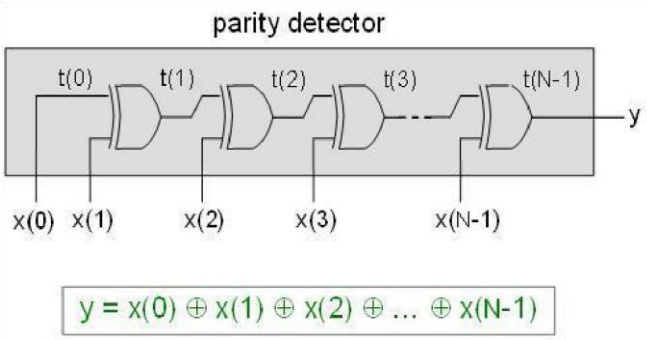
Adithya Alva
DMD

283864587
Assignment 2

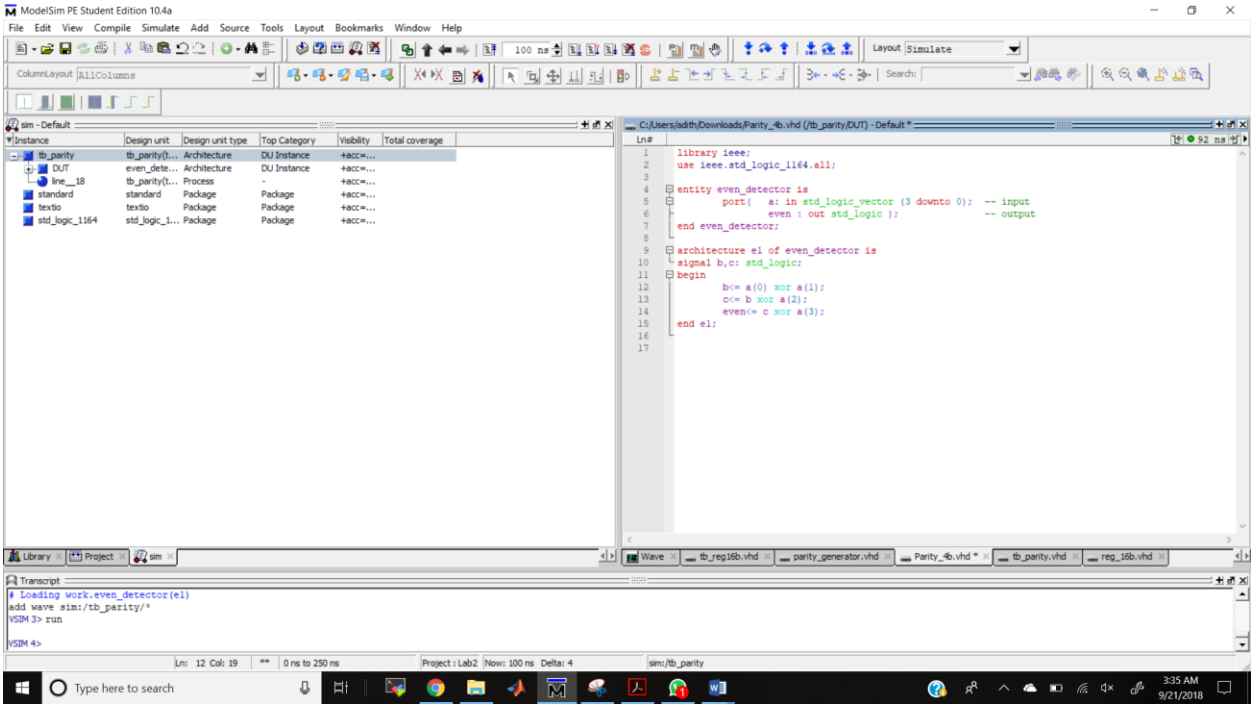
Parity Checker

A parity check is the process that ensures accurate data transmission between nodes during communication. Parity checker output = the xor operation of all the inputs.

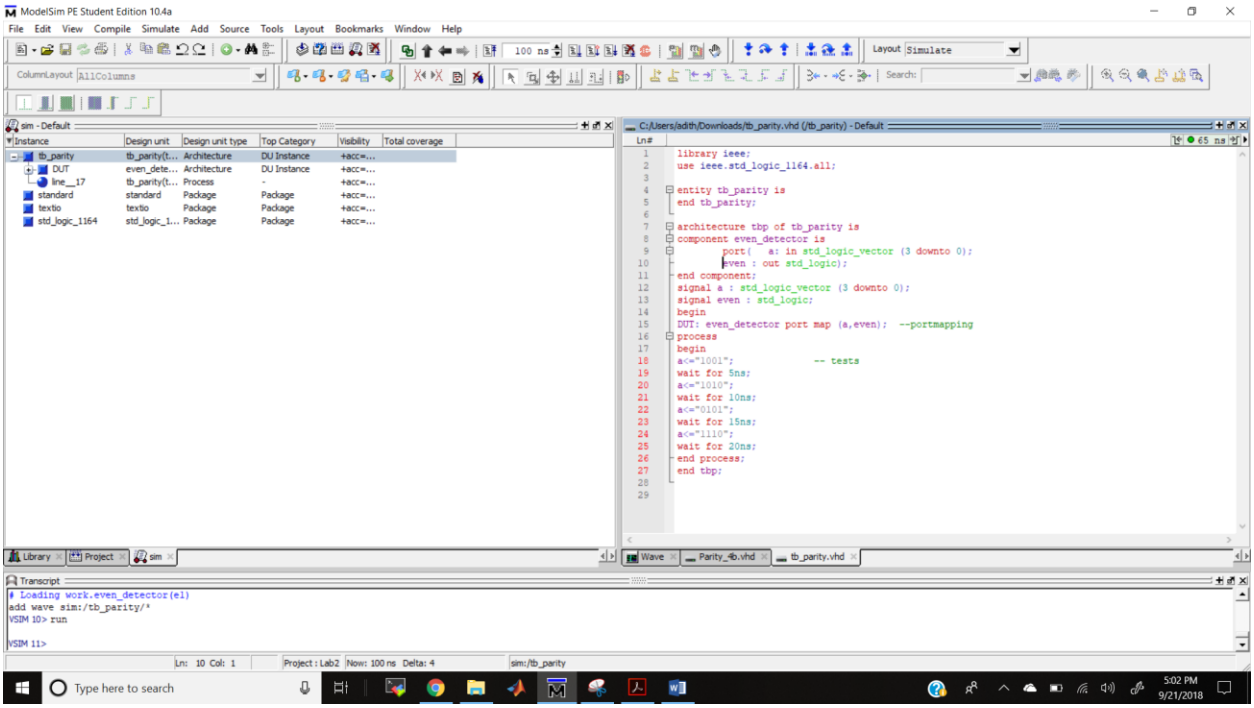
The same can be seen in the code where a(0) xor a(1) xor a(2) and a(3)



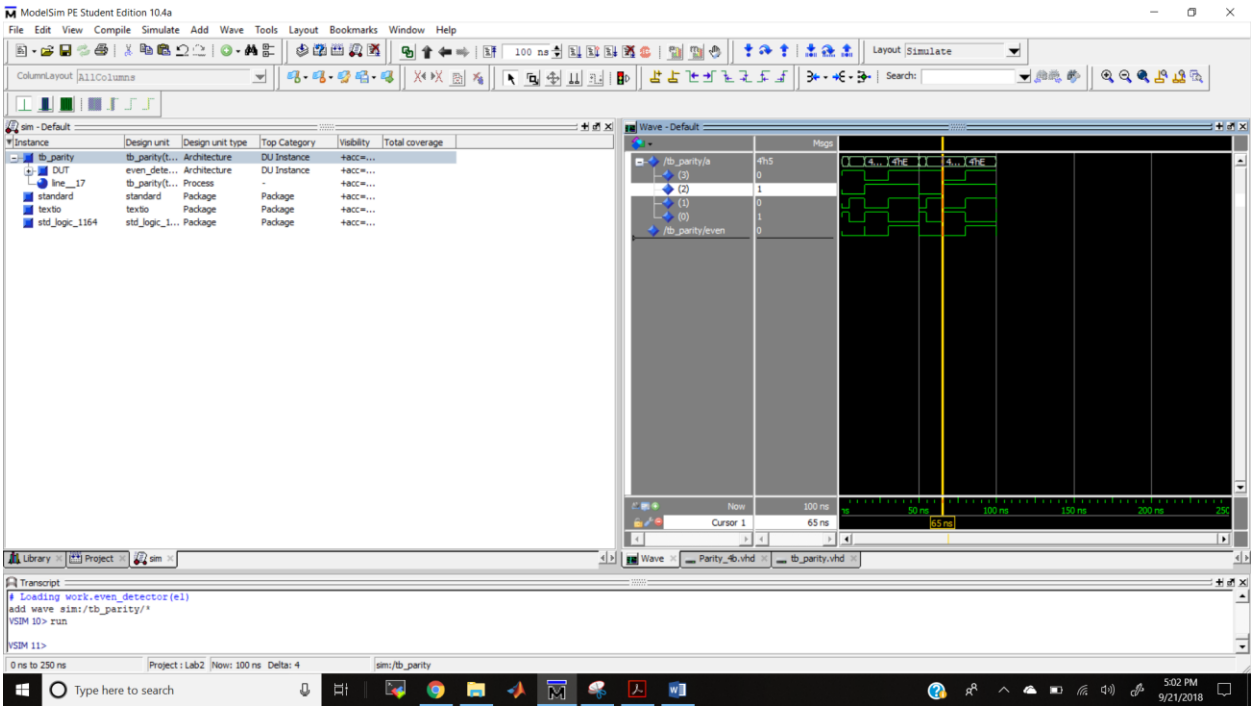
Source Code



TestBench


ModelSim PE Student Edition 10.4a
File Edit View Compile Simulate Add Source Tools Layout Bookmarks Window Help
sim - Default
Instance
Design unit
Design unit type
Top Category
Visibility
Total coverage
tb_parity
DUT
line__17
standard
textio
std_logic_1164
C:/Users/adith/Downloads/tb_parity.vhd (/tb_parity) - Default
library ieee;
use ieee.std_logic_1164.all;
entity tb_parity is
end tb_parity;
architecture tbp of tb_parity is
component even_detector is
port(a: in std_logic_vector (3 downto 0);
even : out std_logic);
end component;
signal a : std_logic_vector (3 downto 0);
signal even : std_logic;
begin
DUT: even_detector port map (a,even); --portmapping
process
begin
a<="1001"; -- tests
wait for 5ns;
a<="1010";
wait for 10ns;
a<="0101";
wait for 15ns;
a<="1110";
wait for 20ns;
end process;
end tbp;
Transcript
Loading work.even_detector(el)
add wave sim:/tb_parity/*
VSIM 10> run
VSIM 11>
Type here to search
5:02 PM
9/21/2018

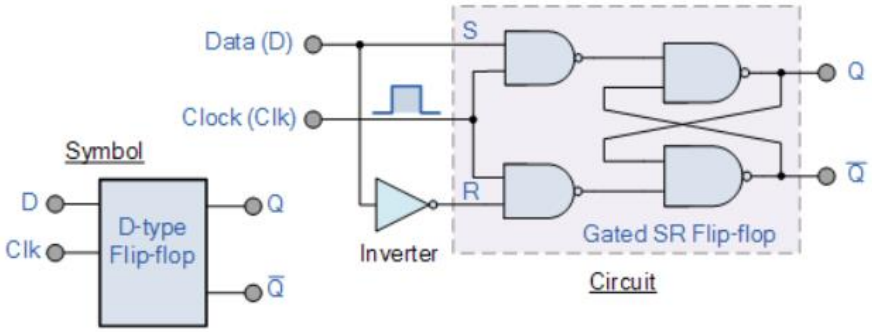
Waveform


ModelSim PE Student Edition 10.4a
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Wave - Default
/tb_parity/a
/tb_parity/even
Now
Cursor 1
65 ns
Loading work.even_detector(el)
add wave sim:/tb_parity/*
VSIM 10> run
VSIM 11>
Type here to search
5:02 PM
9/21/2018

