



ECE/CS 5710/6710 - Lab 2

CMOS Inverter - Physical Design

Pre-lab Assignment: Check for the date on Canvas.

Lab Report: Check for the date on Canvas.

1 Objectives

The goal of this second lab is to teach you how to draw the layout of a CMOS inverter using the Skywater design kit. You will also learn how to verify your design through these DRC and LVS steps and how to extract the layout parasitics for post-layout simulation purposes. For the DRC, LVS and PEX steps, you will use an industrial tool: Mentor Calibre®.

2 Pre-lab Assignment

Answer the following questions and submit a .pdf file through Canvas:

1. What does DRC, LVS and PEX stand for? What are they used for?
2. Draw the transistor schematic of a 2-input NAND gate and give its truth table.
3. How are the transistors in a 2-input NAND gate usually sized? Justify your answer.

3 Introduction to the Tools

In this lab, you will reuse Virtuoso® for the layout as well as electrical simulations. You will also use Calibre® is the suite for full-custom layout verification and parasitics extraction. Although it is coming from another vendor (Mentor, now part of Siemens), this tool is easily integrated into the Virtuoso® environment.

4 Lab Assignment

4.1 CMOS Inverter Layout Drawing

In this part, you will learn how to draw the layout of the CMOS inverter. With this knowledge, you will then be able to draw the layout of more complex logic gates. To start this lab, launch Virtuoso® from your *virtuoso* subdirectory, as you did in the previous lab:

```
./start_virtuoso.sh
```

4.1.1 Layout Environment

R As explained in the previous lab, don't forget to check that in your inverter schematic view, the width of the *pmos* is set to *840nm* and not as a parameter, otherwise, you will have some issues when instantiating it in your layout.

1. Create a new cellview for your inverter Layout. To do so, go to the open your inverter schematic and do: *Launch - Layout XL ...*). Click OK on the first window. For the next window, select layout as the view name, and Layout XL as the desired application as indicated in Fig. 1 (the library and cell name will of course depend on what you chose earlier).

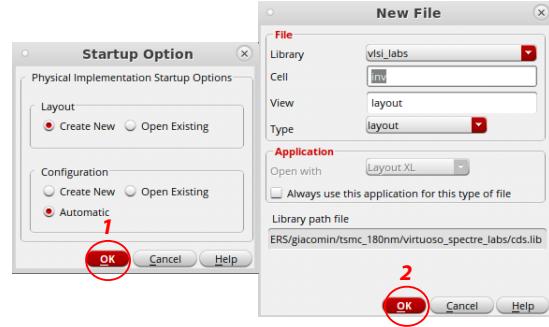


Figure 1: New layout view.

2. The Virtuoso® Layout XL Editor window will appear. All available fabrication layers are listed in the layer sub-window on the left (Fig 2). From here, you can select which current layers are visible (V) or selectable (S). You can also choose to only display the layers currently in use in your current layout but selecting *Used*. Many useful commands can be found in the command toolbar below the layout display area. Most of the shortcuts to save, delete an instance, rotate a component, etc. are located in the menus and tools area. As you saw in class, integrated circuits are manufactured through a lot of process steps and are composed by a large number of planar layers, stacked on top of each others. The layout is composed of the geometric shapes of each layers that corresponds to the actual pattern of those materials in your integrated circuit. Based on the layers you choose for your layout, precise masks will be generated during fabrication. This is why for a given layer (e.g. metal 1), several purpose layers are available (dummy, blockage, drawing).

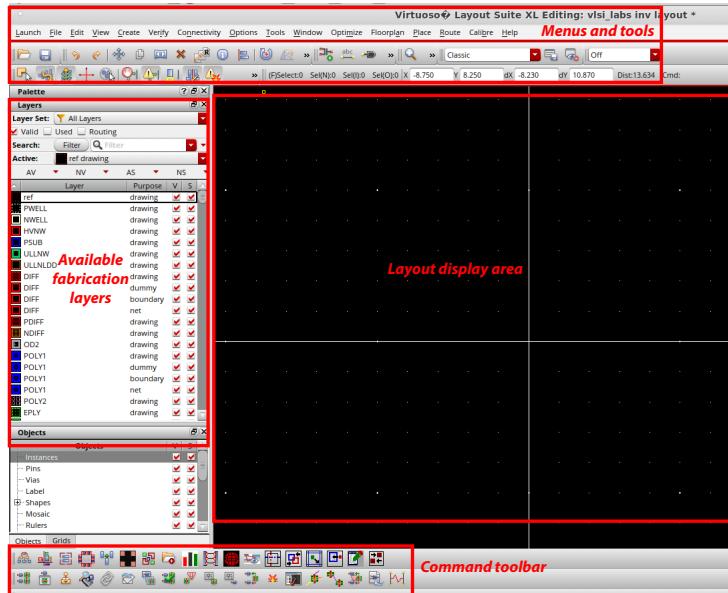


Figure 2: Layout XL window.

