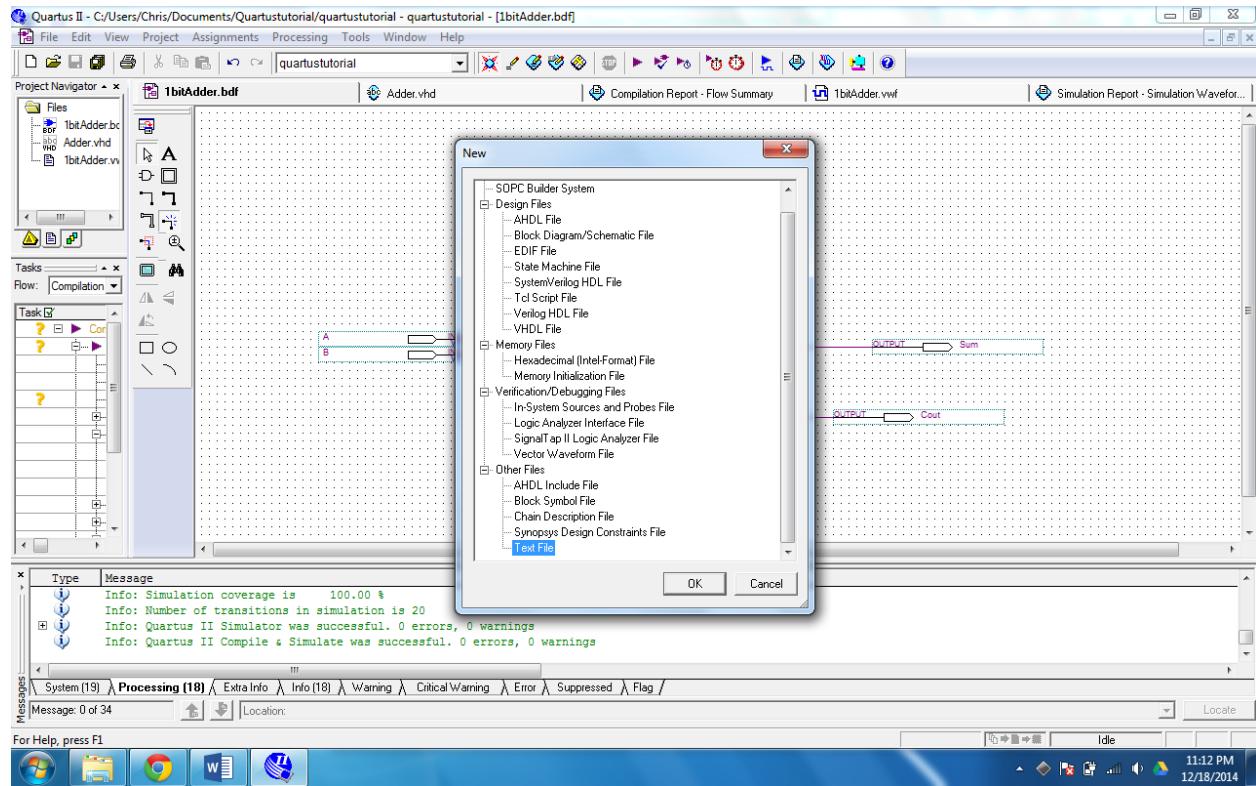
21. Check your waveforms and verify the outputs from the Simulation Report created.