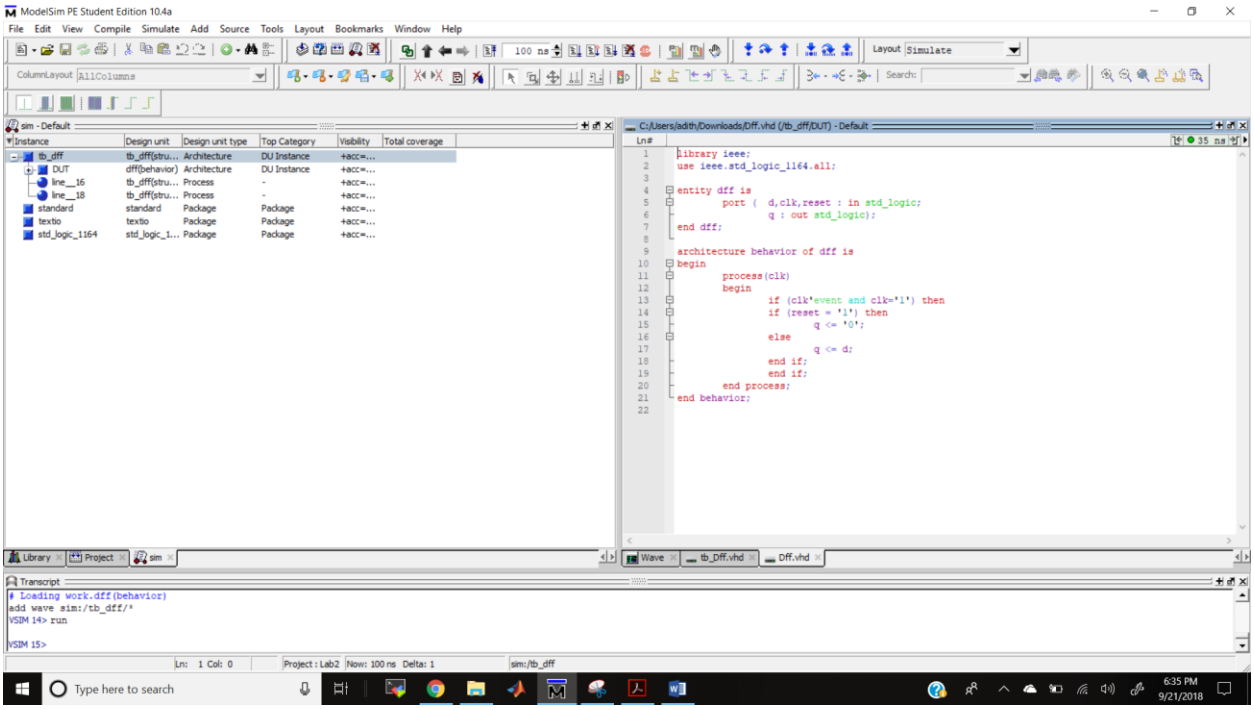
Adithya Alva 283864587

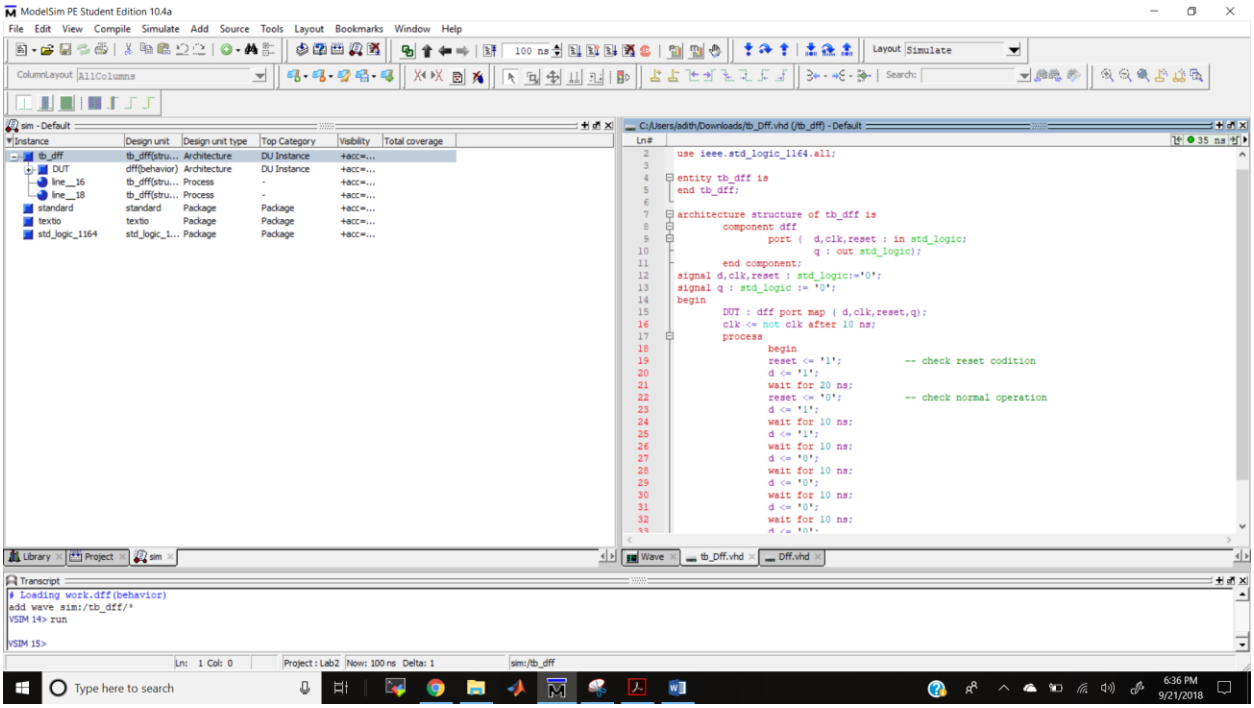
D flipflop

The “D flip flop” will store and output whatever logic level is applied to its data terminal so long as the clock input is HIGH.

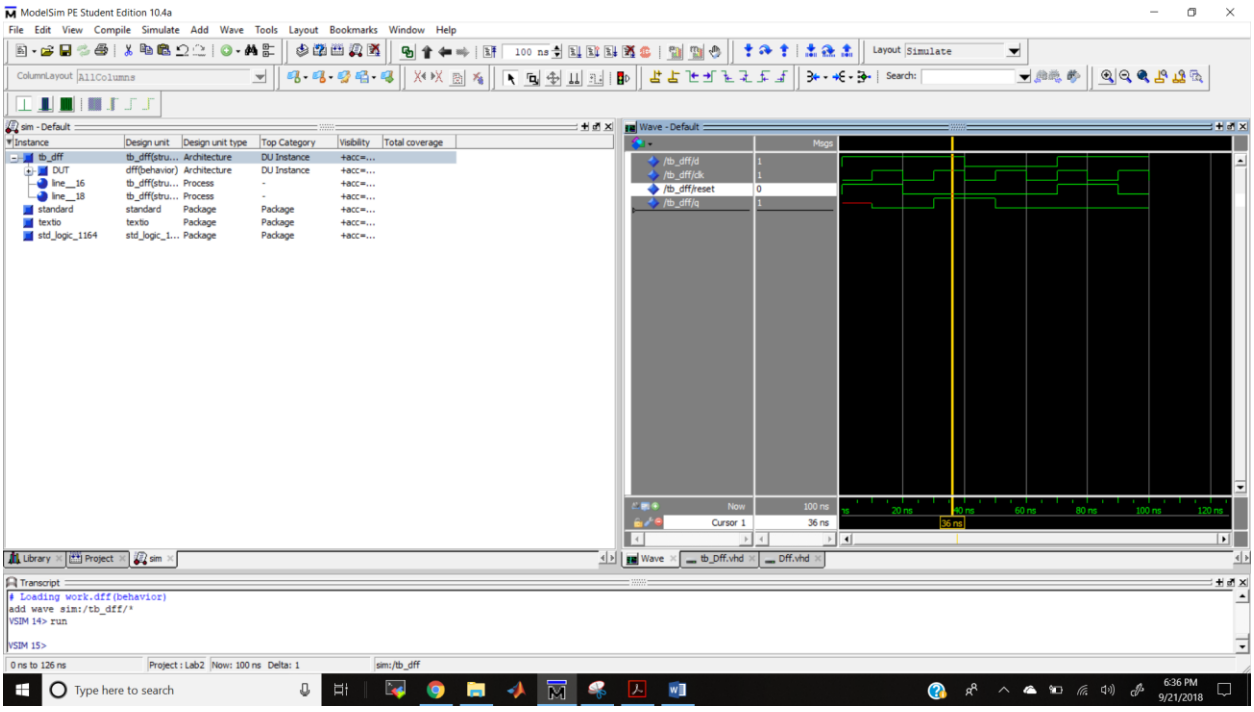


Source code





Waveform



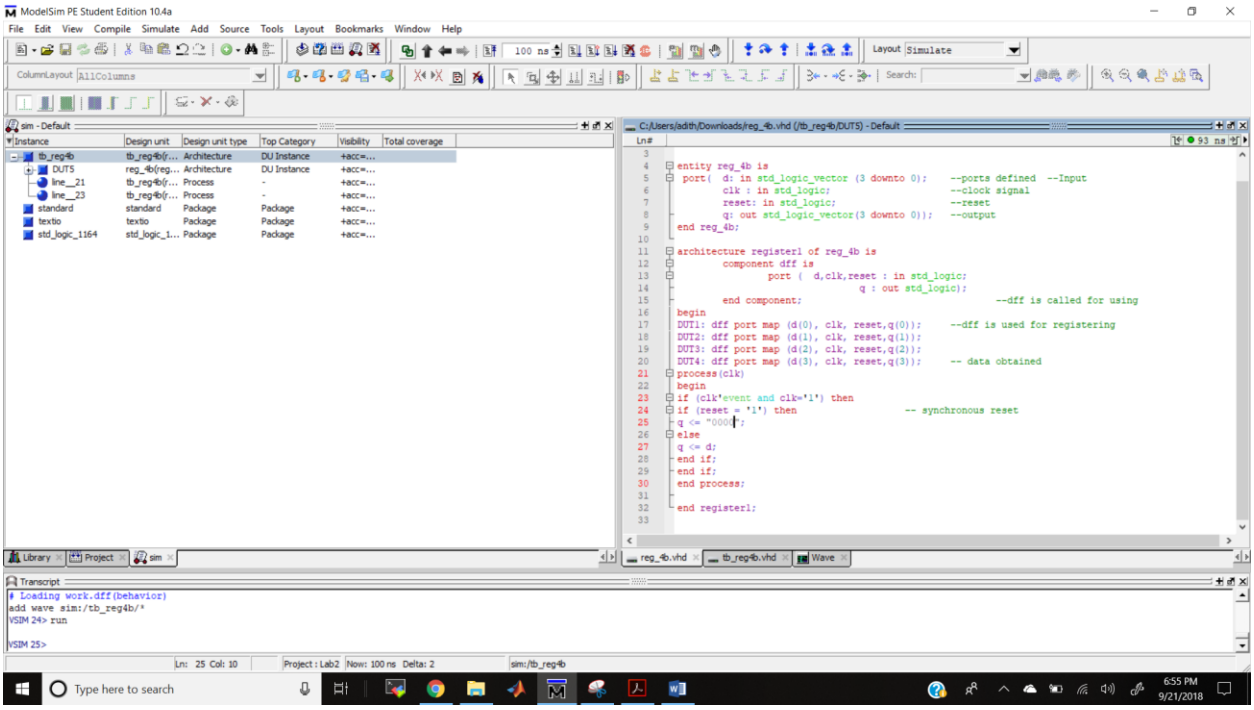
Register – 4Bit

Registers are used for data storage or for the movement of data and are therefore commonly used inside calculators or computers to store data

