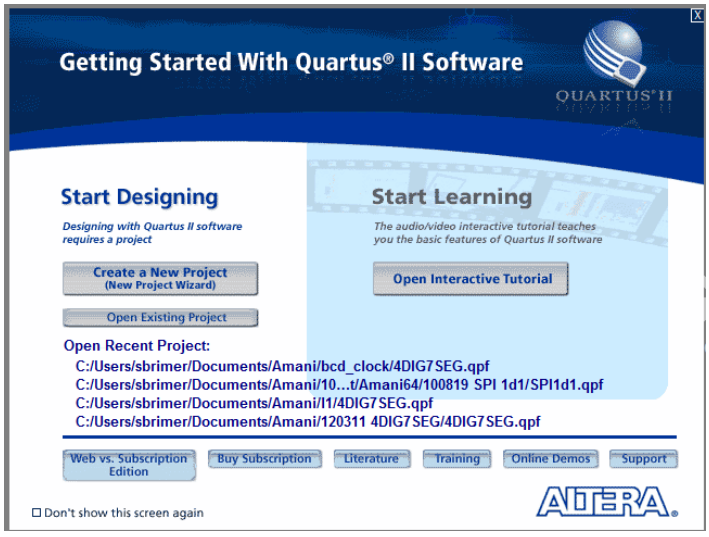
CSUS Tutorial for

ALTERA: QUARTUS II WEB EDITION

Intended for use in CSGC as well as CPE 64

This tutorial assumes you are using an ECS Lab computer or your own computer that has had Altera Quartus II Web Edition installed properly and the IDE has recognized your JTAG Programmer. If your software has trouble finding the JTag programmer, please refer to the index, *Troubleshooting the JTag Programmer*.

Getting Started with Quartus II

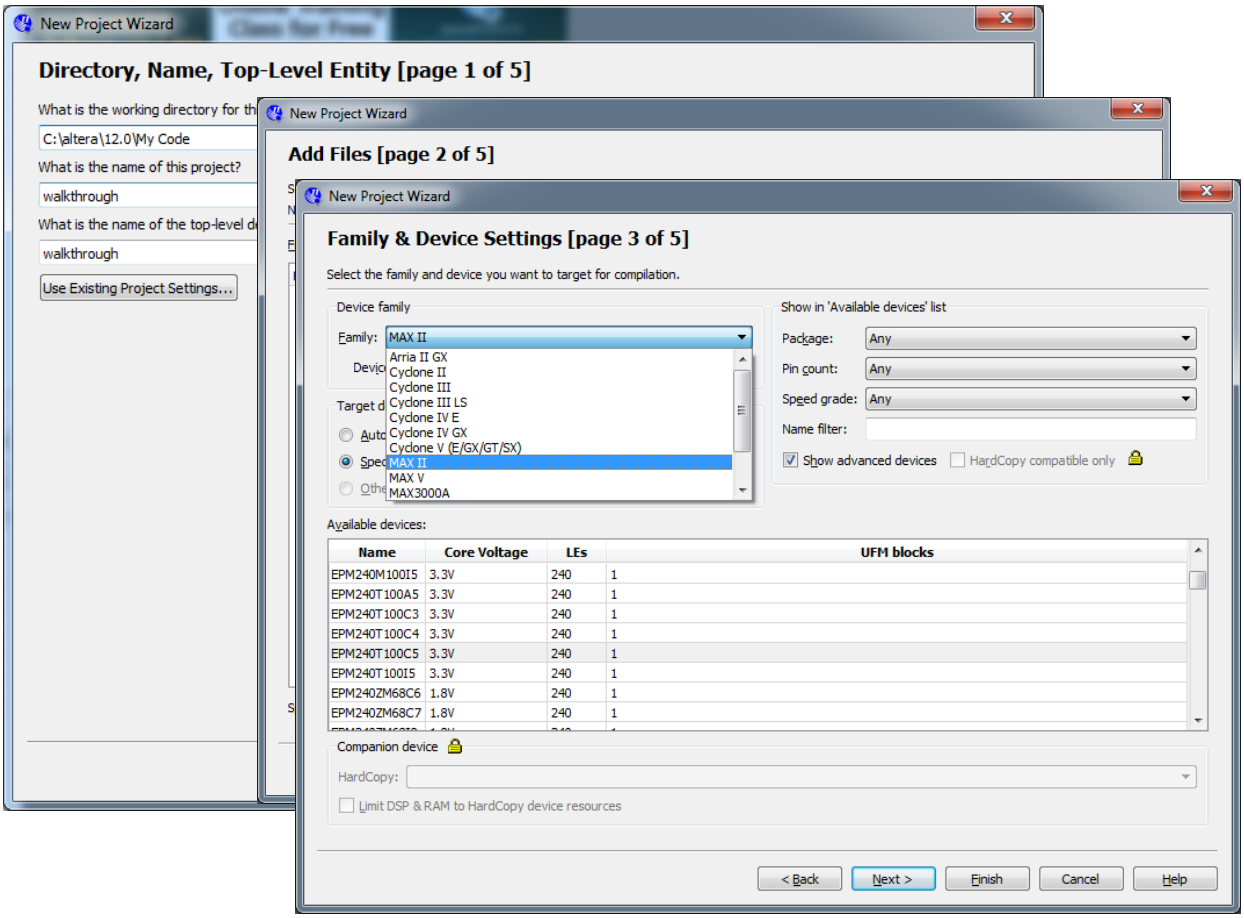


From the Start menu type ‘quartus’ then select ‘Quartus II 12.0 Web Edition’

