California State University, Northridge

Department of Electrical & Computer Engineering

ECE 526

Homework 5

By

Avinash Damse

CSUN ID- 203131064

1.Write a SystemVerilog function called factorial that takes a non-negative integer input argument n, and returns the factorial of that number. For example, if the input argument is 5, the output should be 120 (i.e., 5 x 4 x 3 x 2 x 1).

Ans –

```
function automatic int factorial(int n);
  int result = 1;
  for (int i = 1; i <= n; i=i+1) begin
    result = result * i;
  end
  return result;
endfunction
```

We can this use function like this :

```
int my_factorial = factorial(5);  // my_factorial will be set to 120
```

2.Write a SystemVerilog module called adder that uses the generate construct to create four instances of a 2-bit full-adder module. The adder module takes two 8-bit input arguments a and b, and outputs their sum. The module should use the generate construct to create four instances of the full-adder module to implement a 8-bit adder.

Ans –

```
module full_adder( a, b, carry_in, sum, carry_out);

        input logic a;
        input logic b;
        input logic carry_in;
        output logic sum;
        output logic carry_out;

   assign sum = a ^ b ^ carry_in;
   assign carry_out = (a & b) | (a & carry_in) | (b & carry_in);

endmodule
```

```
module adder( a, b, sum);

          input logic [7:0] a;
          input logic [7:0] b;
          output logic [7:0] sum;

    genvar i;
    generate
        for (i = 0; i < 8; i++) begin : adder_inst
            full_adder full_adder_inst(.a(a[i]), .b(b[i]), .carry_in(adder_inst[i-1].carry_out),
.sum(sum[i]), .carry_out(adder_inst[i].carry_in));
        end
    endgenerate

endmodule
```

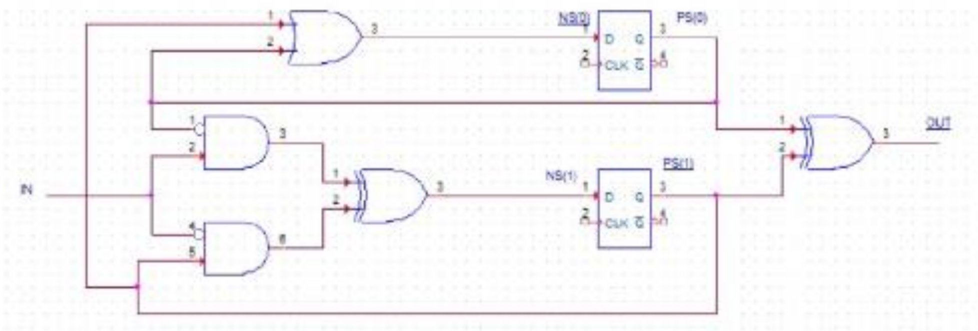
3.What are the outputs for the following block of code?

```
module enum_method;
typedef enum {red, blue, green, yellow, black} color;
color c;
initial begin c = c.first();
$display(" %s ",c.name);
c = c.next(2);
$display(" %s ",c.name);
c = c.last();
$display(" %s ",c.name);
c = c.prev(3);
$display(" %s ",c.name);
c = c.next();
$display(" %s ",c.name);
end
endmodule
```

