EE 323002
Lab 6: Post-Layout Simulation and Hierarchical Layout

Version 1.0 (Sept 2023)

Objective

- In lab 5, you completed the inverter and nand2 layout designs, and they passed the DRC and LVS checks. In this lab, you are going to perform the circuit simulation called **post-layout simulation** using their extracted views (i.e. extracted netlists).

- We developed the schematic view of the ring oscillator in lab 3 using the hierarchy design structure, with the **inv** and **nand2** cells being the sub-modules of this design. The layout of the ring oscillator will be built by use of the layouts of **inv** and **nand2** cells. Post-layout simulations will be carried out to see the difference in performance with different capacitive loads applied at the output terminal.

I. Standard-cells based design

We have developed some commonly used logic cells in our **323002mylib** library. All cell layouts should be designed with a **fixed height** so that they can be abutted side by side to forms rows for sharing a common power and ground bus. Moreover, these fixed height cells can enable automated placement and routing of cells.

Step 1. **Ensure all** the **layouts** of **inv** cell, **nand2** cell and **buf** cell having the **same height**, if not, you should correct them.

- Re-run DRC and LVS checks if you make any changes on the layout views.

II. The extracted view generation from the layout

Circuit extraction is performed after the layout design is completed. The circuit extractor generates an extracted view from a layout view. What is the use of the extracted view? The circuit extractor is able to estimate actual devices' dimensions, interconnections and the parasitic components present between layers from the layout view, this information is then stored in the extracted view. Thus, the resulting circuit simulation results obtained from its extracted view are a better estimation of the circuit's real performance.

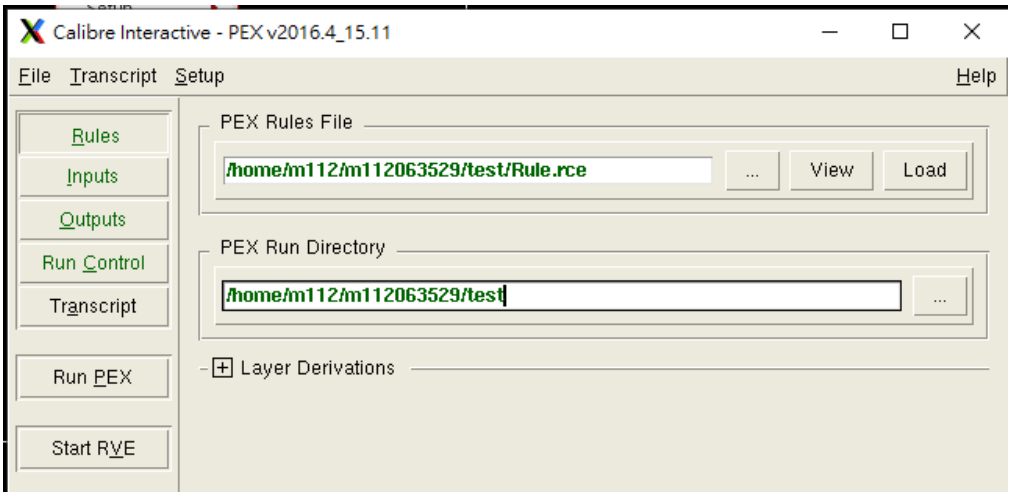
Step 2. To generate the extracted view of the **inv** cell; Open the **layout** view of the **inv** cell.

Although you did LVS before, you have to perform LVS again **everytime before** doing the extraction.

