Christopher Lall

In class lab assignment 1 – BUZZER

CSc 34300 & CSc 34200

Due 2/16/2022

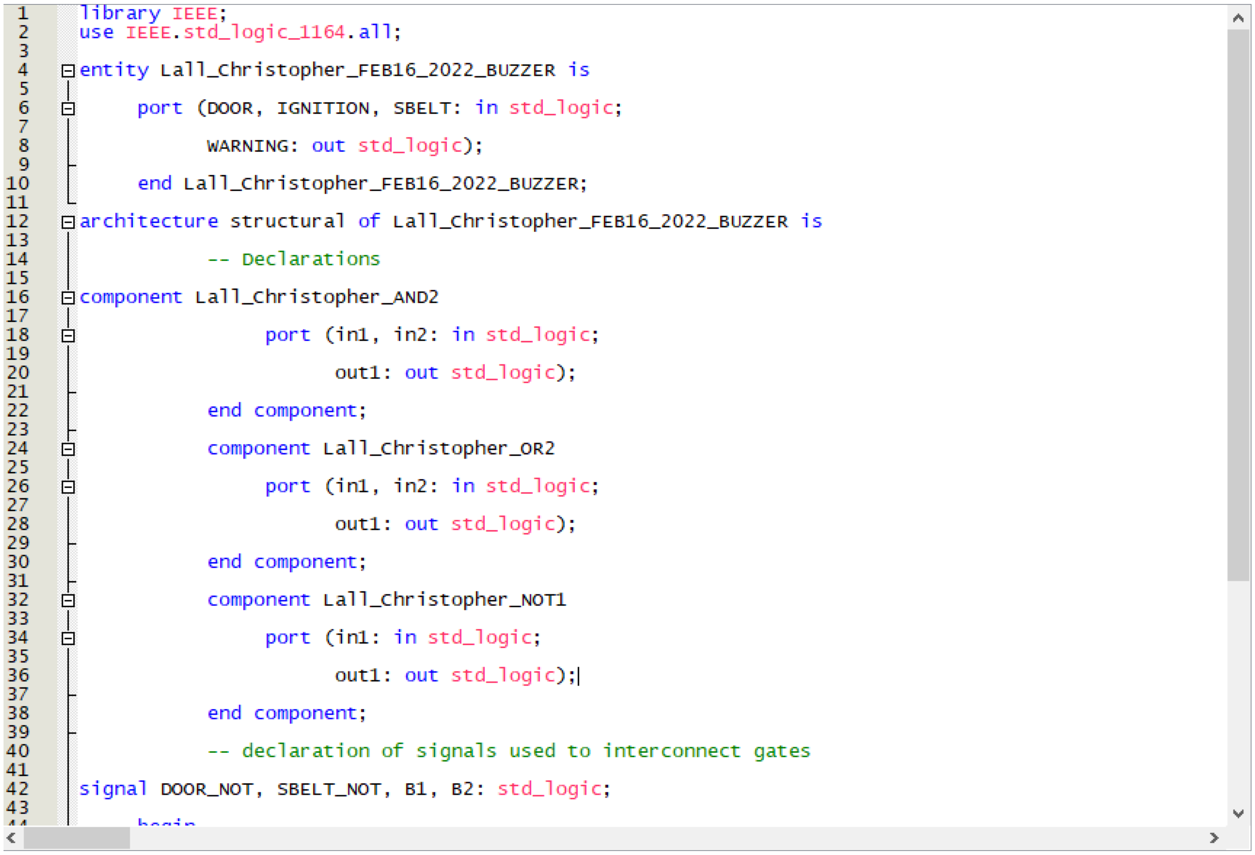
Table of Contents

Objective

The objective of this lab is to simulate the Buzzer and 4to1 Mux code and verify its accuracy with the truth tables.

