UNIVERSITY OF ENGINEERING & MANAGEMENT, KOLKATA

Course Name : Computer Organization & Architecture Laboratory

VHDL program

- Verilog Hardware Description Language.
- VHSIC (Very High Speed Integrated Circuit) hardware description language.
- Defined by IEEE standards.
- Used to describe digital and mixed-signal systems.
- In a VHDL design, each unit to be modelled is known as design entity/VHDL entity.
- Two types of ingredients: entity declaration and architecture declaration.

Structure of VHDL program

•Every VHDL program consists of at least one entity/architecture pair.

ENTITY

•List with specifications of all inputs and outputs pins of the circuit.

•Syntax – *entity NAME is*

```
port (
        port_name : signal_mode signal_type;
        port_name : signal_mode signal_type;
        . . . );
end NAME;
```

- Example - *entity OR_gate is*

```
port (
        A : in std_logic;
        B : in std_logic;
        Y : out std_logic;
);
end OR_gate;
```

- Name of the **ENTITY** should not be reserved word in VHDL.
- **ENTITY** name should be same as the file name.

ARCHITECTURE

- description of how the circuit of design works.
- Syntax –

```
architecture ARCHITECTURE_NAME of ENTITY_NAME is
        [declarations]
            begin
                (code)
            end ARCHITECTURE_NAME;
```

- Example –

```
architecture or_logic of OR_gate is
        begin
            Y <= A OR B;
        end or_logic;
```

- Name of the **ARCHITECTURE** should not be reserved word in VHDL.

Mode of the signal

- Mode of the signal may be In, OUT, INOUT or BUFFER.
- Unidirectional pin – IN, OUT.
- Bidirectional pin – INOUT.
- BUFFER – When output signal is used internally in design.

Type of the signal

- Type of the signal BIT, STD_LOGIC, INTEGER, etc.

Create a New Project in ModelSim

Open ModelSim -> Click File -> New -> Project -> Provide Project Name and Location -> Click OK -> Project will be created and Add Items to the project dialog box opens -> Click on Create New File -> Create a new project file opens up -> Give a file name AND_gate and select the file type as VHDL -> Click OK -> Project will be created and the AND_gate will be added -> Double click on the named file AND_gate.vhd and editor will open -> Write the code.

Compile the code on ModelSim

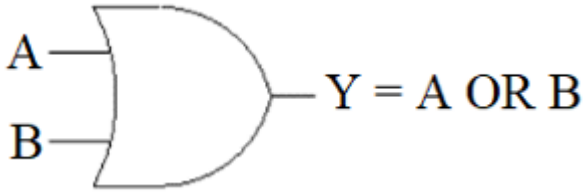
Save file by clicking Ctrl+S -> Click on Compile on the toolbar -> Compile All.
Green tick mark appears along with the Compile of the entity_name.vhd was successful" if the code is error free.
Red mark appears if the code has error.

Simulate the design circuit

Click on Simulate on the toolbar - > Start Simulate -> A dialog box opens and a folder appears Work -> Click on Work -> Select entity_name.vhd (for example, OR_gate.vhd) -> Click on OK. Simulation window opens -> Click on Add -> Click on Wave -> Click on Cursor (A yellow cursor will be added) - >Right click on the entity_name.vhd -> Click on Add all signals to wave -> Wave form window opens with declared inputs and outputs on the right side of the window -> Right click on inputs a and b -> Select Force -> Set the value for a as 0 and do the same for b -> Click OK -> Click on the Run option on the toolbar.

OR Gate

Symbol:

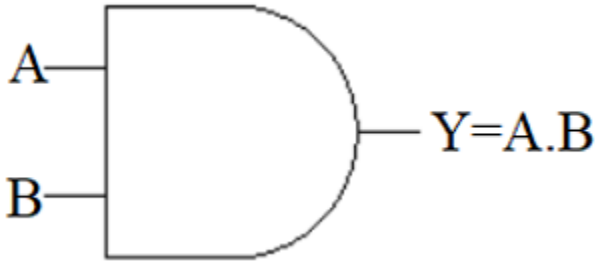


Truth Table:

A	B	Y = A + B
0	0	0
0	1	1
1	0	1
1	1	1

AND Gate

Symbol:

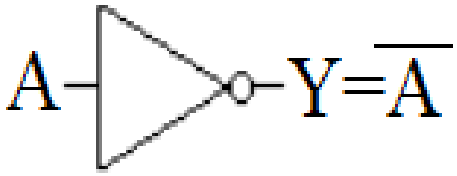


Truth Table:

A	B	Y = A . B
0	0	0
0	1	0
1	0	0
1	1	1

NOT Gate

Symbol:

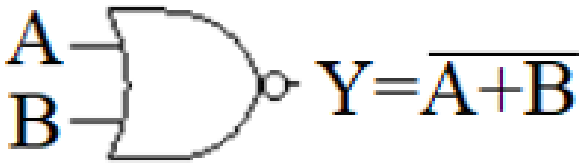


Truth Table:

A	Y = A'
0	1
1	0

NOR Gate

Symbol:

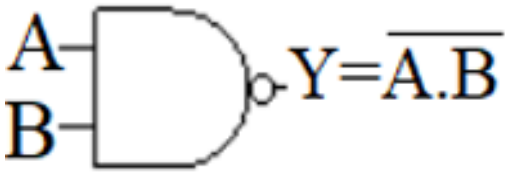


Truth Table:

A	B	Y = A . B
0	0	1
0	1	0
1	0	0
1	1	0

NAND Gate

Symbol:

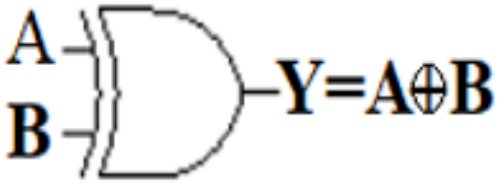


Truth Table:

A	B	Y = A . B
0	0	1
0	1	1
1	0	1
1	1	0

XOR Gate

Symbol:

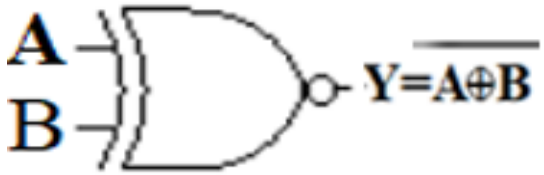


Truth Table:

A	B	Y = A . B
0	0	0
0	1	1
1	0	1
1	1	0

XNOR Gate

Symbol:



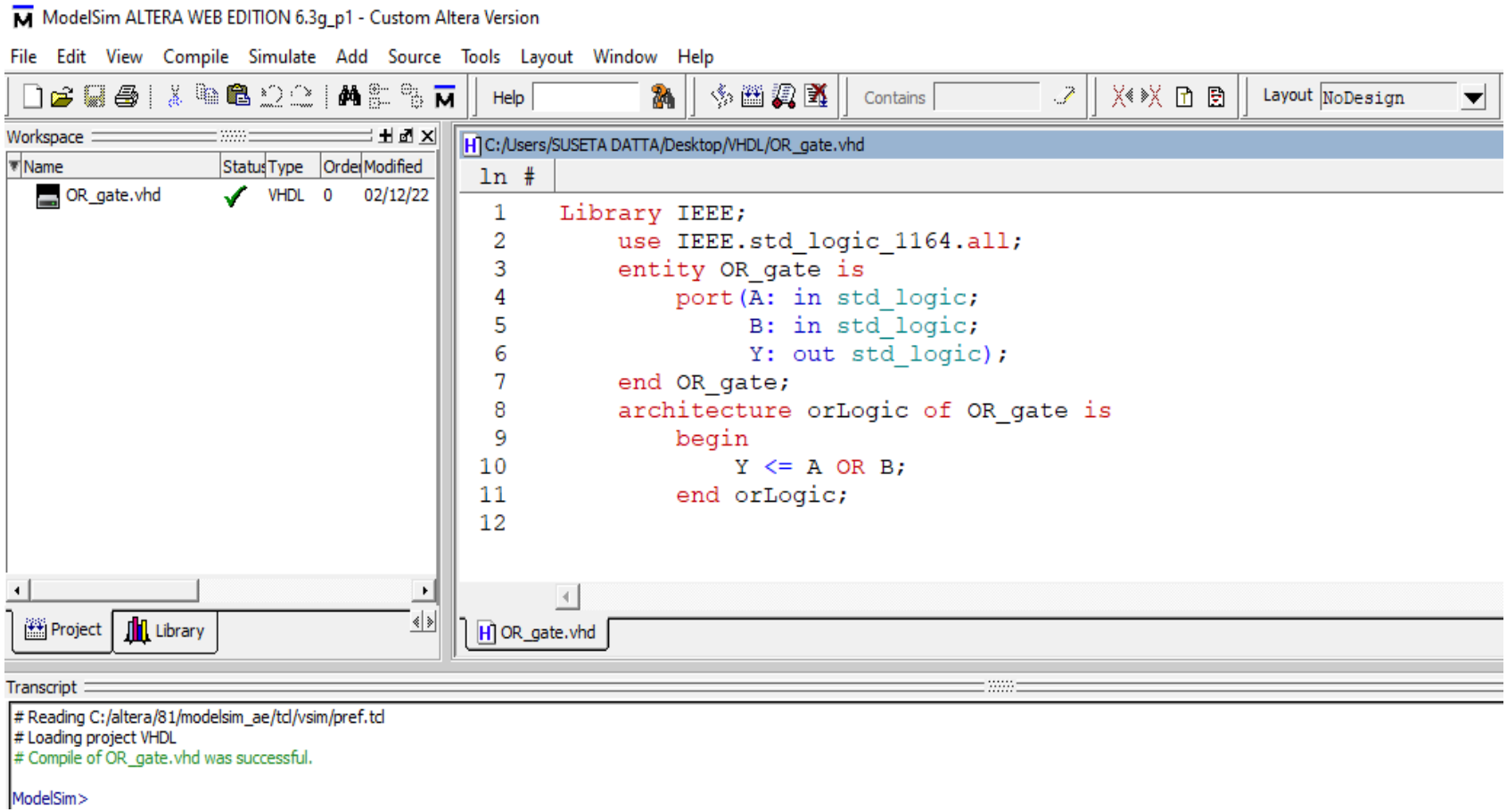
Truth Table:

A	B	Y = A . B
0	0	1
0	1	0
1	0	0
1	1	1

Libraries imported

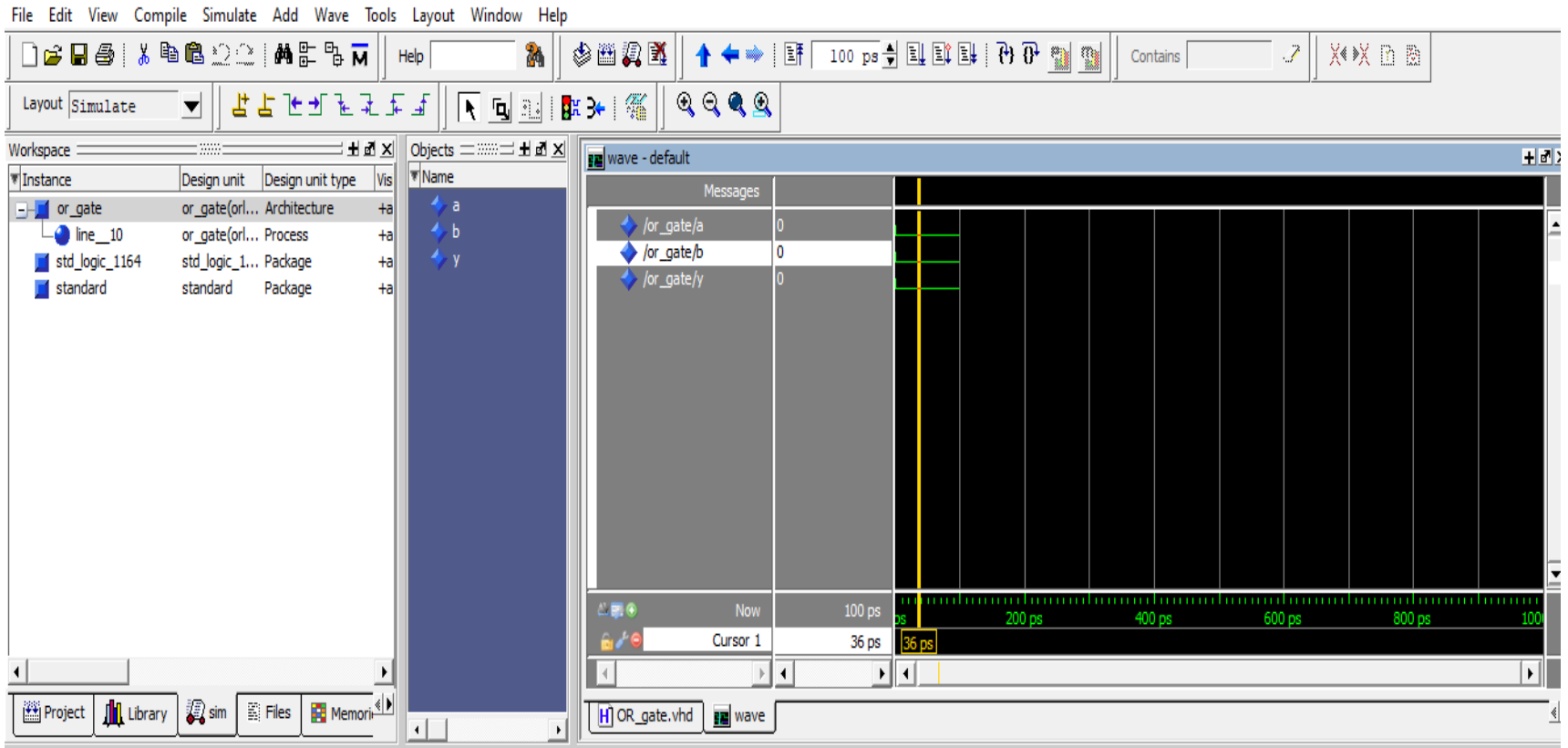
```
Library IEEE;
use IEEE.std_logic_1164.all;
```

