Miniproject 2: Making a D-Flip Flop with a Complementary Set-Reset Logic Latch using XSCHEM and Magic

Kristtiya Guerra

March 2021

1 Introduction

The purpose of this project is to create a working D-flip flop that implements a set-reset latch. I used the layout provided by Massino Antonio Sivilotti [1]. Using Xschem and Magic, I attempted to recreate one of his proposed circuits for an edge-triggered CSRL D-flip flop.

Link to view repository:
https://github.com/Kristtiya/MAD_VLSI/tree/main/MiniProject2-CSRLDFF

2 Schematic of the CSRL D-Flip Flop and Shift Register

Sivilotti's proposed layout allows for a gate to become edge-sensitive, meaning depending on the layout, the gate will become active at the rise or fall of a signal, typically a clock edge. The schematic I used in this lab is shown in figure 1.

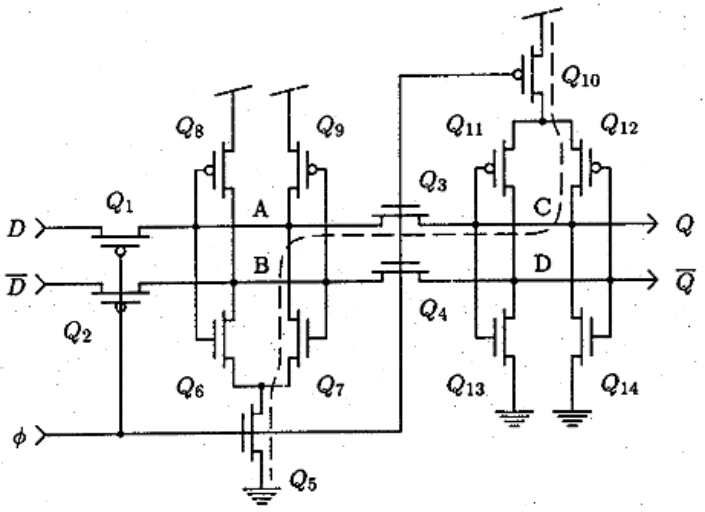

Figure 1: M. A Sivilotti's proposed edge-sensitive CSRL latch design

The schematic has two complementary stages- the loading/leader, and the state-holding stage/follower The Reset stage is initiated when the clock is low and goes back high while a new input is at D. M_1 and M_2 act as access devices for the D and $\bar{D}$ input.They lead to a pair of cross-coupled transistors. The M_1 and M_2 access transistors are gated by the clock so when it goes low D and $\bar{D}$ are allowed to flow into the inverters. When the clock goes high, the M_1/M_2 access devices are closed. At the same time opening the

access devices M_3 and M_4, giving access to the follower stage. This loads the data that was in the previous set of cross-coupled inverters into the next set of inverters. When the clock returns to low, the access device M_3/M_4 shut, allowing the data in the follower stage to remain in place. The pull-up transistor amplifies weak 1's allow for more accurate outputs at Q and $\bar{Q}$.

In order for the circuit to run properly (in theory), I need my access devices to have a weaker strength than their associated pull-up and pull-down transistors. A typically chosen strength ratio is around 4. This means the strength ratio between the pull-down transistor, Q_5, and access devices, M_1 and M_2, have to have a strength ratio of 4:1. The same goes for the strength ratio between the pull-up transistor, M_{10} and access devices, Q_3 and Q_4.

The ratio can be calculated using the relationship:

$$\frac{(W/L)_{Q_a}}{(W/L)_{Q_b}} = 4 \tag{1}$$

Where Q_a would be the pull-up/pull-down transistor and Q_b would be one of the transistors in the access devices. I chose to elongate my access devices as it was the most convenient mathematically and made it easier to implement in the Magic layout. I made the length of my 4 access device transistors $0.6um$.

2.1 Creating schematics in Xschem

2.2 D-FlipFlop

For Xschem, I followed the layout of figure 1 exactly to ensure that I built it properly.