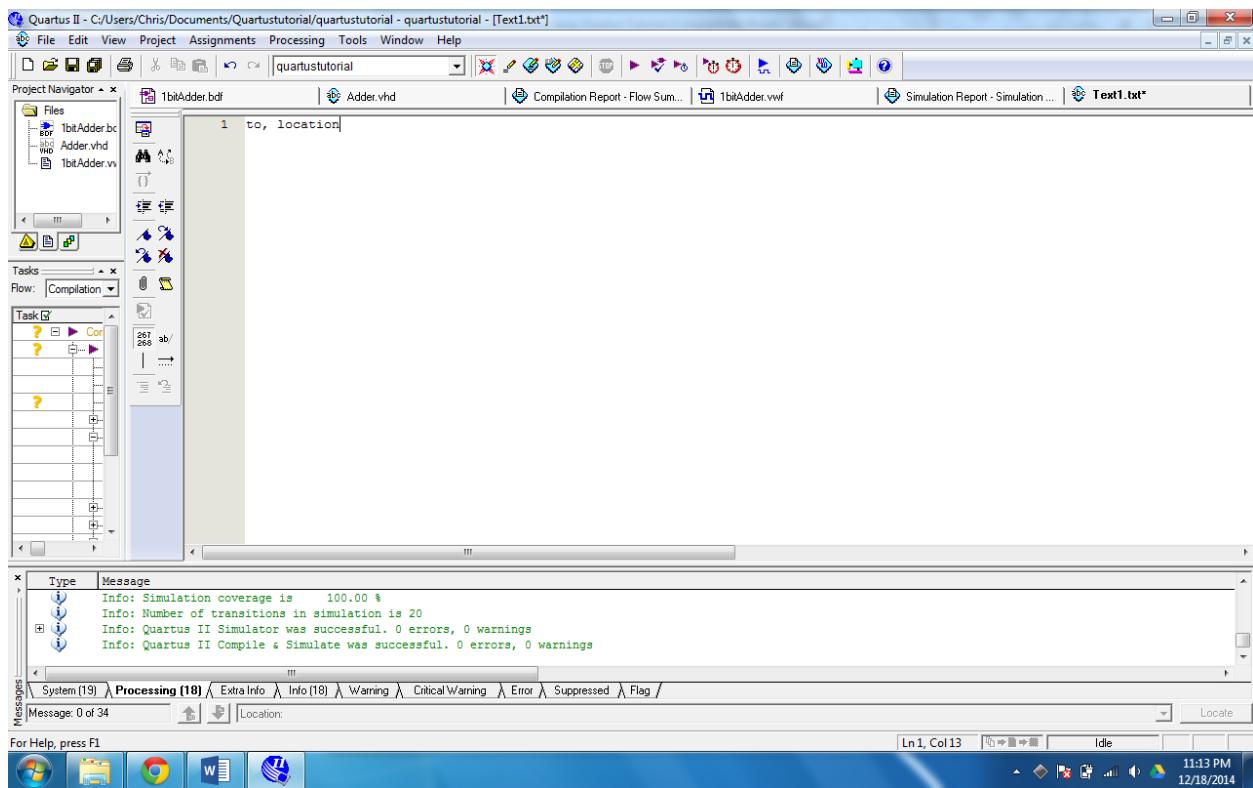
22. When the waveform is high, the input/output has a value of 1. When it is low, it has a value of 0. Each increments of 10ns will represent the values in the truth table. Since the values from the table match with the waveforms, the circuit has been verified.

Adding pin assignments

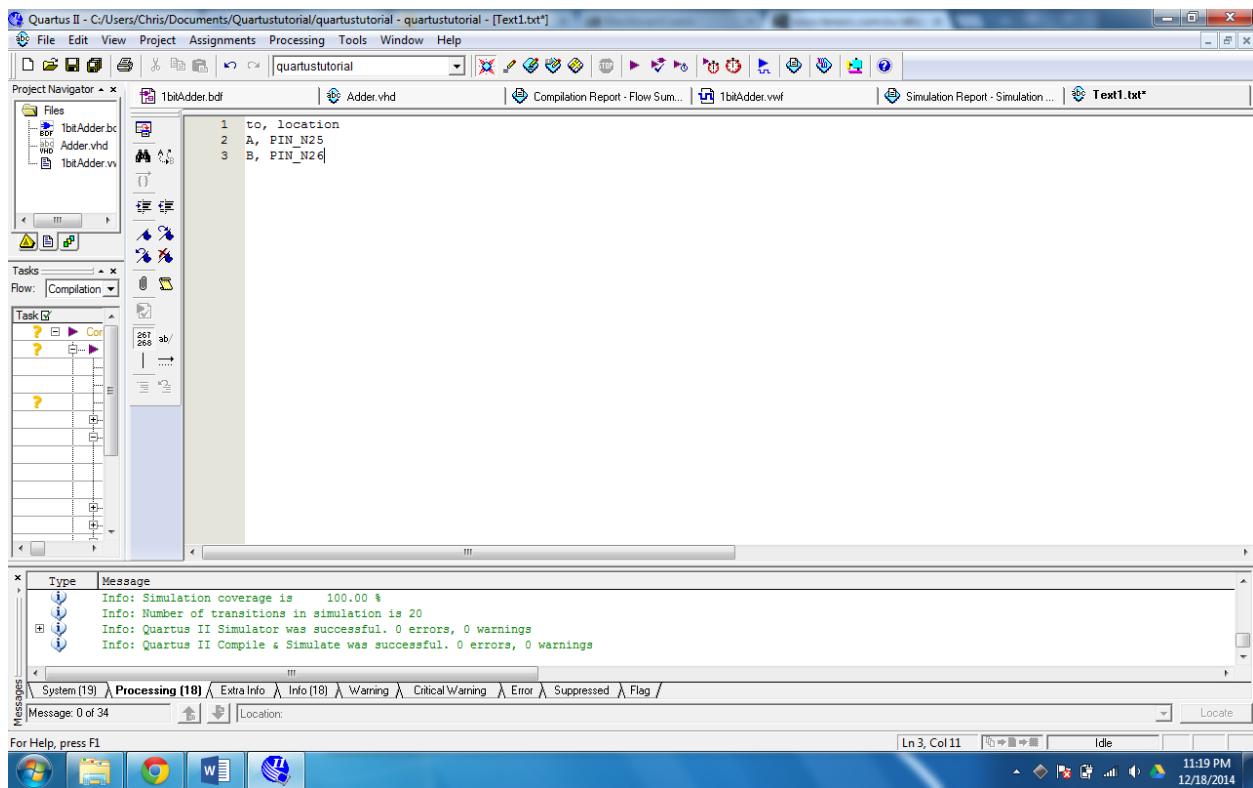
1. To test the circuit on the DE2 board, we must assign pins to each input and output.
2. Create a new text file. New -> Text File



