**Lab-1.v**

`timescale 1ns / 1ps

module MUX2x1(in,sel,out);

input [1:0] in;

input sel;

output reg out;

always @(\*) begin

if (sel == 0)

out <= in[0];

else

out <= in[1];

end

endmodule

module MUX4x1(in,sel,out);

input [3:0] in;

input [1:0] sel;

output out;

wire [1:0] m\_out;

MUX2x1 m1(in[1:0], sel[0], m\_out[0]);

MUX2x1 m2(in[3:2], sel[0], m\_out[1]);

MUX2x1 m3(m\_out, sel[1], out);

endmodule

**Lab-1-tb.v**

`timescale 1ns / 1ps

module main\_tb();

reg [3:0] in;

reg [1:0] sel;

wire out;

MUX4x1 m1(in, sel, outp);

integer i;

initial begin

$monitor("in: %b\nsel: %b\nout: %b\n\n", in, sel, out);

in = 4'b0110;

sel = 2'b00;

for ( i=0; i < 3; i = i + 1 ) begin

#100;

sel = sel + 1;

end

end

endmodule

**$monitor Output**

in: 0110

sel: 00

out: z

in: 0110

sel: 01

out: z

in: 0110

sel: 10

out: z

in: 0110

sel: 11

out: z

![A screenshot of a computer

Description automatically generated]()