



EE457: Digital IC Design

Project 3 Report Cover Sheet

Due 12/07/2023

PROJECT TITLE: CMOS 4-bit shift register

Student Name: Dylan Kirdahy

Topics (Do Not change order of sections)	GRADES
Section1: Executive Summary (1/2 page)	Required
Section 2: Introduction and Background	/5
Section 3: Electric Circuit Schematics (transistors only)	/10
Section 4: LTSPICE Simulation for Schematic	/10
Section 5: IRSIM for Schematic	/10
Section 6: Electric Layouts (landscape mode for the final version and others to show all layout details)	/20
Section 7: LTSPICE Simulation for Layouts	/10
Section 8: IRSIM simulations for Layout	/10
Section 9: Compare LTSPICE Measurements for Schematic and Layout (<u>must provide comparisons between the two in table format</u>) Provide some delays, rise and fall times.	/10
Section 10: Measurements of <u>chip area and number of transistors</u> for the layout	/5
Section 11: Pathwave Advanced Design System	/10
Section 12: Conclusions and References	Required
Late Penalty (-5 points per day with prior 24 hour email notice) Max. of 2 days for late submission. Submit on Blackboard. -10 points for no Electric schematic and layout files	
TOTAL	/100

Do not erase: *Penalty rules: A) -5pts per day for late submission. After two days (12/9), a score of zero will be given, but you are still required to complete the report. B) -2pts for any violations and delays submitted after 8PM.

Contents

1	Executive Summary	1
2	Introduction and Background	1
3	Electric Circuit Schematics	3
4	LTspice Simulation for Schematic	6
5	IRSIM Simulation for Schematic	7
6	Electric Layouts	7
7	LTspice Simulation for Layout	11
8	IRSIM Simulation for Layout	13
9	Comparison of Schematic and Layout LTspice Measurements	13
10	Layout Measurements of Chip Size and Transistor Count	15
11	Conclusion	15
12	References	16

1 Executive Summary

The goal of this project was to build a 4-bit shift register in Electric. A shift register is a digital device which can shift bits across a number of outputs. In our case, there are 4 outputs, Q0 through Q3. Every clock cycle, the value of the bit stored at Q2 is shifted to Q3, the value of Q1 is shifted to Q2, Q0 is shifted to Q1, and the state of input D is stored in Q0. This is a very useful circuit for a number of applications, as we can control four outputs with only two inputs: A clock pin and a data pin. Moreover, multiple shift registers can be chained together to create a larger shift register with more outputs. For example, I once worked on a project in which 8-bit shift registers were chained together to control 64 LEDs with just two pins on a microcontroller.

For this project, the shift register shifts data on the positive clock edge. This is done by making the shift register out of four positive-edge-triggered D flip-flops, each flip-flop representing one bit. The shift register also has a reset pin, which is active low. When the reset pin is pulled low, all four outputs are reset to logic low.

The project involves designing the shift register as both a schematic and a layout in Electric. Both the schematic and layout are simulated in LTspice and IRSIM to confirm that the shift register design works properly. Measurements are also taken using LTspice to determine the rise times, fall times, propagation delay, and power draw of the designs. For this project specifically, the layout is simulated with parasitic resistances and capacitances. This makes the simulation much more realistic, since more of the physical properties of the design are taken into account.

2 Introduction and Background

The design for the shift register consists of a couple of layers of designs. In order to build the shift register, we must first build a positive-edge-triggered master slave D flip-flop, and in order to build the flip-flop, we must first build a 2-input NAND gate and a 3-input NAND gate. The design for the positive-edge-triggered master-slave D flip-flop is shown in Figure 1. The design consists of two 2-input NAND gates and four 3-input NAND gates. The flip-flop has inputs for the clock signal, data signal, and reset line. It outputs Q and Q prime, however we will only be using Q for the shift register. The flip-flop works like a set of latches; while the clock is low, the "master" latch is set and the "slave" latch is held, and while the clock is high, the "master" latch is held while the "slave" latch is set. Because of this setup, the output of the flip-flop only changes to reflect the input on the positive clock edge.

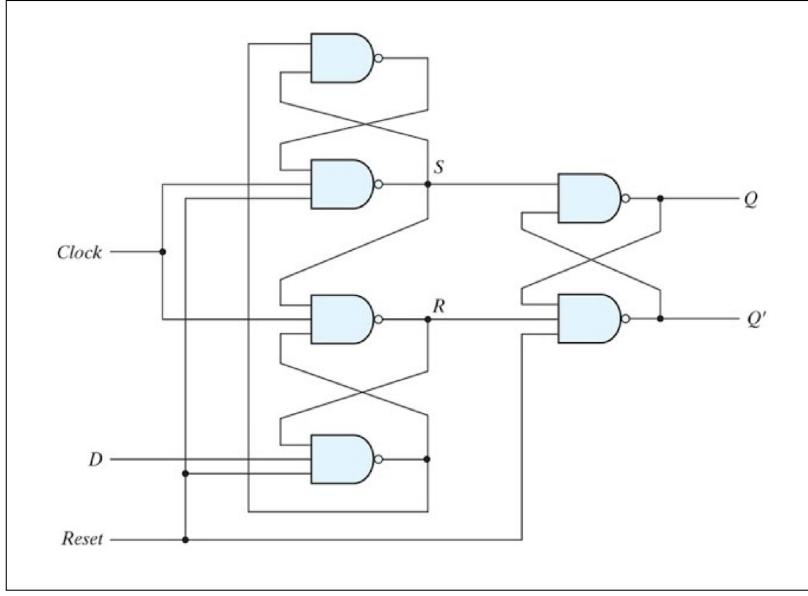


Figure 1: The design for the positive-edge-triggered master-slave flip-flop.

Now that we understand how the flip-flop works, we can understand how the shift register works. As shown in Figure 2, the 4-bit shift register is simply four positive-edge-triggered master-slave D flip-flops chained together. All of the reset pins of the flip-flops are connected together to create the reset line for the shift register. This reset line will reset all outputs to low when it is pulled low. The clock inputs are also tied together so that all of the flip-flops are triggered simultaneously at the positive clock-edge. The D input of the shift register is connected to the D input of the first flip-flop, and then the D input of each successive flip-flop is connected to the output of the previous flip-flop. This way, the data is shifted from left to right each positive clock edge.