Code for OR Gate

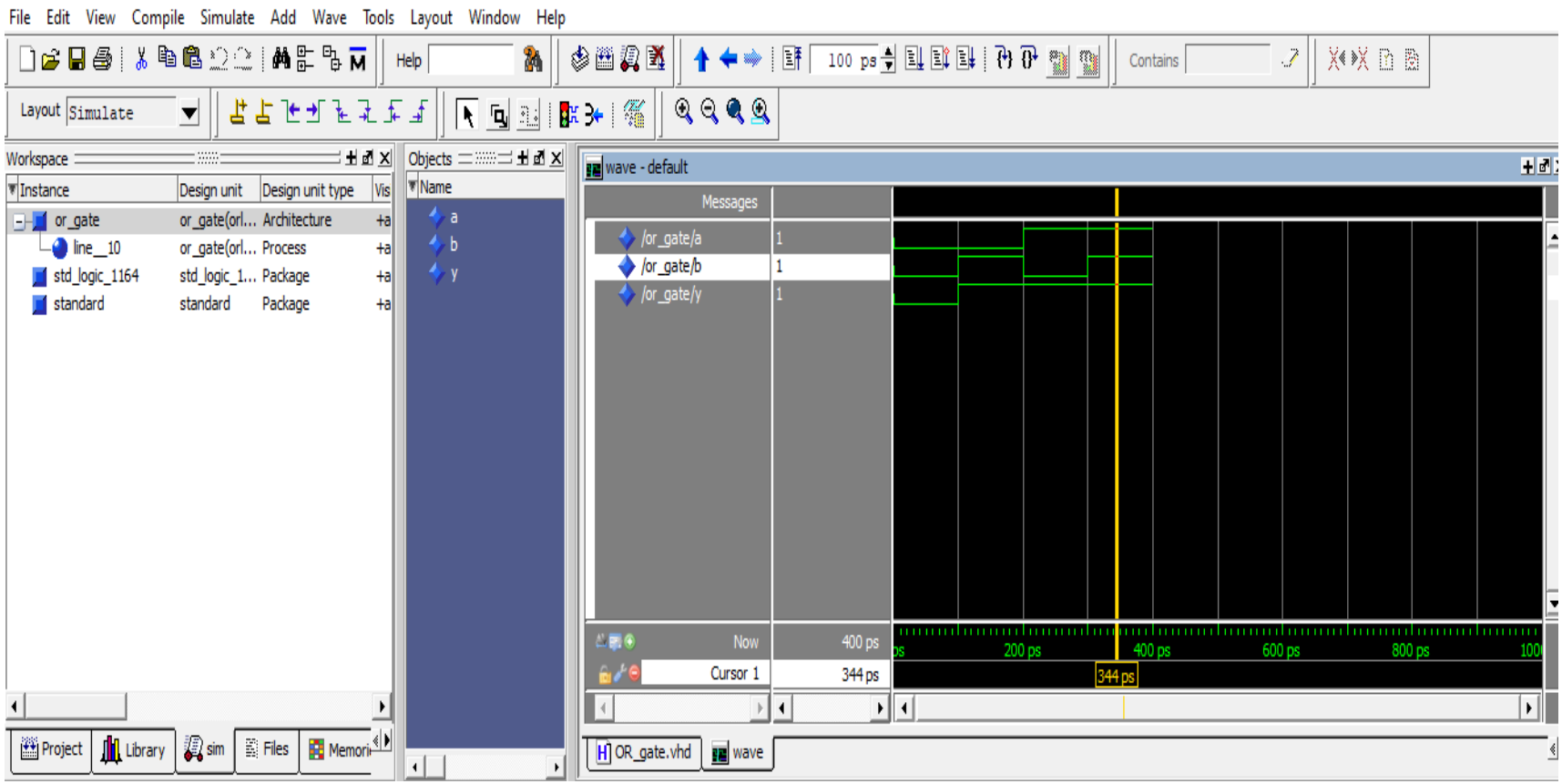


Output for OR Gate

Considering a=0 and b=0. Waveforms on the graph for 100 ps.



