Electrical Engineering Department

This tutorial provides step by step instructions for the installation procedure for the Xilinx ISE V14.2 design environment for FPGA synthesis and implementations. In addition, it goes through a complete synthesis and board programming example based on the NEXYS 2 prototyping board from Digilent Inc.

A) Installation Steps:

1. Download the XILINX ISE tool from the internet (~6GB). And Start the installation process. You will get the window shown in Figure 1A. Choose ***ISE WebPack***.

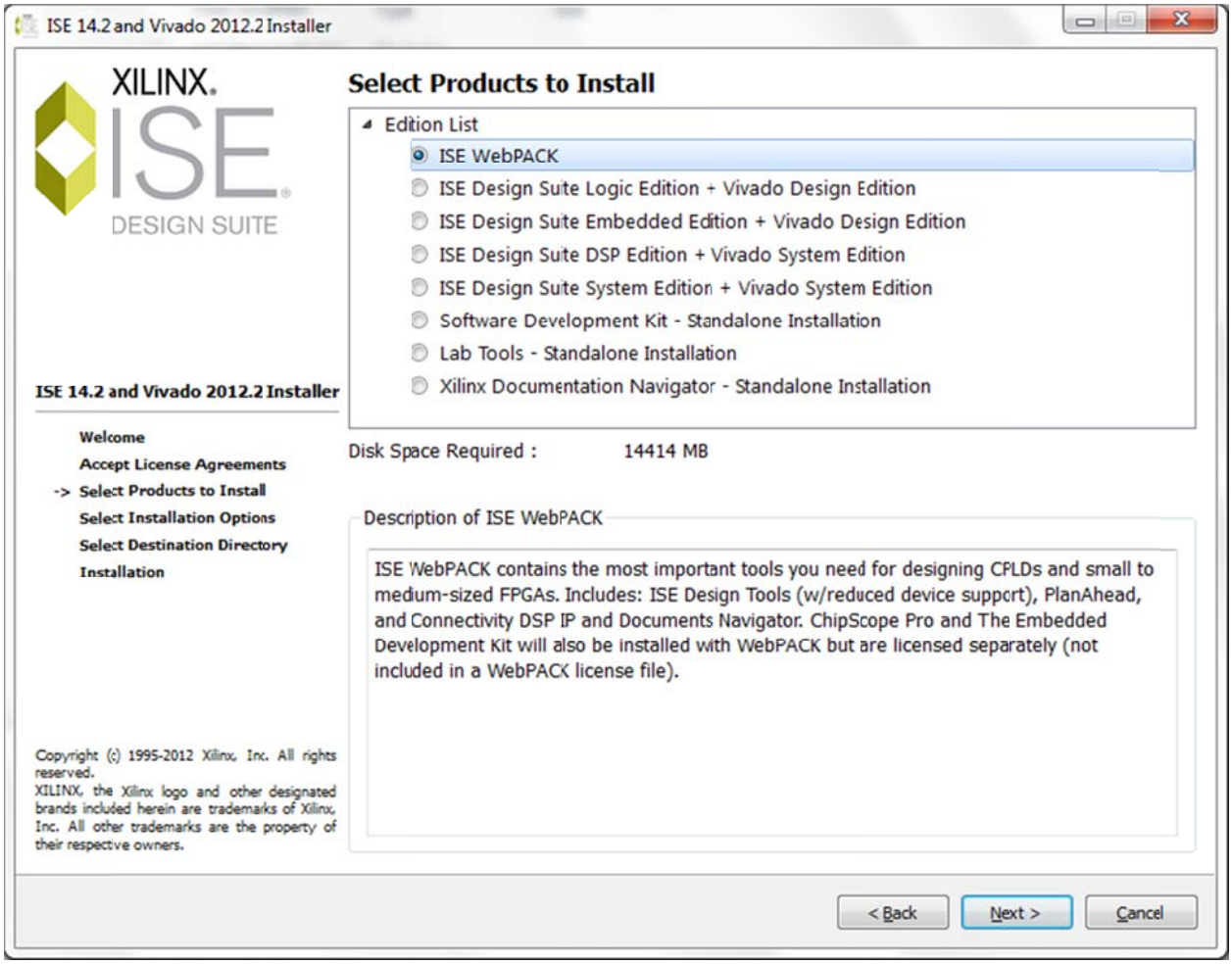

Figure 1A

Dr. Mohammad S. Sharawi

2. Select the tools to be installed as shown in Figure 2A and the destination location as in Figure 3A.

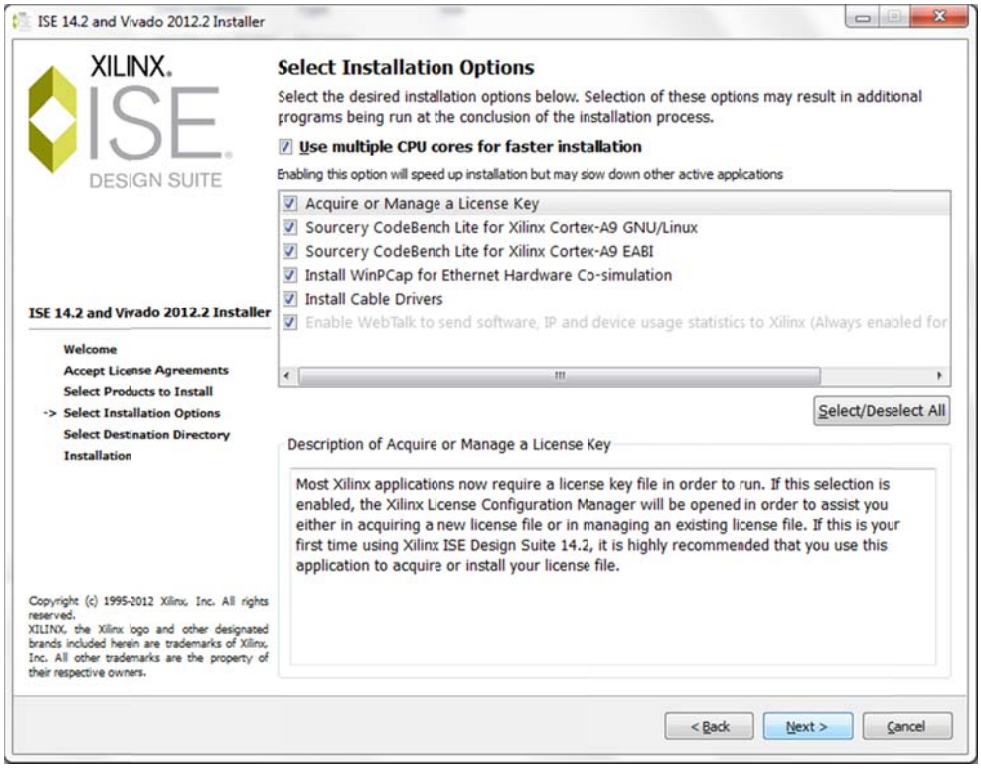

Figure 2A

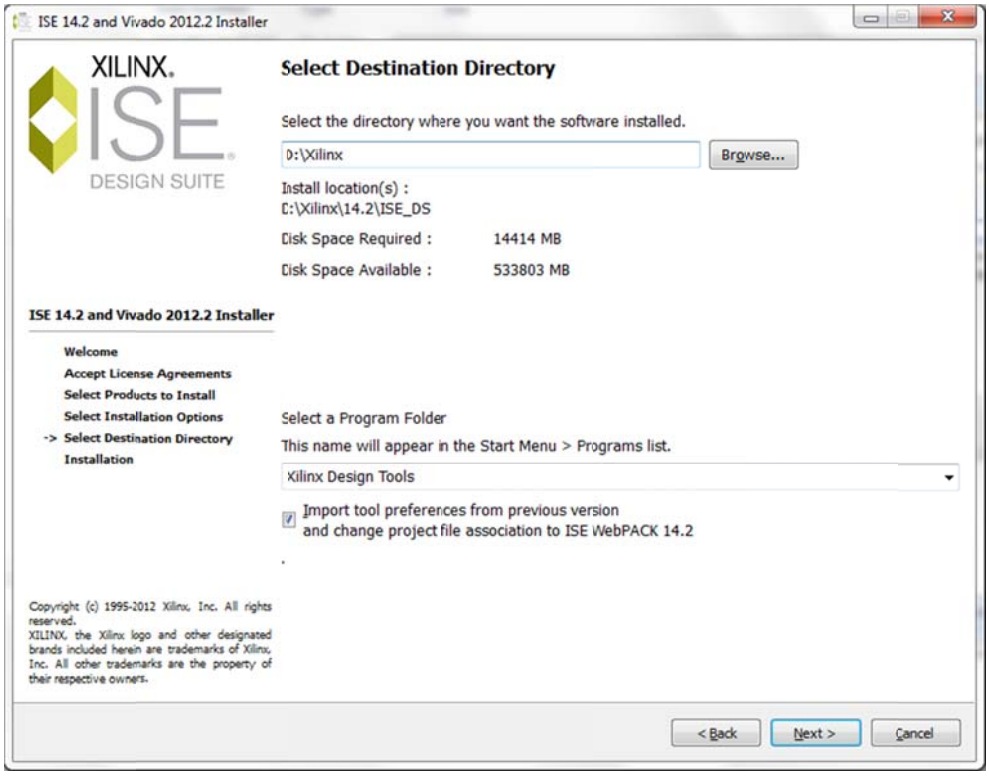

Figure 3A

3. The installation process will start as shown in Figure 4A. Allow the various drivers and packages to be installed once prompted during the last portion of the installation. Choose the ***Free Web Pack License*** as shown in Figure 5A, and you have to register through Xilinx website to get a valid license.

