SRAM Lab

1.0 Top level floorplan

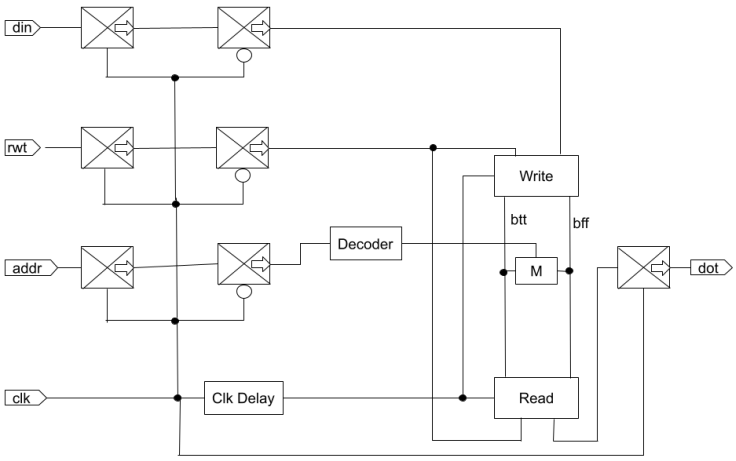

Figure 1: Top-Level Design

1.1 Notes

The design is similar to what was given in the lecture. The clk delay is what mainly changed is just a standard delay for both the read and write blocks.

2.0 Write Block

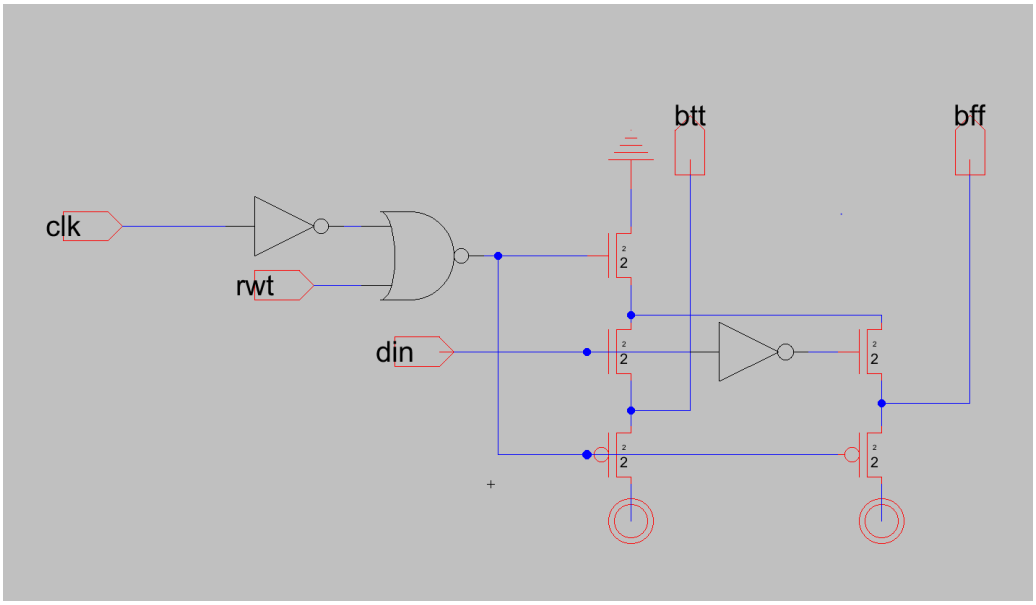

Figure 2: Write Block Sch

```
.subckt write1 btt bff dii rwt clk sz=100
Xiclk clkf clk inv

Xnor2 wrt rwt clkf nor2

Xtn1 i1 wrt gnd nn
Xtn2 btt dii i1 nn
xtp3 vdd wrt btt pp

Xidi diif dii inv

Xtn4 bff diif i1 nn
Xtn5 vdd wrt bff pp
.ends write1
```

Figure 3: Write Block Code

2.1 Notes

The write block uses NOR gate for timing with the clock and could be changed later for NAND gate. I could not get the redesign to work due to triggering on the wrong clock cycle.