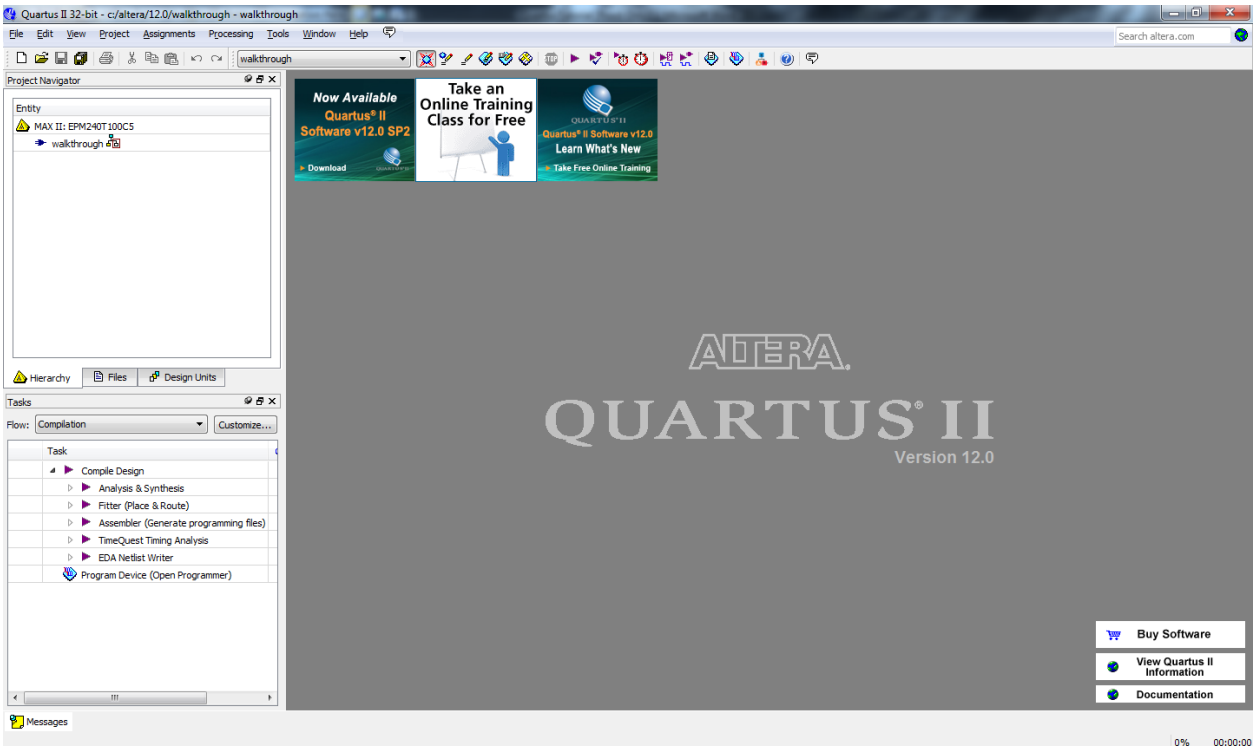
After the software opens, click on the button ‘*Create a New Project*’

You will be prompted to select a directory and a name for your project. If you are on your own computer it is advised that you switch to ‘c:\altera\12.0\My Code’ if you are on a lab computer you may have to save to the desktop or the T: drive. Be sure to create a folder ‘My Code’ to contain your project files. For this demonstration, name your project ‘walkthrough,’ click next.

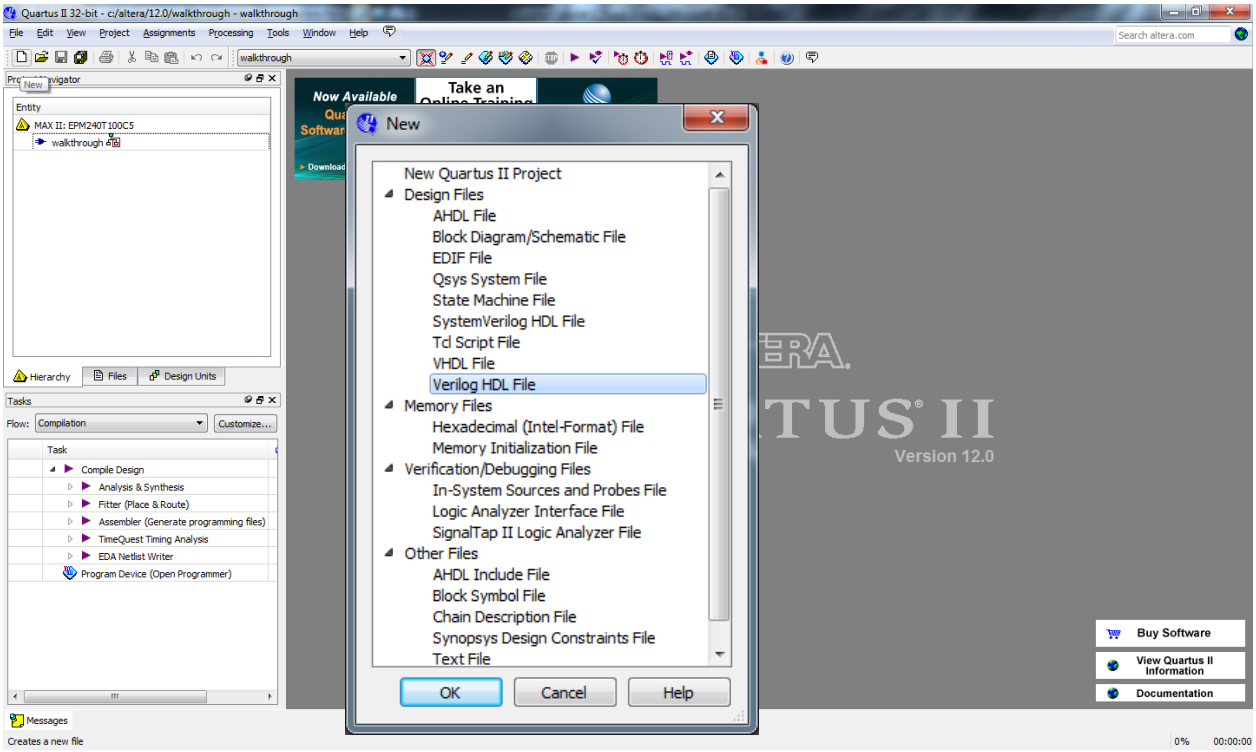
You will be brought to a screen that says ‘Add Files,’ for this tutorial we will not be loading files, so click next. You will be brought to a screen that says ‘Family & Device Settings’ here you will select your device family. If you look closely, the main chip on your Amani board should say one of the options in the dropdown menu. Select the appropriate option, then go down to ‘Available Devices’ and select the device ID from your chip, click ‘Finish.’



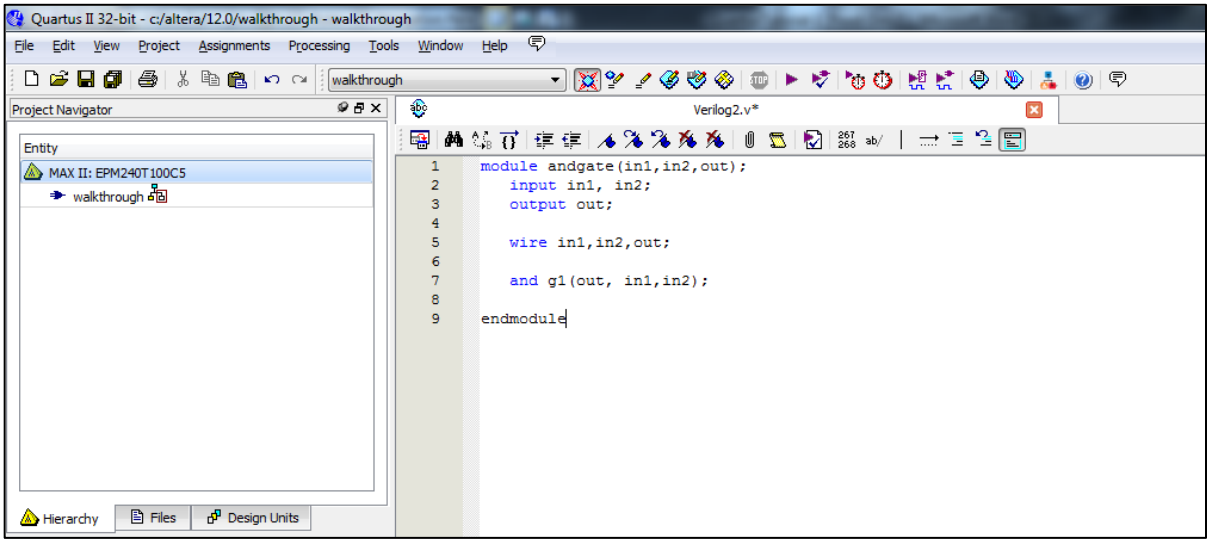
Your screen should look like this. This is the main window for the Quartus II IDE (Integrated Development Environment.) Here we can add Verilog files to our project, change code, create symbols that represent code, wire symbols together, map out pins, compile and load our designs to chip.