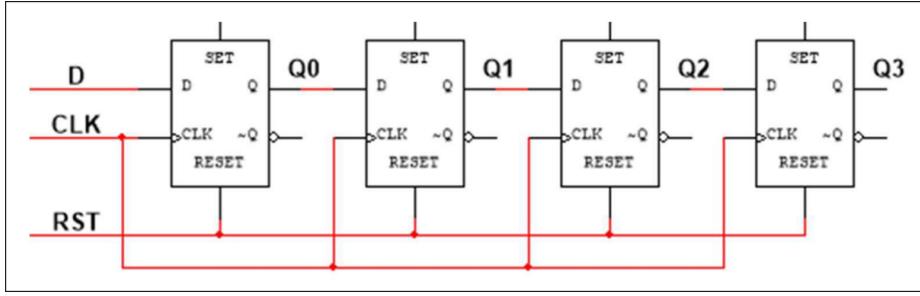


Figure 2: The design for the 4-bit shift register consisting of four positive-edge-triggered master-slave D flip-flops.

To give an example of how the shift register works, let us consider the following scenario. First, we pulse the reset pin low temporarily and then keep it high again. Now all outputs Q_0 , Q_1 , Q_2 , and Q_3 are low. We will set D high for the first positive-edge of the clock. After the positive-edge of the clock pulse, Q_0 is now high. After another positive-edge of the clock, the 1 stored at Q_0 is shifted to Q_1 . Then after another positive clock edge, Q_2 is high. Next positive clock edge, Q_3 is high. And then after another positive clock edge, the signal from D going low again has been shifted through the whole shift register and all of the outputs are low again. This example demonstrates

how the high pulse on D "shifts" through the shift register, making the jump every positive clock edge.

3 Electric Circuit Schematics

The basic building blocks for the shift register, as discussed in the previous section, are the 2-input and 3-input NAND gates. The schematic for the 2-input NAND gate is shown in Figure 3. It is a very simple static CMOS design with two PMOS transistors in parallel and two NMOS transistors in series to create the NAND logic. The schematic for the 3-input NAND gate is shown in Figure 4. The design is the same as for the 2-input NAND gate, except that there are three NMOS transistors and three PMOS transistors to support three inputs.

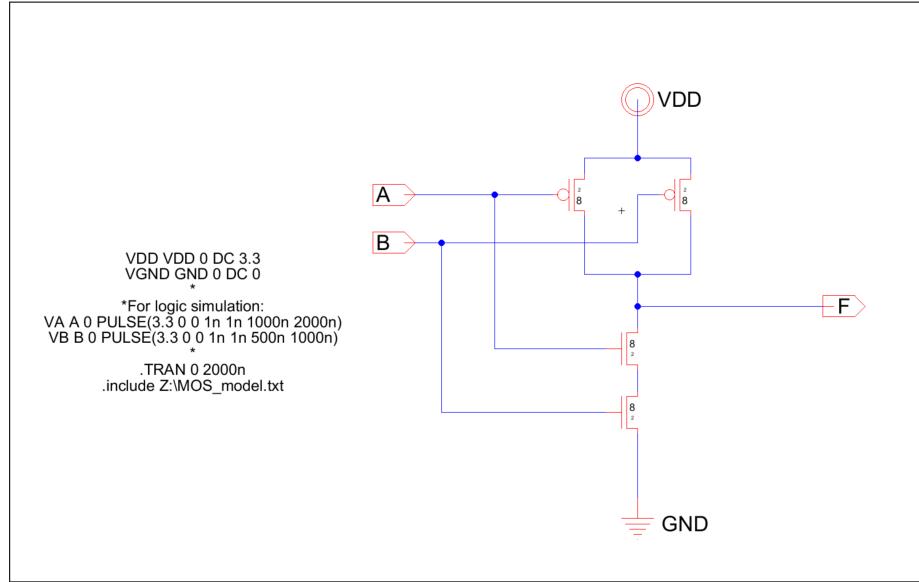


Figure 3: The schematic for the 2-input NAND gate, one component of the D flip-flops that make up the shift register.

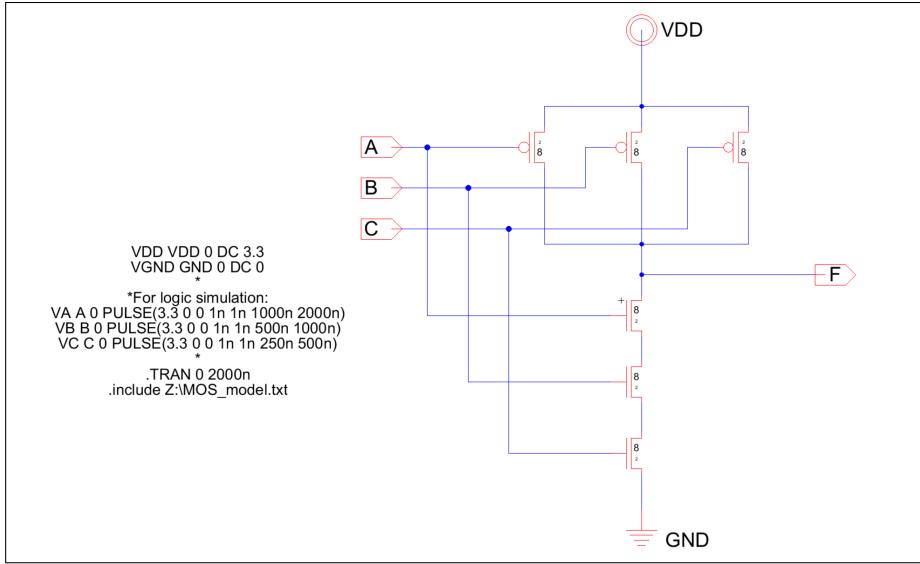


Figure 4: The schematic for the 3-input NAND gate, one component of the D flip-flops that make up the shift register.

Two 2-input NAND gates and four 3-input NAND gates are combined in the configuration shown in Figure 1 to create the schematic for the positive-edge-triggered master-slave D flip-flop that the shift register is based on. The schematic for the flip-flop is shown in Figure 5. The NAND gates can be clearly seen in the design. As described previously, the flip-flop design has inputs for the clock signal, data signal, and reset line and has outputs Q and Q prime. Crucially, the flip-flop is positive-edge-triggered due to the master-slave design that acts like two latches chained together.