3.0 Read Block

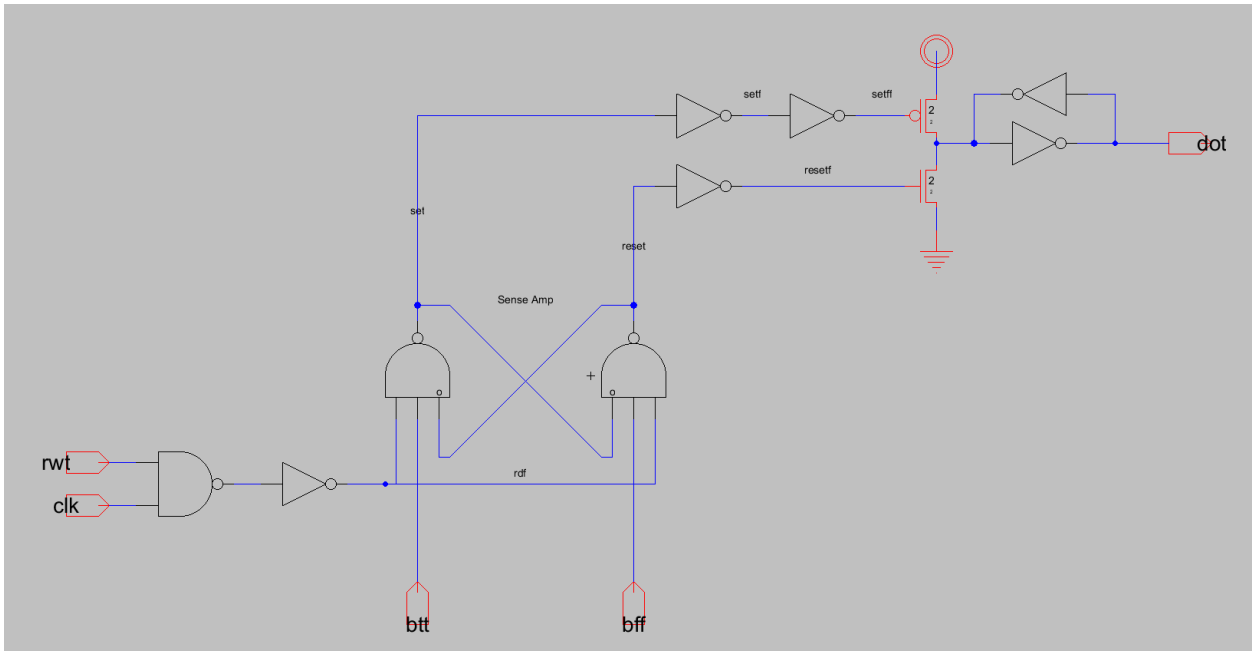

Figure 4: Read Block Sch

```
.subckt read1 btt bff out rwt clk
Xn0 rd rwt clk nnd2
Xn1 rdf rd inv

Xn2 reset btt set rdf nnd3 size =40
Xn3 set bff reset rdf nnd3 size =40

Xn4 resetf reset inv

Xn5 setf set inv
Xn6 setff setf inv

Xn7 vdd setff i1 pp
Xn8 i1 resetf gnd nn

Xn9 out i1 inv
Xn10 i1 out inv
.ends read1
```

Figure 5: Read Block Code

3.1 Notes

The read block has the senseamp within the subcircuit for better ease of following the nodes while coding. I tried messing with the inverters at the end of the sense amp and found that combination of delay for the set and reset line to be the best, which is essentially what was

given. When changing the inverters the timing of what signal arrived would not behave right and the output would be wrong.