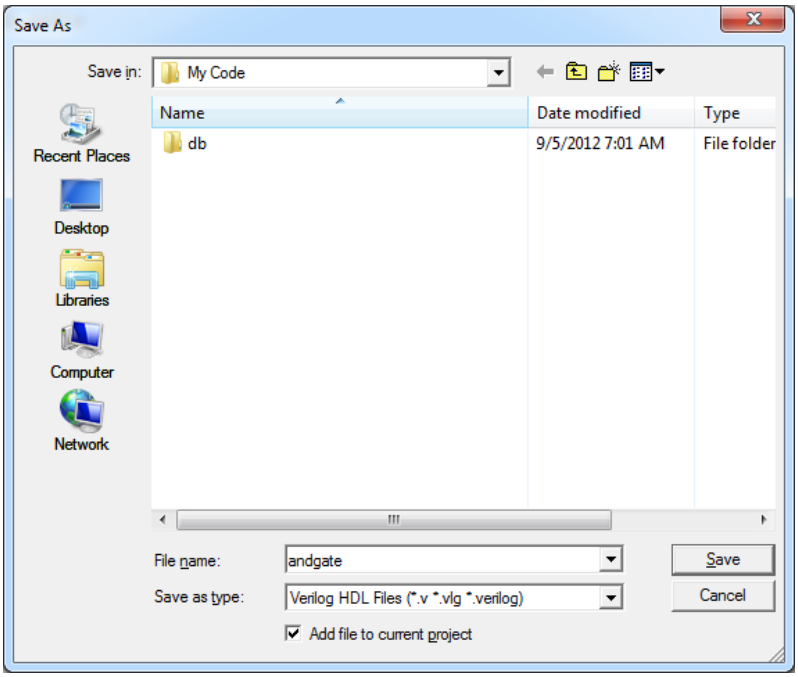
Let's get started writing some Verilog code. You can click the 'New File' icon or 'New File' from the File menu. The 'New' menu will now pop up. Select 'Verilog HDL File' then click ok.



Now your main window will have a white background where you can type up your Verilog code. Now you can type in the code that you see in my window.



When you are done typing your code go to the File menu and select 'Save As…' Be sure to use the file name that corresponds to the name of the module from the code. When your design gets large and includes many Verilog files it is very important that the file names reflect the modules contained within.



After saving your file with the correct filename, repeat the process of opening a new file, then write in each of these modules and be sure to save them as separate files with corresponding file names.

```
module andgate(in1,in2,out);
      input in1,in2;
      output out;

      wire in1,in2,out;

      and g1(out,in1,in2);

endmodule
```

```
module notgate(in1,out);
      input in1;
      output out;

      wire in1, out;

      not g1(out,in1);

endmodule
```

```
module orgate(in1,in2,out);
      input in1,in2;
      output out;

      wire in1,in2,out;

      or g1(out,in1,in2);

endmodule
```

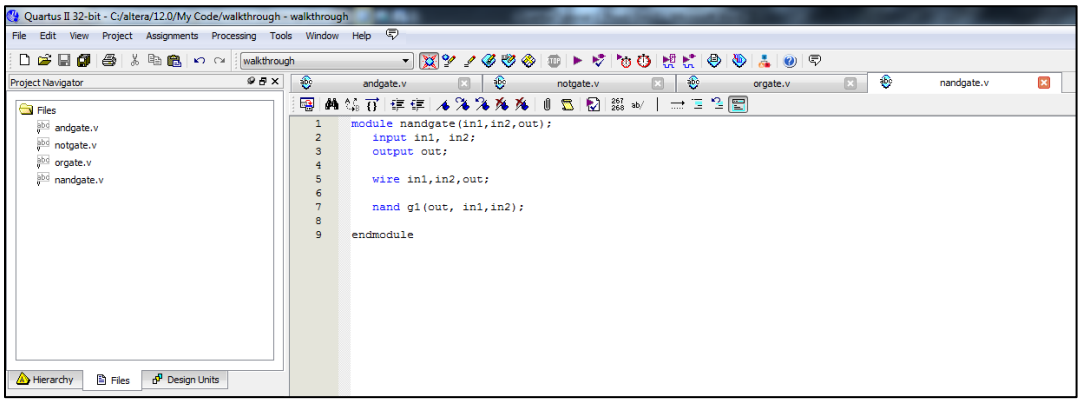
```
module nandgate(in1,in2,out);
      input in1,in2;
      output out;

      wire in1,in2,out;

      nand g1(out,in1,in2);

endmodule
```

Now you may be thinking, I typed and saved my code files where did they go? Well, they can be found in a couple of places. I clicked on the tab marked ‘Files’ in the section on the left and that will show me a list of the files that I have created in this project or have included in this project. I left my files open after making them so they are also tabbed across the top, in case I needed to check that I programmed my files correctly.