Ans –

```
 red
 green
 black
 yellow
 black
```

4. Consider the following finite state machine (FSM) circuit

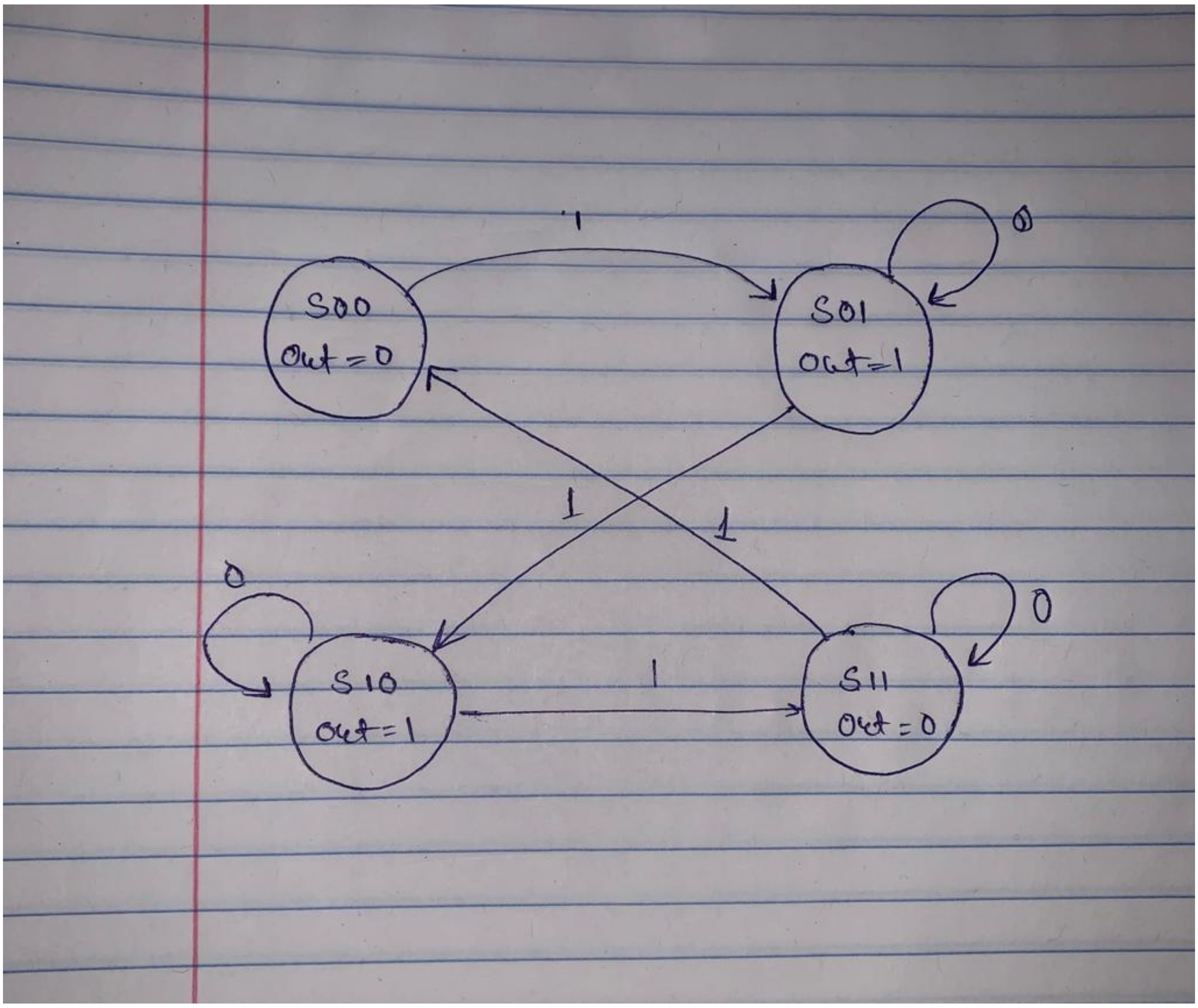


(a) Write a Verilog description of the circuit using continuous assignment for the NS and OUT signals.
(b) Draw a state transition diagram describing the behavior of the circuit. Write out the truth table to see what the transitions are.

Ans-

```
module Circut( Clock Reset , In , Out )
        input Clock , Reset , In;
        output Out;
        reg [ 1 : 0 ] PS , NS;
        always @(posedge Clock) begin
        if ( Reset ) PS <= 2’b00;
        else PS <= NS;
        end
        assign NS [ 0 ] = PS[ 1 ] | PS[ 0 ];
        assign NS[ 1 ] = ( In & ˜PS[ 0 ] ) ^ ( ˜In & PS[ 1 ] );
        assign Out = PS[ 1 ] ^ PS[ 0 ];
endmodule
```

PS[1]	PS[0]	In	NS[1]	NS[0]
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0


1
0
S00
Out=0
S01
Out=1
1
1
0
0
S10
Out=1
1
S11
Out=0