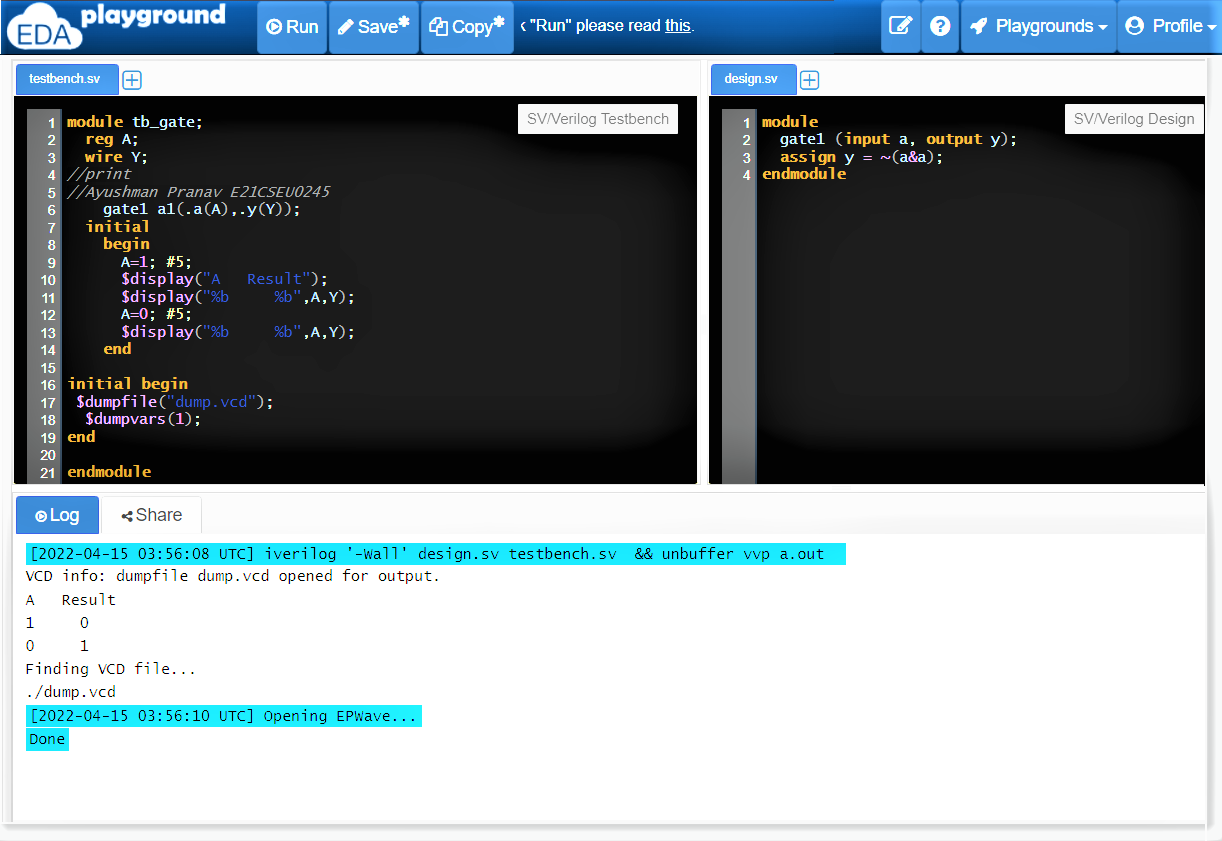
|  |  |
| --- | --- |
| Course- BTech | Type- Core |
| Course Code: CSET 105 | Course Name- Digital Logic Design Lab |
| Year- 2022  Name – Ayushman Pranav  Rollno. – E21CSEU0245  Batch – EB10 | Semester- Even |

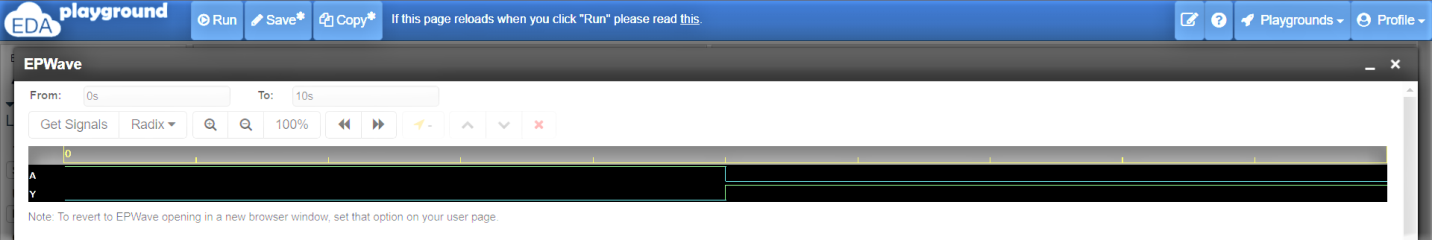
**Lab Assignment 4.1**

Perform the following operations:

* 1. Derive the Boolean expression.
  2. Write the truth table for the above expression.
  3. Write a Verilog code for each Boolean expression and then test using wave form and compare with truth table whether your circuit produced same output or not?

1. Represent the following gates using only NAND gates
2. NOT





**Testbench.sv**

module tb\_gate;

reg A;

wire Y;

//print

//Ayushman Pranav E21CSEU0245

gate1 a1(.a(A),.y(Y));

initial

begin

A=1; #5;

$display("A Result");

$display("%b %b",A,Y);

A=0; #5;

$display("%b %b",A,Y);

