

Lab 3 Guideline

1. Design a latch (refer to the following pages) using the inverter and transmission cell.
2. Verify the function of a latch in cadence.
3. Design a flip-flop (refer to the following pages) using your latch cell.
4. Verify the function of the flip-flop in cadence
5. Check your results with the TA *before you leave*

Lab Questions:

1. In your lab report, please draw the schematic of latch and flip-flop (gate-level).
2. Draw the schematic of transmission gate and inverter (transistor-level)
3. Tell the difference between the latch and flip-flop, combinational logic and sequential logic.

Useful Command

1. To find out how to use existing cell (in Virtuoso).
 - a) Select **Create -> Instance**
 - b) Choose the Browse.
 - c) Choose a view name under View which you want to use.
 - d) Put the Cell the position you want on the Virtuoso Layout editor.

If you want to change the direction of the cell, you can use the “Rotate”, “Sideways”, or “Upside Down” in the Create Instance window.
2. To access or edit a descendent cell (selected by Create->Instance) in your working cell.
 - a) Choose the descendent cell.
 - b) Select **Design->Hierarchy->Descendent**
 - c) Once the descendent cell is opened, you can edit the cell.
(But, in this Lab, you don't need to edit the cell)

d) To return to the your working cell, Select **Design->Hierarchy->Return**

FAQ:

1. In your working cell, **the pin names of descendent cell is not meaningful.**
Have to add Pin Names like VDD and VSS Again in your working Cell.
2. To make the descendent cell a black box in your working cell, Use **Ctrl+F** Key.
Reversely, to view the descendent cell in your working cell, Use **Shift+F** Key.
3. To change the direction of a cell, **MOVE** command can be used.
4. Metal2 can't be directly connected to Poly, metal2 should be connected to metal1,
and then connect metal1 to poly.

LAB3. D FLIP-FLOP Design

In this lab, you will draw the layout for a D Flip-Flop using Cadence. Upon completion of the layout and parameter extraction, you'll simulate the netlist using Cadence or Hspice. Before designing the D flip-flop, design a one-bit level-sensitive D latch. Label the input data as **D**, the control signal as **S** (store), and **SB** (storebar), and the output as **QB**. Use Cadence cell hierarchy to design the latch. In order to simplify the layout, use one inverter cell and one transmission gate as subcells in this lab. The top-level parent cell, **latch**, will contain four children cells, **inv_0**, **inv_1**, **tg_0**, and **tg_1**. When using cell hierarchy in Cadence, make sure that the labels, **D**, **QB**, **S**, and **SB**, are placed on a square of paint contained in the top-level parent cell, **latch**. When testing the latch, the data and control signals should not both be changing at the same time. The data should remain stable or valid as the latch control signal goes low. Then, using D latches, design a D flip-flop. Make sure that the labels of this flip-flop are **D**, **Q**, **S** (CLK), and **SB** (CLKB).

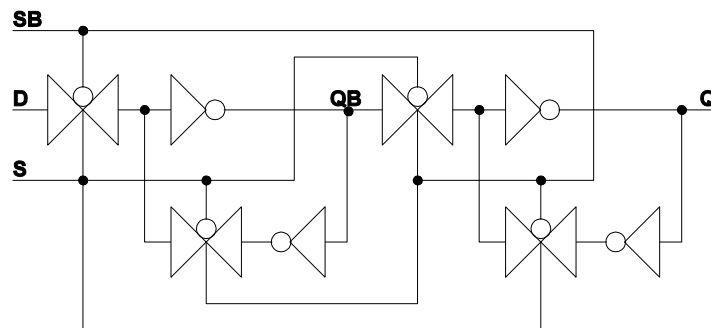


Fig.L2.0. A schematic of D flip-flop

STEP1: Design of Cells – Inverter, TG, Latch, and D F/F

The following shows the steps for designing the cells using a stick diagram:

- a. Design an inverter and a transmission gate whose stick diagrams are as shown in Fig.L2.1. Make sure that the height of these cells is the same ($< 40\lambda$), and make the cells as small as possible. You can take advantage of the inverter in Lab1 to create the transmission gate because both the transmission gate and the inverter have one PMOS and one NMOS. Modify the inverter in Lab1 to create the transmission gate. Of course, you can use the inverter itself for Lab3.

