lAB aSSIGNMENT #1 - mux

Due 2/13/2022

Christopher Lall

CSc 342/343 – Professor Gertner

Table of Contents

[Section 1: Objective: 2](#_Toc95674111)

[Section 2: Tutorial Screenshots: 2](#_Toc95674112)

[Section 3: 2:1 Mux Screenshots 13](#_Toc95674113)

[Section 3.1: 2:1 Mux VDHL Code 15](#_Toc95674114)

[Section 3.2: 2:1 Mux VDHL Code Compilation 15](#_Toc95674115)

[Section 4: 2:1 MUX 32bit VHDL code 16](#_Toc95674116)

[Section 4.1 Mux 32bit VDHL Code Compilation 18](#_Toc95674117)

[Section 5: Explanation 18](#_Toc95674118)

[Section 6: Conclusion 19](#_Toc95674119)

# Section 1: Objective:

The objective of this lab is to familiarize myself with Quartus Prime Lite and create 2:1 multiplexer where each signal is one bit and 2:1 multiplexer where each signal is 32 bits and the selector signal is one bit.

# Section 2: Tutorial Screenshots:

Graphical user interface, application

Description automatically generated Following the tutorial pdf, we can learn how to setup and become familiar with Quartus.

Figure 1. Opening Screen of Quartus

Now, we can get familiar with the file menu that will help us navigate:

Graphical user interface, text, application, email

Description automatically generated

Figure 2. File Menu

From here we can create a directory.

Graphical user interface, text, application, email

Description automatically generated

Figure 3. Creating Directory

Since we want to create a directory, we click next and in figure 4, below, we click yes to confirm.

Graphical user interface, text, application

Description automatically generated

Figure 4. Confirmation of creating directory

Graphical user interface, text, application, email

Description automatically generated

Figure 5. Selecting Project Type

Once we click next we arrive at figure 6.