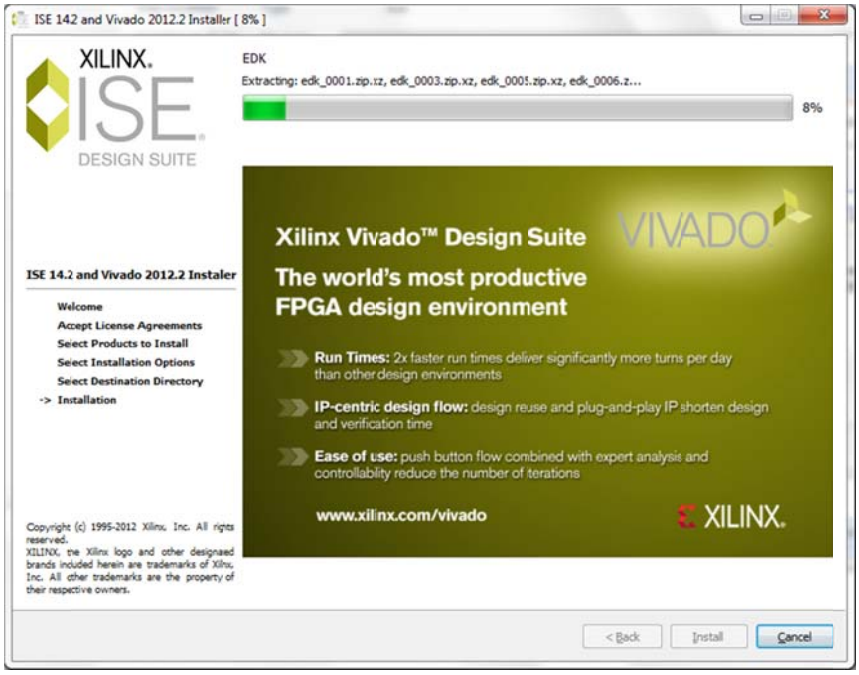

Figure 4A

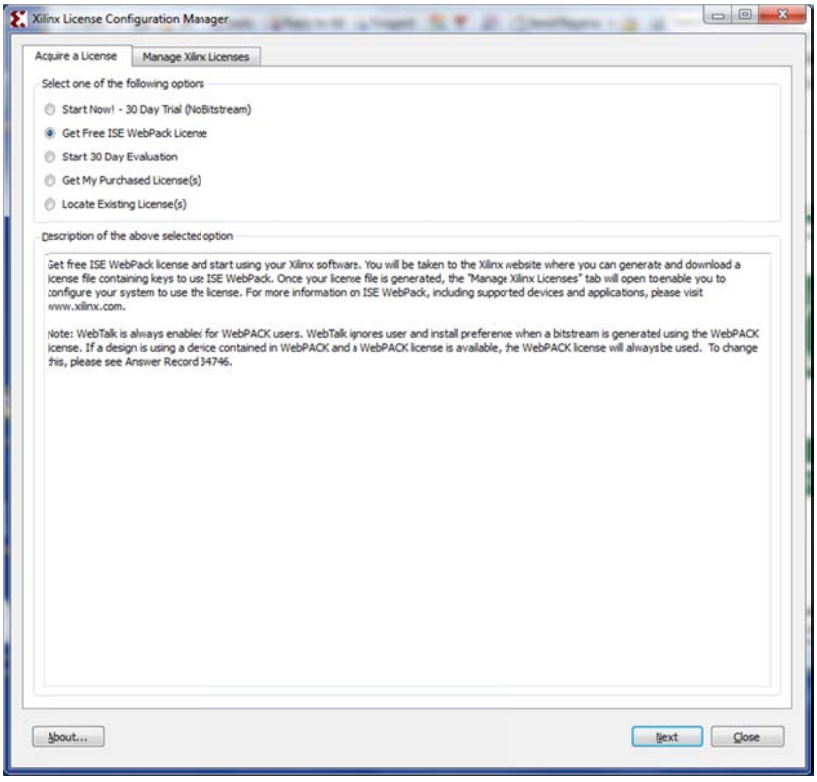

Figure 5A

4. Register for a user account on Xilinx website to get a valid license. Choose the option shown in Figure 6A. You will get a license file via email for your machine.
   Save the license file in a specific folder, and then click on copy button in Figure 7A, and direct the tool to the location of the saved license file.