end

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

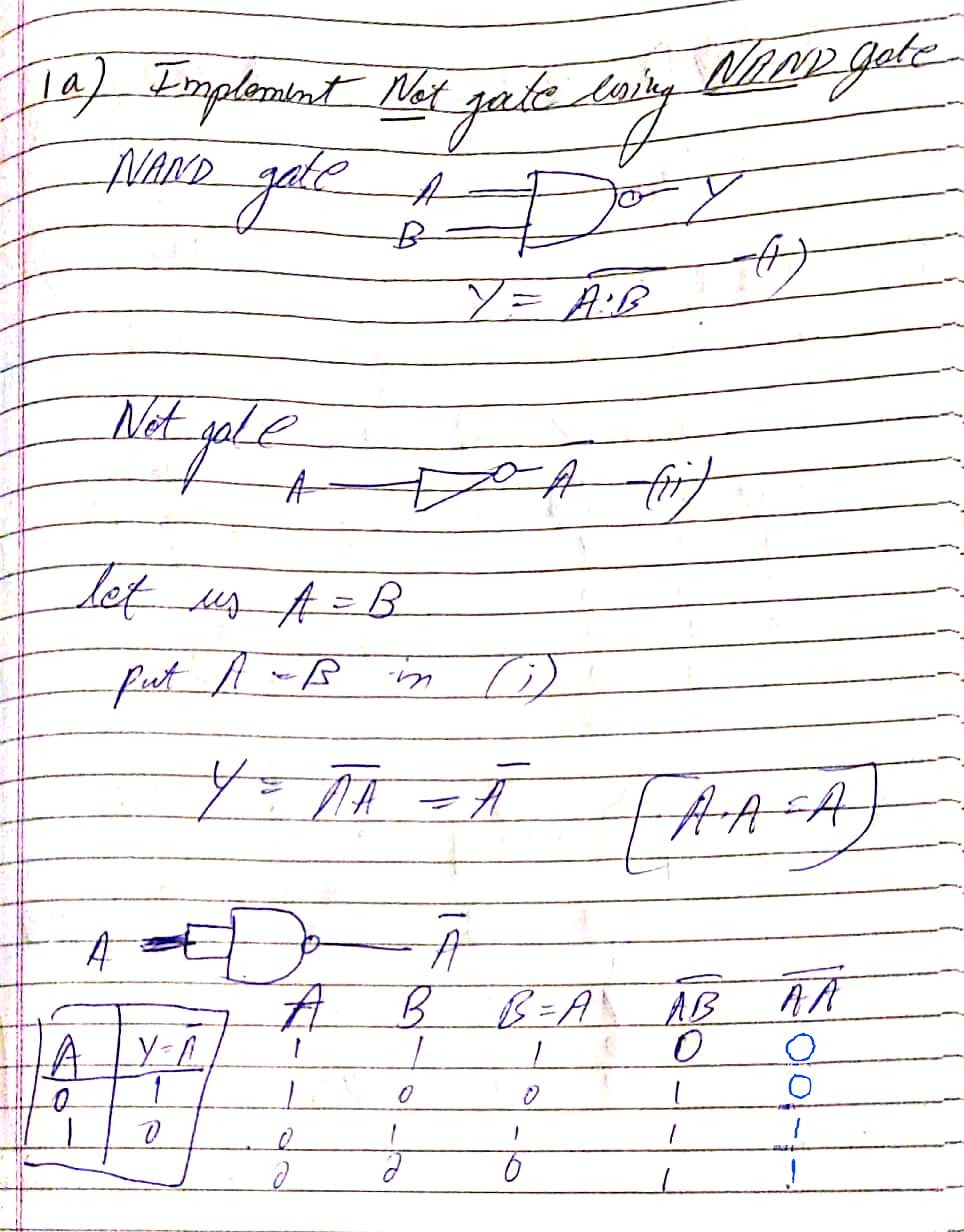
**design.sv**

module

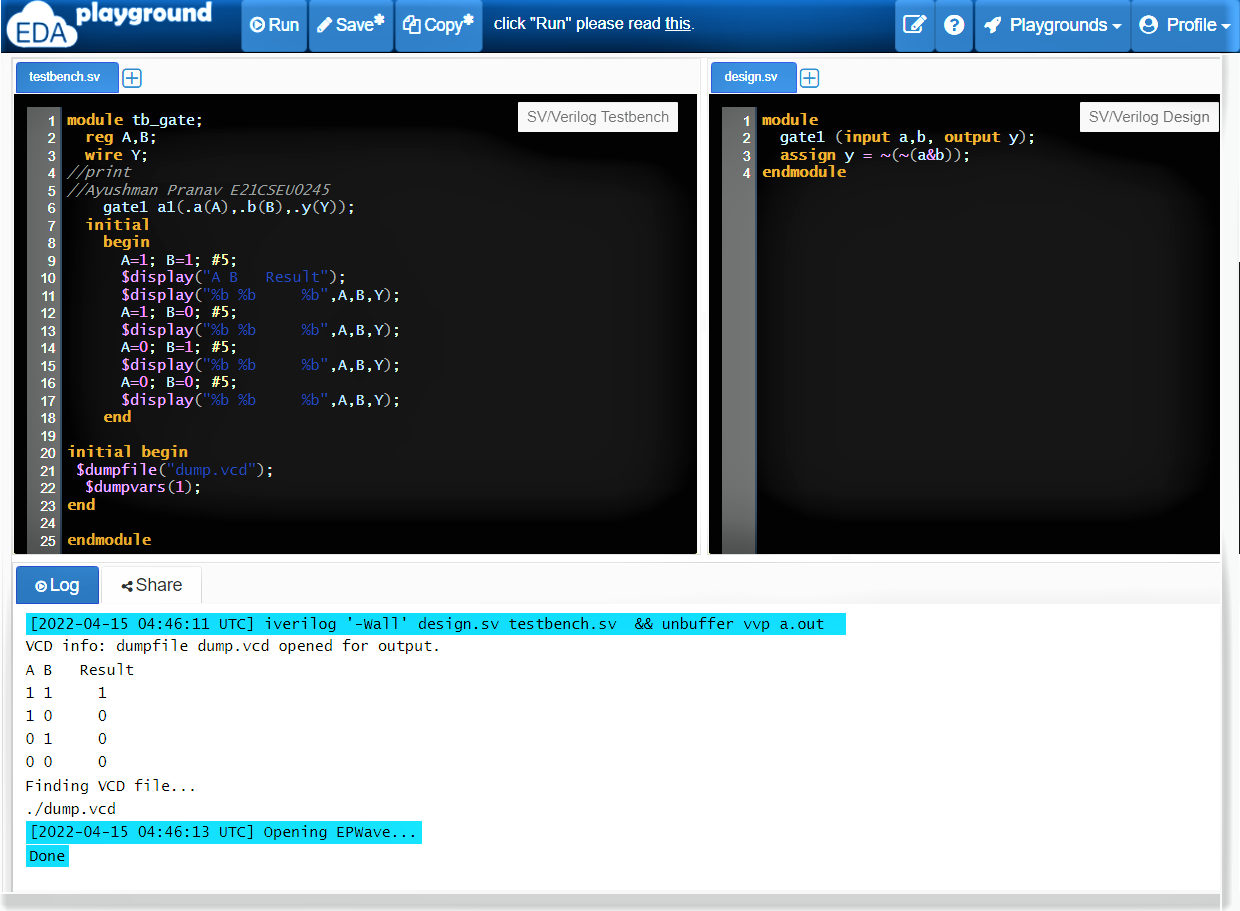
gate1 (input a, output y);

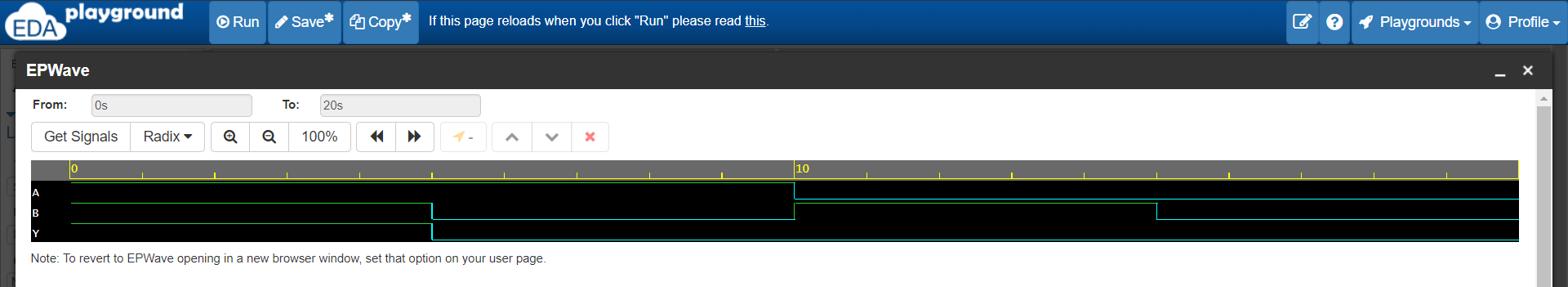
assign y = ~(a&a);

endmodule



1. AND





**Testbench.sv**

module tb\_gate;

reg A,B;

wire Y;

//print

//Ayushman Pranav E21CSEU0245

gate1 a1(.a(A),.b(B),.y(Y));

initial

begin

A=1; B=1; #5;

$display("A B Result");

$display("%b %b %b",A,B,Y);

A=1; B=0; #5;

$display("%b %b %b",A,B,Y);

A=0; B=1; #5;

$display("%b %b %b",A,B,Y);

A=0; B=0; #5;

$display("%b %b %b",A,B,Y);

end

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

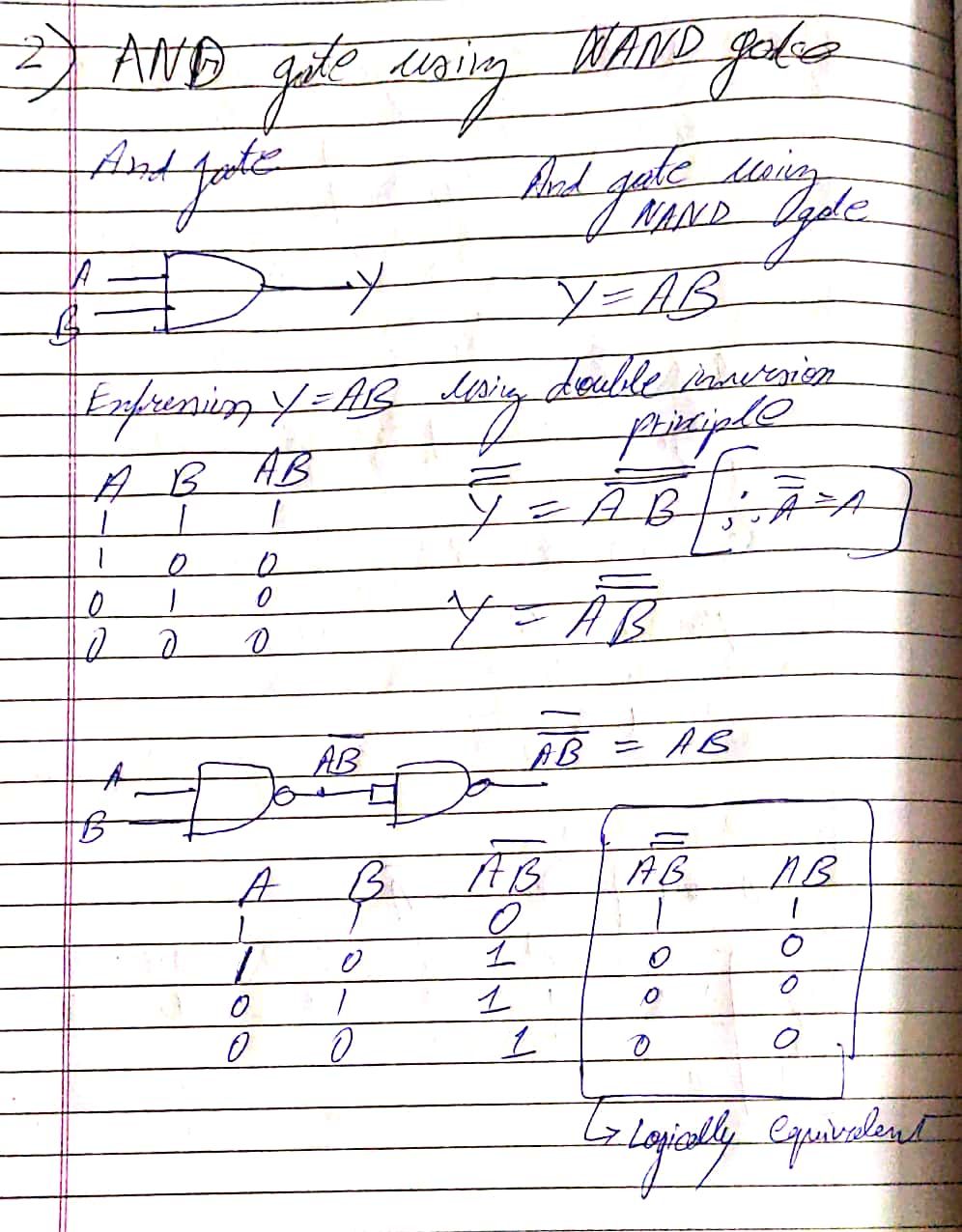
**Design.sv**

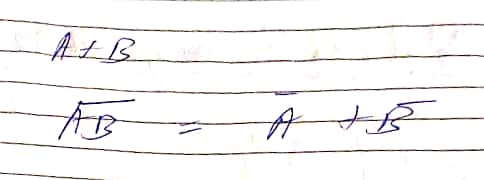
module

gate1 (input a,b, output y);

