

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

Objective

Testbench for N bit adder / subtractor

To write testbench file in VHDL to verify correctness of N bit adder/ subtractor unit. You must verify the correctness for signed integers.

In addition

1. Testbench file has to verify if an operation sets correctly the OVERFLOW bit.
2. Testbench file has to verify if an operation sets correctly the ZERO bit.
3. Testbench file has to verify if an operation sets correctly the NEGATIVE bit.

Do the above For N=4, 16, 32 bits.

e.g. for N=4 we have 256 possible inputs.

To demonstrate that your test bench code is correct, you can introduce an error in your design file and show that your test bench code detects the error, prints out the input values, and the expected correct value, corresponding simulation time.

Enhanced Testbench for N bit comparator unit

1. Create a testbench file for comparator based on the template given here.
2. It is often required to locate a word stored in an array of size N. Each element in an array is 32 bit word. In order to locate the word in an array, you compare query word with every element in an array until you find it.

In VHDL, create an array of size N. Each Word in the array is 32 bit. Create a digital circuit in VHDL that locates the query Word.

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

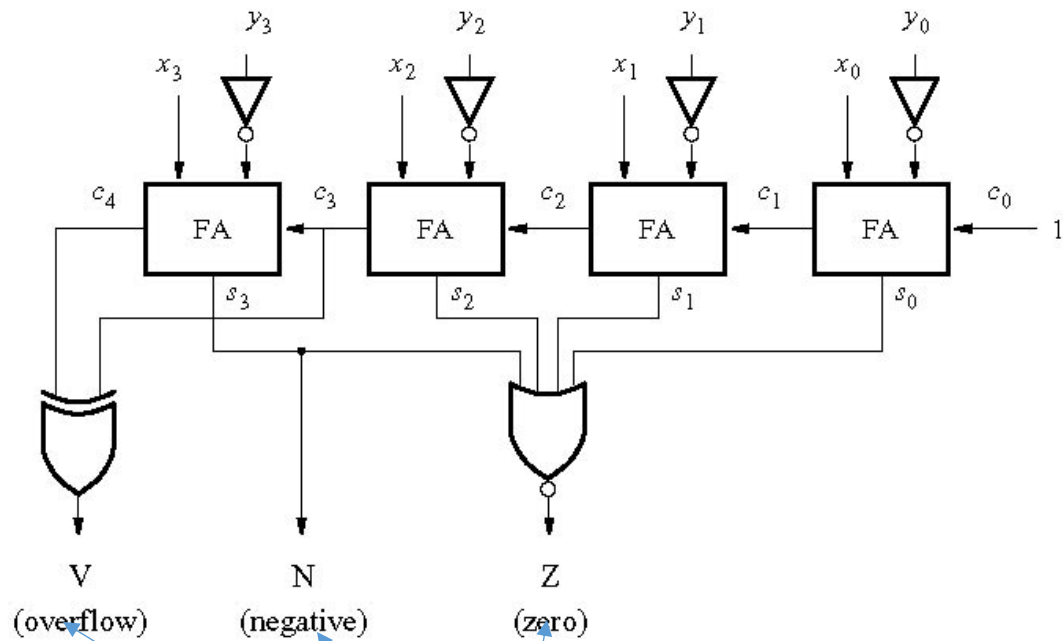


Figure 1. 4 Bit Comparator circuit. It outputs 4 FLAGS:
V,N,Z.

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

1. Template for test bench file for My4Bit adder using MODELSIM Tool

1.1 ModelSim installation(REVIEW):

ModelSim is a multi-language HDL simulation environment by Mentor Graphics, for simulation of hardware description languages such as VHDL and Verilog. Simulation is performed using the graphical user interface (GUI), or automatically using scripts.

For this part you must have ModelSim installed in your computer. Altera gives you the choice to download ModelSim as bundle when you download Quartus from their website.

- If you followed our tutorial on installing Quartus and ModelSim, you are ready to go and you can ignore the bullet point bellow. (Alternatively, if you haven't installed any of these programs, please follow our tutorial on Quartus and ModelSim download and setup, then come back to this lab).
- If you have Quartus installed but not ModelSim, you can download it independently from their website, just make sure you choose the same version of Quartus you already have in your computer before starting download. Go to:

http://dl.altera.com/?edition=subscription&product=modelsim_ae#tabs-2

The following shows the download page from Altera. It is important that you know which version of Quartus you have installed in your computer. If you don't know, open up Quartus, then go to **Help > About Quartus II**