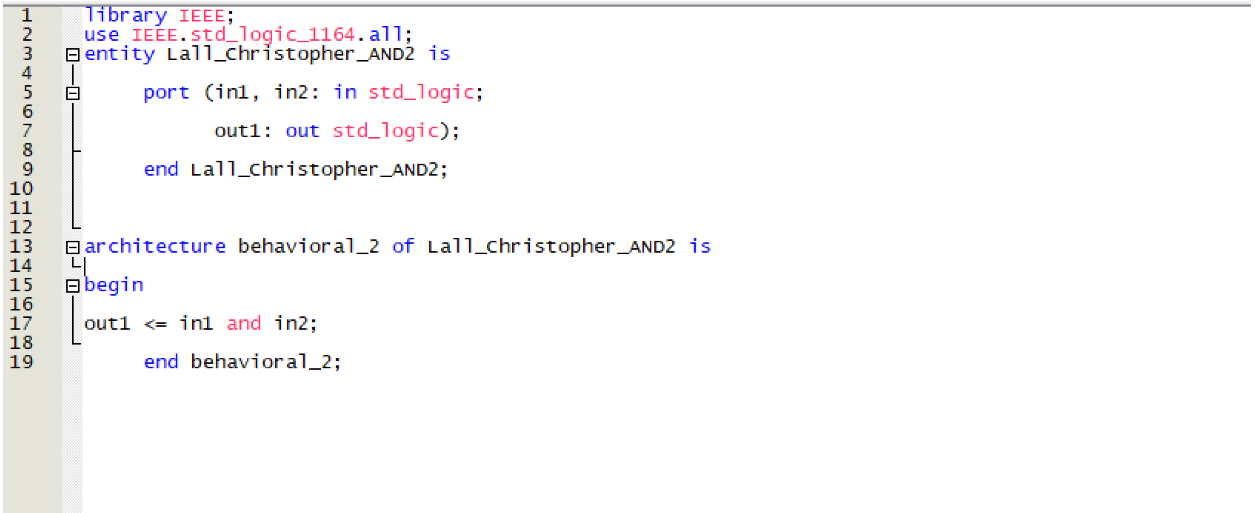
Buzzer

Screenshots:



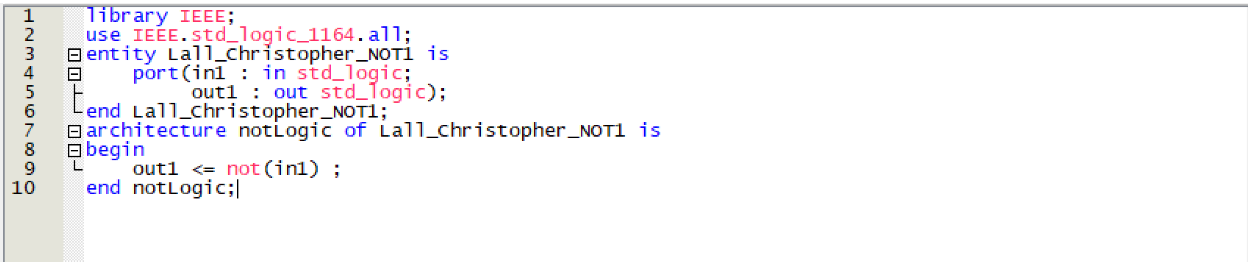
```
library IEEE;
use IEEE.std_logic_1164.all;

entity Lall_Christopher_FEB16_2022_BUZZER is

     port (DOOR, IGNITION, SBELT: in std_logic;

           WARNING: out std_logic);

     end Lall_Christopher_FEB16_2022_BUZZER;

architecture structural of Lall_Christopher_FEB16_2022_BUZZER is

           -- Declarations

component Lall_Christopher_AND2

                port (in1, in2: in std_logic;

                      out1: out std_logic);

           end component;

           component Lall_Christopher_OR2

                port (in1, in2: in std_logic;

                      out1: out std_logic);

           end component;

           component Lall_Christopher_NOT1

                port (in1: in std_logic;

                      out1: out std_logic);

           end component;

           -- declaration of signals used to interconnect gates

signal DOOR_NOT, SBELT_NOT, B1, B2: std_logic;

     begin
```

