CALIFORNIA STATE POLYTECHNIC UNIVERSITY, POMONA
COLLEGE OF ENGINEERING

LAB 2

4x16 Decoder Design on Nexys A7-100T FPGA
ECE 3300L Summer 2025

Digital Circuit Design using Verilog

Professor Mohamed Aly

By: Clay Kim and Changwe Musonda

Objective

Design, simulate, and implement a 4-to-16 decoder with an enable input. Once enabled, exactly one of the 16 outputs goes HIGH based on the 4-bit input and when disabled, all outputs are LOW. This lab will allow us to practice structural coding, behavioral coding, writing a selfchecking testbench, reading Vivado’s resource utilization, and documenting work clearly in a beginner friendly report.

Hardware Components

Nexys A7-100T Kit

- Inputs: SW3…SW0 → 4-bit input A[3:0]
- Enable: SW4 → enable signal E
- Outputs: LD0…LD15 → 16-bit output Y[15:0]

Verilog Code

Gate-Level Implementation

```
module decoder4x16_gate(
    input wire [3:0] A,
    input E,
    output wire [15:0] Y
    );
    assign Y[0] = E&~A[3]&~A[2]&~A[1]&~A[0];
    assign Y[1] = E&~A[3]&~A[2]&~A[1]&A[0];
    assign Y[2] = E&~A[3]&~A[2]&A[1]&~A[0];
    assign Y[3] = E&~A[3]&~A[2]&A[1]&A[0];
    assign Y[4] = E&~A[3]&A[2]&~A[1]&~A[0];
    assign Y[5] = E&~A[3]&A[2]&~A[1]&A[0];
    assign Y[6] = E&~A[3]&A[2]&A[1]&~A[0];
    assign Y[7] = E&~A[3]&A[2]&A[1]&A[0];
    assign Y[8] = E&A[3]&~A[2]&~A[1]&~A[0];
    assign Y[9] = E&A[3]&~A[2]&~A[1]&A[0];
    assign Y[10] = E&A[3]&~A[2]&A[1]&~A[0];
    assign Y[11] = E&A[3]&~A[2]&A[1]&A[0];
    assign Y[12] = E&A[3]&A[2]&~A[1]&~A[0];
    assign Y[13] = E&A[3]&A[2]&~A[1]&A[0];
    assign Y[14] = E&A[3]&A[2]&A[1]&~A[0];
    assign Y[15] = E&A[3]&A[2]&A[1]&A[0];
endmodule
```

Behavioral Implementation

```
module decoder4x16_behavioral(
    input wire[3:0] A,
    input E,
    output reg [15:0] Y
    );
    always @(*) begin
        if(E) begin //enable check
            case (A)
                4'h0: Y = 16'b0000_0000_0000_0001; //map first switch to first LED
                4'h1: Y = 16'b0000_0000_0000_0010; //map second switch to second LED
                4'h2: Y = 16'b0000_0000_0000_0100; //map third switch to third LED
                4'h3: Y = 16'b0000_0000_0000_1000; //map fourth switch to fourth LED
                4'h4: Y = 16'b0000_0000_0001_0000; //map fifth switch to fifth LED
                4'h5: Y = 16'b0000_0000_0010_0000; //map sixth switch to sixth LED
                4'h6: Y = 16'b0000_0000_0100_0000; //map seventh switch to seventh LED
                4'h7: Y = 16'b0000_0000_1000_0000; //map eighth switch to eighth LED
                4'h8: Y = 16'b0000_0001_0000_0000; //map ninth switch to ninth LED
                4'h9: Y = 16'b0000_0010_0000_0000; //map tenth switch to tenth LED
                4'hA: Y = 16'b0000_0100_0000_0000; //map eleventh switch to eleventh LED
                4'hB: Y = 16'b0000_1000_0000_0000; //map twelvth switch to twelvth LED
                4'hC: Y = 16'b0001_0000_0000_0000; //map thirteenth switch to thirteenth LED
                4'hD: Y = 16'b0010_0000_0000_0000; //map fourteenth switch to fourteenth LED
                4'hE: Y = 16'b0100_0000_0000_0000; //map fifthteenth switch to fifthteenth LED
                4'hF: Y = 16'b1000_0000_0000_0000; //map sixteenth switch to sixteenth LED
            endcase
        end
        else begin
```

```
            Y = 16'h0; //reset
        end
    end
endmodule
```

