Lab Report 2 - Decoders and Muxes

by Xikang(Eric) Luo
SID: X676529

EECS120A - Section 24

Lab Partner:
Steven Strickland

Overview

In this Lab, we continue to use the operations learned in Lab1 to design and substantialize it in the Xilinx environment. We both use the schematical/ structural/ behavior coding to design the Mux and Decoder Module. It makes our understanding of both new modules we learned and the digital circuit methodology.

New Concepts

- Decoder: binary decoder is a combinational logic circuit that converts binary information from the n coded inputs to a maximum of 2n unique outputs.
- Mux: a multiplexer (or mux), also known as a data selector, is a device that selects between several analogs or digital input signals and forwards it to a single output line
- Behavior Modeling: A way to design the FPGA code, mainly focused to use a diversity of logic equations to give the description of the behavior how values interact with each other in the module.
- Structure Modeling: A way to design the FPGA code, mainly focused to use the module function to describe how circuit be connected in which kind of structure to give the digital process in module.

Analysis

1. Do the tablet, then transfer the tablet into logic Equation
2. Start the project, creat module, do the simulaiton
3. Create the behavioral test benches for Module
4. Debug Module by do simulation

<u>Part1:</u>

Logic:

d0 = E * A' * B' * C'
d1 = E * A' * B' * C
d2 = E * A' * B * C'
d3 = E * A' * B * C
d4 = E * A * B' * C'
d5 = E * A * B' * C
d6 = E * A * B * C'
d7 = E * A * B * C

Table:

E	A	B	C	d0	d1	d2	d3	d4	d5	d6	d7

0	x	x	x	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	1	0	0	0	0	0
1	0	1	1	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	0	1	0	0	0
1	1	0	1	0	0	0	0	0	1	0	0
1	1	1	0	0	0	0	0	0	0	1	0
1	1	1	1	0	0	0	0	0	0	0	1

Part 2:

Logic:

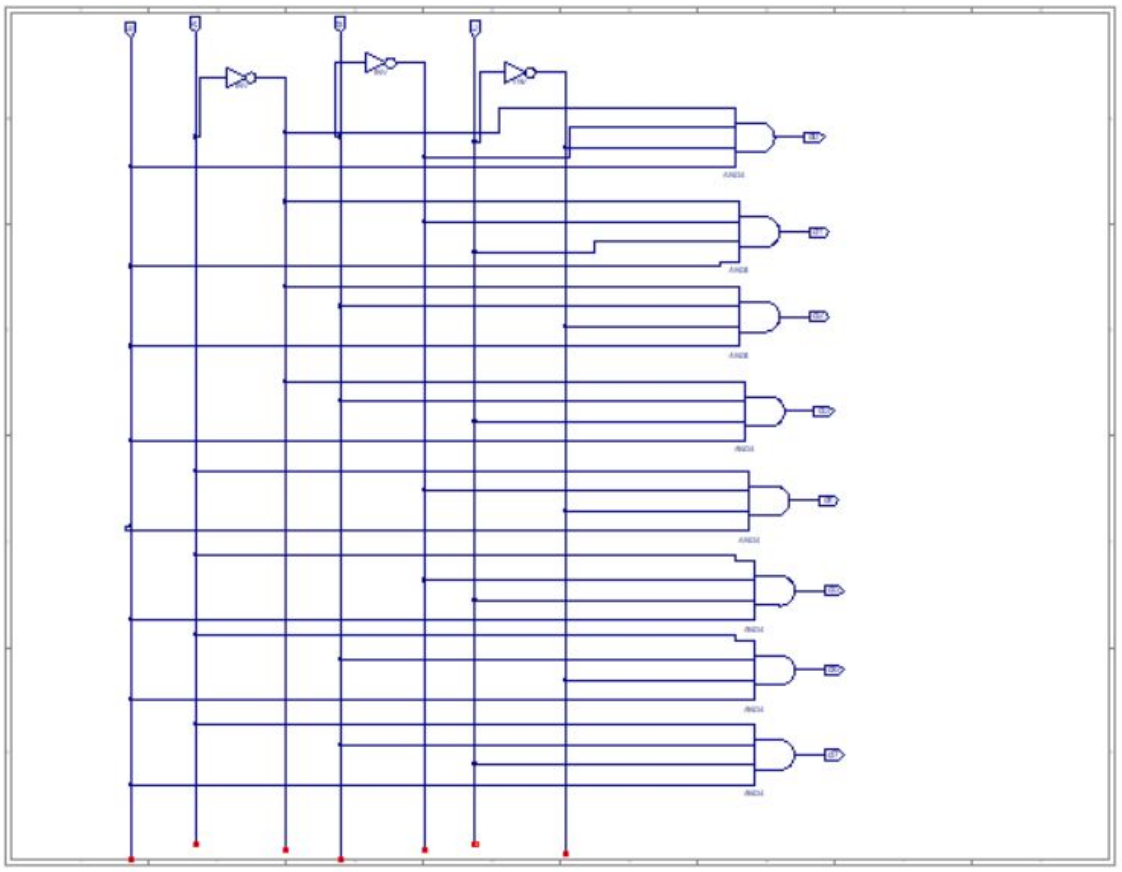
d = (S1' * S2' * i0) + (S1' * S2 * i1) + (S1 * S2' * i2) + (S1 * S2 * i3)

