Mixed Analog-Digital VLSI Mini-Project II: 4-Bit Shift Register

Qingmu Deng

February 24, 2021

Project Links

Project Github: https://github.com/QingmuDeng/MADVLSI/tree/main/mini2
Project zip file: https://github.com/QingmuDeng/MADVLSI/blob/main/mini2/Deng-mini2.zip?raw=true

1 Background

The D flip-flop design in this project is inspired by the thesis work by Sivilotti. [Sivilotti91] One important aspect of the D flip-flop operation is the ratio between the strength ratio of the pull up/down transistors and that of the pass transistors. Too strong of a pass transistor could potentially propagate the behaviors of downstream flip-flops upstreams and result in set/reset issues. The equation of merit is

$$\frac{S_{Q_{pull}}}{S_{Q_{pass}}} > \frac{(\phi - V_{Tn} - nV_A)}{2n(\phi - V_{Tn})V_A - n^2V_A^2} \tag{1}$$

where ϕ is the clock voltage, V_{Tn} the threshold voltage of the transistors, n the subthreshold slope factor, and V_A the voltage at the drain of the pass transistor. Plotting the righthand side equation against the clock voltage, while assuming $V_A = V_{Tn} = 0.52V$ and $n = 1.4$, gives us Figure 1. The

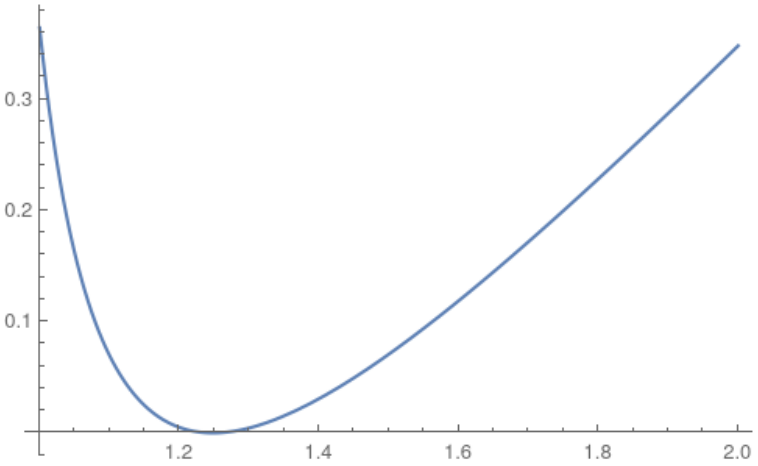


Figure 1: Minimal ratio of the pull/pass transistor strength ratio.

conclusion is that a ratio of 1 should work, as will be shown in the simulations of the next section. In actual D flip-flop design, we adopt the alternative schematic shown in Figure 2 for better schematic and layout symmetry.

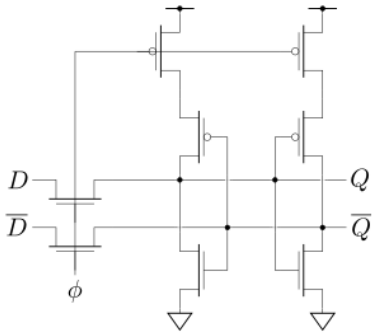
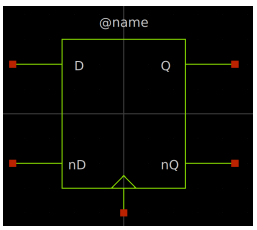


Figure 2: Alternative CSRL Design to be used in the rest of the project.