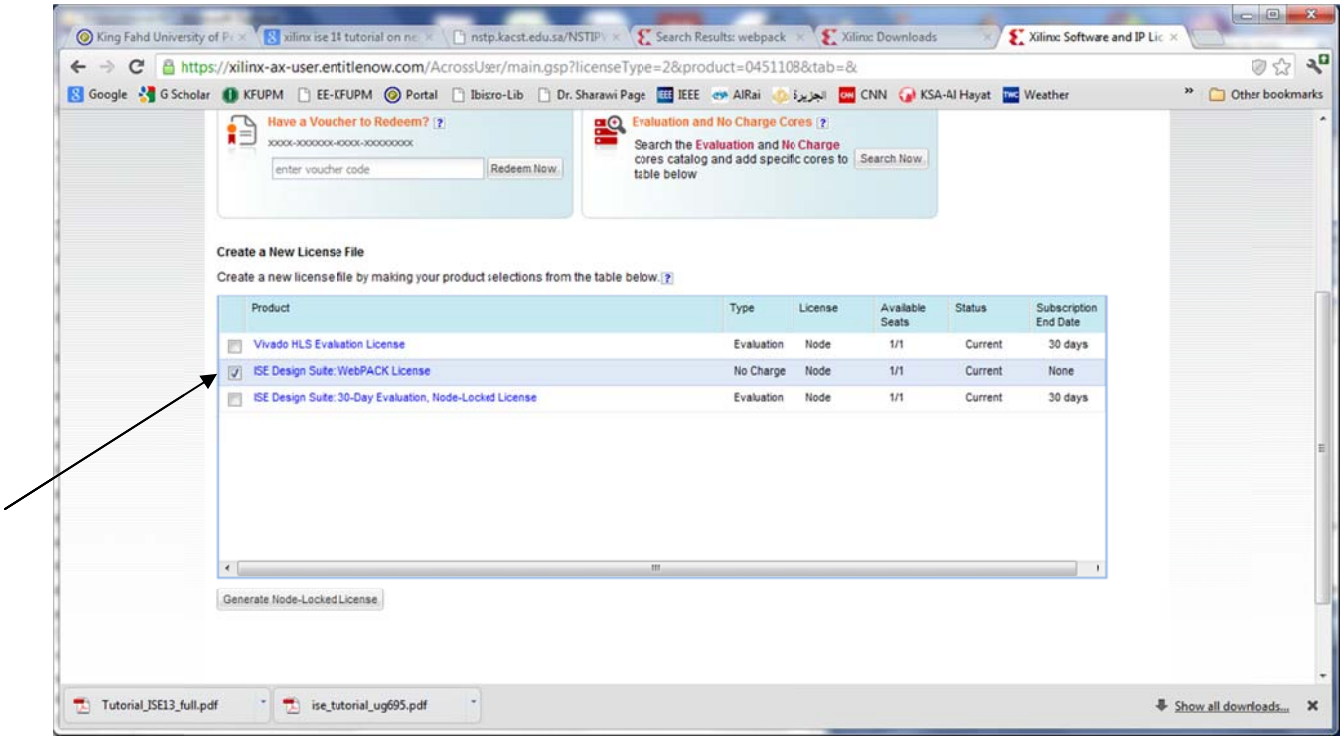

Figure 6A

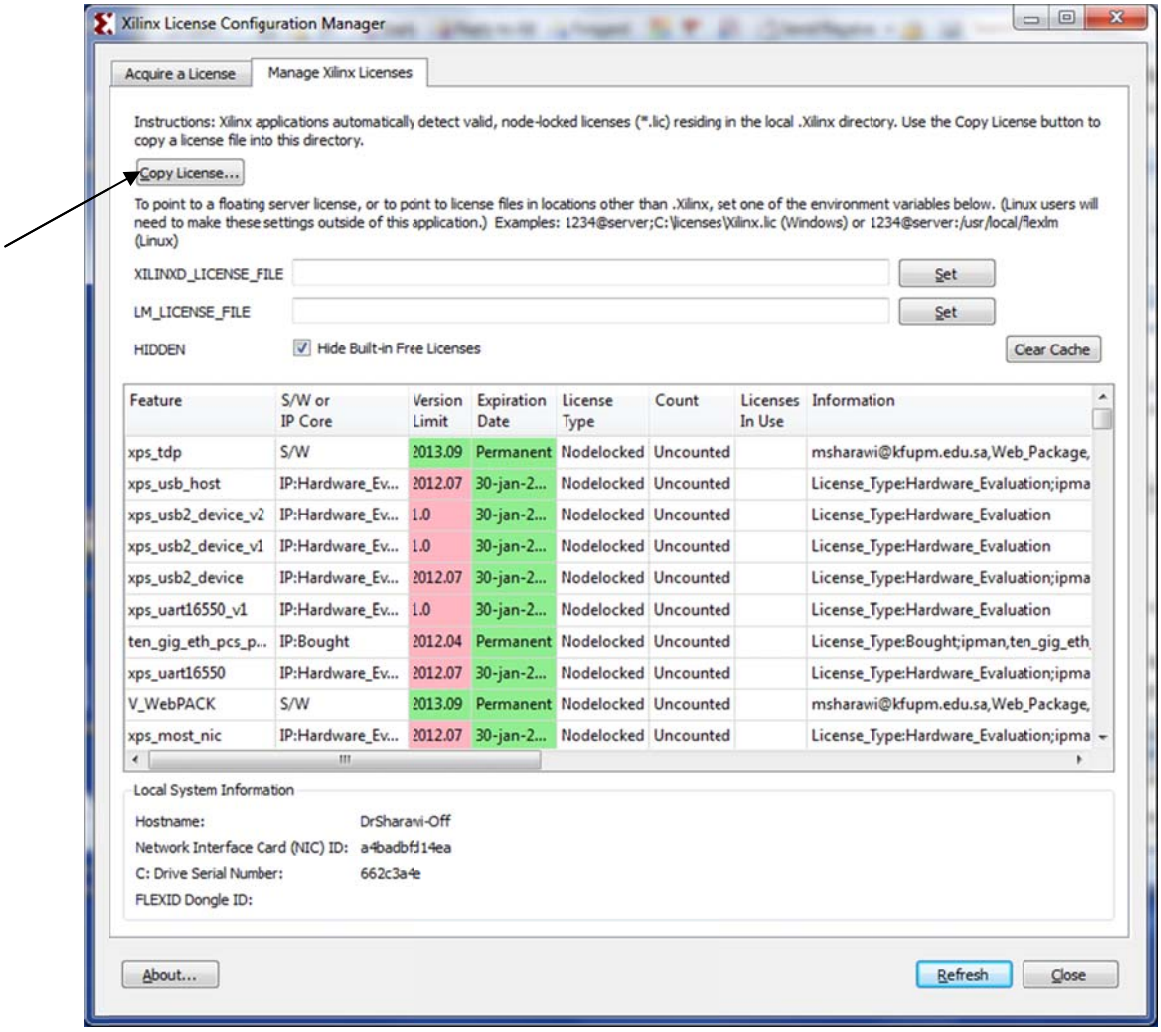

Figure 7A

5. You should the message “License Installation Successful”.

   Your Installation is COMPLETE NOW!

B) Synthesizing a Design:

In this section, we will go over the complete design steps to synthesize and program the Verilog code/module into the Spartan 3E FPGA on the NEXYS 2 Digilent Board.

1. Start the XILINX ISE Design Environment. To create a new design, select new project as shown in Figure 1B.

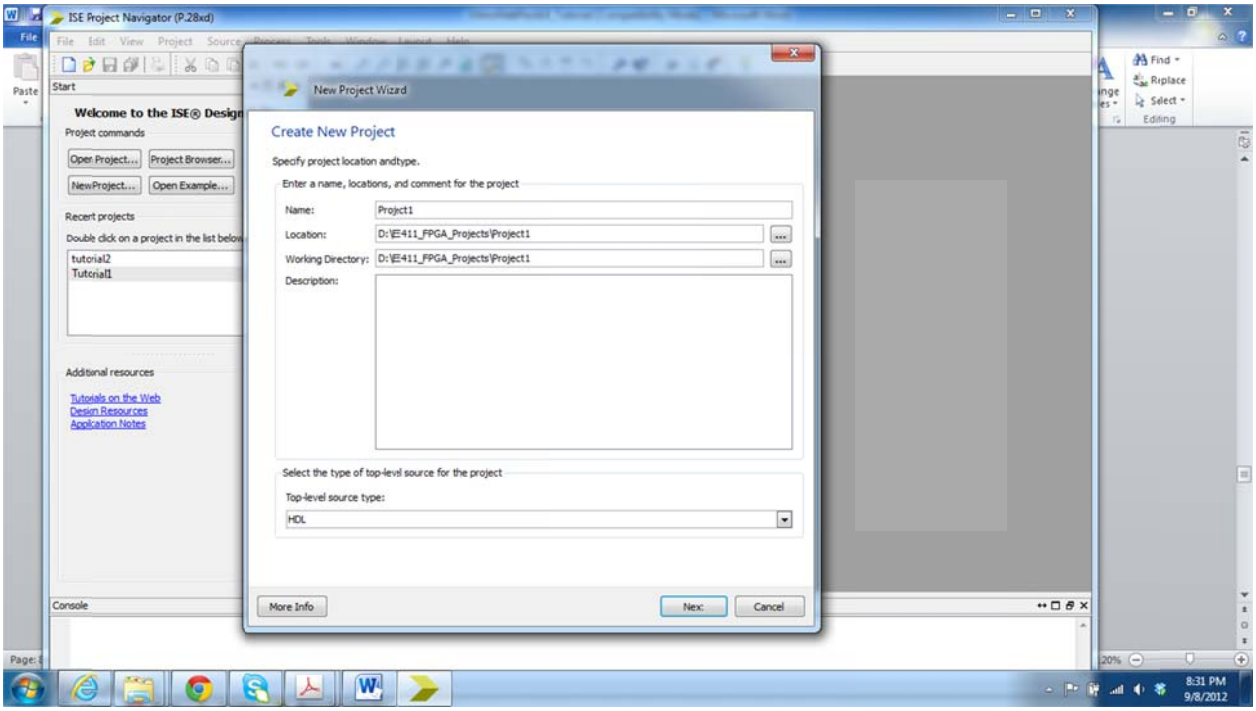

Figure 1B

2. You need to select the appropriate settings for your FPGA device. For the NEXYS 2 board, we have an XC3S1200E chip on the board. Thus we choose the options in Figure 2B. Then you click ***finish***.

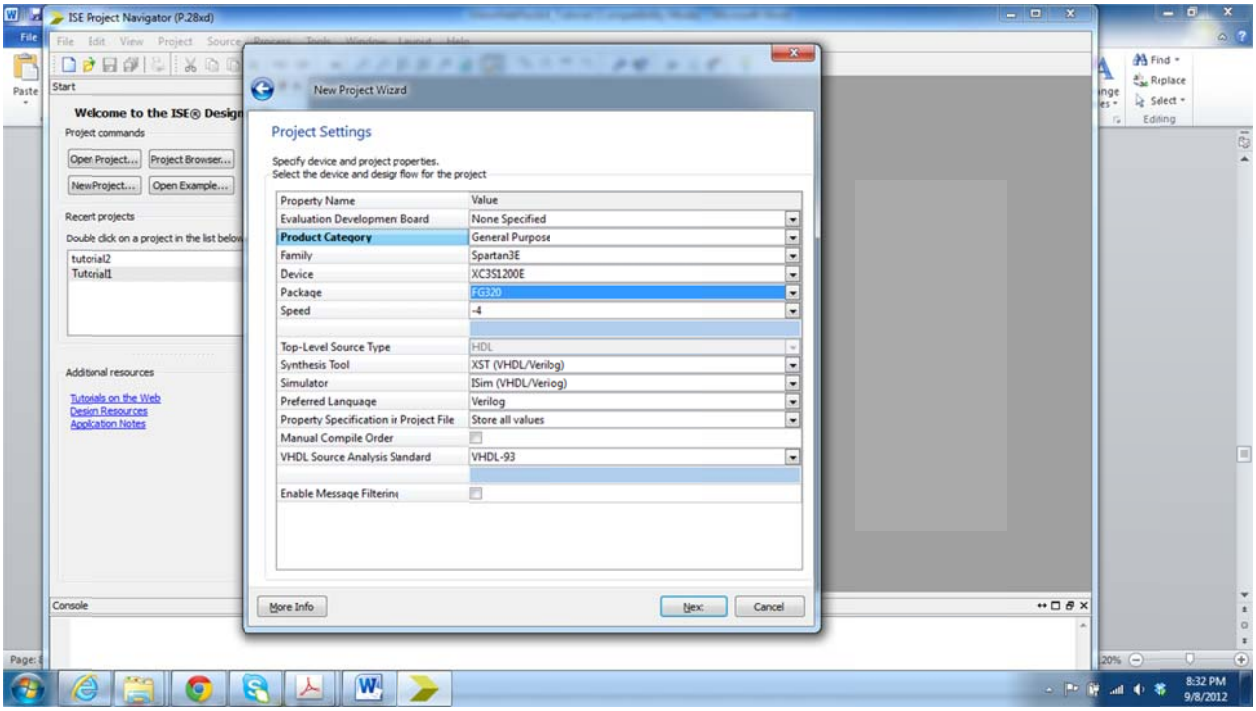

Figure 2B

3. Now you can add your design files to the compiler. Or you can create new ones. We will create a new simple Verilog design for a simple logic function with 4 variables and two output single bit Boolean functions. The functions are:

   F1 = a & b & c;

   F2 = ~a | d;

4. We will add a Verilog module, and call it ***function1.v*** to the chip. Right click on the chip, and choose ***New Source wizard***, then select a ***Verilog module*** as shown in Figure 3B.