CS 343, Spring 2021

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

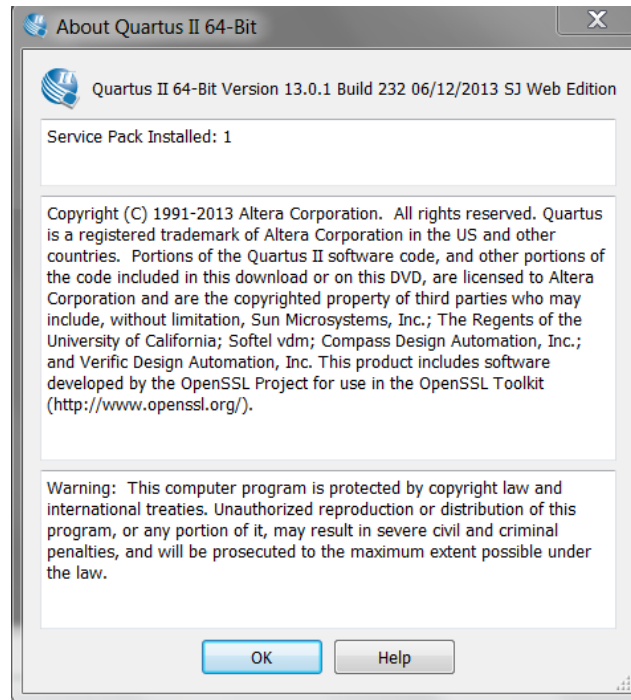


Figure 6. Screenshot of our Quartus version.
Our version is 13.0 service pack 1, yours may be different.

The next screenshot shows the download page for the Quartus software. Notice that at the top it says Subscription Edition although you may have downloaded the Free Web edition. This is irrelevant: it doesn't matter if it is part of the FREE or SUBSCRIPTION edition, ModelSim as a component is the same for both editions. Make sure you are logged into your Altera account (since you installed Quartus, we assume you have an account already with them).

CS 343, Spring 2021

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

Design Software

Embedded Software

Archives

Licensing

Programming Software

Drivers

Board System Design

Board Layout and Test

Legacy Software

Quartus II Subscription Edition

Release date: May, 2015
Latest Release: v15.0

Select release: 15.0

Operating System: Windows Linux

Download Method: Akamai DLM3 Download Manager Direct Download

✓ The Quartus II software version 15.0 supports the following device families: Arria II, Arria 10, Arria V, Arria V GZ, Cyclone IV, Cyclone V, MAX II, MAX V, MAX 10 FPGA, Stratix IV, and Stratix V. [More](#)

Choose Quartus version already installed in your computer

Choose your Operating System

Combined FilesIndividual FilesDVD FilesAdditional SoftwareUpdates

Download and install instructions: [More](#)
[Read Altera Software v15.0 Installation FAQ](#)
[Quick Start Guide](#)

☒ Select All

Quartus II Subscription Edition

☐ Quartus II Software (includes Nios II EDS)
Size: 1.8 GB MD5: 8527AE8C93F89153E82E83975D453E51

☒ ModelSim-Altera Edition (includes Starter Edition)
Size: 1.1 GB MD5: 7413FDF22BE9D84E5A6B7B2B524CCED0

Devices

You must install device support for at least one device family to use the Quartus II software.

☐ Arria II device support
Size: 664.8 MB MD5: 785C9A0BF694590DE11589769B522FA1

☐ Arria 10 device support i

☐ Arria 10 device support Part 1
Size: 2.7 GB MD5: 7EE28ADA59DE580AABECD1F48A1D6929

☐ Arria 10 device support Part 2
Size: 3.4 GB MD5: 5381FC37A11F686B84EA0C53582DCE6B

☐ Arria 10 device support Part 3
Size: 2.7 GB MD5: 05C480999A64EABFA26E7780C0BEAB8D

☐ Arria V device support
Size: 1.3 GB MD5: 273AD4B5D3801612D019FB549671DF34

☐ Arria V GZ device support
Size: 1.9 GB MD5: 21E04667CEF54DDD346D1E1A6A6D1668

☐ Cyclone IV device support
Size: 463.9 MB MD5: 49C3B14231152085309E076717A7044D

☐ Cyclone V device support
Size: 1.1 GB MD5: DF0EEE4512E0F3037438C037AFDEAF41

☐ MAX II, MAX V device support
Size: 11.3 MB MD5: F5D177113877FB8EA5B5E20ADA365500

☐ MAX 10 FPGA device support
Size: 295.1 MB MD5: 732AF29B714D339142936E978833CBFE

☐ Stratix IV device support
Size: 535.0 MB MD5: DA8835EF1C24359665C59A4B9095FB4B

☐ Stratix V device support
Size: 2.8 GB MD5: 12B5B90DD49F731B7451966B85CE5927

Make sure this is checked

Click download

Download Selected Files

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

Figure 7. ModelSim download page

1.2 ModelSim basics (REVIEW):

1. Open ModelSim. To run ModelSim, go to terminal and type vsim. This tutorial was done on a Windows environment, so we will show the corresponding Windows system screenshots but other operating systems should follow similar steps.

If you are on Windows go to **Start > Run**. Type in “cmd” (without quotes) in the input field, then hit **Enter**. Alternatively, you could also search for Command Prompt in the search field in the Start Menu.

If you are on Linux, right click your desktop and click Terminal.