Graphical user interface, text, application, email

Description automatically generated

Figure 6. Adding Files

Graphical user interface, application

Description automatically generated

Figure 7. Choosing Family, Device, and Board Settings

Text

Description automatically generated with low confidence

Figure 8. EDA Tool Settings

Graphical user interface, text, application

Description automatically generated

Figure 9. Summary

Graphical user interface, application

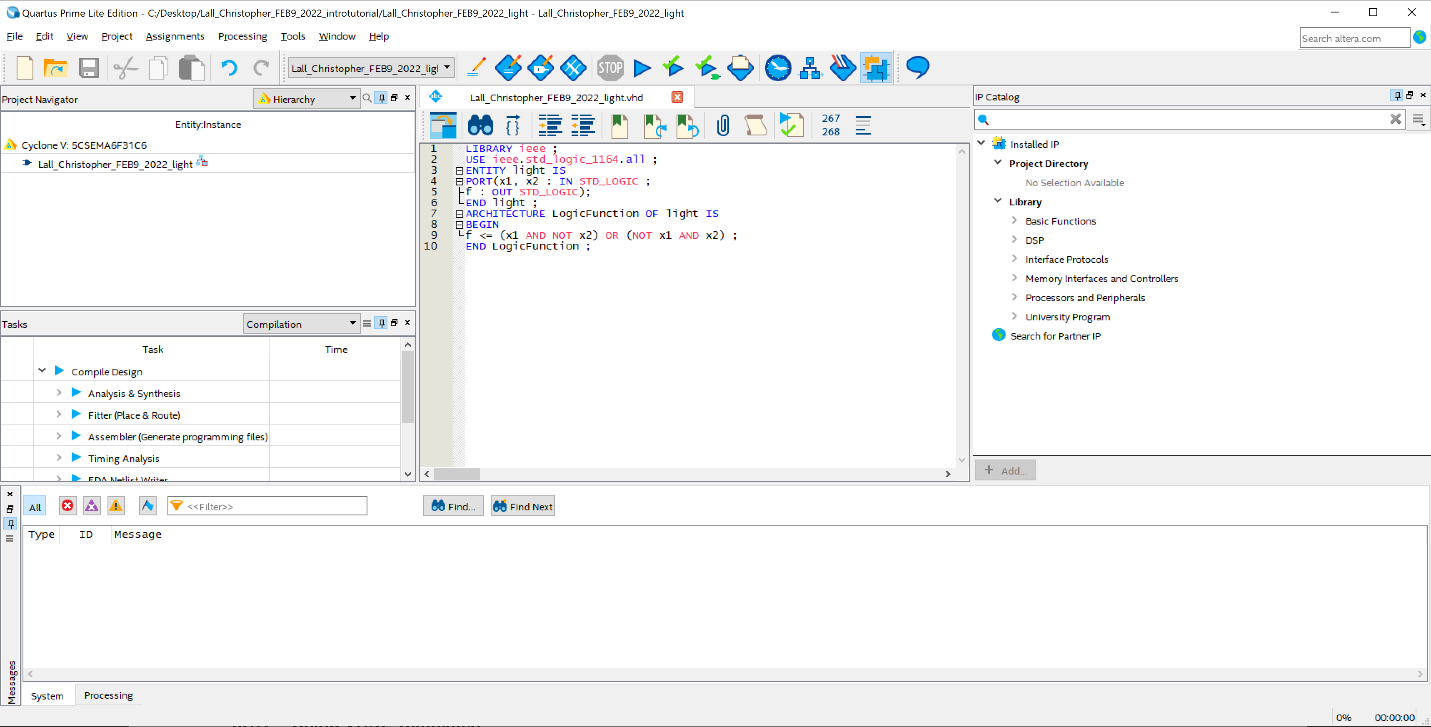
Description automatically generated

Figure 10. Screen after Finishing

Graphical user interface, application

Description automatically generated

Figure 11. VHDL Selection

A screenshot of a computer

Description automatically generated

Figure 12. Select File

Figure 13. VHDL Edit

Graphical user interface, text, application, email

Description automatically generated

Figure 14. FILE APPLY

Graphical user interface, text, application

Description automatically generated

Figure 15. Compilation

After compilation, I am going to do pin assignments:

Graphical user interface, application

Description automatically generated

Figure 16. Pin Assignment

Graphical user interface, text, application, email

Description automatically generated

Figure 17. Export Assignment

# Section 3: 2:1 Mux Screenshots

Graphical user interface, text, application, email

Description automatically generated

Figure 18. Create directory for MUX project

Graphical user interface, application

Description automatically generated

Figure 19. Creating VHDL File for MUX

Once we create this file, we can being coding our design to compile.

## Section 3.1: 2:1 Mux VDHL Code

Our VHDL file for the 2:1 Mux was created by:

Text

Description automatically generated

Figure 20. 2:1 MUX Vhdl

## Section 3.2: 2:1 Mux VDHL Code Compilation

Once we compile this code, we get a compilation report. See figure, 19.

Graphical user interface, application

Description automatically generated

Figure 21. Compilation report for 2:1 MUX

Here, we can see that the compilation report was successful as it did not throw any errors.

# Section 4: 2:1 MUX 32bit VHDL code

Text

Description automatically generated

Figure 22. VHDL Code for MUX 32Bit

## Section 4.1 Mux 32bit VDHL Code Compilation

Graphical user interface

Description automatically generated with medium confidence

Figure 21. Compilation of MUX 32 VHDL Code 32bit

# Section 5: Explanation

The tutorial screenshots was a demonstration to familiarize myself with Quartus Prime Lite. By following the tutorial, I also got screenshots very similar, if not the same, as the given pdf. In the screenshots, you can see how I started a project, named my project, started a file, named my file, create specific directories, open menus, select important options, open files, and run code.

Moving on the 2:1 Mux where each signal is one bit, I use what I learned in the tutorial to help me create and run my code. I start by creating a project. The screenshots in section 3 show how I was able to create a directory to put my project in. Within that project I was able to create a VHDL file where I will code the MUX where each signal is one bit. Once that is done, I simply compile the program, and since there were no errors, I move on.

Very similarly to the 2:1 Mux where each signal is one bit, section 4 focuses on another Mux where each signal is 32 bits, and the selector signal is one bit. Using the exact same steps for section 3 (which I did not include in section 4 because it is exactly the same apart from the naming convention), I was able to open a new directory, and create a VHDL file where I coded this new Mux. Once that was done, I then compiled the code and got no error which means it was a success for me.

# Section 6: Conclusion

The introduction to Quartus Prime Lite via the tutorial helped me to familiarize myself with all options, menus, and ways to do important things. Using the tutorial, I created 2 mux’s. One Mux was a Mux where each signal is one bit, and the second Mux was a Mux where each signal is 32bits, and the selector signal is one bit. This means that the project was doable and I was able to create code that compiled, as seen in the screenshots.