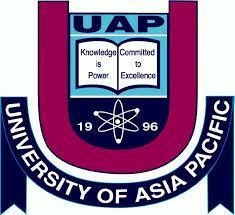
UNIVERSITY OF ASIA PACIFIC

Department of Computer Science of Engineering



**Lab Final Exam:**

**Course code: CSE 458**

**Course Title: Design and Testing of VLSI Lab**

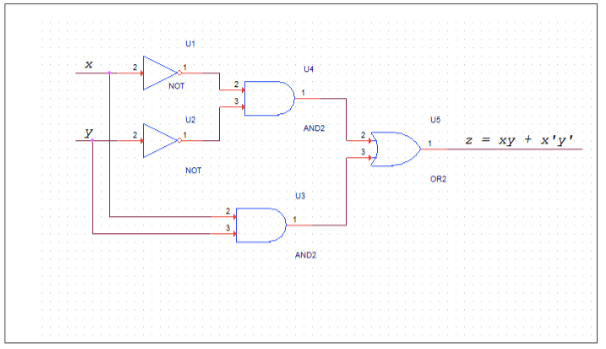
**Submitted By:**                                                         **Submitted To:**

Rumman Ahmed Prodhan                   Jahir Ibna Rafiq

ID: 18101018                                                                          Assistant Professor

Sec: A1

**1.Write Verilog code and testbench for the following circuit**



**Verilog code:**

// Verilog code: z = xy+x'y'

module task1( // here i'm giving the module name as "task1". i can give any name here.

input x, // taking the input parameter for x

input y, // taking the input parameter for y

output z // this is the output parameter for z

);

assign z = (x&y) | (~x & ~y); // assigning the given equation (z = xy+x'y') here. for "and" as "&", "+" as "|", "not" as "~"

endmodule // ending the module

**Testbench:**

//Testbench for the following circuit: z = xy+x'y'

module Testtask1; // here i'm declaring the name of the module

reg x; // taking reg to store input variable x

reg y; // taking reg to store input variable y

wire z; // for output “wire” is used. So for output 'z' wire is used

task1 uut( // here the name should be same as the name in the Verilog module name. Also, UUT means Unit Under Test

.x(x), // this is for input x

.y(y), // this is for input x

.z(z) // this is for output z

);

//here I will initialize the following the test cases

//if x = 0 and y = 0 then the expected output would be z = 1

//if x = 0 and y = 1 then the expected output would be z = 0

//if x = 1 and y = 0 then the expected output would be z = 0

//if x = 1 and y = 1 then the expected output would be z = 1

initial

begin // initial test case

x = 0 ; y =0 ; //this would be tested first

#18 x = 0 ; y =1 ; //i've given delay as 18 to match my id. Any number of delay can be given here.

#18 x = 1 ; y =0 ; //i've given delay as 18 to match my id. Any number of delay can be given here.

#18 x = 1 ; y =1 ; //i've given delay as 18 to match my id. Any number of delay can be given here.

end // this will end the test case initialize part

// this section will show the output

initial

begin //initializing this section

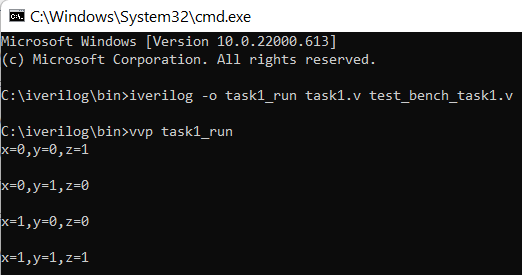
//to show the output, i'm using the monitor function

$monitor("x=%d,y=%d,z=%d \n",x,y,z); //here i'm showing the x,y,z in the monitor function

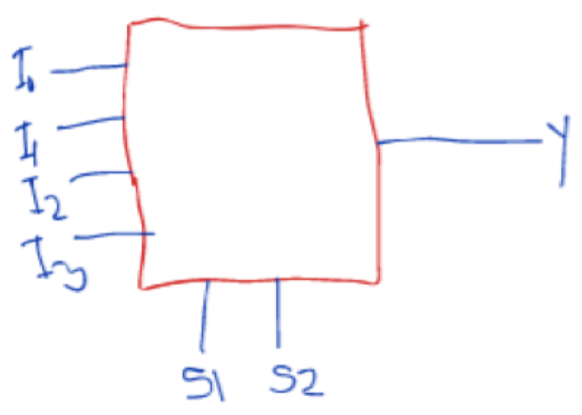
end //end of the initialization of output part

endmodule // ending the module Testtask1

**Output Screenshot is given below:**



**2. Write a Verilog code and testbench for the following 4:1 Mux**



**Verilog code:**

//Write a Verilog code for the 4:1 Mux

//This part will define the module

module task2 ( //here i'm giving the module name as "task2". i can give any name here.

//Here i'm taking 4 inputs as I0,I1,I2,I3, 2 switch as S1,S2 and the output as Y as mentioned in the above image

input I0,

input I1,

input I2,

input I3,

input S1,S2,

output Y

);

// i'm declaring the assign variable to check the following conditions

// "?" operator checks if the value is high then it will select the first condition, else the second condition

// if S2 =1, then (S1 ? I3 : I2)

// if S1 =1, then I3, if S1 =0, then I2

// else

// if S2 = 0, then (S1 ? I1 : I0)

// if S1 =1, then I2, if S1 =0, then I0

assign Y = S2 ? (S1 ? I3 : I2) : (S1 ? I1 : I0); //this is where i assigned the above conditioned into Y

endmodule // end the module task2

**Testbench:**

// Write testbench for the 4:1 Mux

module Testtask2;