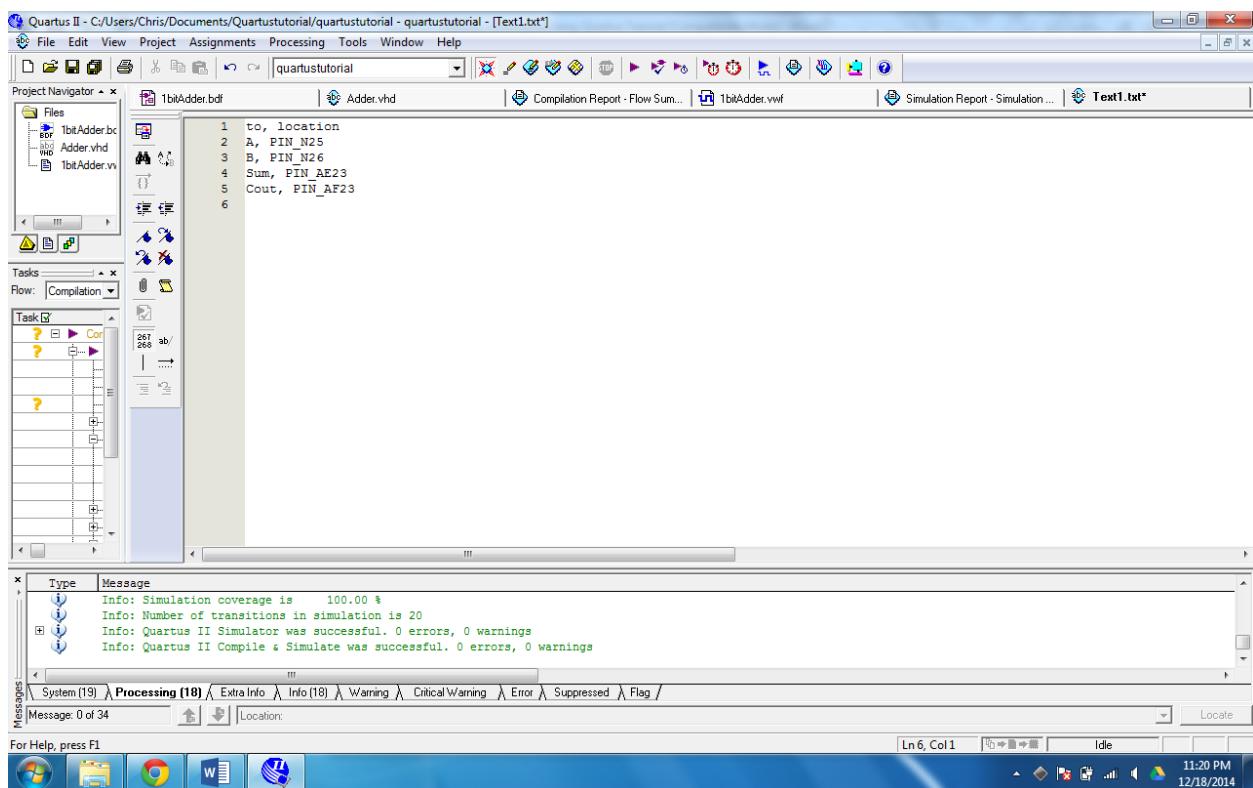
3. First line begin with: “to, location”



