Digital Design and Computer Organisation Laboratory

3rd Semester, Academic Year 2024

DATE : 11/8/2024

NAME : ABHISHEKP **SRN** : PES2UG23AM002 **SECTION**:A

WEEK1 **PROGRAM – 1**

WRITE A VERILOG PROGRAM TO MODEL A TWO INPUT AND GATE. GENERATE THE VVP OUTPUT AND SIMULATION WAVEFORM USING GTKWAVE. VERIFY THE OUTPUT AND WAVEFORM WITH THE AND GATE TRUTH TABLE

ANDGATE

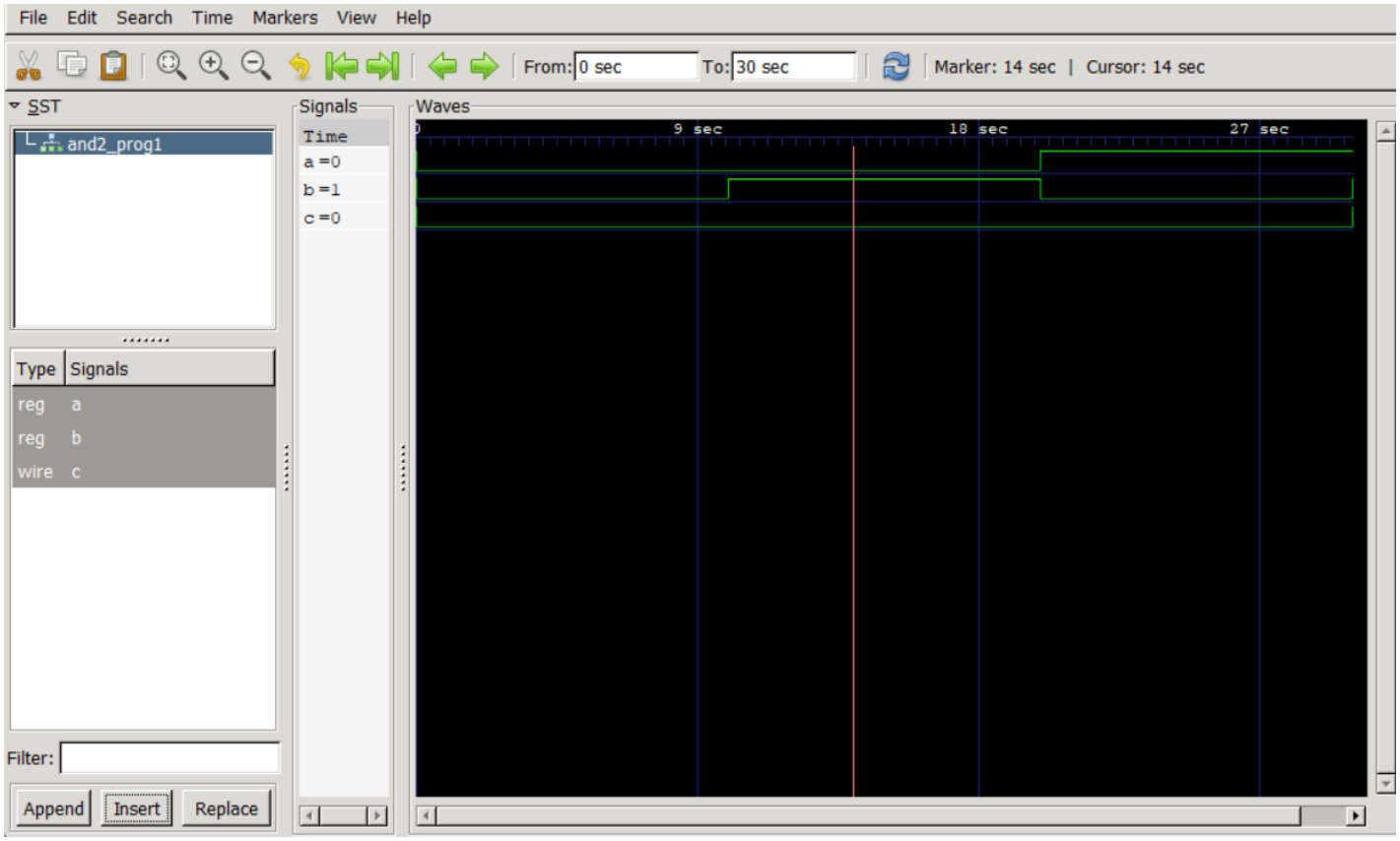
```
module andgate(
    input a,
    input b,
    output y
);
assign y = a&b;
endmodule
```

```
module and_test;
    reg a, b;
    wire y;
    and and_test (y, a, b);
    initial begin
      #0 a = 0;b = 0;
      #10 a = 0;b = 1;
      #10 a = 1;b = 0;
      #10 a = 1;b = 1;
    end
    initial begin
        $monitor($time, " a = %b, b = %b, y = %b", a, b, y);
    end
    initial begin
        $dumpfile("and_gate.vcd");
        $dumpvars(0, and_test);
    end
endmodule
```