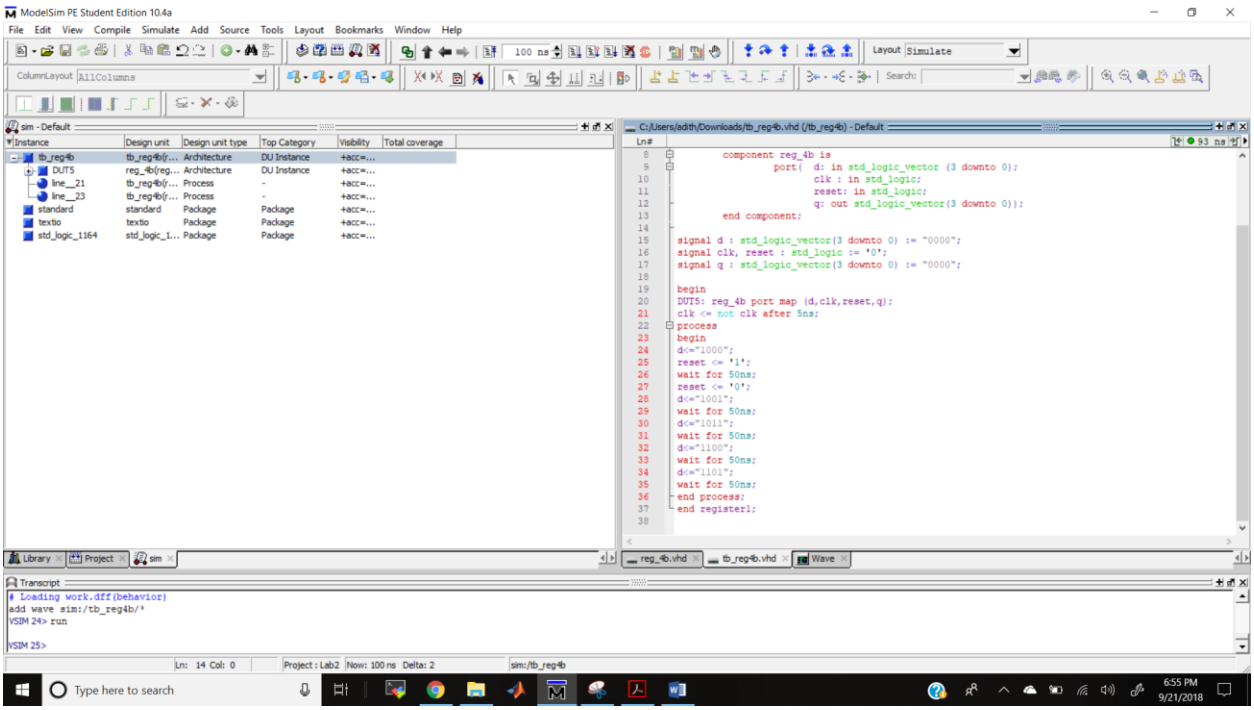
Adithya Alva
DMD

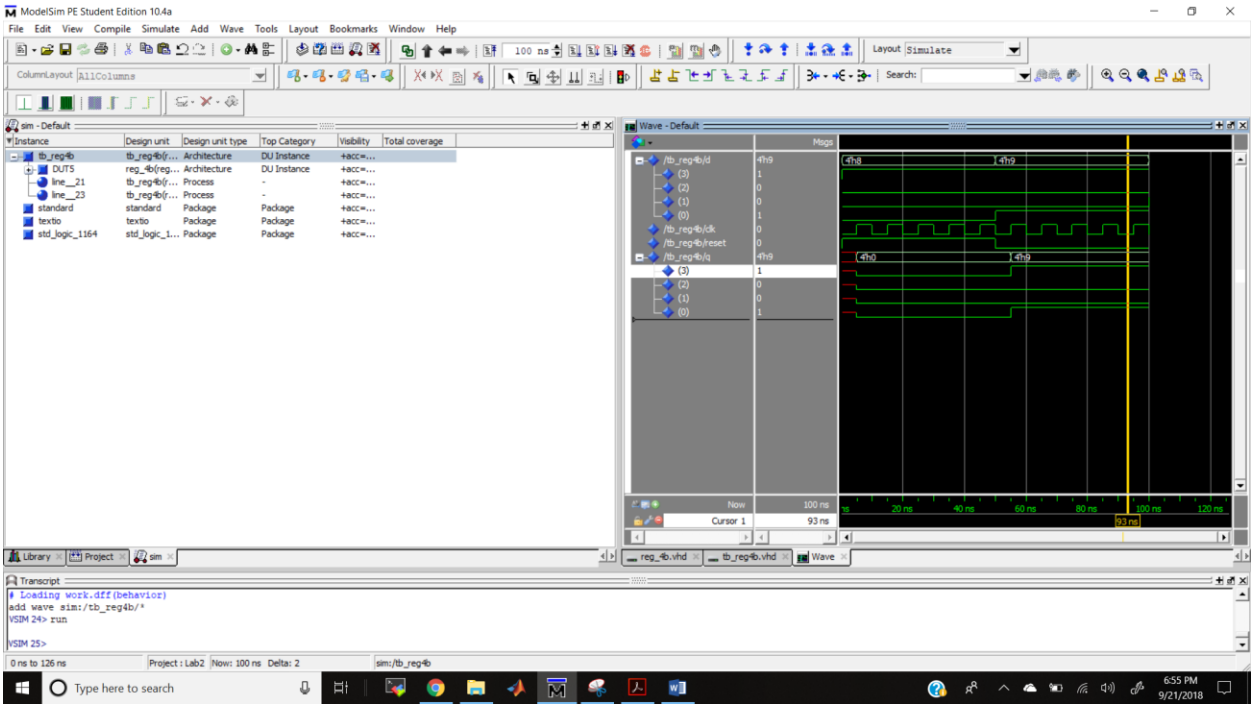
283864587
Assignment 2

Source code

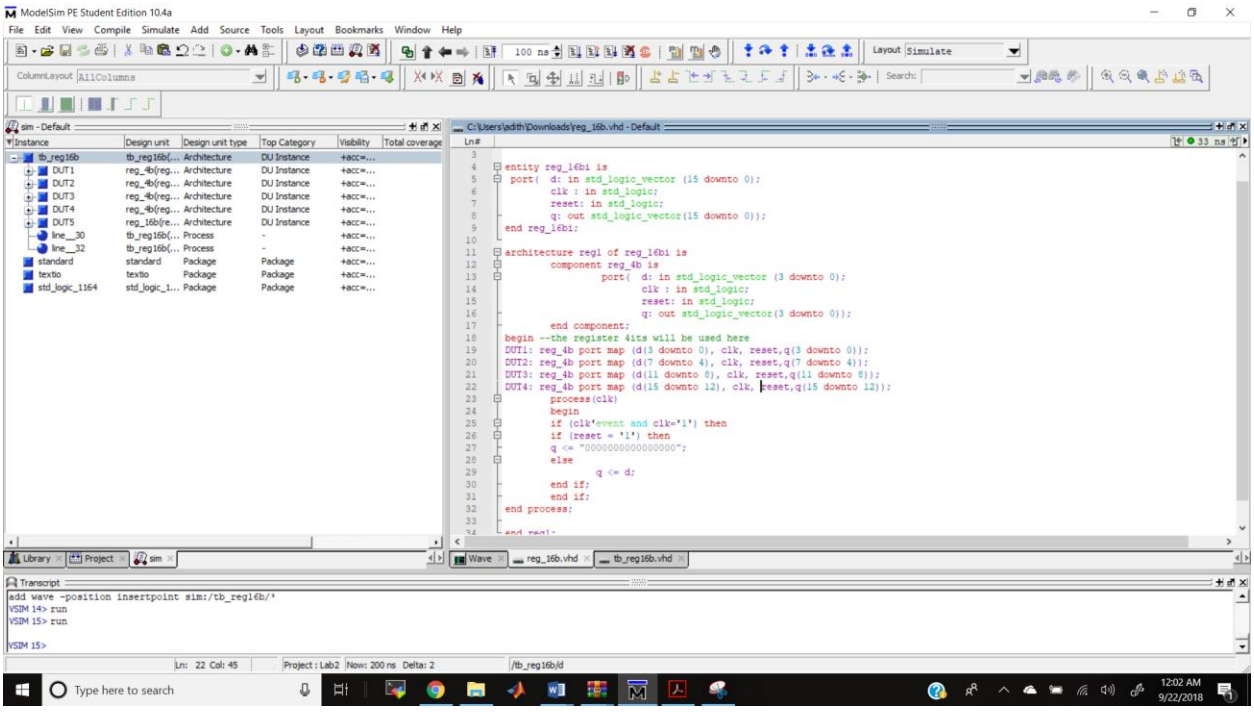


Test Bench

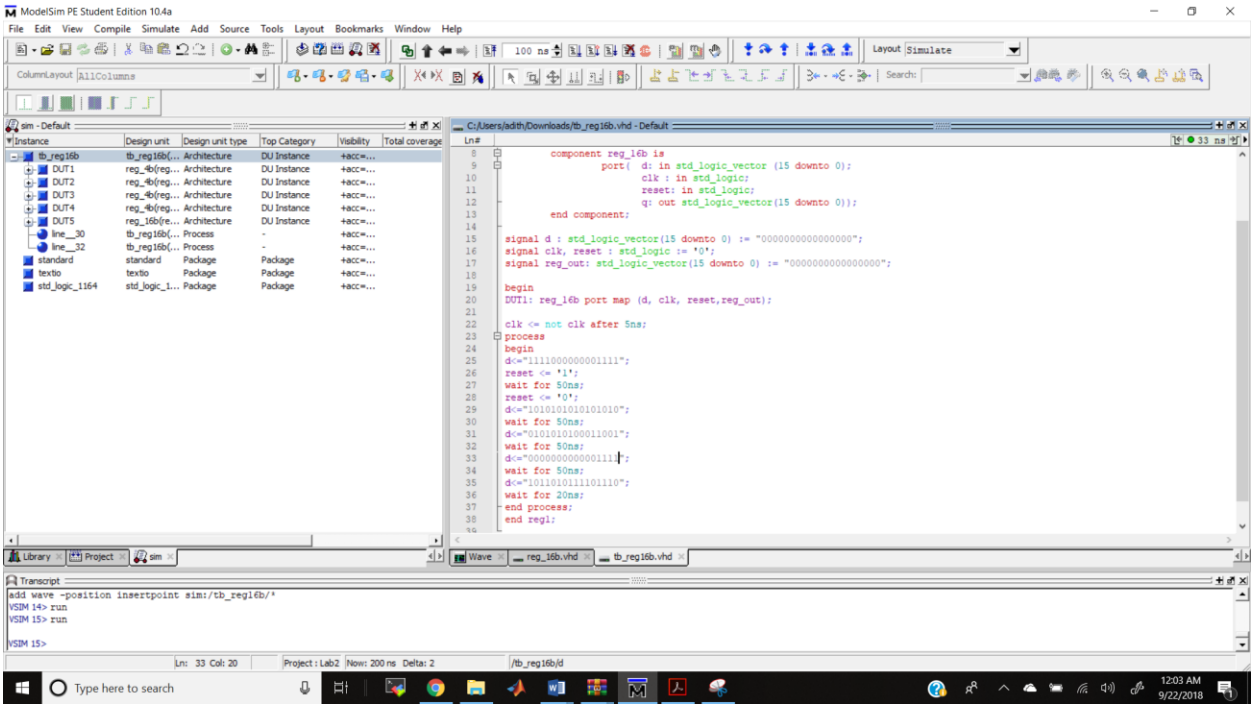




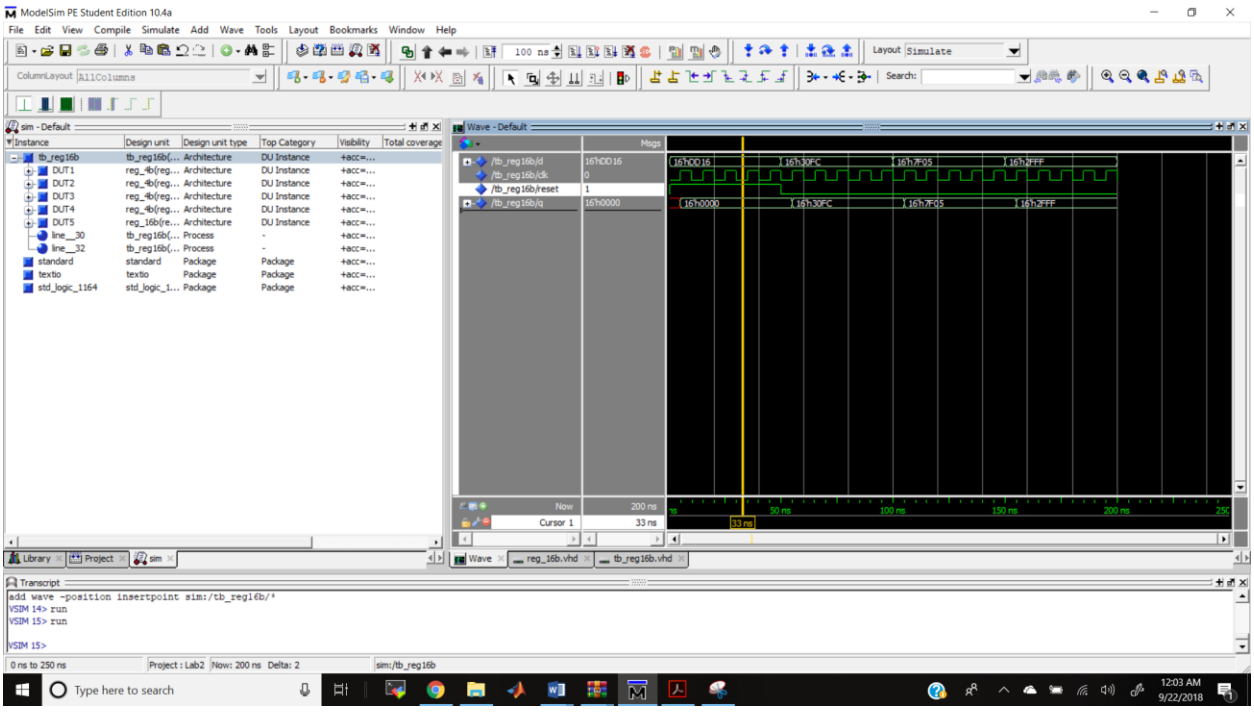
Register 16bit.