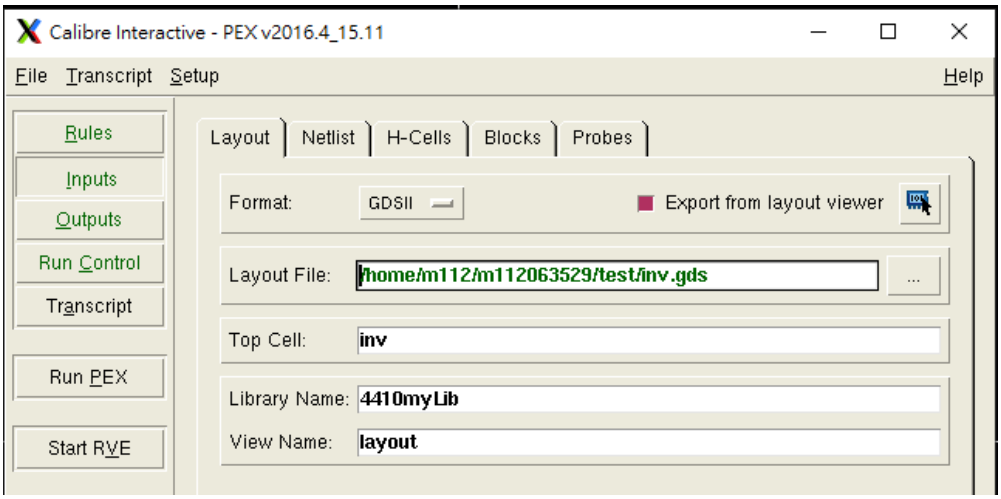
- In the layout view, do the LVS again (refer to Lab5).
- Export gds: File→export→stream→inv.gds
- As LVS finished, click on *Calibre* → ***Run PEXs,***
- Change the content in Rule.rce and make sure Rule_08KA and Rule_20KA is in the same directory as Rule.rce. Change source path, source primary, layout path, and layout primary.

```
SOURCE PATH      "inv.spi"
SOURCE PRIMARY   "inv"
SOURCE SYSTEM    SPICE

LAYOUT PATH      "inv.gds"
LAYOUT PRIMARY   "inv"

LAYOUT SYSTEM    GDSII

LVS REPORT       lvs.rep
```

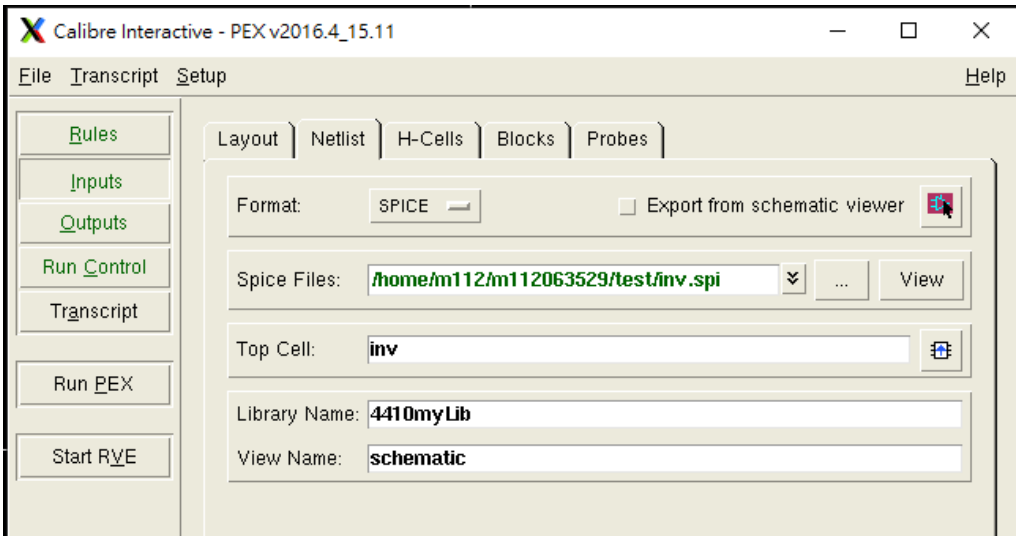
- Rules→ import “Rule.rce”



