Run 2

(a);

Verilog Code :

module NOTgate (z,a);

input a;

output z;

not #1 (z,a);

endmodule

Test bench :

module NOT\_testbench; //Arshdeep Singh 2020A7PS0144U

reg p;

wire q;

initial

begin

$dumpfile("’dump.vcd"); $dumpvars;

#000 p=0;

#100 p=1;

#100 p=0;

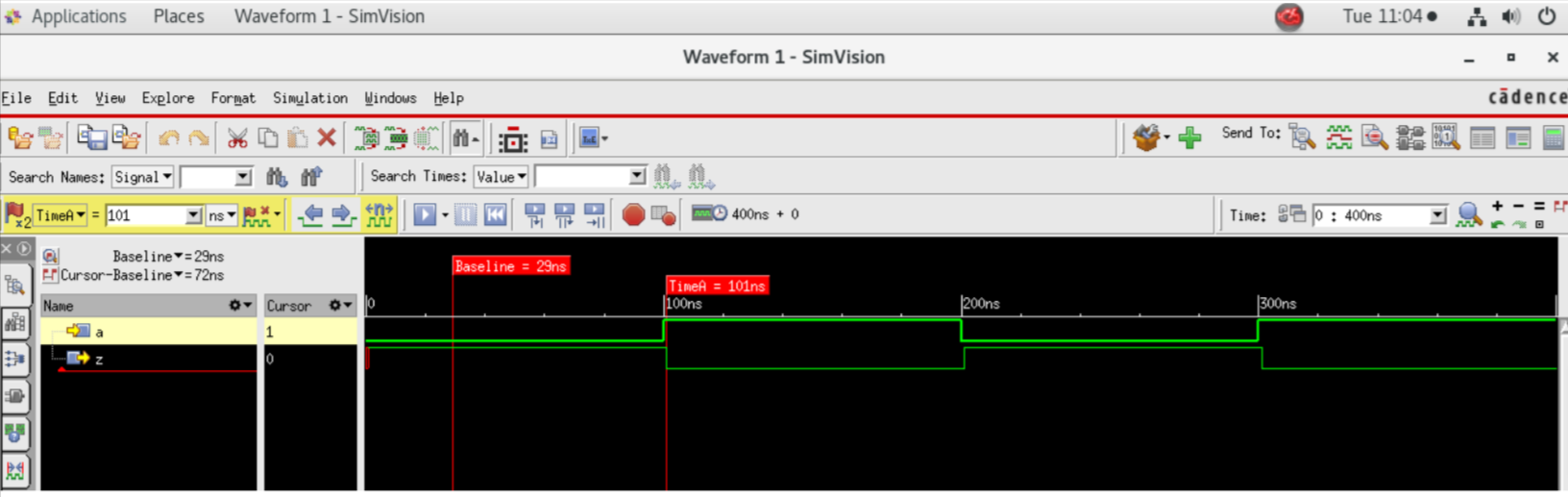
#100 p=1;

#100 $stop;

end

NOT\_testbench n1(q,p);

Endmodule



Run 3

(a)

Verilog Code:

module OR\_2\_behavioral (output reg Z, input E, F);

always @ (E or F)

begin

if (E==0 & F==0) begin

#10 Z = 0;

end

else

#10 Z = 1;

end

endmodule

Testbench :

module testbench\_ORgate; //Arshdeep Singh 2020A7PS0144U

reg p,q;

wire r;

initial

begin

$dumpfile (“dump. vcd");

$dumpvars (1, testbench\_ORgate);

#000 p = 0; q = 0;

#100 p = 0; q = 1;

#100 p = 1; q = 0;

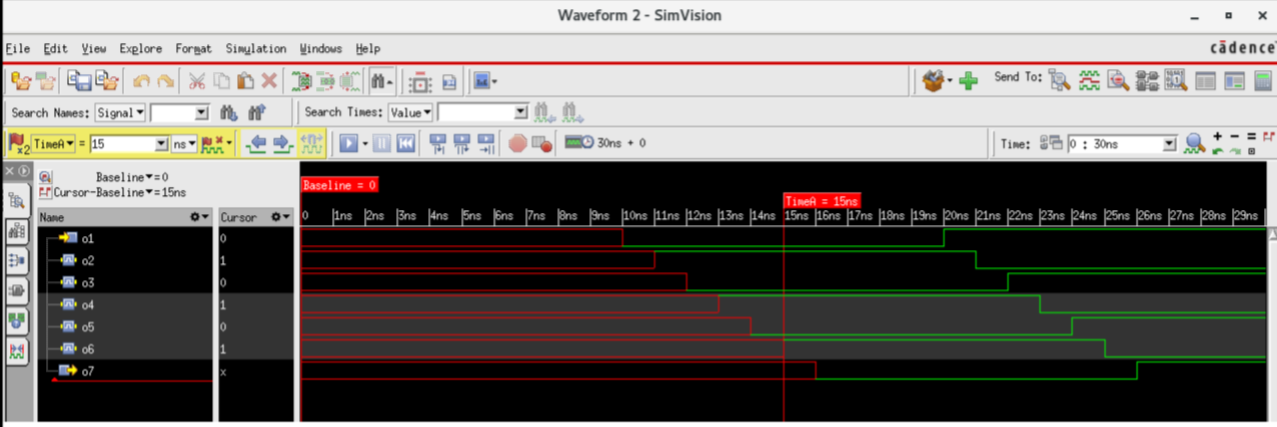
#100 p = 1; q = 1;