Testbench



Waveform



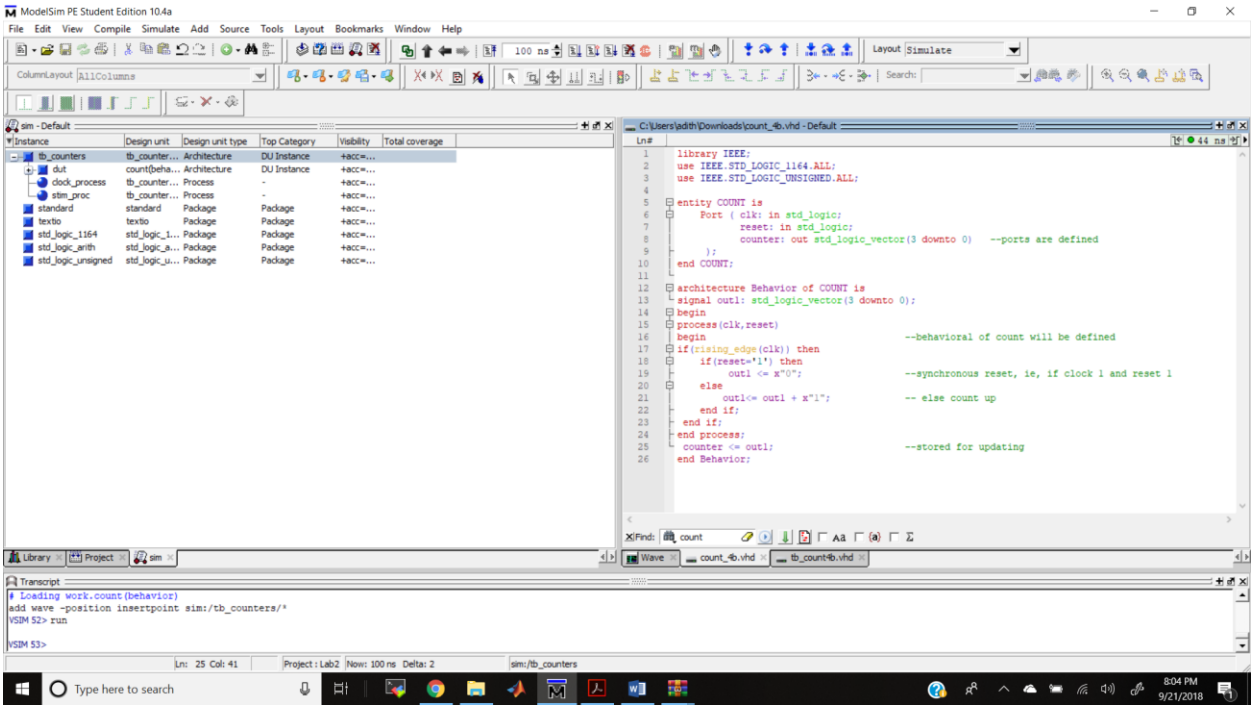
Counter

A 4 bit counter is as shown below. The incrementation of a bit takes place in every cycle.

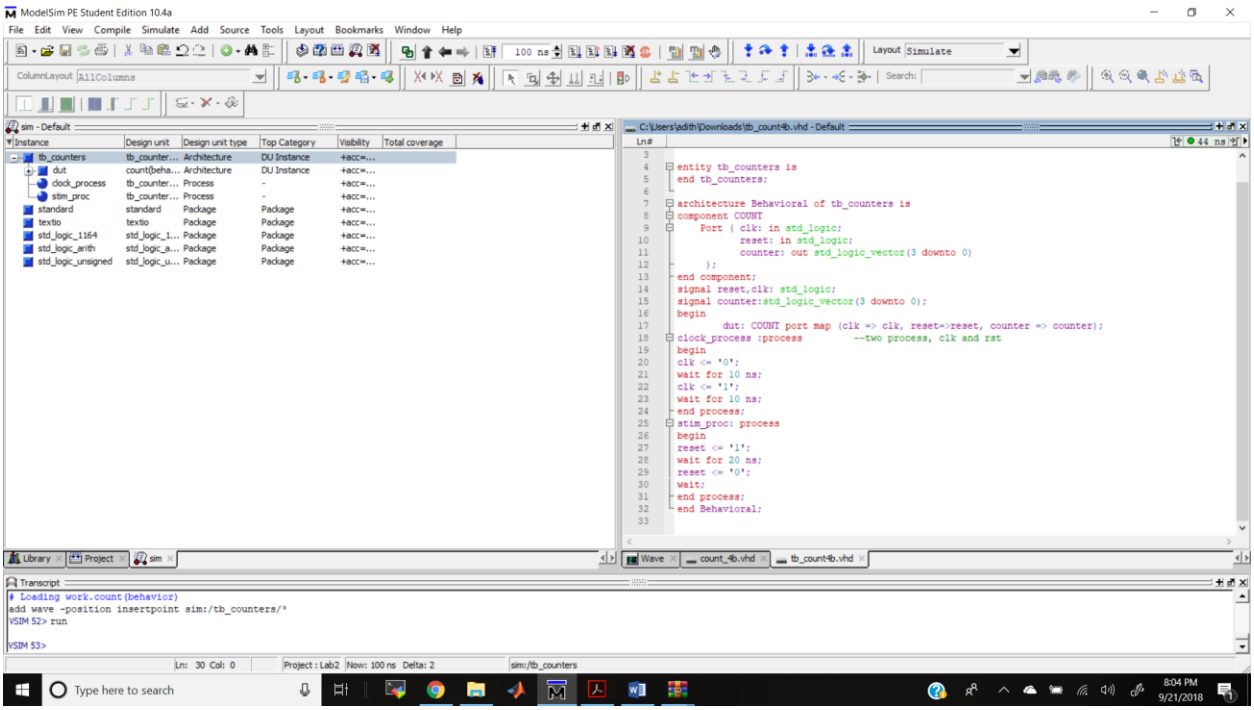
For example, the output when the input is 0111 is 1000. Ie from 3 to 4.