Figure 1. Code for Buzzer file



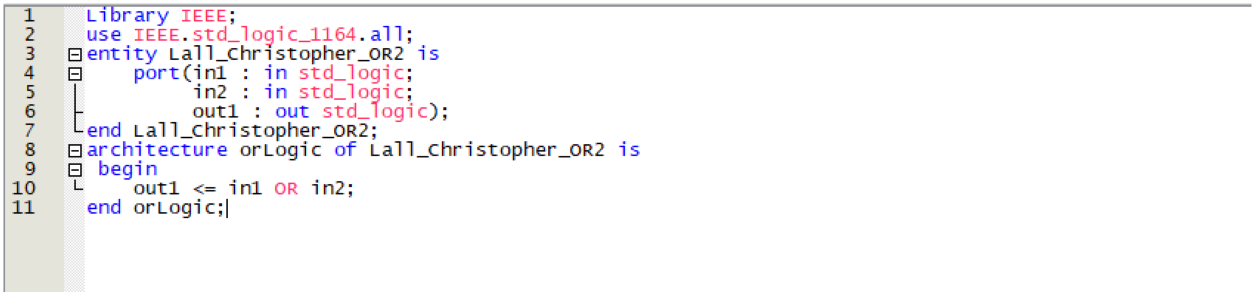
```
library IEEE;
use IEEE.std_logic_1164.all;
entity Lall_Christopher_AND2 is

     port (in1, in2: in std_logic;

           out1: out std_logic);

     end Lall_Christopher_AND2;



architecture behavioral_2 of Lall_Christopher_AND2 is

begin

out1 <= in1 and in2;

     end behavioral_2;
```

Figure 2. Code for AND2



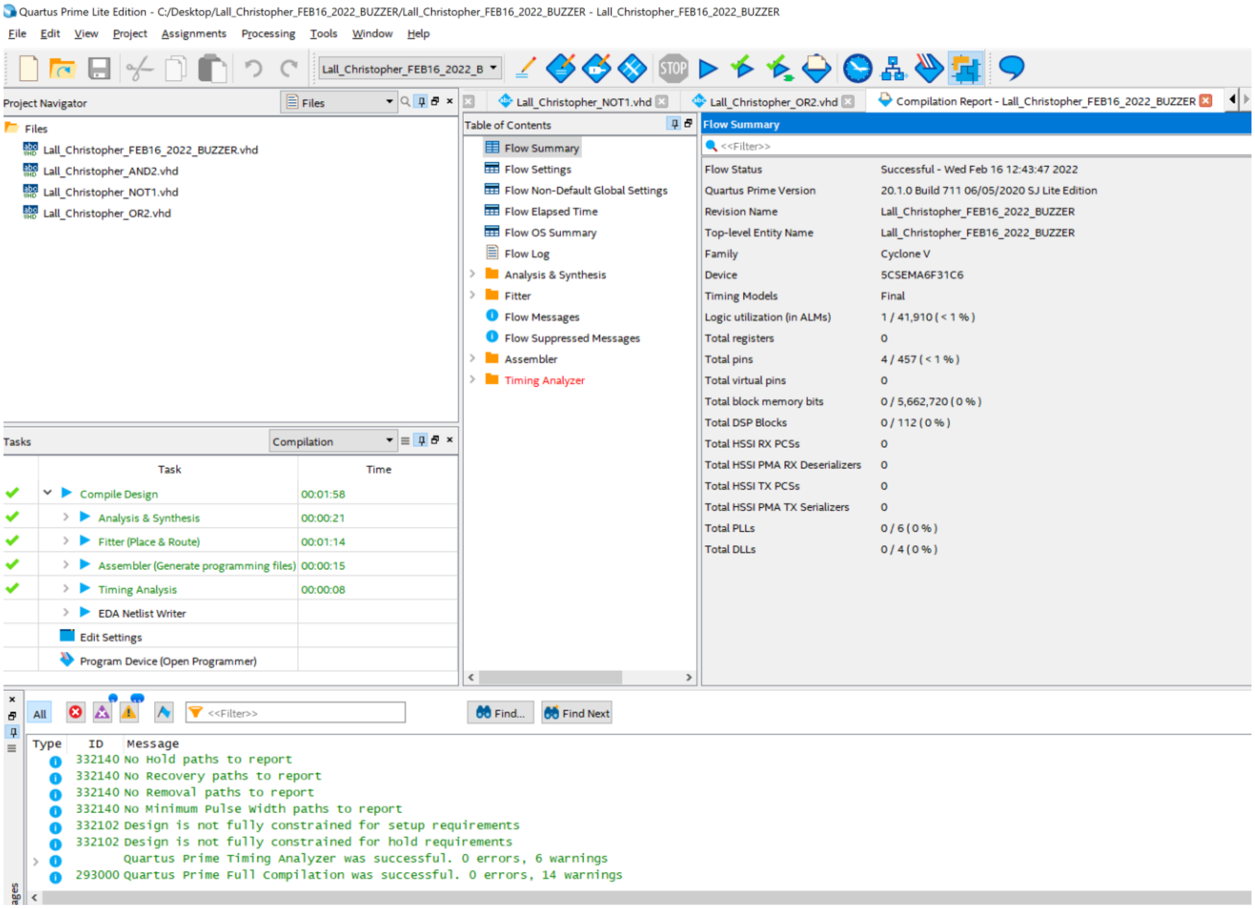
```
library IEEE;
use IEEE.std_logic_1164.all;
entity Lall_Christopher_NOT1 is
    port(in1 : in std_logic;
         out1 : out std_logic);
end Lall_Christopher_NOT1;
architecture notLogic of Lall_Christopher_NOT1 is
begin
    out1 <= not(in1) ;
end notLogic;
```

