

ELEC301

Lab 6: Hierarchical Layout Design

Revision: 2.0

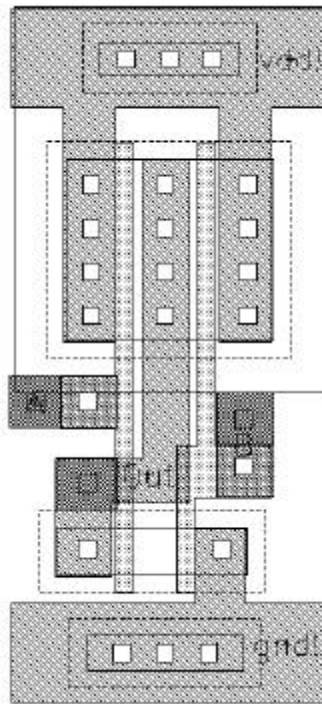
Date: July 98

Objective

This lab provides a practice to extend the knowledge on layout technique based on the knowledge acquired in Lab3. First part of this lab requires creating a 2-input NAND gate layout structure and performing design rule checking to verify the correctness of the physical design. This layout has to associate with the schematic and symbol built up in myLib of Lab2.

The second part of the exercise is to realize the hierarchy layout structure similar to the schematic hierarchy structure of which the top schematic view is based on the sub-cell. To create the “clkgen” layout cellview, it has to associate with the schematic and symbol of the clkgen. Finally is to add pins to all the layouts to prepare for running LVS in next lab.

Layout example of
NAND2

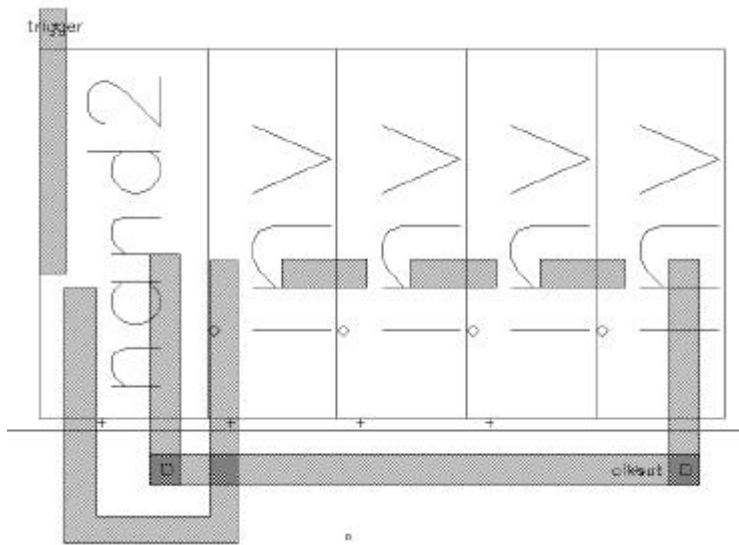


Step 1. Change to the working directory and start Cadence.

Step 2. Create a layout view for “nand2” with the library name of “myLib”.

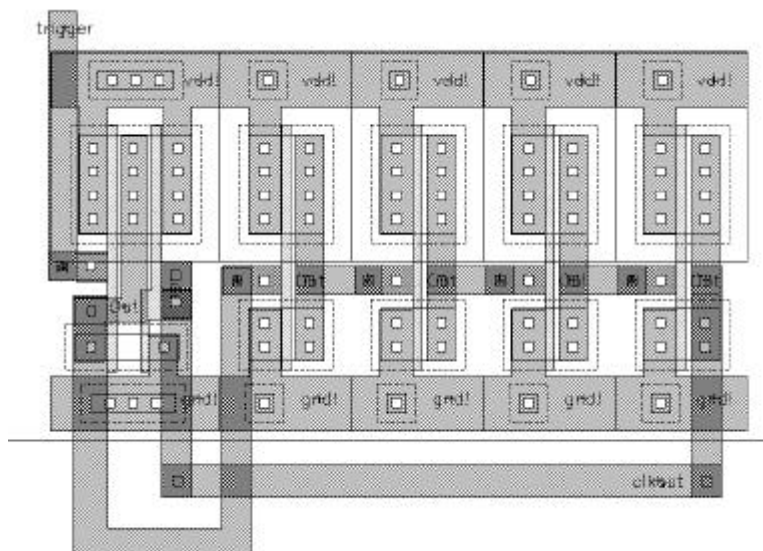
Step 3. Proceed to create the layout of *nand2* by activate every entity in *LSW* successively to build up the structure. There is a constraint that has to follow. It is the height of the layout cellview of *nand2* has to match with the height of the “inv” layout cellview. (In other words, every student has different dimension in height for its “inv” layout cellview. Say, X um. The layout cellview for its height have to be X um as well).

- Step 4. Perform DRC check to debug for any dimension errors with respect to the fabrication process limitation and correct until no DRC error found.
- Step 5. Save the cellview and prepare for the second part of this tutorial. It has to open a new design layout cellview with the name of “clkgen” with library of “myLib”.
- Step 6. From the fix menu, select “**Instance**” to invoke the subcells of **nand2** and **inv** successively to configure the layout of the subcells to becomes the clkgen layout structure. It has to pay attention to the height of the subcell **nand2**, **inv** and have to be the same. The power line of vdd and gnd have to align accurately by used of the Metal1 entity. Metal2 entity can be used as connecting path for the signal of input and output.