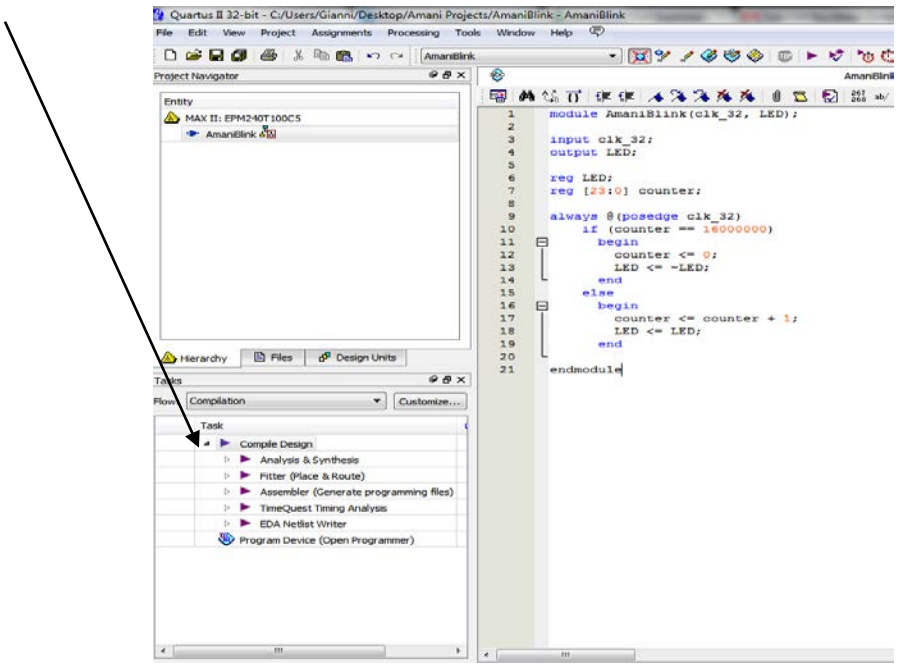


Programming the Amani - LED Blink Example

Now that you know how to set-up a Quartus project with some Verilog code, let's upload a basic blink LED program onto our board. Begin by creating a new project called: ***AmaniBlink***. Create a new verilog file; copy and paste the following Verilog code into the text editor:

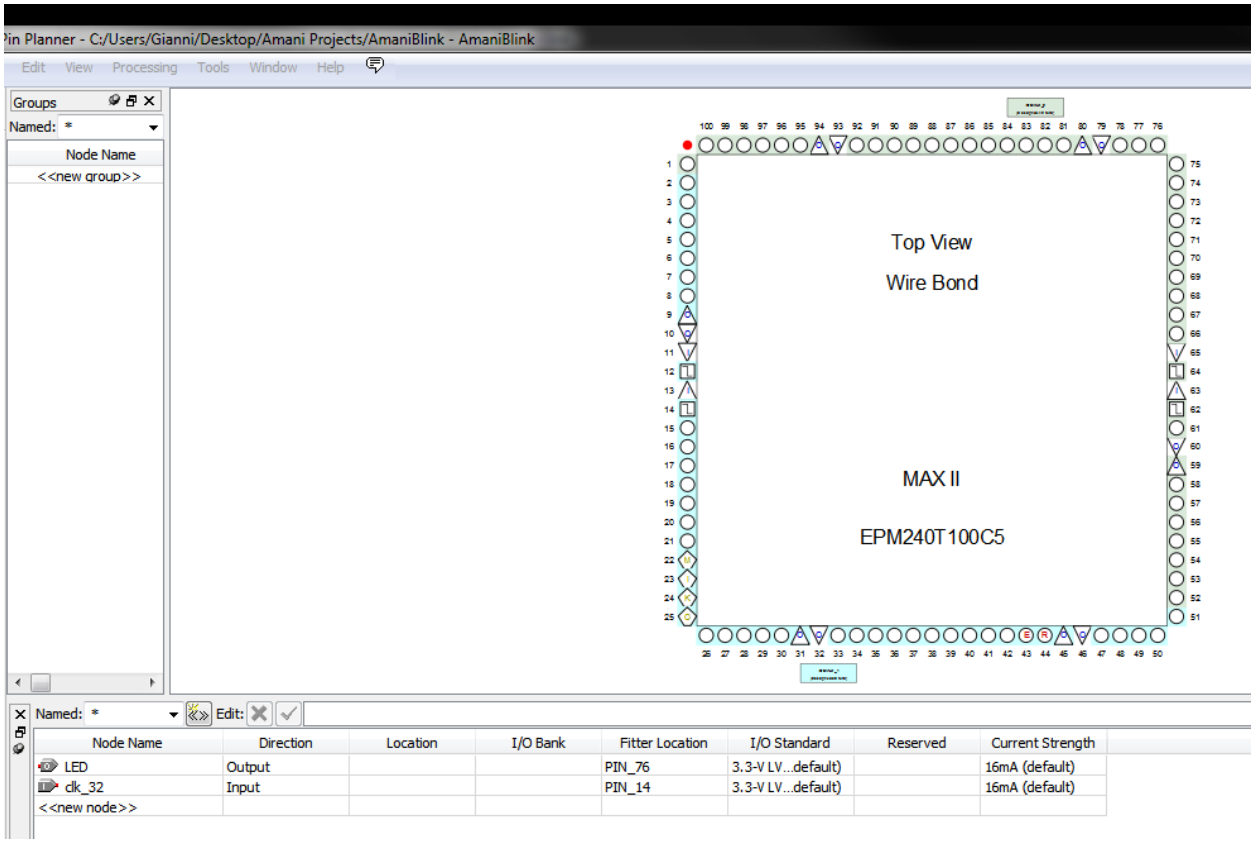
```
module AmaniBlink(clk_32, LED);
input clk_32;
output LED;
reg LED;
reg [23:0] counter;
always @(posedge clk_32)
   if (counter == 16000000)
     begin
       counter <= 0;
       LED <= ~LED;
     end
   else
     begin
       counter <= counter + 1;
       LED <= LED;
     end
endmodule
```

Next, we will need to verify that our code is valid by compiling our design. Locate the compiler tool on the bottom left corner of the Quartus IDE.