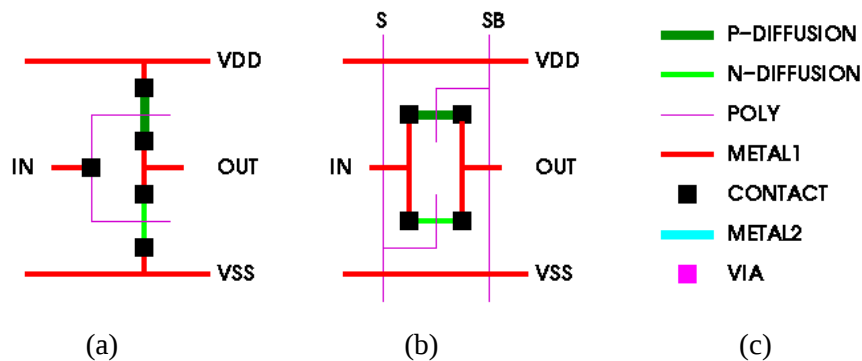


Fig.L2.1 (a) Stick diagrams of inverter and (b) transmission gate, and (c) the legend of this stick diagram

- b. Design a latch

Put two instances of an inverter and a transmission gate needed to each other on a new cell as shown in Fig.L2.2 (ignore the inverter and the transmission gate in the middle, which is an intermediate process). Merge one of the inverters and with a transmission gate in the right-upper corner of Fig.L2.2. This will now be the upper half of your CMOS latch. Flip the second inverter and transmission gate vertically and horizontally as shown below. You will now have the bottom half of your CMOS latch as shown in

Fig.L2.2. Note that you need to remove all labels when using cell hierarchy from all cells. Once the final design is completed, label this design.

Next, merge the two halves of the latches to create a negative sensitive latch as shown in Fig.L2.3 (of course, you can layout all cells in a row instead of two rows). The polarity (QB) is not important in this latch because the flip-flop consists of two identical latches. In other words, the polarity is negative of negative, or simply positive. When merging the inverter and the transmission gates, use metal2 and via layers for routing easily (in order to avoid shorting wires) as shown by the step in Fig.L2.3. Note that the routing process only shows connectivity of cells in Fig.L2.4, thus you should compact your layout keeping minimum sizes and shortest paths.

c. Design a D flip-flop

Place two instances of the latch you just created on the new cell, and merge them as shown below in Fig.L2.4. Note that the routing process only shows connectivity of cells in Fig.L2.4, thus you should compact your layout keeping minimum sizes and short paths. Don't forget to draw an N-well layer in your design.