Figure 3. Code for NOT1



```
Library IEEE;
use IEEE.std_logic_1164.all;
entity Lall_Christopher_OR2 is
    port(in1 : in std_logic;
         in2 : in std_logic;
         out1 : out std_logic);
end Lall_Christopher_OR2;
architecture orLogic of Lall_Christopher_OR2 is
 begin
    out1 <= in1 OR in2;
end orLogic;
```

Figure 4. Code for OR2



Quartus Prime Lite Edition - C:/Desktop/Lall_Christopher_FEB16_2022_BUZZER/Lall_Christopher_FEB16_2022_BUZZER - Lall_Christopher_FEB16_2022_BUZZER
File Edit View Project Assignments Processing Tools Window Help
Lall_Christopher_FEB16_2022_B
Project Navigator
Files
Lall_Christopher_NOT1.vhd
Lall_Christopher_OR2.vhd
Compilation Report - Lall_Christopher_FEB16_2022_BUZZER
Tasks
Compilation
All
<<Filter>>
Find...
Find Next

Files
- Lall_Christopher_FEB16_2022_BUZZER.vhd
- Lall_Christopher_AND2.vhd
- Lall_Christopher_NOT1.vhd
- Lall_Christopher_OR2.vhd

	Task	Time
✓	Compile Design	00:01:58
✓	Analysis & Synthesis	00:00:21
✓	Fitter (Place & Route)	00:01:14
✓	Assembler (Generate programming files)	00:00:15
✓	Timing Analysis	00:00:08
	EDA Netlist Writer	
	Edit Settings	
	Program Device (Open Programmer)	

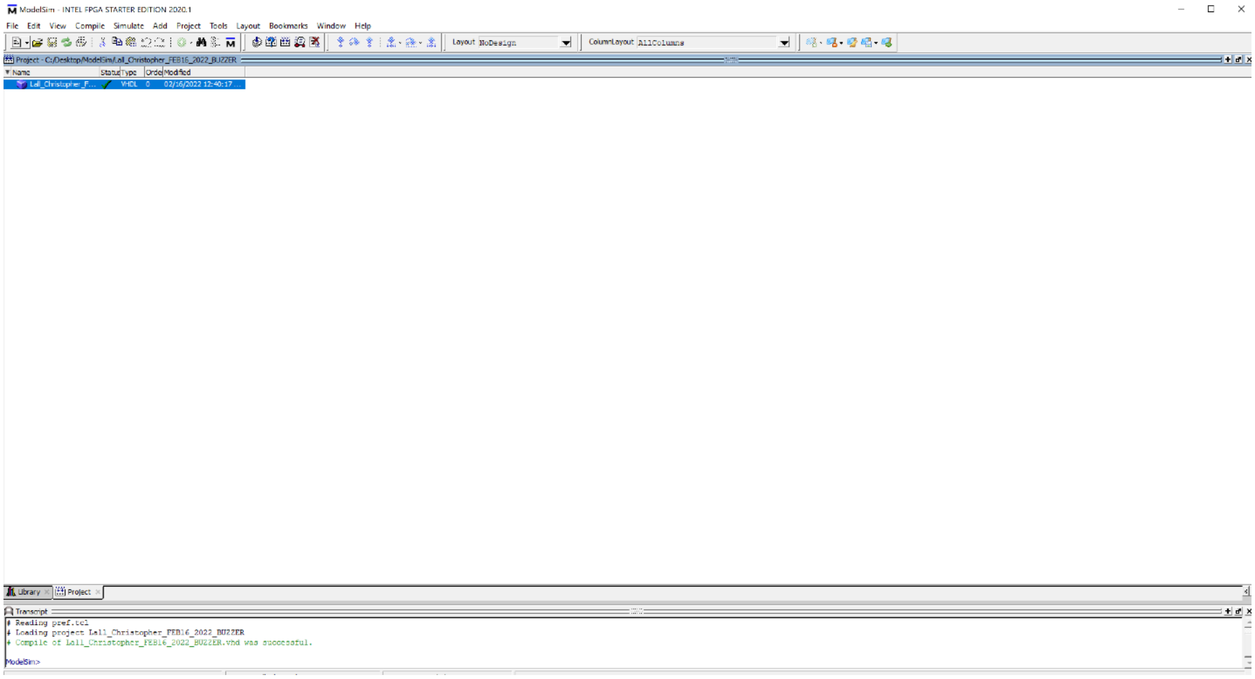
Table of Contents
- Flow Summary
- Flow Settings
- Flow Non-Default Global Settings
- Flow Elapsed Time
- Flow OS Summary
- Flow Log
- Analysis & Synthesis
- Fitter
- Flow Messages
- Flow Suppressed Messages
- Assembler
- Timing Analyzer