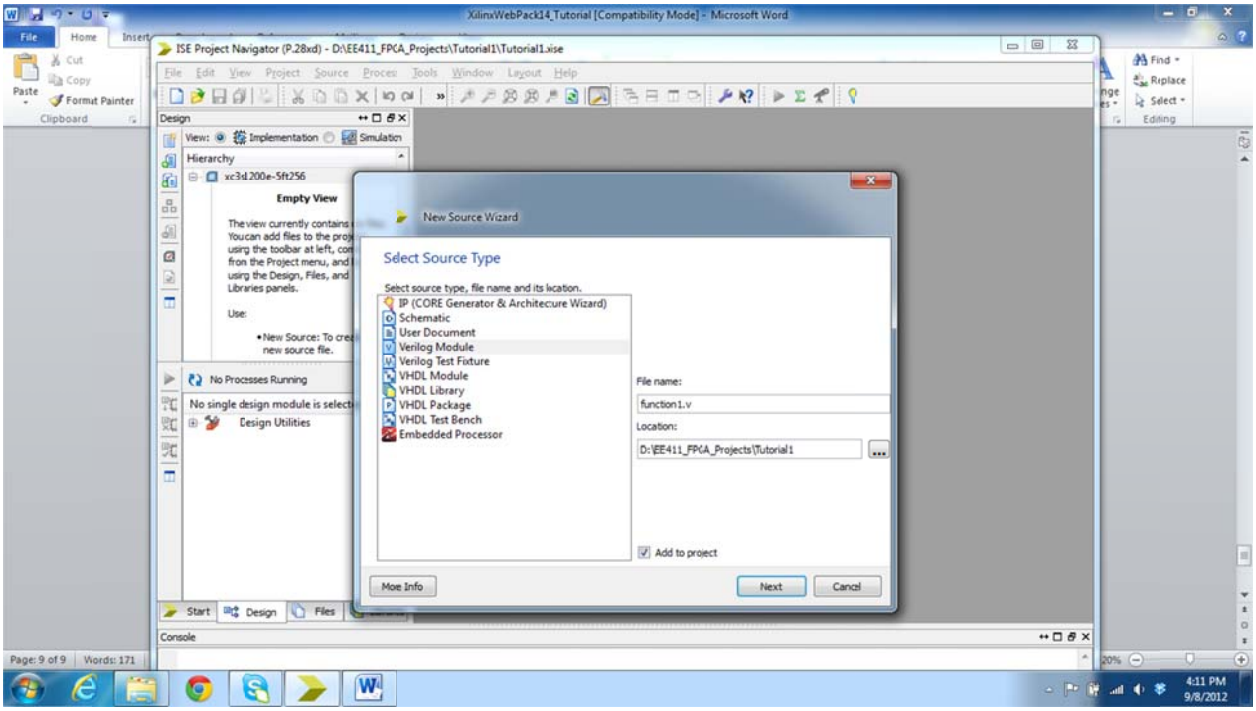

Figure 3B

5. We define the input and output ports of the Verilog module as shown in Figure 4B.

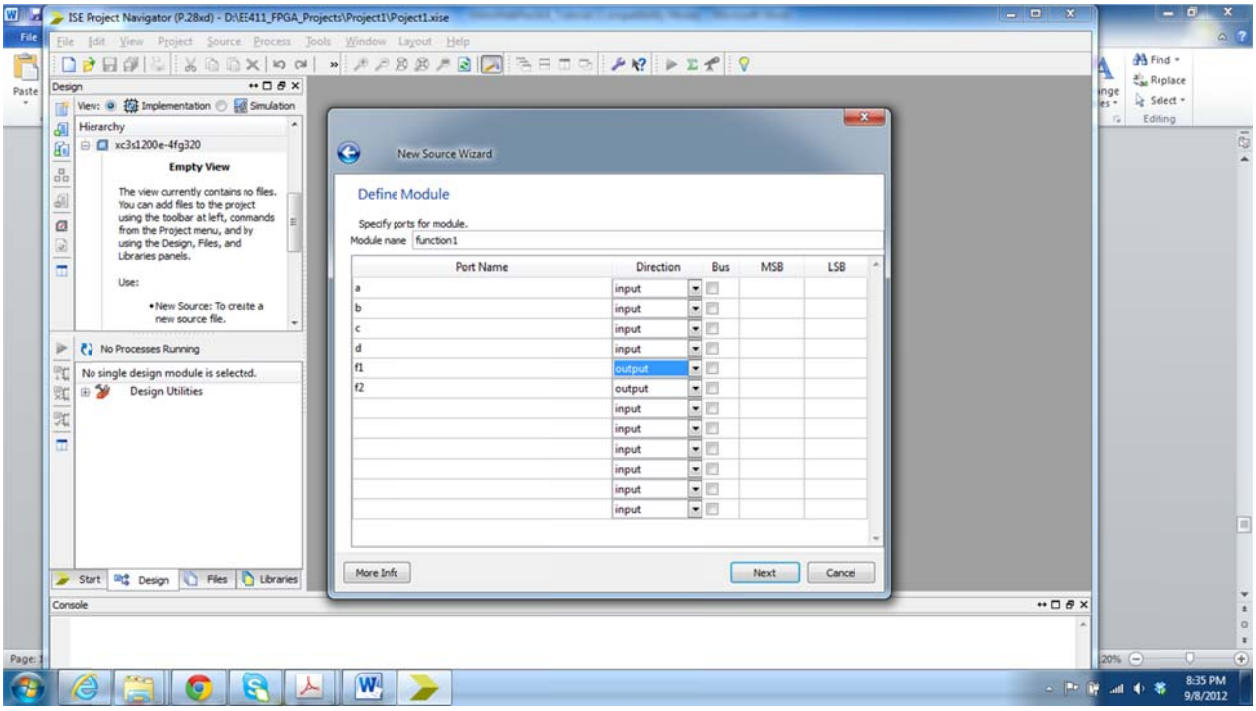

Figure 4B

6. The module structure/Verilog code is automatically generated by the tool. Then we enter the relationships within the module according to our functions as shown in Figure 5B. Save the Verilog module created.

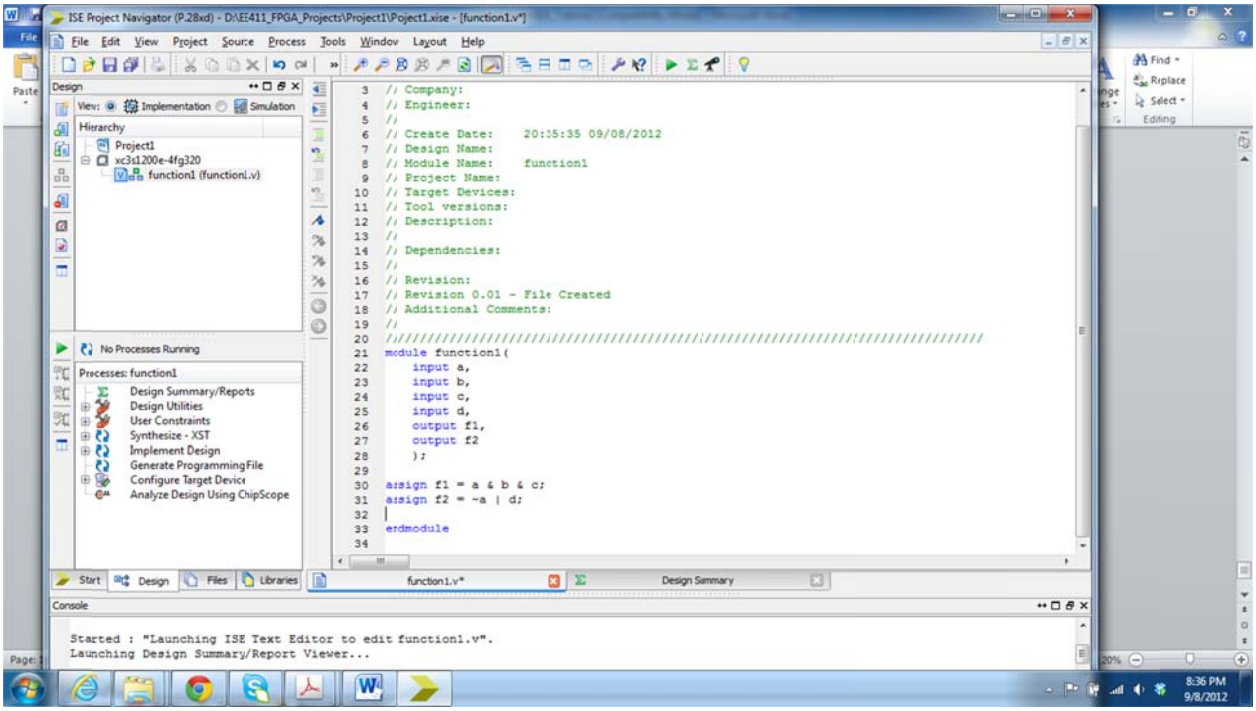

Figure 5B

7. Now you need to check your code syntax for errors. While choosing the Verilog module "function1.v", expand the ***Synthesize-XST*** menu in the ***Processes*** window as shown in Figure 6B, then double click on ***Check Syntax***. A check will appear if your code is error free.