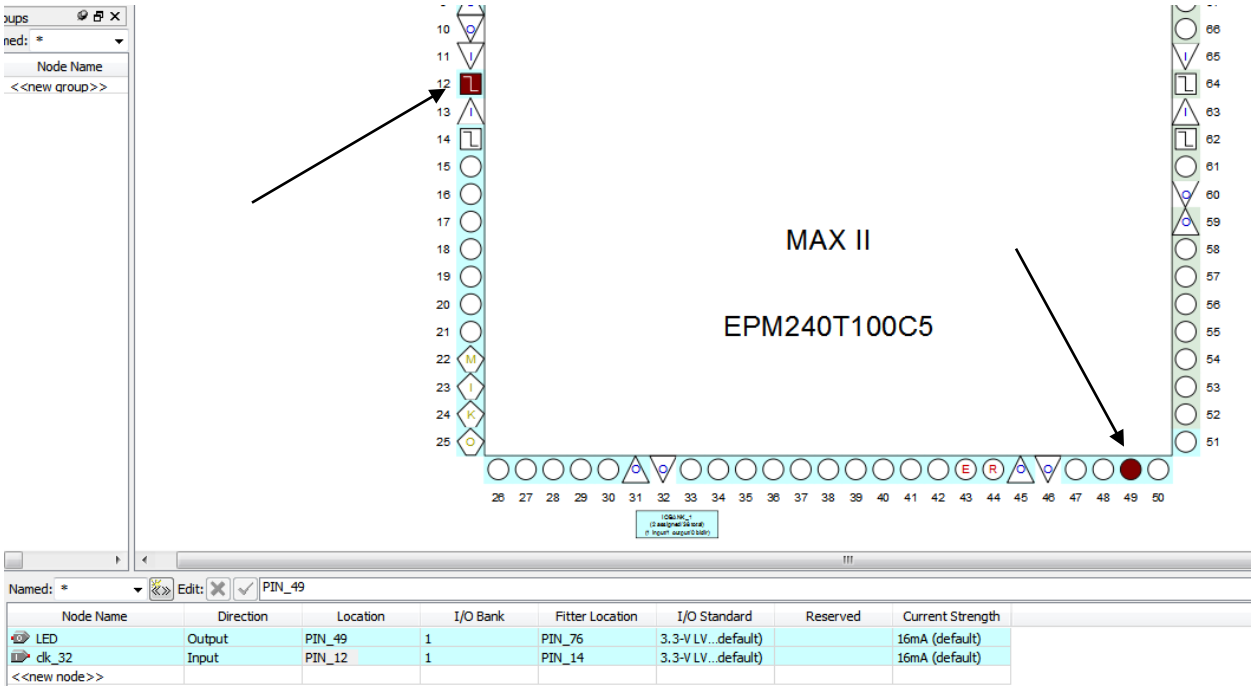
Right click on the ***Compile Design*** and click ***Start*** to begin the compile design process. A few ignorable warnings may pop up if you are using an unlicensed (free) version of Quartus; you should not have any errors. If the compilation process is successful, a *Compilation Report* will pop-up with information relevant to your project; verify that all the device settings are correct.

Now that we have successfully compiled our program, we need to set up our I/O on our board to illustrate our example code on the CPLD. On the ***project navigator***, click on ***assignments****, then* ***pin planner****.* A top-view of the CPLD will pop-up with all the I/O's.

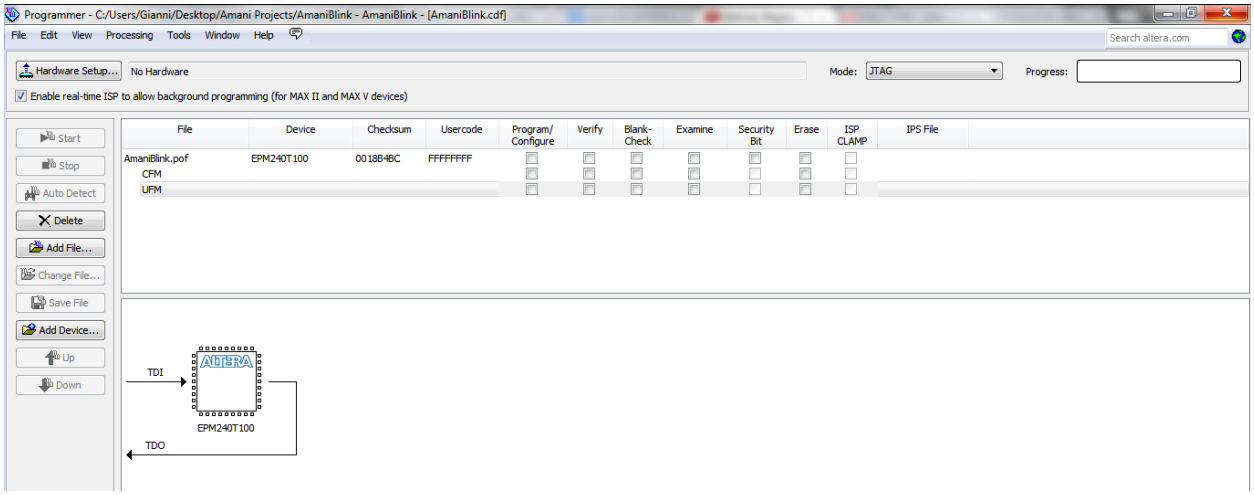


Download a complete I/O map on the amani64 website:
http://majolsurf.net/projects/schematics/120311%20Amani%20Arduino%20Pin%20Map.xlsx

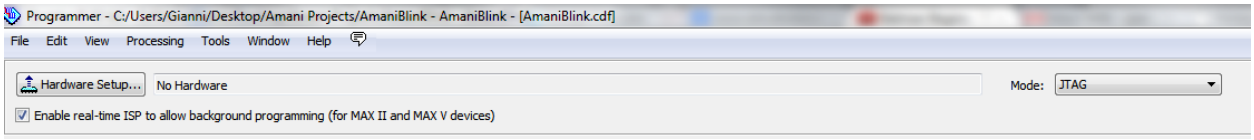
The AmaniGTX is equipped with several global clocks with different MHz cycles. For this demonstration, we'll choose the 64Mhz clock located on Pin 12. Under the location taskbar, double click on the empty box in the clk_32 row; a drop down box will appear and a list of pins will be displayed, select Pin 12 for the clock input. Next we want to define our LED output to view the blinking; set the output to Pin 49. Notice after setting the I/O on the board, the CPLD updates its diagram as well.