4.0 Decoder

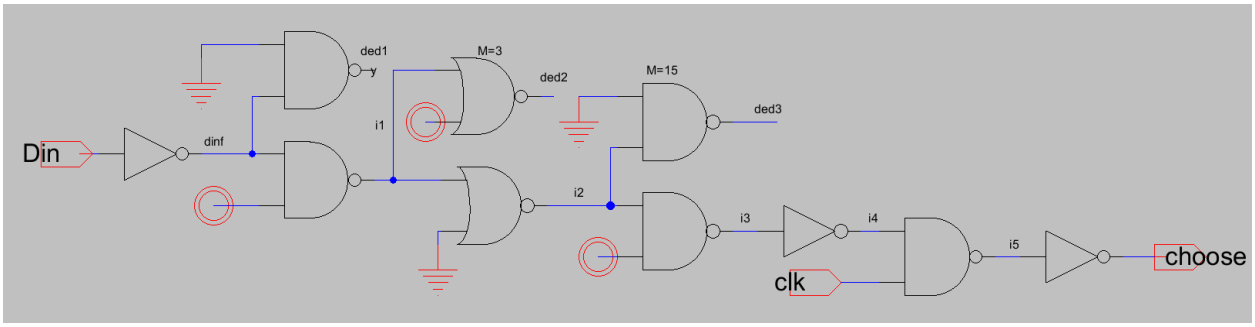

Figure 6: Decoder Block Worst Case

```
.subckt decodeModel choose Din clk sz=100
Xn0 dinf Din inv

Xn1 i1 dinf vdd nnd2
Xn2 ded1 dinf gnd nnd2

Xn3 i2 i1 gnd nor2
Xn4 ded2 i1 vdd nor2

Xn5 i3 i2 vdd nnd2
Xn6 ded3 i2 gnd nnd2

Xn7 i4 i3 inv

Xn8 i5 i3 clk nnd2

Xn9 choose i5 inv
.ends decodeModel
```

Figure 7: Decoder Block code

4.1 Notes

From the one on canvas that utilizes more NOR gates than NAND, the decoder made this year should be more efficient. This is the critical path which only uses one set of NOR gates. These can be designed out if needed.