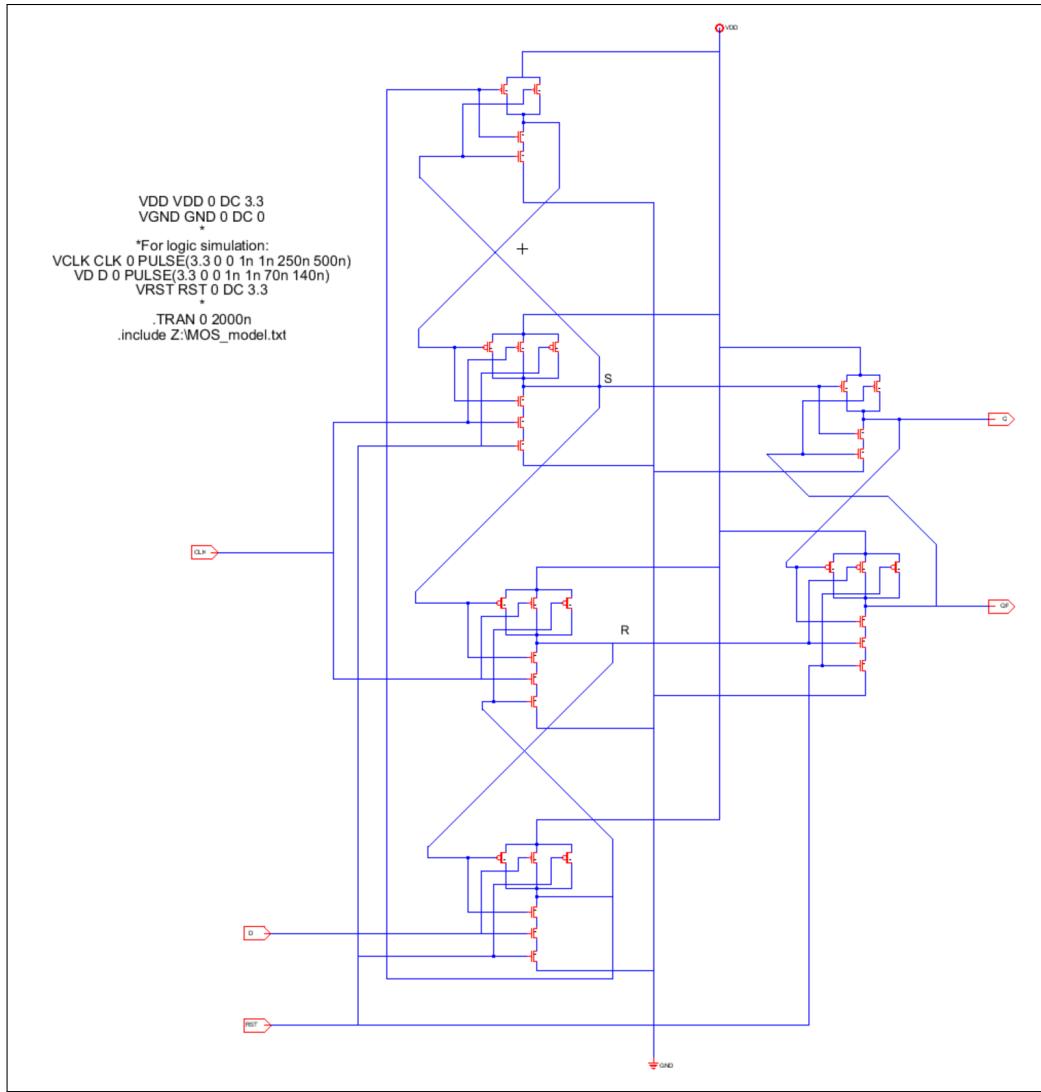


Figure 5: The schematic for the positive-edge-triggered master-slave D flip-flop.

The schematic for the shift register is shown in Figure 6. The schematic is simply four of the flip-flop schematics shown in Figure 5 chained together as described in the previous section.

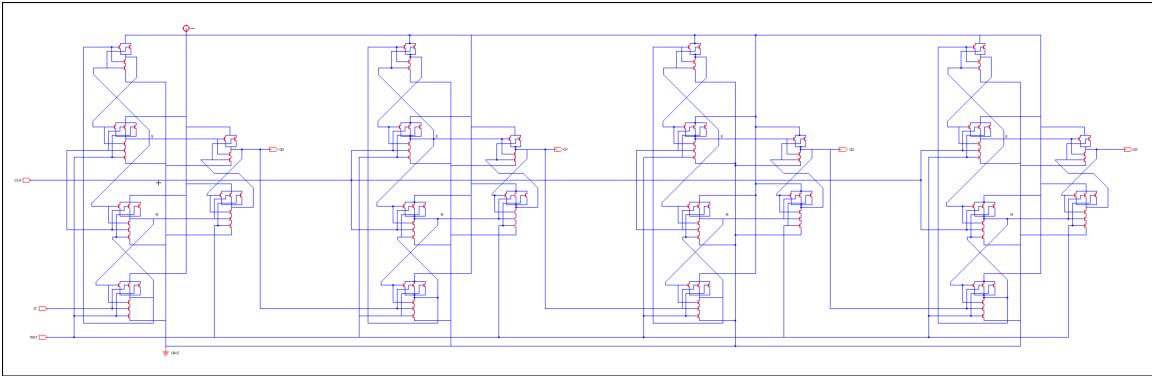


Figure 6: The schematic for the shift register.

4 LTspice Simulation for Schematic

The code for the LTspice simulation of the schematic is shown in Figure 7. The result of the simulation is shown in Figure 8. It is clear from the LTspice simulation how the positive pulse on D is shifted through the successive outputs of the shift register. This is exactly how the shift register should behave, as described in previous sections. At each positive edge of the clock, Q2 is shifted to Q3, Q1 is shifted to Q2, Q0 is shifted to Q1, and D is shifted to Q0.

```

VDD VDD 0 DC 3.3
VGND GND 0 DC 0
*
*For logic simulation:
VCLK CLK 0 PULSE(3.3 0 0 1n 1n 250n 500n)
VD D 0 PULSE(3.3 0 0 1n 1n 200n 300n)
VRST RST 0 PULSE(3.3 0 0 1n 1n 10n 2000n)
*
.TRAN 0 2000n
.include Z:\MOS_model.txt

```

Figure 7: The LTspice code for the simulation of the shift register schematic.