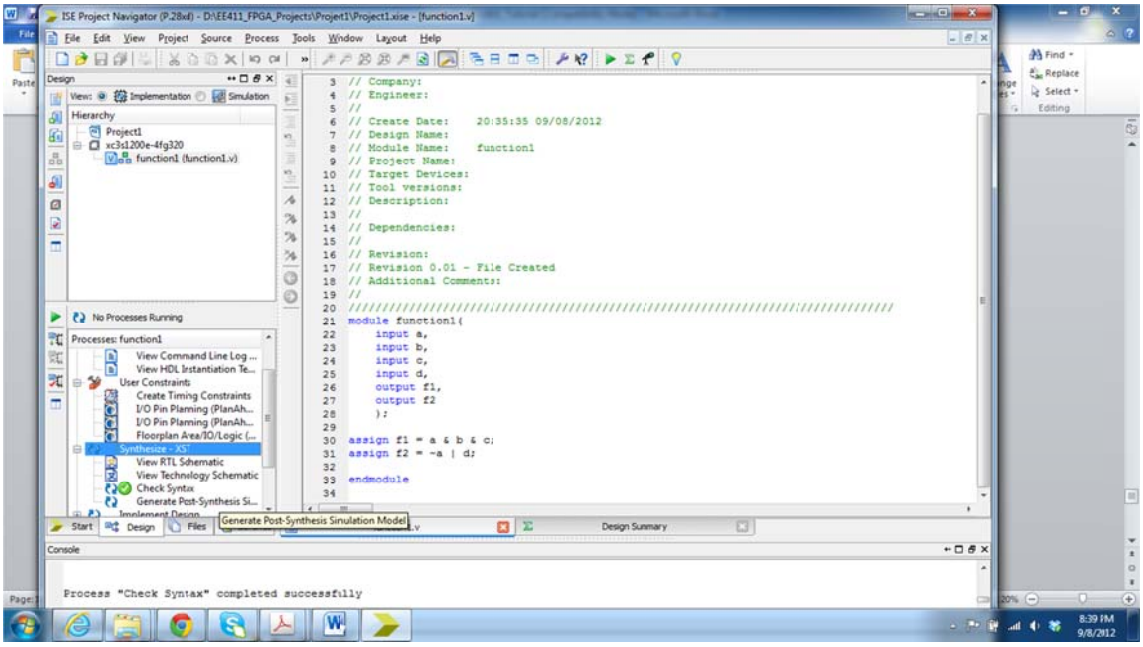

Figure 6B

8. Double click on **Synthesize-XST** to Synthesize the design (generate netlist**)**. You should get a check if the synthesis is ok as shown in Figure 7B.

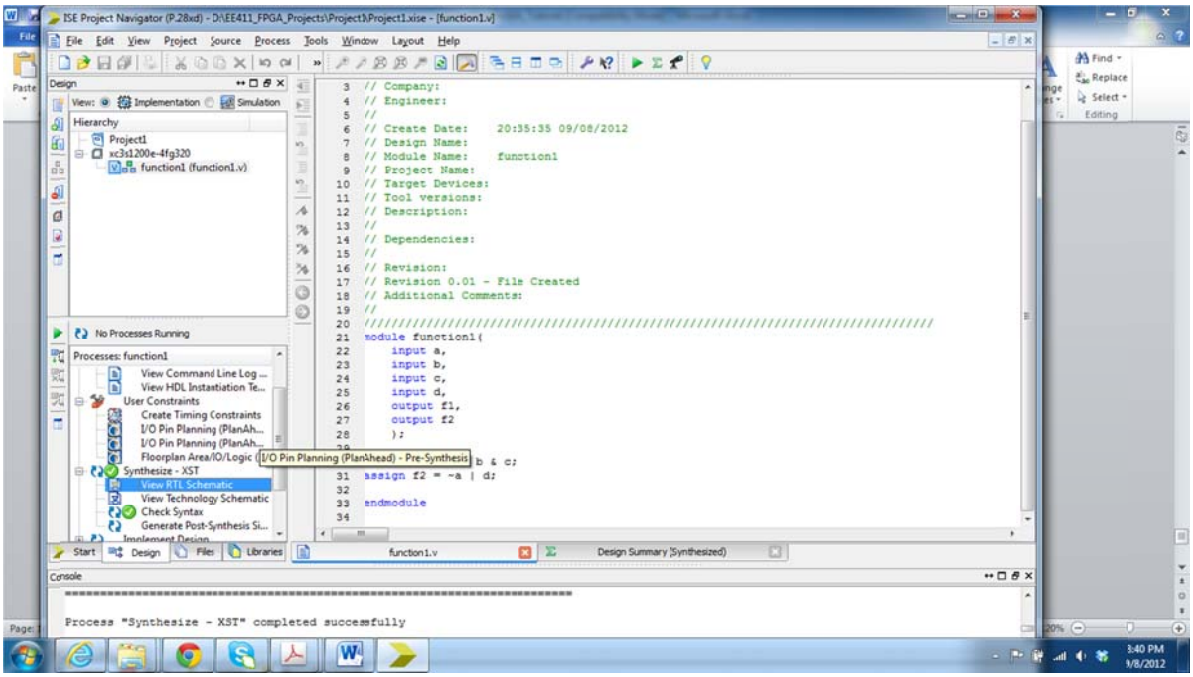

Figure 7B

9. Now we need to assign pins to the input and output ports of the module. This is done using ***the I/O Pin Planning*** tool under the ***User Constraints***. Once you double click on ***the I/O Pin planning***, the tool will ask you for a UCT file. Click ok on the message below shown in Figure 8B.

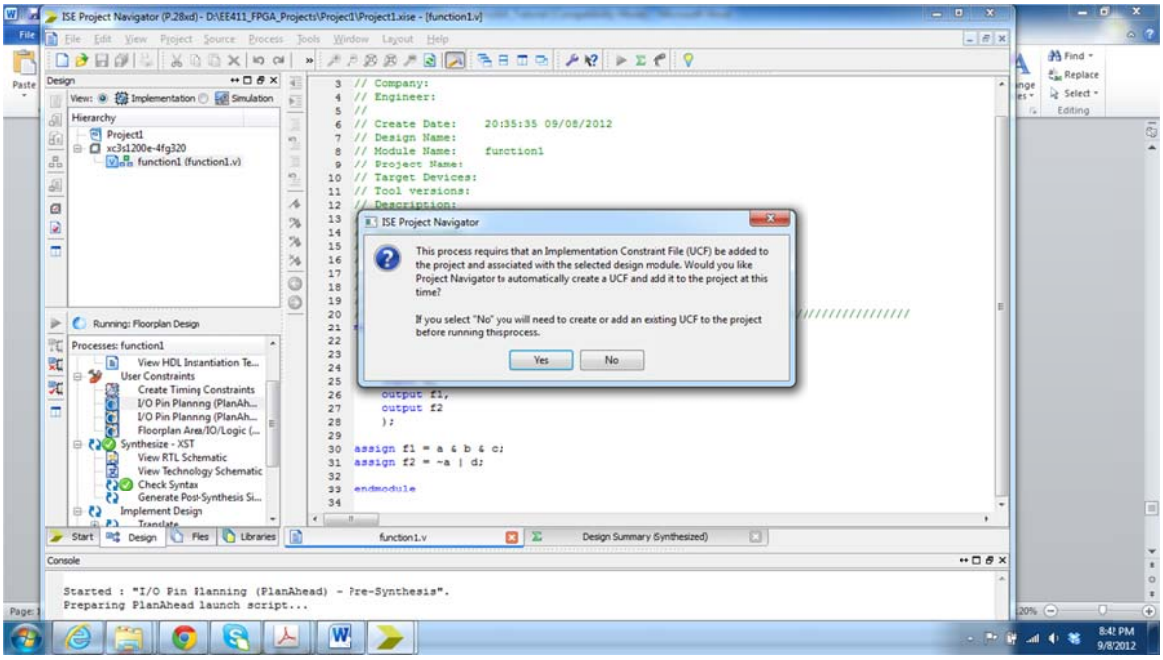

Figure 8B

10. The ***Plan Ahead tool*** for pin assignment and routing on the FPGA opens up. It looks like in Figure 9B.

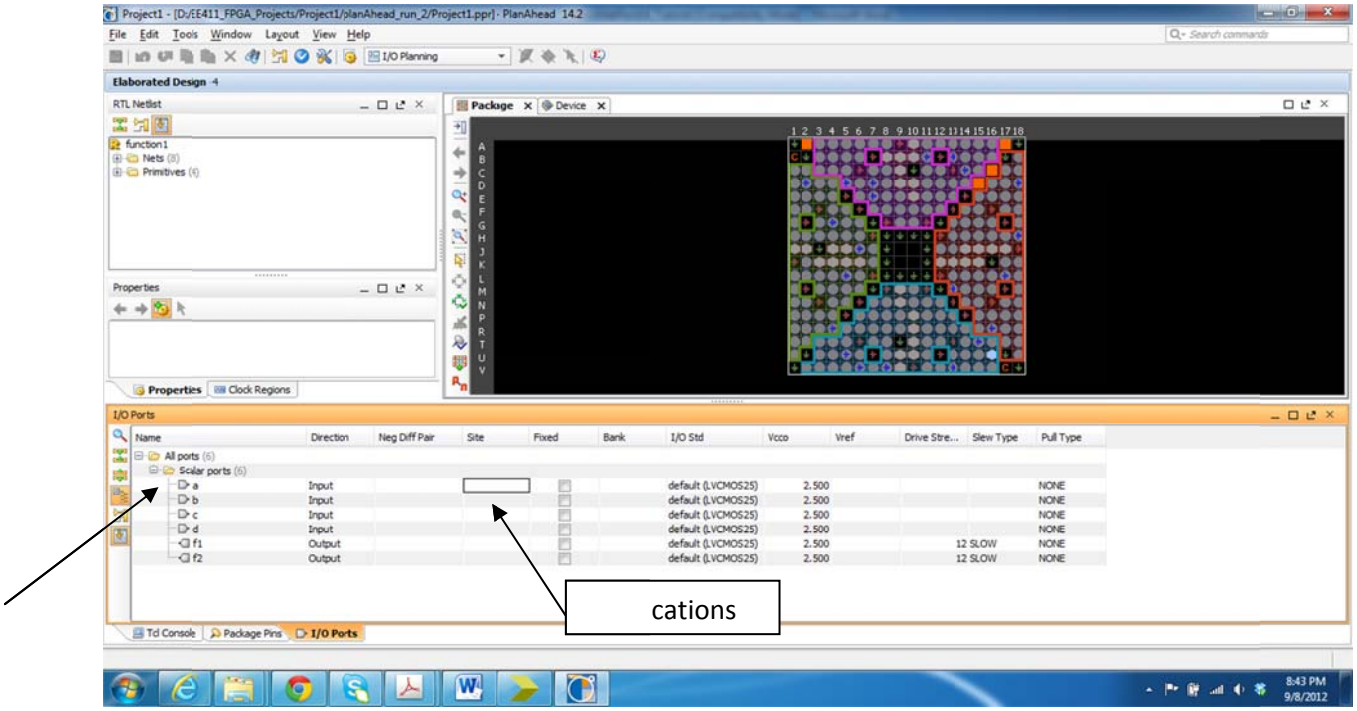

Figure 9B

11. Expand the Scalar Ports folder to see all the input/output ports you have in your top level module. There are two ways to assign the pins, either you drag to the pin location on the FPGA floorplan, or you can write the pin location next to each port. We will assign the ports to the pins needed according to the ports needed on the FPGA Digilent board. These boards have some predefined connections to Switches, Push buttons, LEDs, and seven segments displays. Check the pins you need from the board data sheet. A board picture is shown in Figure 10B.