ANDGATEVVPOUTPUTSCREENSHOT

```
VCD info: dumpfile and_gate.vcd opened for output.
                   0 a = 0, b = 0, y = 0
                  10 a = 0, b = 1, y = 0
                  20 a = 1, b = 0, y = 0
                  30 a = 1, b = 1, y = 1
```

ANDGATEGTKWAVE:



ANDGATETRUTHTABLE:

INPUTS		OUTPUTS
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

PROGRAM : 2

WRITE A VERILOG PROGRAM TO MODEL A TWO INPUT OR GATE. GENERATE THE VVP OUTPUT AND SIMULATION WAVEFORM USING GTKWAVE. VERIFY THE OUTPUT AND WAVEFORM WITH THE OR GATE TRUTH TABLE

ORGATECODE

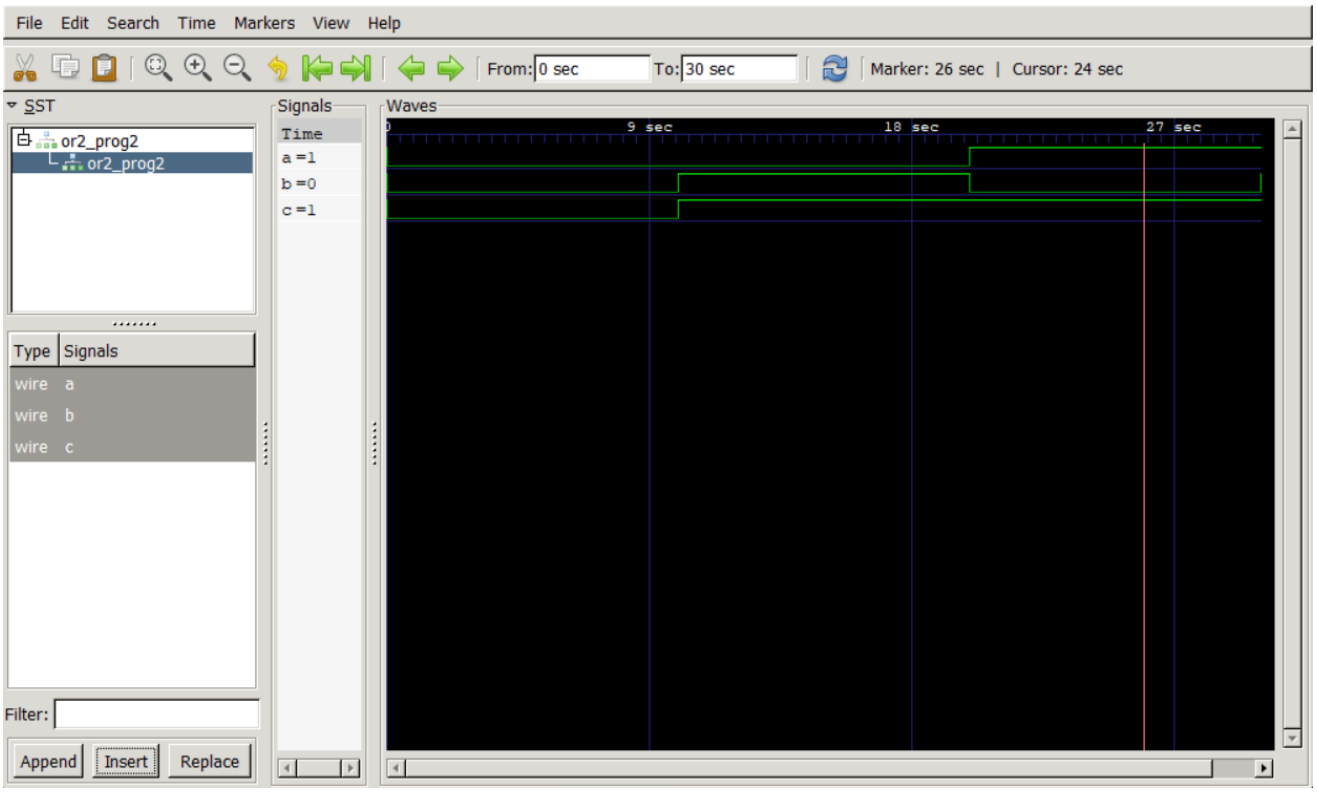
```
module orgate(
    input a,
    input b,
    output y
);
assign y = a|b;
endmodule
```

```
module or_test;
    reg a, b;
    wire y;
    or or_test (y, a, b);
    initial begin
      #0 a = 0;b = 0;
      #10 a = 0;b = 1;
      #10 a = 1;b = 0;
      #10 a = 1;b = 1;
    end
    initial begin
        $monitor($time, " a = %b, b = %b, y = %b", a, b, y);
    end
    initial begin
        $dumpfile("or_gate.vcd");
        $dumpvars(0, or_test);
    end
endmodule
```