Flow Summary

Flow Status	Successful - Wed Feb 16 12:43:47 2022
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	Lall_Christopher_FEB16_2022_BUZZER
Top-level Entity Name	Lall_Christopher_FEB16_2022_BUZZER
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 %)

```
Type   ID     Message
       332140 No Hold paths to report
       332140 No Recovery paths to report
       332140 No Removal paths to report
       332140 No Minimum Pulse Width paths to report
       332102 Design is not fully constrained for setup requirements
       332102 Design is not fully constrained for hold requirements
              Quartus Prime Timing Analyzer was successful. 0 errors, 6 warnings
       293000 Quartus Prime Full Compilation was successful. 0 errors, 14 warnings
```

Figure 5. Compilation Report in Quartus


ModelSim - INTEL FPGA STARTER EDITION 2020.1
File Edit View Compile Simulate Add Project Tools Layout Bookmarks Window Help
Layout NoDesign
ColumnLayout AllColumns
Library
Project
Transcript

Project - C:/Desktop/ModelSim/Lall_Christopher_FEB16_2022_BUZZER

Name	Status	Type	Order	Modified
Lall_Christopher_F...	✓	VHDL	0	02/16/2022 12:40:17 ...

```
# Reading pref.tcl
# Loading project Lall_Christopher_FEB16_2022_BUZZER
# Compile of Lall_Christopher_FEB16_2022_BUZZER.vhd was successful.

ModelSim>
```

Figure 6. Compilation in ModelSim

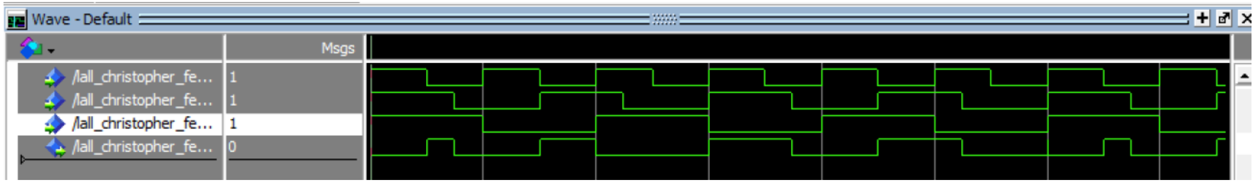

Wave - Default
Msgs
/lall_christopher_fe... 1
/lall_christopher_fe... 1
/lall_christopher_fe... 1
/lall_christopher_fe... 0

Figure 7. Waveform in ModelSim

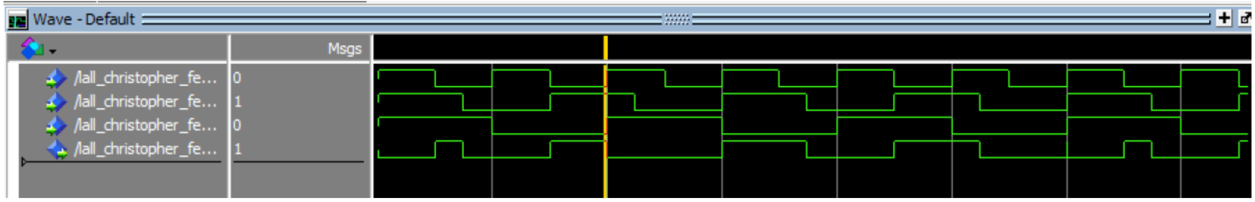

Wave - Default
Msgs
/lall_christopher_fe... 0
/lall_christopher_fe... 1
/lall_christopher_fe... 0
/lall_christopher_fe... 1