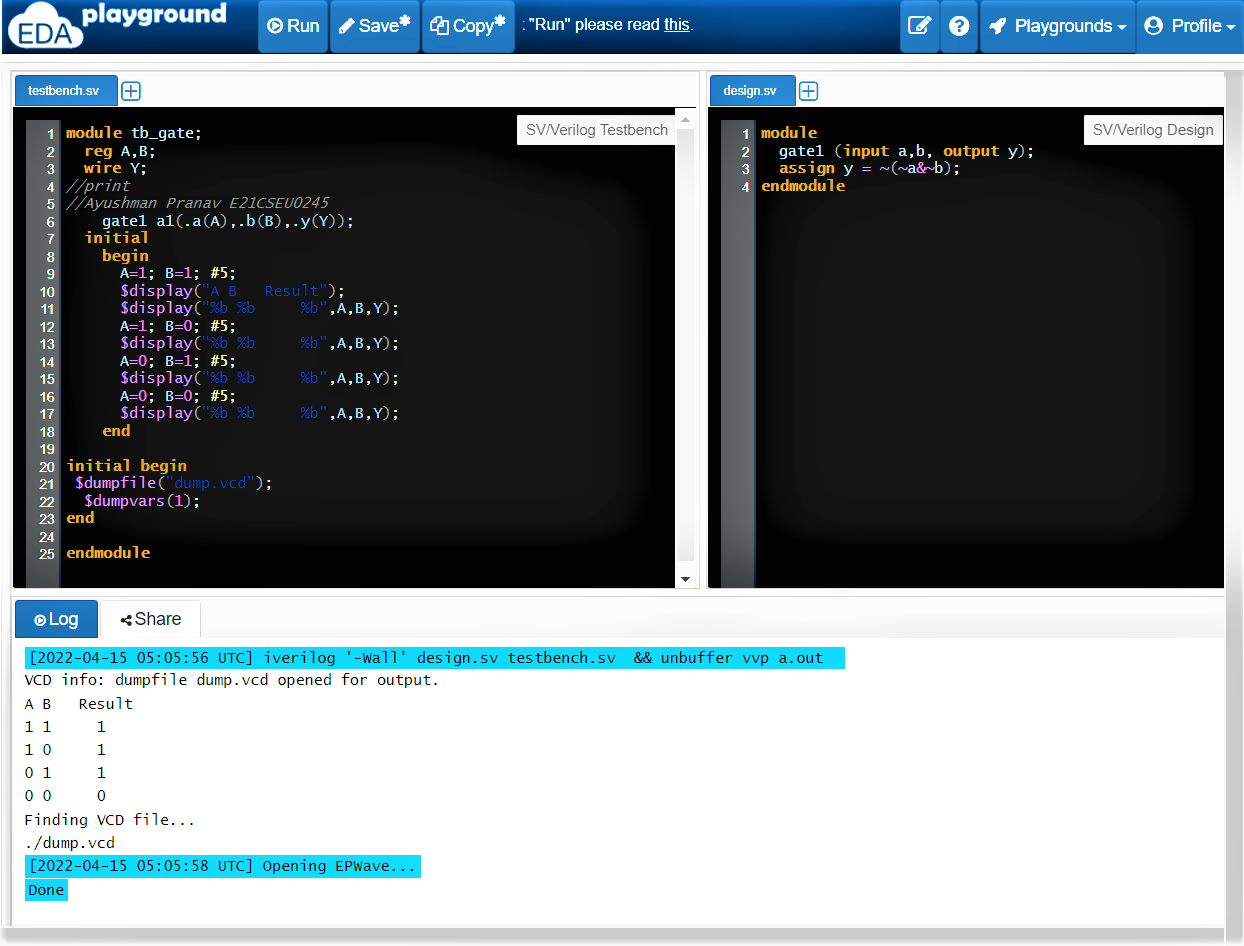
assign y = ~(~(a&b));

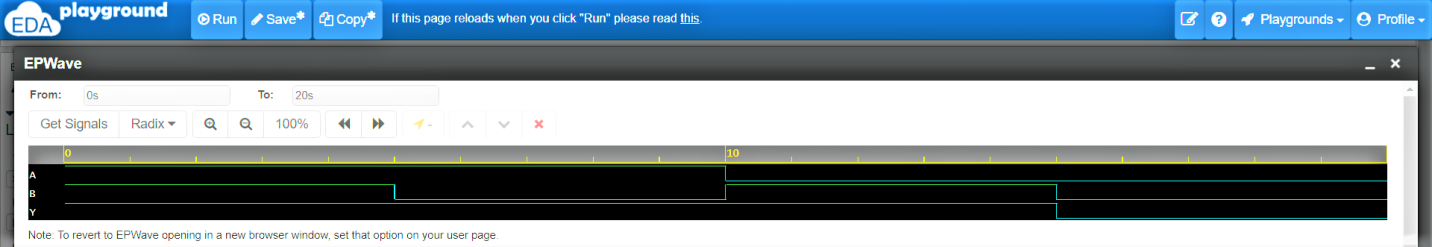
endmodule





1. OR





**Testbench.sv**

module tb\_gate;

reg A,B;

wire Y;

//print

//Ayushman Pranav E21CSEU0245

gate1 a1(.a(A),.b(B),.y(Y));

initial

begin

A=1; B=1; #5;

$display("A B Result");

$display("%b %b %b",A,B,Y);

A=1; B=0; #5;

$display("%b %b %b",A,B,Y);

A=0; B=1; #5;

$display("%b %b %b",A,B,Y);

A=0; B=0; #5;

$display("%b %b %b",A,B,Y);

end

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

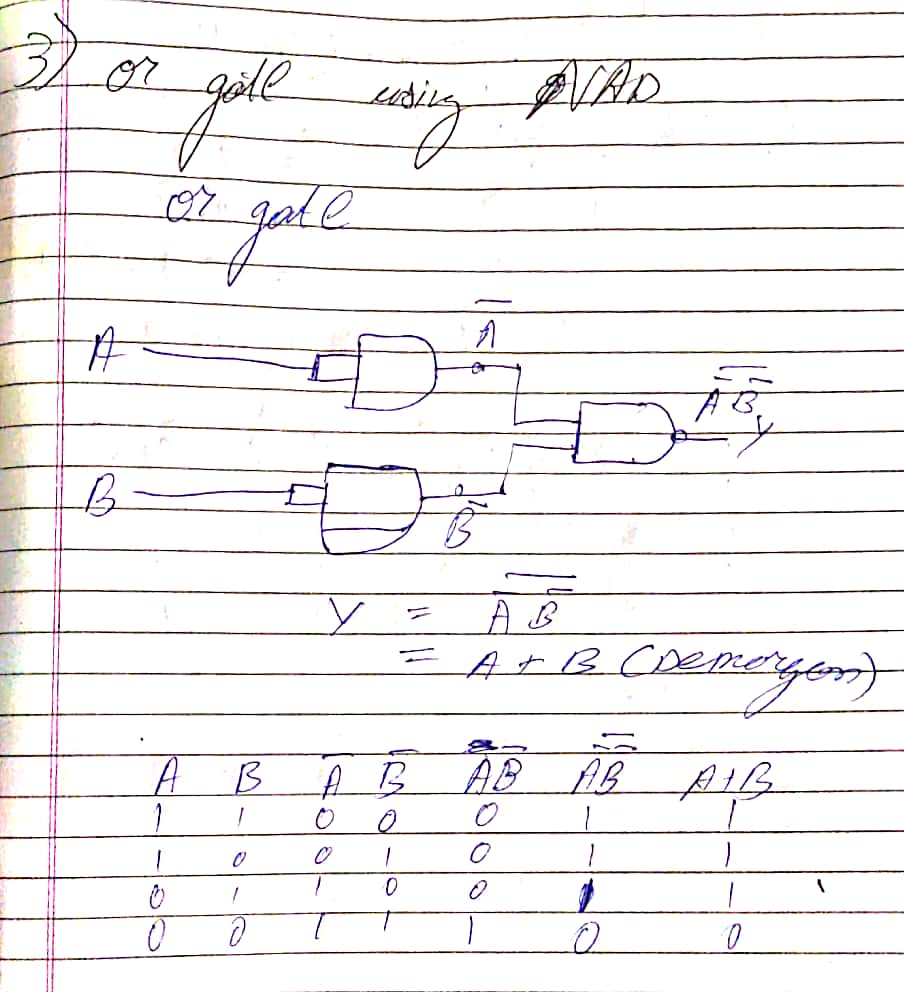
**Design.sv**

module

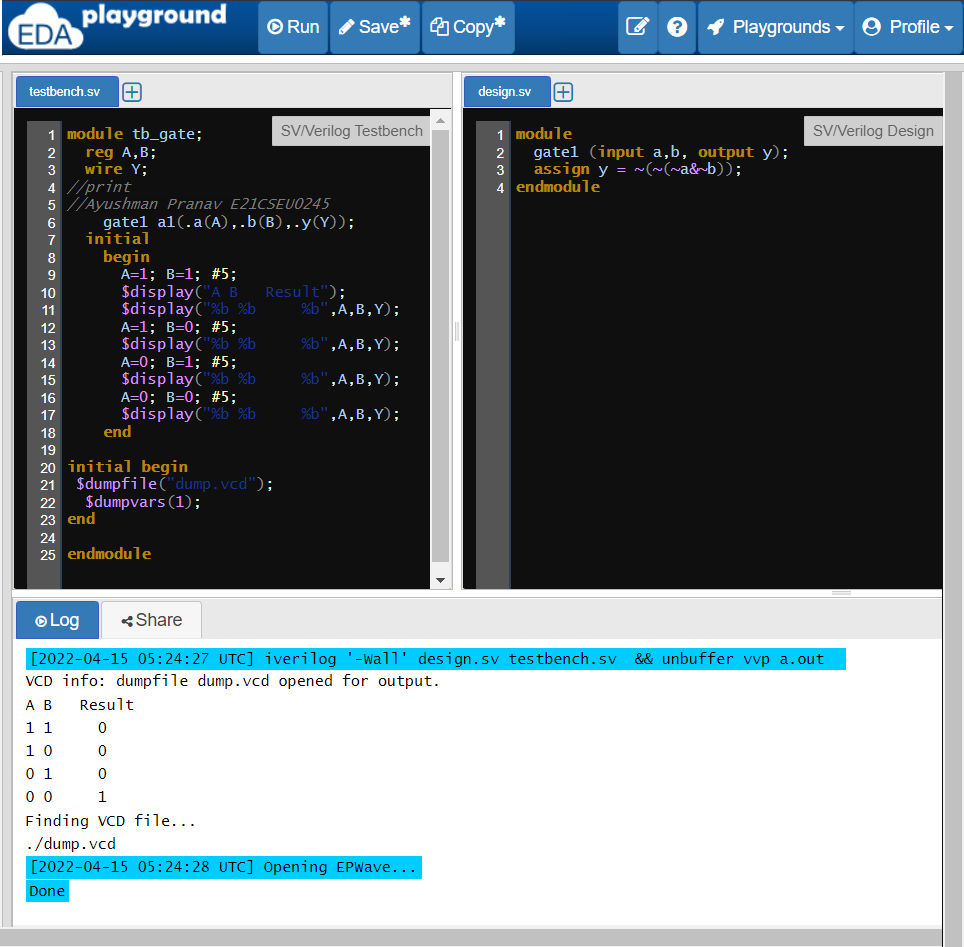
gate1 (input a,b, output y);