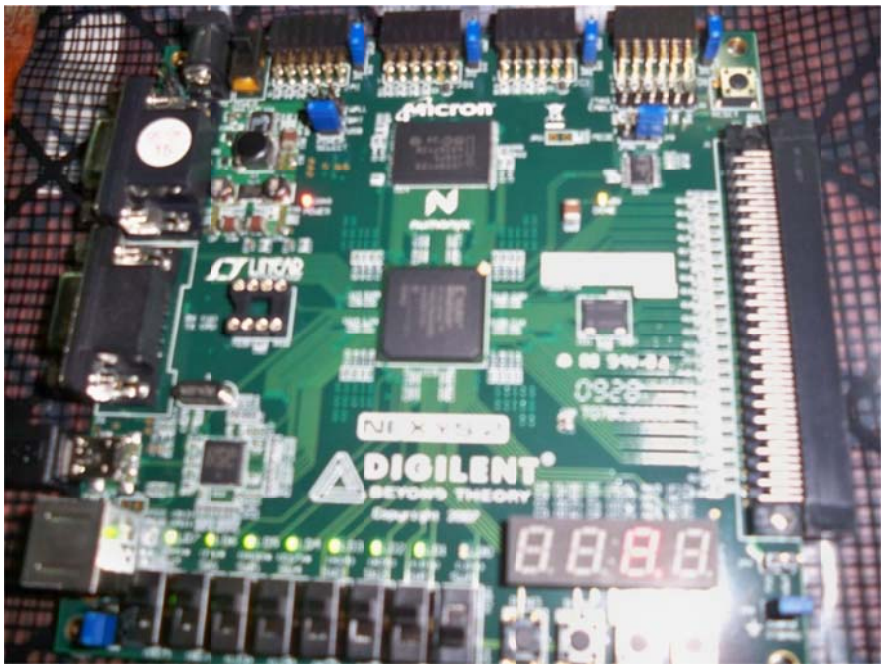

Figure 10B

12. In this small project we will assign the following:

a : R17 b : N17 c : L13 d : L14

f1 : J14 f2 : J15

The final table will look like the one in Figure 11B.

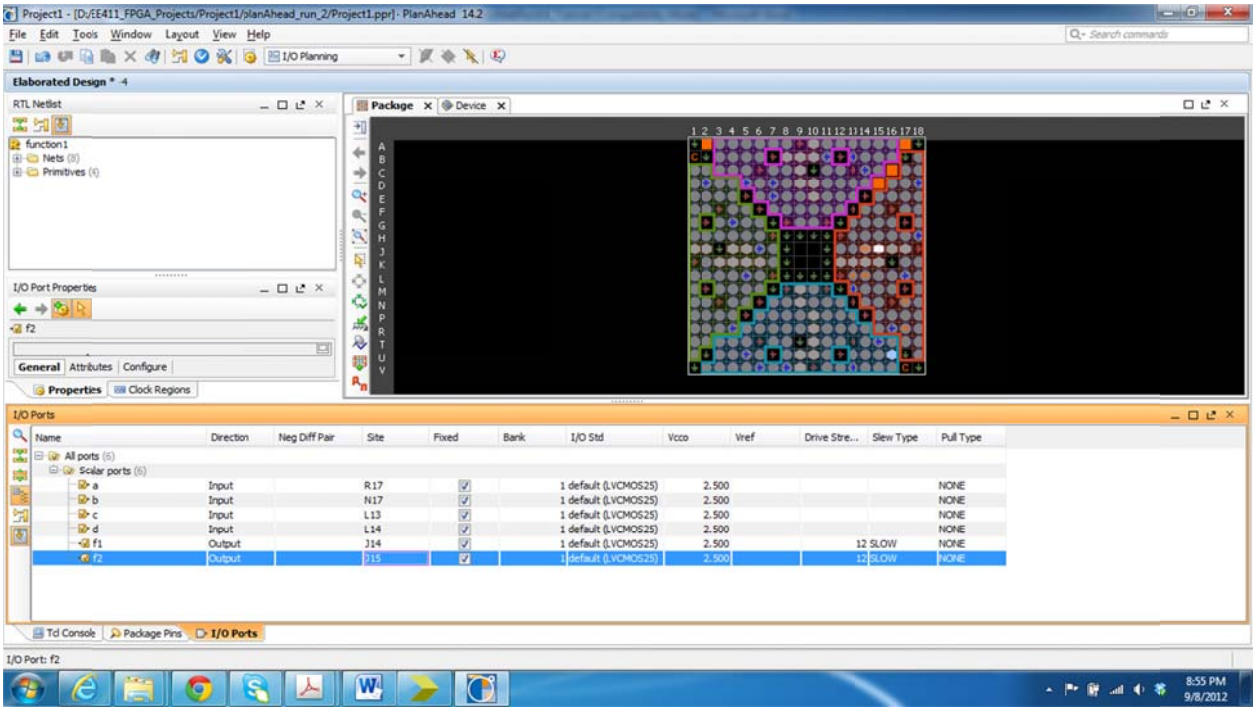

Figure 11B

13. Save your constraint/pin assignment file. Close the Plan Ahead tool.

Now double click on the ***Implement Design*** and wait until you get checks on all its sub-items as shown in Figure 12B.

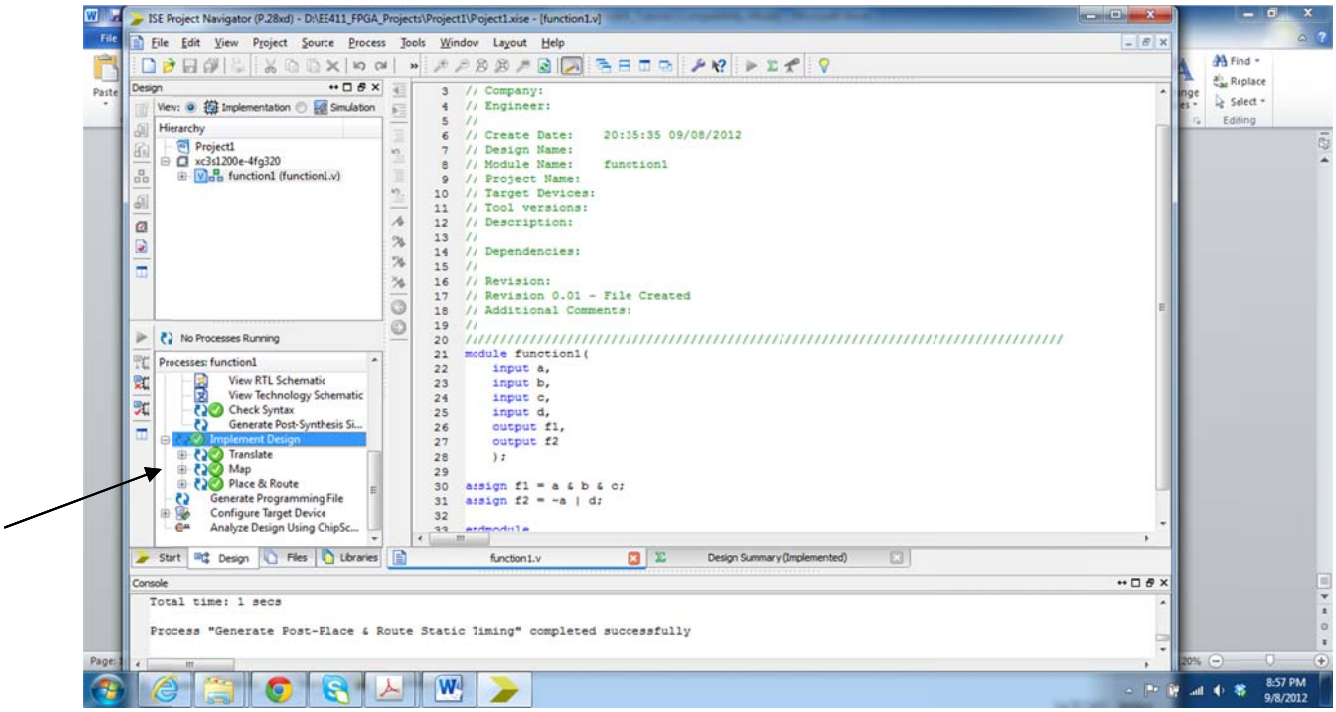

Figure 12B

14. Double click on ***Generate Programming File*** item, and wait until you get a check there to have a successful file.

    Then double click on ***Generate Target PROM/ACE File*** and then click on the message shown in Figure 13B.