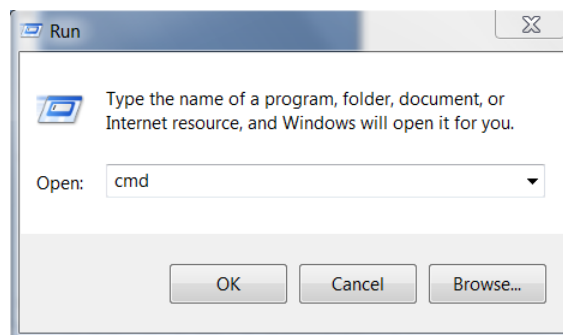


Figure 8. Run Command in Windows

2. In terminal type “vsim” (without the quotes).

CS 343, Spring 2021

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

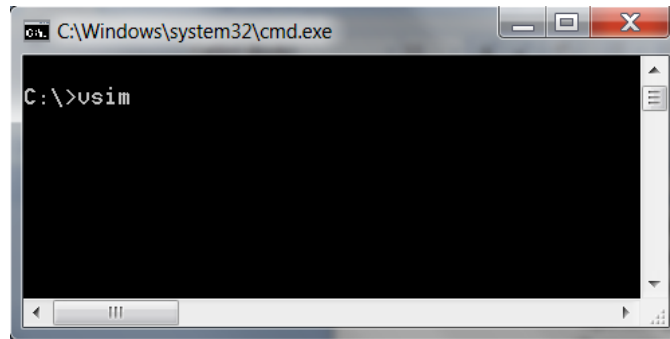


Figure 9. Terminal and ModelSim start command

You are now taken to the main screen of ModelSim, a Welcome splash screen will appear and you are now ready to start testing your circuit designs.

The initial screen of ModelSim is shown below:

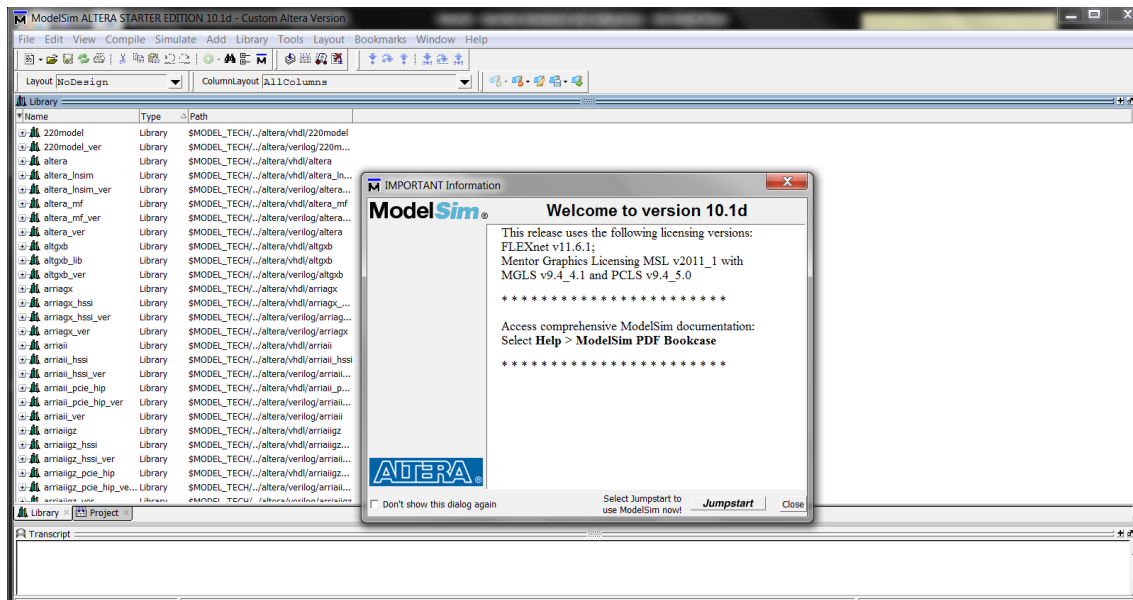


Figure 10. ModelSim Welcome screen

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

3. When the Welcome screen appears, hit Close. Now go to **File > New > Project**. If a message appears asking if you want to close the current project just accept.
4. The Create New Project dialog appears, enter a name for your project, you can call it **Test_Bench_LastName_My4Bit adder**. See figure 11.
5. Browse a location for your new project.
6. In the Default Library Name, if empty, call it **work**.