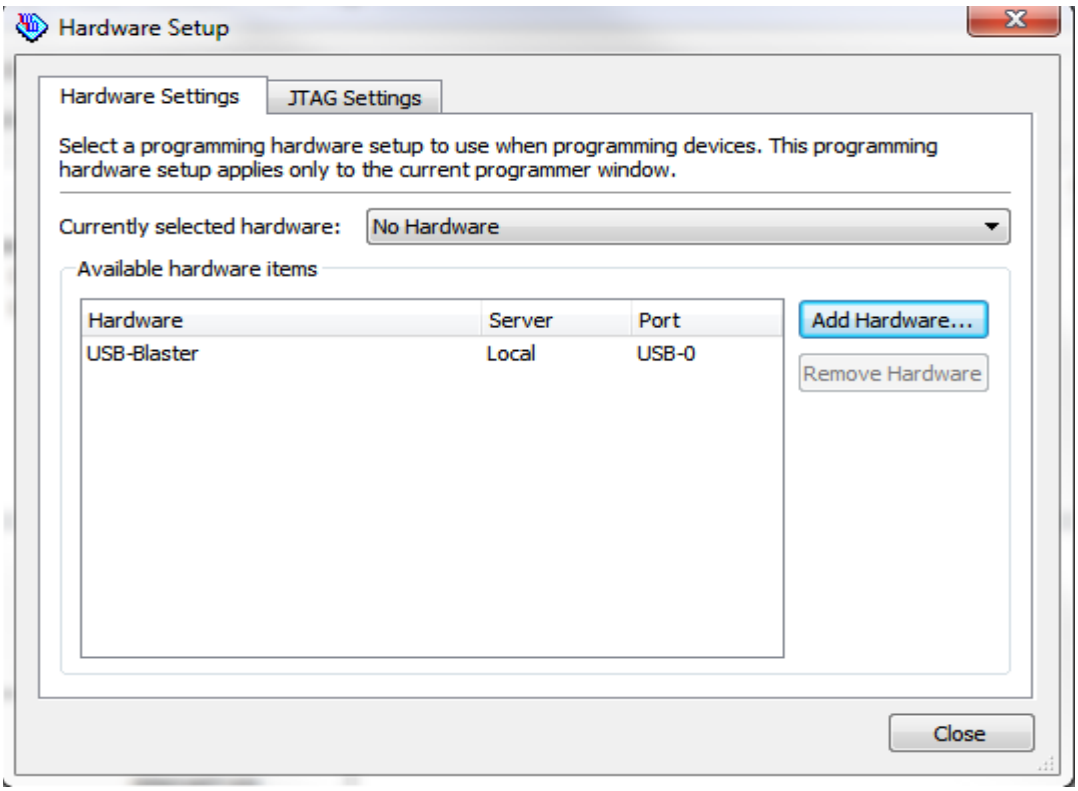


This indicates that you have successfully set your pins, close the pin planner. BUT, before we continue on to program our board, we have to recompile our software with the new I/O assignments, re-run the ***Compile Design*** process. After successfully compiling your project, on the ***project navigator***, click on ***Tools***, then ***Programmer***. The window below will appear



Plug in your device via MiniB connection from the Amani GTX board to your computer. Notice the ***Hardware Setup*** bar shows ***No Hardware***. Click on ***Hardware Set up***

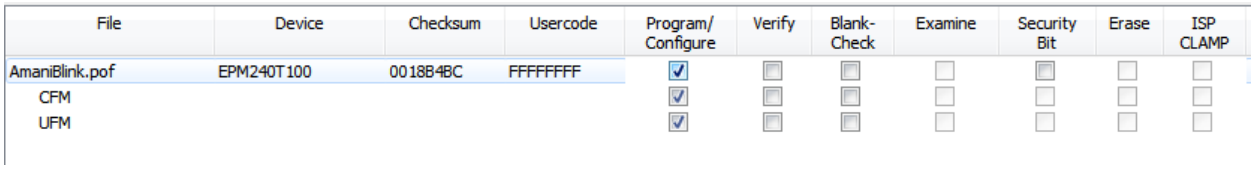




On the ***Currently selected hardware*** drop-down box, change this to ***USB-Blaster [USB-0]***. If this option is not available to you, complete steps in Appendix A, then re-open your project and complete the above steps again.

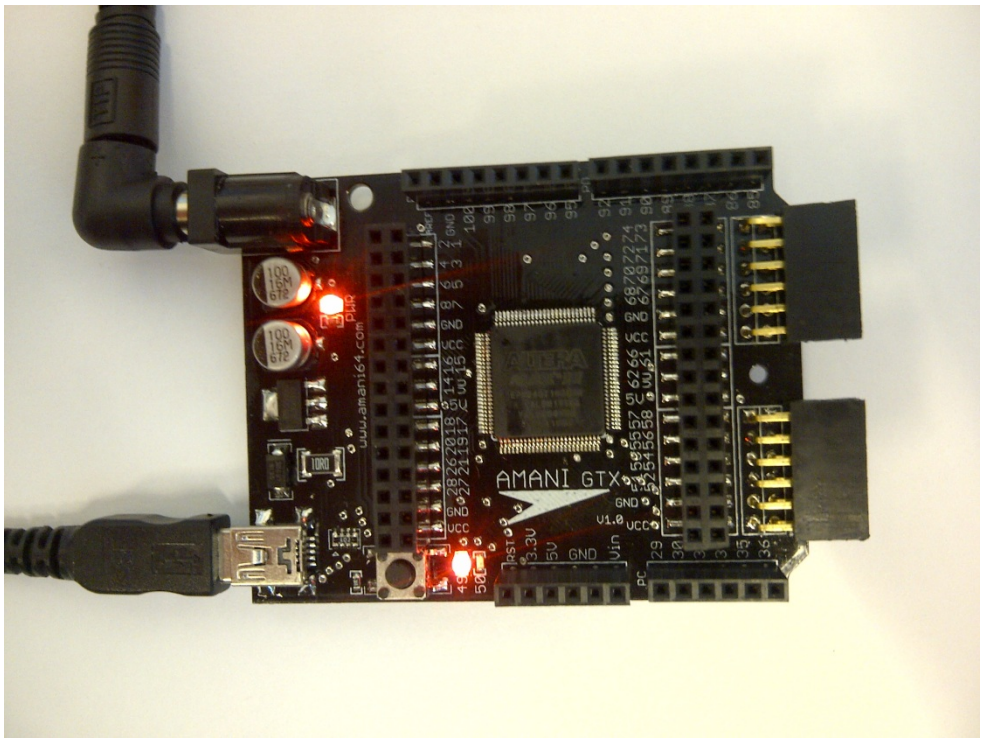
On the left-hand toolbar, click on ***Auto Detect*** to ensure that your computer recognizes the Amani CPLD. Note that there should be a picture of your CPLD on the bottom of the page with the model number.

Click on the ***Program/Configure*** checkbox next to your ***AmaniBlink.pof*** file as shown below:



Click on the ***Start*** button to program your compiled Verilog file onto the CPLD. Do not attempt to disconnect the USB cable while programming; programming is finished when the ***Progress*** bar reaches 100%.

On the AmaniGTX board, LED 49 should be oscillating.



Troubleshooting Tips if LED 49 is not oscillating:

- Try cycling the power of the CPLD
- Remember, a USB port is only capable of driving 500mA to a load, it may be necessary to provide an external power supply OR you can use a Dual USB to MiniB cable to double the current to the board.
- A long USB cable is susceptible to losses and noise if not shielded, consider using a short (<6inch) cable.

Appendix

A. Troubleshooting the JTag Programmer

If you are having trouble locating the USB-Blaster in the CPLD programming example, follow the steps below to install the appropriate drivers onto your computer.

i. Download the quartus12fix.zip file containing the USB-Blaster drivers, your teacher will provide the download link.

ii. Copy the following files into: ***C:\altera\12.0sp2\quartus\bin64***
   a. msvcr100.dll
   b. tbbmalloc.dll
   c. tbbmalloc_proxy.dll

