Digital Design and Computer Organisation Laboratory

3rd Semester, Academic Year 2025

Date:01-09-2025

Name: Aakash Desai	SRN:PES1UG24CS006	Section 3A

Week Number: 4 Program Number: 1

TITLE: Write a verilog program to model 2:4 decoder with input enabled. Generate the VVP output and simulation waveform using GTKWave.

Deliverables

I. Verilog Code Screenshot
II. Verilog VVP Output Screen Shot
III. GTKWAVE Screenshot
IV. Output Table to be completed and included

I.Verilog Code Snippet

- Main code

```
module dec2to4 (
  input wire [1:0] a,
  input wire en,
  output wire [3:0] y
);

  assign y[0] = en & ~a[1] & ~a[0];
  assign y[1] = en & ~a[1] &  a[0];
  assign y[2] = en &  a[1] & ~a[0];
  assign y[3] = en &  a[1] &  a[0];

endmodule
```

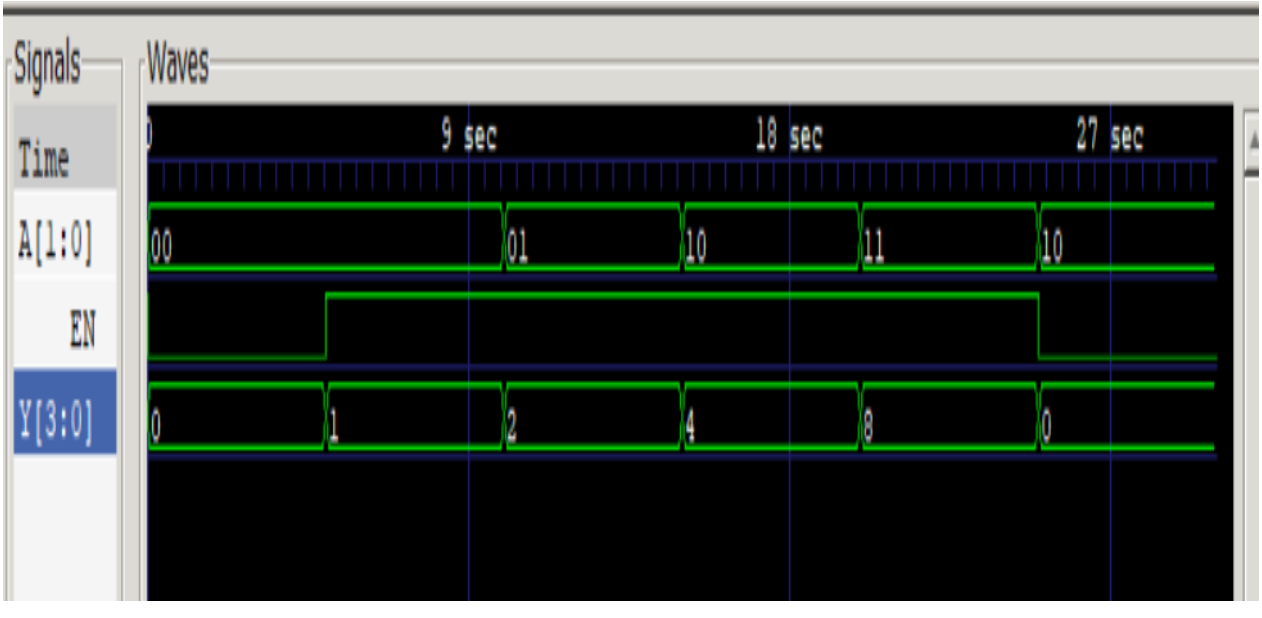
- Test Bench

```
module TB;
  reg [1:0] A;
  reg EN;
  wire [3:0] Y;


  dec2to4 newDEC (
    .a(A),
    .en(EN),
    .y(Y)
  );

  initial begin
    $dumpfile("DEC2_test.vcd");
    $dumpvars(0, TB);
    $monitor("Time=%0t | EN=%b | A=%b | Y=%b", $time, EN, A, Y);


    EN = 1'b0; A = 2'b00; #5;
    EN = 1'b1; A = 2'b00; #5;
    EN = 1'b1; A = 2'b01; #5;
    EN = 1'b1; A = 2'b10; #5;
    EN = 1'b1; A = 2'b11; #5;
    EN = 1'b0; A = 2'b10; #5;

    $finish;
  end
endmodule
```