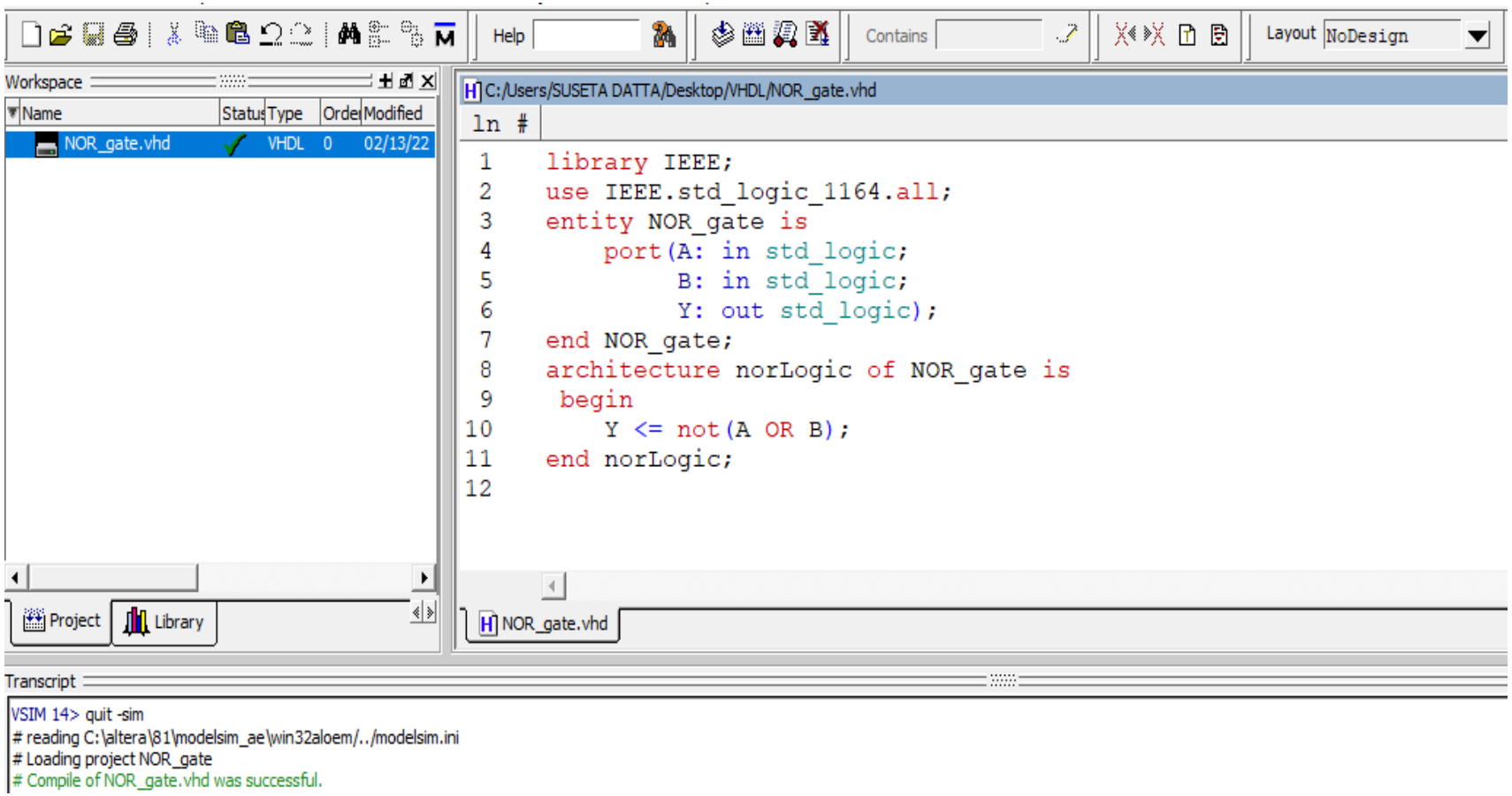
Output for OR Gate



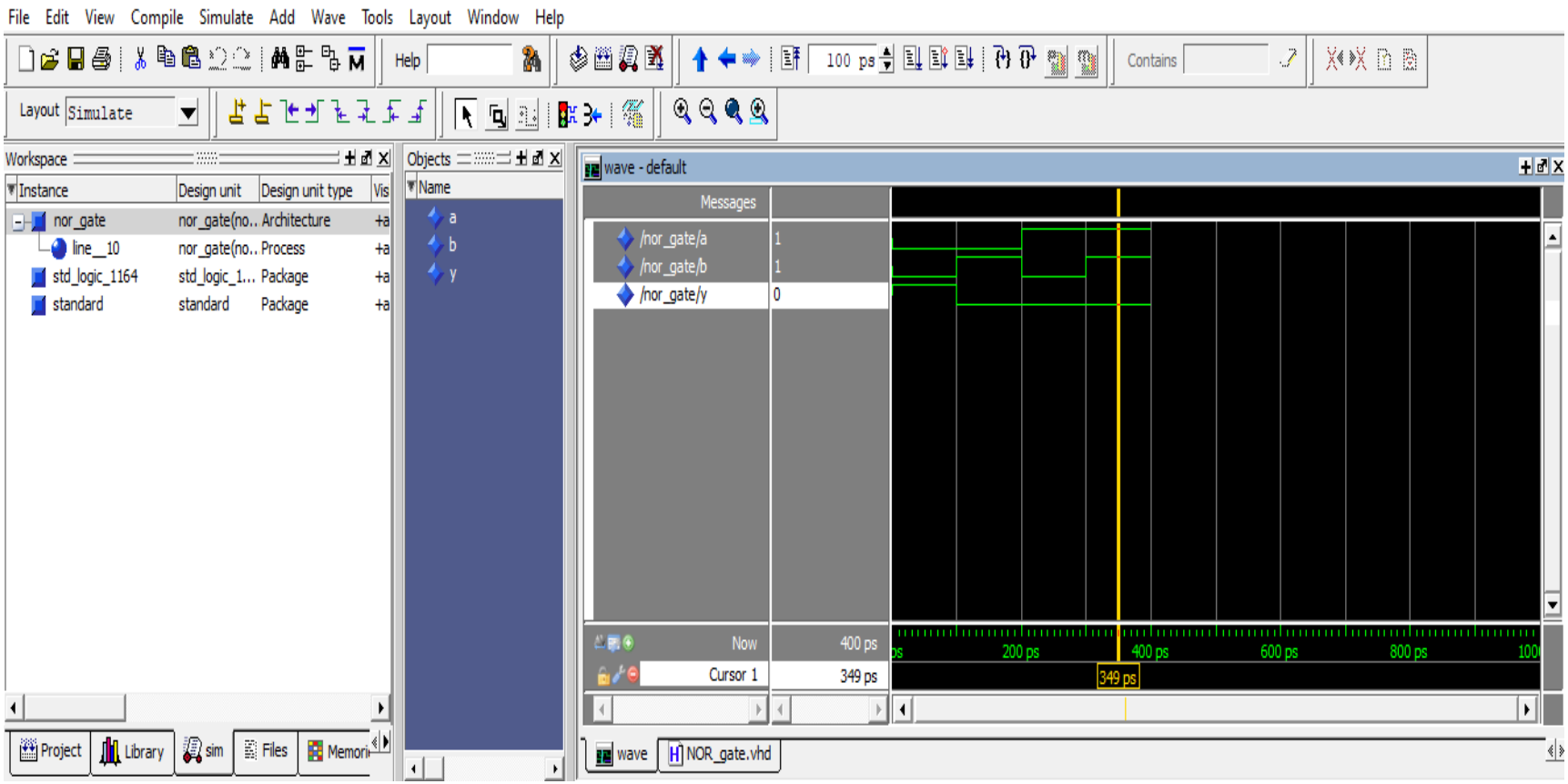
Truth Table for OR Gate