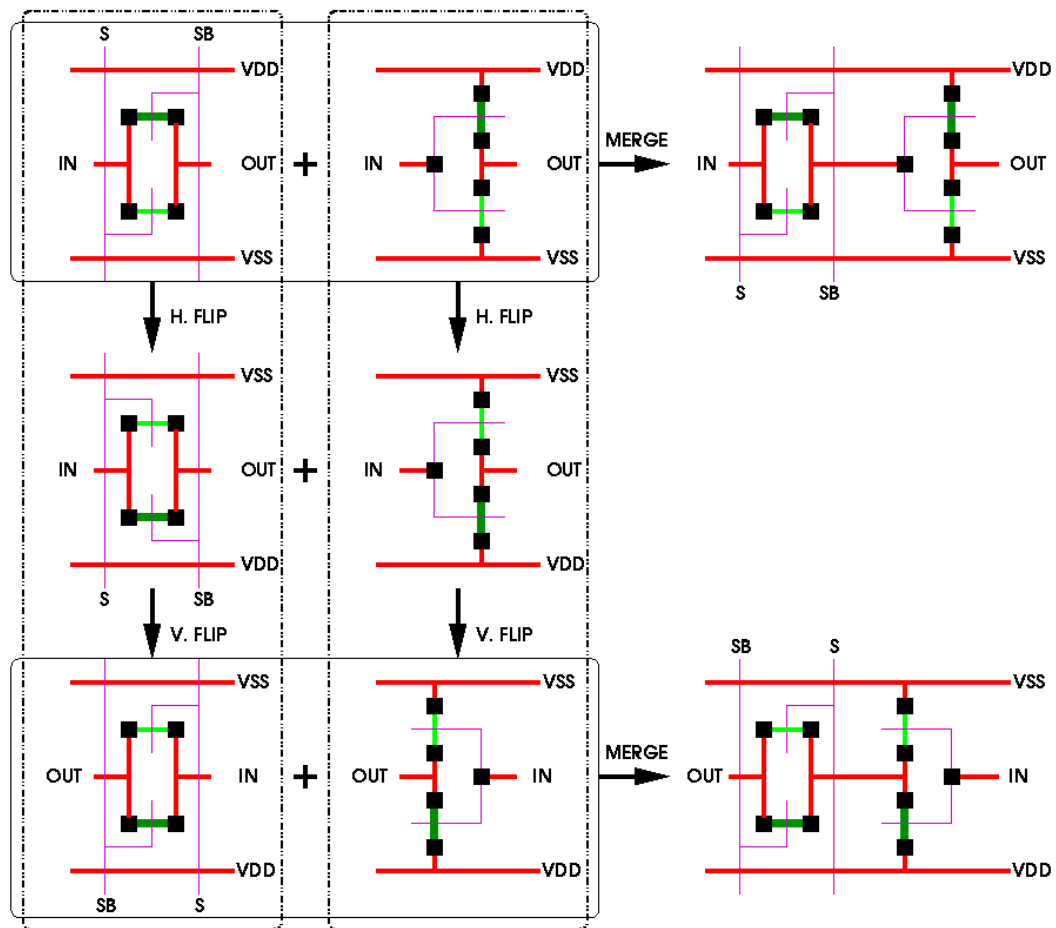


Fig.L2.2 Design steps for two different geometry half latches.