II Verilog VVP output ScreenShot

```
C:\iverilog\bin\Week4>vvp dsn
VCD info: dumpfile DEC2_test.vcd opened for output.
Time=0 | EN=0 | A=00 | Y=0000
Time=5 | EN=1 | A=00 | Y=0001
Time=10 | EN=1 | A=01 | Y=0010
Time=15 | EN=1 | A=10 | Y=0100
Time=20 | EN=1 | A=11 | Y=1000
Time=25 | EN=0 | A=10 | Y=0000
```

III GTKWave ScreenShot



IV Truth Table

E	A	B	D_0	D_1	D_2	D_3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

TITLE: Write a verilog program to model 2:1 multiplexer. Generate the VVP output and simulation waveform using GTKWave.

Deliverables

I. Verilog Code Screenshot
II. Verilog VVP Output Screen Shot
III. GTKWAVE Screenshot
IV. Output Table to be completed and included

I.Verilog Code Snippet

- Main Code

```
module mux2(
    input wire i0,
    input wire i1,
    input wire j,
    output wire o
);
    assign o = (j == 0) ? i0 : i1;
endmodule
```

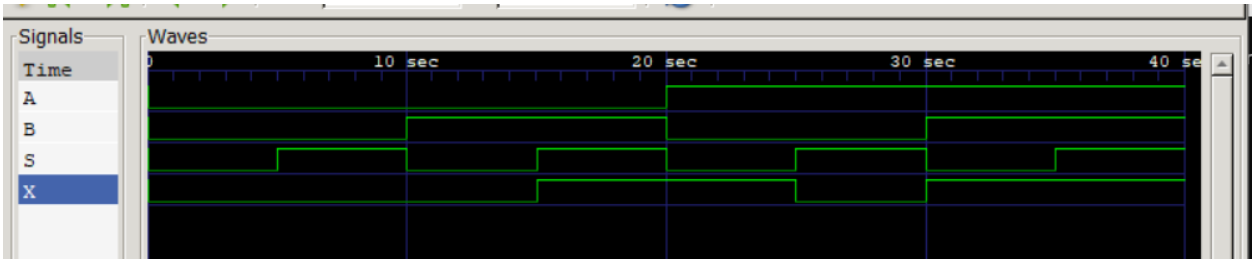
- TestBench

```
module TB;
    reg A, B, S;
    wire X;


    mux2 newMUX(
        .i0(A),
        .i1(B),
        .j(S),
        .o(X)
    );

    initial begin
        $dumpfile("MUX2_test.vcd");
        $dumpvars(0, TB);
        $monitor("Time=%0t | S=%b | A=%b | B=%b | X=%b", $time, S, A, B, X);

        // Test all combinations
        A = 1'b0; B = 1'b0; S = 1'b0; #5;
        A = 1'b0; B = 1'b0; S = 1'b1; #5;
        A = 1'b0; B = 1'b1; S = 1'b0; #5;
        A = 1'b0; B = 1'b1; S = 1'b1; #5;
        A = 1'b1; B = 1'b0; S = 1'b0; #5;
        A = 1'b1; B = 1'b0; S = 1'b1; #5;
        A = 1'b1; B = 1'b1; S = 1'b0; #5;
        A = 1'b1; B = 1'b1; S = 1'b1; #5;

        $finish;
    end
endmodule
```

II Verilog VVP ScreenShot

```
C:\iverilog\bin\Week4>vvp dsn
VCD info: dumpfile MUX2_test.vcd opened for output.
Time=0 | S=0 | A=0 | B=0 | X=0
Time=5 | S=1 | A=0 | B=0 | X=0
Time=10 | S=0 | A=0 | B=1 | X=0
Time=15 | S=1 | A=0 | B=1 | X=1
Time=20 | S=0 | A=1 | B=0 | X=1
Time=25 | S=1 | A=1 | B=0 | X=0
Time=30 | S=0 | A=1 | B=1 | X=1
Time=35 | S=1 | A=1 | B=1 | X=1
```

III GTK wave Screenshot



IV Truth Table

i_0	i_1	j	y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

TITLE: Write a verilog program to model 4:1 multiplexer. Generate the VVP output and simulation waveform using GTKWave.