A	B	Y	Timings (ps)
0	0	0	0-100
0	1	1	100-200
1	0	1	200-300
1	1	1	300-400

Code for NOR Gate



Output for NOR Gate



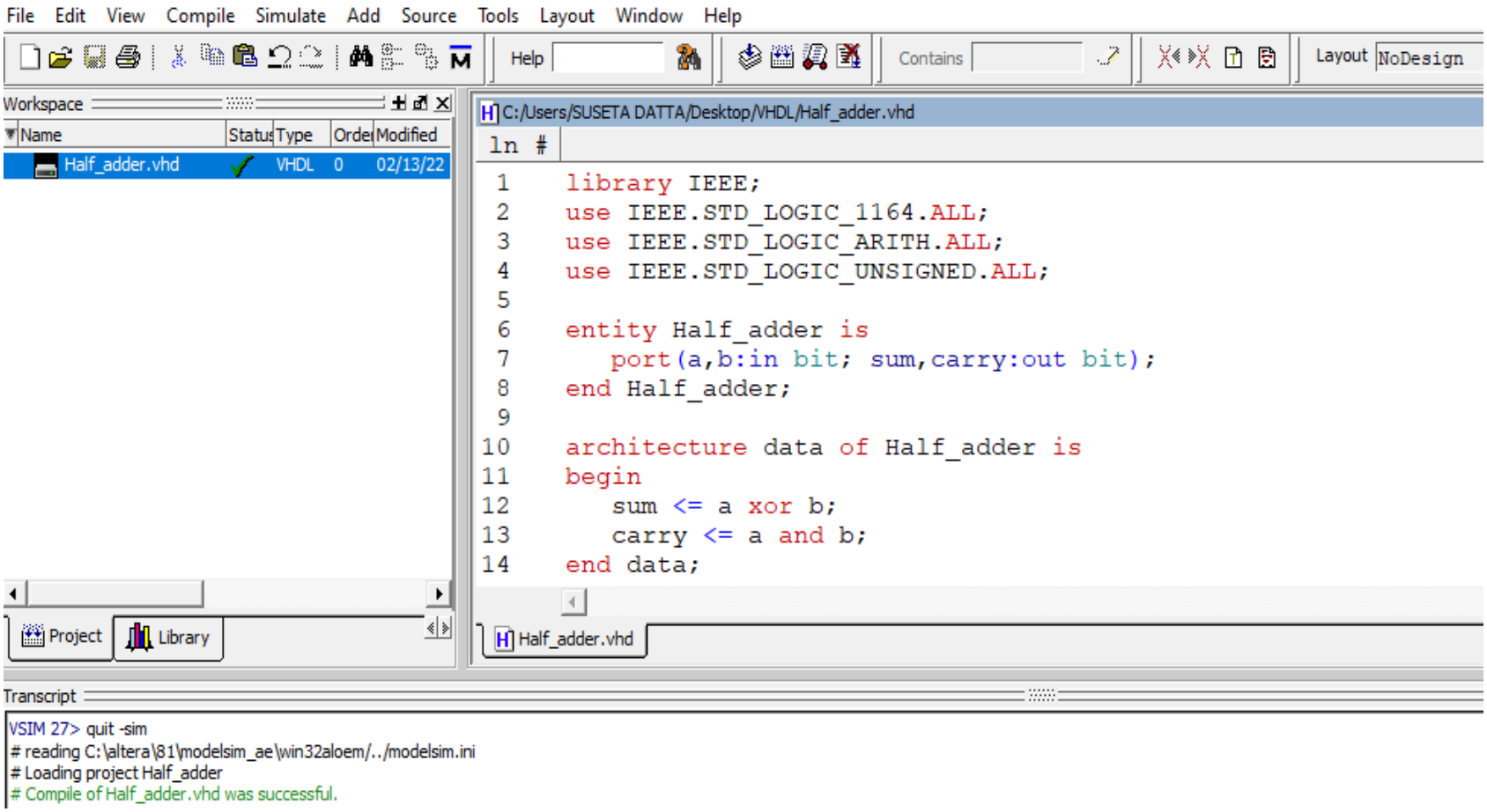
Half Adder

Symbol:

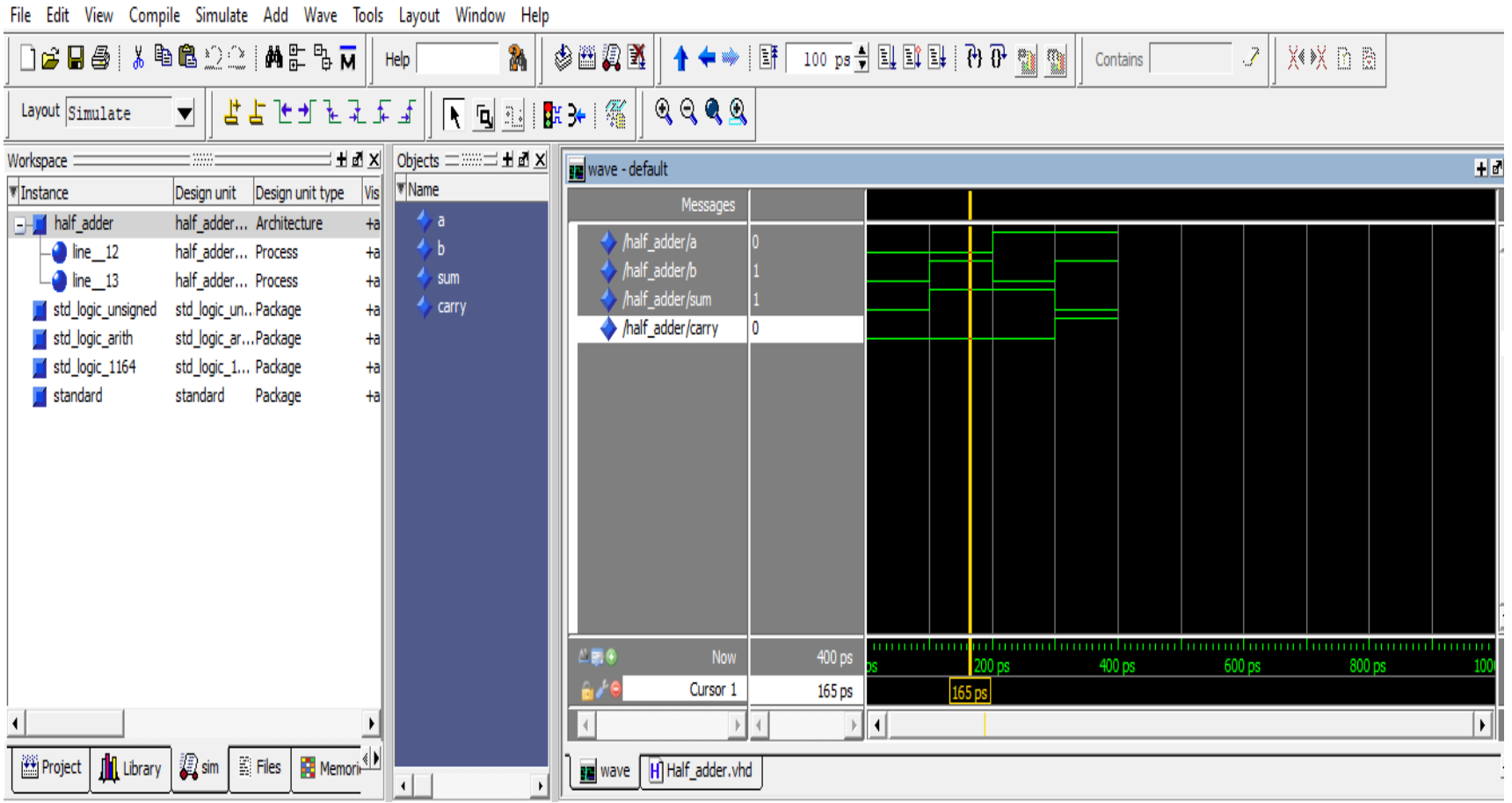
Truth Table:

A	B	C (Carry)	S (Sum)
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Code for Half Adder

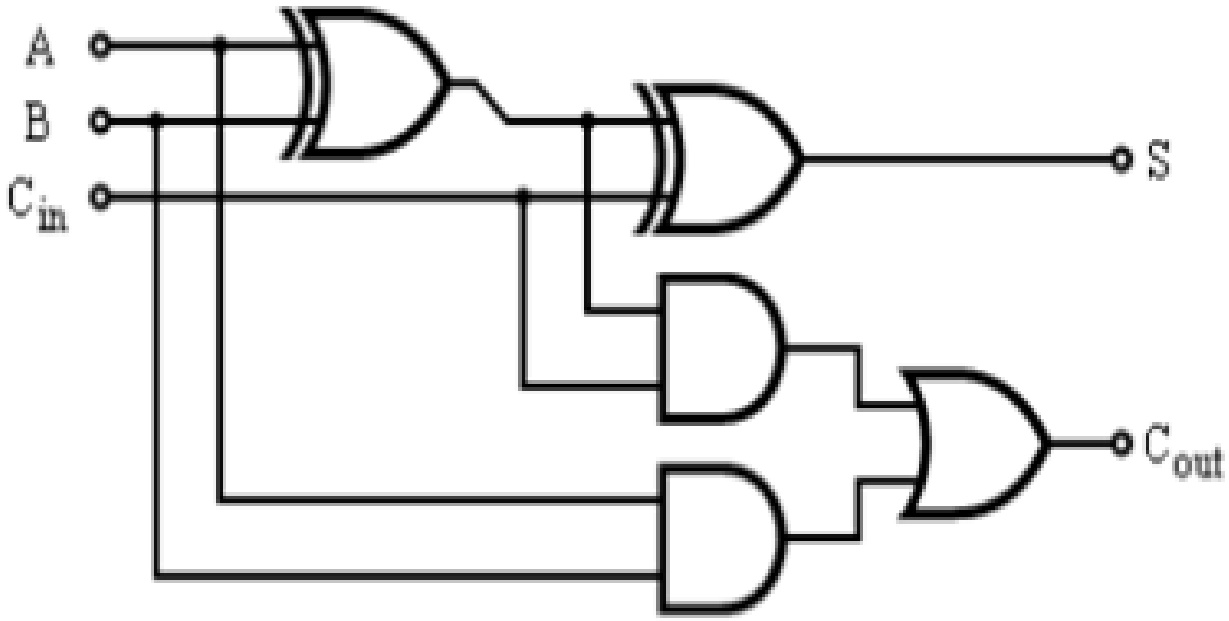


Output for Half Adder



Full Adder

Symbol:



Full Adder

Truth Table:

A	B	Cin	Cout	Sum
0	0	0	0	0
0	1	0	0	1
1	0	0	0	1
1	1	0	1	0
0	0	1	0	1
0	1	1	1	0
1	0	1	1	0
1	1	1	1	1

Thank You