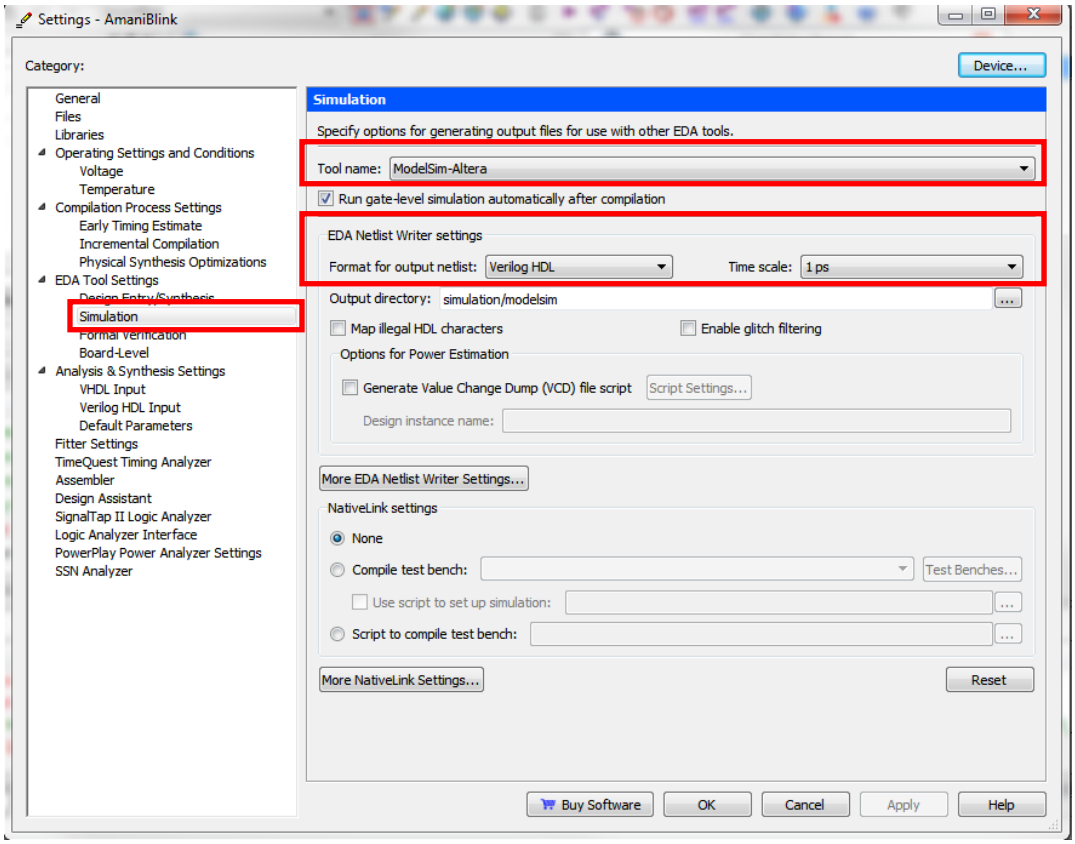
iii. Run your command prompt as an administrator
   a. Type the following into your command prompt
      i. cd C:\altera\12.0sp2\quartus\bin64
      ii. jtagserver --stop
      iii. jtagserver --uninstall
      iv. jtagserver --install
      v. jtagserver --start

iv. Repeat the hardware set-up in the ***Programming the Amani*** section

B. Adding the ModelSim Simulator

If you have only downloaded the Quartus II IDE, it may be necessary to manually configure the Altera simulation software. Go to the Altera website: https://www.altera.com/download/software/quartus-ii-we and located the link: ***ModelSim-Altera Starter***. Follow the instructions on the Altera website and run the installer.

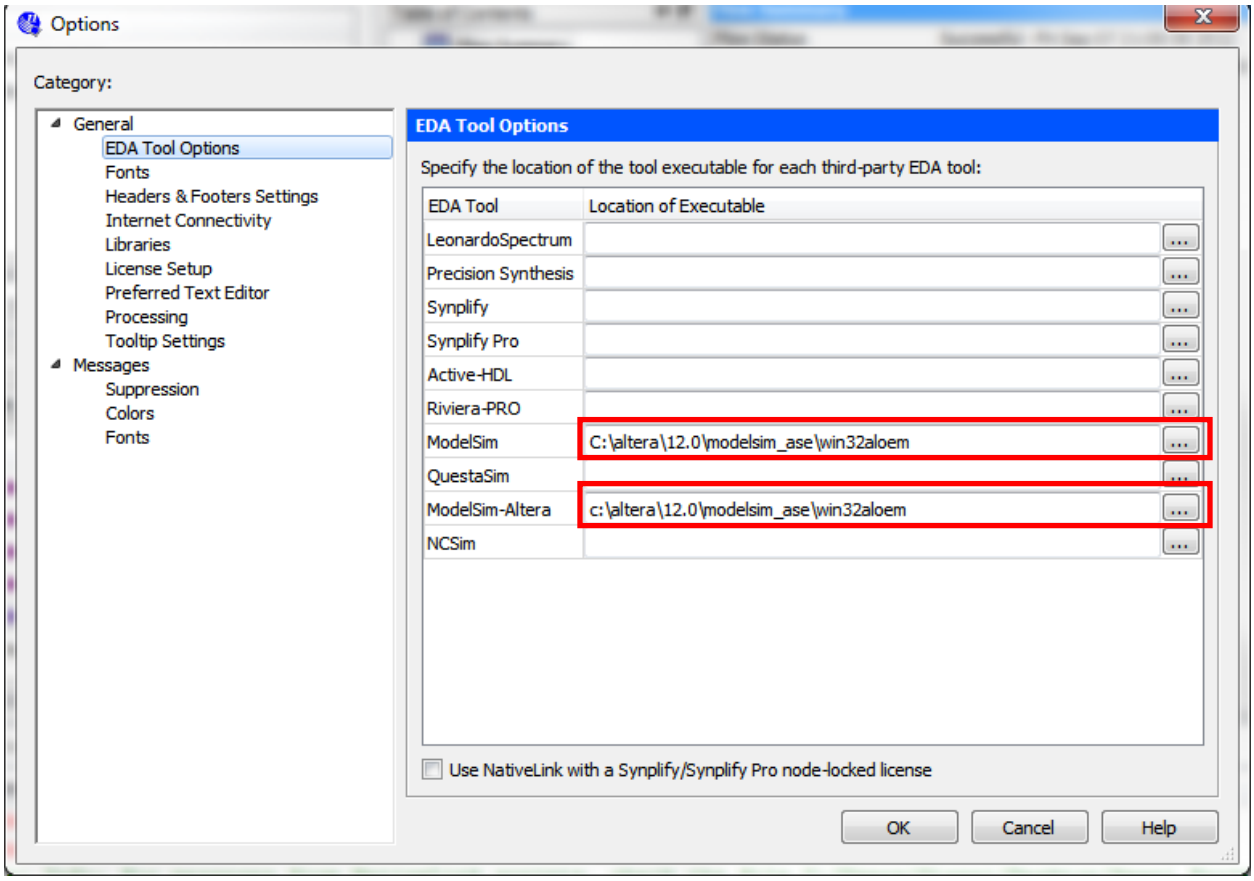
Once the installation process is complete, you will need to define the directories of the ***ModelSim-Altera*** simulation tool manually in the Quartus IDE. In Quartus, go to the ***project navigator***, and click on ***Assignments*** then ***Settings.*** The window on the next page will appear:



Under ***Tool Name***, click the dropdown box and select ***ModerlSim-Altera***.