5.0 Memory

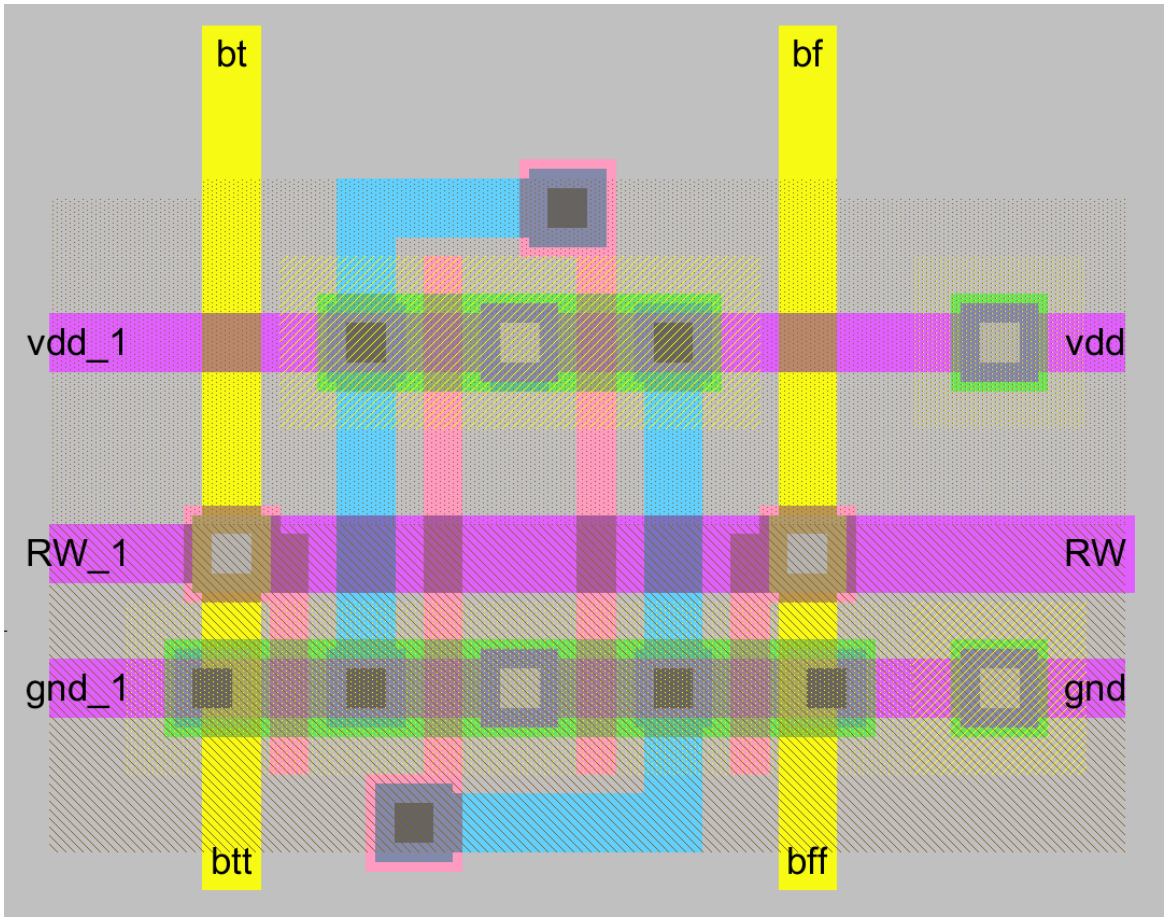

Figure 8 Memory Cell layout

5.1 Notes

The Layout is not quite as small as the one given in canvas and more shrinking can be done vertically I believe. This cell ended up being 58 x 45 in size. When trying to move lines closer together I was getting a lot of line overlap errors. When arraying, I also wanted to make sure the lines were long enough to connect to a full module.

6.0 Summary

I did not get every part functioning together; however I have created all the test finals to test each individual block. Below are the simulation results for the write, read, decoder, and memory blocks.

