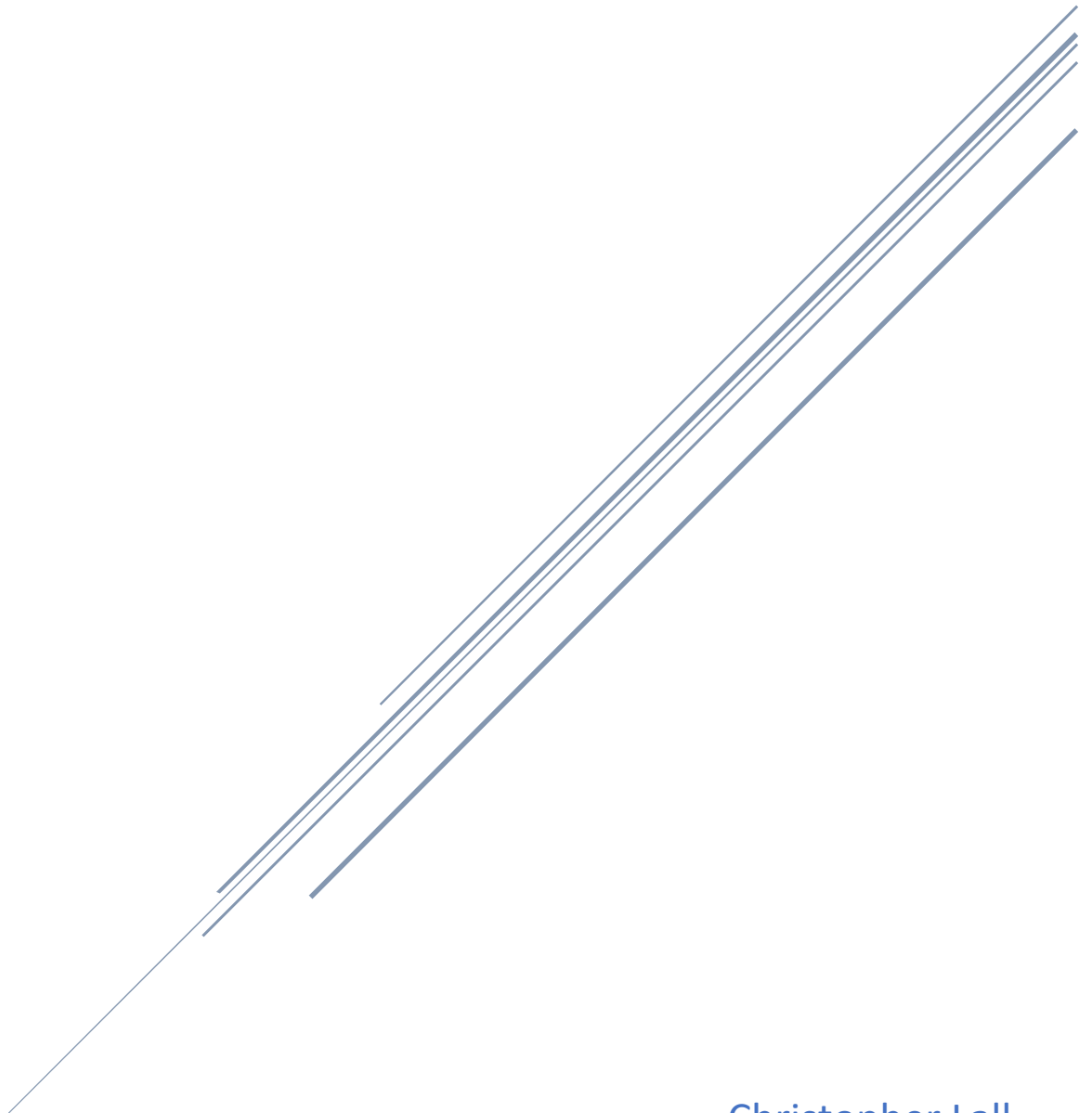


LAB ASSIGNMENT #1 - MUX

Due 2/13/2022



Christopher Lall
CSc 342/343 – Professor Gertner

Table of Contents

Section 1: Objective:	2
Section 2: Tutorial Screenshots:	2
Section 3: 2:1 Mux Screenshots	13
Section 3.1: 2:1 Mux VDHL Code	15
Section 3.2: 2:1 Mux VDHL Code Compilation	15
Section 4: 2:1 MUX 32bit VHDL code	16
Section 4.1 Mux 32bit VDHL Code Compilation	18
Section 5: Explanation	18
Section 6: Conclusion	19

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:

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

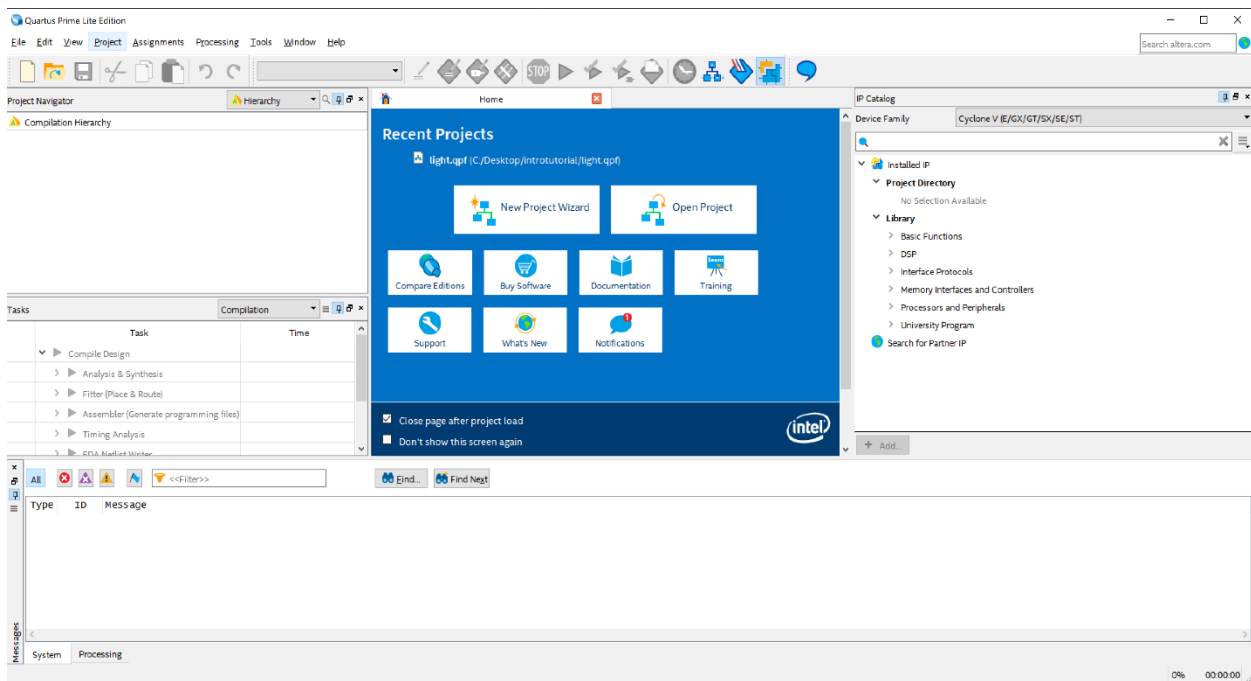


Figure 1. Opening Screen of Quartus

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

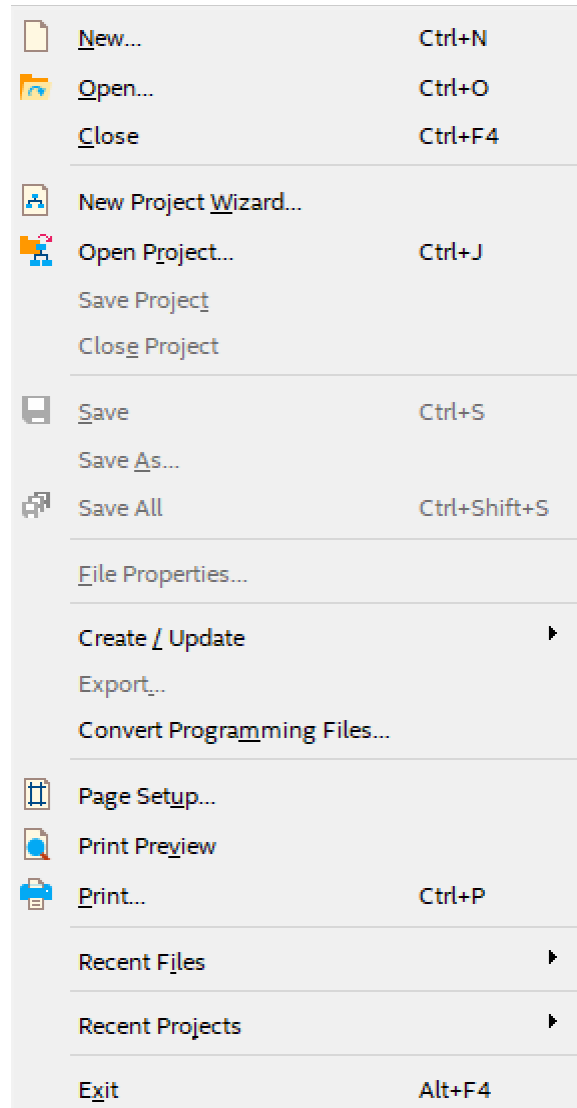
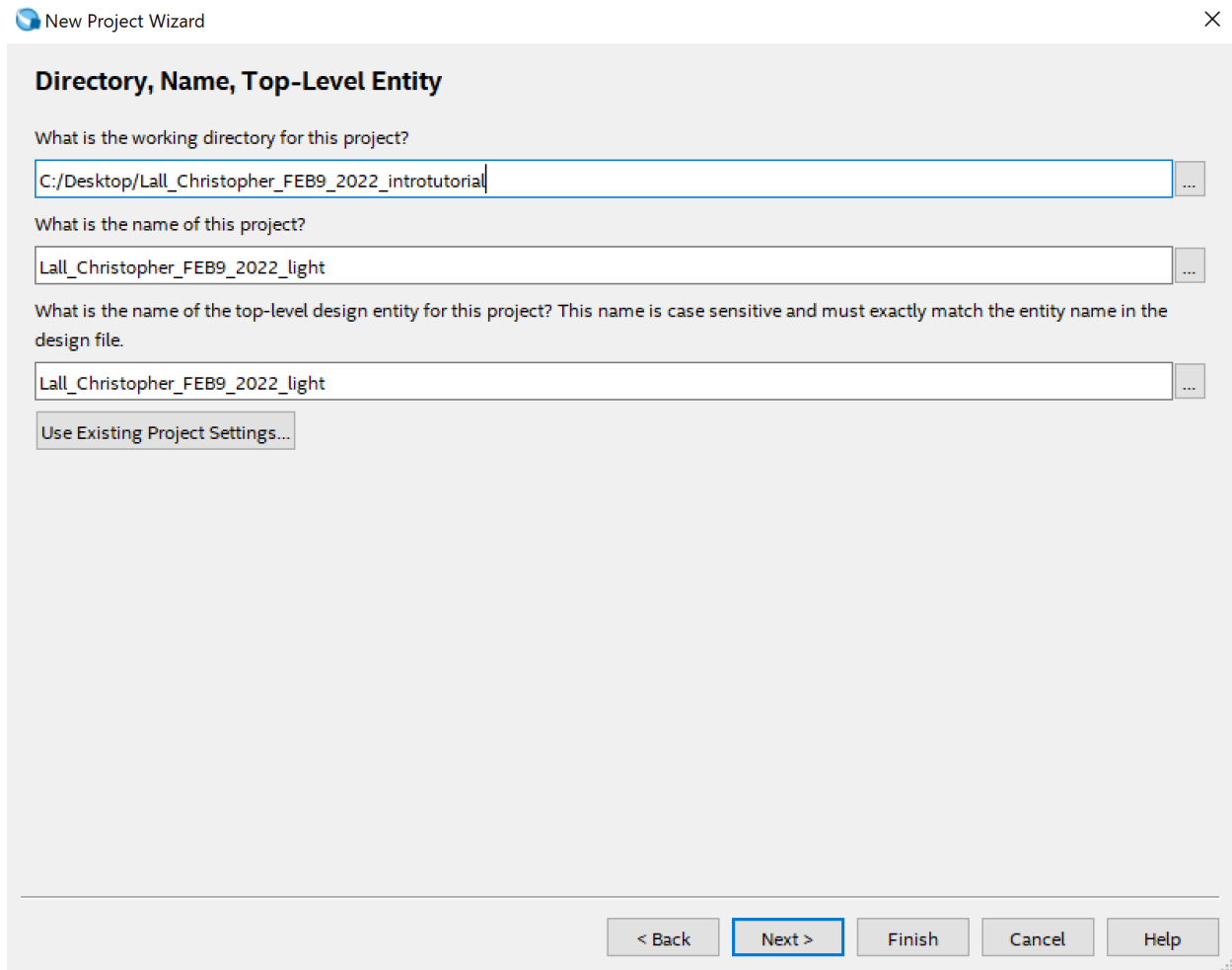


Figure 2. File Menu

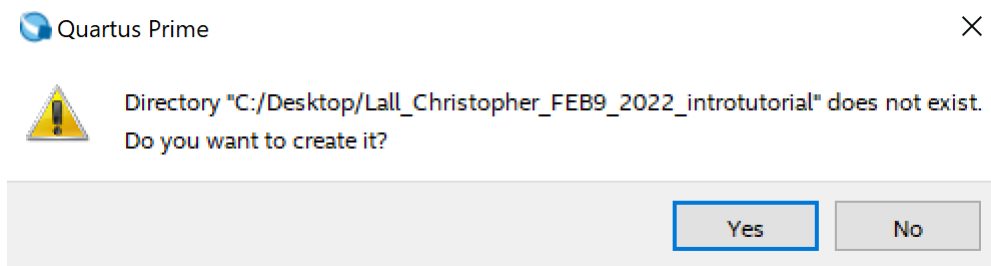
From here we can create a directory.



The image shows the 'New Project Wizard' dialog box in Quartus Prime. The title bar reads 'New Project Wizard' with a close button. The main title is 'Directory, Name, Top-Level Entity'. There are three text input fields: 'What is the working directory for this project?' with the value 'C:/Desktop/Lall_Christopher_FEB9_2022_introtutorial', 'What is the name of this project?' with the value 'Lall_Christopher_FEB9_2022_light', and 'What is the name of the top-level design entity for this project? This name is case sensitive and must exactly match the entity name in the design file.' with the value 'Lall_Christopher_FEB9_2022_light'. Below these fields is a button labeled 'Use Existing Project Settings...'. At the bottom, there are five buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'. The 'Next >' button is highlighted with a blue border.