Deliverables

I. Verilog Code Screenshot
II. Verilog VVP Output Screen Shot
III. GTKWAVE Screenshot
IV. Output Table to be completed and included

- I Iverilog code

```
module mux4(
    input wire i0,
    input wire i1,
    input wire i2,
    input wire i3,
    input wire [1:0] sel,
    output wire o
);
    assign o = (sel == 2'b00) ? i0 :
               (sel == 2'b01) ? i1 :
               (sel == 2'b10) ? i2 :
                                i3;
endmodule
```

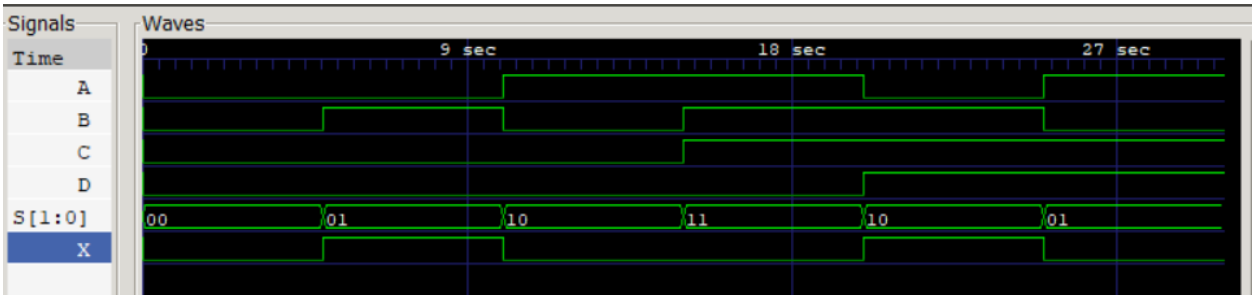
- Testbench

```
module TB;
    reg A, B, C, D;
    reg [1:0] S;
    wire X;


    mux4 newMUX(
        .i0(A),
        .i1(B),
        .i2(C),
        .i3(D),
        .sel(S),
        .o(X)
    );

    initial begin
        $dumpfile("MUX4_test.vcd");
        $dumpvars(0, TB);
        $monitor("Time=%0t | S=%b | A=%b | B=%b | C=%b | D=%b | X=%b", $time, S, A, B, C, D, X);


        A=0; B=0; C=0; D=0; S=2'b00; #5;
        A=0; B=1; C=0; D=0; S=2'b01; #5;
        A=1; B=0; C=0; D=0; S=2'b10; #5;
        A=1; B=1; C=1; D=0; S=2'b11; #5;
        A=0; B=1; C=1; D=1; S=2'b10; #5;
        A=1; B=0; C=1; D=1; S=2'b01; #5;

        $finish;
    end
endmodule
```

II VVP screenshot

```
C:\iverilog\bin\Week4>vvp dsn
VCD info: dumpfile MUX4_test.vcd opened for output.
Time=0 | S=00 | A=0 | B=0 | C=0 | D=0 | X=0
Time=5 | S=01 | A=0 | B=1 | C=0 | D=0 | X=1
Time=10 | S=10 | A=1 | B=0 | C=0 | D=0 | X=0
Time=15 | S=11 | A=1 | B=1 | C=1 | D=0 | X=0
Time=20 | S=10 | A=0 | B=1 | C=1 | D=1 | X=1
Time=25 | S=01 | A=1 | B=0 | C=1 | D=1 | X=0
```

III GTKwave screenshot



IV Truth Table

S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

(b) Function table

TITLE: Write a verilog program to model 1:4 demultiplexer. Generate the VVP output and simulation waveform using GTKWave.