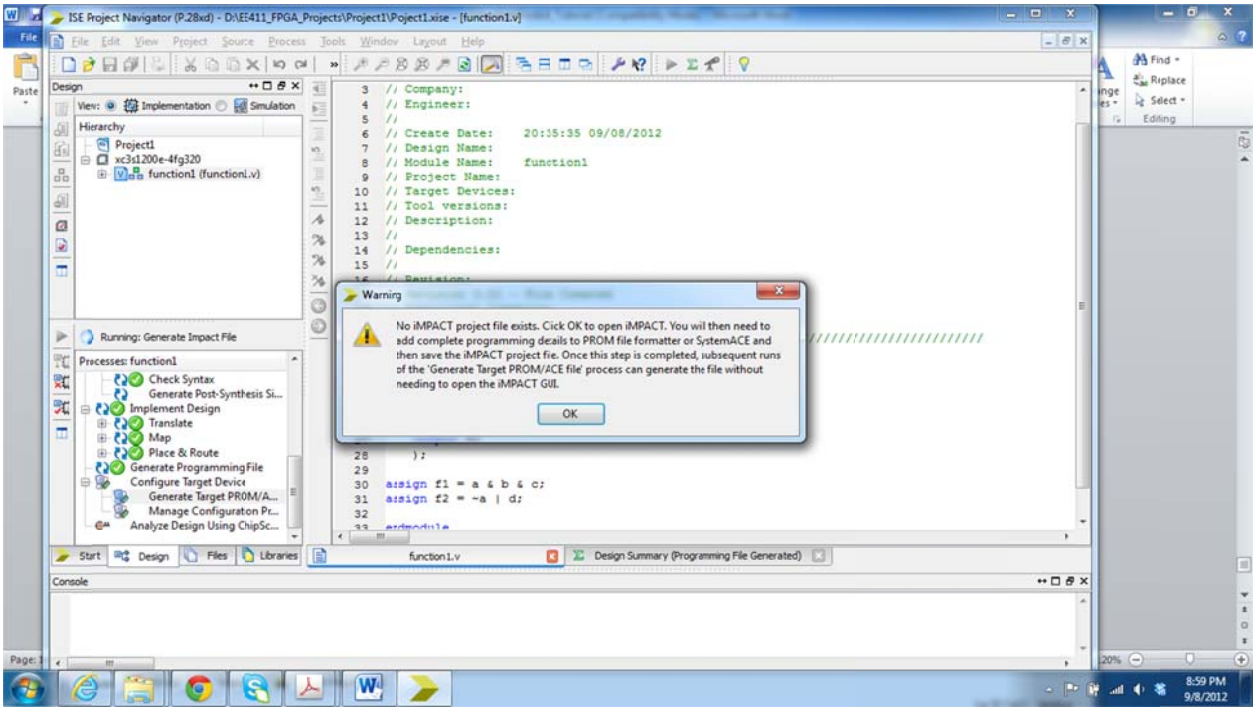

Figure 13B

15. Now, double click on ***Boundary Scan*** and then right click and choose ***Initialize Chain*** as shown in Figure 14B.

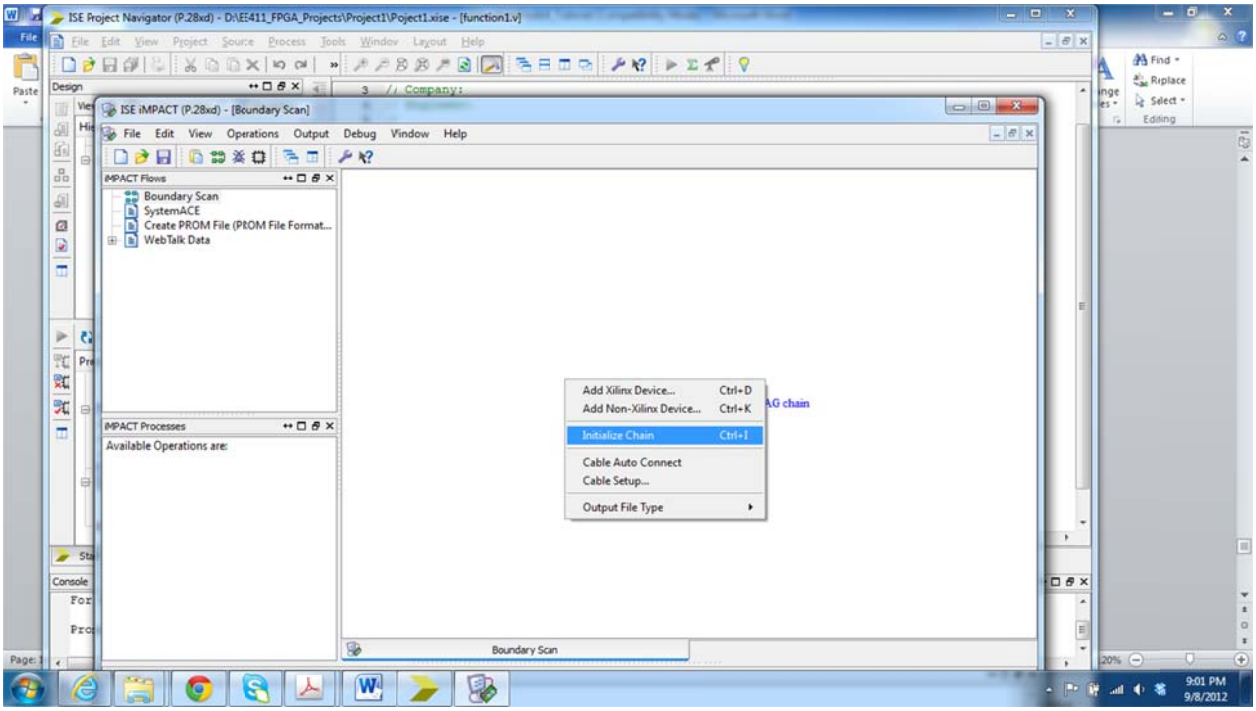

Figure 14B

16. Click ***yes*** to continue with configuration file assignment as shown in Figure 15B. Choose function1.bit as shown in Figure 16B.