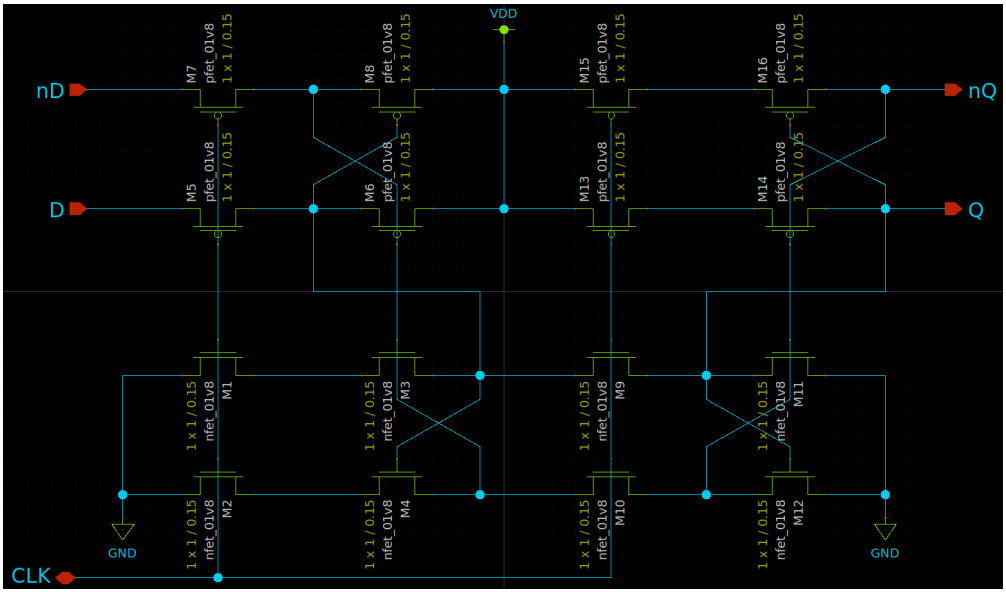
2 Schematic Capture and Simulation

To design a rising-edge triggered complementary set-reset logic D flip-flop, I joined two set-reset latches that are inverted version of one another. The layout driven schematic of this design is shown in Figure 3. The strength ratio of the pull up/down transistor to that of the pass transistor is chosen to be 1, because it not only provides a consistent transistor size for subsequent layout but has been proven to work in Figure 5. Similar tests were performed with hierarchical schematics shown in Figure 6.

Next, I daisy-chained four D flip-flop together to form a 4-bit shift register as shown in Figure 7. I also created the test harness in Figure 8 to validate the functionality of the shift register. The results in Figure 9 shows that the shift register is capable of shifting a bit through the entire device without glitches.



(a) Rising-edge D flip-flop symbol created in Xschem.



(b) D flip-flop layout-driven schematic created in Xschem.

Figure 3: D flip-flop design in Xschem.

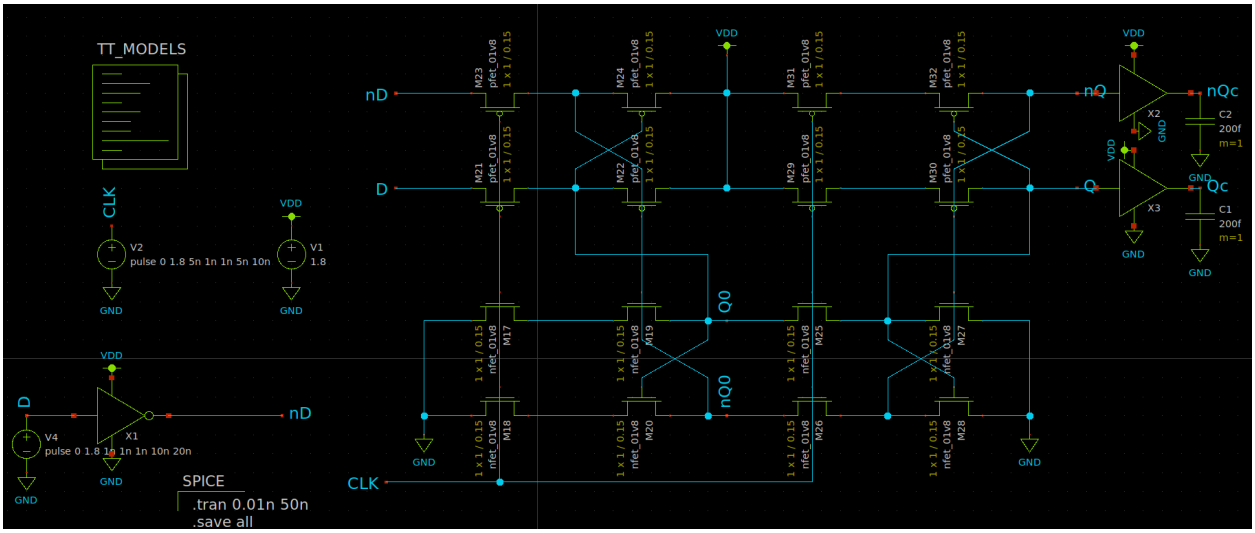


Figure 4: D flip-flop strength ratio test harness created in Xschem.