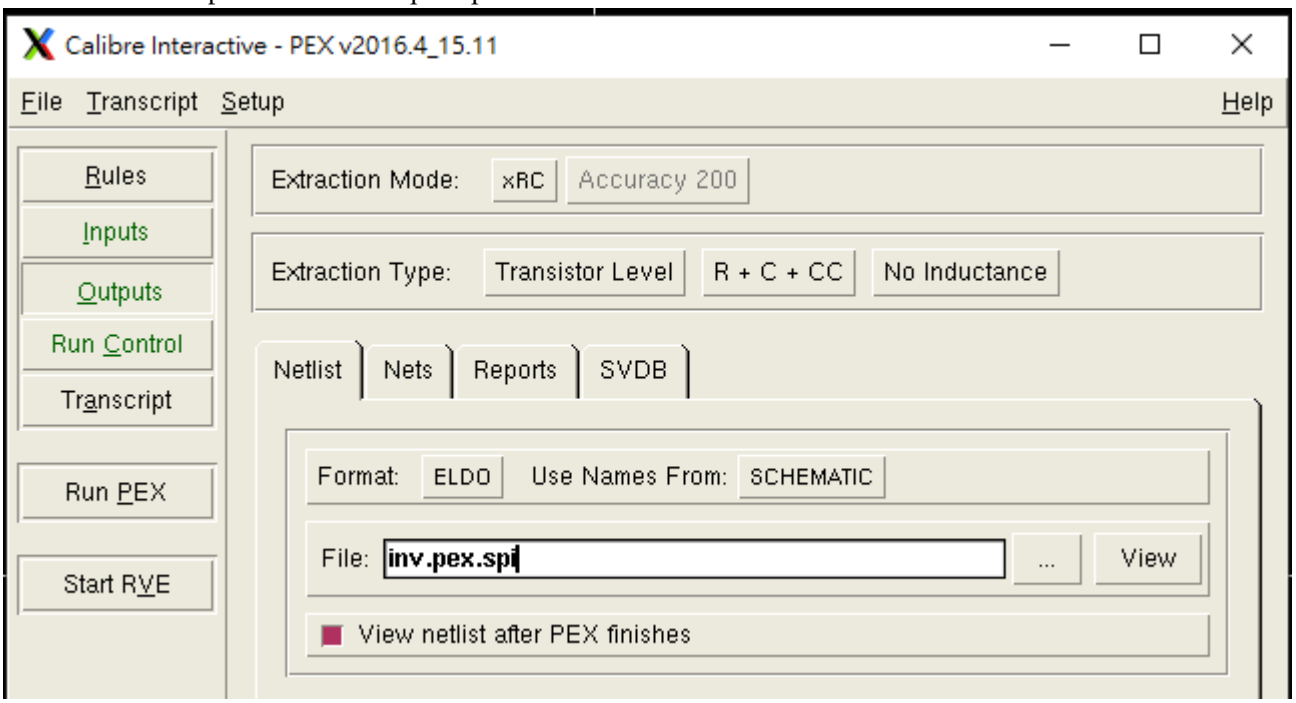
- Inputs→ Layout→”gds file”



- Inputs→ Netlist→”spi file”



- Output→ Netlist→ “pex.spi file”



As the extraction finished, in CIW, the message shows that there is an av_extracted view stored under your ***inv*** cell. Open this newly created view and see what it is. If the extraction process failed, ask TA for help.

Step 3. Follow the above steps to generate the extracted view of the **nand2** cell and **buf** cell.

III. Post-Layout Simulation of the inverter design

We use the **extracted** view to run the post-layout simulation. The netlist generated from the extracted view includes the parasitic components inevitably present between layers. Therefore, the resulting simulation results provide a better estimation on its real performance. You did the **schematic circuit simulation** on **inv** and **nand2** cells before (in lab 2), in this part, you will run the **post-layout simulation**. You should compare the results to those obtained in lab2.

Steps to run the post-layout simulation on **inv** cell’s extracted view,

Run the simulation in the way as you did in lab2.The capacitive load added at the output terminal is 0.3pF.

- Record the $\mathbf{t_{PLH}}$ and $\mathbf{t_{PHL}}$ and report the result to your TA.

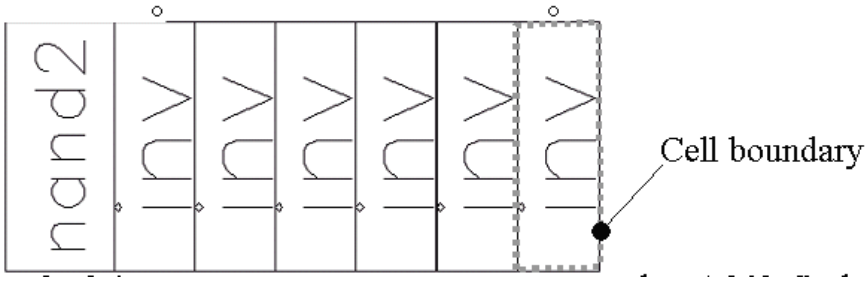
  As the simulation ends**, look into** the **netlist** file (see lab2 for its path). Are any parasitic components shown there? Which nets are they connected to? Draw them on the schematic view of this inverter and show your drawing to TA.

IV. Hierarchy Layout Design for Ring Oscillator

We have built the schematic view of the **clkgen** cell in lab 3 using the hierarchy design structure. Now we are ready to construct the ring oscillator layout view in hierarchy because we have the layouts of **inv** and **nand2** views done (they are the sub-modules of the **clkgen** cell).