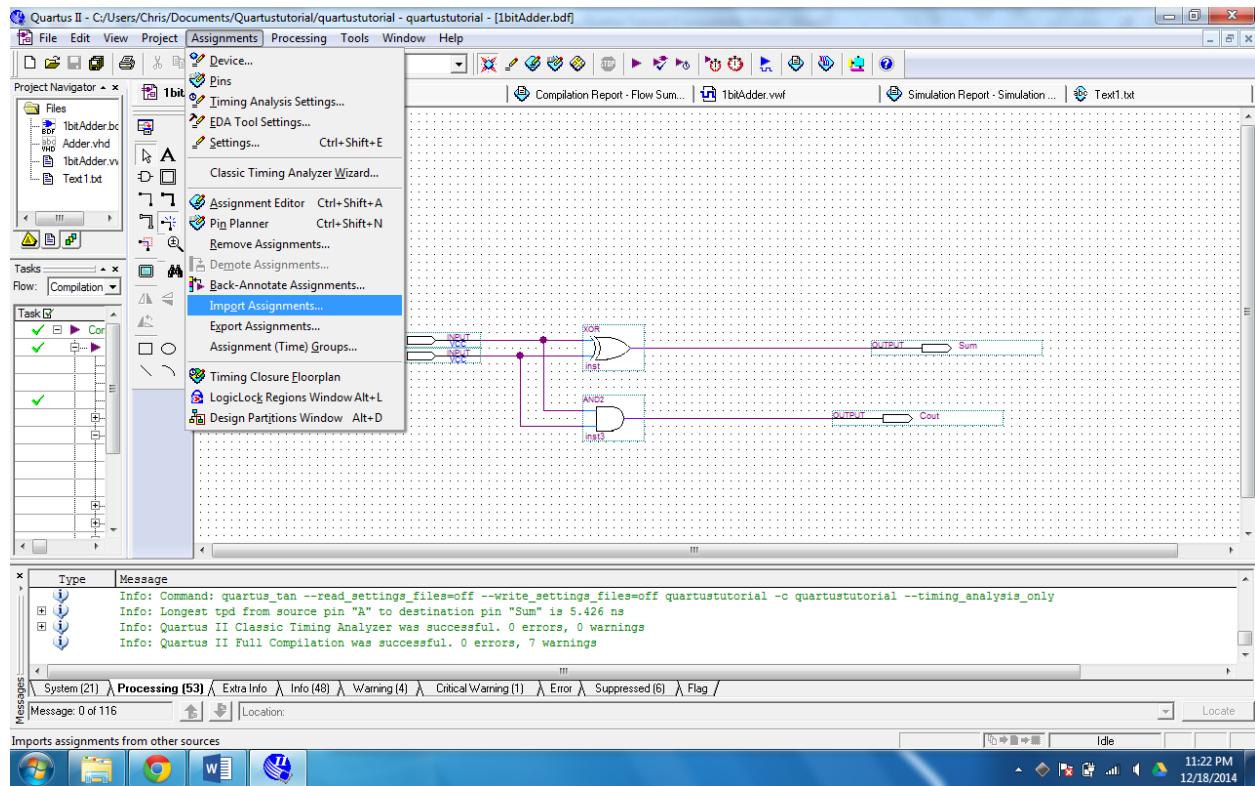
4. In the proceeding lines, continue with the same format from line 1. Start with the name of the input/output. Add a comma and a space, then the corresponding FPGA Pin No. We will be using SW[0] (PIN_N25) and SW[1] (PIN_N26) for A and B, respectively.