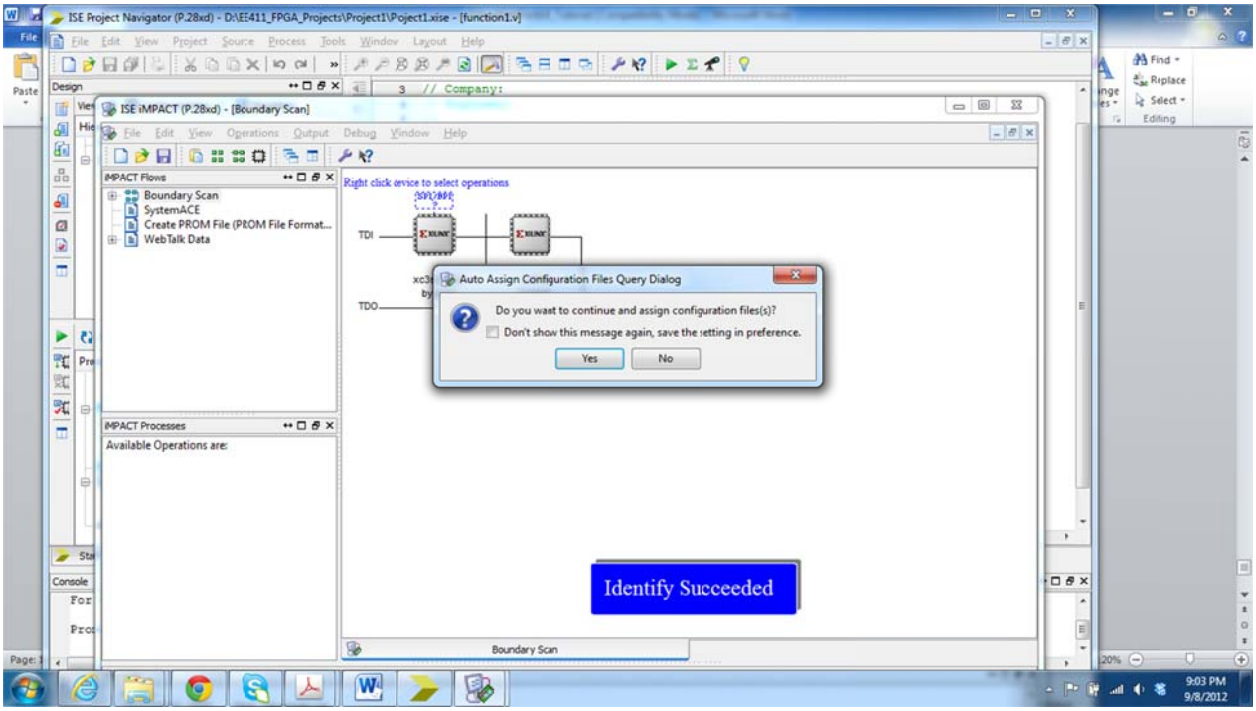

Figure 15B

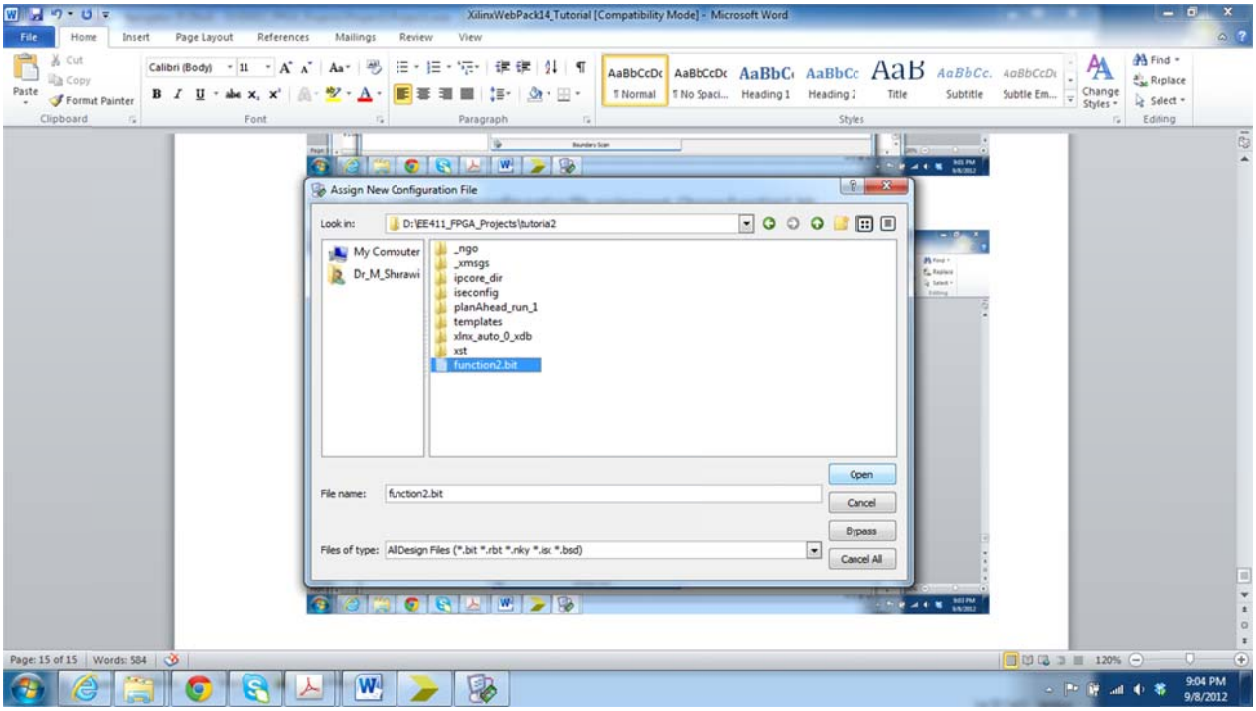

Figure 16B

17. At this time, we are not using the Flash, thus click ***NO*** on the message shown in Figure 17B. And then ***Cancel*** on the following window.

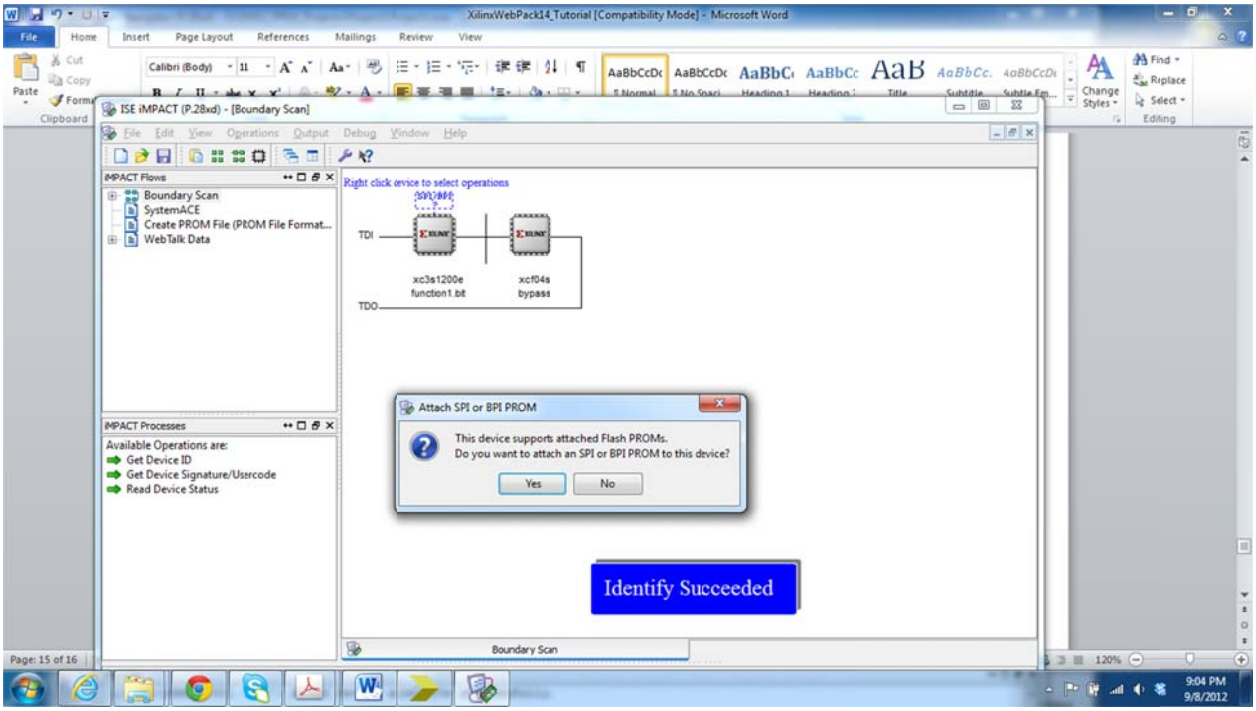

Figure 17B

18. Then, you need to choose the FPGA chip that has the number ***XC3S1200E,*** Click Apply → ***OK***. This is shown in Figure 18B. Click on the FPGA until it becomes **GREEN**, then right click and Choose ***Program***. And if your program is well implements on the board, you will get a ***successful program.***

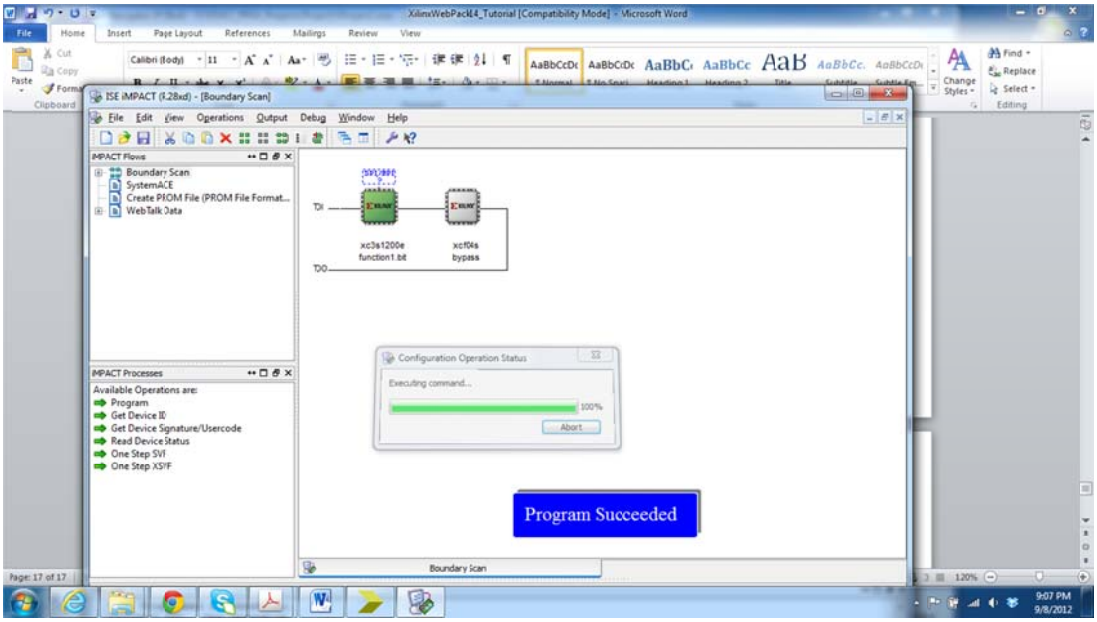

Figure 18B

19. When the switches A, B and C are ON, LD0 turn ON as shown in Figure 19B. This is it!

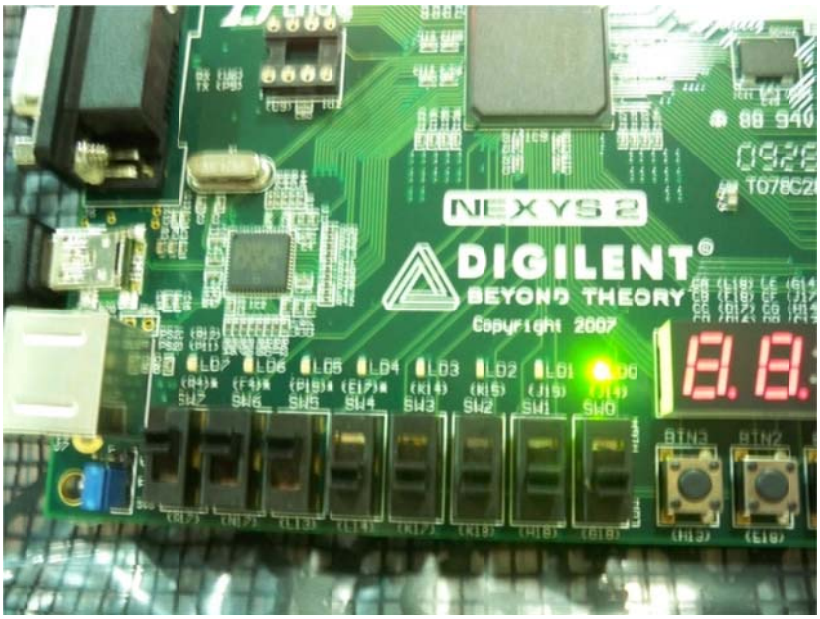

Figure 19B

THE END