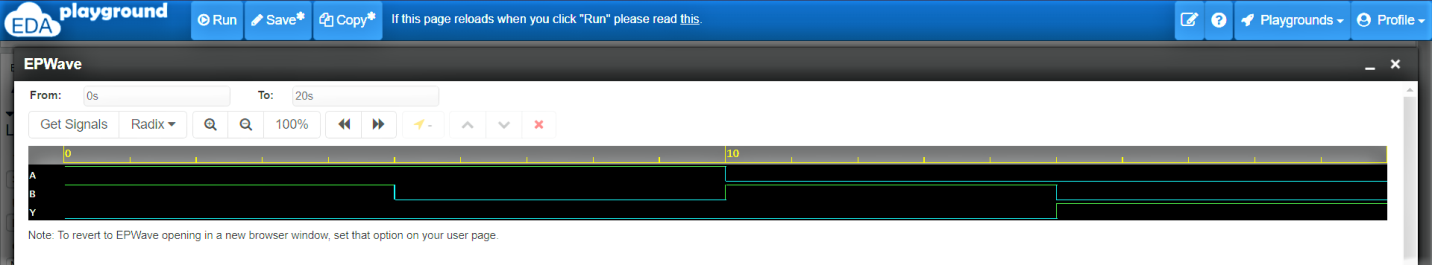
assign y = ~(~a&~b);

endmodule



1. NOR





**Testbench.sv**

module tb\_gate;

reg A,B;

wire Y;

//print

//Ayushman Pranav E21CSEU0245

gate1 a1(.a(A),.b(B),.y(Y));

initial

begin

A=1; B=1; #5;

$display("A B Result");

$display("%b %b %b",A,B,Y);

A=1; B=0; #5;

$display("%b %b %b",A,B,Y);

A=0; B=1; #5;

$display("%b %b %b",A,B,Y);

A=0; B=0; #5;

$display("%b %b %b",A,B,Y);

end

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

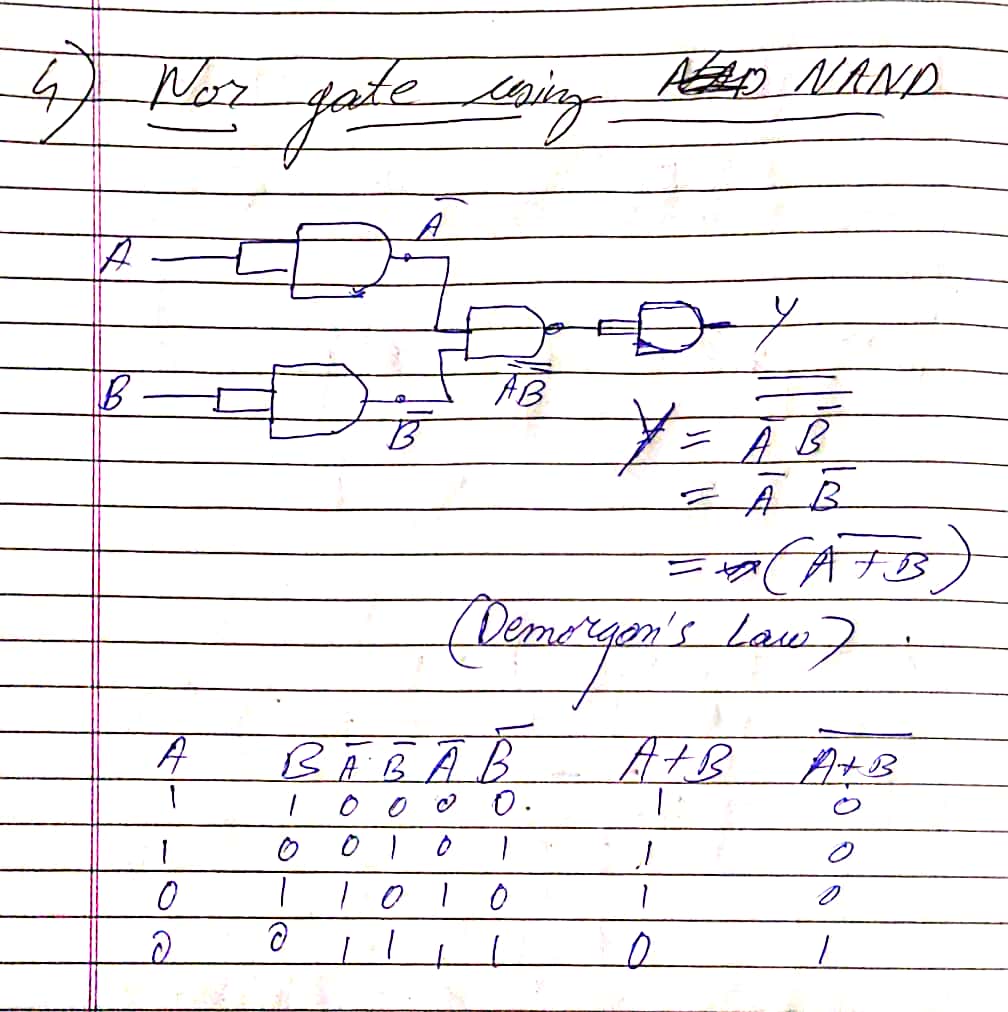
**Design.sv**

module

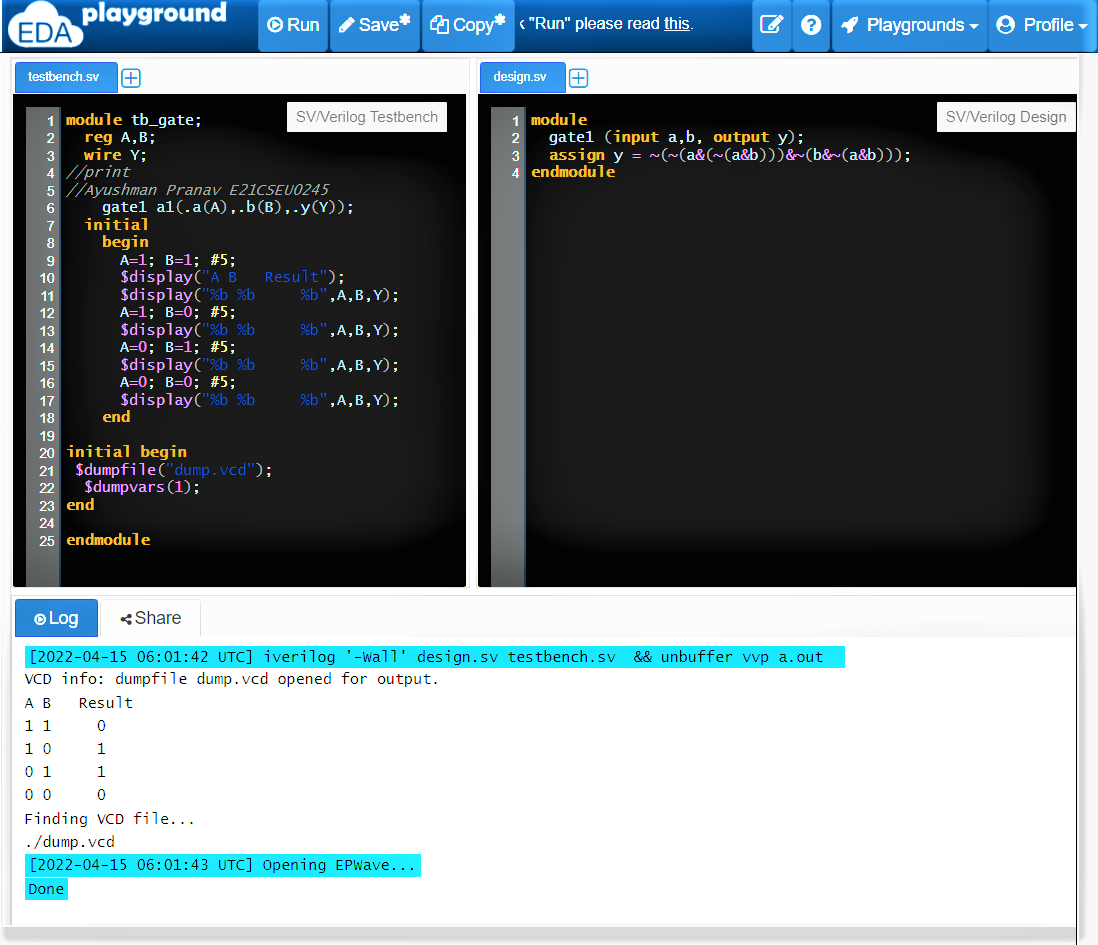
gate1 (input a,b, output y);