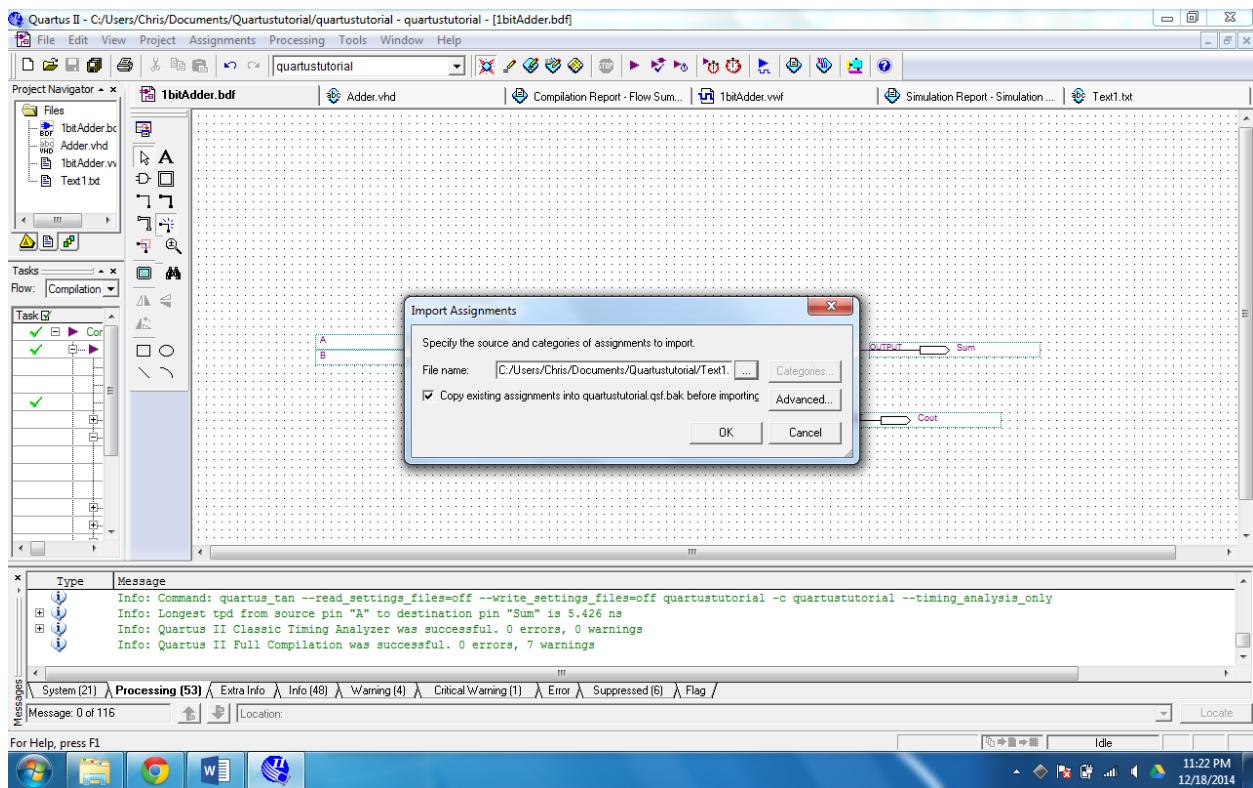
5. Next, the outputs LEDR[0] (PIN_AE23) and LEDR[1] (PIN_AF23) are assigned to Sum and Cout, respectively.