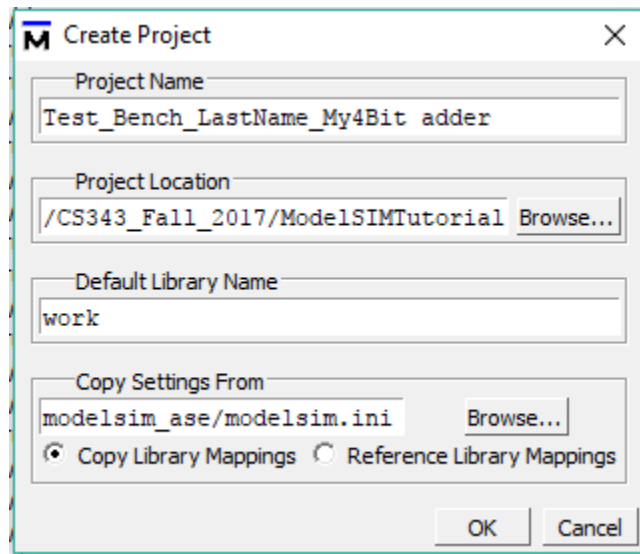


Figure 11. Create New Project dialog in ModelSim

7. The Add items to project dialog appears (See figure 12). In the previous lab we described the 4 bit adder and all components we are going to implement test_bench and we gave you the VHDL code for it. Right now we are going to create a new VHDL file and copy-paste the given code to it. Click **Create New File**.

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

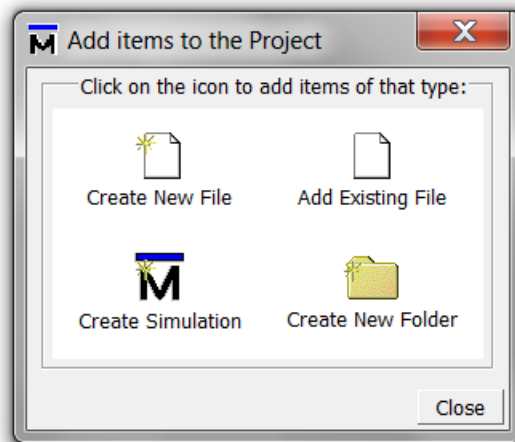


Figure 12. Add items dialog

In the same directory where you design files are stored, CREATE a TESTEBENCH file for 4 bit adder following example below.

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

VHDL CODE for test_bench_4BIT adder

```
--test_adder4pkg.vhd

library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
use ieee.std_logic_unsigned.all;
USE work.fulladd_package.all ;

entity test_my4add is
end test_my4add;

architecture arch_test of test_my4add is
--componengt declaration for the Unit Under Test
component my4add
    PORT (Cin: IN STD_LOGIC;
        X,Y: IN STD_LOGIC_VECTOR(3 DOWNTO 0);
        Sout: OUT STD_LOGIC_VECTOR(3 DOWNTO 0);
        Cout: OUT STD_LOGIC );
end component;

    signal A,B,S :STD_LOGIC_VECTOR(3 DOWNTO 0);
    signal Ci,Co :STD_LOGIC;
begin
    -----Instantiate the Unit Under Test (UUT)
    uut: my4add port map (
        Cin => Ci,
        X    => A,
```

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

```

        Y    => B,
        Sout  => S,
        Cout => Co
    );
---- Test Bench ---User Defined Process
tb : process
begin
--Hold reset state for 100 ns
wait for 100 ns;
report "Hello Simulator";
A<="0000";
B<="0000";
Ci<='0';
--Loop over all values of A
for I in 0 to 2 loop
--Loop over all values of B
    for J in 0 to 2 loop
        --Wait for output to update
        wait for 10 ns;

--report " the A+B = " & integer'image(to_integer(unsigned((A+B))));
--The statement below checks for ALL possible input values if the output is correct.
assert (S = A+B) report "The sum from 4 bit adder is S= " &
integer'image(to_integer(unsigned((S)))) &
" while the expected A+B = " &
integer'image(to_integer(unsigned((A+B)))) severity ERROR;
--Increment to the next value of B
B<=B+"0001";
end loop;
--Increment to the next value of A
A<=A+"0001";

--Echo to users test is finished
end loop;
```

CS 343, Spring 2021

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

```
report "Test completed";  
wait; -- will wait for ever  
end process;  
---END User Defined Process  
end arch_test;
```