Figure 3. Creating Directory

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



The image shows a warning dialog box from Quartus Prime. The title bar reads 'Quartus Prime' with a close button. The main text says: 'Directory "C:/Desktop/Lall_Christopher_FEB9_2022_introtutorial" does not exist. Do you want to create it?'. There is a yellow warning triangle icon on the left. At the bottom, there are two buttons: 'Yes' and 'No'. The 'Yes' button is highlighted with a blue border.

Figure 4. Confirmation of creating directory

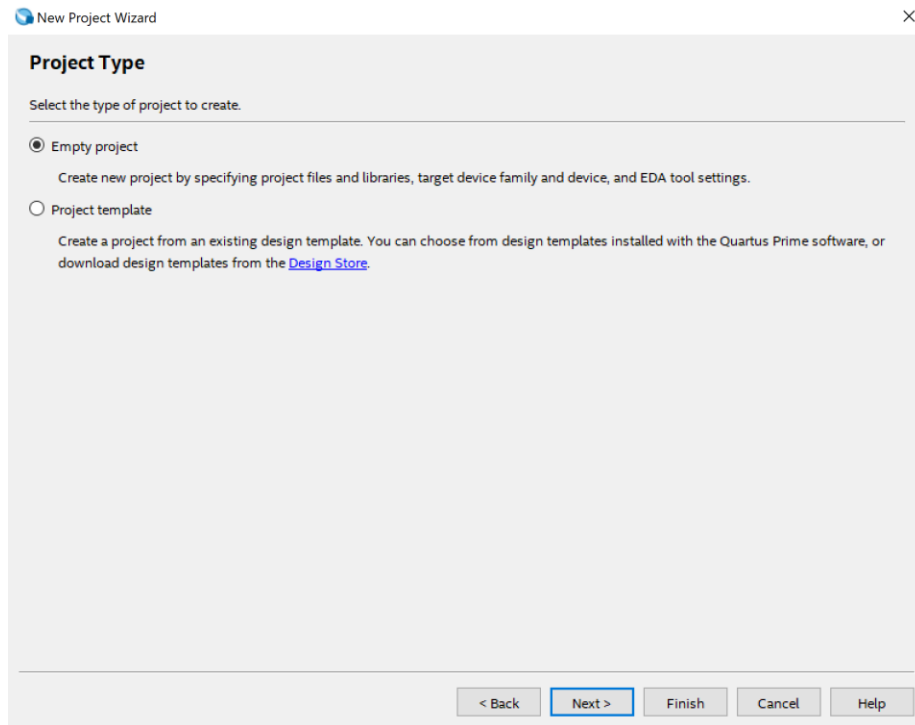


Figure 5. Selecting Project Type

Once we click next we arrive at figure 6.

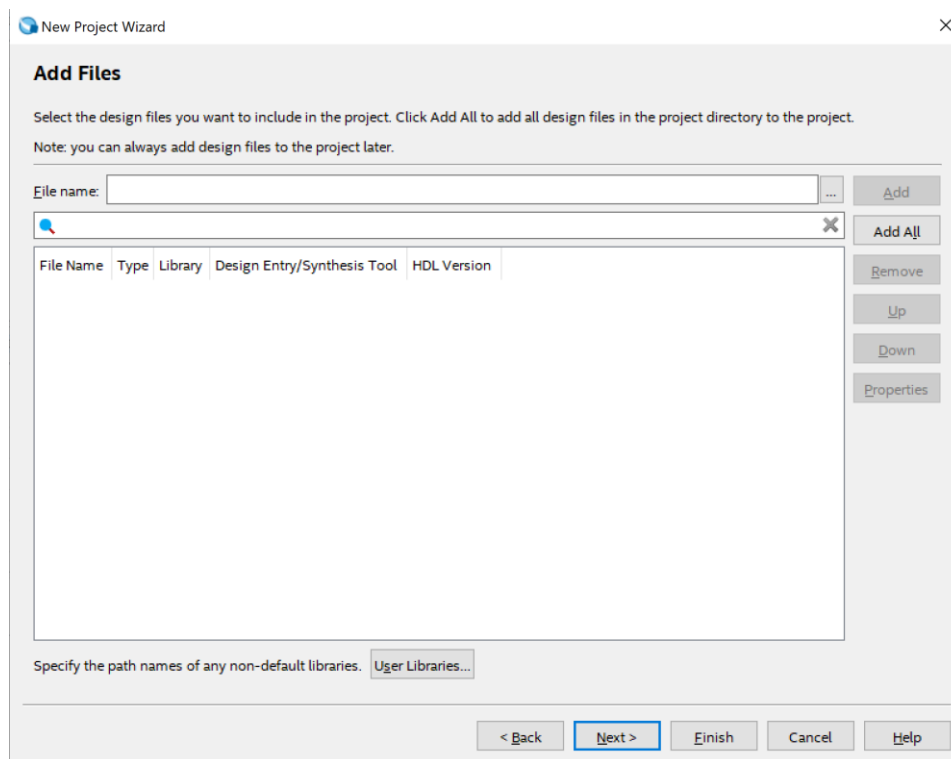


Figure 6. Adding Files

New Project Wizard

Family, Device & Board Settings

Device Board

Select the family and device you want to target for compilation.
You can install additional device support with the Install Devices command on the Tools menu.

To determine the version of the Quartus Prime software in which your target device is supported, refer to the [Device Support List](#) webpage.

Device family

Family: Cyclone V (E/GX/GT/SX/SE/ST)

Device: All

Target device

☐ Auto device selected by the Fitter

☒ Specific device selected in 'Available devices' list

☐ Other: n/a

Show in 'Available devices' list

Package: Any

Pin count: Any

Core speed grade: Any

Name filter:

☒ Show advanced devices

Available devices:

Name	Core Voltage	ALMs	Total I/Os	GPIOs	GXB Channel PMA	GXB Channel PCS	PCIe Ha
5CSEMA6F31A7	1.1V	41910	457	457	0	0	0
5CSEMA6F31C6	1.1V	41910	457	457	0	0	0
5CSEMA6F31C7	1.1V	41910	457	457	0	0	0
5CSEMA6F31C8	1.1V	41910	457	457	0	0	0
5CSEMA6F31I7	1.1V	41910	457	457	0	0	0

< Back Next > Finish Cancel Help

Figure 7. Choosing Family, Device, and Board Settings

New Project Wizard

EDA Tool Settings

Specify the other EDA tools used with the Quartus Prime software to develop your project.