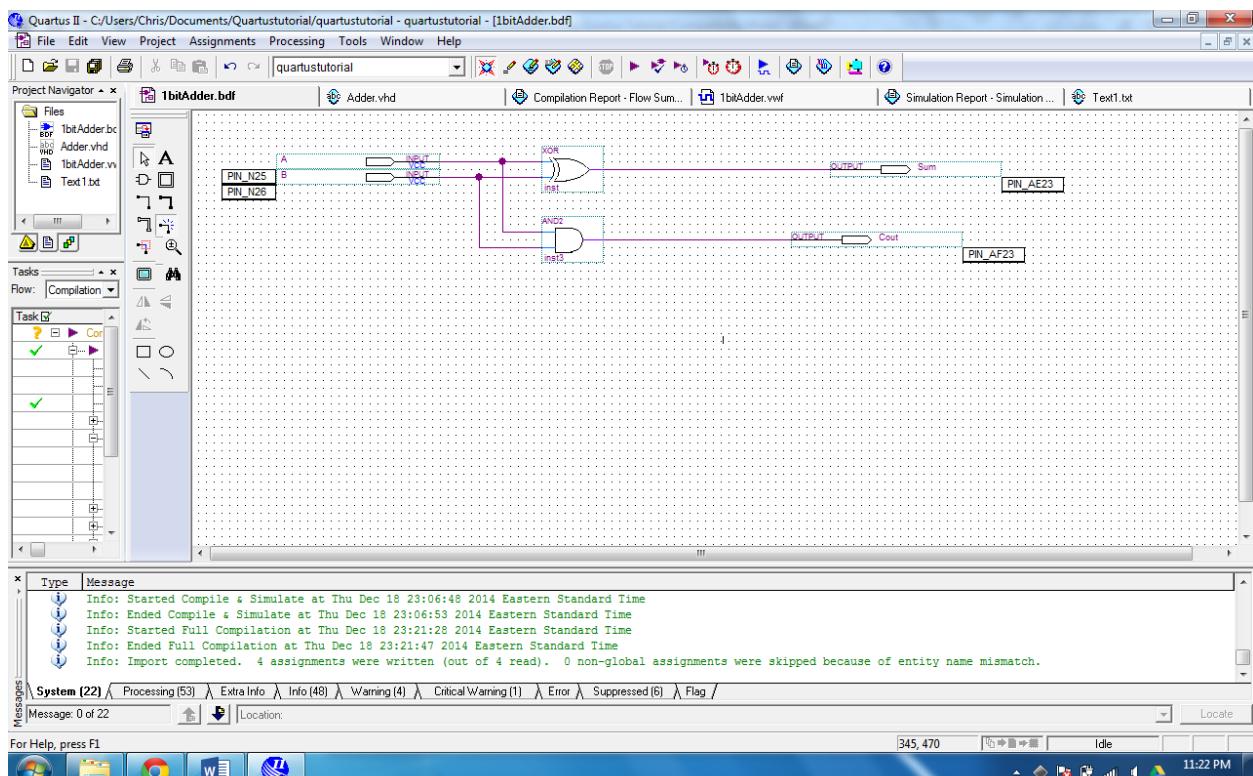
6. When you are finished, save the text file.
7. Go the block diagram file (.bdf) that you want to import the assignments to.
8. Make sure it is compiled and is the Top-Level Entity before you do this.
9. Select Assignments -> Import Assignments.



10. Select the text file that you saved previously and press Ok.



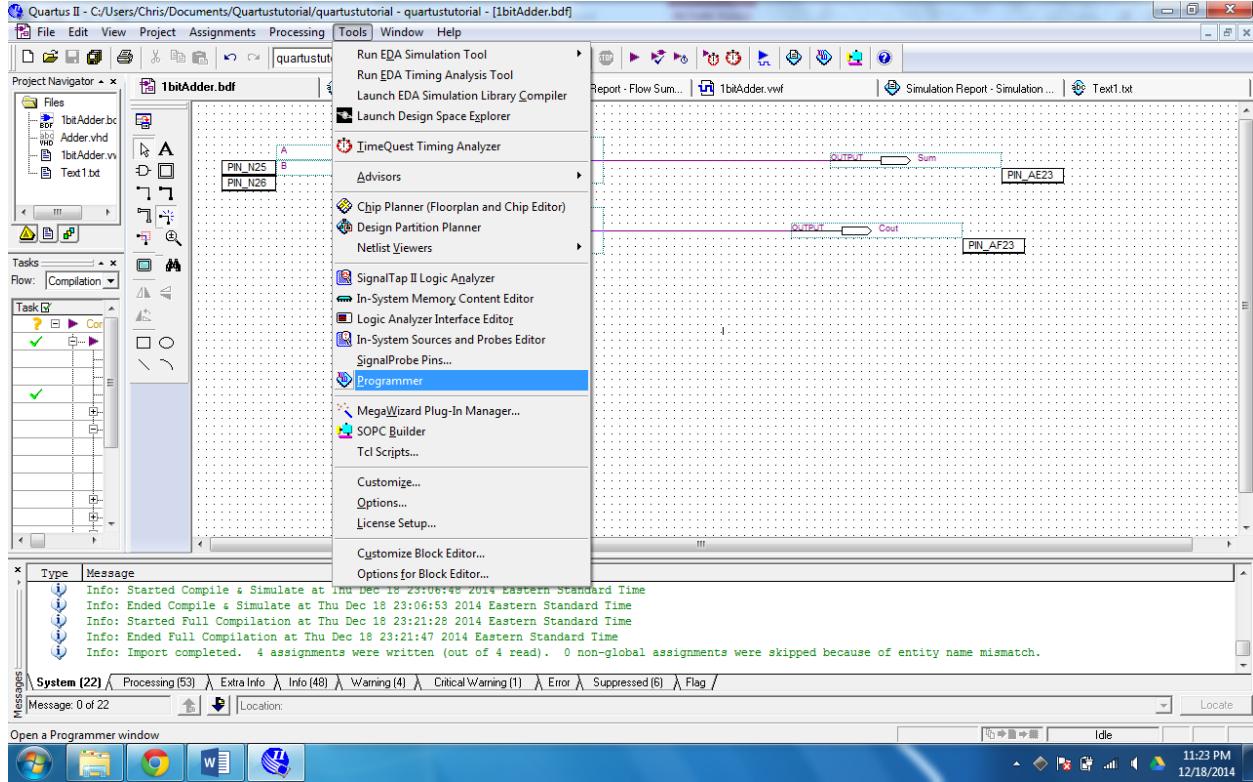
11. Once the pins are assigned, they should appear on the block diagram file as shown:



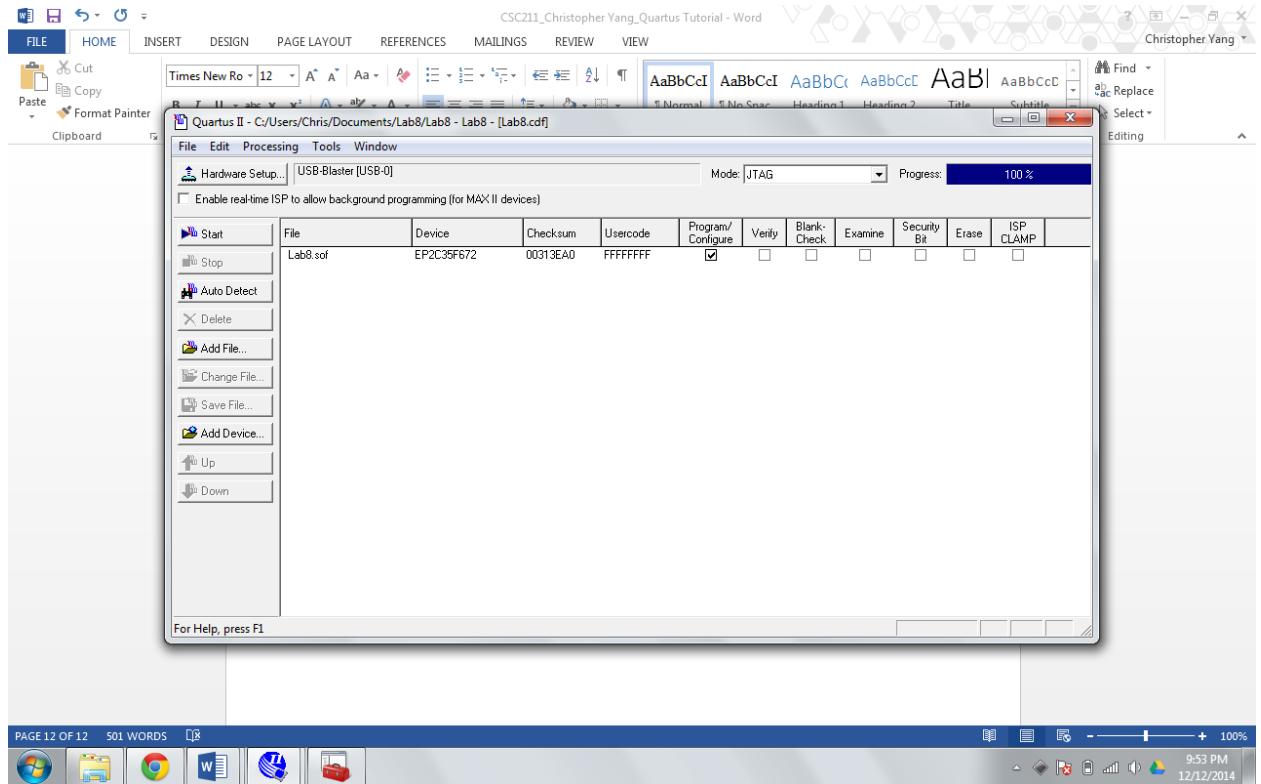
12. Compile the file again to verify that there are no errors after pin assignment.

Verifying your circuit on the DE2 Board

1. After assigning the pins, simulating and verifying that the circuit works, plug in the DE2 Board's USB.
2. In Quartus, go to Tools -> Programmer.



3. Click Start.



4. Now you can begin verifying the circuit on your DE2 Board.