DIGITAL DESIGN AND COMPUTER ORGANIZATION LABORATORY

UE23CS251A

Computer Science and Engineering

Program 3.(2:1 Multiplexer)
Verilog Source File MUX2to1.v

Students need to complete this Verilog Source File MUX2to1.v

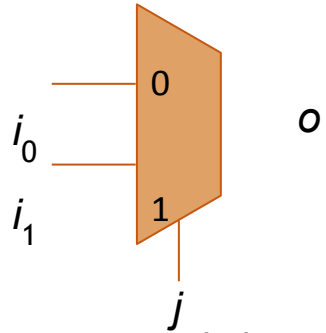
```
module mux2 (input wire i0, i1, j, output wire o);

assign o = ----------;

endmodule
```

Verilog Source File MUX

2:1 mux 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

2:1 mux symbol:



Data inputs: i_0, i_1

Control input: j

2 to 1 MUX Testbench

```
module TB;
reg A,B,S;
wire X;
initial
begin
$dumpfile("MUX2_test.vcd");
$dumpvars(0,TB);
end
mux2 newMUX(.i0(A), .i1(B), .j(S), .o(X));
```

2 to 1 MUX Testbench(Contd…)

```
initial
begin
    S = 1'b0;    A = 1'b0;B = 1'b0;
#5 S = 1'b0;A = 1'b0;B = 1'b1;
#5 S = 1'b0;A = 1'b1;B = 1'b0;
#5 S = 1'b0;A = 1'b1;B = 1'b1;
#5 S = 1'b1;A = 1'b0;B = 1'b0;
#5 S = 1'b1;A = 1'b0;B = 1'b1;
#5 S = 1'b1;A = 1'b1;B = 1'b0;
#5 S=1'b1;A = 1'b1;B = 1'b1;
end
endmodule
```

2:1 MUX Simulation

Execute the files MUX2to1 and testBench2to1 as mentioned below

Step1) iverilog -o testmux2 MUX2to1.v MUX2to1_tb.v

If the compilation went OK, you won't see any output.

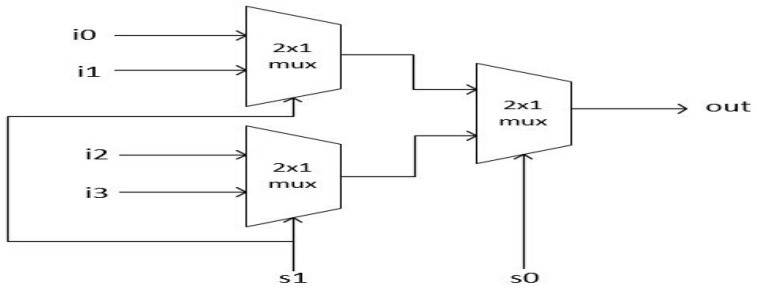
Step2) vvp testmux2

You can observe output on the console

Step3) gtkwave mux2_test.vcd

Output waveform will be observed

4:1 Mux using 2:1 Mux Multiplexers



i0	i1	i2	i3	s0	s1	out
1	0	1	1	0	1	0
0	1	0	0	0	1	1
0	0	1	0	1	0	1
0	0	0	1	1	1	1
1	0	0	0	0	0	1

Program 4(4:1 Multiplexer)
Verilog Source File MUX4to1.v

Students need to complete this Verilog Source File MUX4to1.v

```
module mux4 (input wire [0:3] i, input wire j1, j0, output
wire o);
wire  t0, t1;
mux2 mux2_0 (------------------);
mux2 mux2_1 (------------------);
mux2 mux2_2 (------------------);
Endmodule
```

4 to 1 MUX Testbench

```
module TB;
reg [0:3]ii;
reg s0;reg s1;
‘wire yy;
initial
Begin
$dumpfile("MUX4_test.vcd");
$dumpvars(0, TB);
end
mux4 newMUX(.i(ii), .j0(s0),.j1(s1),.o(yy));
initial
begin
```

4 to 1 MUX Testbench(Contd…)

```
 ii = 4'b0000;s0=1'b0;s1=1'b0;
#5      ii = 4'b1000;s0=1'b0;s1=1'b0;
#5      ii = 4'b0000;s0=1'b0;s1=1'b1;
#5      ii = 4'b0100;s0=1'b0;s1=1'b1;
#5      ii = 4'b0000;s0=1'b1;s1=1'b0;
#5      ii = 4'b0010;s0=1'b1;s1=1'b0;
#5      ii = 4'b0000;s0=1'b1;s1=1'b1;
#5      ii = 4'b0001;s0=1'b1;s1=1'b1;
end
endmodule
```

4 to 1 MUX Truthtable (Contd…)

i0	i1	i2	i3	s0	s1	y
0	0	0	0	0	0	
1	0	0	0	0	0	
0	0	0	0	0	1	
0	1	0	0	0	1	
0	0	0	0	1	1	
0	0	1	0	1	1	
0	0	0	0	1	1	
0	0	0	1	1	1	

4 to 1 MUX Testbench(Contd…)

```
 ii = 4'b0000;s0=1'b0;s1=1'b0;
#5      ii = 4'b1000;s0=1'b0;s1=1'b0;
#5      ii = 4'b0000;s0=1'b0;s1=1'b1;
#5      ii = 4'b0100;s0=1'b0;s1=1'b1;
#5      ii = 4'b0000;s0=1'b1;s1=1'b0;
#5      ii = 4'b0010;s0=1'b1;s1=1'b0;
#5      ii = 4'b0000;s0=1'b1;s1=1'b1;
#5      ii = 4'b0001;s0=1'b1;s1=1'b1;
end
endmodule
```