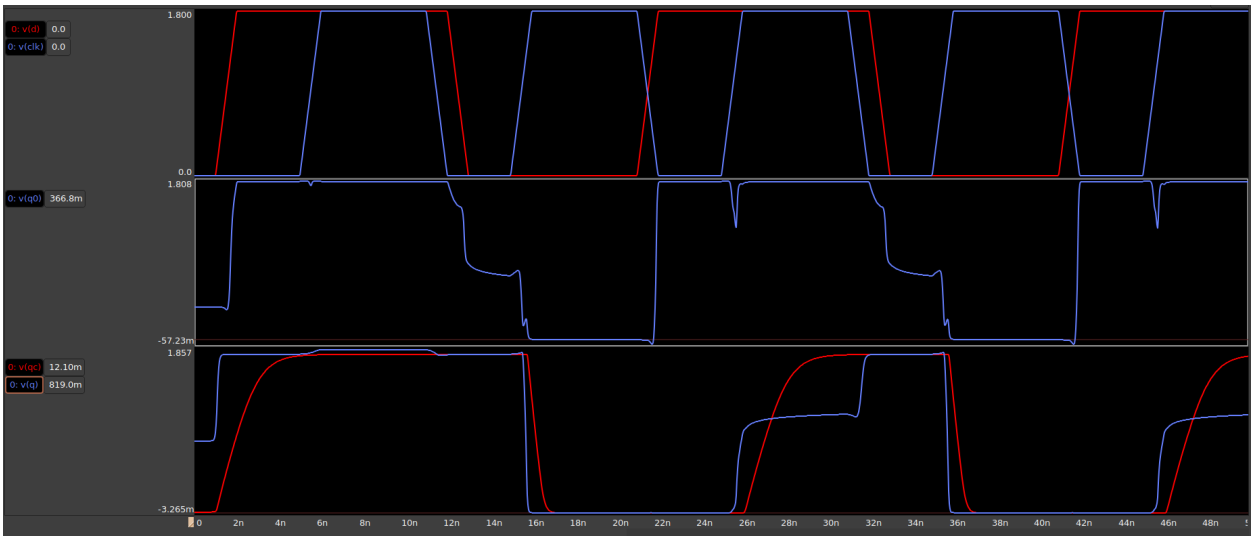


Figure 5: Pull up/down-to-Pass ratio test result for a ratio of 1.