The vhdl code for it is as shown in the image. Waveform is also obtained

Source code



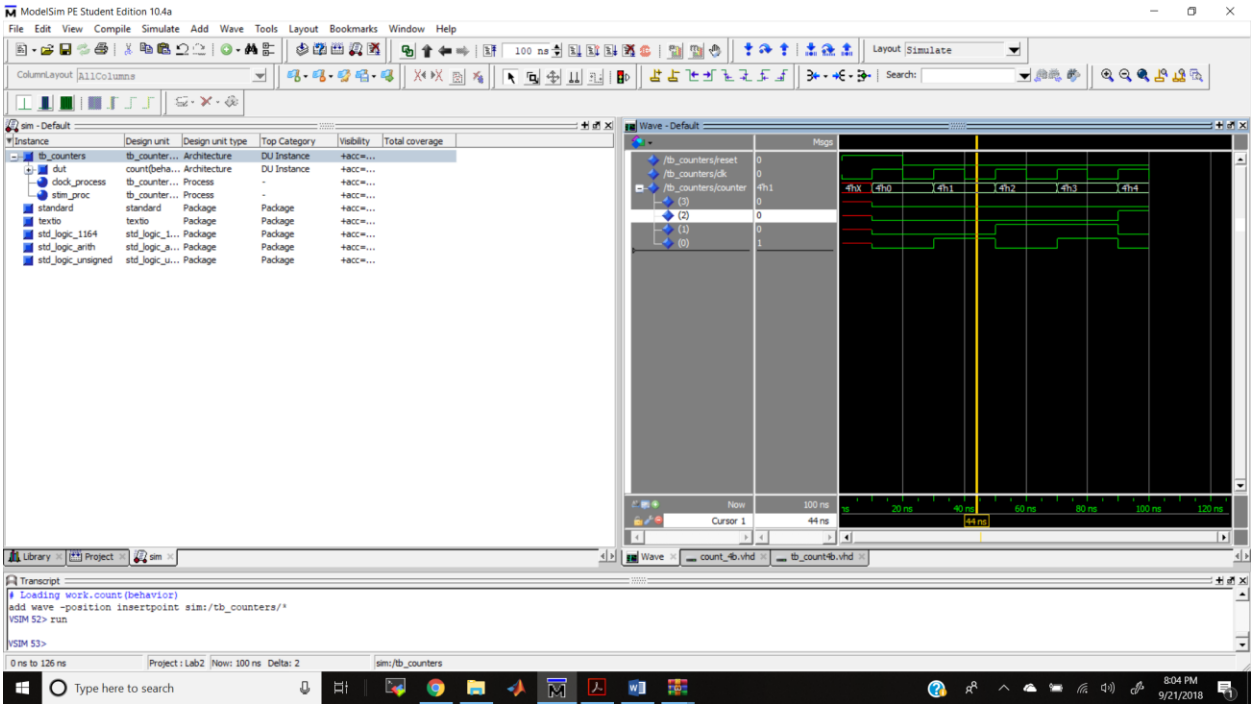
Test bench



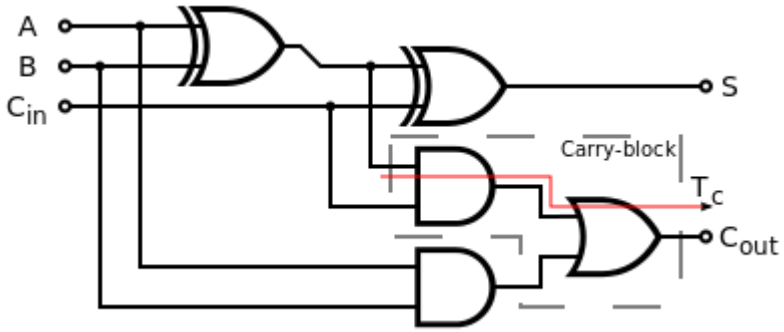
Waveform

Adithya Alva
DMD

283864587
Assignment 2



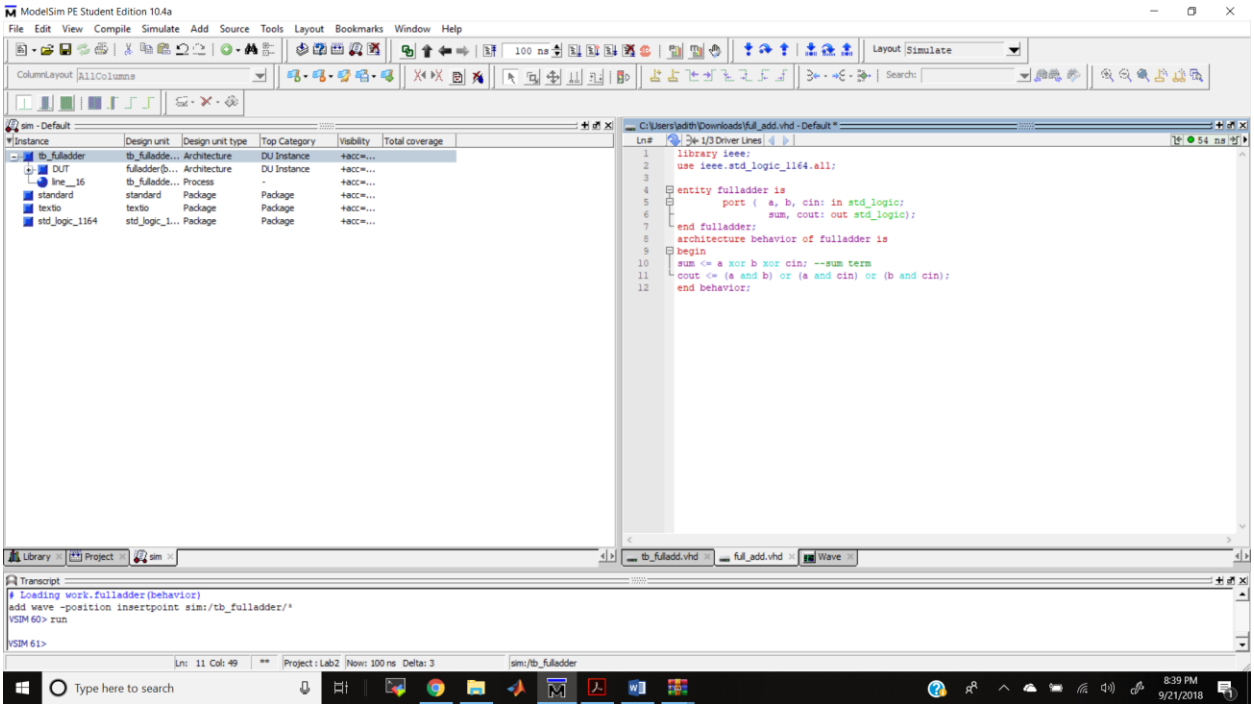
Full adder



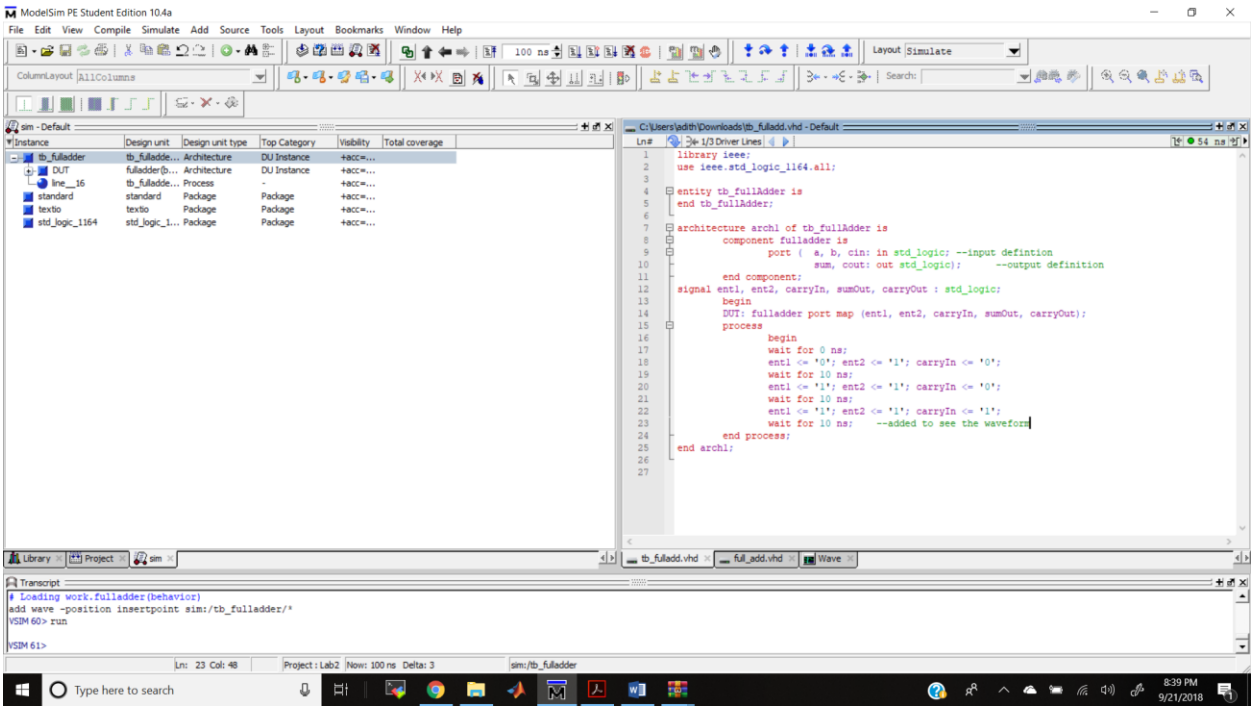
S= A xor B xor Cin

C = A.B + (Cin(A xor B))

