Name: Aswad jamal

EE-272L Digital Systems Design

Reg. No.: 2023-EE-84

Marks Obtained: _____

Lab Manual

DSD Lab Manual Evaluation Rubrics

Assessment	Total Marks	Marks Obtained	0-30%	30-60%	70-100%
Code Organization (CLO1)	3		No Proper Indentation and descriptive naming, no code organization. Zero to Some understanding but not working	Proper Indentation or descriptive naming or code organization. Mild to Complete understanding but not working	Proper Indentation and descriptive naming, code organization. Complete understanding, and proper working
Simulation (CLO2)	5		Simulation not done or incorrect, without any understanding of waveforms	Working simulation with errors, don't cares's(x) and high impedance(z), partial understanding of waveforms	Working simulation without any errors, etc and complete understanding of waveforms
FPGA (CLO2)	2		Not implemented on FPGA and questions related to synthesis and implementation not answered.	Correctly Implemented on FPGA or questions related to synthesis and implementation answered.	Correctly Implemented on FPGA and questions related to synthesis and implementation answered.

Question no 1

(a)

Truth table of circuit:

p	q	r	m	n
0	0	0	1	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	0
1	1	0	0	1
1	1	1	1	1

(b)

Error in codes:

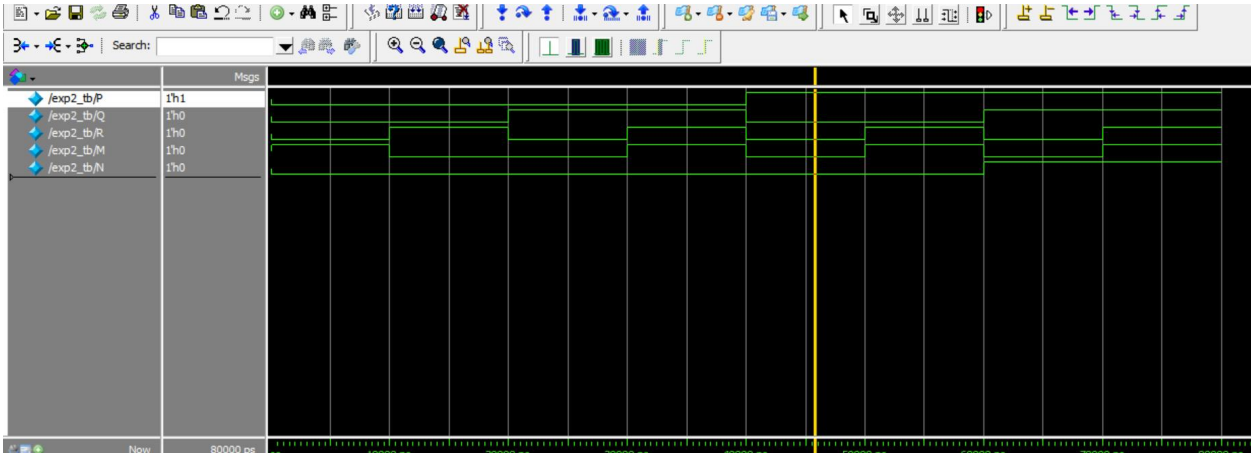
1. RTL code:

Assign is not used with sum and in the carry equation (c(a ^ b)); looks like that c is a function so & should be used as (c &(a ^ b).

2. Test bench code:

- carry1 is not declared.
- Instant name is not declared as in my code full_adder as a1.
- Module uses signals a1, b1, c1 but in many signals a, b, c were used instead of a1, b1, c1.

Waveform:



Corrected codes

RTL Code:

```
module full_adder(

input logic a,

input logic b,

input logic c,

output logic sum,

output logic carry

);

assign sum = (a ^ b) ^ c;

assign carry= (a&b)| (c &(a ^ b));

endmodule
```

Test bench code:

```
module tb_3();

logic a1;

logic b1;

logic c1;

logic sum1;

logic carry1;

full_adder  a1(

   .a(a1),

   .b(b1),

   .c(c1),

   .sum(sum1),

   .carry(carry1)

   );

initial

begin

      a1=0 ; b1=0 ; c1=0 ;

      #10;

      a1=0 ; b1=0 ; c1=1 ;

      #10;

      a1=0 ; b1=1 ; c1=0 ;

      #10;
```

```
    a1=0 ; b1=1 ; c1=1 ;
    #10;
    a1=1 ; b1=0 ; c1=0 ;
    #10;
    a1=1 ; b1=0 ; c1=1 ;
    #10;
    a1=1 ; b1=1 ; c1=0 ;
    #10;
    a1=1 ; b1=1 ; c1=1 ;
    #10;
    $stop;
end
initial
begin
    $monitor("sum=%b, carry=%b, a=%b, b=%b, c=%b" , sum1,carry1,a1,b1,c1);
End
Endmodule
```

Testbench code of circuit

```
module exp2_tb();
  logic p, q, r;
  logic m, n;
```

```
  exp2 tb(
    .a(p),
    .b(q),
    .c(r),
    .x(m),
    .y(n)
  );
initial
    begin
    p = 0; q = 0; r = 0; #10;
    p = 0; q = 0; r = 1; #10;
    p = 0; q = 1; r = 0; #10;
    p = 0; q = 1; r = 1; #10;
    p = 1; q = 0; r = 0; #10;
    p = 1; q = 0; r = 1; #10;
    p = 1; q = 1; r = 0; #10;
    p = 1; q = 1; r = 1; #10;
    $stop;
  end
  initial
  begin
    $monitor("p = %b q = %b r = %b | m = %b n = %b", p, q, r, m, n);
```

```
  end

endmodule
```