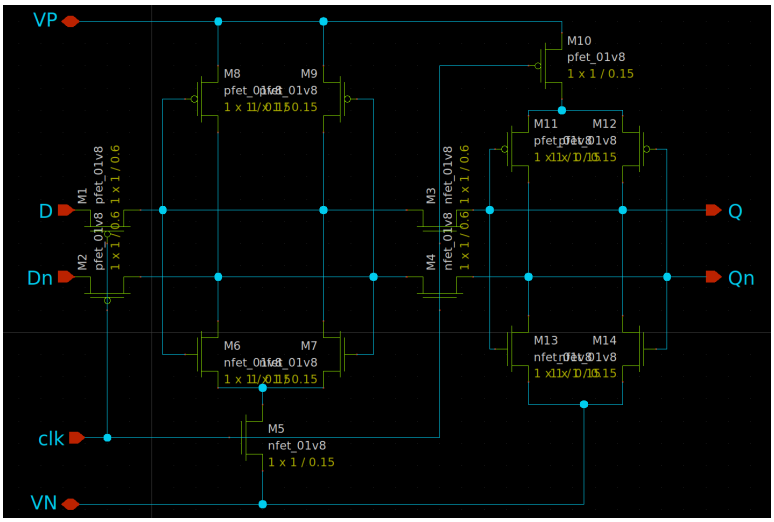

Figure 2: Schematic for CSRL D-Flip Flop

I then created the symbol. The design is a rectangle with the logic inputs on the left, ground (VN) and clock (clk) inputs on the bottom, with clock having a triangle indication, power (VP) on top, and outputs Q and Q_n on the right side.

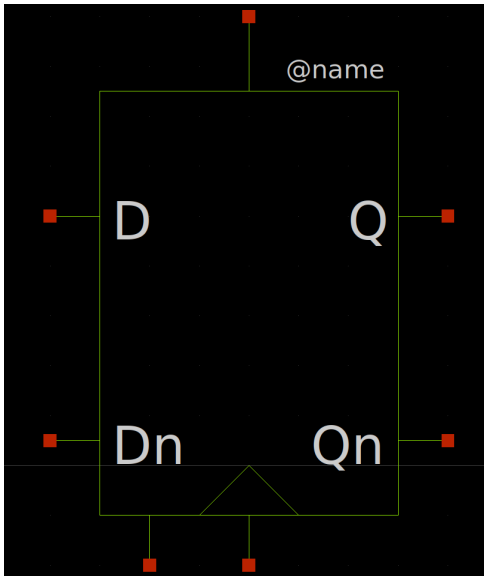

Figure 3: Symbol Designed for the CSRL D-Flip Flop

2.3 Shift Register

The shift register is composed of an inverter leading into 4 D-flip flops in series. At each rising edge of the clock, the shift register should take in the data at it's PMOS access point, input D, and output that value at the fall of the clock. It is called a shift register because at each clock cycle, value previous to a flip flop becomes its new output and the original output of the said flip flop becomes the output of the following flip flop, thus "shifting" the values.

I created a schematic for my shift register separate from my testbench as when comparing the netlist between my schematic and layout for the shift registers there were conflicting nets due to voltage inputs. The figure below shows the schematic for my shift register with labeled inputs and outputs.

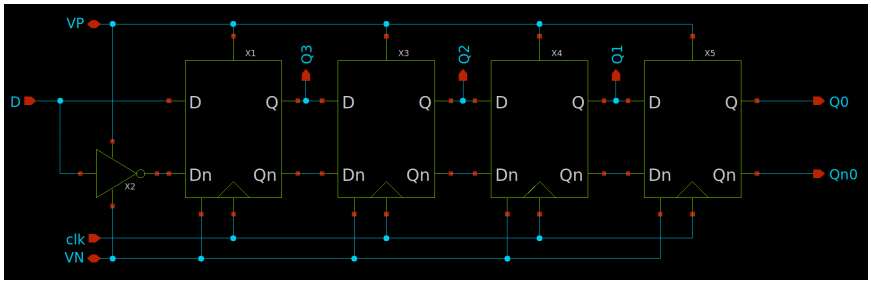

Figure 4: Layout of Shift Register

2.4 Testing the Circuit

At the moment, my shift register does not work properly. The current testbench being used does a transient simulation and features 3 voltage sources: A VDD source, labeled VP, a clock source that has a high and low time of 15 ns and a cycle of 32ns, and a D input that is a single pulse that lasts around 40 seconds and is offset by around 19ns.