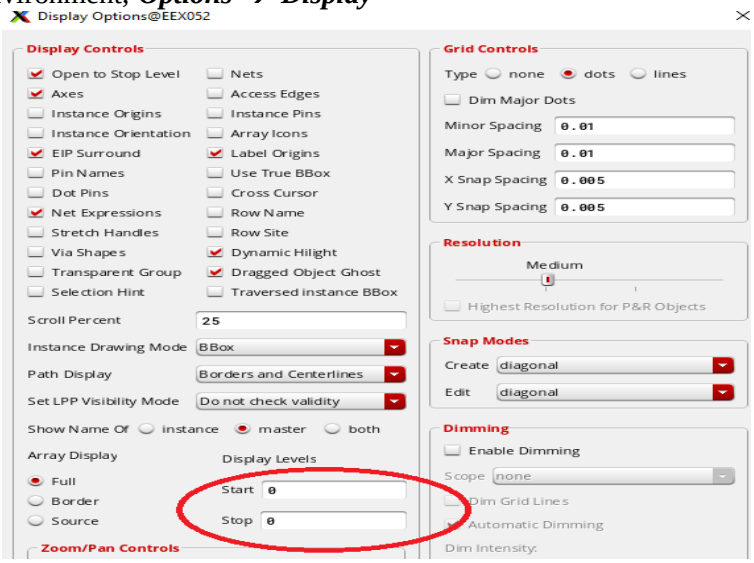
Step 4. Create a layout view of **clkgen** cell using Library Manager.

- In the layout editing window, insert the **nand2 layout** view by clicking on ***Create → Instance***.
- Insert the **inv layout** view in the same way.
- Line up these layout blocks as shown below. The details of the layouts do not show up because the display is set to 'level 0' by default, that is the top level.



- You can look into these blocks using the short-cut key '**shift-f**', then '**control-f**' to restore. **Or** you can set your display environment, ***Options → Display***

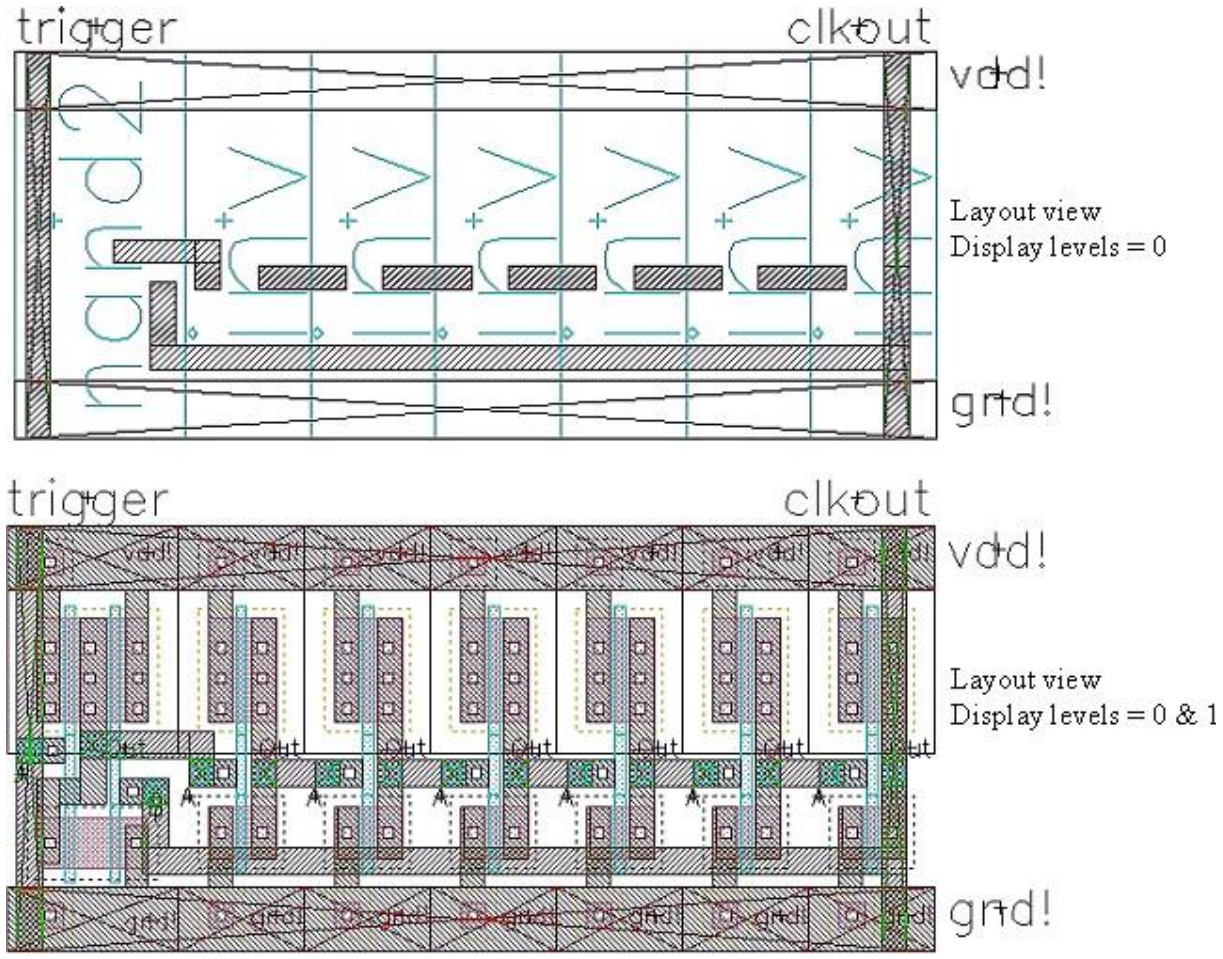
To display down one level, set this number to '**1**' instead of '0'.