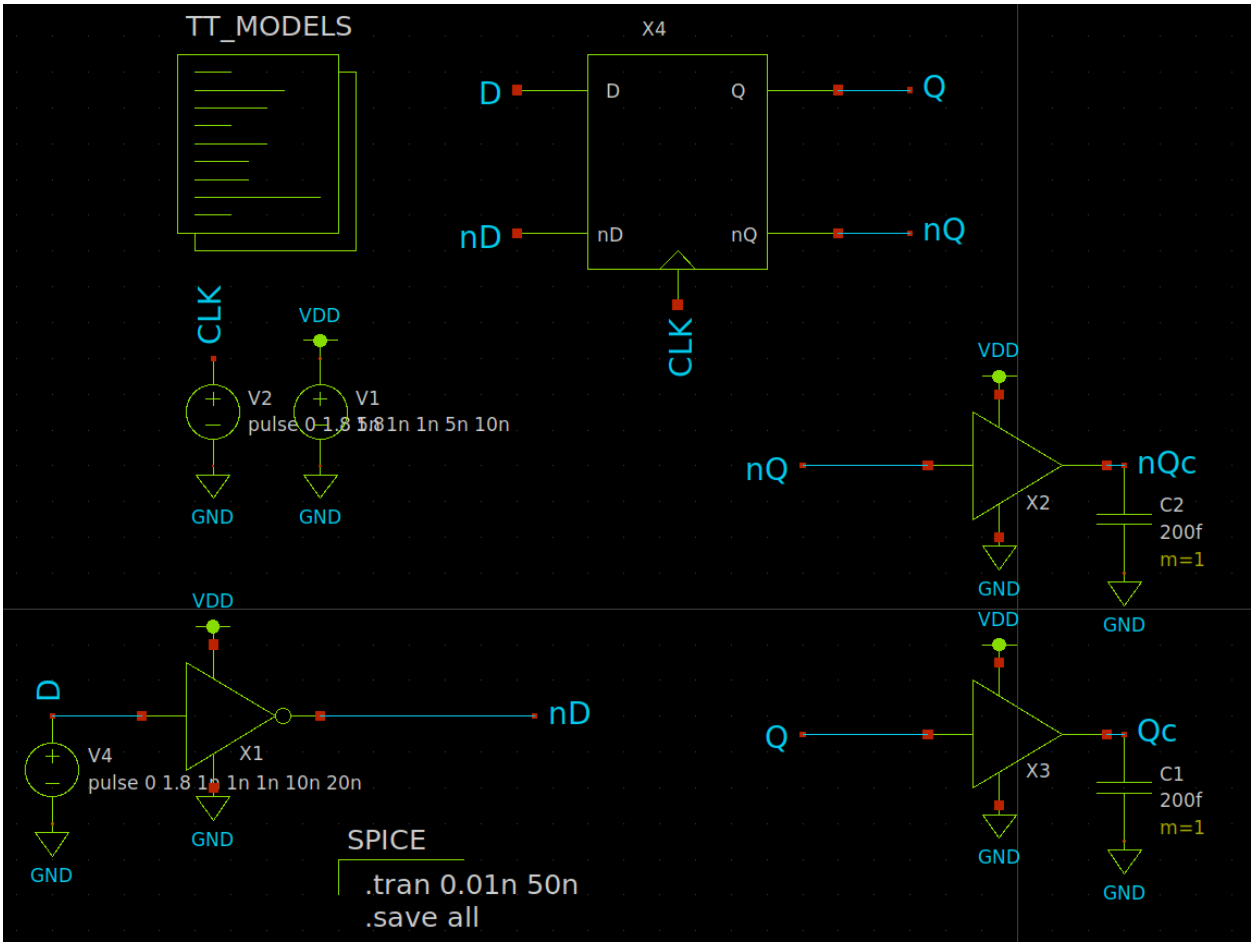


Figure 6: D flip-flop function test harness created in Xschem.