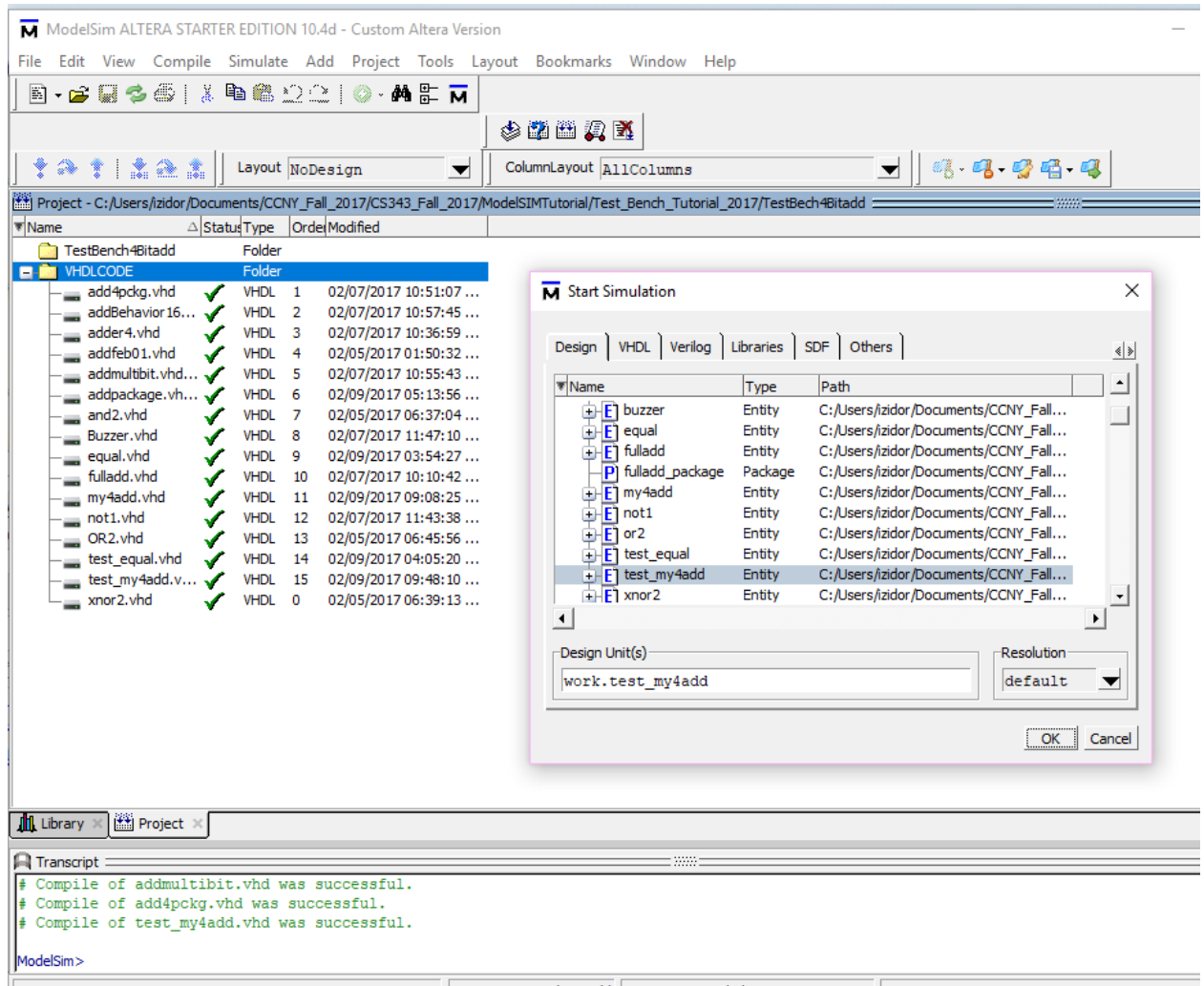
8. Create all VHDL code for components and testBecnh file,
compile all.
Select Start Simulation->Start Simulation

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021



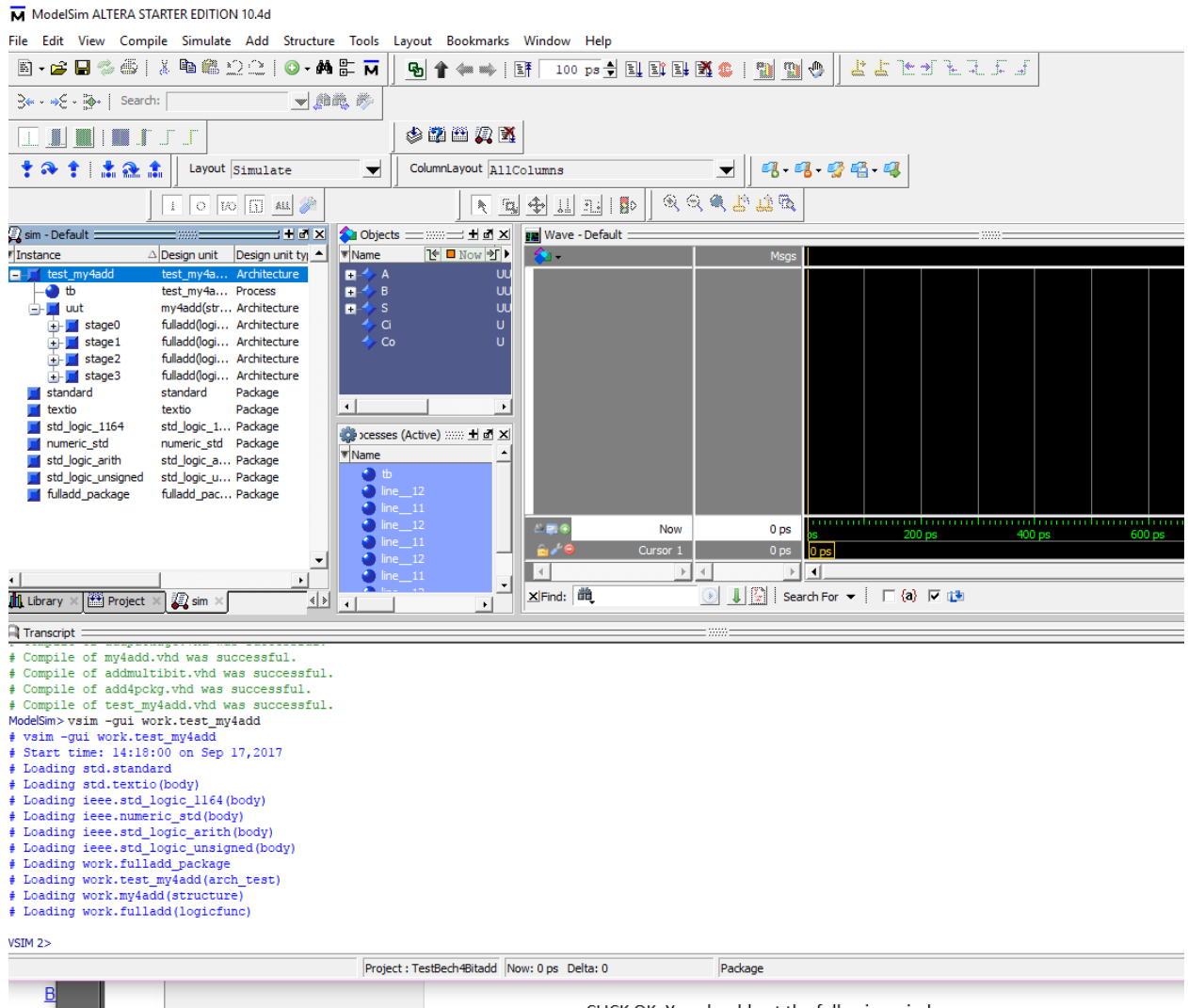
CLICK OK. You should get the following window