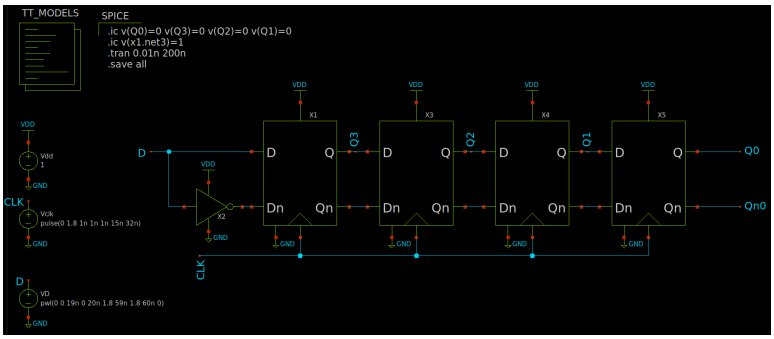

Figure 5: Layout of test bench for the Shift Register

2.4.1 Simulation Results

As shown in the results for the simulation, the my shift register does not work properly. The first register peaks to 1V when the clock rises, but it does not hold state, making it unable to pass any values to the next flip flop.

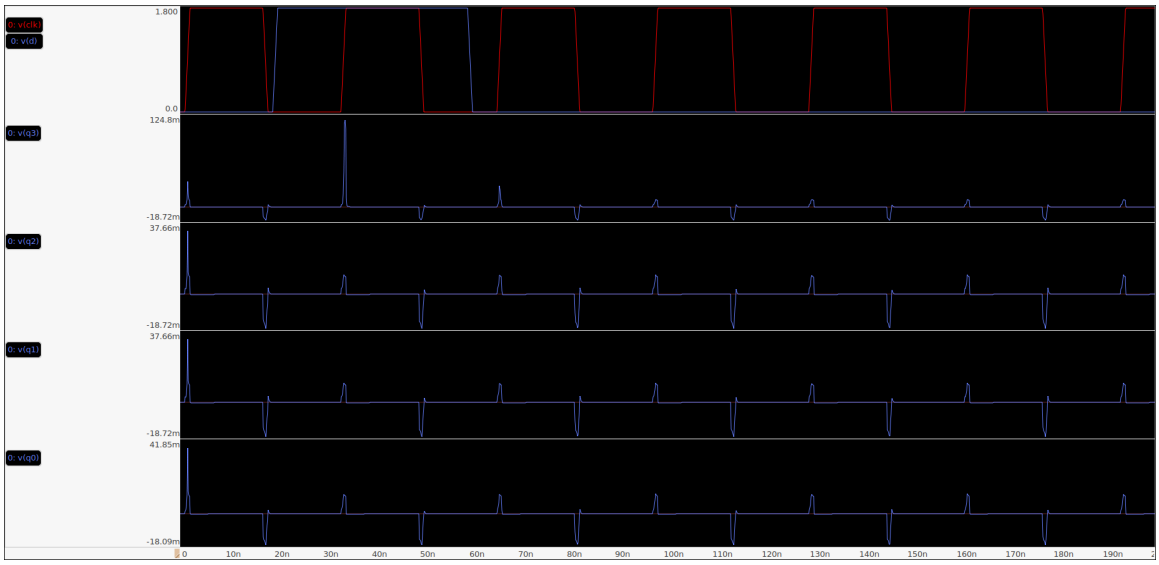

Figure 6: Waveform from simulated test

3 Creating Layouts in Magic VLSI

3.1 Inverter

First, I built the inverter. Although I already had one built from the previous project, I wanted to make a new one that allows the inverter to fit in series with the D-flip flops in series.