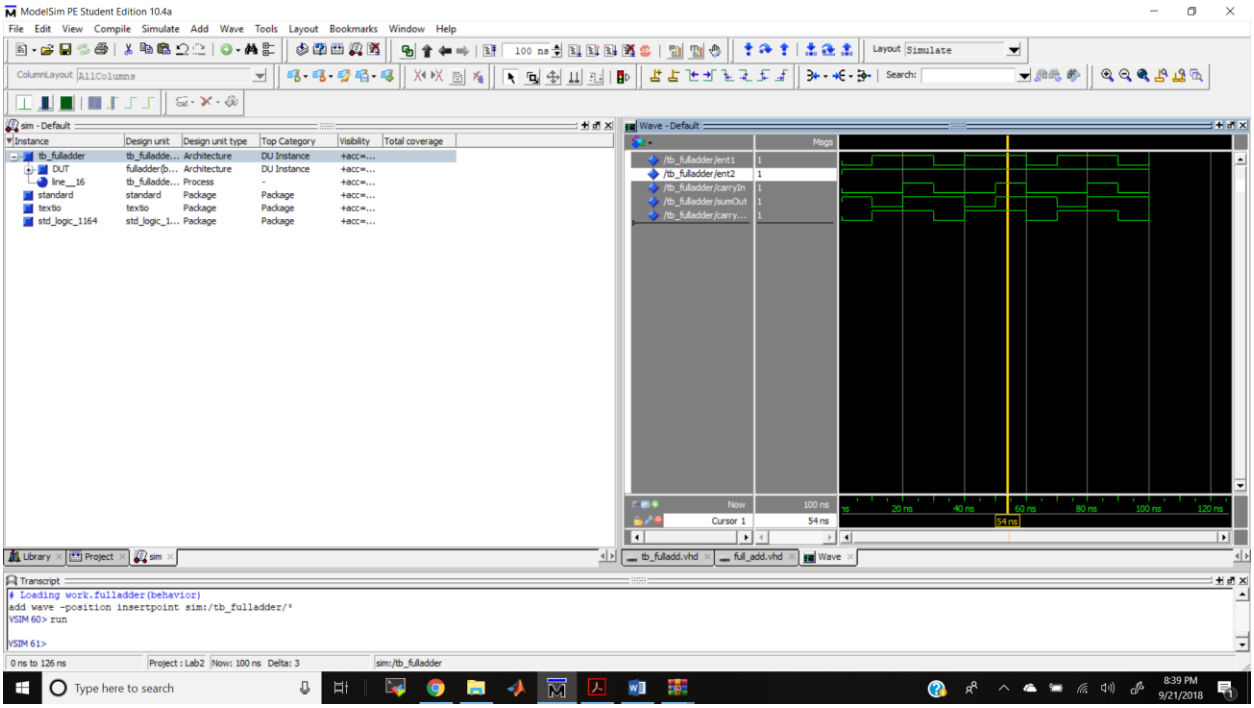
Source



Test bench

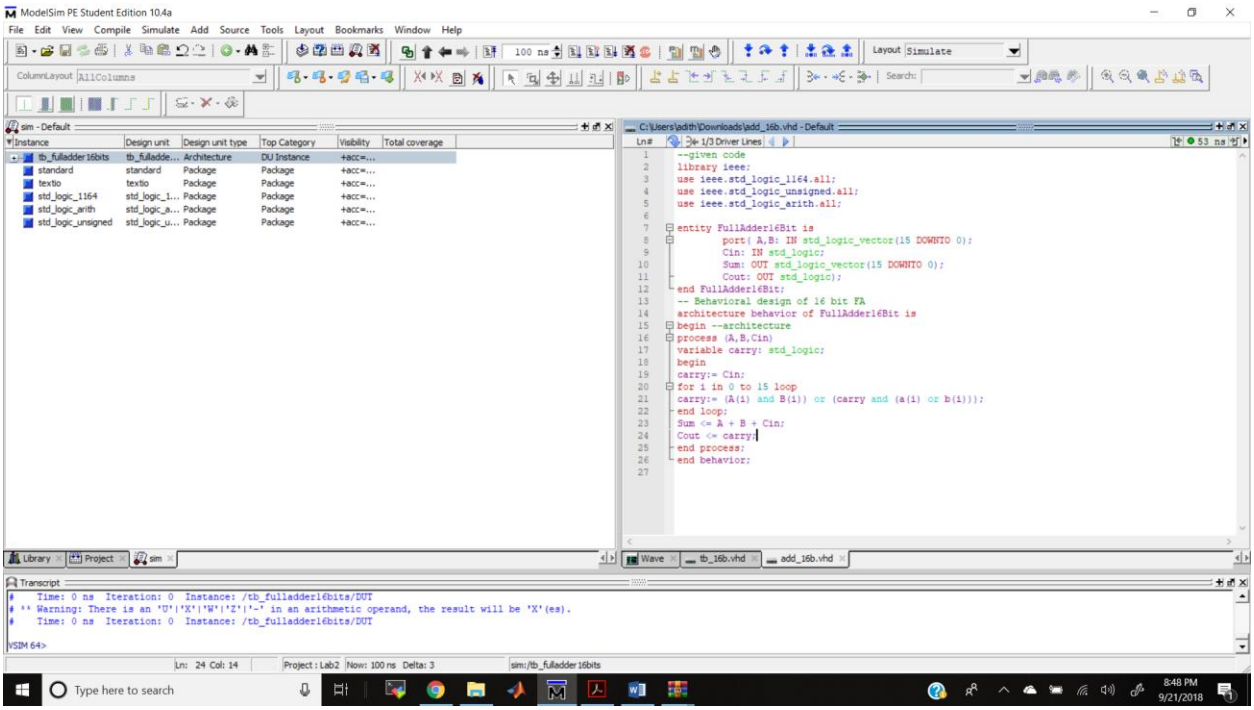


Waveform

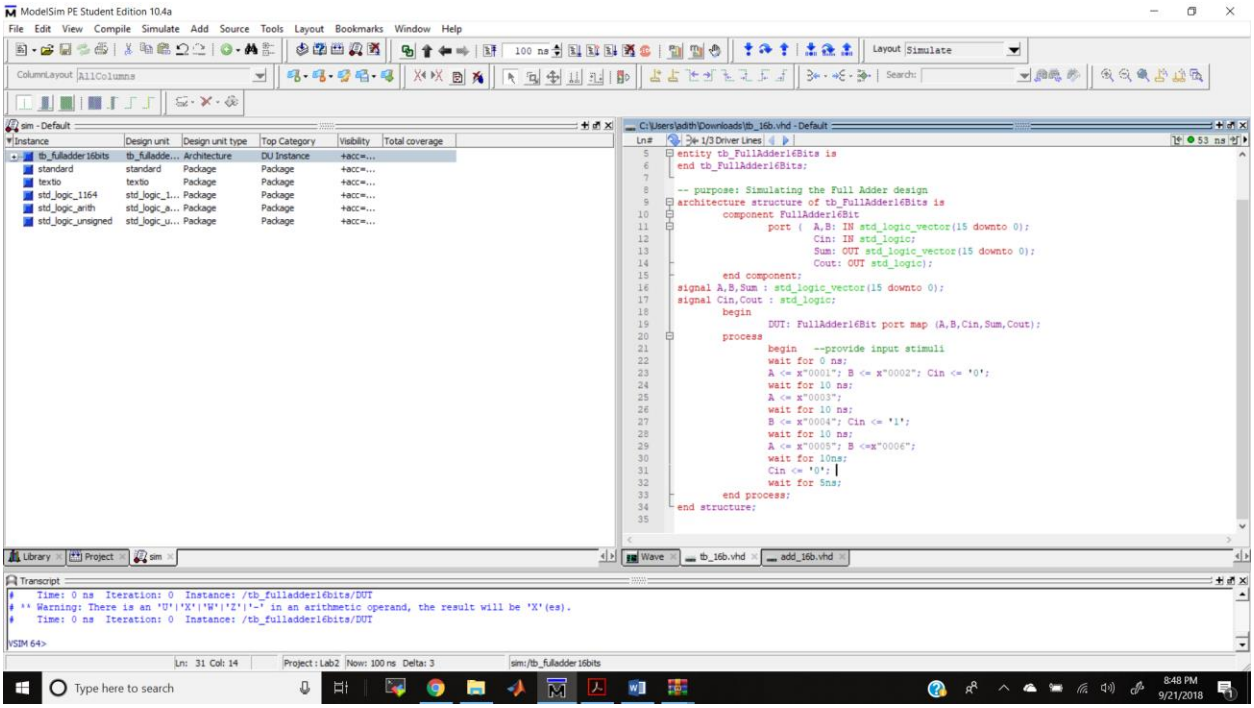


Full adder – 16bits

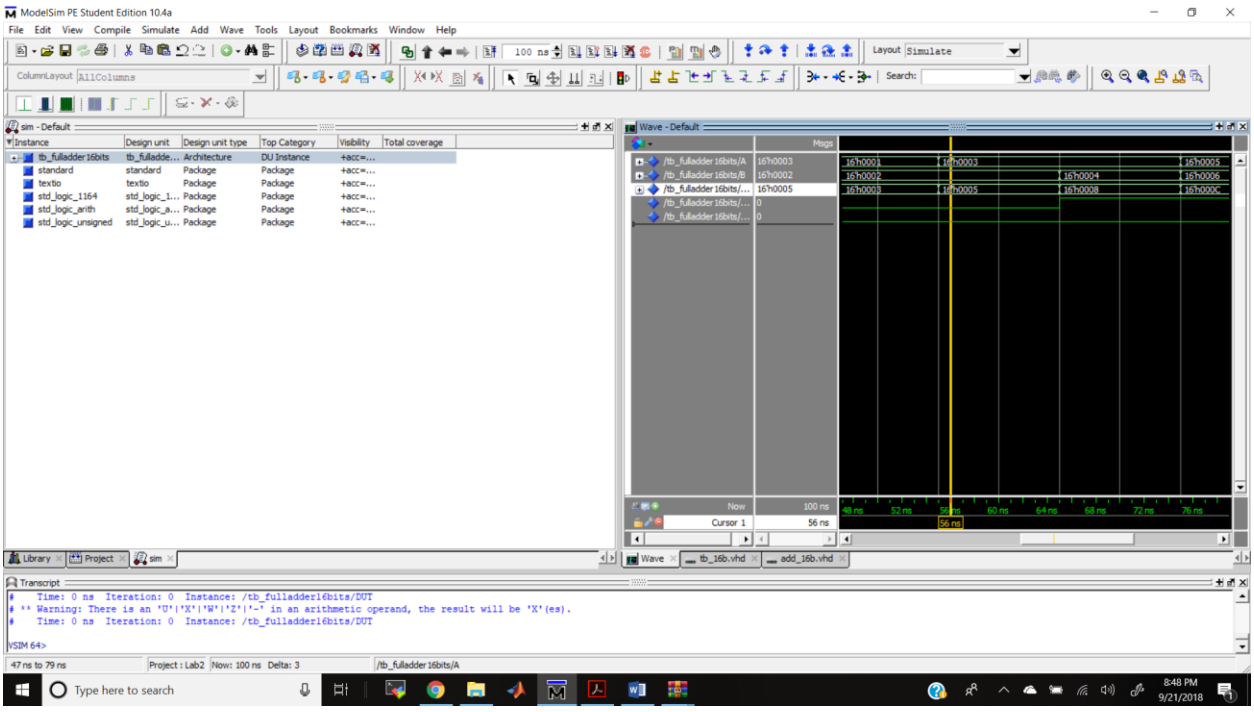
Similarly a 16 bits full adder is created using full adders



Test bench


ModelSim PE Student Edition 10.4a
File Edit View Compile Simulate Add Source Tools Layout Bookmarks Window Help
sim - Default
Instance Design unit Design unit type Top Category Visibility Total coverage
tb_fulladder16bits tb_fulladde... Architecture DU Instance +acc=...
standard standard Package Package +acc=...
textio textio Package Package +acc=...
std_logic_1164 std_logic_1... Package Package +acc=...
std_logic_arith std_logic_a... Package Package +acc=...
std_logic_unsigned std_logic_u... Package Package +acc=...
C:\Users\adith\Downloads\tb_16b.vhd - Default
entity tb_FullAdder16Bits is
end tb_FullAdder16Bits;
-- purpose: Simulating the Full Adder design
architecture structure of tb_FullAdder16Bits is
component FullAdder16Bit
port (A,B: IN std_logic_vector(15 downto 0);
Cin: IN std_logic;
Sum: OUT std_logic_vector(15 downto 0);
Cout: OUT std_logic);
end component;
signal A,B,Sum : std_logic_vector(15 downto 0);
signal Cin,Cout : std_logic;
begin
DUT: FullAdder16Bit port map (A,B,Cin,Sum,Cout);
process
begin --provide input stimuli
wait for 0 ns;
A <= x"0001"; B <= x"0002"; Cin <= '0';
wait for 10 ns;
A <= x"0003";
wait for 10 ns;
B <= x"0004"; Cin <= '1';
wait for 10 ns;
A <= x"0005"; B <=x"0006";
wait for 10ns;
Cin <= '0';
wait for 5ns;
end process;
end structure;
Library Project sim
Wave tb_16b.vhd add_16b.vhd
Transcript
Time: 0 ns Iteration: 0 Instance: /tb_fulladder16bits/DUT
** Warning: There is an 'U'|'X'|'W'|'Z'|'-' in an arithmetic operand, the result will be 'X'(es).
Time: 0 ns Iteration: 0 Instance: /tb_fulladder16bits/DUT
VSIM 64>
Ln: 31 Col: 14
Project : Lab2 Now: 100 ns Delta: 3
sim:/tb_fulladder16bits
Type here to search
8:48 PM
9/21/2018

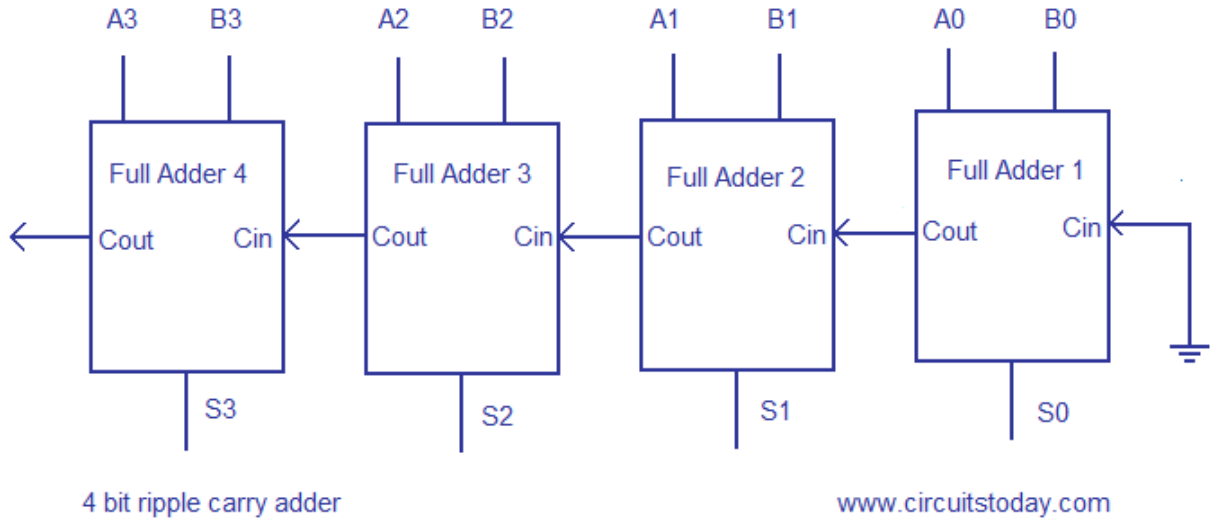
Waveform


ModelSim PE Student Edition 10.4a
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
sim - Default
Instance Design unit Design unit type Top Category Visibility Total coverage
tb_fulladder16bits tb_fulladde... Architecture DU Instance +acc=...
standard standard Package Package +acc=...
textio textio Package Package +acc=...
std_logic_1164 std_logic_1... Package Package +acc=...
std_logic_arith std_logic_a... Package Package +acc=...
std_logic_unsigned std_logic_u... Package Package +acc=...
Wave - Default
Msgs
/tb_fulladder16bits/A 16'h0003
/tb_fulladder16bits/B 16'h0002
/tb_fulladder16bits/... 16'h0005
/tb_fulladder16bits/... 0
/tb_fulladder16bits/... 0
16'h0001 16'h0003 16'h0005
16'h0002 16'h0004 16'h0006
16'h0003 16'h0005 16'h0008 16'h000C
Now 100 ns
Cursor 1 56 ns
48 ns 52 ns 56 ns 60 ns 64 ns 68 ns 72 ns 76 ns
Library Project sim
Wave tb_16b.vhd add_16b.vhd
Transcript
Time: 0 ns Iteration: 0 Instance: /tb_fulladder16bits/DUT
** Warning: There is an 'U'|'X'|'W'|'Z'|'-' in an arithmetic operand, the result will be 'X'(es).
Time: 0 ns Iteration: 0 Instance: /tb_fulladder16bits/DUT
VSIM 64>
47 ns to 79 ns
Project : Lab2 Now: 100 ns Delta: 3
/tb_fulladder16bits/A
Type here to search
8:48 PM
9/21/2018

Adithya Alva
DMD

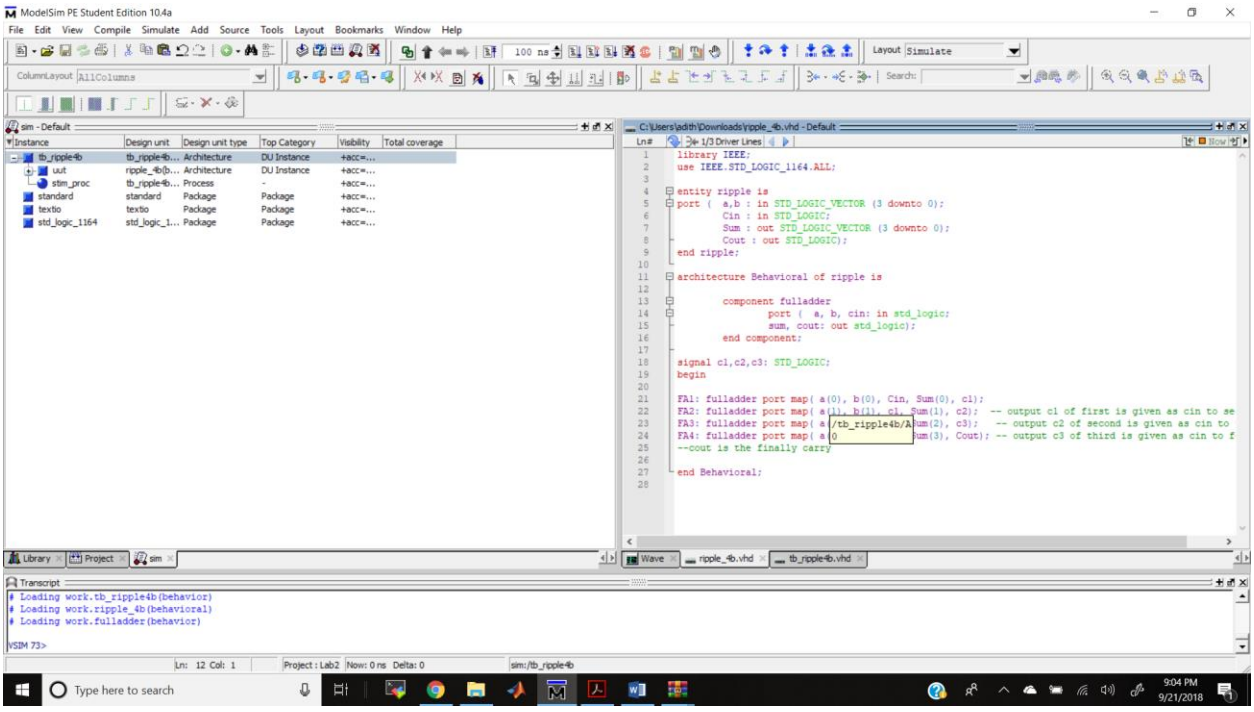
283864587
Assignment 2

Ripple carry adder - 4bit



As seen above the cout of first is given as an input to the next and so on.