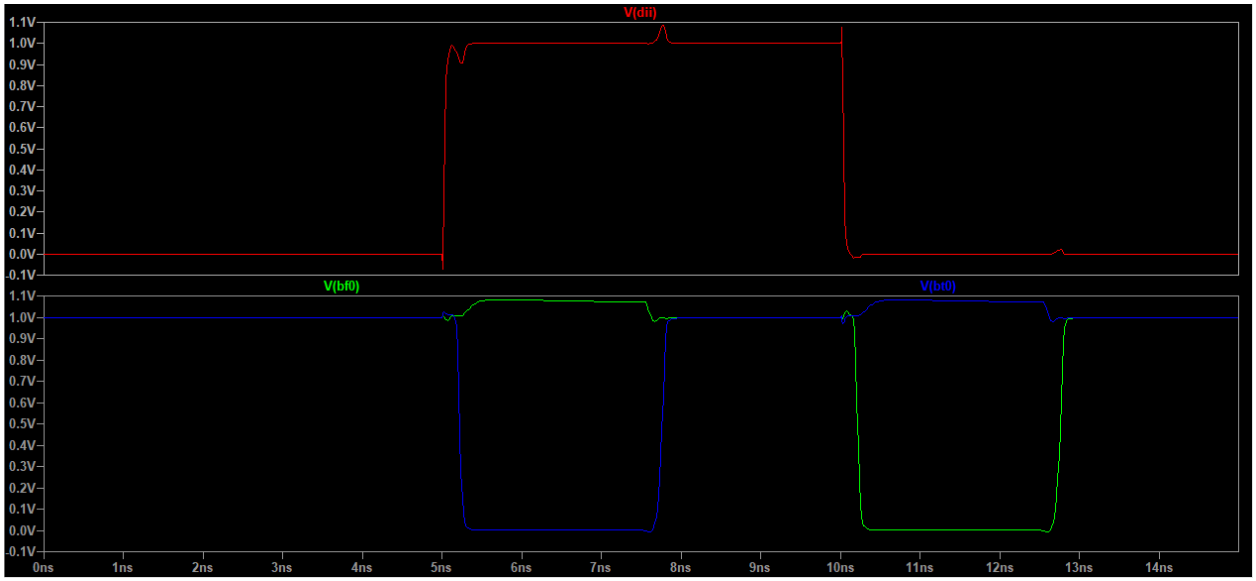

Figure 9 : Write Simulation

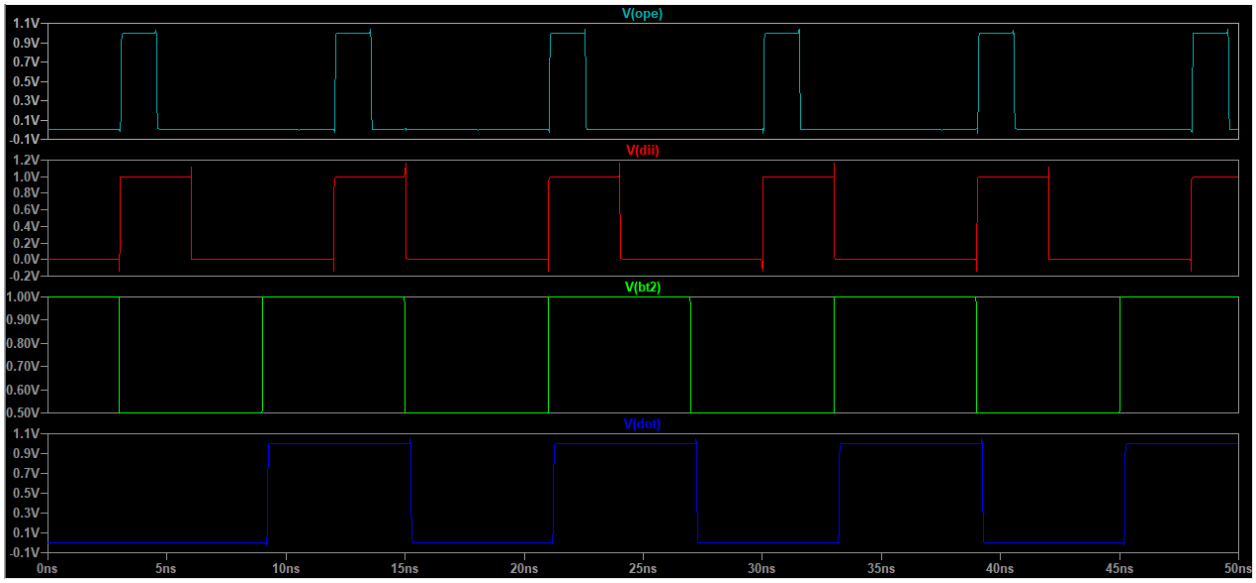

Figure 10: Read Simulation

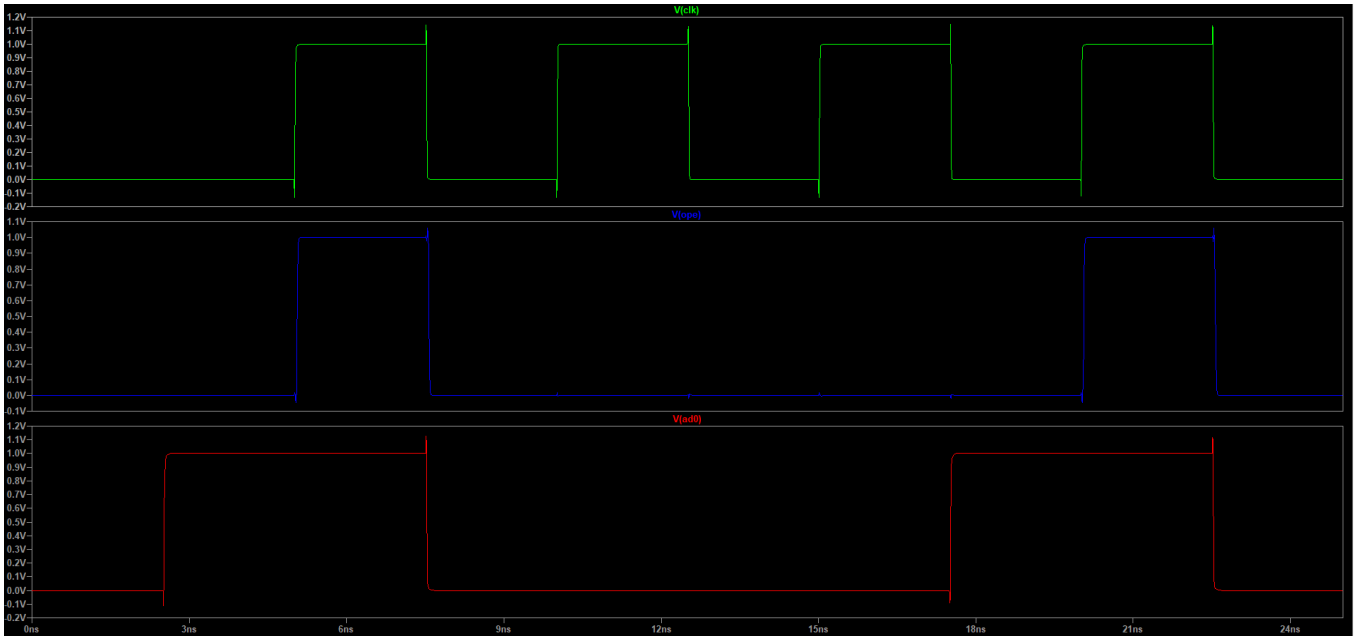

Figure 11: Decode simulation

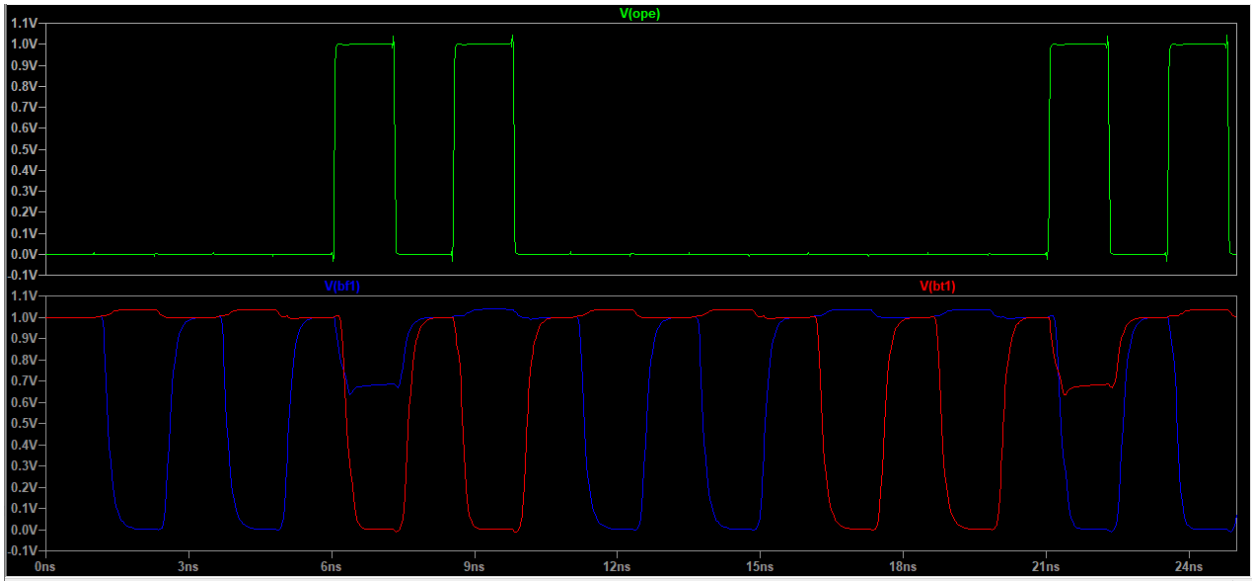

Figure 12: Mem simulation

7.0 Reflections

This is probably one of the hardest labs I have done in my career at Oregon State. The scope and time-frame of the project got me out of my comfort zone and forced me to learn a lot. Lt spice was something I had only used sparingly before this class and now I am doing complex simulations for blocks I never previously thought I would have to make. In terms of development as an Electrical Engineer, I think I have done the most growth within this class.