Testbench

```
module tb_decoder4x16(
    );
    reg [3:0] A;
    reg E;
    wire [15:0] Y;
    integer i;
    decoder4x16_gate TB_TESTING(.A(A), .Y(Y), .E(E));
   reg [15:0] expectedY [0:15];
    initial begin
        expectedY[0] = 16'h0001;
        expectedY[1] = 16'h0002;
        expectedY[2] = 16'h0004;
        expectedY[3] = 16'h0008;
        expectedY[4] = 16'h0010;
        expectedY[5] = 16'h0020;
        expectedY[6] = 16'h0040;
        expectedY[7] = 16'h0080;
        expectedY[8] = 16'h0100;
        expectedY[9] = 16'h0200;
        expectedY[10] = 16'h0400;
        expectedY[11] = 16'h0800;
        expectedY[12] = 16'h1000;
        expectedY[13] = 16'h2000;
        expectedY[14] = 16'h4000;
        expectedY[15] = 16'h8000;
        E = 1'b1;
        $display("ENABLE HIGH");
        for(i=0; i<16; i=i+1) begin
            A=i;
            #20;
            $display("INPUT: %04b | EXPECTED: %04h | RESULT: %04h (%s)", A, expectedY[i], Y, (expectedY[i] == Y) ? "PASSED" : "FAILED");
        end
        E = 1'b0;
        $display("ENABLE LOW");
        for(i=0; i<16; i=i+1) begin
            A=i;
            #20;
            $display("INPUT: %04b | EXPECTED: %04h | RESULT: %04h (%s)", A, 16'h0000, Y, (16'h0000 == Y) ? "PASSED" : "FAILED");
        end
        $finish;
    end
endmodule
```