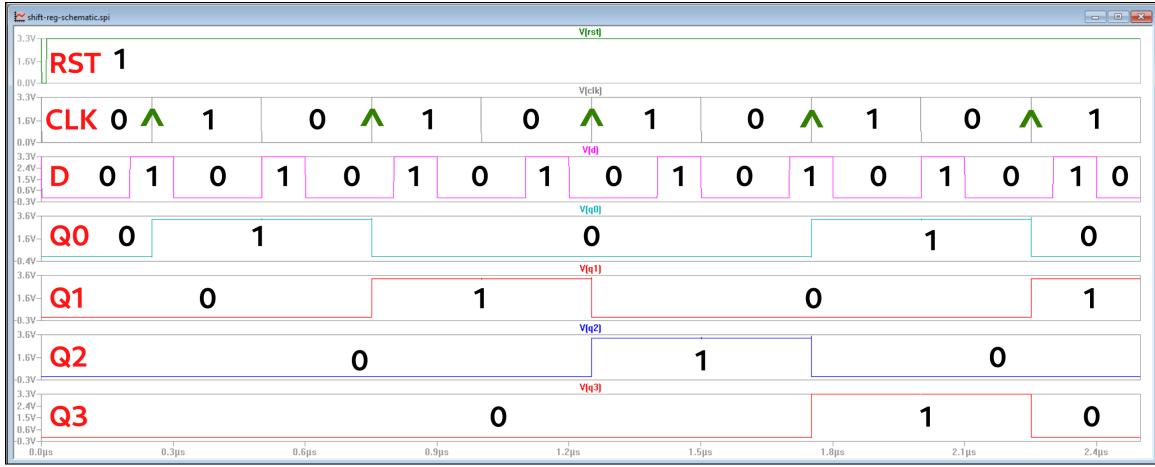


Figure 8: The LTspice simulation of the schematic for the shift register showing how a high pulse on D is shifted through the shift register at each positive clock edge.

5 IRSIM Simulation for Schematic

The IRSIM simulation of the schematic is shown in Figure 9. Just like the LTspice simulation in the last section, it is clear that the shift register is working, as the positive pulse on D is shifted through the four outputs of the shift register. One thing to note is that change in outputs is a bit delayed relative to the positive edges of the clock. This is simply a quirk of IRSIM.

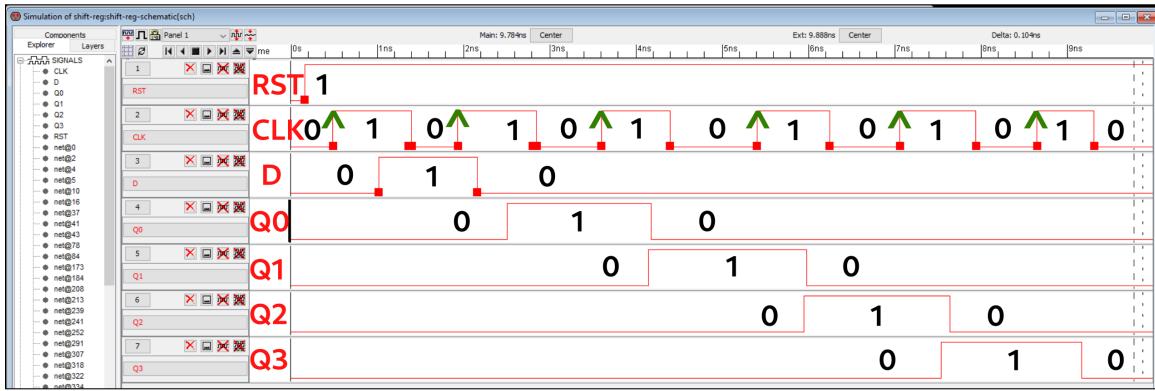


Figure 9: The IRSIM simulation of the schematic for the shift register showing how a high pulse on D is shifted through the shift register at each positive clock edge.

6 Electric Layouts

As described earlier, the positive-edge-triggered master-slave D flip-flop consists of 2-input and 3-input NAND gates. The layout for the 2-input NAND gate is shown in Figure 10. The layout for the 3-input NAND gate is shown in Figure 11.

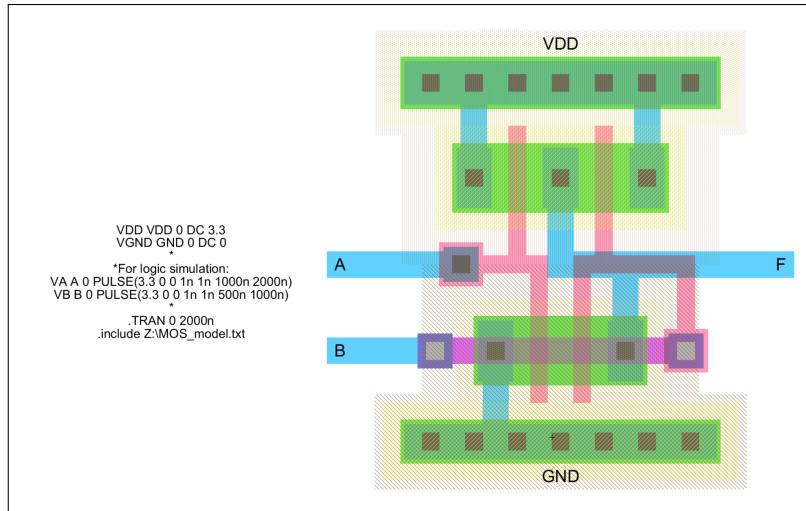


Figure 10: The layout for the 2-input NAND gate.

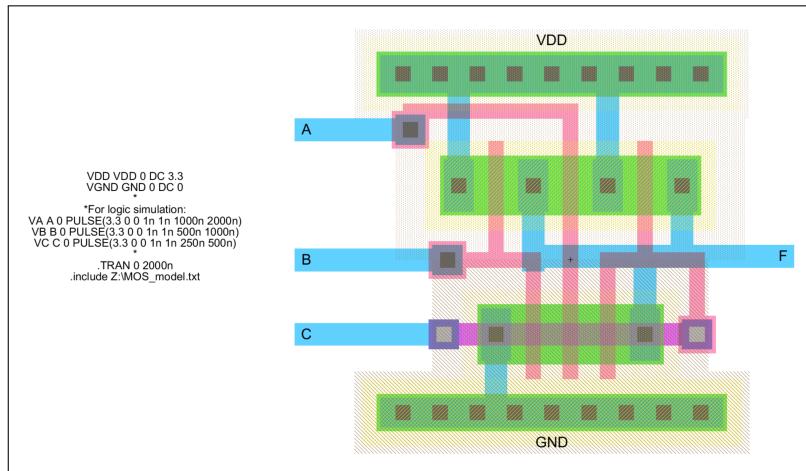


Figure 11: The layout for the 3-input NAND gate.

The layout for the positive-edge-triggered master-slave D flip-flop, consisting of the NAND gates shown in Figure 10 and Figure 11, is shown in Figure 12. The design consists of two 2-input NAND gates and four 3-input NAND gates and follows the configuration shown in Figure 1.