Deliverables

I. Verilog Code Screenshot
II. Verilog VVP Output Screen Shot
III. GTKWAVE Screenshot
IV. Output Table to be completed and included

I.Verilog code

- Main code

```
module demux1to4(
    input wire i,
    input wire [1:0] sel,
    output wire y0, y1, y2, y3
);
    assign y0 = (sel == 2'b00) ? i : 1'b0;
    assign y1 = (sel == 2'b01) ? i : 1'b0;
    assign y2 = (sel == 2'b10) ? i : 1'b0;
    assign y3 = (sel == 2'b11) ? i : 1'b0;
endmodule
```

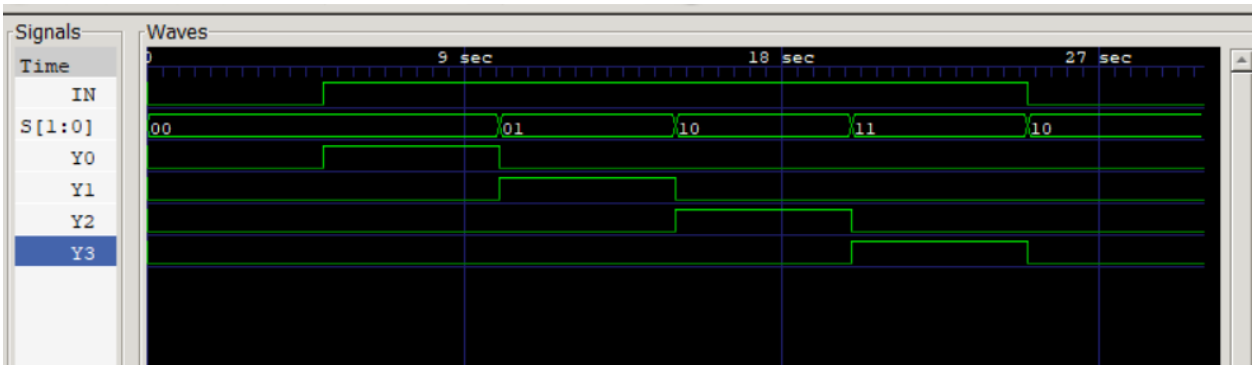
- Test bench

```
module TB;
    reg IN;
    reg [1:0] S;
    wire Y0, Y1, Y2, Y3;


    demux1to4 newDEMUX(
        .i(IN),
        .sel(S),
        .y0(Y0),
        .y1(Y1),
        .y2(Y2),
        .y3(Y3)
    );

    initial begin
        $dumpfile("DEMUX1to4_test.vcd");
        $dumpvars(0, TB);
        $monitor("Time=%0t | S=%b | IN=%b | Y0=%b | Y1=%b | Y2=%b | Y3=%b", $time, S, IN, Y0, Y1, Y2, Y3);


        IN = 1'b0; S = 2'b00; #5;
        IN = 1'b1; S = 2'b00; #5;
        IN = 1'b1; S = 2'b01; #5;
        IN = 1'b1; S = 2'b10; #5;
        IN = 1'b1; S = 2'b11; #5;
        IN = 1'b0; S = 2'b10; #5;

        $finish;
    end
endmodule
```

II VVP Screenshot

```
C:\iverilog\bin\Week4>vvp dsn
VCD info: dumpfile DEMUX1to4_test.vcd opened for output.
Time=0 | S=00 | IN=0 | Y0=0 | Y1=0 | Y2=0 | Y3=0
Time=5 | S=00 | IN=1 | Y0=1 | Y1=0 | Y2=0 | Y3=0
Time=10 | S=01 | IN=1 | Y0=0 | Y1=1 | Y2=0 | Y3=0
Time=15 | S=10 | IN=1 | Y0=0 | Y1=0 | Y2=1 | Y3=0
Time=20 | S=11 | IN=1 | Y0=0 | Y1=0 | Y2=0 | Y3=1
Time=25 | S=10 | IN=0 | Y0=0 | Y1=0 | Y2=0 | Y3=0
```

III GTKWave output



IV Truth Table

E	A	B	D0	D1	D2	D3
1	x	x	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

TITLE: Write a verilog program to model 1:2 demultiplexer. Generate the VVP output and simulation waveform using GTKWave.