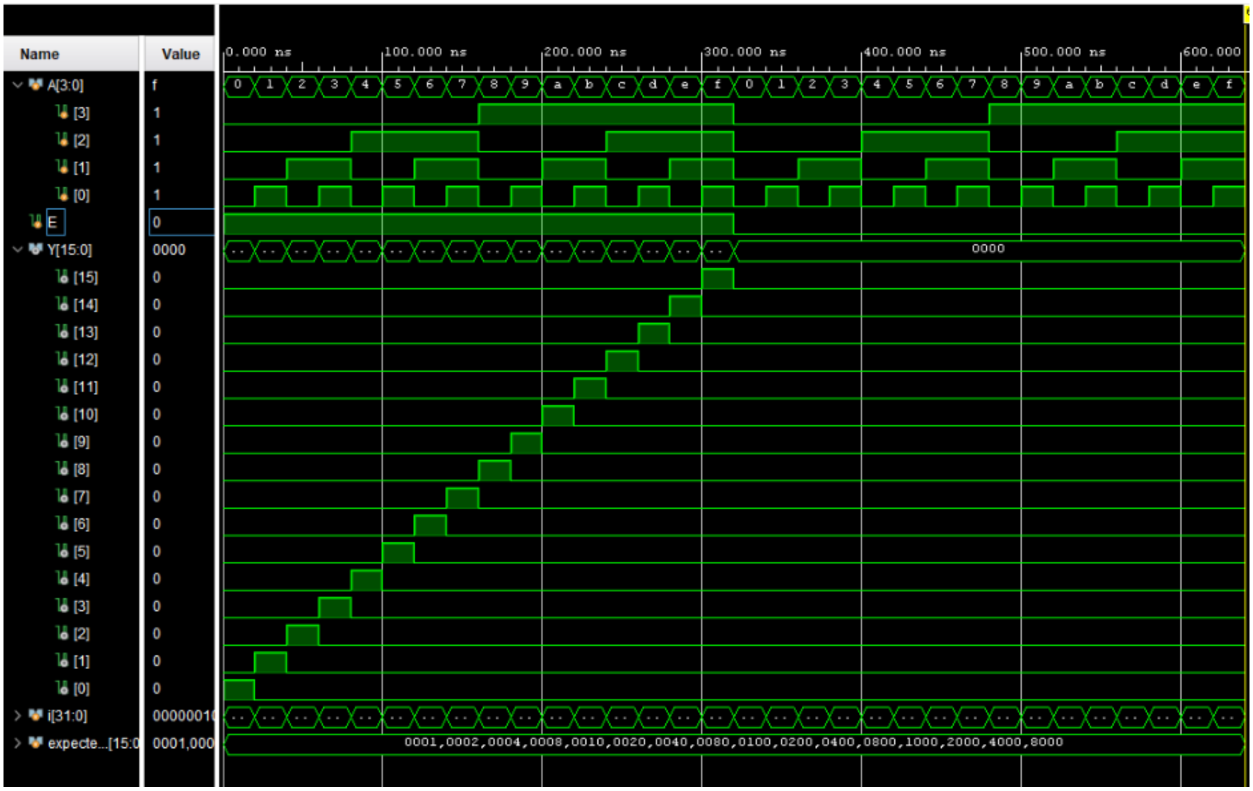
XDC Snippets

```
# Clock signal
set_property -dict { PACKAGE_PIN E3    IOSTANDARD LVCMOS33 } [get_ports { CLK100MHZ }]; #IO_L12P_T1_MRCC_35 Sch=clk100mhz
create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports {CLK100MHZ}];



##Switches


set_property -dict { PACKAGE_PIN J15   IOSTANDARD LVCMOS33 } [get_ports { A[0] }]; #IO_L24N_T3_RS0_15 Sch=sw[0]
set_property -dict { PACKAGE_PIN L16   IOSTANDARD LVCMOS33 } [get_ports { A[1] }]; #IO_L3N_T0_DQS_EMCCLK_14 Sch=sw[1]
set_property -dict { PACKAGE_PIN M13   IOSTANDARD LVCMOS33 } [get_ports { A[2] }];
#IO_L6N_T0_D08_VREF_14 Sch=sw[2]

set_property -dict { PACKAGE_PIN R15   IOSTANDARD LVCMOS33 } [get_ports { A[3] }];
#IO_L13N_T2_MRCC_14 Sch=sw[3]

set_property -dict { PACKAGE_PIN R17   IOSTANDARD LVCMOS33 } [get_ports { E }];
#IO_L12N_T1_MRCC_14 Sch=sw[4]

#set_property -dict { PACKAGE_PIN T18   IOSTANDARD LVCMOS33 } [get_ports { y[2] }];
#IO_L7N_T1_D10_14 Sch=sw[5]

#set_property -dict { PACKAGE_PIN U18   IOSTANDARD LVCMOS33 } [get_ports { y[1] }];
#IO_L17N_T2_A13_D29_14 Sch=sw[6]

#set_property -dict { PACKAGE_PIN R13   IOSTANDARD LVCMOS33 } [get_ports { y[0] }];
#IO_L5N_T0_D07_14 Sch=sw[7]

#set_property -dict { PACKAGE_PIN T8    IOSTANDARD LVCMOS18 } [get_ports { c_in }]; #IO_L24N_T3_34
Sch=sw[8]

#set_property -dict { PACKAGE_PIN U8    IOSTANDARD LVCMOS18 } [get_ports { SW[9] }]; #IO_25_34
Sch=sw[9]

#set_property -dict { PACKAGE_PIN R16   IOSTANDARD LVCMOS33 } [get_ports { SW[10] }];
#IO_L15P_T2_DQS_RDWR_B_14 Sch=sw[10]

#set_property -dict { PACKAGE_PIN T13   IOSTANDARD LVCMOS33 } [get_ports { SW[11] }];
#IO_L23P_T3_A03_D19_14 Sch=sw[11]

#set_property -dict { PACKAGE_PIN H6    IOSTANDARD LVCMOS33 } [get_ports { SW[12] }];
#IO_L24P_T3_35 Sch=sw[12]

#set_property -dict { PACKAGE_PIN U12   IOSTANDARD LVCMOS33 } [get_ports { SW[13] }];
#IO_L20P_T3_A08_D24_14 Sch=sw[13]

#set_property -dict { PACKAGE_PIN U11   IOSTANDARD LVCMOS33 } [get_ports { SW[14] }];
#IO_L19N_T3_A09_D25_VREF_14 Sch=sw[14]

#set_property -dict { PACKAGE_PIN V10   IOSTANDARD LVCMOS33 } [get_ports { SW[15] }];
#IO_L21P_T3_DQS_14 Sch=sw[15]



## LEDs



set_property -dict { PACKAGE_PIN H17   IOSTANDARD LVCMOS33 } [get_ports { Y[0] }];
#IO_L18P_T2_A24_15 Sch=led[0]

set_property -dict { PACKAGE_PIN K15   IOSTANDARD LVCMOS33 } [get_ports { Y[1] }];
#IO_L24P_T3_RS1_15 Sch=led[1]

set_property -dict { PACKAGE_PIN J13   IOSTANDARD LVCMOS33 } [get_ports { Y[2] }];