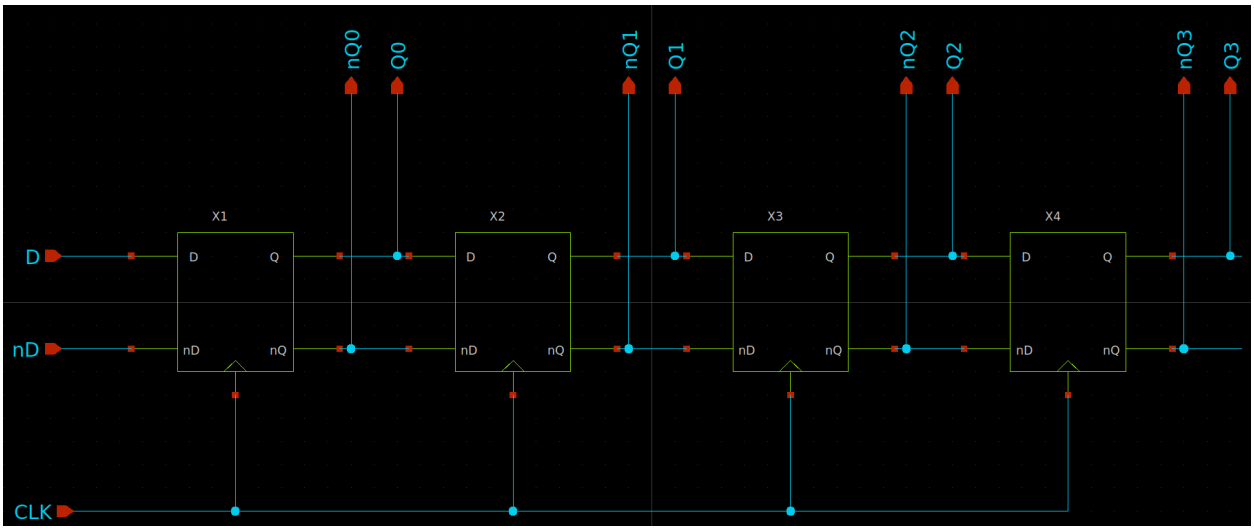


Figure 7: NAND2 schematic created in Xschem.

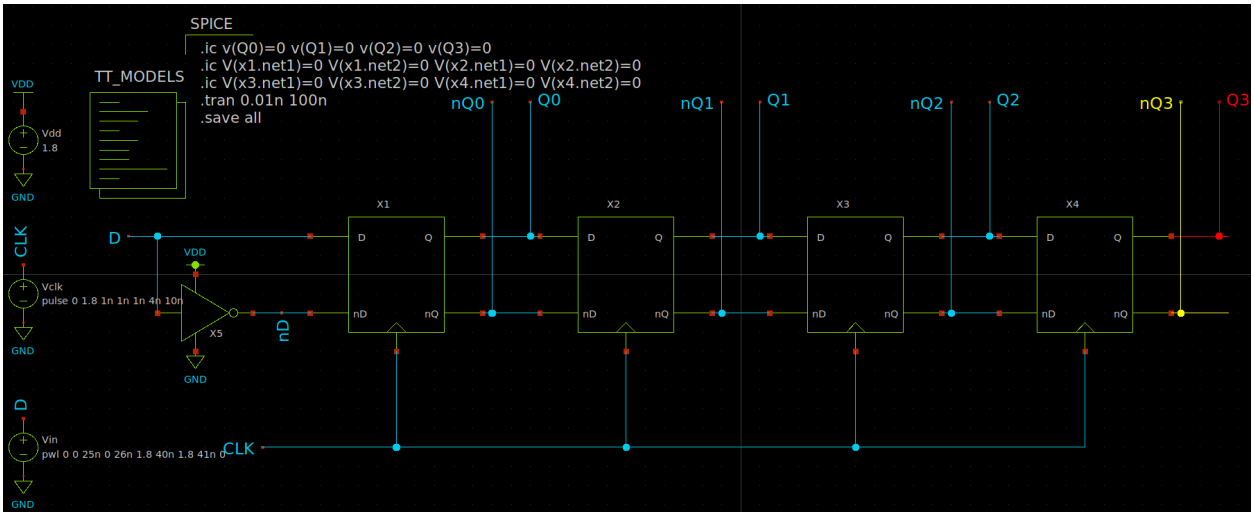


Figure 8: D flip-flop function test harness created in Xschem.

Figure 9: D flip-flop function test harness created in Xschem.

3 Layout Design

To begin with, I laid out my D flip-flop design in Magic. As shown in Figure 10, a **standalone** cell has the following dimensions.

- Width: 3.40 μm
- Height: 9.90 μm

However, the standalone cell include extra *nwell* region necessary for meeting the DRC at the cell level. When abutting adjecent cells together to form a shift register, individual D flip-flop are allowed to overlap, resulting in the following **effective** dimensions.

- Width: 3.15 μm
- Height: 9.90 μm

Figure 11 shows such a design. The overall size of the shift register without inverter is

- Width: 12.85 μm
- Height: 9.90 μm

To allow for the accomodation of the shift register design for digital data whose inverse is not readily availble, Figure 12 shows a design with inverter staged before the first D flip-flop to generate the inverse signal for input. The overall size of the design with an inverter is

- Width: 14.35 μm
- Height: 9.90 μm

. There are no design rule violations. The layout width can be further minimized if I move the bulk contact from in between the rows of pMOS/nMOS transistors to be entirely above/below the rows of the pMOS/nMOS. In that case, the rightmost drain/source regions of the previous D flip-flop cells can be directly laid on top of the leftmost drain/source regions of the next D flip-flop cells without design rule violations. However, I decided not to further pursue that approach for the symmetric looks within the two rows of pMOS/nMOS.

