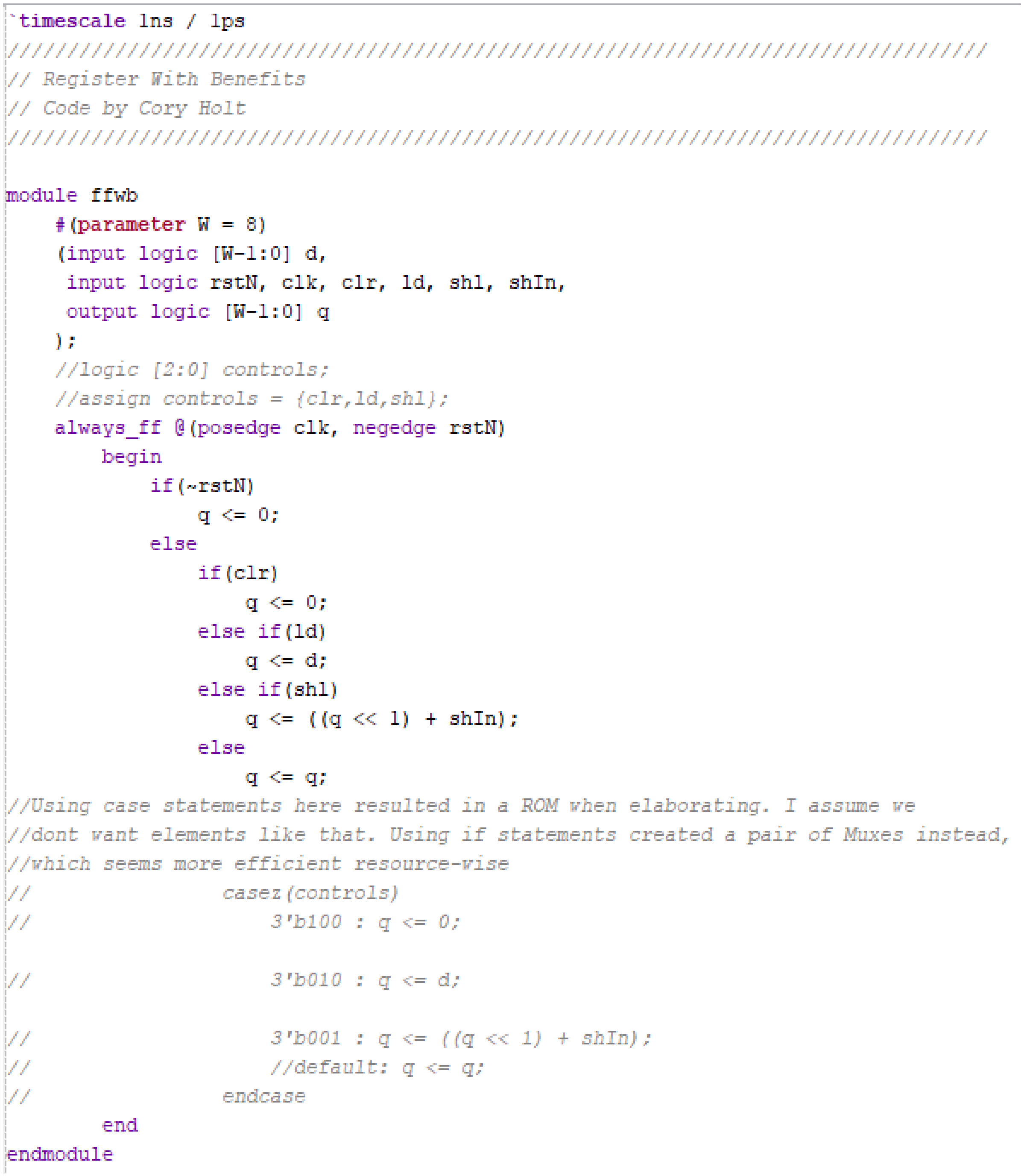
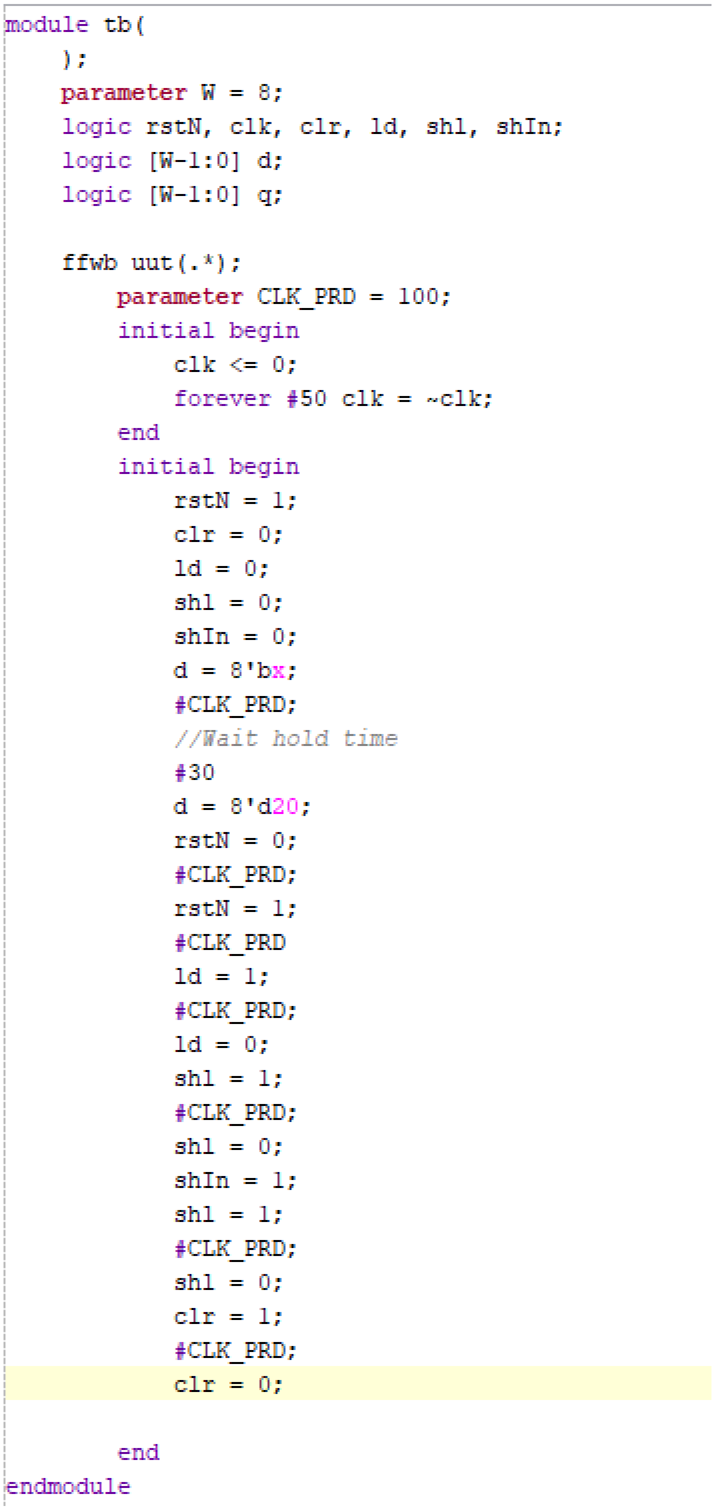
Cory Holt

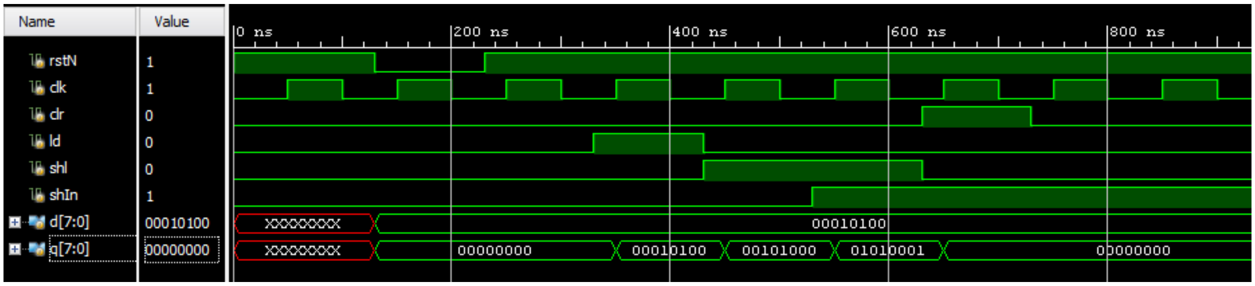
ECE 440 H3



```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Register With Benefits
// Code by Cory Holt
//////////////////////////////////////////////////////////////////////////////////

module ffwb
    #(parameter W = 8)
    (input logic [W-1:0] d,
     input logic rstN, clk, clr, ld, shl, shIn,
     output logic [W-1:0] q
    );
    //logic [2:0] controls;
    //assign controls = {clr,ld,shl};
    always_ff @(posedge clk, negedge rstN)
        begin
            if(~rstN)
                q <= 0;
            else
                if(clr)
                    q <= 0;
                else if(ld)
                    q <= d;
                else if(shl)
                    q <= ((q << 1) + shIn);
                else
                    q <= q;
//Using case statements here resulted in a ROM when elaborating. I assume we
//dont want elements like that. Using if statements created a pair of Muxes instead,
//which seems more efficient resource-wise
//                  casez(controls)
//                      3'b100 : q <= 0;