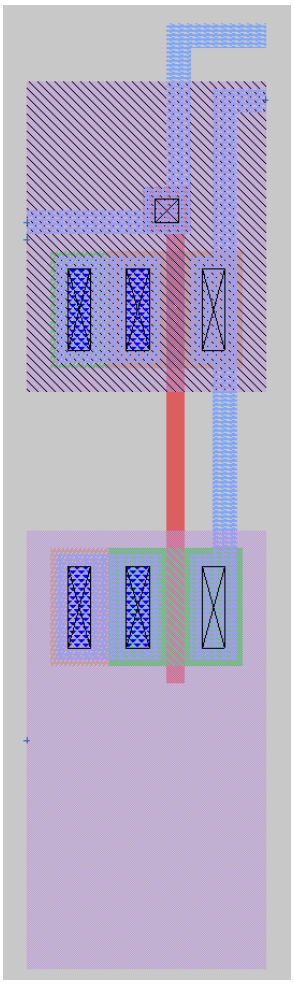

Figure 7: Top level image of layout for an inverter

As shown in figure 7, my inverter contains a pmos and nmos transistor with linked gates taking in a single input, A, and outputting A and the inverterted counterpart, Y, on the north east end of the layout. The VN and VP metal plates spread to the same width as the width of the m1 layers in my flip flop, allowing for it to snap in place at the beginning of the series of flip flops and take in all of the inputs for the shift register.

My inverter had an overall dimension of **2.05 by 8.10 microns**.

3.2 CSRL D-Flip Flop

The layout of my CSRL D-flip flop is shown in figure **??**. The inputs come in from the left side and outputs come out on the right side. The layour has a counterclockwise flow, taking inputs from the north western end, flowing to the south, and coming back up on the north eastern end. I tried to avoid too many intersecting li and poly lines but it got unavoidable at some point.

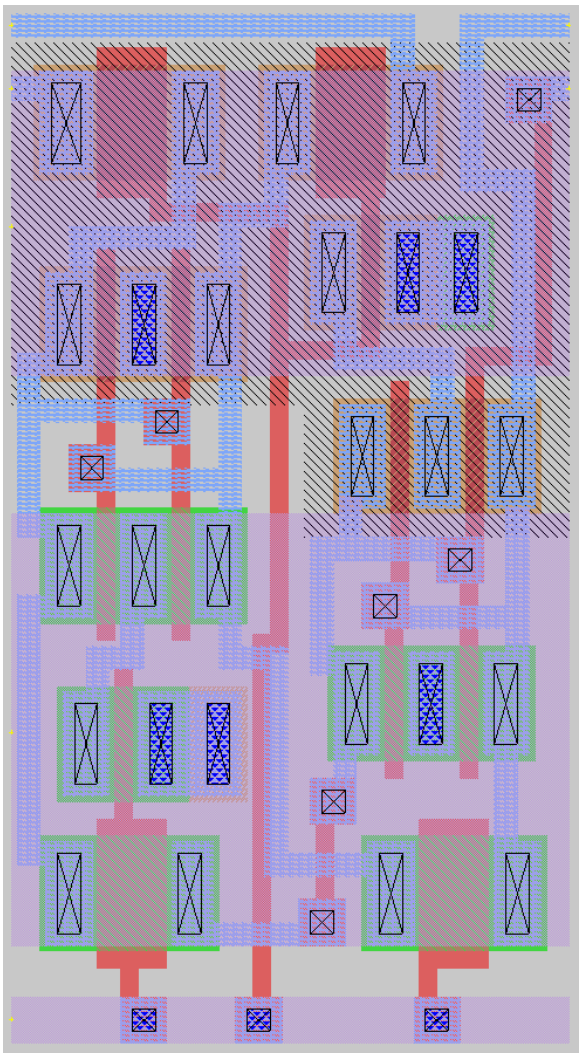

Figure 8: Top level image of layout for CSRL D-flip flop

I attempted to fit my transistors within 2 columns and 5 rows. Once this was done, I made minor changes in order to squeeze the transistors in closer while still following DRC rules. There are 3 metal (m1) rails from top to bottom- VP, VN, and clk.