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021



CLICK OK. You should get the following window

In the figure UUT stands for Unit Under Test.

Now you ready to run simulation.

9. Simlutate->RUN and select one
 - a. Run 100
 - b. Run next

Self-Check 3B

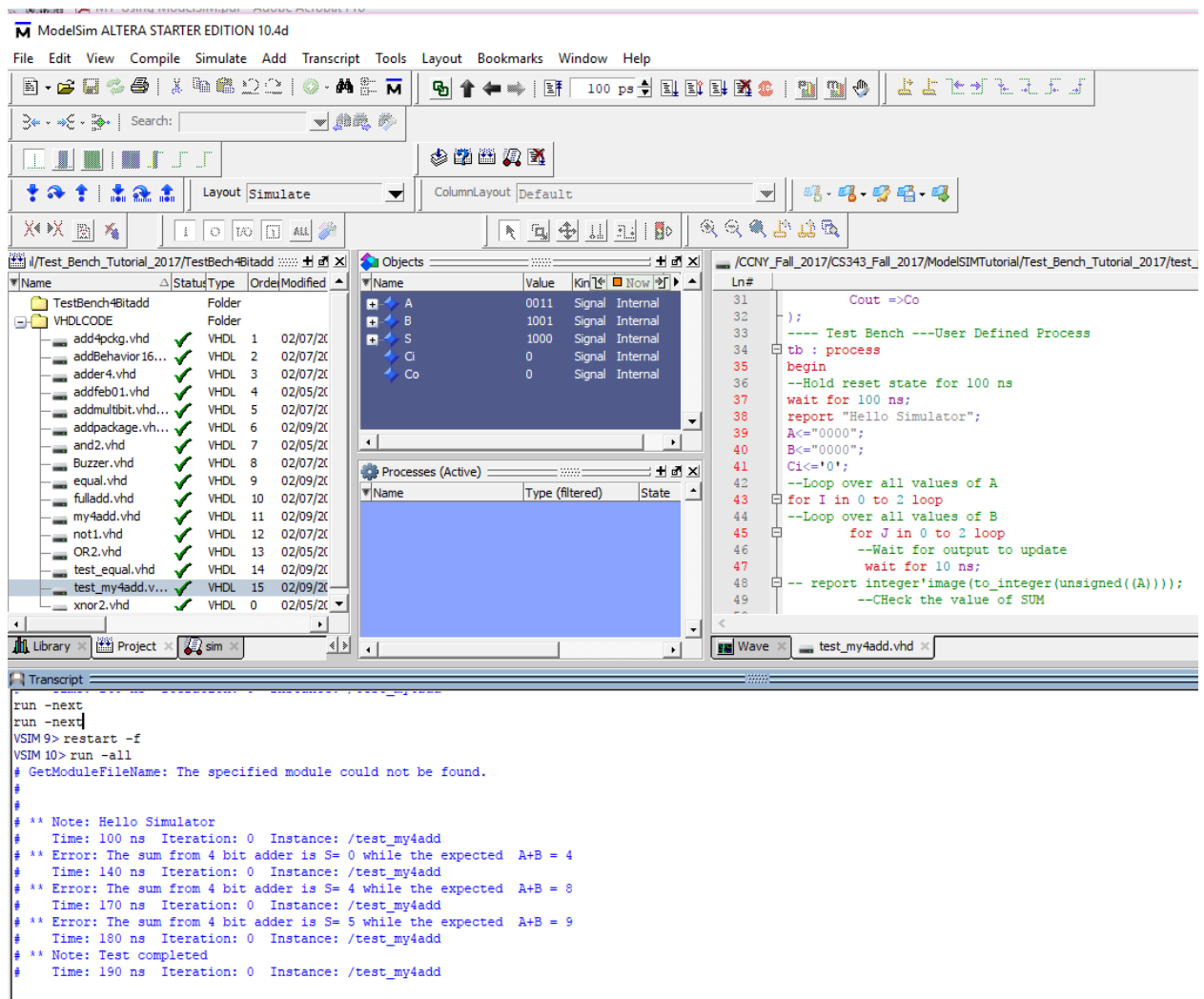
TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