//                      3'b010 : q <= d;

//                      3'b001 : q <= ((q << 1) + shIn);
//                      //default: q <= q;
//                  endcase
        end
endmodule
```



```
module tb(
    );
    parameter W = 8;
    logic rstN, clk, clr, ld, shl, shIn;
    logic [W-1:0] d;
    logic [W-1:0] q;

    ffwb uut(.*);
        parameter CLK_PRD = 100;
        initial begin
            clk <= 0;
            forever #50 clk = ~clk;
        end
        initial begin
            rstN = 1;
            clr = 0;
            ld = 0;
            shl = 0;
            shIn = 0;
            d = 8'bx;
            #CLK_PRD;
            //Wait hold time
            #30
            d = 8'd20;
            rstN = 0;
            #CLK_PRD;
            rstN = 1;
            #CLK_PRD
            ld = 1;
            #CLK_PRD;
            ld = 0;
            shl = 1;
            #CLK_PRD;
            shl = 0;
            shIn = 1;
            shl = 1;
            #CLK_PRD;
            shl = 0;
            clr = 1;
            #CLK_PRD;
            clr = 0;

        end
endmodule
```



Name	Value
rstN	1
clk	1
clr	0
ld	0
shl	0
shIn	1
d[7:0]	00010100
q[7:0]	00000000