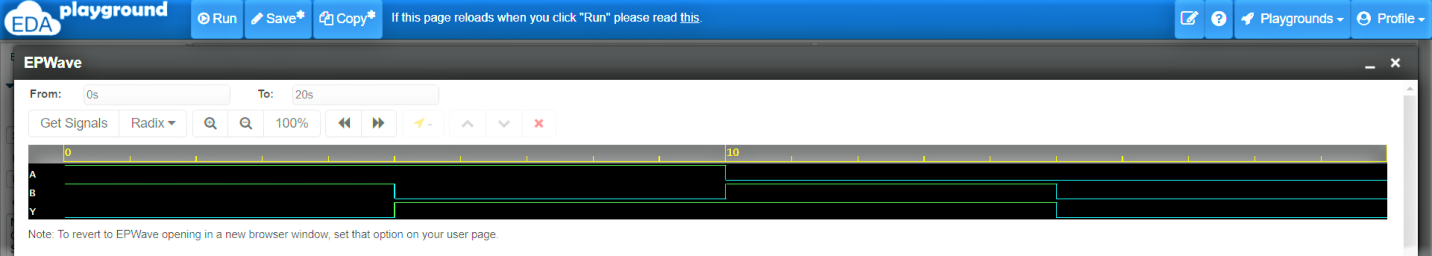
assign y = ~(~(~a&~b));

endmodule



1. XOR





**Testbench.sv**

module tb\_gate;

reg A,B;

wire Y;

//print

//Ayushman Pranav E21CSEU0245

gate1 a1(.a(A),.b(B),.y(Y));

initial

begin

A=1; B=1; #5;

$display("A B Result");

$display("%b %b %b",A,B,Y);

A=1; B=0; #5;

$display("%b %b %b",A,B,Y);

A=0; B=1; #5;

$display("%b %b %b",A,B,Y);

A=0; B=0; #5;

$display("%b %b %b",A,B,Y);

end

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

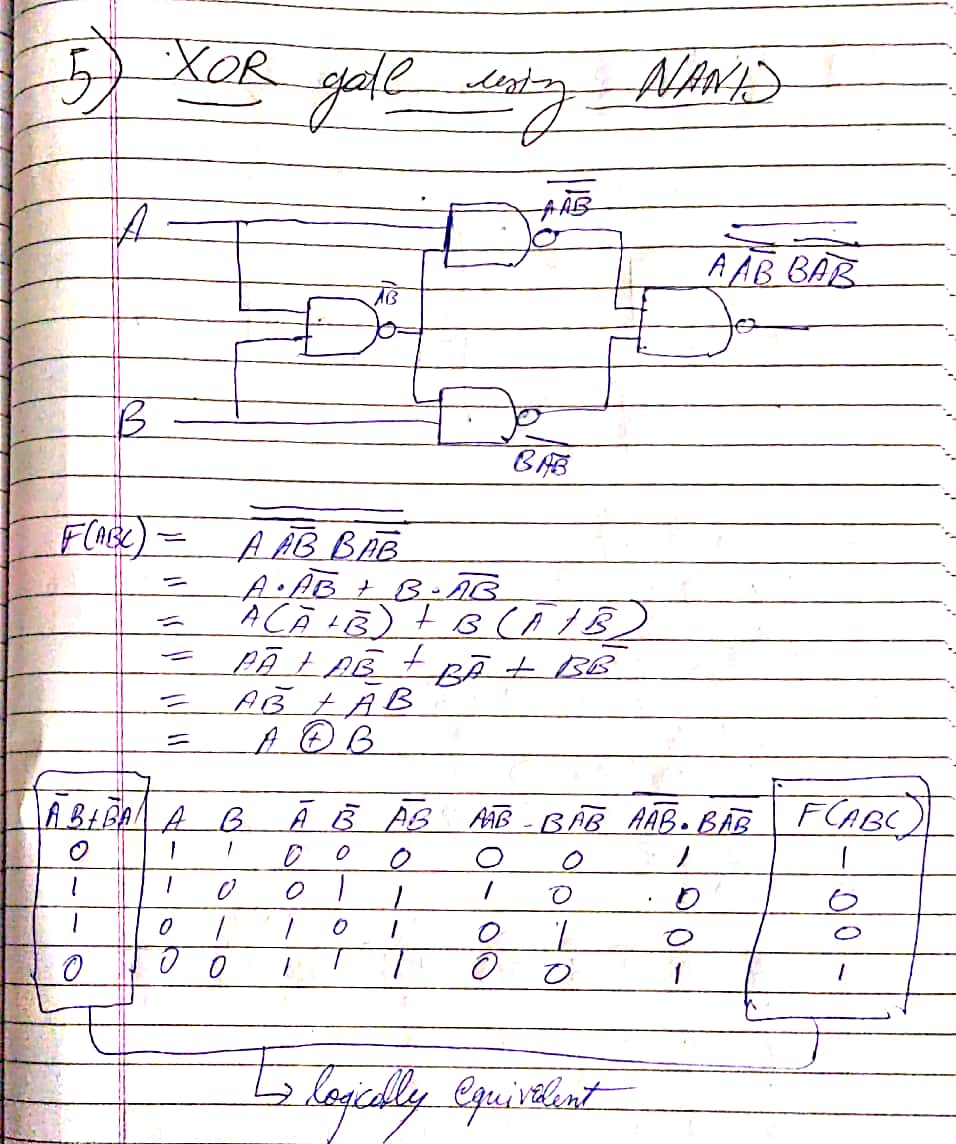
**Design.sv**

module

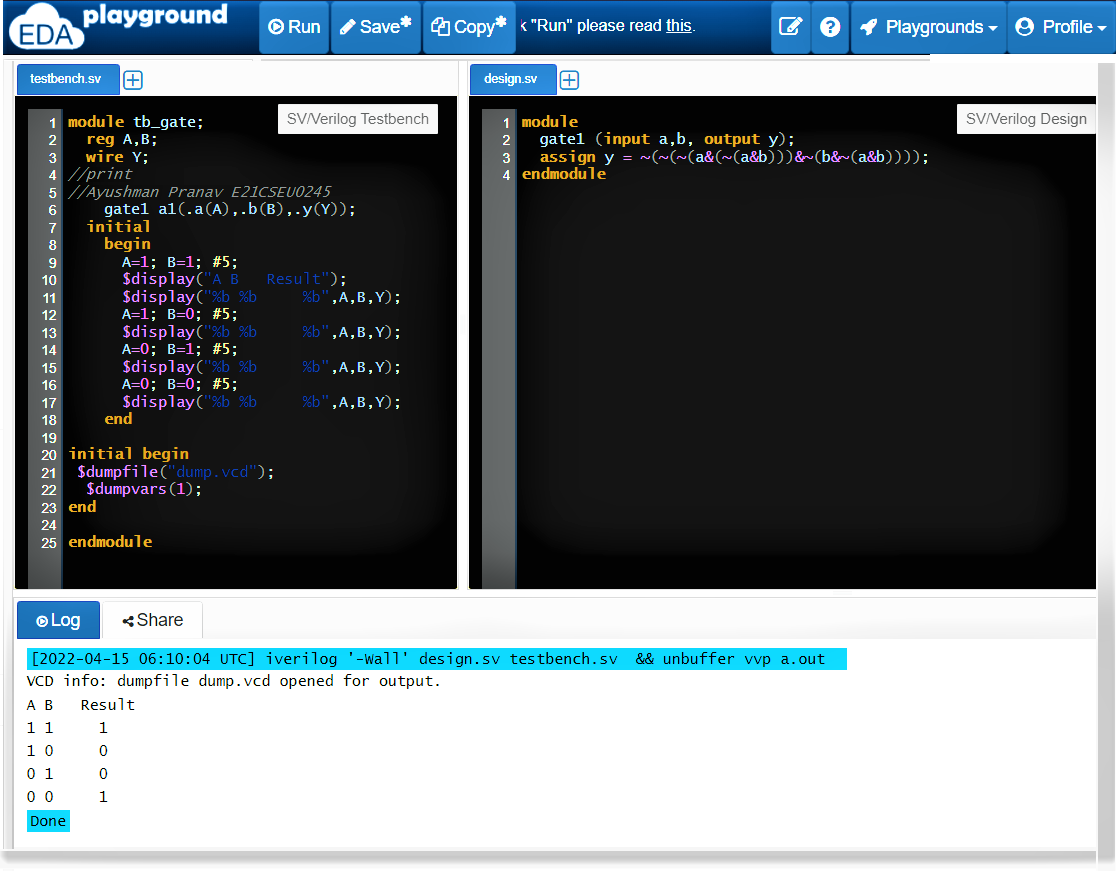
gate1 (input a,b, output y);

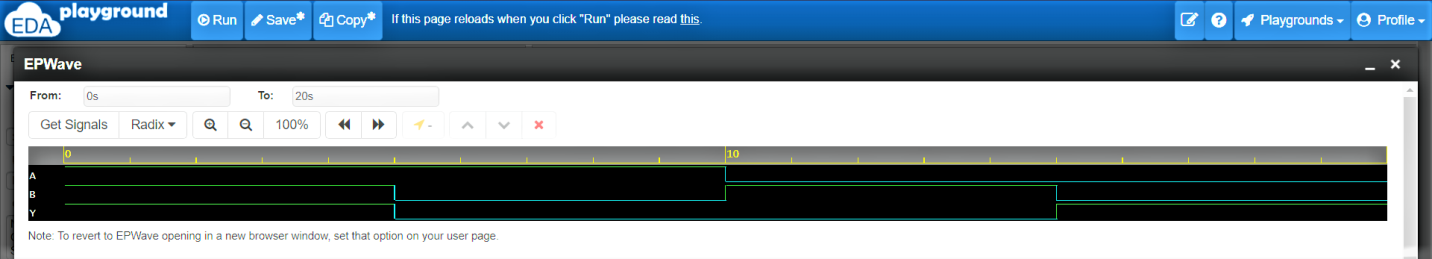
assign y = ~(~(a&(~(a&b)))&~(b&~(a&b)));

endmodule



1. XNOR





**Testbench.sv**

module tb\_gate;

reg A,B;

wire Y;