Step 5. Refer to the clkgen's schematic view, properly connect the sub-modules (the **inv** and **nand2** cells) by using the **Metal 2** (drw) layer (the clkgen layout shown below is for reference only).

- Pay attention to the height of the cells, their power and ground railsshouldbe joined and aligned properly.
- Perform DRC, LVS and extraction, and debug the design if there are any errors.

Layout view of clkgen (for reference only)



V. Post-Layout Simulation for clkgen - without loading effect

Step 6. Run the post-layout simulation on the **extracted** view of the **clkgen** cell,

- **NO capacitive load** at its clkout terminal.
- Record the oscillating frequency and fill in the table on the last page .

VI. Post-Layout Simulation for clkgen - with loading effect

Step 7. Repeat step 8 and rename the test.input file to test-003p.input and add a loading capacitance of **0.3pF** in your **test-003p.input file**.

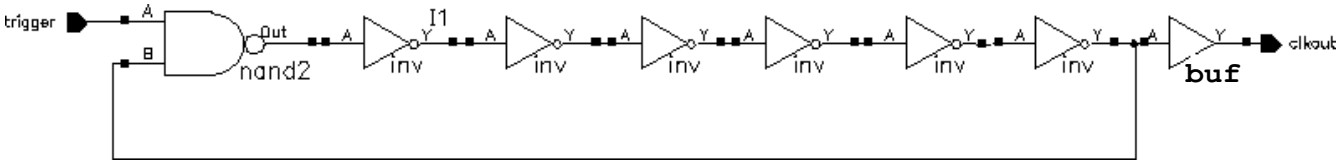
- Proceed with the necessary step and then run the simulation again.
- Record the oscillating frequency and fill in the table on the last page.
- Compare the results obtained in step 6 and step 7.

VII. Use of Buffer

From the results obtained in step 6 and 7, the oscillating frequency of the ring oscillator depends on the load it is connected to. In this part, a buffer will be added to the output and the effect of buffering is evaluated.

Step 8. Build a **new cell** called '**Bclkgen'** under the library **323002myLib**.Step 9.Create a schematic view of **Bclkgen** as shown below.

- Hint: you can copy the schematic view from **clkgen** to **Bclkgen** and then modify it



Step 10. Create its symbol view.

Step 11. Create its layout view.

- Hint: you can copy the layout view from **clkgen** to **Bclkgen** and then modify it
- Perform DRC, LVS and extraction to make sure there is no error.

Step 12. Perform the post-layout simulation.

- Create a new simulation directory for simulation.
- In its test.input file, connect a **0.3pF** capacitive load at its output. You can copy the test.input file used in step 7 for simulation.
- Record the oscillation frequency and fill in the value in table below.
- Compare the results listed below.
- When the simulation has ended**, look into** the **netlist** file. How are they connected? Draw them on the schematic view of Bclkgen and show your drawing to TA

Summary of the simulation results of the ring oscillator.

Results of	Conditions	Oscillating frequency obtained (Hz)
Step 6	Clkgen, without capacitive load	
Step 7	Clkgen, with 0.3pF load	
Step 12	Clkgen, with 0.3pF load plusbuffer inserted.	

The End of Lab6