ORGATEVVPOUTPUT

```
VCD info: dumpfile or_gate.vcd opened for output.
                   0 a = 0, b = 0, y = 0
                  10 a = 0, b = 1, y = 1
                  20 a = 1, b = 0, y = 1
                  30 a = 1, b = 1, y = 1
[Done] exit with code=0 in 0.275 seconds
```

ORGATEGTKWAVEOUTPUT:



ORGATETRUTHTABLE:

INPUTS		OUTPUTS
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

WRITEAVERILOG PROGRAMTO MODELANOTGATE. GENERATETHEVVPOUTPUTAND SIMULATIONWAVEFORMUSING GTKWAVE. VERIFYTHE OUTPUTANDWAVEFORMWITH THENOTGATETRUTHTABLE

NOTGATECODE:

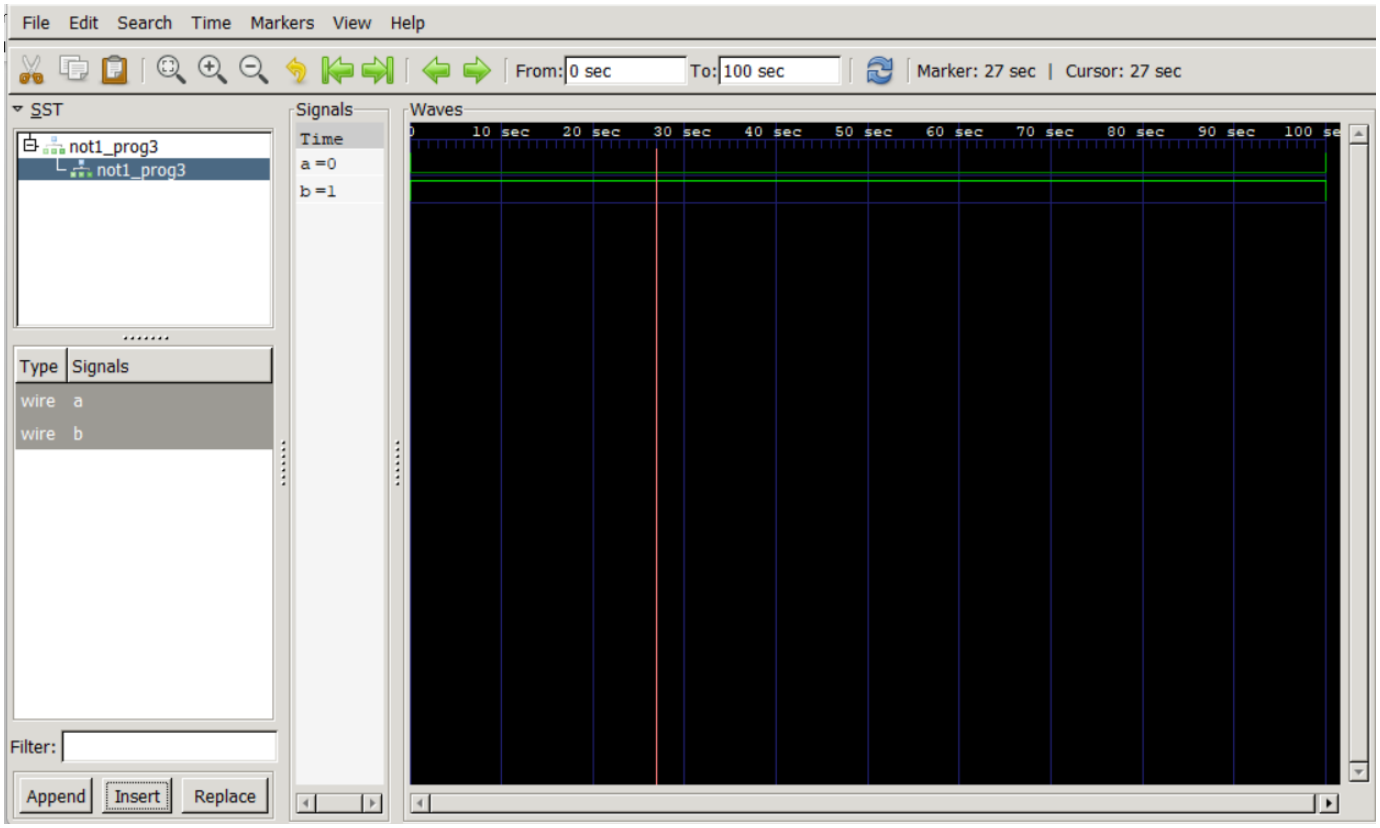
```
module notgate(
    input a,
    output y);
assign y=~a;
endmodule
```

```
module not_test;
    reg a;
    wire y;
    not not_test(y,a);
    initial begin
    #0 a=0;
    #10 a=1;
    end
    initial begin
        $monitor($time, " a = %b, y = %b", a, y);
    end
    initial begin
        $dumpfile("not_gate.vcd");
        $dumpvars(0, not_test);
    end
endmodule
```