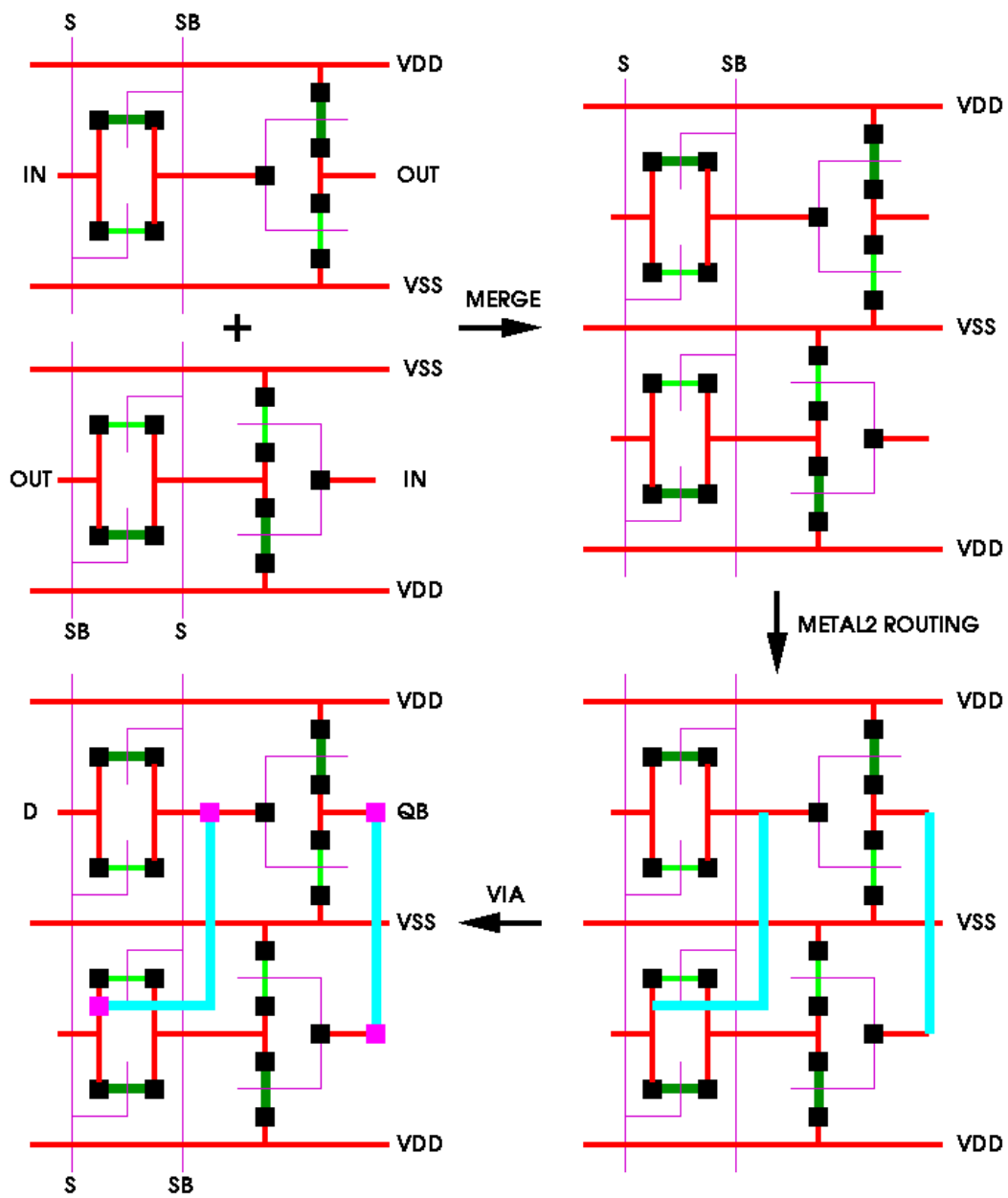


Fig.L2.3 Design steps for a negative polarity latch.