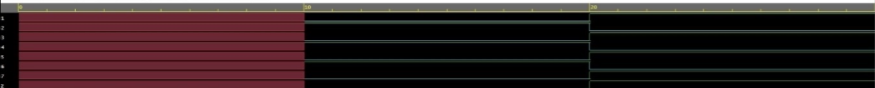
#100 S$stop;

end

OR\_2\_behavioral U3(r.p,a);

endmodule





(b);

Verilog Code:

module OR\_3\_structural (output 2, input A,B,C);

wire Pp;

OR\_2\_gate Gi(p,A,B);

OR\_2\_gate G2(Z,C,p);

endmodule

module OR\_2\_gate(output X, input L,M);

or #2 (X,L,M)3

endmodule

Testbench

module testbench\_ORn3; //Arshdeep Singh 2020A7PS0144U

reg inl,in2,in3

wire o1;

initial begin

$dumpfile (“dump.vcd"); $dumpvars (1, testbench\_OR3);

#000 in3 = 0; in2 = 0; inl = 0;

#100 in3 = 0; in2 = 0; inl = 1;

#100 in3 = 0; in2 = 1; ini = 0;

#100 in3 = 0; in2 = 1; inl = 1;

#100 in3 = 1; in2 = 0; inl = 0;

#100 in3 = 1; in2 = 0; inl = 1;

#100 in3 = 1; in2 = 1; inl = 0;

#100 in3 = 1; in2 = 1; inl = 1;

#100 $stop;

end

OR\_3\_structural U1(o1,in1,in2,in3);

endmodule

(c);

Verilog Code;

module NOT\_testbench (output z, input a);

assign z = ~a;

endmodule

TestBench

module NOT\_testbench; //Arshdeep Singh 2020A7PS0144U

reg p;

wire q;

initial

begin

$dumpfile("’dump.vcd"); $dumpvars;

#000 p=0;

#100 p=1;

#100 p=0;

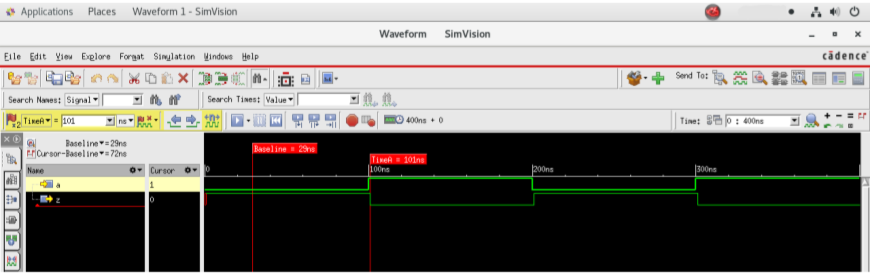
#100 p=1;

#100 $stop;

end

NOT\_testbench n1(q,p);

endmodule



(d);

Verilog Code;

module NOT\_BM (output reg Z, input A);

always @ (A) begin

If (A == 0 ) begin

Z=1;

end

else

Z=0;

end

endmodule

TestBench

module NOT\_testbench; //Arshdeep Singh 2020A7PS0144U

reg p;

wire q;

initial

begin

$dumpfile("’dump.vcd"); $dumpvars;

#000 p=0;

#100 p=1;

#100 p=0;

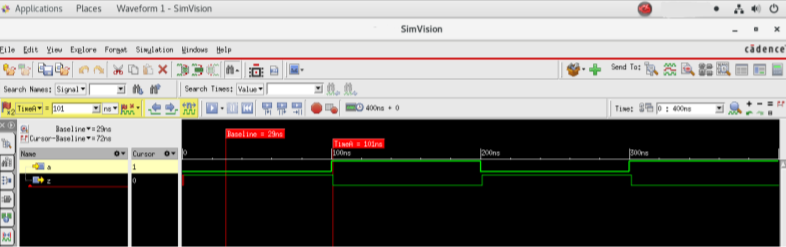
#100 p=1;

#100 $stop;

end

NOT\_testbench n1(q,p);

endmodule



EDA PLAYGROUND LINKS

<https://www.edaplayground.com/x/BgM_> - NAND

<https://www.edaplayground.com/x/ZtMG> - XOR

<https://www.edaplayground.com/x/fH98> - 1’s complement of 8 bit number

<https://www.edaplayground.com/x/PcYw> - Not gate (dataflow modeling)

<https://www.edaplayground.com/x/7xyj> - Not gate (behavioral modeling)