- c. Run all
 - d. Continue
10. In my example I run Run All.

Errors are detected



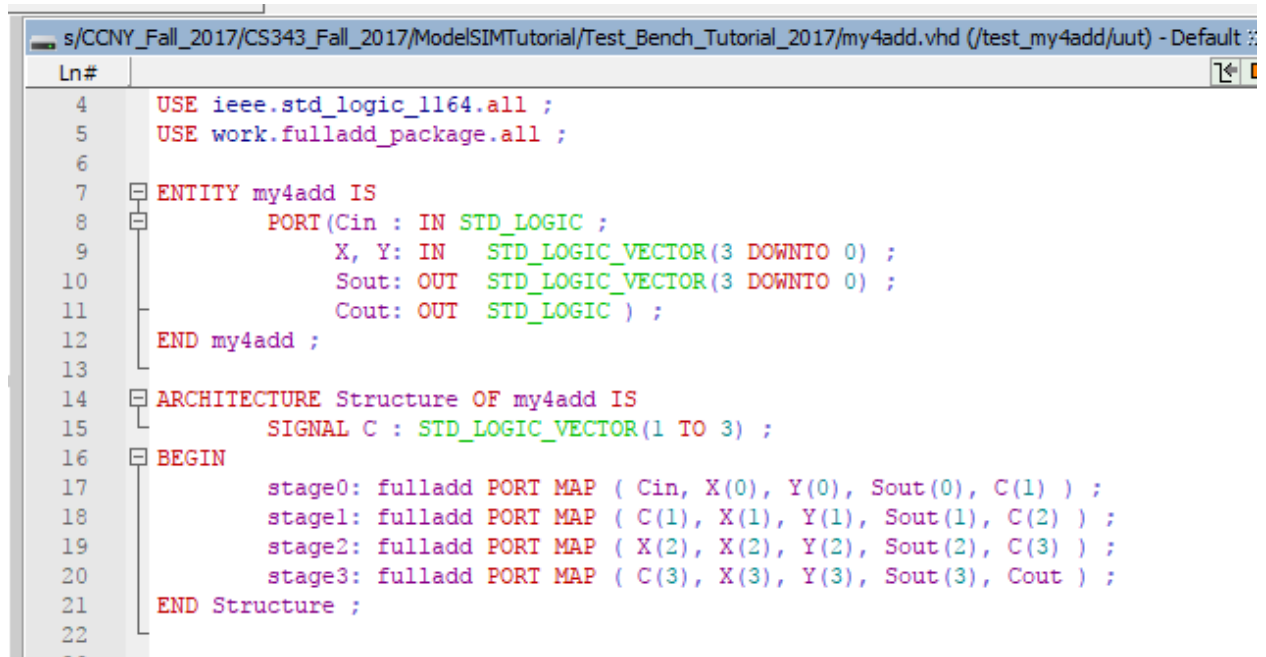
To find an error I examine my design file my4bitadder.vhdl

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021



```
Ln# | s/CCNY_Fall_2017/CS343_Fall_2017/ModelSIMTutorial/Test_Bench_Tutorial_2017/my4add.vhd (/test_my4add/uut) - Default
4   | USE ieee.std_logic_1164.all ;
5   | USE work.fulladd_package.all ;
6   |
7   | ENTITY my4add IS
8   |     PORT (Cin : IN STD_LOGIC ;
9   |           X, Y: IN  STD_LOGIC_VECTOR(3 DOWNTO 0) ;
10  |           Sout: OUT STD_LOGIC_VECTOR(3 DOWNTO 0) ;
11  |           Cout: OUT STD_LOGIC ) ;
12  | END my4add ;
13  |
14  | ARCHITECTURE Structure OF my4add IS
15  |     SIGNAL C : STD_LOGIC_VECTOR(1 TO 3) ;
16  | BEGIN
17  |     stage0: fulladd PORT MAP ( Cin, X(0), Y(0), Sout(0), C(1) ) ;
18  |     stage1: fulladd PORT MAP ( C(1), X(1), Y(1), Sout(1), C(2) ) ;
19  |     stage2: fulladd PORT MAP ( X(2), X(2), Y(2), Sout(2), C(3) ) ;
20  |     stage3: fulladd PORT MAP ( C(3), X(3), Y(3), Sout(3), Cout ) ;
21  | END Structure ;
22  |
23  |
```

I detect a typo in stage2. Carry in is signal X(2) and it should be C(2).