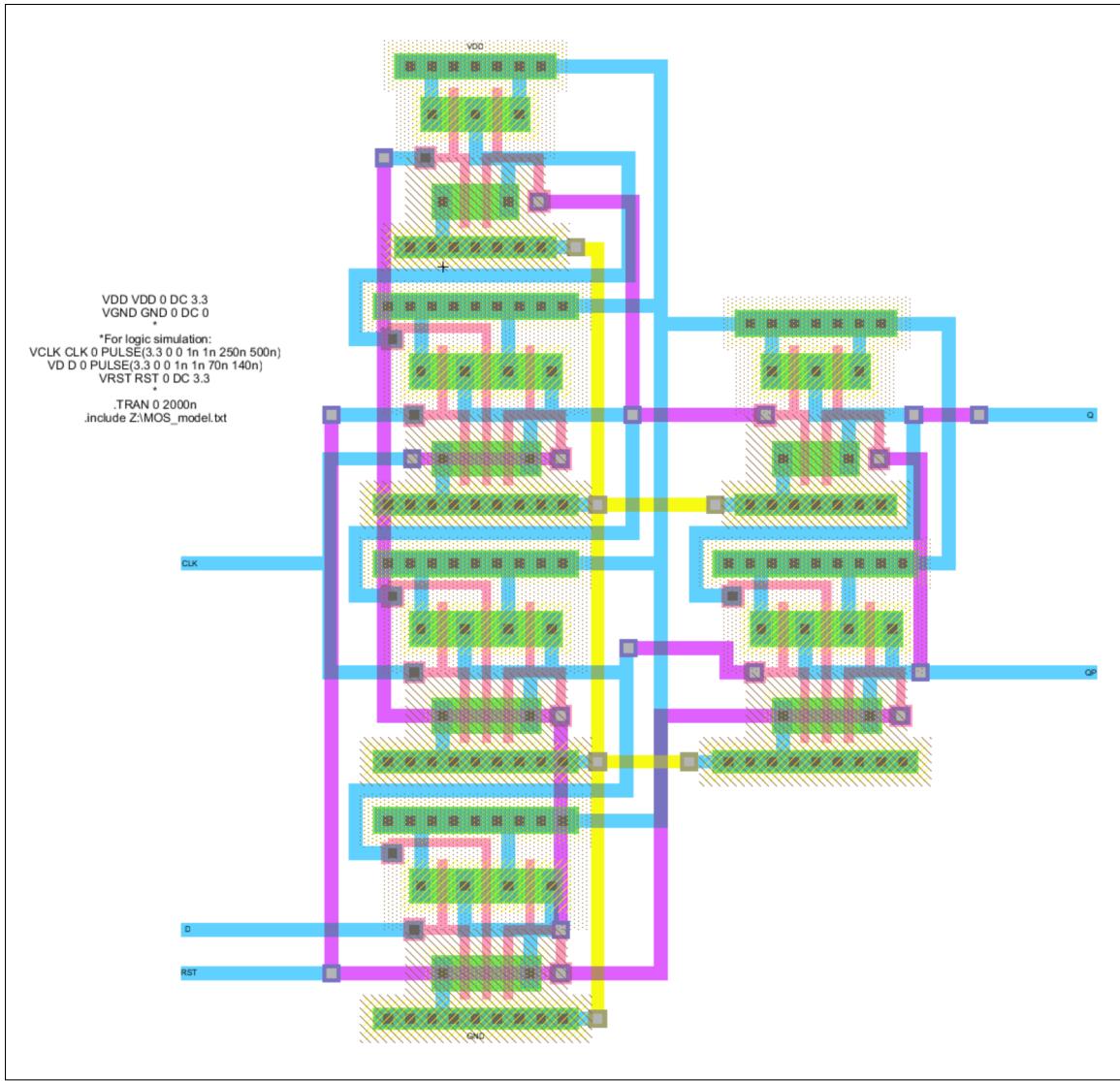


Figure 12: The layout for the positive-edge-triggered master-slave D flip-flop.

The layout for the 4-bit shift register, composed of four of the flip-flops in Figure 12, is shown in Figure 13. A closer look at the inputs is shown in Figure 14 and a closer look at the outputs is shown in Figure 15.

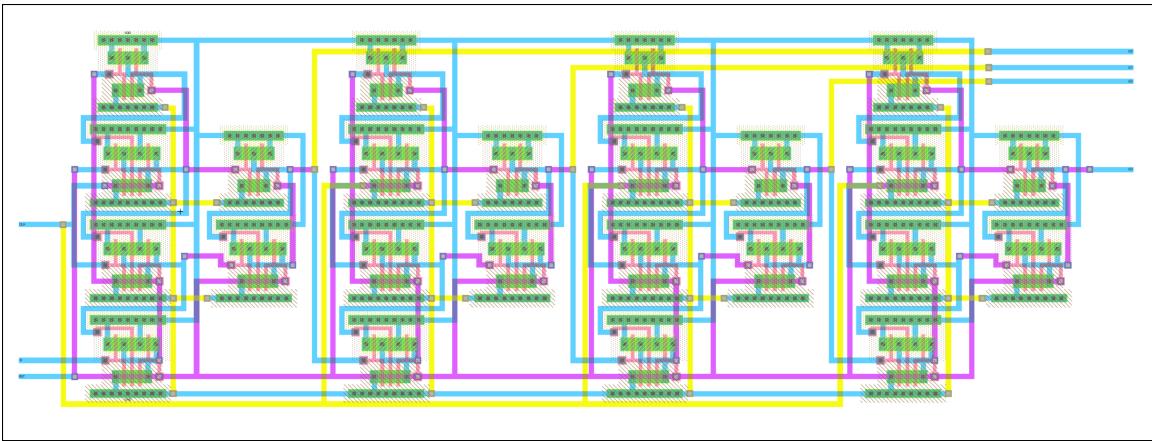


Figure 13: The layout for the 4-bit shift register.