// declared variable(I0,I1,I2,I3,S1,S2) and output variable (out)

wire Y; // for output “wire” is used. So for output 'Y' wire is used

reg I0; //taking reg to store input variable I0

reg I1; //taking reg to store input variable I1

reg I2; //taking reg to store input variable I2

reg I3; //taking reg to store input variable I3

reg S1; //taking reg to store switch variable S1

reg S2; //taking reg to store switch variable S1

task2 uut( // here the name should be same as the name in the Verilog module name "task2". Also, UUT means Unit Under Test

.Y(Y), //// this is for output variable Y

.I0(I0), // this is for input variable I0

.I1(I1), // this is for input variable I1

.I2(I2), // this is for input variable I2

.I3(I3), // this is for input variable I3

.S1(S1), // this is for switch variable S1

.S2(S2) // this is for switch variable S2

);

// if S2=0 and S1=0, then Y=I0

// if S2=0 and S1=1, then Y=I1

// if S2=1 and S1=0, then Y=I2

// if S2=1 and S1=1, then Y=I3

initial

begin // initial test case

//i've added all possible 16 combinations for the 4 input

//i've given delay as 18 to match my id. Any number of delay

S2=0; S1 = 0;

I0=0; I1=0; I2=0; I3=0;

#18 I0=0; I1=0; I2=0; I3=1;

#18 I0=0; I1=0; I2=1; I3=0;

#18 I0=0; I1=0; I2=1; I3=1;

#18 I0=0; I1=1; I2=0; I3=0;

#18 I0=0; I1=1; I2=0; I3=1;

#18 I0=0; I1=1; I2=1; I3=0;

#18 I0=0; I1=1; I2=1; I3=1;

#18 I0=1; I1=0; I2=0; I3=0;

#18 I0=1; I1=0; I2=0; I3=1;

#18 I0=1; I1=0; I2=1; I3=0;

#18 I0=1; I1=0; I2=1; I3=1;

#18 I0=1; I1=1; I2=0; I3=0;

#18 I0=1; I1=1; I2=0; I3=1;

#18 I0=1; I1=1; I2=1; I3=0;

#18 I0=1; I1=1; I2=1; I3=1;

//i've added all possible 16 combinations for the 4 input

#18 S2=0; S1 =1;

#18 I0=0; I1=0; I2=0; I3=0;

#18 I0=0; I1=0; I2=0; I3=1;

#18 I0=0; I1=0; I2=1; I3=0;

#18 I0=0; I1=0; I2=1; I3=1;

#18 I0=0; I1=1; I2=0; I3=0;

#18 I0=0; I1=1; I2=0; I3=1;

#18 I0=0; I1=1; I2=1; I3=0;

#18 I0=0; I1=1; I2=1; I3=1;

#18 I0=1; I1=0; I2=0; I3=0;

#18 I0=1; I1=0; I2=0; I3=1;

#18 I0=1; I1=0; I2=1; I3=0;

#18 I0=1; I1=0; I2=1; I3=1;

#18 I0=1; I1=1; I2=0; I3=0;

#18 I0=1; I1=1; I2=0; I3=1;

#18 I0=1; I1=1; I2=1; I3=0;

#18 I0=1; I1=1; I2=1; I3=1;

//i've added all possible 16 combinations for the 4 input

#18 S2=1; S1 = 0;

#18 I0=0; I1=0; I2=0; I3=0;

#18 I0=0; I1=0; I2=0; I3=1;

#18 I0=0; I1=0; I2=1; I3=0;

#18 I0=0; I1=0; I2=1; I3=1;

#18 I0=0; I1=1; I2=0; I3=0;

#18 I0=0; I1=1; I2=0; I3=1;

#18 I0=0; I1=1; I2=1; I3=0;

#18 I0=0; I1=1; I2=1; I3=1;

#18 I0=1; I1=0; I2=0; I3=0;

#18 I0=1; I1=0; I2=0; I3=1;

#18 I0=1; I1=0; I2=1; I3=0;

#18 I0=1; I1=0; I2=1; I3=1;

#18 I0=1; I1=1; I2=0; I3=0;

#18 I0=1; I1=1; I2=0; I3=1;

#18 I0=1; I1=1; I2=1; I3=0;

#18 I0=1; I1=1; I2=1; I3=1;

//i've added all possible 16 combinations for the 4 input

#18 S2=1; S1 =1;

#18 I0=0; I1=0; I2=0; I3=0;

#18 I0=0; I1=0; I2=0; I3=1;

#18 I0=0; I1=0; I2=1; I3=0;

#18 I0=0; I1=0; I2=1; I3=1;

#18 I0=0; I1=1; I2=0; I3=0;

#18 I0=0; I1=1; I2=0; I3=1;

#18 I0=0; I1=1; I2=1; I3=0;

#18 I0=0; I1=1; I2=1; I3=1;

#18 I0=1; I1=0; I2=0; I3=0;

#18 I0=1; I1=0; I2=0; I3=1;

#18 I0=1; I1=0; I2=1; I3=0;

#18 I0=1; I1=0; I2=1; I3=1;

#18 I0=1; I1=1; I2=0; I3=0;

#18 I0=1; I1=1; I2=0; I3=1;

#18 I0=1; I1=1; I2=1; I3=0;

#18 I0=1; I1=1; I2=1; I3=1;

end // this will end the test case initialize part

// this section will show the output

initial begin //initializing this section

//to show the output, i'm using the monitor function

$monitor("S2=%d,S1=%d,I0=%d,I1=%d=I2=%d,I3=%d,Y=%d\n",S2,S1,I0,I1,I2,I3,Y); //here i'm showing the S2,S1,I0,I1,I2,I3,Y in the monitor function

end //end of the initialization of output part

endmodule // ending the module Testtask2

**Output Screenshot is given below:**

