CalPolyPomona
College of Engineering

Lab 2 Report

by

Jonathan Huynh #016137719

Adam Godfrey #015981472

Instructor: Dr. Mohamed Aly

Class: ECE 3300L.E02-OU - Verilog Design

June 25th, 2025

Summary:

In lab 2, we are tasked with designing, simulating, and implementing a 4-to-16 decoder along with an enable input on our Digilent Nexys A7-100T board.

Design:

Behavior Level Snippet:

```
always @(*) begin
  Y = 16'b0; // 🟩 reset all outputs to 0
  if (E) begin // 🟦 only decode when enabled
    case (A)
      4'b0000: Y = 16'b0000_0000_0000_0001; // 🟧 output 0
      4'b0001: Y = 16'b0000_0000_0000_0010;
      4'b0010: Y = 16'b0000_0000_0000_0100;
      4'b0011: Y = 16'b0000_0000_0000_1000;
      4'b0100: Y = 16'b0000_0000_0001_0000;
      4'b0101: Y = 16'b0000_0000_0010_0000;
      4'b0110: Y = 16'b0000_0000_0100_0000;
      4'b0111: Y = 16'b0000_0000_1000_0000;
      4'b1000: Y = 16'b0000_0001_0000_0000;
      4'b1001: Y = 16'b0000_0010_0000_0000;
      4'b1010: Y = 16'b0000_0100_0000_0000;
      4'b1011: Y = 16'b0000_1000_0000_0000;
      4'b1100: Y = 16'b0001_0000_0000_0000;
      4'b1101: Y = 16'b0010_0000_0000_0000;
      4'b1110: Y = 16'b0100_0000_0000_0000;
      4'b1111: Y = 16'b1000_0000_0000_0000;
    endcase
  end
end
```

Gate Level Snippet:

```
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];
```

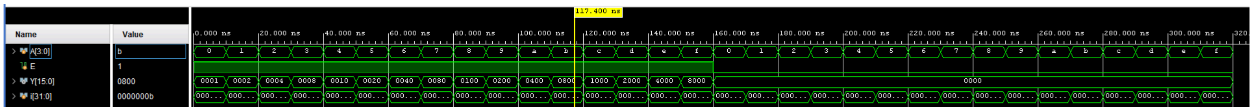
Simulation:

Testbench Description:

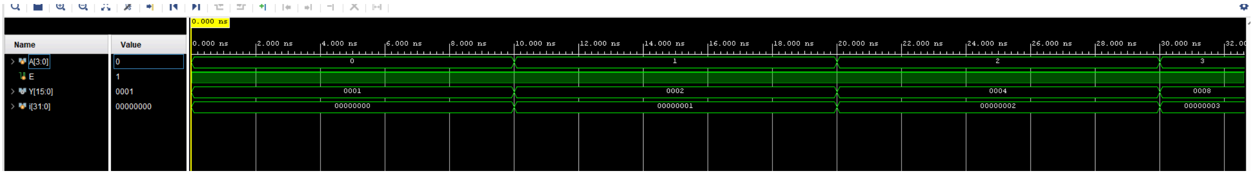
The testbench tb_decoder4x16 is designed to verify the correct functionality of a 4-to-16 line decoder module (decoder4x16_behav). The decoder takes a 4-bit input A, an enable signal E, and produces a 16-bit one-hot output Y.

- Enable = 1 (Active Phase):
    - When the enable signal E is asserted (E = 1), the testbench iterates through all 16 possible values of the 4-bit input A (from 0000 to 1111)
    - For each input value, it checks that the decoder output Y produces a one-hot encoding by comparing it to 1 << A
    - If the output does not match, the simulation halts with a failure message; otherwise, a success message is displayed.
- Enable = 0 (Disabled Phase):
    - After verifying all valid inputs with the decoder enabled, the testbench disables the decoder by setting E = 0
    - It again iterates over all values of A to ensure that the output Y remains zero regardless of the input
    - Any non-zero output is treated as an error and flagged.

Sample Waveform:



Zoomed in to 3 periods:



Implementation:

Resource Utilization Table

Resource	Used	Available	Utilization
LUTs	8	63,400	0.0126%
Registers (FFs)	0	126,800	0%
Bonded IOB	21	210	10%

Timing Summary:

Worst Negative Slack (WNS)	inf
Total Negative Slack (TNS)	0
Worst Hold Slack (WHS)	inf

Contributions:

Team Member	Contribution	% Effort
Jonathan Huynh	Demo, Debugging, Synthesis, Simulation, Written Report	50%
Adam Godfrey	Verilog Code, Test Bench, Written Report	50%

Link To Video:

https://youtu.be/pE-wBqzjdYA