EDA tools:

Tool Type	Tool Name	Format(s)	Run Tool Automatically
Design Entry/Synth...	<None>	<None>	<input type="checkbox"/> Run this tool automatically to synthesize the current design
Simulation	<None>	<None>	<input type="checkbox"/> Run gate-level simulation automatically after compilation
Board-Level	Timing	<None>	
	Symbol	<None>	
	Signal Integrity	<None>	
	Boundary Scan	<None>	

< Back Next > Finish Cancel Help

Figure 8. EDA Tool Settings

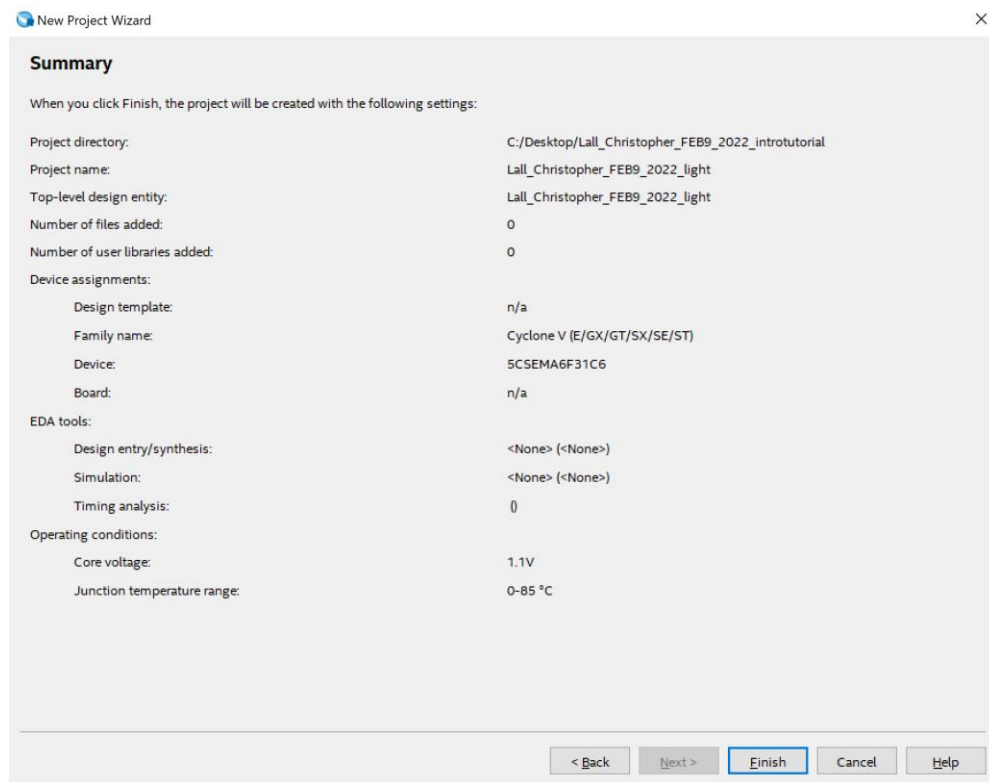


Figure 9. Summary

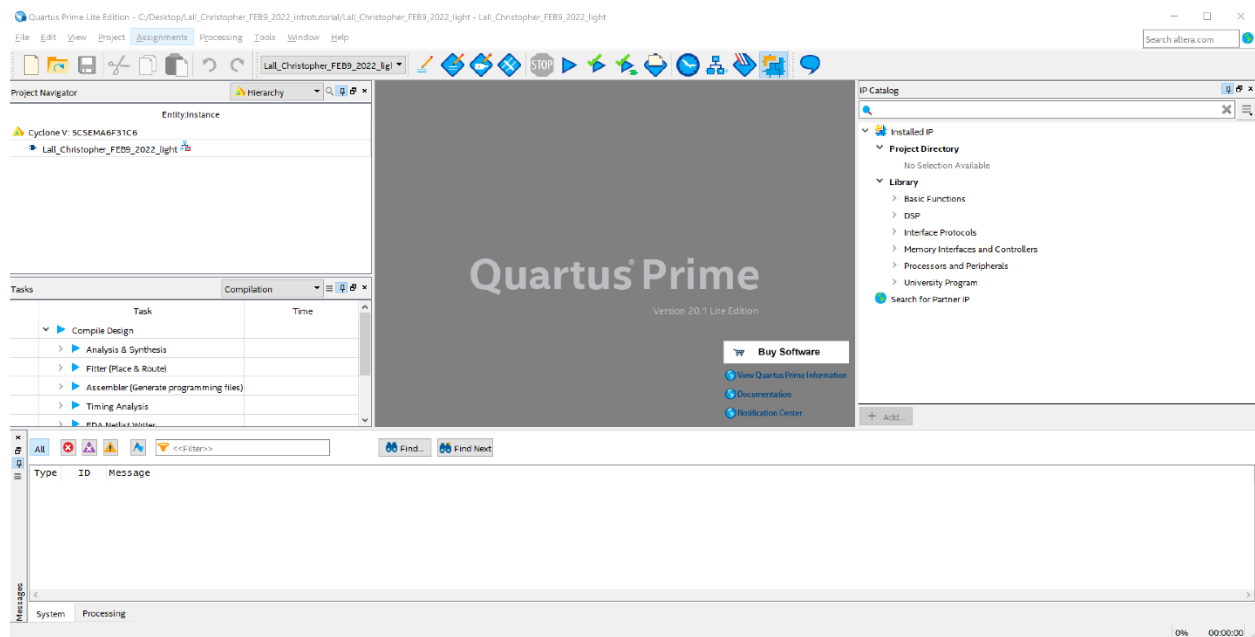


Figure 10. Screen after Finishing

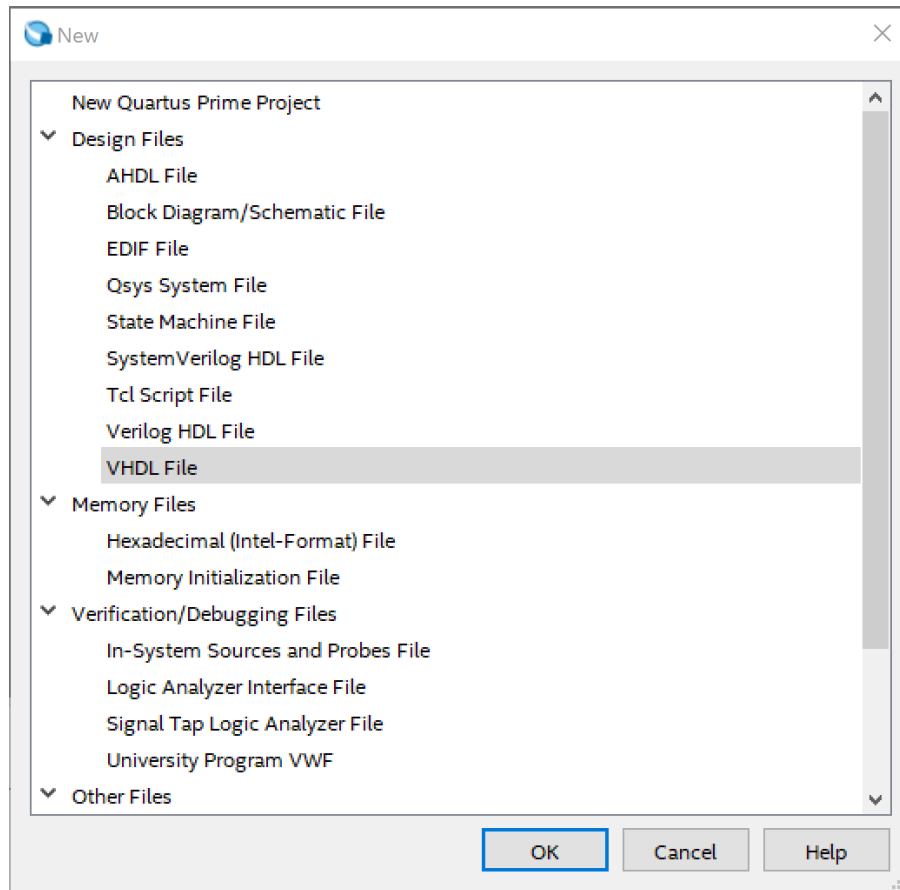


Figure 11. VHDL Selection

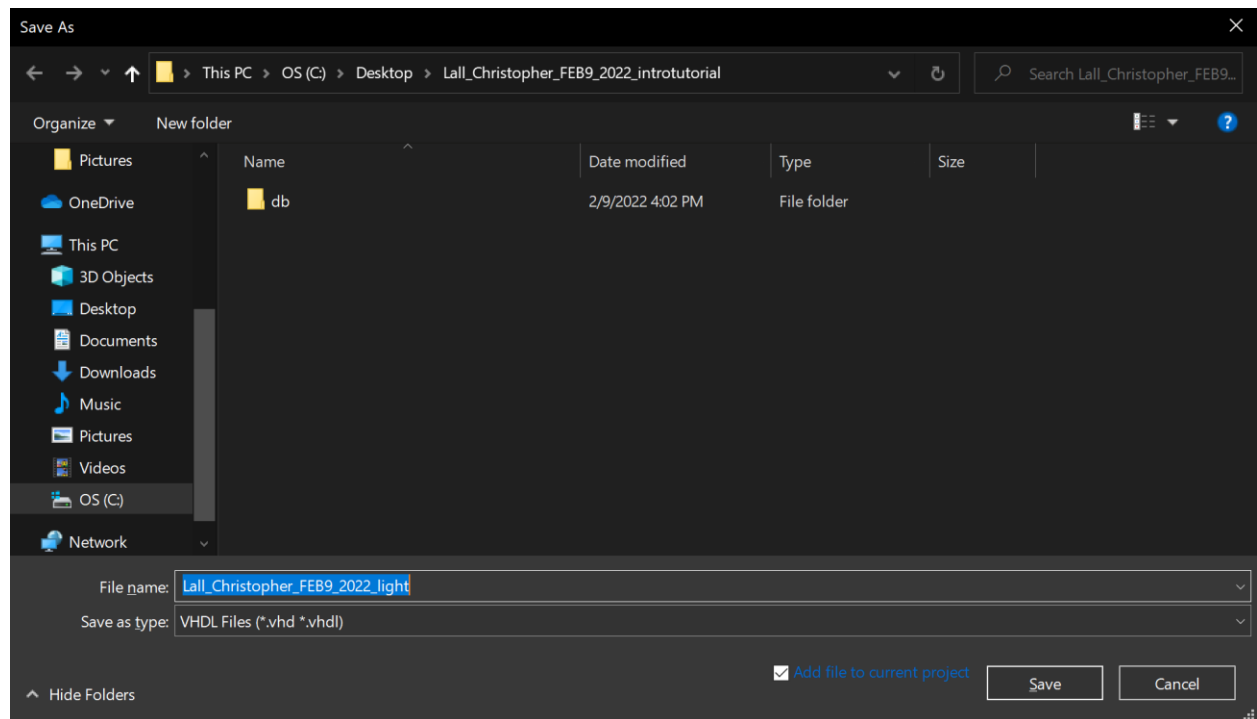


Figure 12. Select File

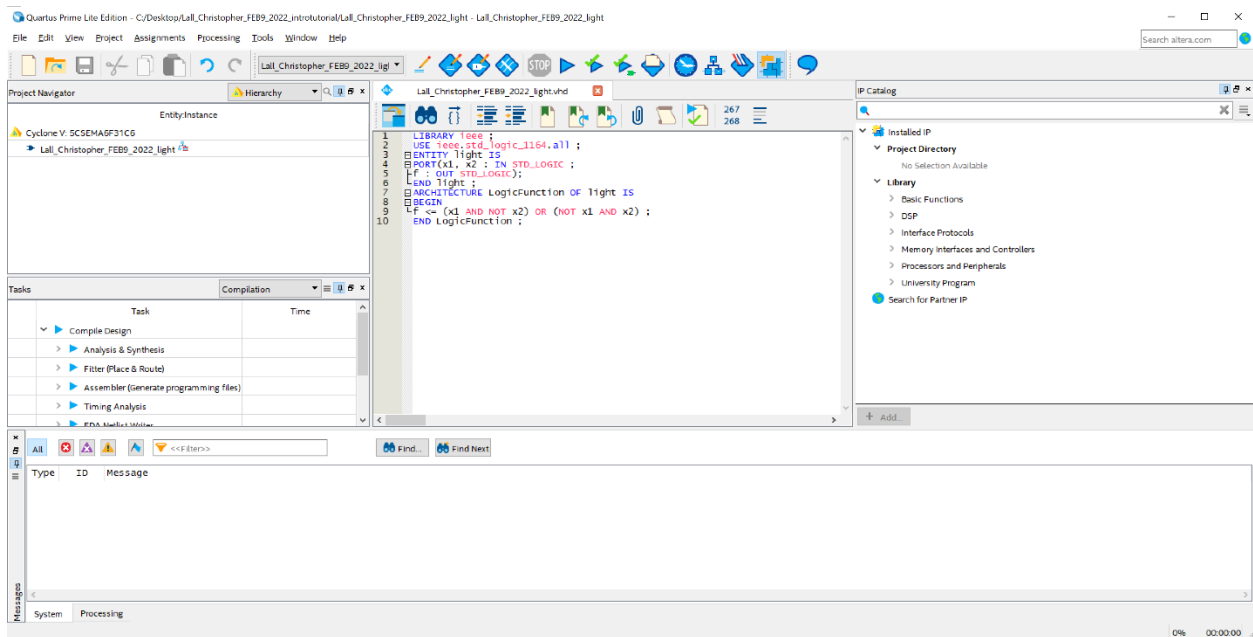
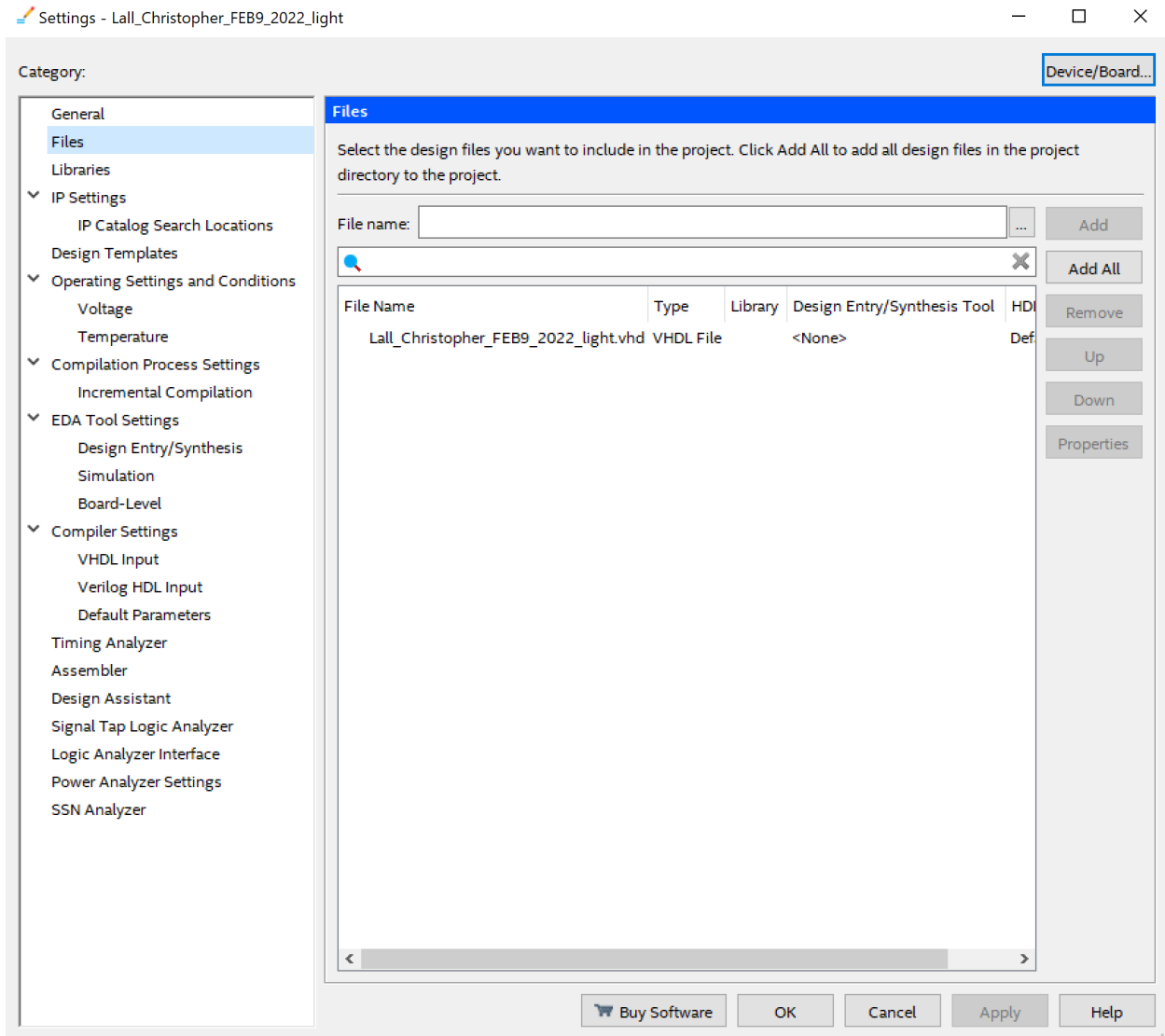