#IO_L17N_T2_A25_15 Sch=led[2]
```

```
set_property -dict { PACKAGE_PIN N14   IOSTANDARD LVCMOS33 } [get_ports { Y[3] }];
#IO_L8P_T1_D11_14 Sch=led[3]

set_property -dict { PACKAGE_PIN R18   IOSTANDARD LVCMOS33 } [get_ports { Y[4] }];
#IO_L7P_T1_D09_14 Sch=led[4]

set_property -dict { PACKAGE_PIN V17   IOSTANDARD LVCMOS33 } [get_ports { Y[5] }];
#IO_L18N_T2_A11_D27_14 Sch=led[5]

set_property -dict { PACKAGE_PIN U17   IOSTANDARD LVCMOS33 } [get_ports { Y[6] }];
#IO_L17P_T2_A14_D30_14 Sch=led[6]

set_property -dict { PACKAGE_PIN U16   IOSTANDARD LVCMOS33 } [get_ports { Y[7] }];
#IO_L18P_T2_A12_D28_14 Sch=led[7]

set_property -dict { PACKAGE_PIN V16   IOSTANDARD LVCMOS33 } [get_ports { Y[8] }];
#IO_L16N_T2_A15_D31_14 Sch=led[8]

set_property -dict { PACKAGE_PIN T15   IOSTANDARD LVCMOS33 } [get_ports { Y[9] }];
#IO_L14N_T2_SRCC_14 Sch=led[9]

set_property -dict { PACKAGE_PIN U14   IOSTANDARD LVCMOS33 } [get_ports { Y[10] }];
#IO_L22P_T3_A05_D21_14 Sch=led[10]

set_property -dict { PACKAGE_PIN T16   IOSTANDARD LVCMOS33 } [get_ports { Y[11] }];
#IO_L15N_T2_DQS_DY_CSO_B_14 Sch=led[11]

set_property -dict { PACKAGE_PIN V15   IOSTANDARD LVCMOS33 } [get_ports { Y[12] }];
#IO_L16P_T2_CSI_B_14 Sch=led[12]

set_property -dict { PACKAGE_PIN V14   IOSTANDARD LVCMOS33 } [get_ports { Y[13] }];
#IO_L22N_T3_A04_D20_14 Sch=led[13]

set_property -dict { PACKAGE_PIN V12   IOSTANDARD LVCMOS33 } [get_ports { Y[14] }];
#IO_L20N_T3_A07_D23_14 Sch=led[14]

set_property -dict { PACKAGE_PIN V11   IOSTANDARD LVCMOS33 } [get_ports { Y[15] }];
#IO_L21N_T3_DQS_A06_D22_14 Sch=led[15]
```

Synthesis and Implementation

Screenshot:



Group Video

https://youtu.be/2DiERelA0Ic

Reflections

This lab provided an effective hands-on experience in learning Verilog coding for implementation of a 4-to-16 decoder onto the Nexys A7 Board. It also solidified the difference between gate level versus behavioral coding and their associated advantages and disadvantages for hardware implementation. Being able to create a testbench for my decoder put me in the mindset of verification and validation when dealing with FPGAs. Experiencing timing reports and resource utilization review via the Vivado implementation of my projects exposed me to the necessity of proper efficiency and optimization awareness within a design. This all adds to my future efforts in the field with digital logic design and FPGA usage for future complex embedded systems.