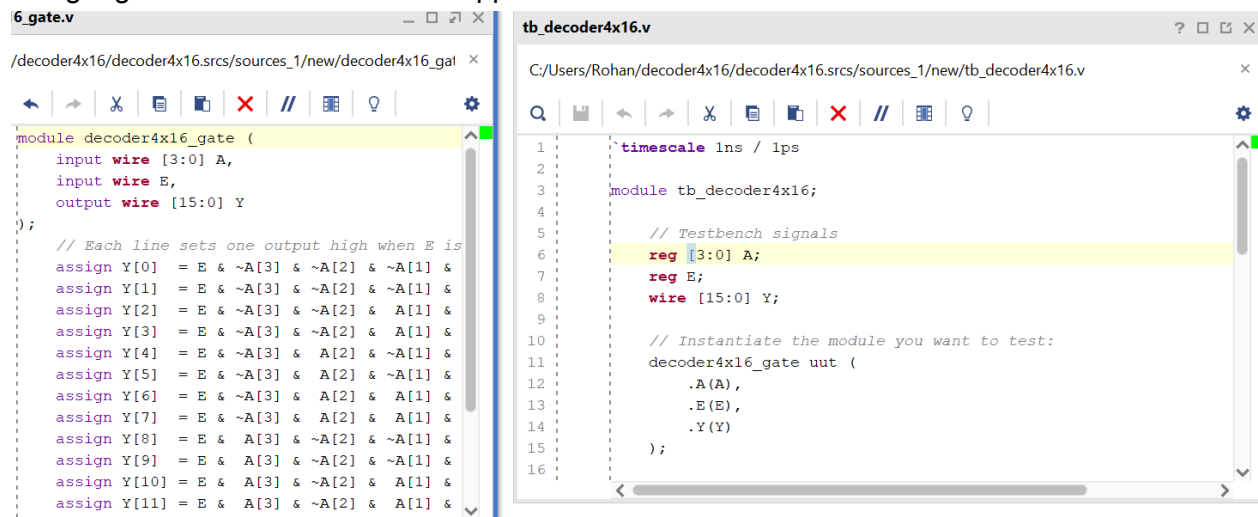


3300 Lab 2 Report

Rohan Walia
Parsa Ghasemi

Design: gate-level vs. behavioral snippets



```
6_gate.v
/decoder4x16/decoder4x16.srcs/sources_1/new/decoder4x16_gat

module decoder4x16_gate (
    input wire [3:0] A,
    input wire E,
    output wire [15:0] Y
);
    // Each line sets one output high when E is
    assign Y[0] = E & ~A[3] & ~A[2] & ~A[1] &
    assign Y[1] = E & ~A[3] & ~A[2] & ~A[1] &
    assign Y[2] = E & ~A[3] & ~A[2] & A[1] &
    assign Y[3] = E & ~A[3] & ~A[2] & A[1] &
    assign Y[4] = E & ~A[3] & A[2] & ~A[1] &
    assign Y[5] = E & ~A[3] & A[2] & ~A[1] &
    assign Y[6] = E & ~A[3] & A[2] & A[1] &
    assign Y[7] = E & ~A[3] & A[2] & A[1] &
    assign Y[8] = E & A[3] & ~A[2] & ~A[1] &
    assign Y[9] = E & A[3] & ~A[2] & ~A[1] &
    assign Y[10] = E & A[3] & ~A[2] & A[1] &
    assign Y[11] = E & A[3] & ~A[2] & A[1] &

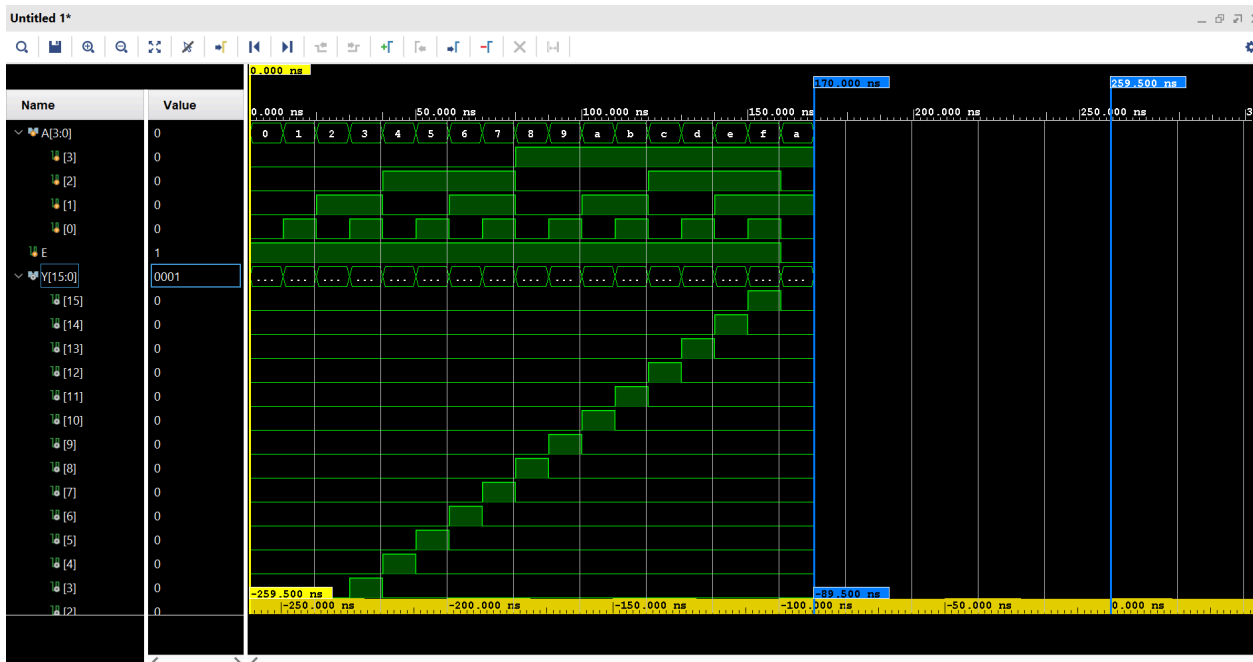
tb_decoder4x16.v
C:/Users/Rohan/decoder4x16/decoder4x16.srcs/sources_1/new/tb_decoder4x16.v

1 timescale 1ns / 1ps
2
3 module tb_decoder4x16;
4
5     // Testbench signals
6     reg [3:0] A;
7     reg E;
8     wire [15:0] Y;
9
10    // Instantiate the module you want to test:
11    decoder4x16_gate uut (
12        .A(A),
13        .E(E),
14        .Y(Y)
15    );
16
```