Figure 13. VHDL Edit

*Figure 14. FILE APPLY*

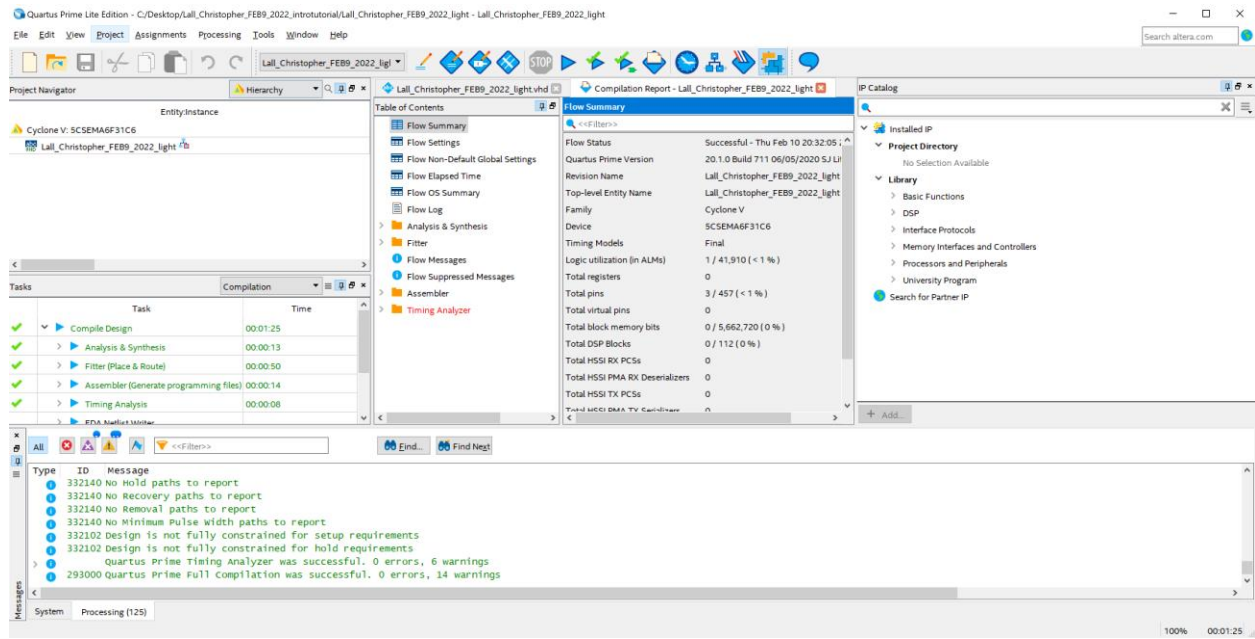


Figure 15. Compilation

After compilation, I am going to do pin assignments:

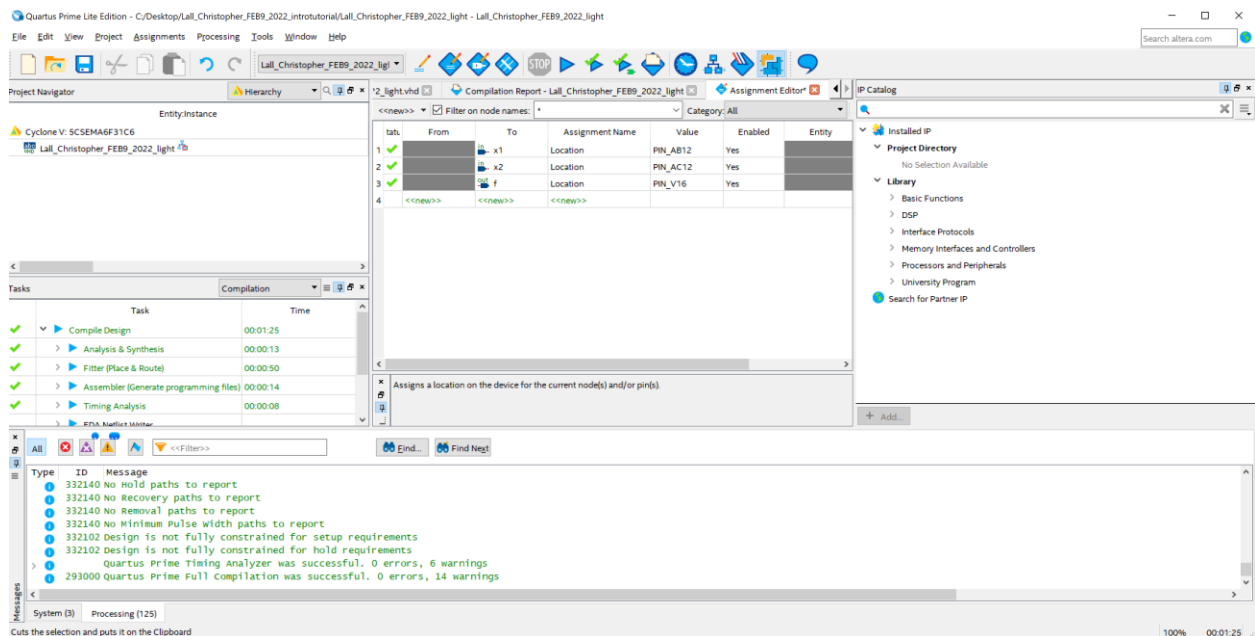


Figure 16. Pin Assignment

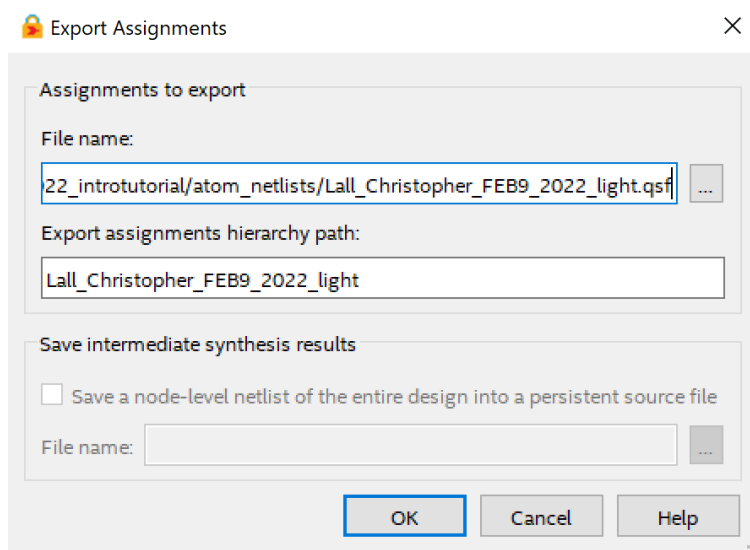
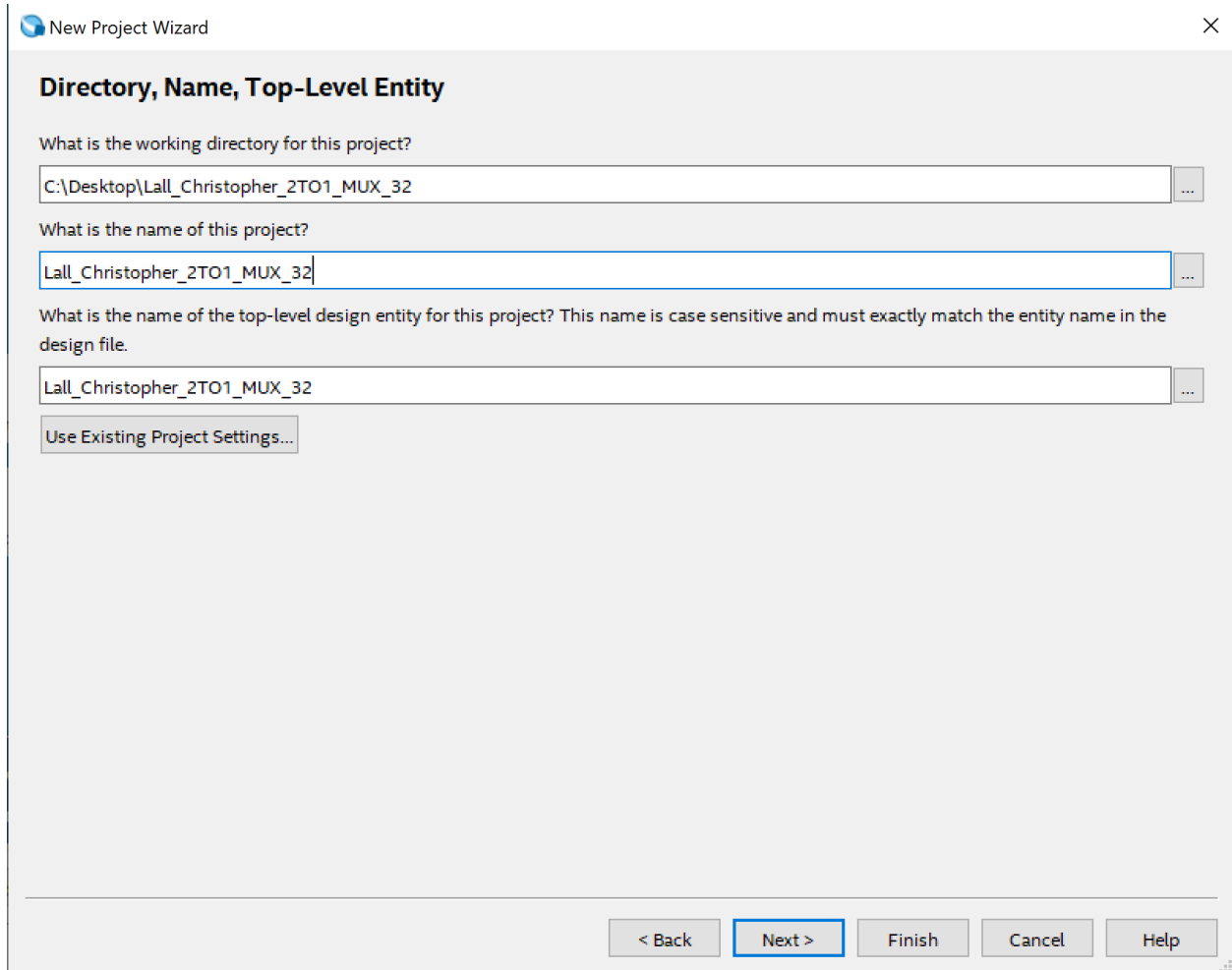


Figure 17. Export Assignment

Section 3: 2:1 Mux Screenshots



The image shows a 'New Project Wizard' dialog box with a title bar containing a blue icon and the text 'New Project Wizard' and a close button. The main area is titled 'Directory, Name, Top-Level Entity'. It contains three text input fields, each with a browse button (three dots) to its right. The first field is labeled 'What is the working directory for this project?' and contains the text 'C:\Desktop\Lall_Christopher_2TO1_MUX_32'. The second field is labeled 'What is the name of this project?' and contains the text 'Lall_Christopher_2TO1_MUX_32'. The third field is labeled 'What is the name of the top-level design entity for this project? This name is case sensitive and must exactly match the entity name in the design file.' and contains the text 'Lall_Christopher_2TO1_MUX_32'. Below the third field is a button labeled 'Use Existing Project Settings...'. At the bottom of the dialog are five buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'. The 'Next >' button is highlighted with a blue border.

New Project Wizard

Directory, Name, Top-Level Entity

What is the working directory for this project?

C:\Desktop\Lall_Christopher_2TO1_MUX_32

What is the name of this project?

Lall_Christopher_2TO1_MUX_32

What is the name of the top-level design entity for this project? This name is case sensitive and must exactly match the entity name in the design file.

Lall_Christopher_2TO1_MUX_32

Use Existing Project Settings...