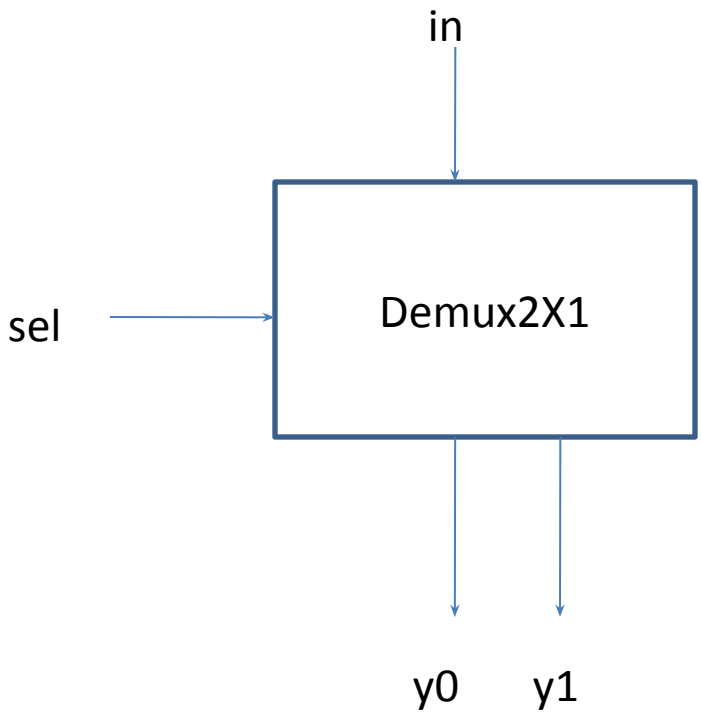
4:1 MUX Simulation

Program 4.(4:1 Multiplexer using 2:1 Multiplexer)
- students have to complete MUX4to1.v file

Execute the files MUX2to1.v MUX4to1.v and MUX4to1_tb.v using steps below and observe the output

1) iverilog -o test_mux4 MUX2to1.v MUX4to1.v MUX4to1_tb.v
2) vvp test_mux4
3) gtkwave mux4_test.vcd

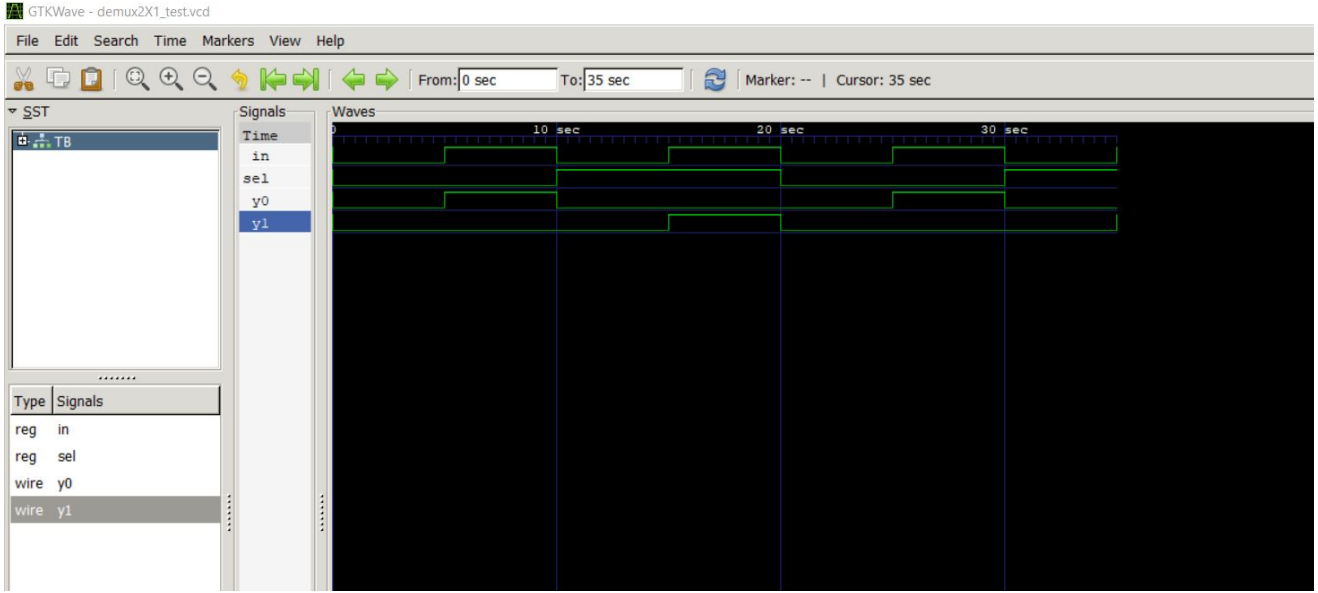
1X2 Demux



1X2 Demux

sel	in	y0	y1
0	0	0	0
0	1	1	0
1	0	0	0
1	1	0	1

1X2 Demux


GTKWave - demux2X1_test.vcd
File Edit Search Time Markers View Help
From: 0 sec
To: 35 sec
Marker: --
Cursor: 35 sec
SST
TB
Type Signals
reg in
reg sel
wire y0
wire y1
Signals
Time
in
sel
y0
y1
Waves
10 sec
20 sec
30 sec