Easy fix. Recompile only myadd4bit.vhdl file.

You have to RESTART the simulation.

Then Run all

We get simulation completed, No errors!

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

The screenshot displays the ModelSim ALTERA STARTER EDITION 10.4d interface. The top menu bar includes File, Edit, View, Compile, Simulate, Add, Process, Tools, Layout, Bookmarks, Window, and Help. The toolbar contains various icons for file operations, simulation control, and viewing. The main workspace is divided into several panes:

- Instance:** A tree view showing the design hierarchy. The selected instance is 'test_my4add', which is an Architecture. Other instances include 'tb' (Process), 'my4add(str...)' (Architecture), and various standard packages like 'std_logic_1164', 'numeric_std', 'std_logic_arith', 'std_logic_unsigned', and 'fulladd_package'.
- Objects:** A table listing signals and their values. The signals are A (0011), B (1001), S (1100), Ci (0), and Co (0). All are of kind 'Signal' and 'Internal'.
- Processes (Active):** A table showing active processes. The table has columns for Name, Type (filtered), and State. It is currently empty.
- Wave - Default:** A pane for viewing waveforms, currently showing a blank area.
- Transcript:** A text area at the bottom showing the simulation log. It includes the command '# vsim -gui work.test_my4add', the start time '14:54:10 on Sep 17, 2017', and the results of the simulation run.

The Transcript pane shows the following text:

```
# vsim -gui work.test_my4add
# Start time: 14:54:10 on Sep 17, 2017
# Loading std.standard
# Loading std.textio(body)
# Loading ieee.std_logic_1164(body)
# Loading ieee.numeric_std(body)
# Loading ieee.std_logic_arith(body)
# Loading ieee.std_logic_unsigned(body)
# Loading work.fulladd_package
# Loading work.test_my4add(arch_test)
# Loading work.my4add(structure)
# Loading work.fulladd(logicfunc)
VSIM2> run -all
# ** Note: Hello Simulator
# Time: 100 ns Iteration: 0 Instance: /test_my4add
# ** Note: Test completed
# Time: 190 ns Iteration: 0 Instance: /test_my4add
VSIM3>
```

CS 343, Spring 2021

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

As we can see no errors.

In the objects window we inspect A=0011 (+3). B= 1001 (-7).

Adder output S= 1100 (-4). The expected output $A+B=0011 + 1001 = 1100$.

Self-Check 3B

TUTORIAL Introduction to Test_bench VHDL using ModelSim software tools using as an example digital N bit ADDER / Subtractor circuits

Instructor: Professor Izidor Gertner

Spring Semester , 2021

ModelSim - INTEL FPGA STARTER EDITION 10.5b

File Edit View Compile Simulate Add Project Tools Layout Bookmarks Window Help

Layout: NoDesign ColumnLayout: AllColumns

Project - C:/Users/izidor/Documents/CCNY_SPRING_2017/CS343_Spring2017/Intro toVHDL/My_comp_Feb_01/TestAd

Name	Status	Type	Order	Modified
add4pkg.vhd	✓	VHDL	9	02/07/2017 10:51:07 ...
addBehavior16bit...	✓	VHDL	11	02/07/2017 10:57:45 ...
adder4.vhd	✓	VHDL	7	02/07/2017 10:36:59 ...
addfeb01.vhd	✓	VHDL	0	02/05/2017 01:50:32 ...
addmultibit.vhd	✓	VHDL	10	02/07/2017 10:55:43 ...
addpackage.vhd	✓	VHDL	8	02/09/2017 05:13:56 ...
and2.vhd	✓	VHDL	1	02/05/2017 06:37:04 ...
Buzzer.vhd	✓	VHDL	5	02/07/2017 11:47:10 ...
equal.vhd	✓	VHDL	12	02/09/2017 03:54:27 ...
fulladd.vhd	✓	VHDL	6	02/07/2017 10:10:42 ...
my4add.vhd	✓	VHDL	14	02/09/2017 09:08:25 ...
not1.vhd	✓	VHDL	4	02/07/2017 11:43:38 ...
OR2.vhd	✓	VHDL	2	02/05/2017 06:45:56 ...
test_equal.vhd	✓	VHDL	13	02/09/2017 04:05:20 ...
test_my4add.vhd	✓	VHDL	15	02/09/2017 09:48:10 ...
xnor2.vhd	✓	VHDL	3	02/05/2017 06:39:13 ...

Library Project

Transcript

```
# Compile of my4add.vhd was successful.  
# Compile of test_my4add.vhd was successful.  
# 16 compiles, 0 failed with no errors.
```

ModelSim>

Project : TestAdder <No Design Loaded>