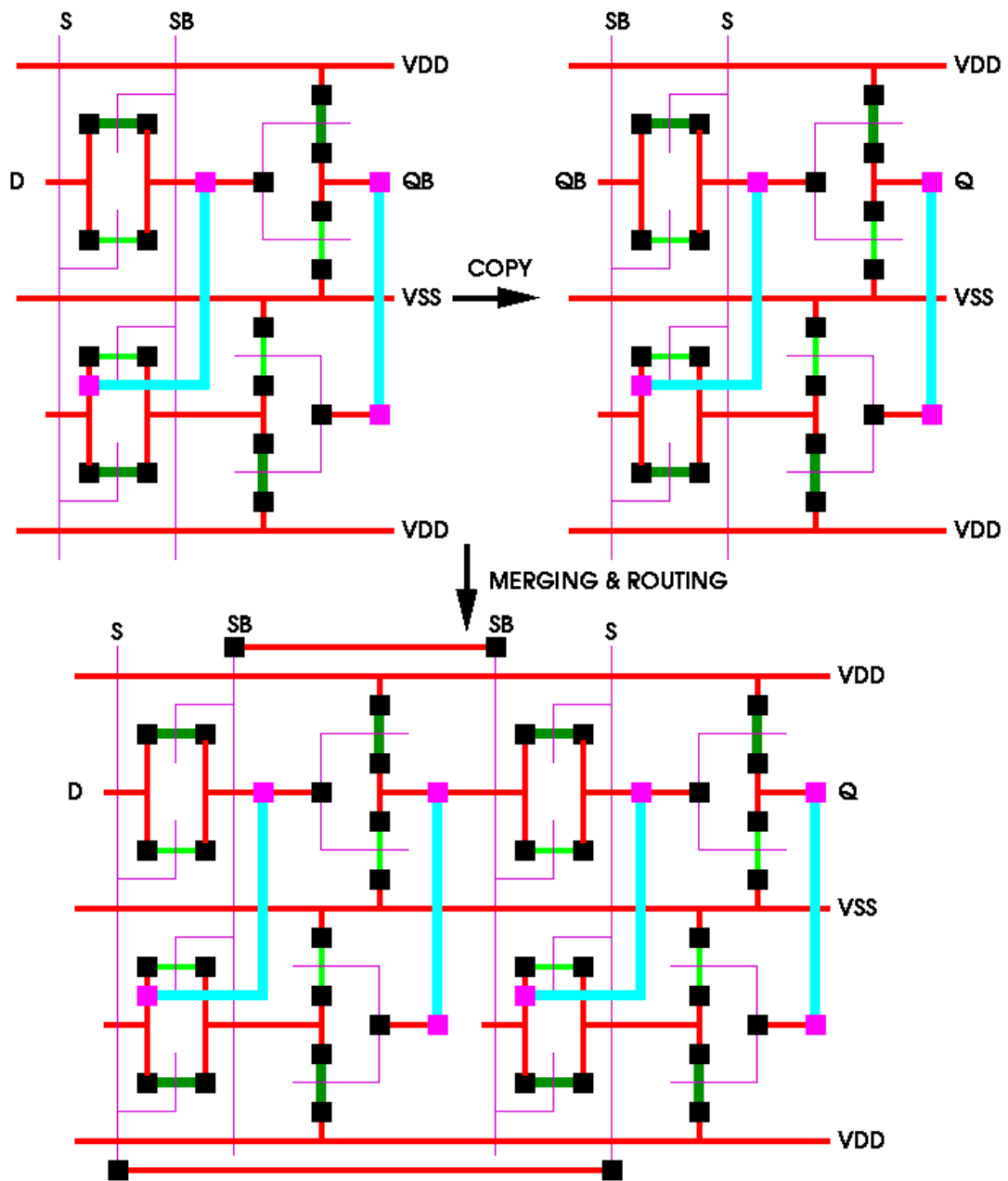
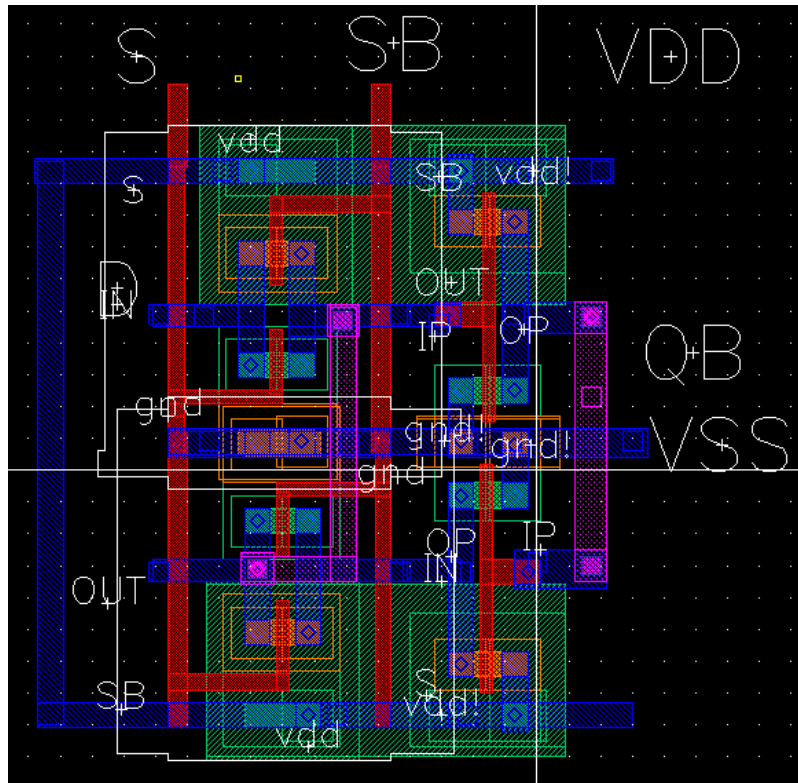


Fig.L2.4 Design steps for a D flip-flop

Final Layout

1. Latch



2. Flip_Flop