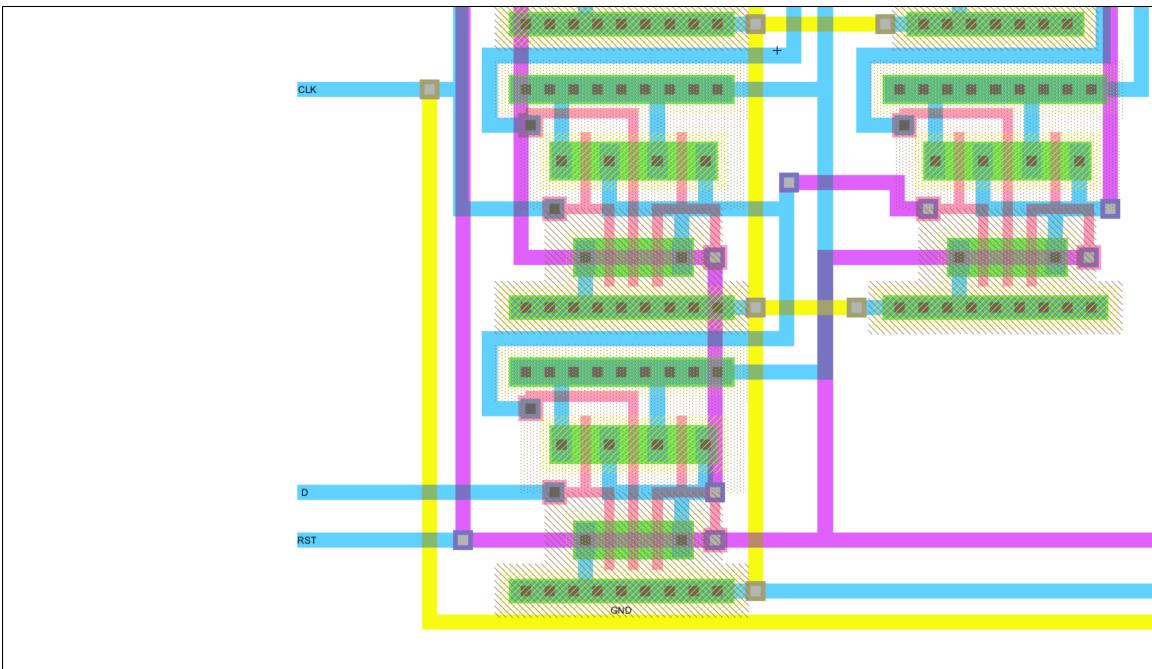


Figure 14: A close-up of the inputs of the layout for the 4-bit shift register.

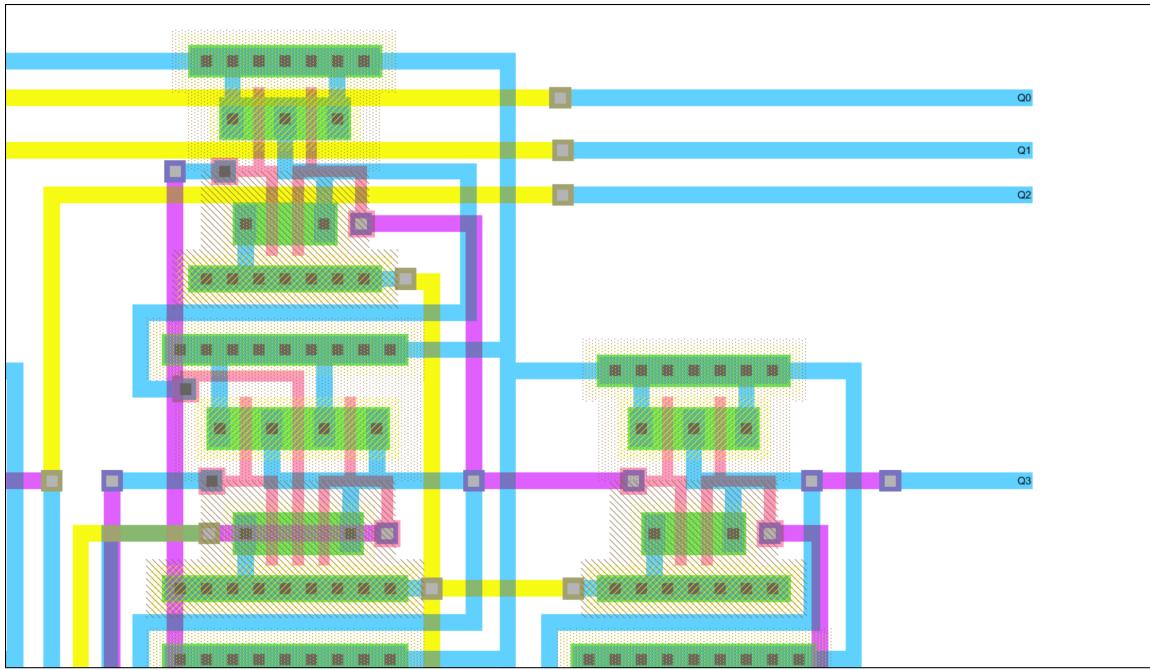


Figure 15: A close-up of the outputs of the layout for the 4-bit shift register.

7 LTspice Simulation for Layout

The code for the LTspice simulation of the layout is shown in Figure 16. The result of the simulation is shown in Figure 17. It is clear from the LTspice simulation how the positive pulse on D is shifted through the successive outputs of the shift register. This is exactly how the shift register should behave, as described in previous sections. At each positive edge of the clock, Q2 is shifted to Q3, Q1 is shifted to Q2, Q0 is shifted to Q1, and D is shifted to Q0.

```

VDD VDD 0 DC 3.3
VGND GND 0 DC 0
*
*For logic simulation:
VCLK CLK 0 PULSE(3.3 0 0 1n 1n 250n 500n)
VD D 0 PULSE(3.3 0 0 1n 1n 200n 300n)
VRST RST 0 PULSE(3.3 0 0 1n 1n 10n 2000n)
*
.TRAN 0 2000n
.include Z:\MOS_model.txt

```

Figure 16: The LTspice code for the simulation of the shift register layout.

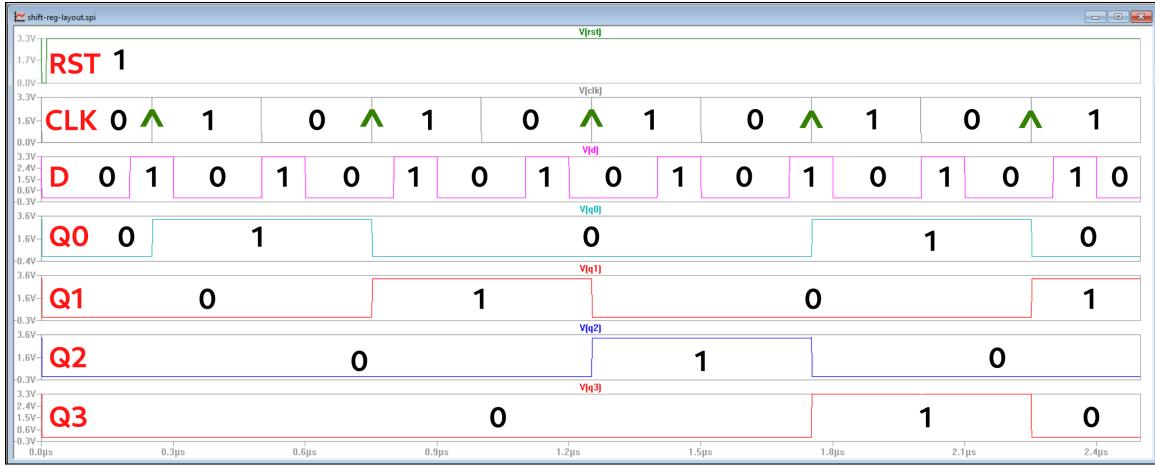


Figure 17: The LTspice simulation of the layout for the shift register showing how a high pulse on D is shifted through the shift register at each positive clock edge.