Deliverables

I. Verilog Code Screenshot
II. Verilog VVP Output Screen Shot
III. GTKWAVE Screenshot
IV. Output Table to be completed and included

I Verilog Code

- Main code

```
module demux1to2(
    input wire i,
    input wire sel,
    output wire y0, y1
);
    assign y0 = (sel == 0) ? i : 1'b0;
    assign y1 = (sel == 1) ? i : 1'b0;
endmodule
```

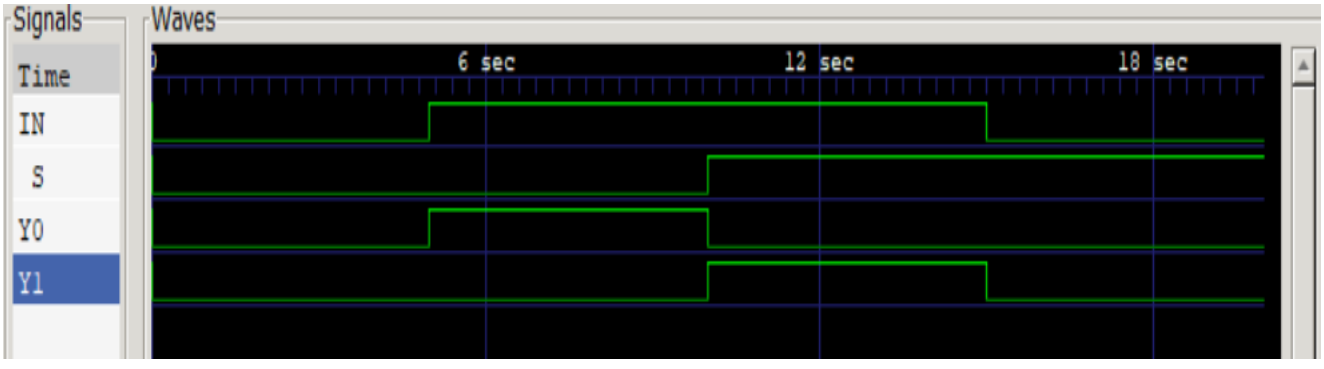
- Test bench

```
module TB;
    reg IN;
    reg S;
    wire Y0, Y1;


    demux1to2 newDEMUX(
        .i(IN),
        .sel(S),
        .y0(Y0),
        .y1(Y1)
    );

    initial begin
        $dumpfile("DEMUX1to2_test.vcd");
        $dumpvars(0, TB);
        $monitor("Time=%0t | S=%b | IN=%b | Y0=%b | Y1=%b", $time, S, IN, Y0, Y1);


        IN = 1'b0; S = 1'b0; #5;
        IN = 1'b1; S = 1'b0; #5;
        IN = 1'b1; S = 1'b1; #5;
        IN = 1'b0; S = 1'b1; #5;

        $finish;
    end
endmodule
```

II VVP output Screenshot

```
C:\iverilog\bin\Week4>vvp dsn
VCD info: dumpfile DEMUX1to2_test.vcd opened for output.
Time=0 | S=0 | IN=0 | Y0=0 | Y1=0
Time=5 | S=0 | IN=1 | Y0=1 | Y1=0
Time=10 | S=1 | IN=1 | Y0=0 | Y1=1
Time=15 | S=1 | IN=0 | Y0=0 | Y1=0
```

III GTKwave output screenshot



IV Truth Table

i (IN)	sel (S)	y0	y1
0	x	0	0
1	0	1	0
1	1	0	1

TITLE: Write a verilog program to model 4:2 encoder with input enabled. Generate the VVP output and simulation waveform using GTKWave.