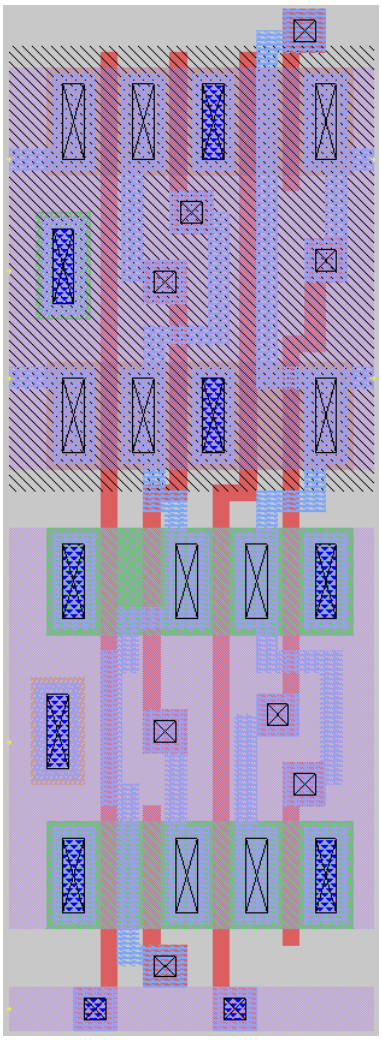


Figure 10: *Magic* layout of the D flip-flop.

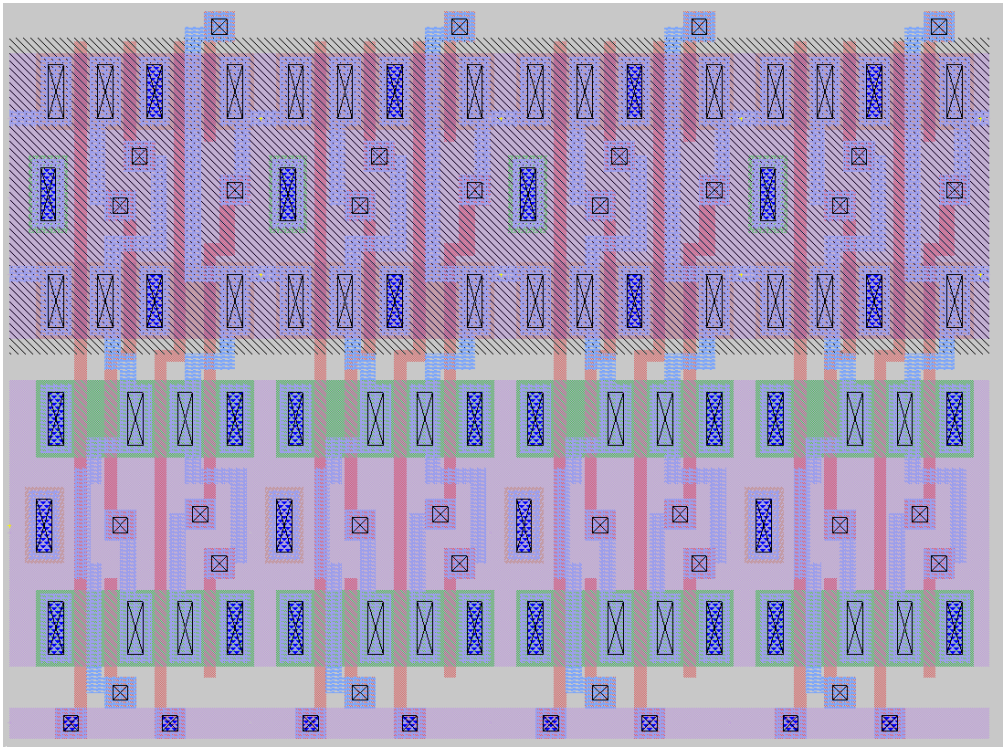


Figure 11: *Magic* layout of the shift register by abutting four D flip-flops.

