ECE3300L

Lab 2

By Justin Wong, Hector Garibay

Summer 2025

Report Date: June 25, 2025

Objective

We will design, simulate, and implement a 4-to-16 decoder with an enable input. When enabled, exactly one of 16 outputs goes HIGH based on the 4-bit input; when disabled, all outputs are LOW.

Verilog Code:

Behavioral Code:

```
module decoder4x16_behav (
    input  wire [3:0] A,
    input  wire       E,
    output reg  [15:0] Y
);

// [GREEN] reset outputs every combinational evaluation
always @(*) begin
    Y = 16'b0;

    // [BLUE] decode only when enabled
    if (E) begin
        // [ORANGE] mapping a to y[i]
        case (A)
            4'b0000: Y = 16'b0000_0000_0000_0001; // Y0
            4'b0001: Y = 16'b0000_0000_0000_0010; // Y1
            4'b0010: Y = 16'b0000_0000_0000_0100; // Y2
            4'b0011: Y = 16'b0000_0000_0000_1000; // Y3
            4'b0100: Y = 16'b0000_0000_0001_0000; // Y4
            4'b0101: Y = 16'b0000_0000_0010_0000; // Y5
            4'b0110: Y = 16'b0000_0000_0100_0000; // Y6
            4'b0111: Y = 16'b0000_0000_1000_0000; // Y7
            4'b1000: Y = 16'b0000_0001_0000_0000; // Y8
            4'b1001: Y = 16'b0000_0010_0000_0000; // Y9
            4'b1010: Y = 16'b0000_0100_0000_0000; // Y10
            4'b1011: Y = 16'b0000_1000_0000_0000; // Y11
            4'b1100: Y = 16'b0001_0000_0000_0000; // Y12
            4'b1101: Y = 16'b0010_0000_0000_0000; // Y13
            4'b1110: Y = 16'b0100_0000_0000_0000; // Y14
            4'b1111: Y = 16'b1000_0000_0000_0000; // Y15
            default: Y = 16'b0;
        endcase
    end
end
endmodule
```

Gate-Level Code:

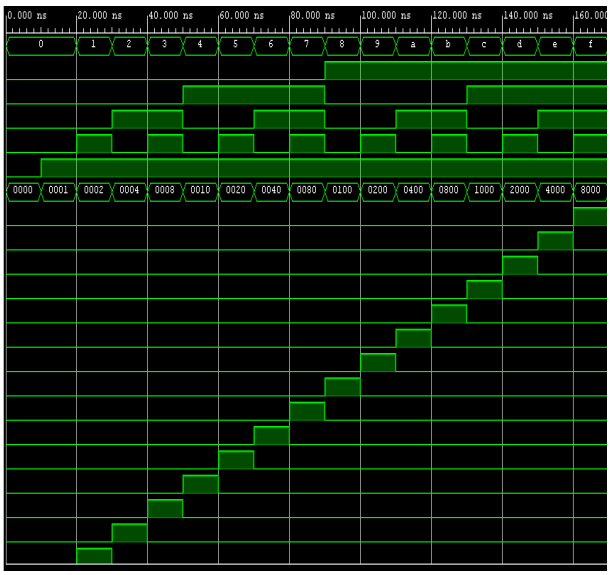
```
module decoder4x16_gate (
    input  wire [3:0] A,   // A[3] -> A[0] from SW3 -> SW0
    input  wire       E,   // Enable from SW4
    output wire [15:0] Y   // outputs to LD15 -> LD0
);

// block enables
wire en_low  = E & ~A[3];   // select Y[7:0]
wire en_high = E &  A[3];   // select Y[15:8]

// instantiate two 3-to-8 decoders
dec3to8_gate low_block  (.a(A[2:0]), .en(en_low ), .y(Y[7:0]
dec3to8_gate high_block (.a(A[2:0]), .en(en_high), .y(Y[15:8

endmodule



// 3-to-8 decoder gate implementationn

module dec3to8_gate (
    input  wire [2:0] a,
    input  wire       en,
    output wire [7:0] y
);
/* assign outputs based on enable and inputs.*/
assign y[0] = en & ~a[2] & ~a[1] & ~a[0];
assign y[1] = en & ~a[2] & ~a[1] &  a[0];
assign y[2] = en & ~a[2] &  a[1] & ~a[0];
assign y[3] = en & ~a[2] &  a[1] &  a[0];
assign y[4] = en &  a[2] & ~a[1] & ~a[0];
assign y[5] = en &  a[2] & ~a[1] &  a[0];
assign y[6] = en &  a[2] &  a[1] & ~a[0];
assign y[7] = en &  a[2] &  a[1] &  a[0];
endmodule
```

Simulation:

Test bench Code

```
    reg  [3:0] A;
    reg        E;
    wire [15:0] Y;


    decoder4x16_behav test(.A(A), .E(E), .Y(Y));   // instantiating with

    integer i;

    initial begin

       // Check that outputs stay LOW when disabled
        E = 1'b0;
        A = 4'b0000;
        #10;
        if (Y !== 16'b0) $fatal(1, "FAIL: Y not zero when E=0"); // if 1
        $display("PASS: E=0 forces Y=0");

        //all 16 input patterns with enable HIGH
        E = 1'b1; // set Enable on
       // looping through possible input values
        for (i = 0; i < 16; i = i + 1) begin
            A = i[3:0];
            #10; // 10 unit delay
            if (Y !== (16'b1 << i)) // checking if output Y is not equal
            //if not, print error
                $fatal(1,
                    "FAIL: A=%0h  expected=%016b  got=%016b",
                    A, (16'b1 << i), Y
                );
        end
        $display("PASS: All 16 combinations decoded correctly");

       // display message that all tests passed
        $display("\nALL TESTS PASSED\n");
```

Waveform sample



Implementation:

Utilization Table:

Resource	Utilization	Available	Utilization %
LUT	8	63400	0.01
IO	21	210	10.00

Timing Summary

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	inf	Worst Hold Slack (WHS):	inf	Worst Pulse Width Slack (WPWS):	NA
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	NA
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	NA
Total Number of Endpoints:	16	Total Number of Endpoints:	16	Total Number of Endpoints:	NA

Video:

https://youtu.be/CyE5YjkD8ik

Reflection:

In this lab, we developed a Verilog module for the Nexys A7 100T Board. The goal of this lab was to create a 4-16 decoder with an enable switch. This was accomplished by creating a Verilog module. We simulated our code using a separate testbench that we created. When the enable is LOW, all LEDS should be turned off, when the enable is HIGH, a single LED will turn on. The combination of the four switches on the board represents a number in binary. LEDs 2-16 are used to represent a 4-bit number.

Contributions:

Justin Wong: XDC, decoder4x16, behavioral code, testbench code, report. 50% for all.
Hector Garibay: XDC, decoder4x16, behavioral code, testbench code, report. 50% for all.