Figure 8. Verification of an example

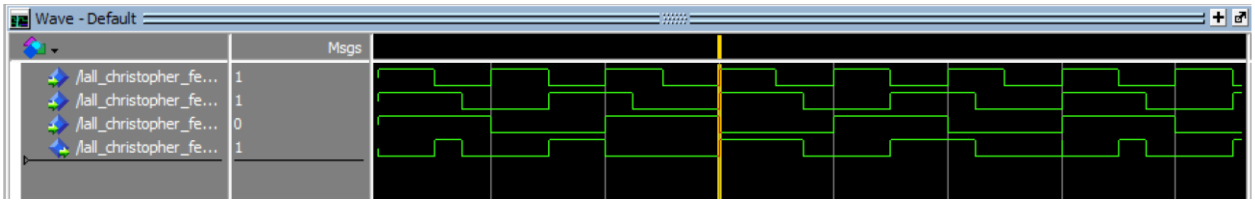
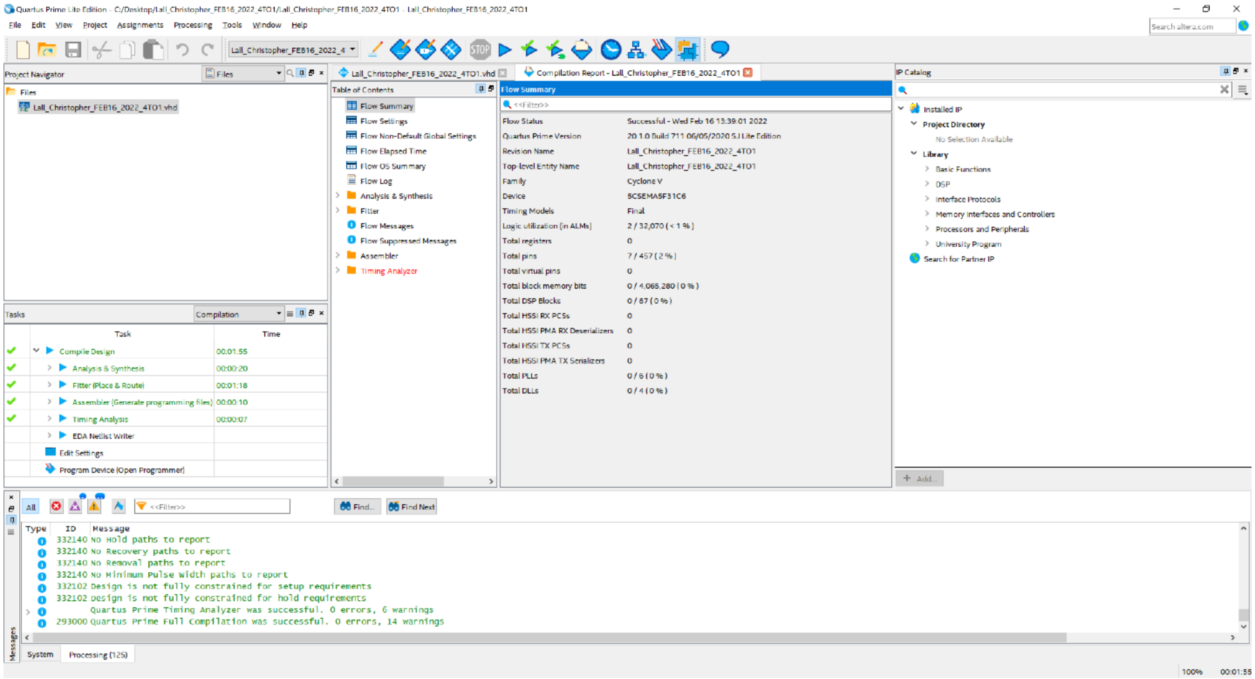

Wave - Default
Msgs
/lall_christopher_fe... 1
/lall_christopher_fe... 1
/lall_christopher_fe... 0
/lall_christopher_fe... 1