Table:

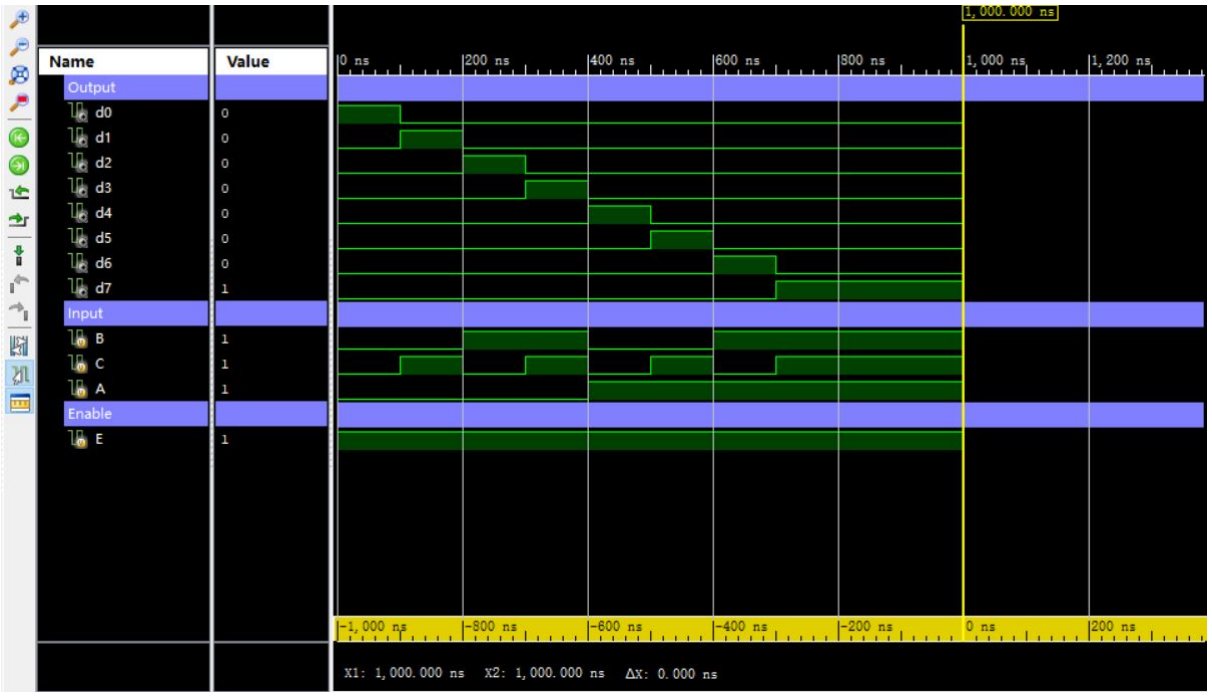
S1	S2	i0	i1	i2	i3	d
0	0	1	0	0	0	1
0	1	0	1	0	0	1
1	0	0	0	1	0	1
1	1	0	0	0	1	1