The overall dimensions of my D-flip flop layout were **4.80 by 8.95 microns**.

3.3 Shift Register

The shift register is constructed by putting an inverter in series with 4, edge-triggered D-flip flops. At the top level assembly, I had to add additional metal plates and local interconnects (li) at the main inputs and outputs, so li at input A (coined D at the lop level), m1 along the left edges of the VP, VN, and clk inputs, and li at each of the ports that I wanted measured. These ports corresponded to Q3, Q2, Q1, Q0, and Qn0 shown in the schematic in figures 4 and 5.

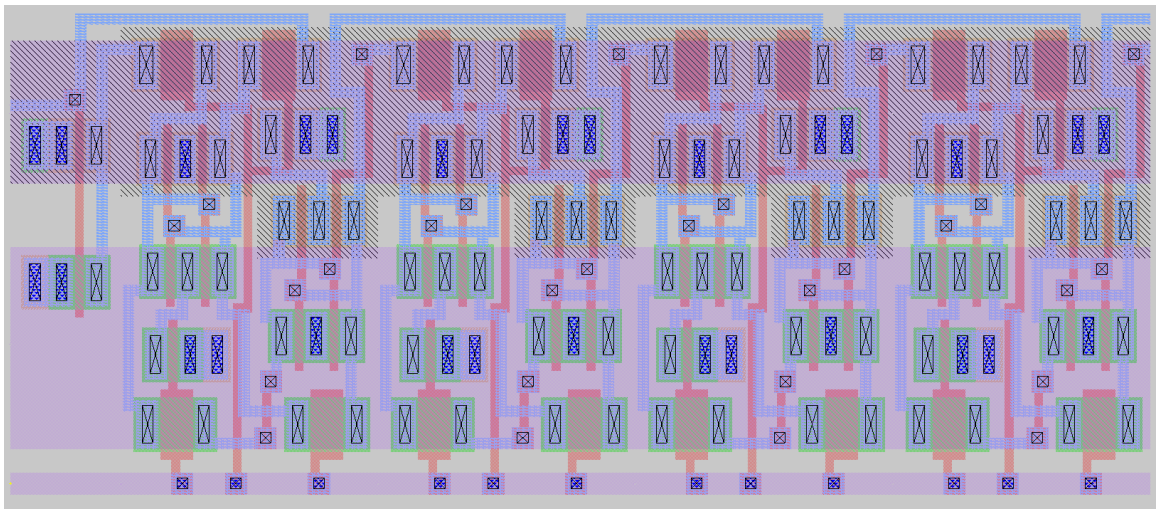

Figure 9: Top level image of Shift Register

The overall dimensions of my shift register layout were **21.25 by 8.95 microns**.

4 Netlist

After the main schematic was made in Xschem and layout was made in Magic VLSI, I generated a netlist and compared between the two D-flip flops and Shift Registers using Netgen's LVS function. Using "lvs", I compared the two netlists. Two successfully matched netlists will result in the terminal displaying the lines below each cell comparison:

Circuits match uniquely.
Netlists match uniquely.

My results can be viewed in my respository linked in the Introduction section in the file ShiftReg_Match.txt and DFF_Match.txt located in the SPICE folder.

5 Reflection

Through this assignment I got a better understanding of the importance of the strength ratios between transistors in a circuit. I also got a better grasp on what cross-coupling inverters do as it had been a while since I was first introduced to them so I did not understand their function until discussing it in class. Since my simulation did not run properly despite my circuit being lain out properly based off of the provided schematic and calculated strength ratios, I am curious if changing my approach on reaching the ratio of 4 will improve my results. I only changed the length of the 4 access device transistors, but I could also try experimenting with messing around with the length and width of those transistors and even the pull-up and pull-down transistors. I would also like to mess around with my layout to try to get it to be thinner.

6 References

1. M. A. Sivilotti, *Wiring Considerations in Analog VLSI Systems, with Application to Field-Programmable Networks*. Pasadena, California USA: California Institute of Technology, 1991.