Figure 9. Another example

4 to 1 Mux



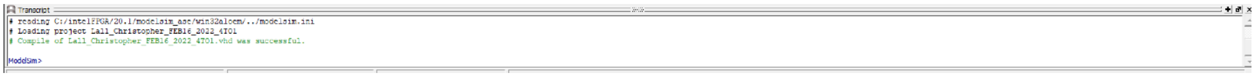
Flow Summary

Flow Status	Successful - Wed Feb 16 13:39:01 2022
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	Lall_Christopher_FEB16_2022_4TO1
Top-level Entity Name	Lall_Christopher_FEB16_2022_4TO1
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	2 / 32,070 (< 1 %)
Total registers	0
Total pins	7 / 457 (2 %)
Total virtual pins	0
Total block memory bits	0 / 4,065,280 (0 %)
Total DSP Blocks	0 / 87 (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 %)

	Task	Time
✓	Compile Design	00:01:55
✓	Analysis & Synthesis	00:00:20
✓	Fitter (Place & Route)	00:01:18
✓	Assembler (Generate programming files)	00:00:10
✓	Timing Analysis	00:00:07
	EDA Netlist Writer	
	Edit Settings	
	Program Device (Open Programmer)	

```
Type  ID     Message
      332140 No Hold paths to report
      332140 No Recovery paths to report
      332140 No Removal paths to report
      332140 No Minimum Pulse Width paths to report
      332102 Design is not fully constrained for setup requirements
      332102 Design is not fully constrained for hold requirements
             Quartus Prime Timing Analyzer was successful. 0 errors, 6 warnings
      293000 Quartus Prime Full Compilation was successful. 0 errors, 14 warnings
```

Figure 10. Compile in Quartus for Mux



```
# reading C:/intelFPGA/20.1/modelsim_ase/win32aloem/../modelsim.ini
# Loading project Lall_Christopher_FEB16_2022_4TO1
# Compile of Lall_Christopher_FEB16_2022_4TO1.vhd was successful.

ModelSim>
```