Step 7. Sub-cell hierarchy structure of “clkgen” being display (above) by cell boundary.

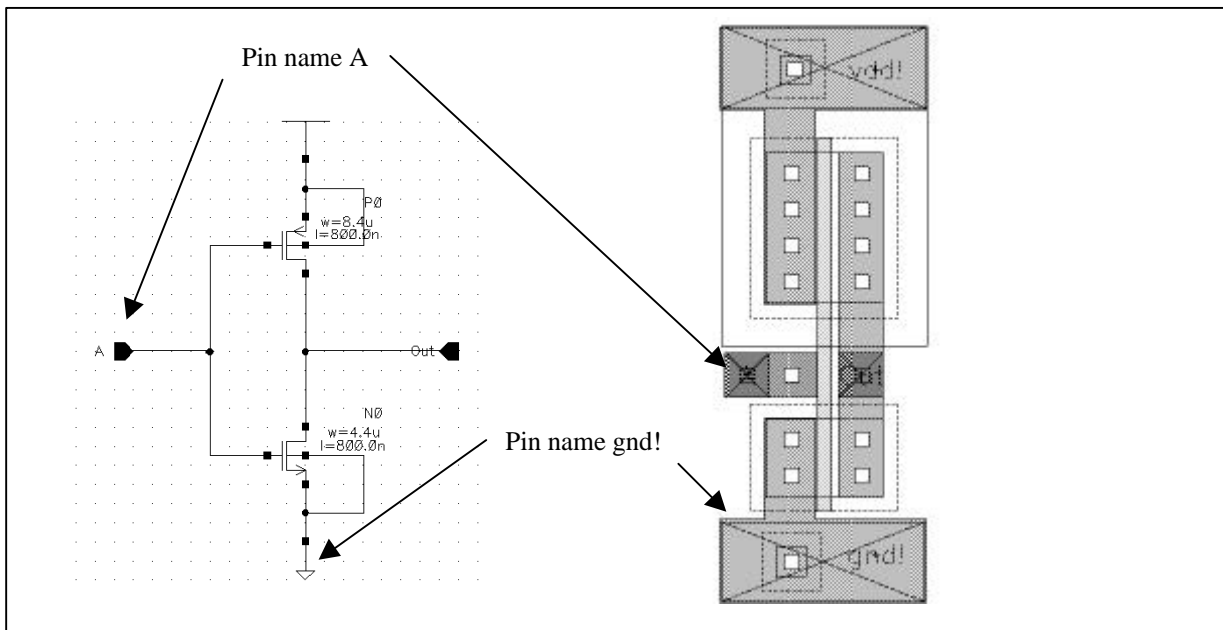
- Use **shift-f** key to show the subcell entities structure of the clkgen layout.
- Use **control-f** key to show back the hierarchy structure of the clkgen layout.



Step 8. Perform the DRC check to debug for any dimension errors until no DRC error found for this layout cellview of “clkgen”.

Creating Pins for LVS

We need to create pins (not labels) to tell LVS tools what is the starting point for comparing layout and schematic. For example, pins A and gnd! In layout are created (crossed rectangle) corresponding to schematic pins as shown below:



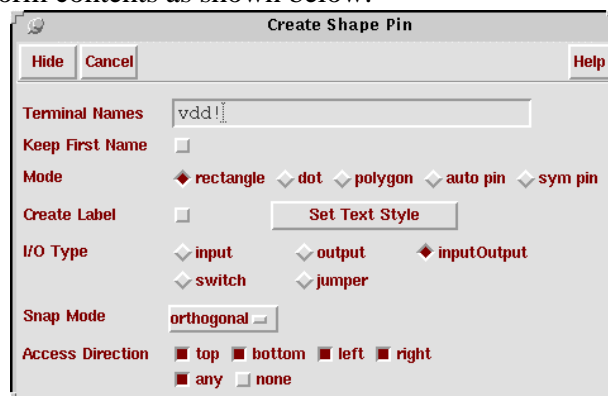
Step 1. Create pin for *vdd!*.

1.1. Open *inv* layout for editing. In *LSW*, select **metal1 (pn)**.

- **(pn)** stand for *pin* purpose.

1.2. In layout window, click on **Create – Pin**.

- Changing the form contents as shown below.



1.3 You should see the messages now in *CIW* to prompt you create pin. Drawing a rectangle in *vdd!* metal as shown in *figure 1*.

- The size of pin need not be the same as metal 1 (dg) since it used only for connectivity identification.

Step 2. Creating pin for *gnd!* by the same procedures.

Step 3. Creating pin for signal *A*.

3.1. Using **metal2 (pn)** instead of **metal1 (pn)**.

3.2. *I/O Type* should be *input*.

Step 4. Creating pin for signal *Out*.

4.1. *I/O Type* should be output and using **metal2 (pn)**.

Step 5. By similar procedure, creating *vdd!*, *gnd!*, *A*, *B*, *Out* for nand2.

Step 6. Similarly, creating *vdd!*, *gnd!*, *trigger*, *clkout* for the clkgen.

Step 7. Perform DRC to ensure that all the layouts are error free.

End of Lab