

# Code

## lab6.sv

```
module lab6 ();

    function logic [7:0] decoder3to8 (logic en_b, logic [2:0] in);
        casex({en_b, in})
            4'b1xxx: return 8'b1111_1111;
            4'b0000: return 8'b1111_1110;
            4'b0001: return 8'b1111_1101;
            4'b0010: return 8'b1111_1011;
            4'b0011: return 8'b1111_0111;
            4'b0100: return 8'b1110_1111;
            4'b0101: return 8'b1101_1111;
            4'b0110: return 8'b1011_1111;
            4'b0111: return 8'b0111_1111;
        endcase
    endfunction

    function logic [2:0] encoder8to3 (logic en_b, logic [7:0] in_b);
        casex({en_b, in_b})
            9'b1_xxxxxxxx: return 3'b111;
            9'b0_xxxxxxxx0: return 3'b000;
            9'b0_xxxxxxxx01: return 3'b001;
            9'b0_xxxxxx011: return 3'b010;
            9'b0_xxxxx0111: return 3'b011;
            9'b0_xxx01111: return 3'b100;
            9'b0_xx011111: return 3'b101;
            9'b0_x0111111: return 3'b110;
            9'b0_01111111: return 3'b111;
        endcase
    endfunction

    logic [2:0] in, out;
    logic      clk;

    assign out = encoder8to3(0, decoder3to8(0, in));

    initial begin
        clk = 1'b1;
        in  = 3'b111;
        forever #5 clk = ~clk;
        $display("Current input/output");
        $monitor("%b, %b", in, out);
    end
end
```

```

always @ (posedge clk)
    in = in + 1;

initial
    #80 $finish();

endmodule

```

## Deliverables