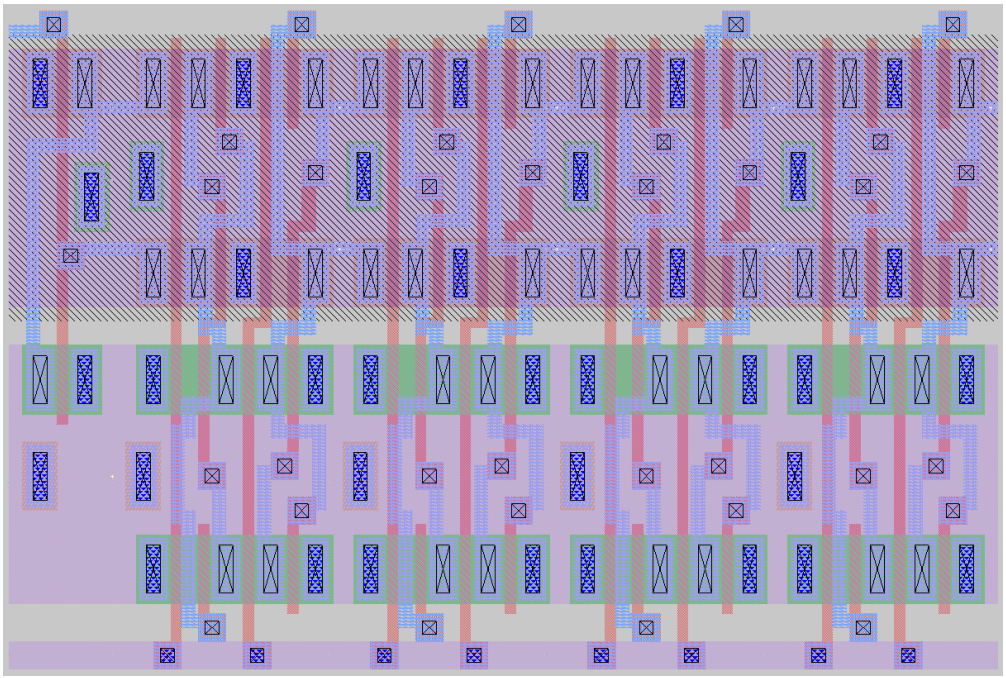


Figure 12: *Magic* layout of the shift register with inverter.

4 Layout versus Schematic

Finally, I performed Layout-versus-Schematic comparison at all levels of the shift register. There are three listings below:

1. Layout versus Schematic for D Flip Flop
2. Layout versus Schematic for Shift Register without Inverter
3. Layout versus Schematic for Shift Register with Inverter

Listing 1: **Layout versus Schematic for D Flip Flop**

```
Equate elements:  no current cell.
Equate elements:  no current cell.

Subcircuit summary:
Circuit 1: layout/dff_lvs.spice                |Circuit 2: schem/dff_lvs.spice
-----------------------------------------------|-----------------------------------------------
sky130_fd_pr__pfet_01v8 (8)                    |sky130_fd_pr__pfet_01v8 (8)
sky130_fd_pr__nfet_01v8 (8)                    |sky130_fd_pr__nfet_01v8 (8)
Number of devices: 16                          |Number of devices: 16
Number of nets: 13                             |Number of nets: 13
-----------------------------------------------|-----------------------------------------------
Resolving automorphisms by property value.
Resolving automorphisms by pin name.
Netlists match uniquely.
Circuits match correctly.
Cells have no pins;  pin matching not needed.
Device classes layout/dff_lvs.spice and schem/dff_lvs.spice are equivalent.
Circuits match uniquely.
```