Records

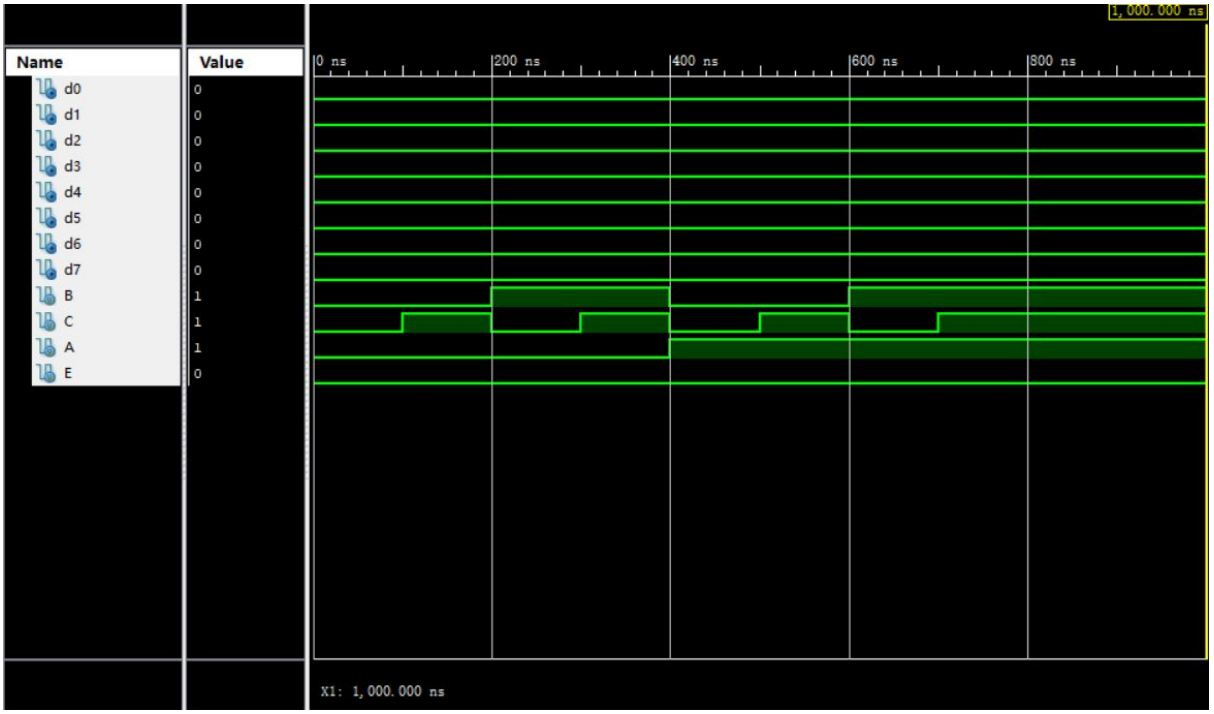
1）Pic1. Schematic and Design of the Gates



2) Simulation Test(Part 1A)