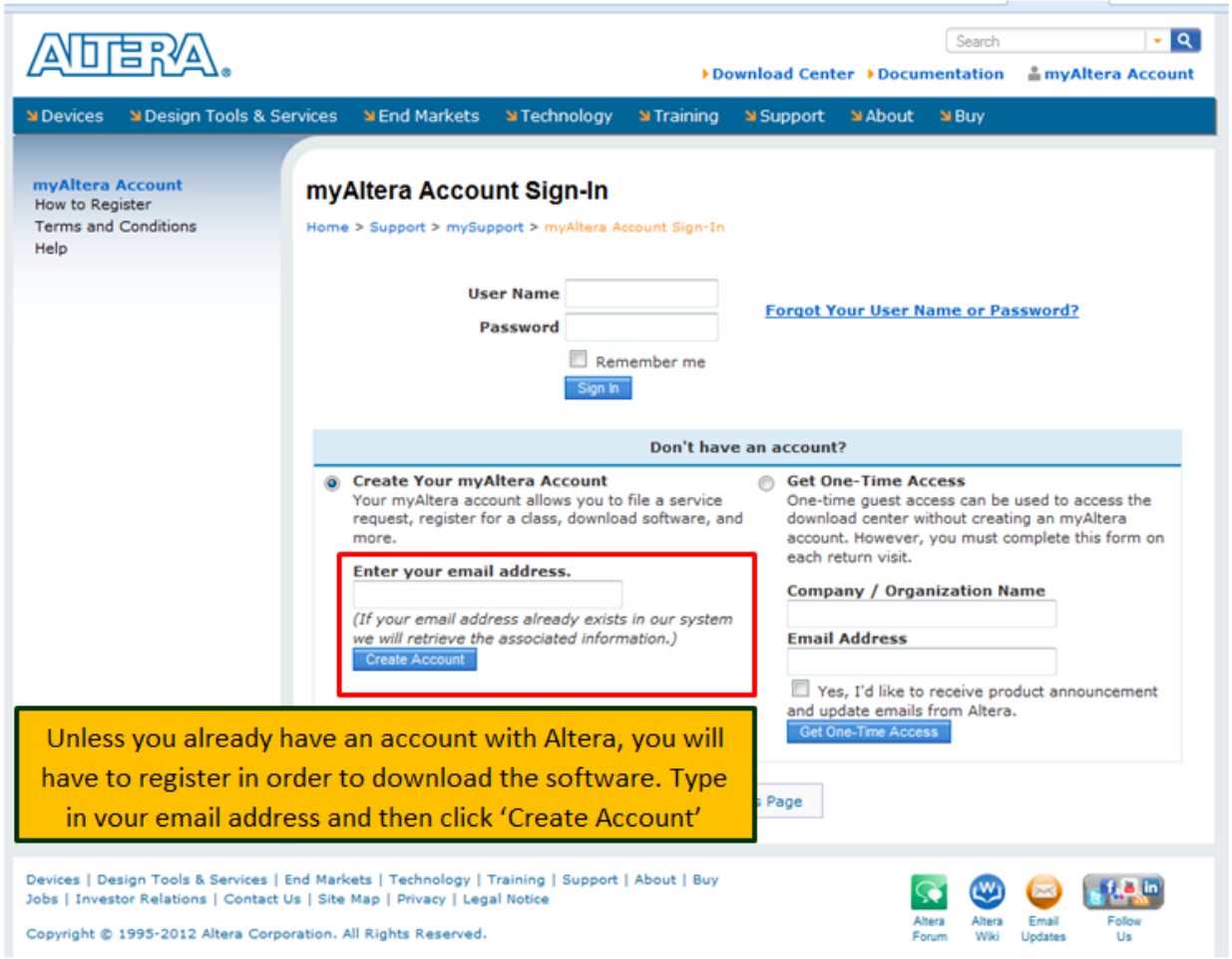
Under the ***EDA Netlist Writer Settings***, set the ***Format for output netlist:*** to Verilog HDL

Click ***Apply*** and exit the window by clicking ***OK***. Next you will need to set the EDA settings; go to the ***project navigator*** in Quartus and click on ***Tool*** then ***Options***.



Verify that both ***ModelSim*** and ***ModelSim-Altera*** directories are set in the folder where its executable file is located; location will vary based on where you installed your program. Click ***OK***.

C. Registering with ALTERA and getting Quartus II Web Edition



myAltera Account
How to Register
Terms and Conditions
Help

myAltera Account Registration

Create Your myAltera Account

(Note: Data must be entered in English)

* First Name
* Last Name
* Company Name
* Address
Address (Line 2)
* City
* Country: USA
* State (USA only)
* Zip / Postal Code
Email Address: KUZJRINX@YAHOO.COM (Edit Email)
* Telephone Number
Example: (408)544-7000x1234
Language Preference: English
* My primary job function is
* My preferred distributor is: No preference

*** For what end applications do you design? (check all that apply)**

- Academic/Research
- Automotive
- Computing and Office Automation
- Consumer Electronics or Digital Entertainment
- Digital Broadcast
- Industrial
- Medical Equipment
- Military or Aerospace
- Semiconductors
- Storage Systems
- Test and Measurement Equipment
- Wireless Communications
- Wireline Communications
- Other

*** Please check all topics that interest you:**

- CPLDs
- Digital signal processing
- Embedded processors
- HardCopy ASICs
- High-density FPGAs
- Low-cost FPGAs

*** When was the last time you designed with an Altera® product?**

- I'm currently designing with Altera products
- More than 1 year ago
- Never

- Yes, I'd like to receive product announcement and update emails from Altera.
- Yes, I'd like to receive the Inside Edge-Altera's Monthly eNewsletter.

*** Create User Name** (3-20 characters, no spaces)
*** Create Password** (Minimum 6 characters)
*** Confirm password**

* I agree to the Terms of Service
Remember me
Create Account

* Indicates required field.

Rate This Page

Please give us feedback

D. Parts List

Your checklist for parts includes:

- Mini-USB Cable
  http://www.amazon.com/C2G-Cables-28107-Mini-B-Y-Cable/dp/B0013LQD7Y/ref=sr_1_1?ie=UTF8&qid=1347393880&sr=8-1&keywords=mini+usb+cable+shield+double+power

- Amani GTX
  http://amani64.com/store/index.php?main_page=product_info&cPath=2&products_id=11&zenid=us5tlhj5j6h3qdpbbonvk6cuk4

- PST-1200UF -- Universal AC/DC Adaptor with USB Output
  http://www.x1up.com/.sc/ms/dd/ee/125