Listing 2: **Layout versus Schematic for 4-bit Shift Register**

```
Equate elements:  no current cell.
Equate elements:  no current cell.

Subcircuit summary:
Circuit 1: dff                                 |Circuit 2: dff
-----------------------------------------------|-----------------------------------------------
sky130_fd_pr__pfet_01v8 (8)                    |sky130_fd_pr__pfet_01v8 (8)
sky130_fd_pr__nfet_01v8 (8)                    |sky130_fd_pr__nfet_01v8 (8)
Number of devices: 16                          |Number of devices: 16
Number of nets: 13                             |Number of nets: 13
-----------------------------------------------|-----------------------------------------------
Resolving automorphisms by property value.
Resolving automorphisms by pin name.
Netlists match uniquely.
Circuits match correctly.
```

```
Subcircuit pins:
Circuit 1: dff                          |Circuit 2: dff
----------------------------------------|----------------------------------------
nD                                      |nD
D                                       |D
CLK                                     |CLK
VN                                      |GND **Mismatch**
VP                                      |VDD **Mismatch**
Q                                       |Q
nQ                                      |nQ
----------------------------------------|----------------------------------------
Cell pin lists are equivalent.
Device classes dff and dff are equivalent.

Subcircuit summary:
Circuit 1: layout/shiftreg_4_lvs.spice  |Circuit 2: schem/shiftreg_4_lvs.spice
----------------------------------------|----------------------------------------
dff (4)                                 |dff (4)
Number of devices: 4                    |Number of devices: 4
Number of nets: 13                      |Number of nets: 13
----------------------------------------|----------------------------------------
Circuits match uniquely.
Netlists match uniquely.
Cells have no pins;  pin matching not needed.
Device classes layout/shiftreg_4_lvs.spice and
schem/shiftreg_4_lvs.spice are equivalent.
Circuits match uniquely.
```

Listing 3: **Layout versus Schematic for 4-bit Shift Register with Inverter**

```
Flattening unmatched subcell shiftreg_4 in circuit layout/sreg.spice
(0)(1 instance)
Equate elements:  no current cell.
Equate elements:  no current cell.

Subcircuit summary:
Circuit 1: dff                          |Circuit 2: dff
----------------------------------------|----------------------------------------
sky130_fd_pr__pfet_01v8 (8)             |sky130_fd_pr__pfet_01v8 (8)
sky130_fd_pr__nfet_01v8 (8)             |sky130_fd_pr__nfet_01v8 (8)
Number of devices: 16                   |Number of devices: 16
Number of nets: 13                      |Number of nets: 13
----------------------------------------|----------------------------------------
Resolving automorphisms by property value.
Resolving automorphisms by pin name.
Netlists match uniquely.
Circuits match correctly.

Subcircuit pins:
```

Circuit 1: dff	Circuit 2: dff
nD	nD
D	D
CLK	CLK
VN	GND **Mismatch**
VP	VDD **Mismatch**
Q	Q
nQ	nQ

Cell pin lists are equivalent.
Device classes dff and dff are equivalent.

Cell inverter disconnected node: CLK

Subcircuit summary:

Circuit 1: inverter	Circuit 2: inverter
sky130_fd_pr__nfet_01v8 (1)	sky130_fd_pr__nfet_01v8 (1)
sky130_fd_pr__pfet_01v8 (1)	sky130_fd_pr__pfet_01v8 (1)
Number of devices: 2	Number of devices: 2
Number of nets: 4	Number of nets: 4

Circuits **match** uniquely.
Netlists **match** uniquely.

Subcircuit pins:

Circuit 1: inverter	Circuit 2: inverter
nD	Y **Mismatch**
D	A **Mismatch**
VP	VP
VN	VN

Cell pin lists are equivalent.
Device classes inverter and inverter are equivalent.

Subcircuit summary:

Circuit 1: layout/sreg.spice	Circuit 2: schem/sreg_lvs.spice
inverter (1)	inverter (1)
dff (4)	dff (4)
Number of devices: 5	Number of devices: 5
Number of nets: 13	Number of nets: 13

Circuits **match** uniquely.
Netlists **match** uniquely.
Cells have no pins; pin matching not needed.

Device classes layout/sreg.spice and schem/sreg_lvs.spice are equivalent.
Circuits **match** uniquely.