（E = 1）



（E = 0）

3) Simulation Part1B

（E = 1）

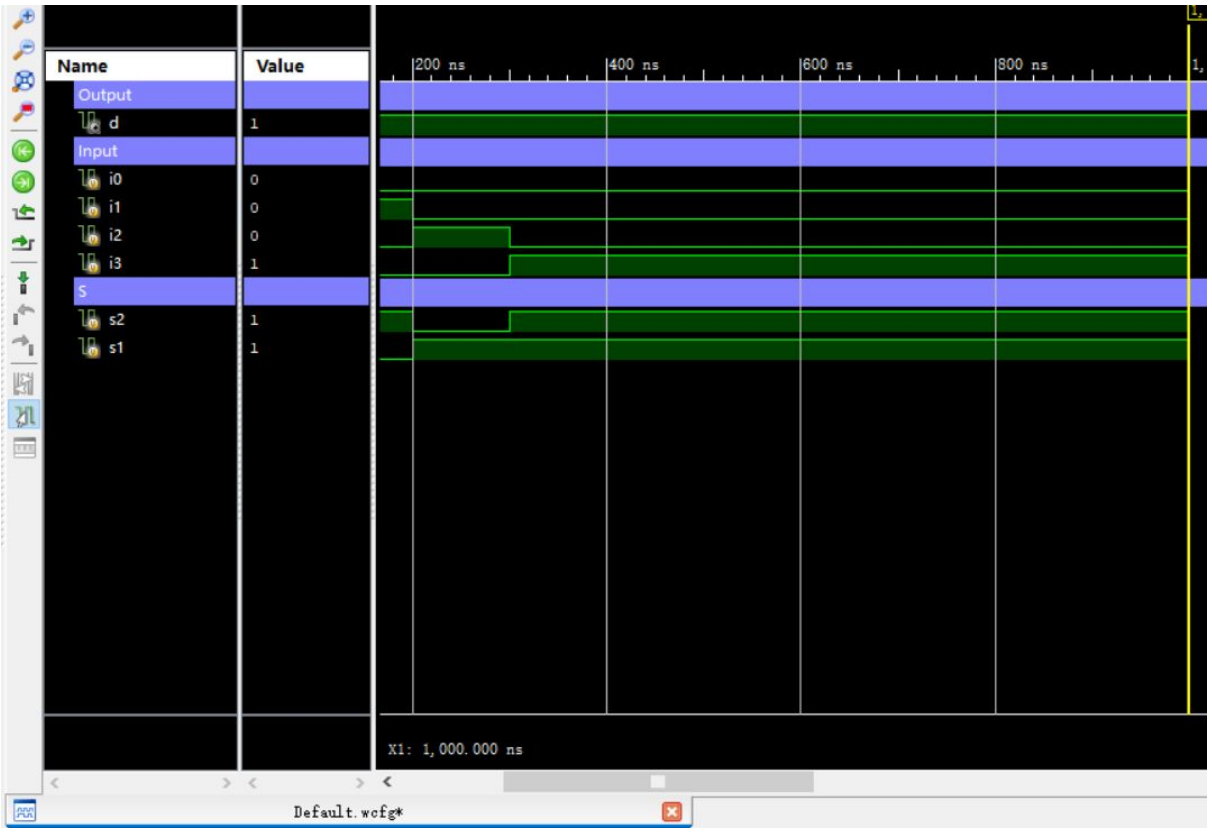
（E = 0）

4) Simulation Test Part1-C

(E = 1)

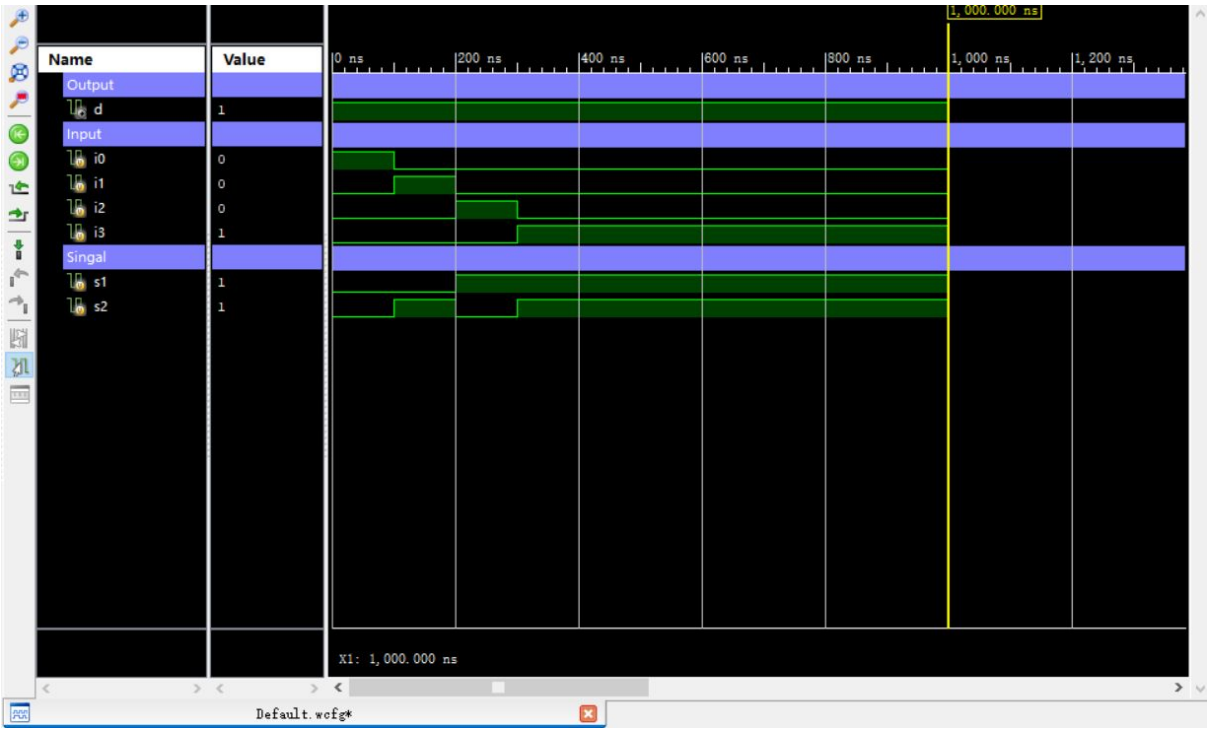
(E = 0)