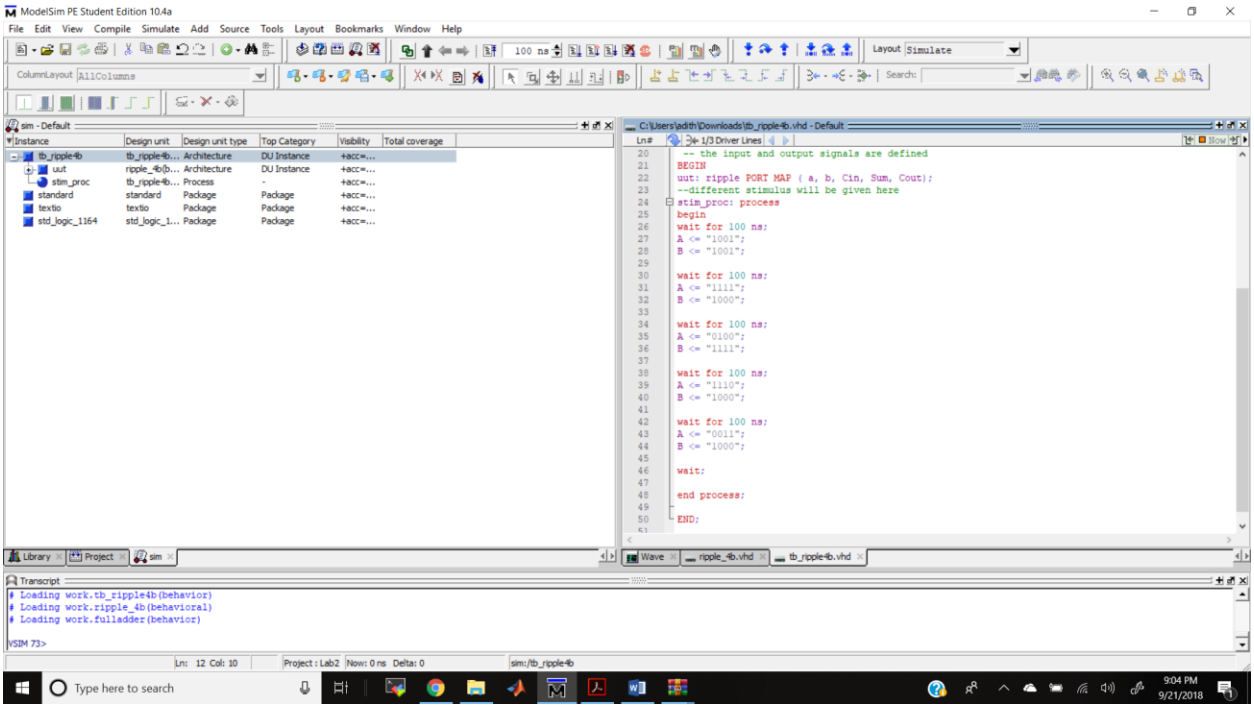
This is how a ripple carry adder is created.



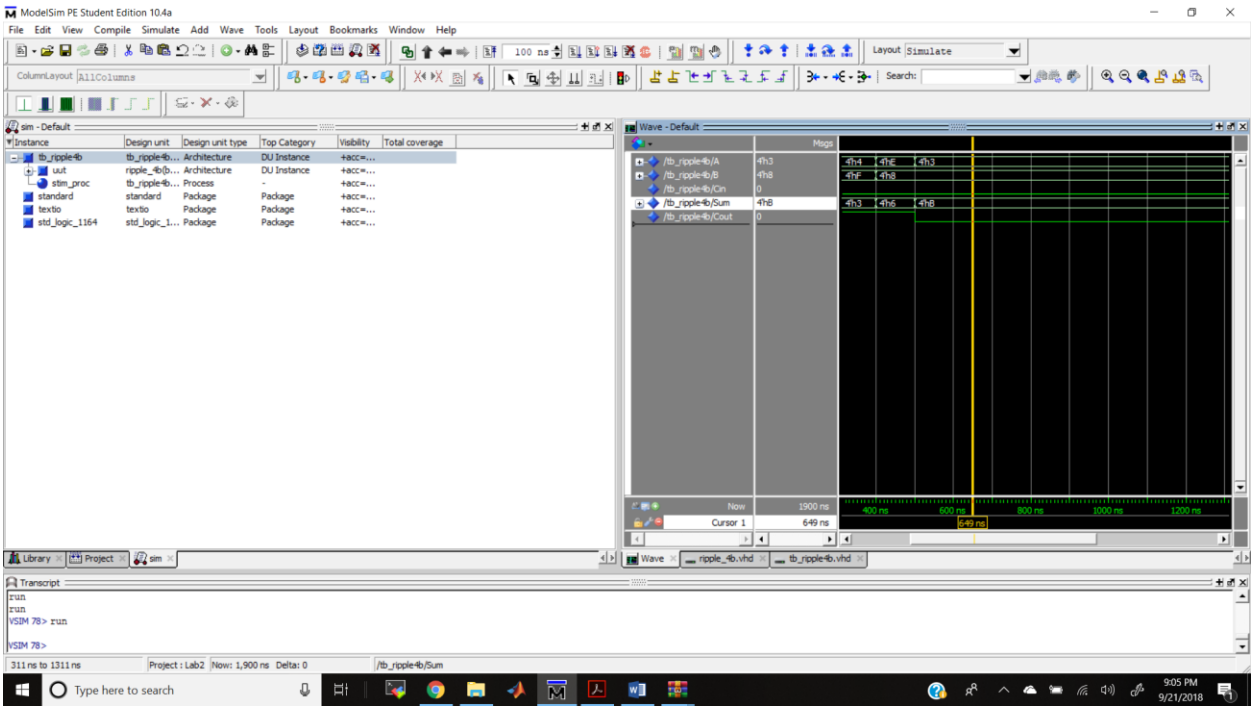
Test bench

Adithya Alva
DMD

283864587
Assignment 2

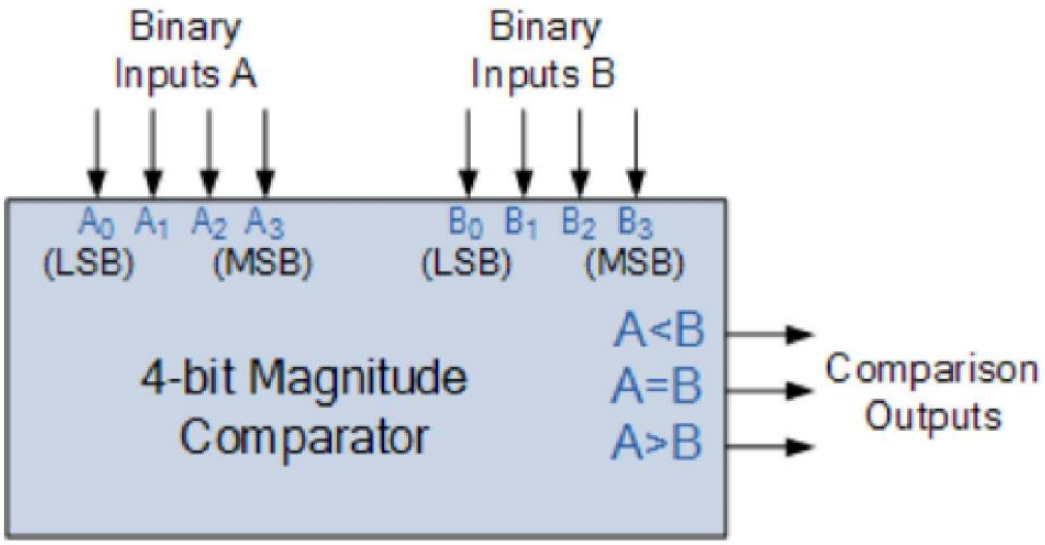


Waveform



Adithya Alva 283864587
DMD Assignment 2

Comparator



Comparator Is basically used to compare the two inputs.

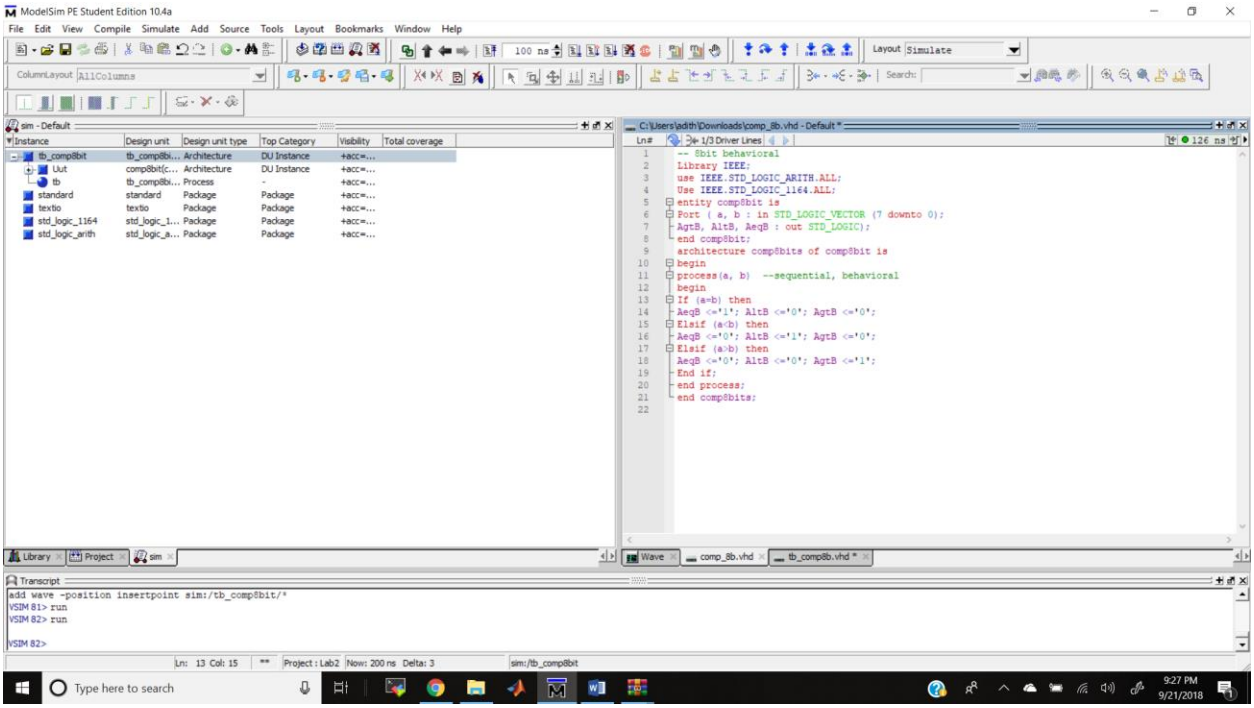
We have 3 inputs,

a>b, a=b, a<b. Depending on the inputs the value of the output will be high.

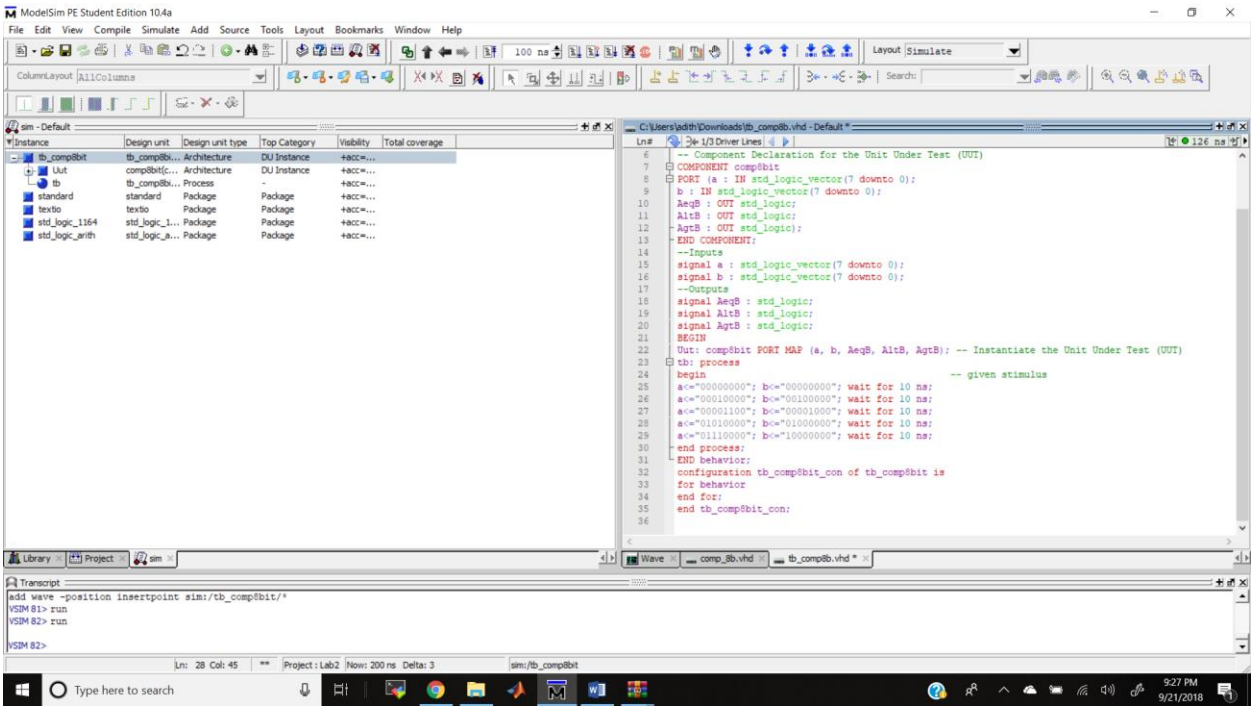
Inputs				Outputs		
A_1	A_0	B_1	B_0	A>B	A=B	A<B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