< Back Next > Finish Cancel Help

Figure 18. Create directory for MUX project

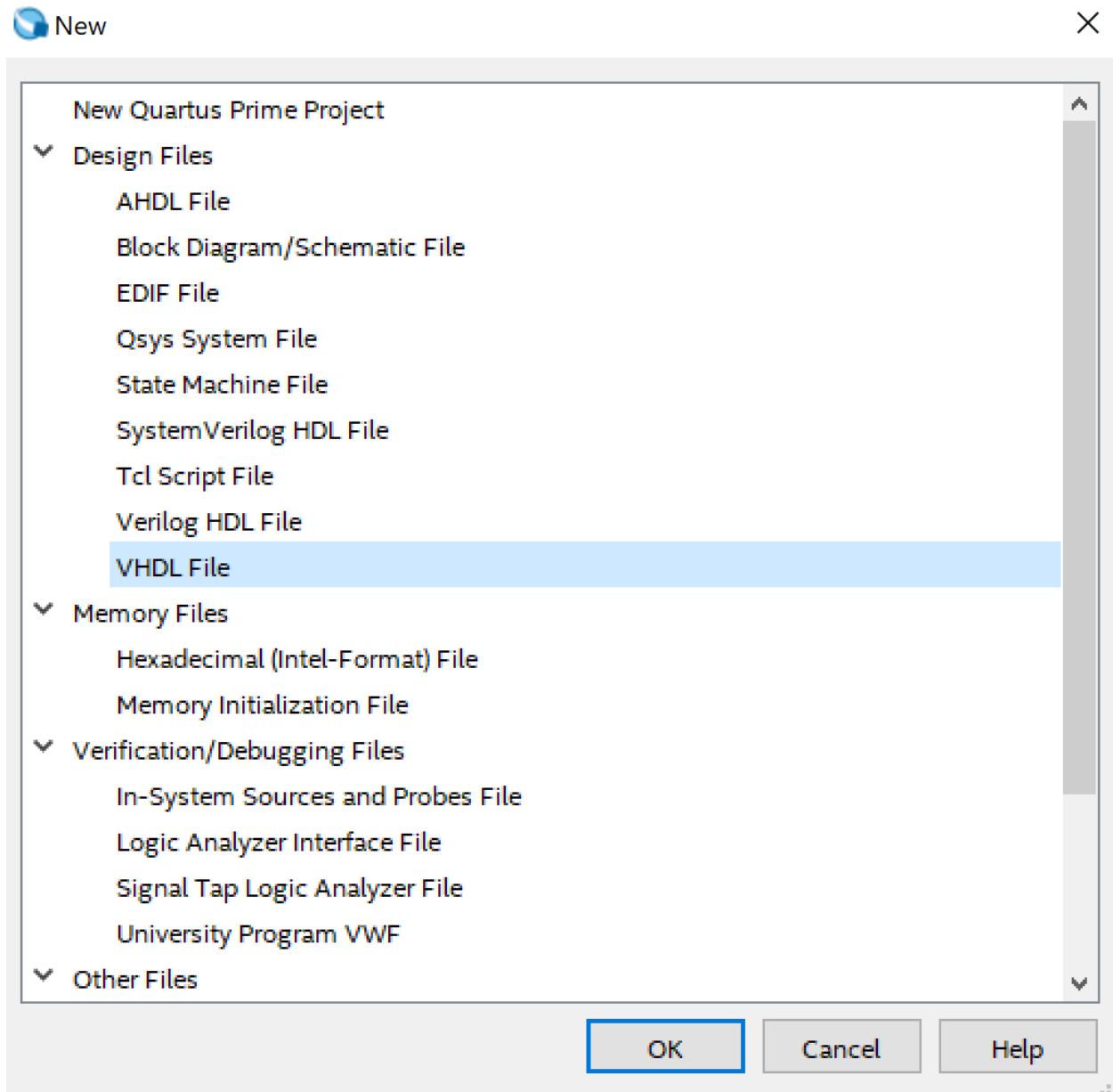


Figure 19. Creating VHDL File for MUX

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

Section 3.1: 2:1 Mux VHDL Code

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

```

2  -- VERSION      "Version 20.1.0 Build 711 06/05/2020 SJ Lite Edition"
3  -- CREATED      "Thu Feb 10 22:14:26 2022"
4
5  LIBRARY ieee;
6  USE ieee.std_logic_1164.all;
7
8  LIBRARY work;
9
10 ENTITY Lall_Christopher_FEB9_2022_2T01_MUX IS
11     PORT
12     (
13         Lall_Christopher_A : IN  STD_LOGIC;
14         Lall_Christopher_B : IN  STD_LOGIC;
15         Lall_Christopher_S : IN  STD_LOGIC;
16         Lall_Christopher_Out : OUT STD_LOGIC
17     );
18 END Lall_Christopher_FEB9_2022_2T01_MUX;
19
20 ARCHITECTURE bdf_type OF Lall_Christopher_FEB9_2022_2T01_MUX IS
21
22     SIGNAL SYNTHESIZED_WIRE_0 : STD_LOGIC;
23     SIGNAL SYNTHESIZED_WIRE_1 : STD_LOGIC;
24     SIGNAL SYNTHESIZED_WIRE_2 : STD_LOGIC;
25
26
27 BEGIN
28
29
30     SYNTHESIZED_WIRE_0 <= NOT(Lall_Christopher_S);
31
32
33
34     SYNTHESIZED_WIRE_1 <= Lall_Christopher_S AND Lall_Christopher_B;
35
36
37     SYNTHESIZED_WIRE_2 <= Lall_Christopher_A AND SYNTHESIZED_WIRE_0;
38
39
40     Lall_Christopher_Out <= SYNTHESIZED_WIRE_1 OR SYNTHESIZED_WIRE_2;
41
42
43
44 END bdf_type;

```

Figure 20. 2:1 MUX Vhdl

Section 3.2: 2:1 Mux VHDL Code Compilation

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


Flow Summary	
 <<Filter>>	
Flow Status	Successful - Thu Feb 10 21:49:35 2022
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	Lall_Christopher_FEB9_2022_2TO1_MUX
Top-level Entity Name	Lall_Christopher_FEB9_2022_2TO1_MUX
Family	Cyclone V
Device	5CSEMA6F31C6
Timing Models	Final
Logic utilization (in ALMs)	1 / 41,910 (< 1 %)
Total registers	0
Total pins	4 / 457 (< 1 %)
Total virtual pins	0
Total block memory bits	0 / 5,662,720 (0 %)
Total DSP Blocks	0 / 112 (0 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	0 / 6 (0 %)
Total DLLs	0 / 4 (0 %)

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

```
1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  LIBRARY work;
5
6
7
8  entity Lall_Christopher_2To1_MUX_32 is
9  port
10     (CLK : in STD_LOGIC;
11      A : in STD_LOGIC_VECTOR (31 downto 0);
12      B : in STD_LOGIC_VECTOR (31 downto 0);
13      X : out STD_LOGIC_VECTOR (31 downto 0));
14 end Lall_Christopher_2To1_MUX_32;
15 architecture Behavioral of Lall_Christopher_2To1_MUX_32 is
16 begin
17     X <= A when (CLK = '1') else B;
18 end Behavioral;
```

Figure 22. VHDL Code for MUX 32Bit

Section 4.1 Mux 32bit VDHL Code Compilation

Flow Status	Successful - Fri Feb 11 18:16:55 2022
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	Lall_Christopher_2TO1_MUX_32
Top-level Entity Name	Lall_Christopher_2TO1_MUX_32
Family	Cyclone V
Device	5CSEMA6F31C6
Timing Models	Final
Logic utilization (in ALMs)	17 / 41,910 (< 1 %)
Total registers	0
Total pins	97 / 457 (21 %)
Total virtual pins	0
Total block memory bits	0 / 5,662,720 (0 %)
Total DSP Blocks	0 / 112 (0 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	0 / 6 (0 %)
Total DLLs	0 / 4 (0 %)

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.