Deliverables

I. Verilog Code Screenshot
II. Verilog VVP Output Screen Shot
III. GTKWAVE Screenshot
IV. Output Table to be completed and included

I Verilog Code Screenshot

- Main code

```
module enc4to2(
    input wire [3:0] d,
    input wire en,
    output wire [1:0] y,
    output wire valid
);

    assign y[1] = en & (d[3] | d[2]);
    assign y[0] = en & (d[3] | d[1]);
    assign valid = en & (d[3] | d[2] | d[1] | d[0]);

endmodule
```

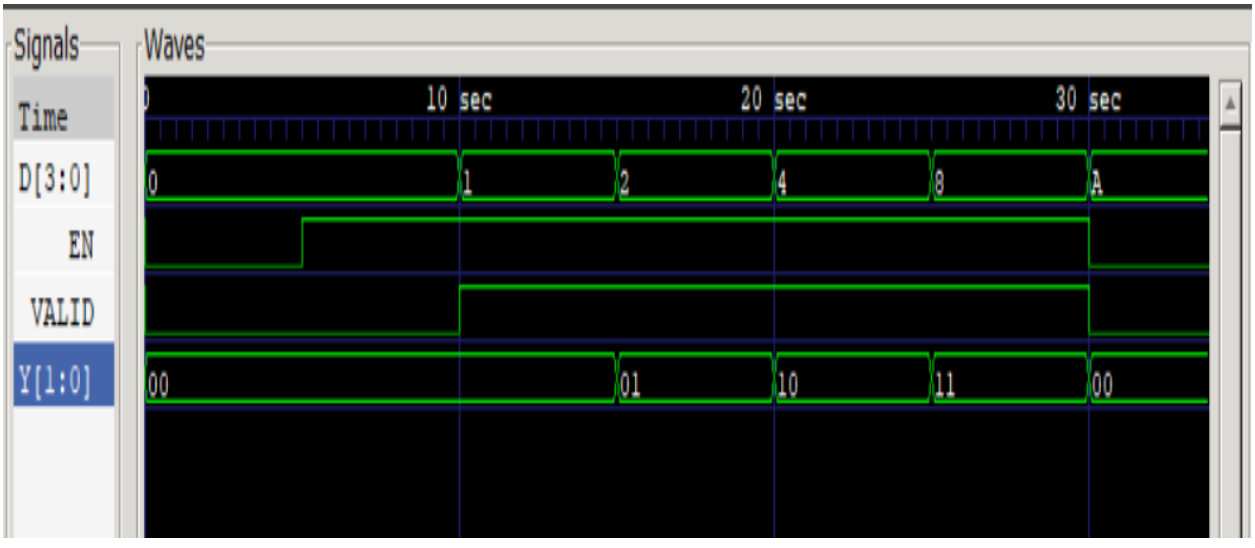
- Test Bench

```
module TB;
    reg [3:0] D;
    reg EN;
    wire [1:0] Y;
    wire VALID;


    enc4to2 newENC(
        .d(D),
        .en(EN),
        .y(Y),
        .valid(VALID)
    );

    initial begin
        $dumpfile("ENC4to2_test.vcd");
        $dumpvars(0, TB);
        $monitor("Time=%0t | EN=%b | D=%b | Y=%b | VALID=%b", $time, EN, D, Y, VALID);


        EN = 1'b0; D = 4'b0000; #5;
        EN = 1'b1; D = 4'b0000; #5;
        EN = 1'b1; D = 4'b0001; #5;
        EN = 1'b1; D = 4'b0010; #5;
        EN = 1'b1; D = 4'b0100; #5;
        EN = 1'b1; D = 4'b1000; #5;
        EN = 1'b0; D = 4'b1010; #5;

        $finish;
    end
endmodule
```

II VVP screenshot

```
C:\iverilog\bin\Week4>vvp dsn
VCD info: dumpfile ENC4to2_test.vcd opened for output.
Time=0 | EN=0 | D=0000 | Y=00 | VALID=0
Time=5 | EN=1 | D=0000 | Y=00 | VALID=0
Time=10 | EN=1 | D=0001 | Y=00 | VALID=1
Time=15 | EN=1 | D=0010 | Y=01 | VALID=1
Time=20 | EN=1 | D=0100 | Y=10 | VALID=1
Time=25 | EN=1 | D=1000 | Y=11 | VALID=1
Time=30 | EN=0 | D=1010 | Y=00 | VALID=0
```

III GTKwave output



IV Truth Table

Inputs				Output	
D_3	D_2	D_1	D_o	B	A
0	0	0	0	x	x
0	0	0	1	0	0
0	0	1	x	0	1
0	1	x	x	1	0
1	x	x	x	1	1