5） Simulation Test Part2-A



6) simulation Part2-B

7) Simulation Part2-C



8) Decoder(Behavior)

```
module Module1C(
    input A,
    input B,
    input C,
    input E,
    output D0,
    output D1,
    output D2,
    output D3,
    output D4,
    output D5,
    output D6,
    output D7
    );

      assign D0 = ~A && ~B && ~C && E;
      assign D1 = ~A && ~B && C && E;
      assign D2 = ~A && B && ~C && E;
      assign D3 = ~A && B && C && E;
      assign D4 = A && ~B && ~C && E;
      assign D5 = A && ~B && C && E;
      assign D6 = A && B && ~C && E;
      assign D7 = A && B && C && E;
endmodule
```

9) Decoder（Sturtural）

```
module Decoder(
    input wire A,
    input wire B,
    input wire C,
    input wire E,
    output wire D0,
    output wire D1,
    output wire D2,
    output wire D3,
    output wire D4,
    output wire D5,
    output wire D6,
    output wire D7
    );


       AND4 c1(D0, E, ~A, ~B, ~C);
       AND4 c2(D1, E, ~A, ~B, C);
       AND4 c3(D2, E, ~A, B, ~C);
       AND4 c4(D3, E, ~A, B, C);
       AND4 c5(D4, E, A, ~B, ~C);
       AND4 c6(D5, E, A, ~B, C);
       AND4 c7(D6, E, A, B, ~C);
       AND4 c8(D7, E, A, B, C);
endmodule
```

10）MUX （Behaviour）

```
module Lab2Part2C(
    input i0,
    input i1,
    input i2,
    input i3,
    input s1,
    input s2,
    output d
    );

       assign d =
(~s1&&~s2&&i0)||(~s1&&s2&&i1)||(s1&&~s2&&i2)||(s1&&s2&&i3);
endmodule
```

11）MUX (Structural)

```
module Part2B(
    input i0,
    input i1,
    input i2,
    input i3,
    input s1,
    input s2,
    output d
    );

      wire m1;
      wire m2;
      wire m3;
      wire m0;
      wire d;

      AND3 a0(m0, i0, ~s1, ~s2);
      AND3 a1(m1, i1, ~s1, s2);
      AND3 a2(m2, i2, s1, ~s2);
      AND3 a3(m3, i3, s1, s2);
      OR4  o0(d, m0, m1, m2, m3);
endmodule
```

Discussion

Because of the similarity between the operations of Lab1 and Lab2. The whole process seems to be very simple and fluent at the beginning. But after simulation, there were two strange output as the bug confuse us for a longe time. They are the X state and Z state. All the module and test bench seem to be coded perfect, but these two strange results comes out.

The key to solve this is to get out what X and Z means. After a search, we found that

1) Z means that there are no wile connected to the output wire
2) X means that, although the output has been connected into the circuit, but it seems that there is no input to the output wire.

After that, we find two related small but important bugs in our code. It also tells us the importance of reading and understanding the message by compiler/simulator

Conclusion

We finally successfully accomplish the Lab2. In this process we

1) Go through the whole process from task to truth table to logical equation to circuit to make it into Verilog code
2) While the debug, we get the meaning of Z state and X state
3) We fully experienced the schematic/ structure/behavior FPGA Programing, and have had a perceptual feeling about their difference and advantages/disadvantages between/of them

Questions

1. What is a waveform?

A waveform is a representation of how values we care about varies with time. It gives us the opportunity to have a direct perception wether out code works well and accelerate our debugging process

2. What is a test bench?

A test bench is a Verilog code that allows us to provide a designed prepared set of stimuli that is portable across different simulators, or t. We can check if our code works at least in the case of our design. And we can also judge if can work in general case from our test cases.

3. Can we replace the 4-input AND gates in the circuit with the 2-input AND gates? If yes, how?

Yeah! We can, just use three 2-input AND gates. Like:

```
F = AND_1(AND_2(i1,i2), AND_3(i3,i4) );
```

F is the output of our compound AND4 module, and i1,i2,i3,i4 is four inputs.