If you look very closely at Figure 17 as compared to Figure 8, the edges look slightly rounded in the simulation for the layout. This is in part due to the extraction of parasitics for the simulation of the layout. Electric allows for the extraction of parasitics to LTspice, meaning that the resistances and capacitances created by the geometry of the design are accounted for in the LTspice simulation of the layout. A snapshot of a portion of the extracted parasitics is shown in Figure 18.

```

C146 RST#85pin@373 polysilicon-1 0 0.144FF
C147 net#885#5pin@375 polysilicon-1 0 0.14FF
C148 Q3#0pin@380 polysilicon-1 0 0.108FF
C149 Q3#1pin@381 polysilicon-1 0 0.144FF
C150 Q3#2pin@382 polysilicon-1 0 0.113FF
C151 RST#83pin@383 polysilicon-1 0 0.113FF
C152 net#15pmos@3 poly-left 0 0.101FF
C153 net#7#2pmos@6 poly-left 0 0.101FF
C154 net#14#2pmos@9 poly-left 0 0.101FF
C155 Q0#0pmos@14 poly-right 0 0.101FF
C156 net#227#2pmos@19 poly-left 0 0.101FF
C157 net#298#16pmos@22 poly-left 0 0.101FF
C158 net#292#15pmos@25 poly-right 0 0.101FF
C159 Q1#14pmos@30 poly-left 0 0.101FF
C160 net#85#5pmos@35 poly-left 0 0.101FF
C161 net#6#6#16pmos@38 poly-left 0 0.101FF
C162 net#6#0#15pmos@41 poly-right 0 0.101FF
C163 Q2#14pmos@46 poly-left 0 0.101FF
C164 net#88#9#2pmos@51 poly-left 0 0.101FF
C165 net#89#1#16pmos@54 poly-left 0 0.101FF
C166 net#885#15pmos@57 poly-right 0 0.101FF
C167 Q3#14pmos@62 poly-left 0 0.101FF
** Extracted Parasitic Resistors ***
R0 Q0#0pmos@14 poly-right Q0#0pmos@14 poly-right##0 6.2
R1 Q0#0pmos@14 poly-right##0 Q0#1pin@68 polysilicon-1 6.2
R2 net#7#0 6.2
R3 net#7#0 net#7#1pin@2 polysilicon-1 6.2
R4 Q0#1pin@68 polysilicon-1 Q0#1pin@68 polysilicon-1##0 9.3
R5 Q0#1pin@68 polysilicon-1##0 Q0#1pin@68 polysilicon-1##1 9.3
R6 Q0#1pin@68 polysilicon-1##1 Q0#4pin@69 polysilicon-1 9.3
R7 Q0#4pin@69 polysilicon-1 Q0#4pin@69 polysilicon-1##0 9.688
R8 Q0#4pin@69 polysilicon-1##0 Q0#4pin@69 polysilicon-1##1 9.688
R9 Q0#4pin@69 polysilicon-1##1 Q0#4pin@69 polysilicon-1##2 9.688
R10 Q0#4pin@69 polysilicon-1##2 Q0#5pin@70 polysilicon-1 9.688
R11 Q0#5pin@70 polysilicon-1 Q0#2contact@46_metal-1-p_act 7.75
R12 net@14 net@14##0 6.717
R13 net@14##0 net@14##1 6.717
R14 net@14##1 net@14##1 contact@61_metal-1-polysilicon-1 6.717
R15 RST#2pin@7#1 polysilicon-1 RST#2pin@7#1 polysilicon-1##0 7.75
R16 RST#2pin@7#1 polysilicon-1##0 RST#2pin@7#1 polysilicon-1##1 7.75
R17 RST#2pin@7#1 polysilicon-1##1 RST#2pin@7#1 polysilicon-1##2 7.75
R18 RST#2pin@7#1 polysilicon-1##2 RST 7.75
R19 RST#2pin@7#1 polysilicon-1##2 RST#2pin@7#1 polysilicon-1##3 7.233

```

Figure 18: A snapshot of some of the extracted parasitics in the layout generated by Electric.

8 IRSIM Simulation for Layout

The IRSIM simulation of the layout is shown in Figure 19. Just like the LTspice simulation in the last section, it is clear that the shift register is working, as the positive pulse on D is shifted through the four outputs of the shift register. One thing to note is that change in outputs is a bit delayed relative to the positive edges of the clock. Again, this is simply a quirk of IRSIM.

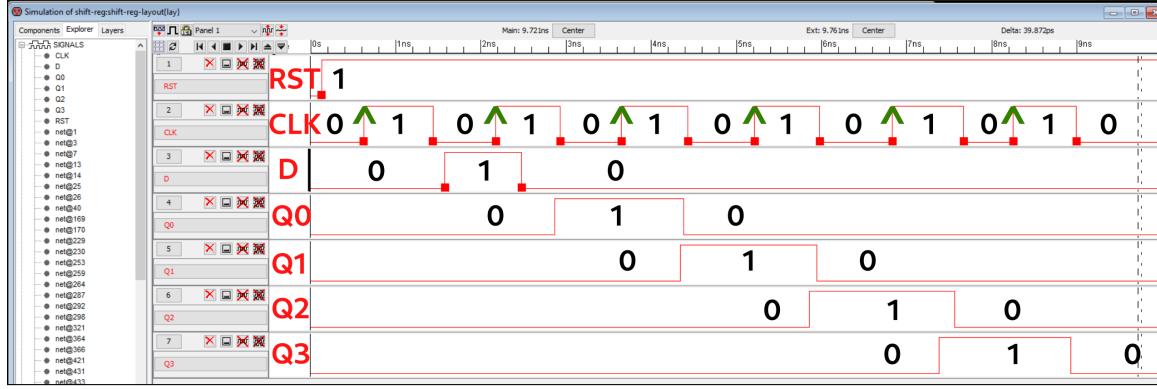


Figure 19: The IRSIM simulation of the layout for the shift register showing how a high pulse on D is shifted through the shift register at each positive clock edge.