//print

//Ayushman Pranav E21CSEU0245

gate1 a1(.a(A),.b(B),.y(Y));

initial

begin

A=1; B=1; #5;

$display("A B Result");

$display("%b %b %b",A,B,Y);

A=1; B=0; #5;

$display("%b %b %b",A,B,Y);

A=0; B=1; #5;

$display("%b %b %b",A,B,Y);

A=0; B=0; #5;

$display("%b %b %b",A,B,Y);

end

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

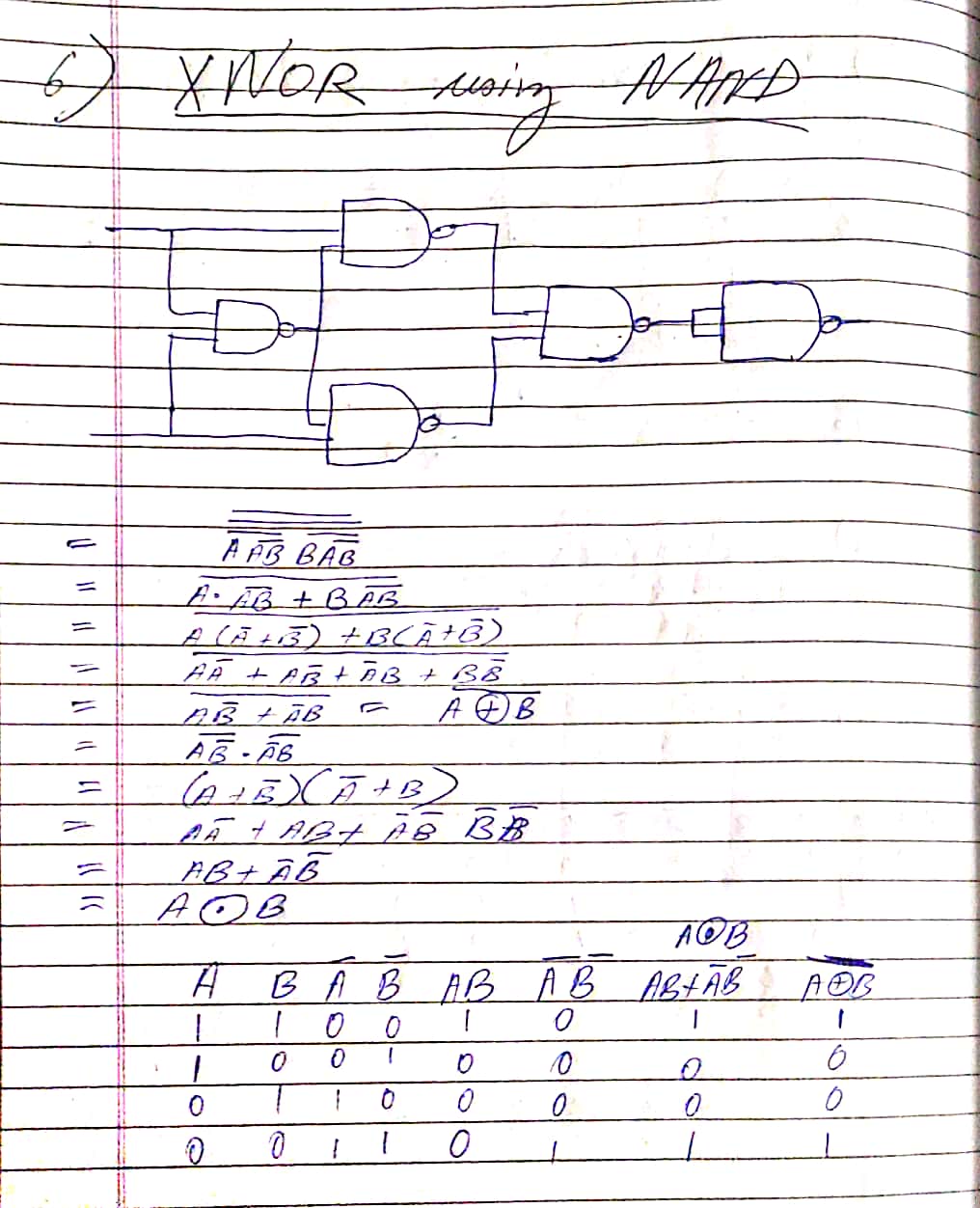
**Design.sv**

module

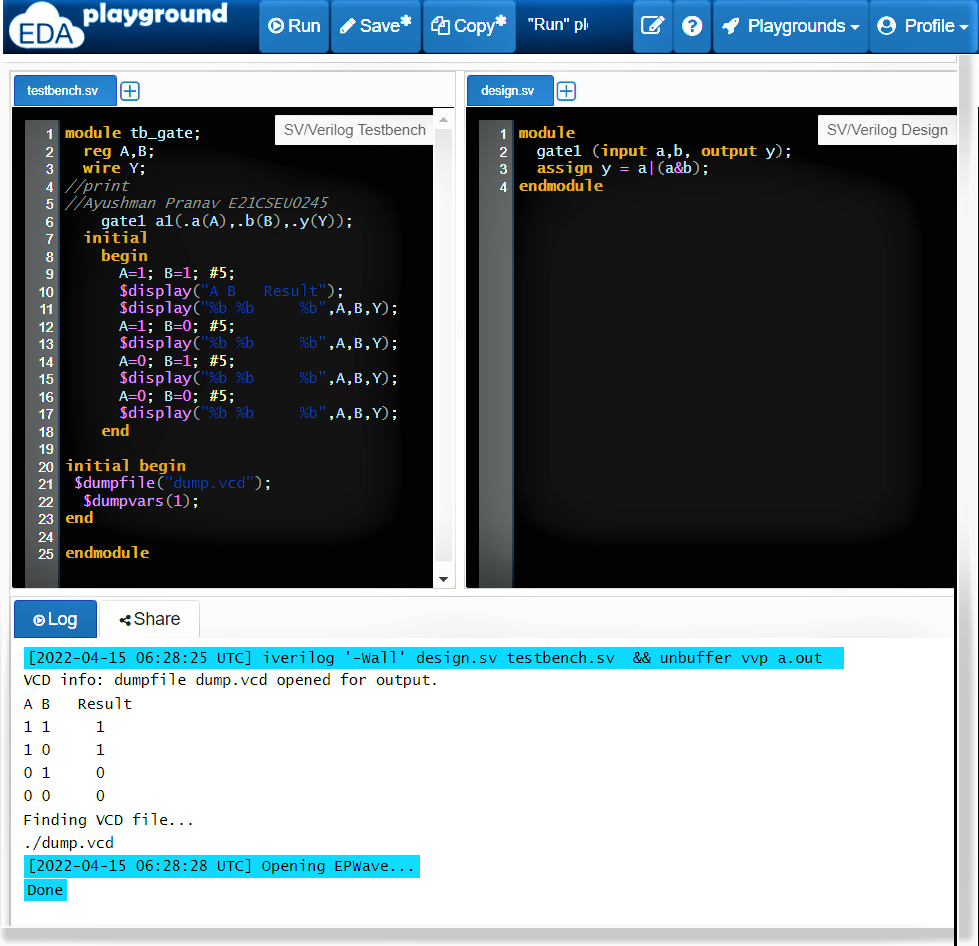
gate1 (input a,b, output y);

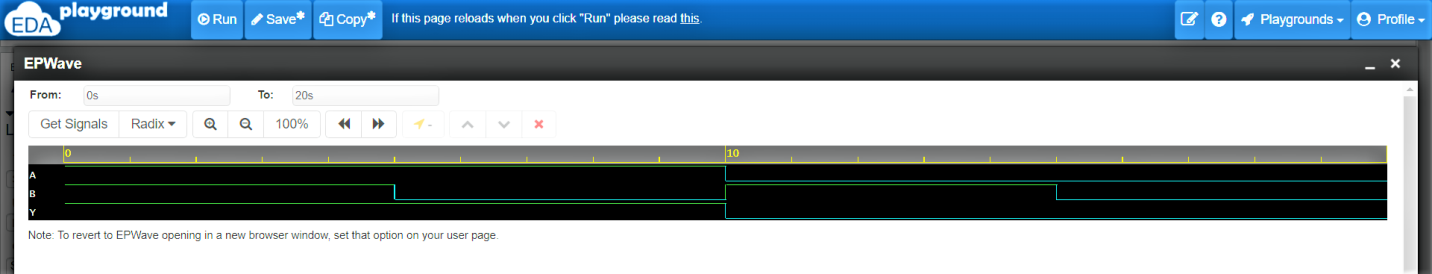
assign y = ~(~(a&(~(a&b)))&~(b&~(a&b)));

endmodule



1. Write a Verilog code to verify Absorption Law and then test using wave form and compare with truth table whether your circuit produced same output or not?