- **behavioral, sequential**

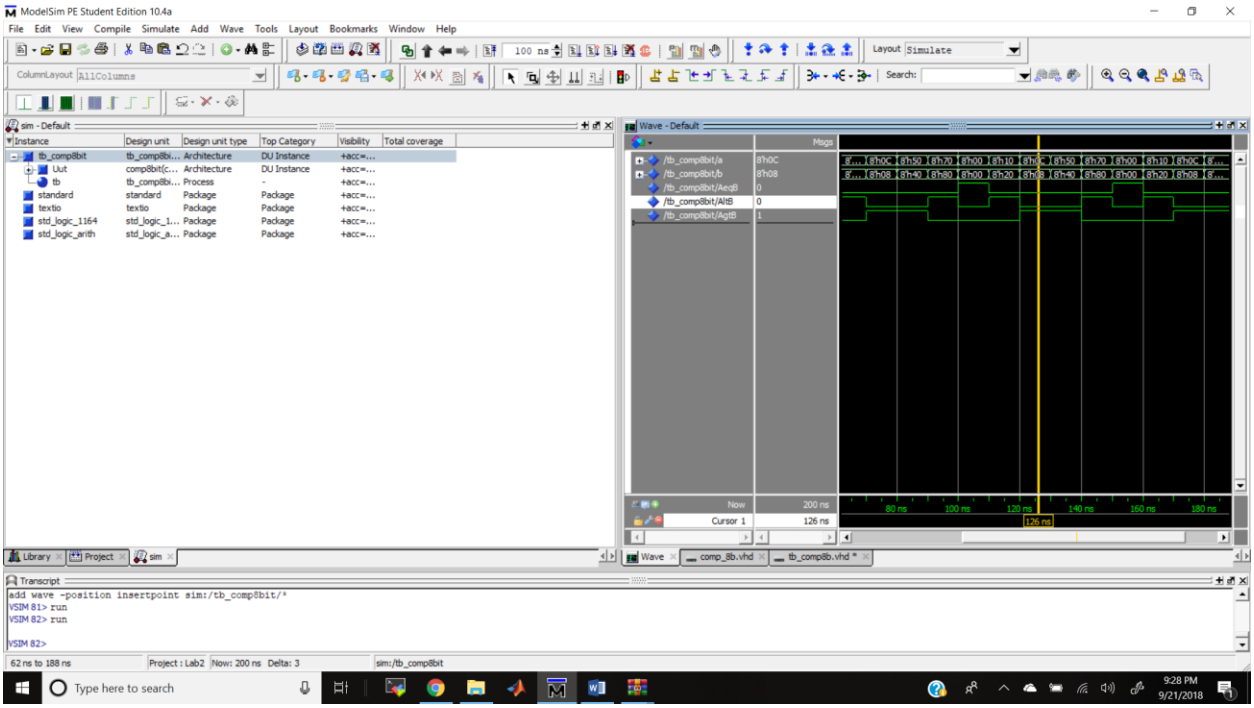
Here the process is used, which makes the lines to be executed one after the other or sequentially.



Testbench

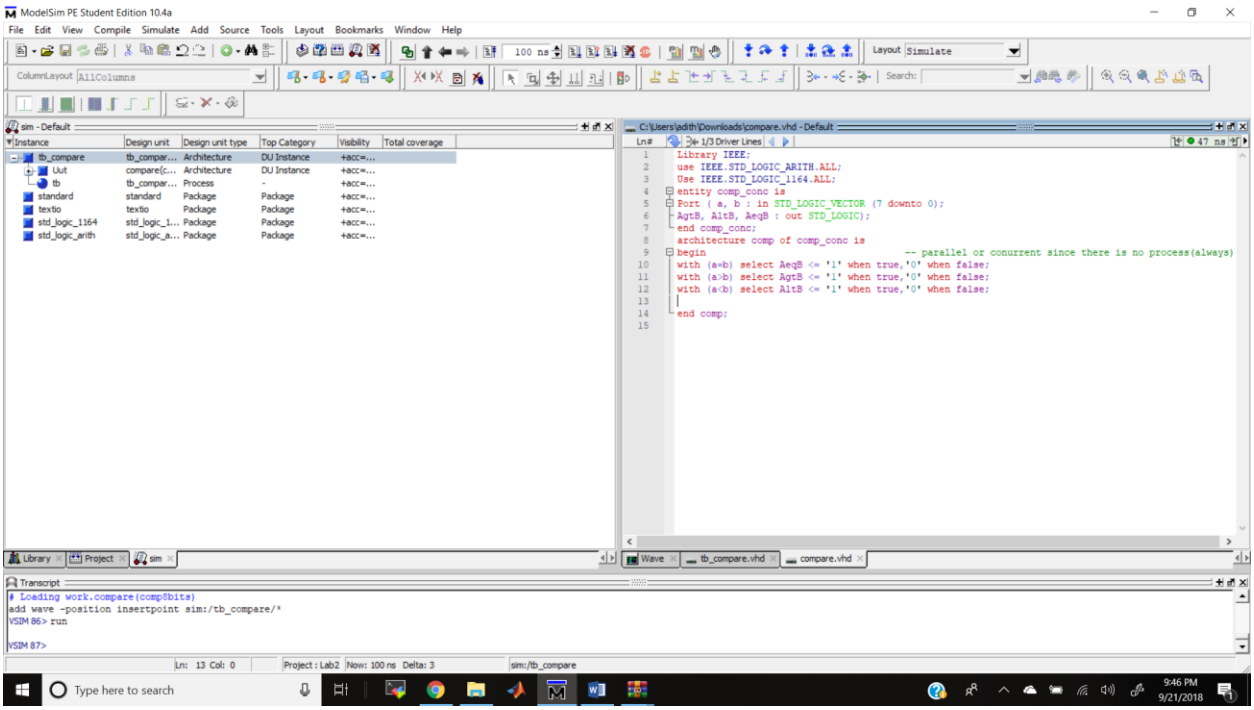


Waveform

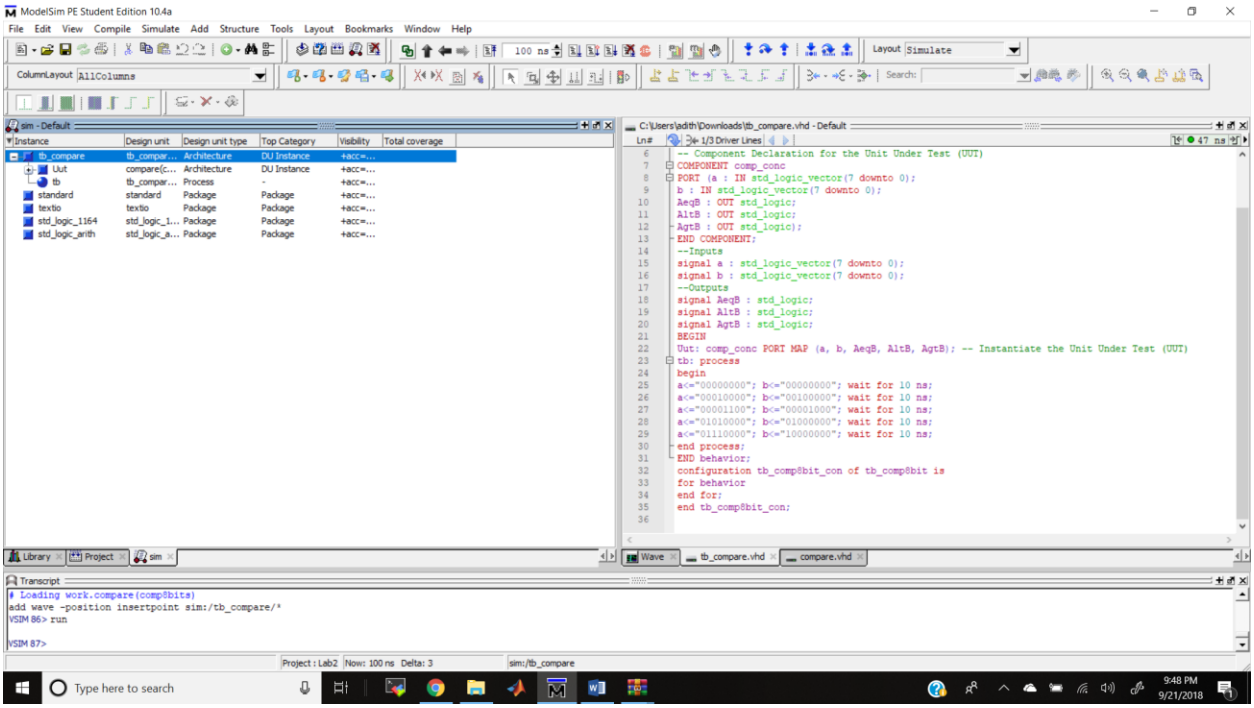


Comparator concurrent

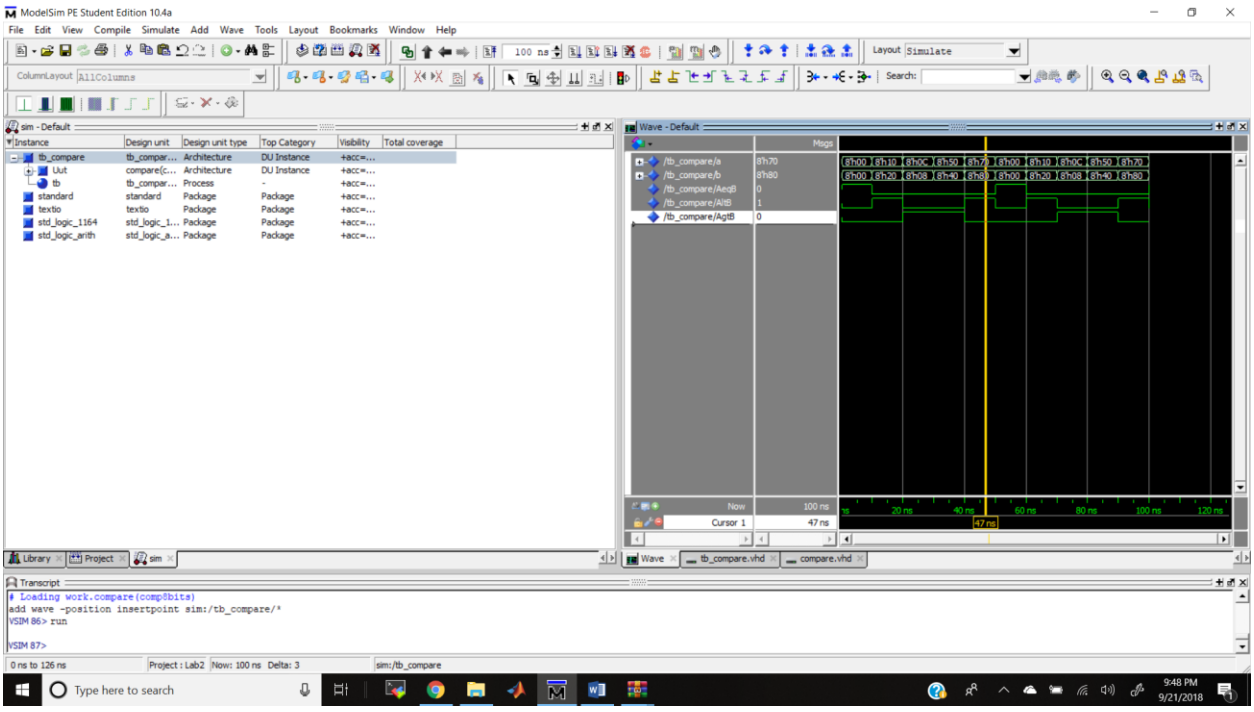
Here the process is not used, the when and select statements are used, which makes the codes to be executed in parallel or concurrently.



Test bench



Waveform



References,

Google images and other websites were used for description.