9 Comparison of Schematic and Layout LTspice Measurements

In this section we will compare measurements taken for the schematic and layout of the shift register. The LTspice code used to take these measurements is shown in Figure 20. The code measures the rise time, fall time, propagation delay, and current draw of the design.

```

VDD VDD 0 DC 3.3
VGND GND 0 DC 0
*
*For logic simulation:
VCLK CLK 0 PULSE(3.3 0 0 1n 1n 250n 500n)
VD D 0 DC PULSE(3.3 0 0 1n 1n 160n 320n)
VRST RST 0 PULSE(3.3 0 0 1n 1n 10n 2000n)
*
.TRAN 0 1000n
.meas risetime TRIG v(Q0)=0.33 TD=0 rise=1 TARG v(Q0)=2.97 TD=0 rise=1
.meas fftime TRIG v(Q0)=2.97 TD=700n fall=1 TARG v(Q0)=0.33 TD=0 fall=2
.meas tpHL TRIG v(CLK)=1.65 TD=700ns rise=1 TARG v(Q0)=1.65 TD=0 fall=2
.meas tpLH TRIG v(CLK)=1.65 TD=0 rise=1 TARG v(Q0)=1.65 TD=0 rise=1
.meas propagationdelay param=(tpHL+tpLH)/2
.meas chip_current I_vdd AVG I(VDD)
.include Z:\MOS_model.txt

```

Figure 20: The LTspice code used to measure the rise time, fall time, propagation delay, and current draw of the design.

The results of the LTspice measurements are shown in Figure 21 for the schematic and Figure 22 for the layout.

```

D SPICE Error Log: Z:\8to1-mux-tg\shift-reg-schematic.log
Circuit: *** SPICE deck for cell shift-reg-schematic(sch) from library shift-reg

Vgnd: both pins shorted together -- ignoring.
Ignoring BSIM parameter XL
Ignoring BSIM parameter XW
Ignoring BSIM parameter XL
Ignoring BSIM parameter XW
Warning: Pd = 0 is less than W.
Warning: Ps = 0 is less than W.
Warning: Pd = 0 is less than W.
Warning: Ps = 0 is less than W.
Direct Newton iteration for .op point succeeded.

risetime=2.51023e-010 FROM 2.51638e-007 TO 2.51889e-007
falltime=1.18003e-010 FROM 7.51716e-007 TO 7.51834e-007
tph1=2.64462e-010 FROM 7.515e-007 TO 7.51764e-007
tph2=2.68534e-010 FROM 2.515e-007 TO 2.51769e-007
propagationdelay: (tph1+tph2)/2=2.66498e-010
chip_current: AVG(i(vdd))=-8.42033e-007 FROM 0 TO 1e-006

Date: Tue Dec 05 14:37:35 2023
Total elapsed time: 0.906 seconds.

tnom = 27
temp = 27
method = modified trap

```

Figure 21: The LTspice measurements for the schematic.

```

D SPICE Error Log: Z:\8to1-mux-tg\shift-reg-layout.log
Circuit: *** SPICE deck for cell shift-reg-layout(lay) from library shift-reg

Vgnd: both pins shorted together -- ignoring.
Ignoring BSIM parameter XL
Ignoring BSIM parameter XW
Ignoring BSIM parameter XL
Ignoring BSIM parameter XW
Direct Newton iteration for .op point succeeded.

risetime=9.25142e-010 FROM 2.52011e-007 TO 2.52936e-007
falltime=5.81168e-010 FROM 7.52132e-007 TO 7.52713e-007
tph1=8.87832e-010 FROM 7.515e-007 TO 7.52388e-007
tph2=9.61175e-010 FROM 2.515e-007 TO 2.52461e-007
propagationdelay: (tph1+tph2)/2=9.24503e-010
chip_current: AVG(i(vdd))=-2.99064e-006 FROM 0 TO 1e-006

Date: Tue Dec 05 14:06:38 2023
Total elapsed time: 2.557 seconds.

tnom = 27
temp = 27
method = modified trap
totiter = 3967
traniter = 3932
tranpoints = 1498
accept = 1374
<
rec=2.5ohms/sq
rec=0.078nhm/sq

```

Figure 22: The LTspice measurements for the layout.

The measurements are presented in Table 1 below. The values for the rise time, fall time, and propagation delay are about 4 times greater for the layout. The layout also uses about 3.5 times

more power.

Parameter	Schematic	Layout
Rise Time	251ps	925ps
Fall Time	118ps	581ps
t_{PHL}	264ps	888ps
t_{PLH}	269ps	961ps
t_p	266ps	925ps
Power Draw	$2.68\mu\text{W}$	$9.87\mu\text{W}$

Table 1: The measurement results for the schematic and layout of the shift register design.

10 Layout Measurements of Chip Size and Transistor Count

The measurements for the chip dimensions and transistor count of the layout are shown in Table 2.

Parameter	Layout
Chip Dimensions (λ)	$641\lambda \times 236\lambda$
Chip Dimensions (μm)	$112.2\mu\text{m} \times 41.3\mu\text{m}$
Chip Area(λ^2)	$151.28*10^3\lambda^2$
Chip Area (μm^2)	$4.6310^3\mu\text{m}^2$
Transistor Count	128

Table 2: The measurements for the layout of the shift register.

11 Conclusion

In this project I designed a schematic and layout for a 4-bit shift register in Electric. I wrote spice code to simulate the schematic and layout in order to ensure the proper functioning of the design and to take measurements of the performance of the design. I used IRSIM as well to confirm the function of the design digitally. After obtaining measurements through LTspice, I was able to compare the performance between the schematic and layout. I also took measurements of the dimensions of the layout and the transistor count.

Overall, I learned a lot about digital IC design from the project. It was very interesting to design a flip-flop that functions properly and then put multiple flip-flops together in order to create a shift register. I have used shift registers before in my own projects, so it was especially interesting to me to build one and see how it works. It was a bit of a challenge to do some of the routing for power and data between the flip-flops, but overall I am happy with the size and performance of my design.

12 References

- [1] EE 457 Lectures 1, 2, 3, and 4h
- [2] https://cmosedu.com/videos/electric/tutorial3/electric_tutorial_3.htm
- [3] https://cmosedu.com/videos/electric/tutorial4/electric_tutorial_4.htm
- [4] <https://siliconvlsi.com/master-slave-d-flip-flop/>
- [5] [https://en.wikipedia.org/wiki/Flip-flop_\(electronics\)](https://en.wikipedia.org/wiki/Flip-flop_(electronics))
- [6] https://en.wikipedia.org/wiki/Shift_register