**Testbench.sv**

module tb\_gate;

reg A,B;

wire Y;

//print

//Ayushman Pranav E21CSEU0245

gate1 a1(.a(A),.b(B),.y(Y));

initial

begin

A=1; B=1; #5;

$display("A B Result");

$display("%b %b %b",A,B,Y);

A=1; B=0; #5;

$display("%b %b %b",A,B,Y);

A=0; B=1; #5;

$display("%b %b %b",A,B,Y);

A=0; B=0; #5;

$display("%b %b %b",A,B,Y);

end

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

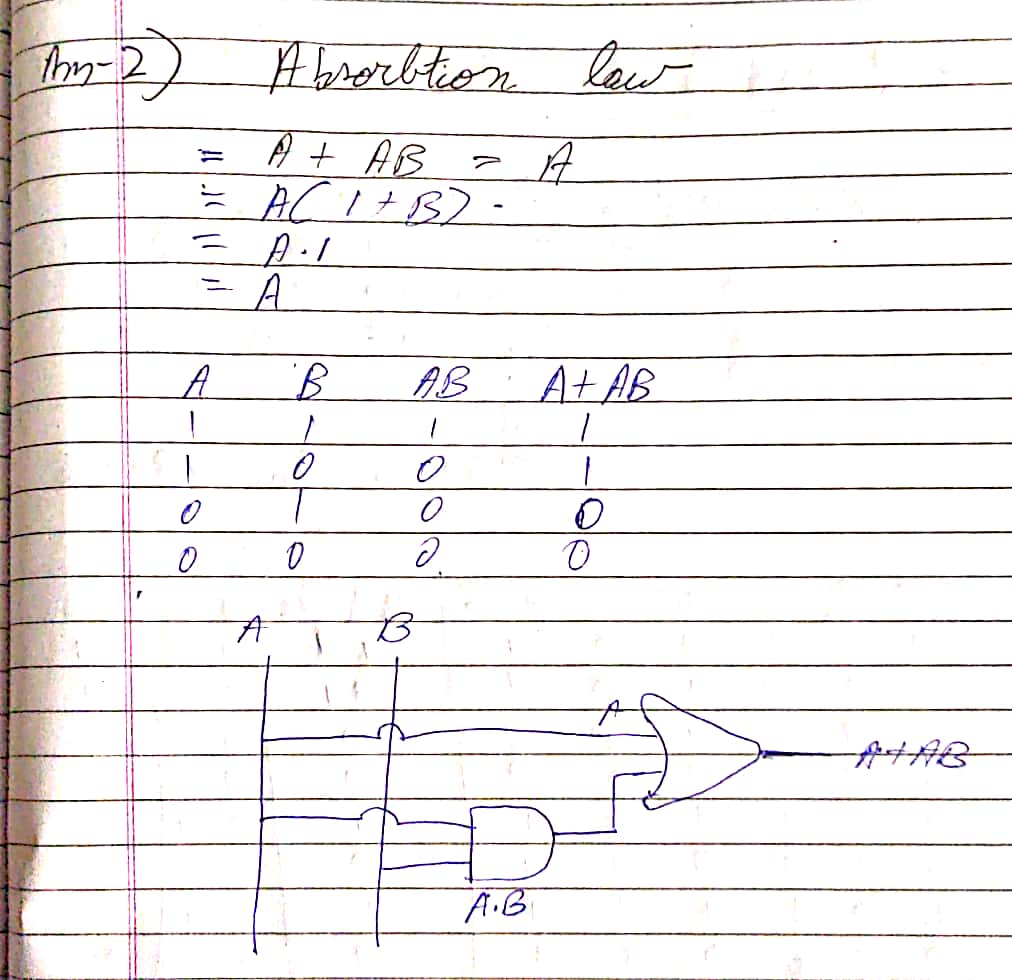
**Design.sv**

module

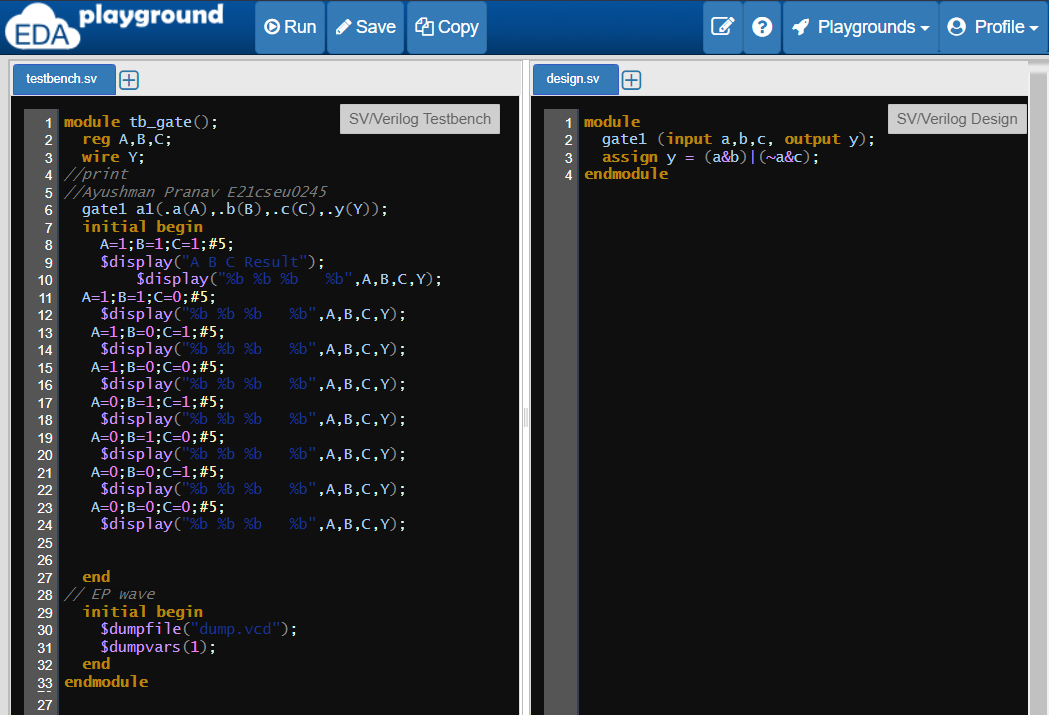
gate1 (input a,b, output y);

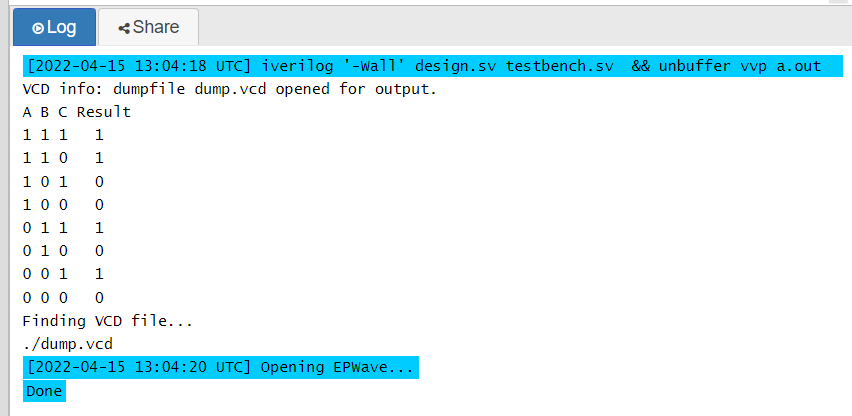
assign y = a|(a&b);

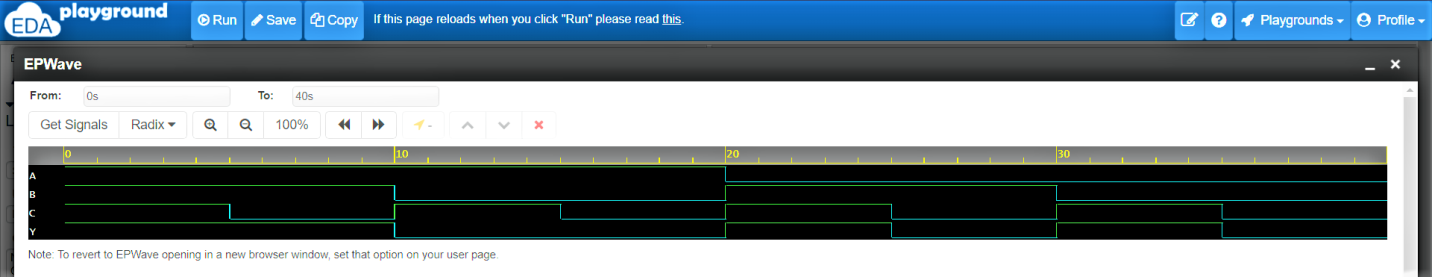
endmodule



1. Write a Verilog code to verify Transposition Law and then test using wave form and compare with truth table whether your circuit produced same output or not?







**Testbench.sv**

module tb\_gate();

reg A,B,C;

wire Y;

//print

//Ayushman Pranav E21cseu0245

gate1 a1(.a(A),.b(B),.c(C),.y(Y));

initial begin

A=1;B=1;C=1;#5;

$display("A B C Result");

$display("%b %b %b %b",A,B,C,Y);

A=1;B=1;C=0;#5;

$display("%b %b %b %b",A,B,C,Y);

A=1;B=0;C=1;#5;

$display("%b %b %b %b",A,B,C,Y);

A=1;B=0;C=0;#5;

$display("%b %b %b %b",A,B,C,Y);

A=0;B=1;C=1;#5;

$display("%b %b %b %b",A,B,C,Y);

A=0;B=1;C=0;#5;

$display("%b %b %b %b",A,B,C,Y);

A=0;B=0;C=1;#5;

$display("%b %b %b %b",A,B,C,Y);

A=0;B=0;C=0;#5;