Comparison of gate-level (left) and behavioral (right) implementations of the 4-to-16 decoder. The gate-level design manually constructs each output using logic expressions, while the behavioral design uses a case statement.

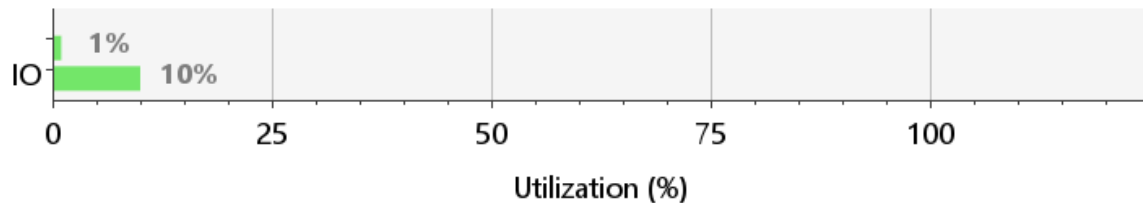
Simulation: testbench description, sample waveform

```
# run 1000ns
Starting test...
PASS: A=0000 E=1 â†’ Y=000000000000000001
PASS: A=0001 E=1 â†’ Y=000000000000000010
PASS: A=0010 E=1 â†’ Y=0000000000000000100
PASS: A=0011 E=1 â†’ Y=0000000000000001000
PASS: A=0100 E=1 â†’ Y=000000000000010000
PASS: A=0101 E=1 â†’ Y=000000000000100000
PASS: A=0110 E=1 â†’ Y=000000000001000000
PASS: A=0111 E=1 â†’ Y=00000000010000000
PASS: A=1000 E=1 â†’ Y=00000000100000000
PASS: A=1001 E=1 â†’ Y=00000001000000000
PASS: A=1010 E=1 â†’ Y=00000010000000000
PASS: A=1011 E=1 â†’ Y=00000100000000000
PASS: A=1100 E=1 â†’ Y=00010000000000000
PASS: A=1101 E=1 â†’ Y=00100000000000000
PASS: A=1110 E=1 â†’ Y=01000000000000000
PASS: A=1111 E=1 â†’ Y=10000000000000000
PASS: A=1010 E=0 â†’ Y=00000000000000000
All tests passed.
$finish called at time : 170 ns : File "C:/Users/Rohan/decoder4x16/decoder4x16.srcs/sources_1/new/tb_decoder4x16.v" Line 45
xsim: Time (s): cpu = 00:00:10 ; elapsed = 00:00:05 . Memory (MB): peak = 1652.516 ; gain = 14.270
INFO: [USF-XSim-96] XSim completed. Design snapshot 'tb_decoder4x16_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:14 ; elapsed = 00:00:22 . Memory (MB): peak = 1652.516 ; gain = 20.633
```



Implementation: resource utilization table & timing summary

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



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

There are no user specified timing constraints.

Contributions: each member's name + % effort

Rohan Walia: 50%

Parsa Ghasemi: 50%

Video link: <https://youtu.be/JPcla5uKTp4>