NOTGATEVVPOUTPUT:

```
VCD info: dumpfile not_gate.vcd opened for output.
                   0 a = 0, y = 1
                  10 a = 1, y = 0
[Done] exit with code=0 in 0.425 seconds
```

NOTGATEGTKWAVE:



NOTGATETRUTHTABLE:

INPUT	OUTPUT
A	Y
0	1
1	0

WRITE A VERILOG PROGRAM TO MODEL A TWO INPUT NAND GATE. GENERATE THE VVP OUTPUT AND SIMULATION WAVEFORM USING GTKWAVE. VERIFY THE OUTPUT AND WAVEFORM WITH THE NAND GATE TRUTH TABLE

NAND GATE CODE:

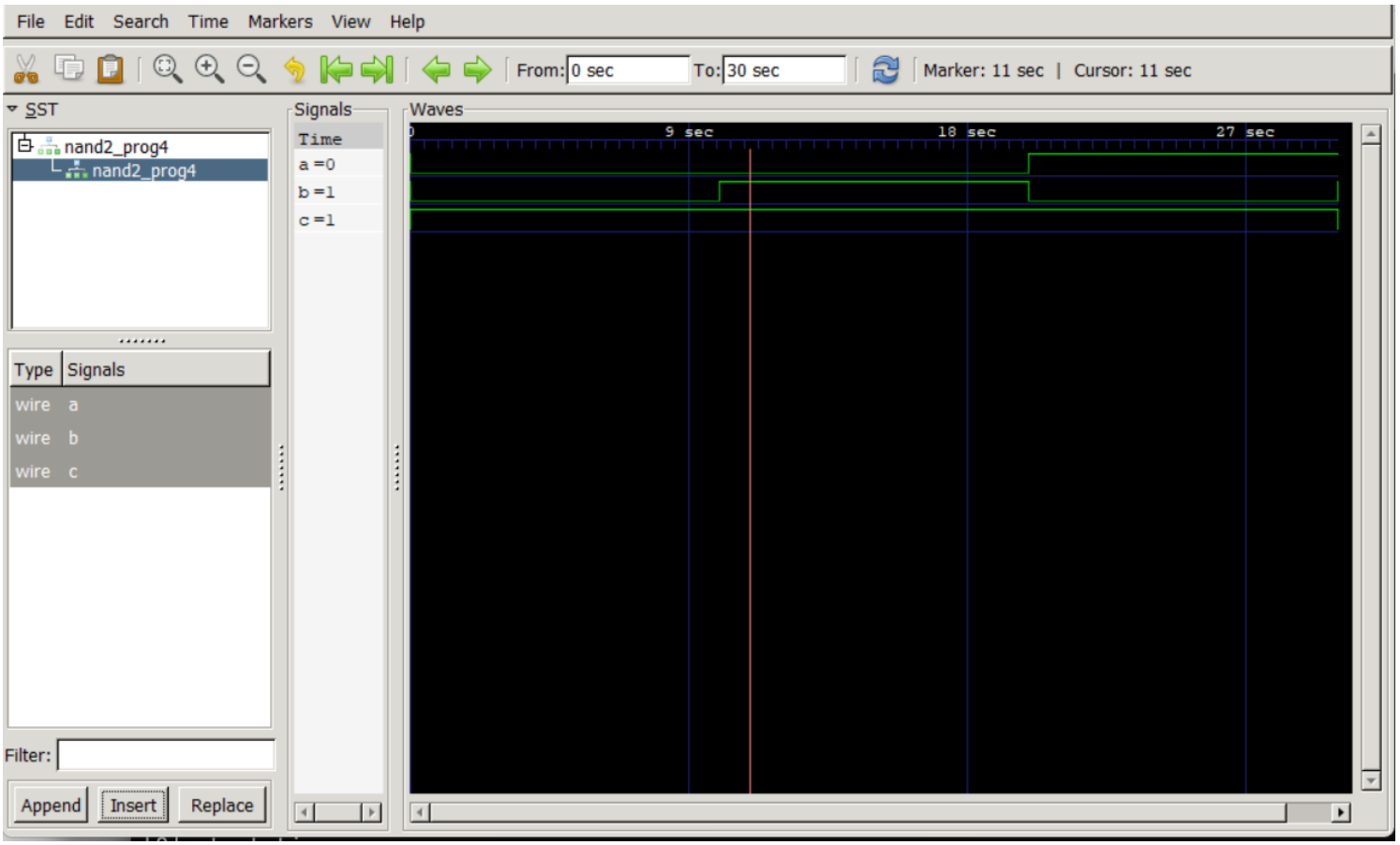
```
module nandgate(
    input a,
    input b,
    output y
);
assign y = ~(a&b);
endmodule
```

```
module nand_test;
    reg a, b;
    wire y;
    nand nand_test (y, a, b);
    initial begin
      #0 a = 0;b = 0;
      #10 a = 0;b = 1;
      #10 a = 1;b = 0;
      #10 a = 1;b = 1;
    end
    initial begin
        $monitor($time, " a = %b, b = %b, y = %b", a, b, y);
    end
    initial begin
        $dumpfile("nand_gate.vcd");
        $dumpvars(0, nand_test);
    end
endmodule
```

NAND GATE VVP OUTPUT:

```
VCD info: dumpfile nand_gate.vcd opened for output.
                   0 a = 0, b = 0, y = 1
                  10 a = 0, b = 1, y = 1
                  20 a = 1, b = 0, y = 1
                  30 a = 1, b = 1, y = 0
[Done] exit with code=0 in 0.263 seconds
```

NAND GATE GTKWAVE OUTPUT:



NANDGATETRUTHTABLE:

INPUTS		OUTPUTS
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

WRITE A VERILOG PROGRAM TO MODEL A TWO INPUT NOR GATE. GENERATE THE VVP OUTPUT AND SIMULATION WAVEFORM USING GTKWAVE. VERIFY THE OUTPUT AND WAVEFORM WITH THE NOR GATE TRUTH TABLE

NOR GATE CODE:

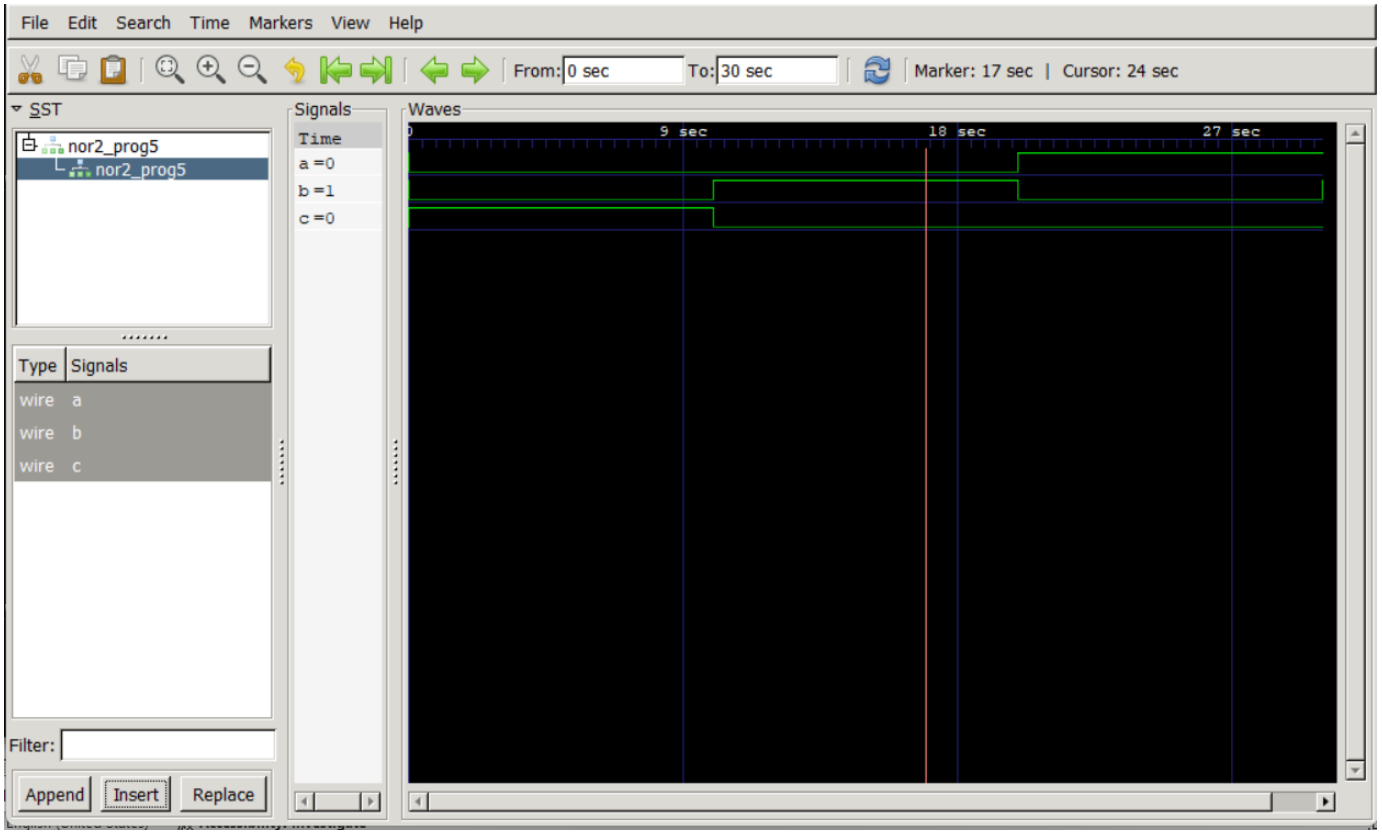
```
module norgate(
    input a,
    input b,
    output y
);
assign y = ~(a|b);
endmodule
```

```
module nor_test;
    reg a, b;
    wire y;
    nor nor_test (y, a, b);
    initial begin
      #0 a = 0;b = 0;
      #10 a = 0;b = 1;
      #10 a = 1;b = 0;
      #10 a = 1;b = 1;
    end
    initial begin
        $monitor($time, " a = %b, b = %b, y = %b", a, b, y);
    end
    initial begin
        $dumpfile("nor_gate.vcd");
        $dumpvars(0, nor_test);
    end
endmodule
```

NOR GATE VVP OUTPUT:

```
VCD info: dumpfile nor_gate.vcd opened for output.
                   0 a = 0, b = 0, y = 1
                  10 a = 0, b = 1, y = 0
                  20 a = 1, b = 0, y = 0
                  30 a = 1, b = 1, y = 0
[Done] exit with code=0 in 0.284 seconds
```

NOR GATE GTKWAVE OUTPUT:



NORGATETRUTHTABLE:

INPUTS		OUTPUTS
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

WRITE A VERILOG PROGRAM TO MODEL A TWO INPUT XOR GATE. GENERATE THE VVP OUTPUT AND SIMULATION WAVEFORM USING GTKWAVE. VERIFY THE OUTPUT AND WAVEFORM WITH THE AND GATE TRUTH TABLE

XOR GATE CODE:

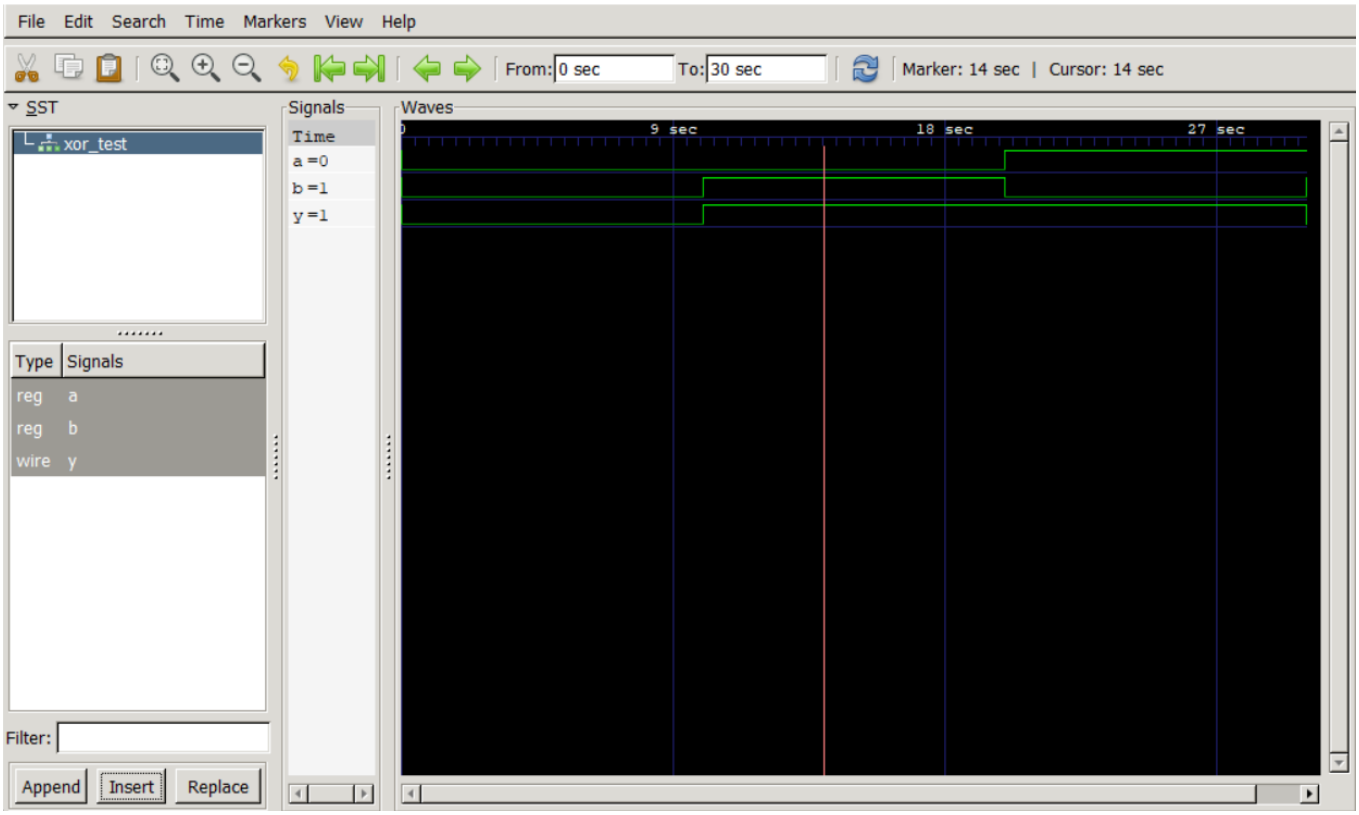
```
module Xorgate(
    input a,
    input b,
    output y
);
assign y = a^b;
endmodule
```

```
module xor_test;
    reg a, b;
    wire y;
    xor xor_test (y, a, b);
    initial begin
      #0 a = 0;b = 0;
      #10 a = 0;b = 1;
      #10 a = 1;b = 0;
      #10 a = 1;b = 1;
    end
    initial begin
        $monitor($time, " a = %b, b = %b, y = %b", a, b, y);
    end
    initial begin
        $dumpfile("xor_gate.vcd");
        $dumpvars(0, xor_test);
    end
endmodule
```

XOR GATE VVP OUTPUT:

```
VCD info: dumpfile xor_gate.vcd opened for output.
                   0 a = 0, b = 0, y = 0
                  10 a = 0, b = 1, y = 1
                  20 a = 1, b = 0, y = 1
                  30 a = 1, b = 1, y = 0
[Done] exit with code=0 in 0.196 seconds
```

XOR GATE GTKWAVE OUTPUT:



XORGATETRUTHTABLE:

INPUTS		OUTPUTS
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

WRITE A VERILOG PROGRAM TO MODEL A TWO INPUT XNOR GATE. GENERATE THE VVP OUTPUT AND SIMULATION WAVEFORM USING GTKWAVE. VERIFY THE OUTPUT AND WAVEFORM WITH THE AND GATE TRUTH TABLE

XNOR GATE CODE:

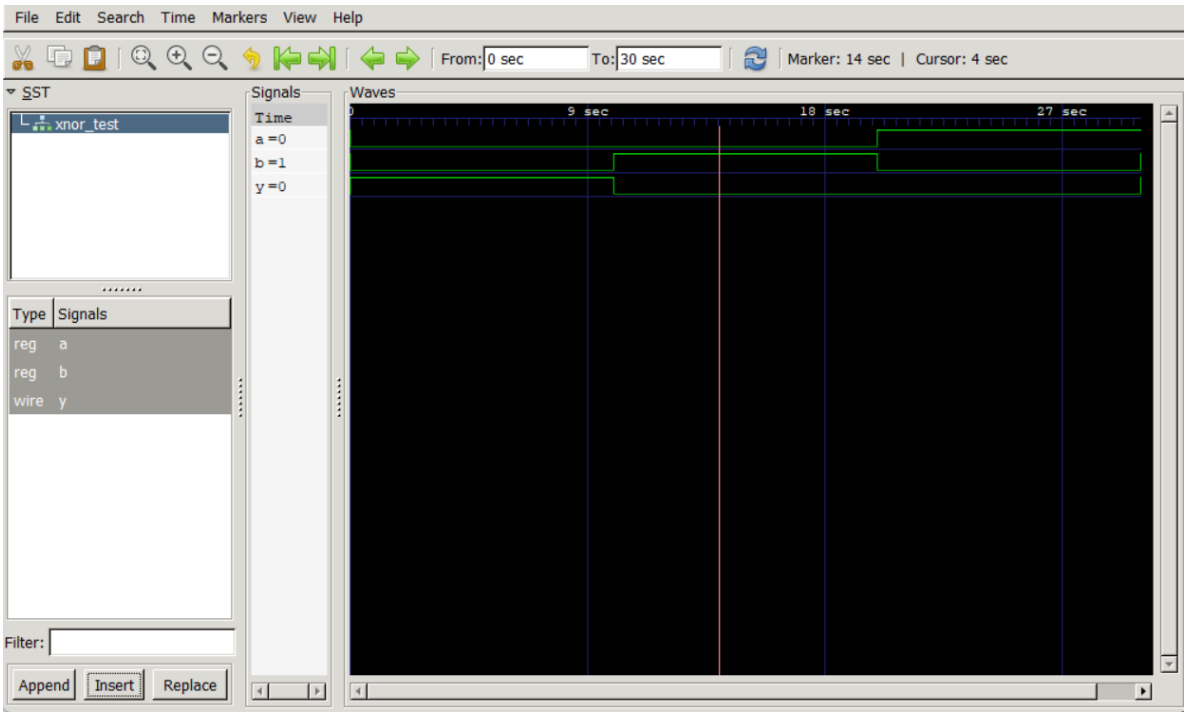
```
module xnorgate(
    input a,
    input b,
    output y
);
assign y = ~(a^b);
endmodule
```

```
module xnor_test;
    reg a, b;
    wire y;
    xnor xnor_test (y, a, b);
    initial begin
      #0 a = 0;b = 0;
      #10 a = 0;b = 1;
      #10 a = 1;b = 0;
      #10 a = 1;b = 1;
    end
    initial begin
        $monitor($time, " a = %b, b = %b, y = %b", a, b, y);
    end
    initial begin
        $dumpfile("xnor_gate.vcd");
        $dumpvars(0, xnor_test);
    end
endmodule
```

XNOR GATE VVP OUTPUT:

```
VCD info: dumpfile xnor_gate.vcd opened for output.
                   0 a = 0, b = 0, y = 1
                  10 a = 0, b = 1, y = 0
                  20 a = 1, b = 0, y = 0
                  30 a = 1, b = 1, y = 1
[Done] exit with code=0 in 0.218 seconds
```

XNOR GATE GTKWAVE OUTPUT:



XNORGATETRUTHTABLE:

INPUTS		OUTPUTS
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

Disclaimer:

· The programs and output submitted is duly written, verified and executed by me.

· I have not copied from any of my peers nor from the external resource such as internet.

· If found plagiarized, I will abide with the disciplinary action of the University.

NAME : ABHISHEK P

SRN :

PES2UG23AM002

SECTION : A