$display("%b %b %b %b",A,B,C,Y);

end

// EP wave

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

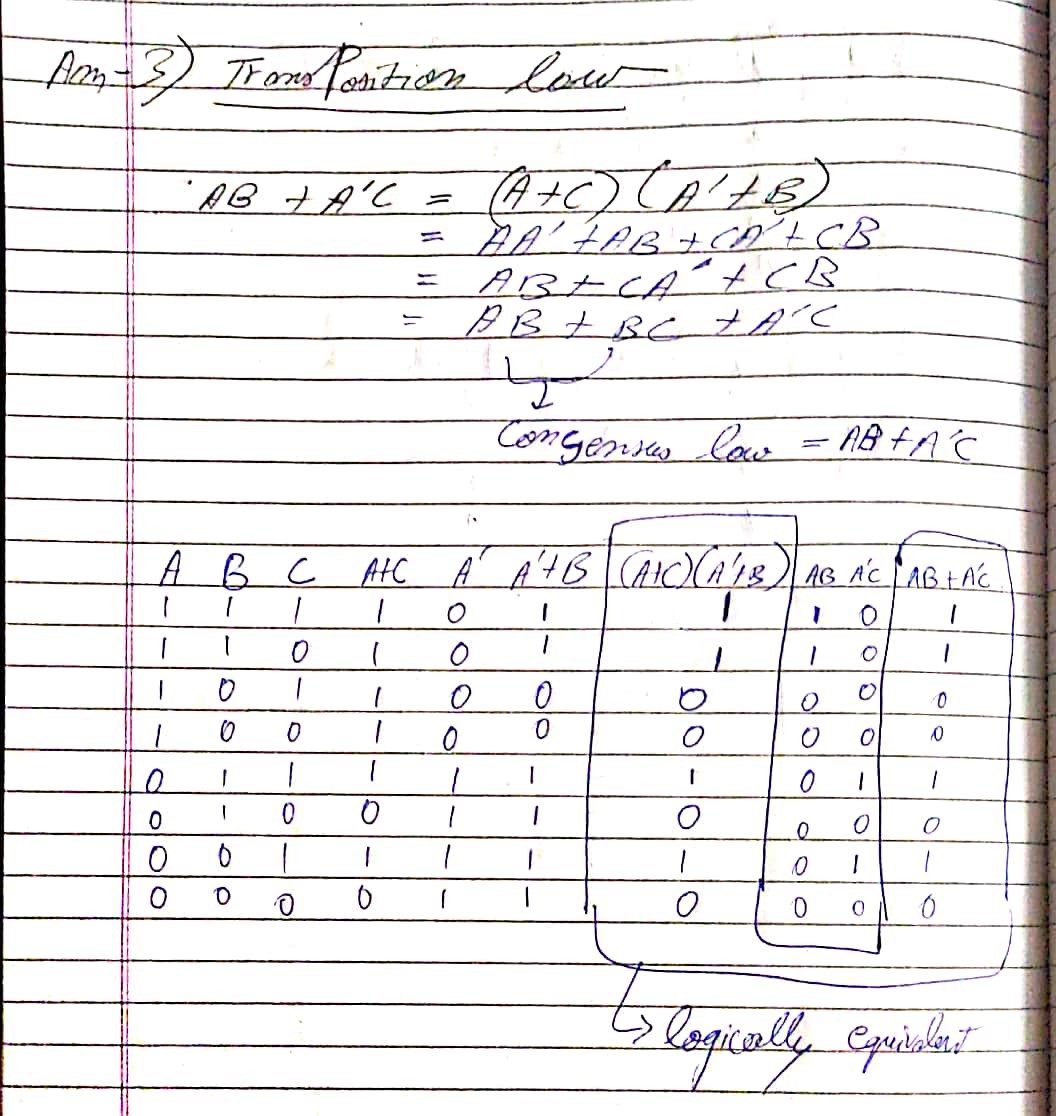
**Design.sv**

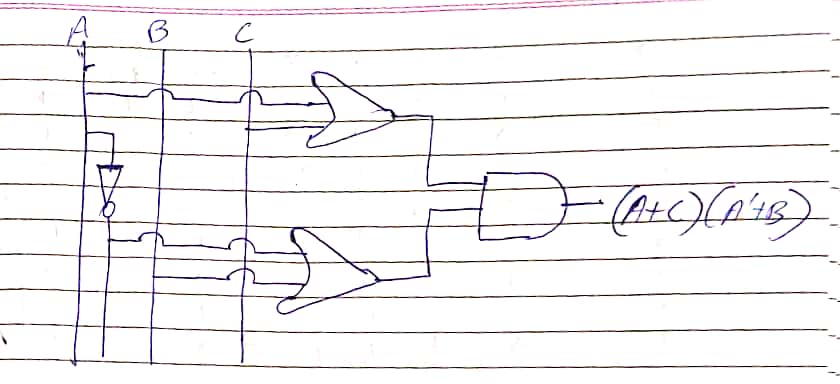
module

gate1 (input a,b,c, output y);

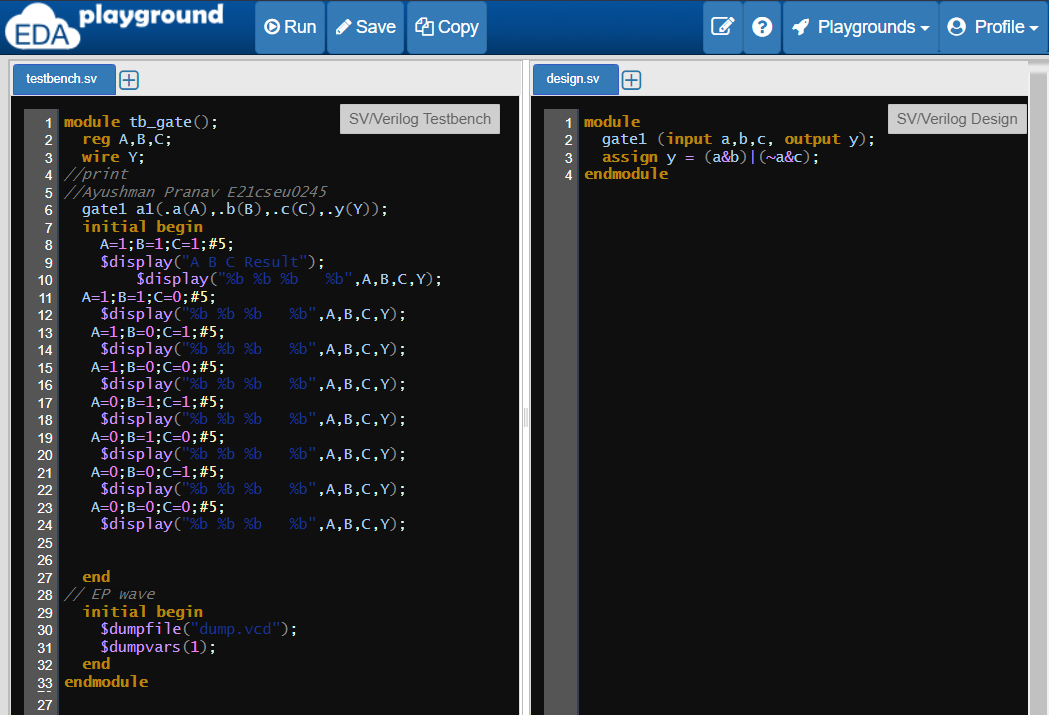
assign y = (a&b)|(~a&c);

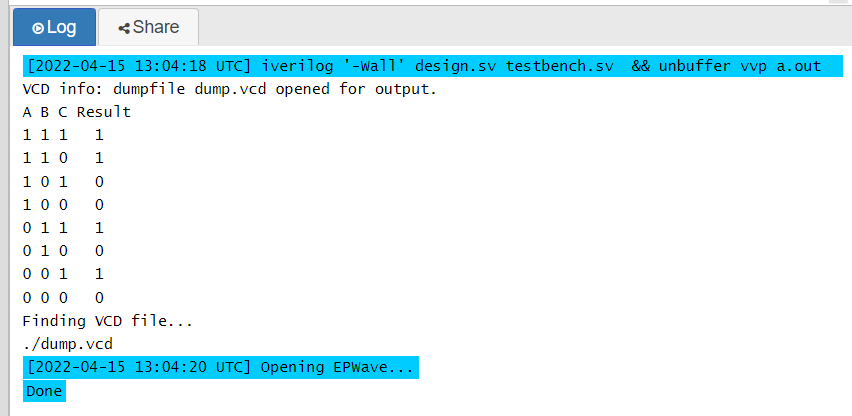
endmodule

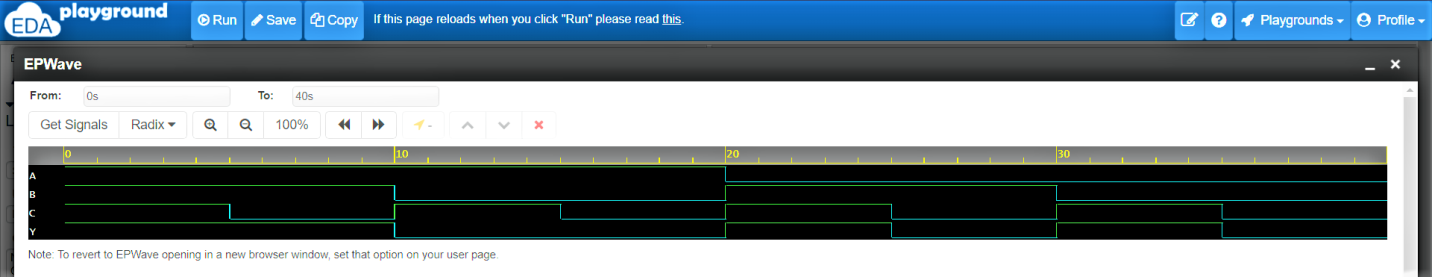




1. Write a Verilog code to verify Consensus Law and then test using wave form and compare with truth table whether your circuit produced same output or not?







**Testbench.sv**

module tb\_gate();

reg A,B,C;

wire Y;

//print

//Ayushman Pranav E21cseu0245

gate1 a1(.a(A),.b(B),.c(C),.y(Y));

initial begin

A=1;B=1;C=1;#5;

$display("A B C Result");

$display("%b %b %b %b",A,B,C,Y);

A=1;B=1;C=0;#5;

$display("%b %b %b %b",A,B,C,Y);

A=1;B=0;C=1;#5;

$display("%b %b %b %b",A,B,C,Y);

A=1;B=0;C=0;#5;

$display("%b %b %b %b",A,B,C,Y);

A=0;B=1;C=1;#5;

$display("%b %b %b %b",A,B,C,Y);

A=0;B=1;C=0;#5;

$display("%b %b %b %b",A,B,C,Y);

A=0;B=0;C=1;#5;

$display("%b %b %b %b",A,B,C,Y);

A=0;B=0;C=0;#5;

$display("%b %b %b %b",A,B,C,Y);

end

// EP wave

initial begin

$dumpfile("dump.vcd");

$dumpvars(1);

end

endmodule

**Design.sv**

module

gate1 (input a,b,c, output y);

assign y = (a&b)|(~a&c);

endmodule

