Islamic University of Technology

Department of Electrical and Electronic Engineering

EEE 4483 – (Digital Electronics and Pulse Technique)

Introduction to VHDL and Modelsim

Introduction

The purpose of this lab is to give a first introduction to the VHDL based design flow for Altera FPGAs.

For code editing Emacs text editor will be used. After writing VHDL code for digital circuits we need to verify the program by testbenches. Up to this stage for compilation and to check the timing diagrams will use Modelsim.

Book:

1. Introduction to Logic Circuits & Logic Design with VHDL by Brock J. LaMeres

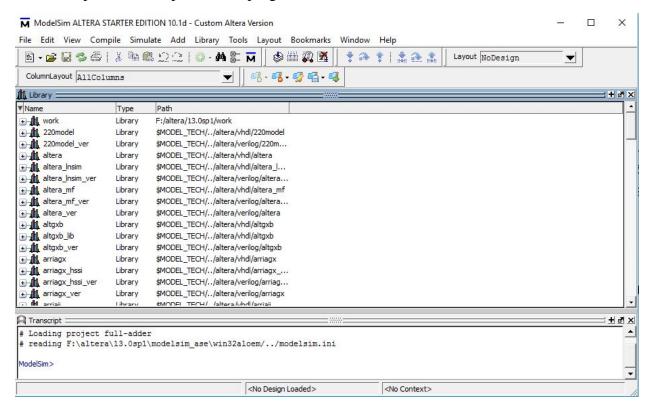
1st edition (2017)

Chapter 5 (section - 5.4, 5.5, 5.6, 5.7)

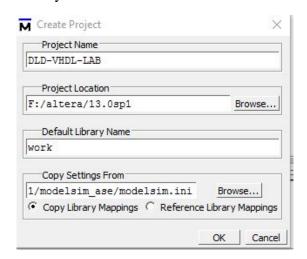
Download Links:

- 1. Quartus II Web Edition (Free), includes Quartus II Software and ModelSim-Altera Starter Edition
- 2. Emacs

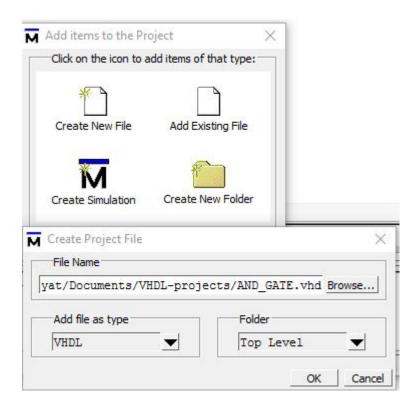
⇒ Steps to run/compile VHDL program in ModelSim:



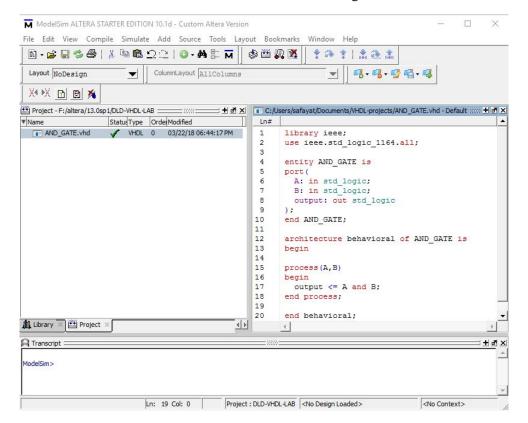
To create a new project in ModelSim: $File \rightarrow New \rightarrow Project$. The project will be saved in the default work library.



Now click on Create New File and show the appropriate VHDL file (*.vhd) from project directory.



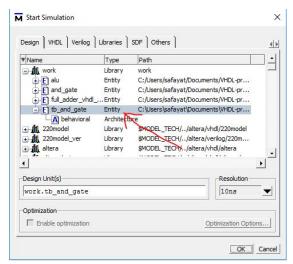
AND_GATE.vhd is the main source-code file of the design.



 \Rightarrow We need to write a testbench to test the design with various inputs.

```
ModelSim ALTERA STARTER EDITION 10.1d
File Edit View Compile Simulate Add Transcript Tools Layout Bookmarks Window Help
 🕸 🕮 🎉 🕴 🛊 💸 🛊 Layout NoDesign
                                                                                                   ▼ ColumnLayout AllColumns
 Project - F:/altera/13.0sp1/DLD-VHDL-LAB =
                                                             Status Type Order Modified
                                                                      Ln#
 -- Testbench to the AND_GATE.vhd
                                                                            library ieee;
                                                                            use ieee.std_logic_1164.all;
                                                                            entity tb AND GATE is
                                                                            end tb AND GATE;
                                                                            architecture behavioral of tb_AND_GATE is
                                                                       9
                                                                              component AND_GATE is
                                                                      10
                                                                               port (
                                                                      11
                                                                                 B: in std_logic;
output: out std_logic
                                                                      12
                                                                      13
                                                                      14
15
                                                                              end component;
                                                                      16
                                                                              signal A, B : std_logic := '0';
signal output : std_logic := '0';
                                                                      17
18
                                                                      19
20
                                                                      21
                                                                              uut: AND_GATE port map( -- Unit Under Test
                                                                                A => A,
B => B,
                                                                      23
                                                                      24
25
                                                                                output => output
                                                                                );
                                                                      26
                                                                      27
                                                                      28
                                                                              begin
                                                                                -- test 1
A <= '0';
B <= '0';
                                                                      29
30
                                                                      31
                                                                      32
                                                                                wait for 50 ns;
                                                                      33
                                                                                -- test 2
                                                                      34
35
                                                                               A <= '0';
B <= '1';
                                                                      36
                                                                      37
38
                                                                                -- test 3
A <= '1';
B <= '0';
                                                                      39
40
                                                                                wait for 50 ns;
                                                                      41
                                                                                A <= '1';
B <= '1';
                                                                      42
                                                                      43
                                                                      44
                                                                                wait for 50 ns;
                                                                      45
                                                                              end process;
                                                                            end architecture behavioral;
                                                                 () Untitled-1  tb_AND_GATE.vhd * ×
Library X Project X
# Compile of AND GATE. vhd was successful.
# Compile of tb AND GATE.vhd was successful.
# 2 compiles, 0 failed with no errors.
```

 \Rightarrow Now we will simulate the testbench and generate the timing diagram. Go to **Simulate** \Rightarrow **Start Simulation**. Set the Resolution as 10ns. Then proceed by hitting Ok.



⇒ A new set of windows will pop up with Objects, Processes, Scope(to see timing diagrams). From the Add select To Wave → All items in region. Next press F9 or click on Run.