Figure 11. Compilation in ModelSim

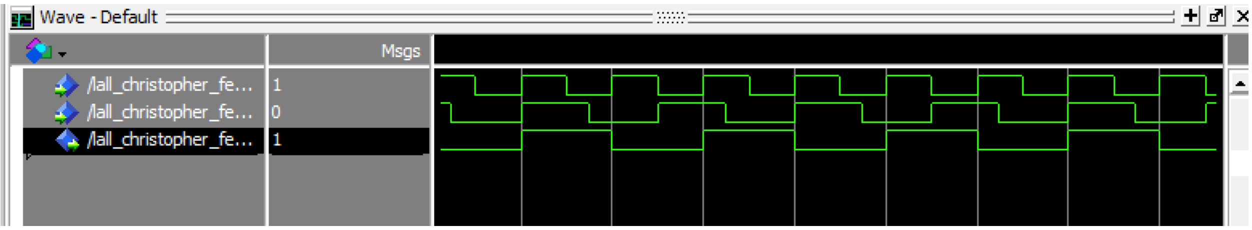


Figure 12. Waveform of 4:1 Mux

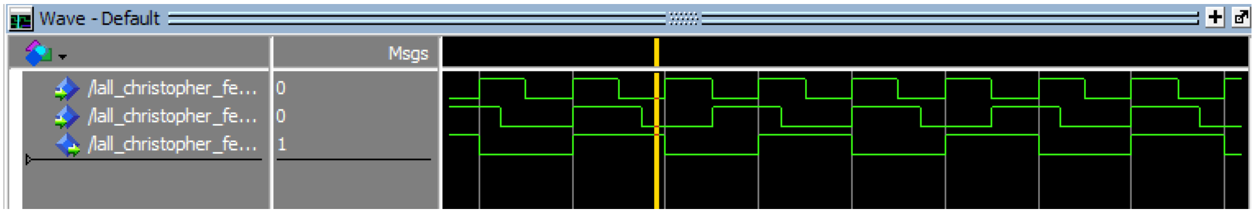
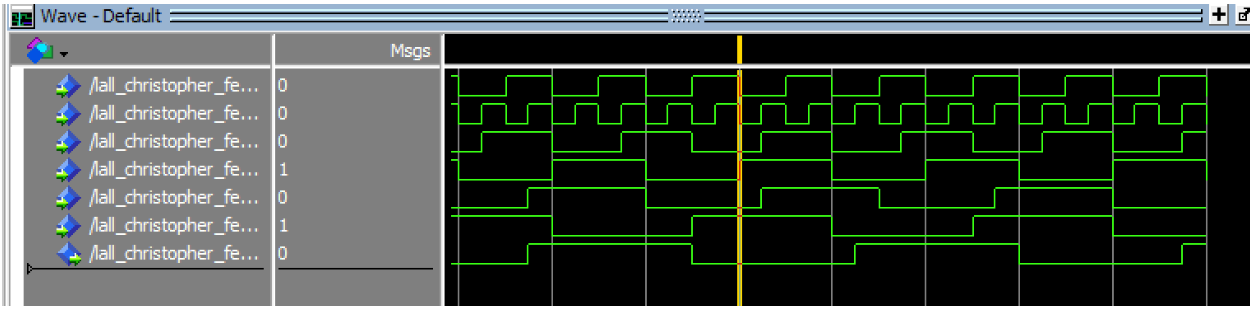
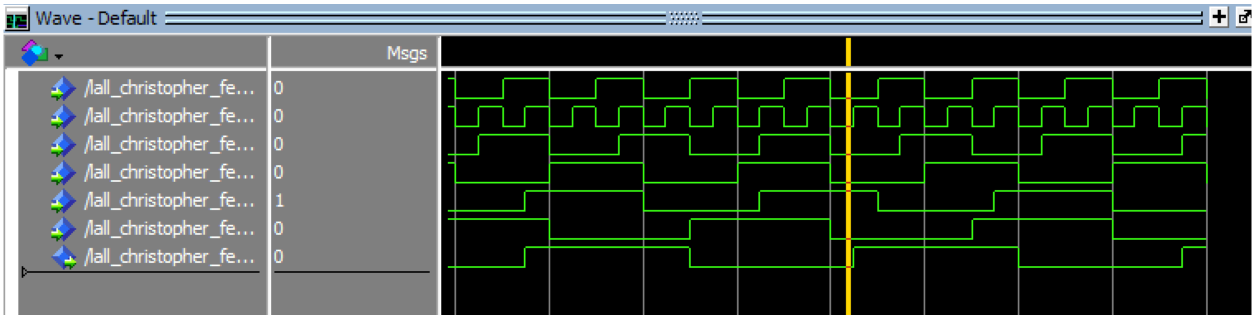


Figure 13. Example of MSGS



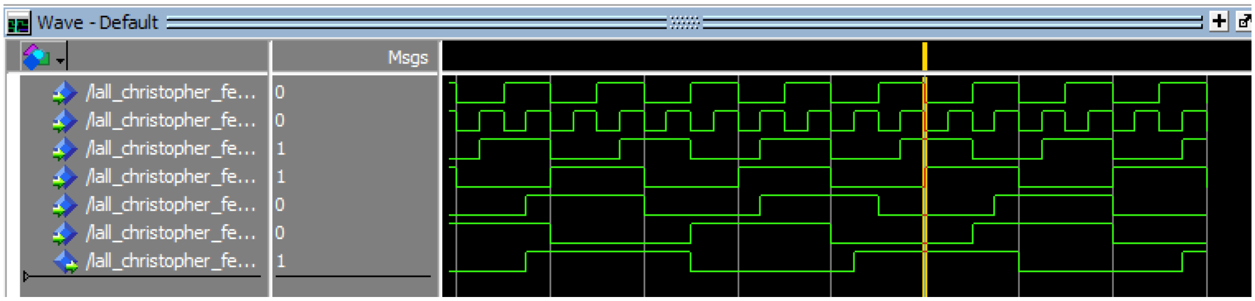
/lall_christopher_fe... 0
/lall_christopher_fe... 0
/lall_christopher_fe... 0
/lall_christopher_fe... 1
/lall_christopher_fe... 0
/lall_christopher_fe... 1
/lall_christopher_fe... 0

Figure 14. Verification example



/lall_christopher_fe... 0
/lall_christopher_fe... 0
/lall_christopher_fe... 0
/lall_christopher_fe... 0
/lall_christopher_fe... 1
/lall_christopher_fe... 0
/lall_christopher_fe... 0

Figure 15. Verification, s1, s0, a, output 0



/lall_christopher_fe... 0
/lall_christopher_fe... 0
/lall_christopher_fe... 1
/lall_christopher_fe... 1
/lall_christopher_fe... 0
/lall_christopher_fe... 0
/lall_christopher_fe... 1

Figure 16. Verification s0 0 s1 1 a 1 output 1

Explanation

The screenshots demonstrate how I was able to reach my goal of compiling the code and then importing it to modelsim to get waves.

Conclusion

From doing the tutorial from the previous week, I was able to compile and get waveforms from ModelSim.