R When closing your layout view and opening it again later, Layout L might be used instead. The only difference is that Layout XL contains more functionalities (such as instantiating the cells automatically, the ability to flatten your cells, etc.). To switch, go to: *Launch -> Layout XL*.

As for the schematic view, lots of shortcuts are available in order to draw your layouts. The most important ones are given in Table. 1.

Table 1: Main Layout® shortcuts.

Shortcut	Menu Access	Action
u	Edit -> Undo	Undo previous command
c	Edit -> Copy	Copy selected instance(s)
m	Edit -> Move	Move selected instance(s)
s	Edit -> Stretch	Stretch selected layer
f	Window -> Fit All	Fit the entire design into the layout window
r	Create -> Rectangle	Create a rectangle shape
i	Create -> Instance	Instantiate another design/layout in the current design
o	Create -> Contact	Create a via/contact from the technology
l	Create -> Label	Create a label for your pins
q	Create -> Properties	Show the properties of selected instance
k	Window -> Create ruler	Create a ruler to perform measurements
ctrl+f	N/A	Shows the current level of hierarchy All the instances in the lower levels are shown as red boxes
shift+f	N/A	Shows all hierarchy levels. All the layers are shown

Step 1: Transistors Instantiation

- Instantiate your transistors. To do so, from the layout editor Command Toolbar, choose Generate All From Source (as in Fig. 3), and set it up to generate only the instances and to preserve the device correspondence (mapping between the schematic and layout) and click OK.
- Your transistors should appear on the layout. **If all you can see is a red box with the transistors names, press shift + f.** It will allow you to display deeper levels of the hierarchy (here the several instances).

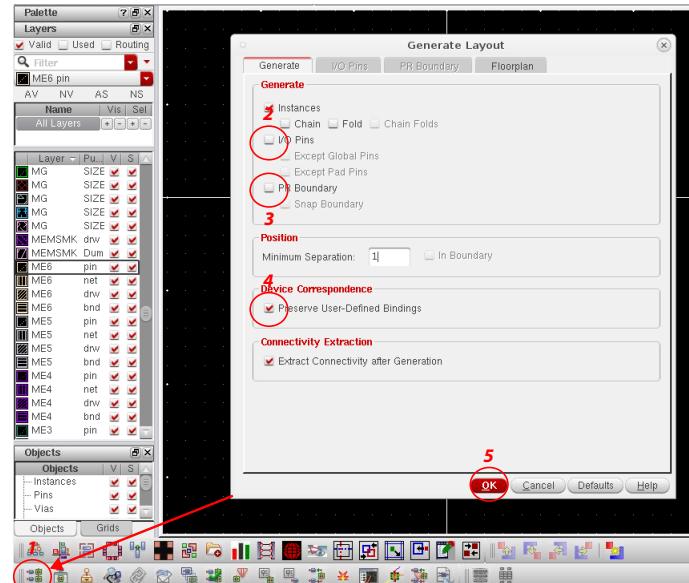


Figure 3: Instantiating the transistors.

- R** Another way of instantiating your transistor is to press **i** and select the appropriate instance from one of the library, as you did for the schematic.

Step 2: Placing your Transistors

Now, let's place the *pmos* transistor closer to the *nmos*. To do so, you need to move it vertically: press **m**, click on the *pmos* transistor and move your mouse in the wanted direction. Here, we want the diffusion regions of our *nmos* and *pmos* to be distant by $.56\mu m$. This distance is arbitrary for now, you don't have to necessarily respect it for all your designs (the important distance to respect is the one between the power supply lines, which will be explained in Step 6). To do so, we need to use some rulers:

- Press **k** to draw a ruler. To draw, select starting and ending points of the ruler by left-click after pressing **k**.
- You can always remove the existing rulers by pressing **Shift+k**.
- Use the ruler to check that the distance between both transistors is respected. If not, you need to move your transistors accordingly.
- Use the ruler again to make sure that the gate of both transistors are aligned. To do so, you can just place the ruler on the side of one of the gate and verify the alignment with the other gate.

Step 3: Drawing the Poly-Silicon Gate Connection

Now, you need to connect the gate of the *nmos* and *pmos* together. To do so, you will draw a poly-silicon wire to connect both gates.

- Select the poly-silicon layer on the drawing layers panel on the left by choosing *poly – drawing*, as depicted in Fig. 5.
- Press **p** and use the left mouse button to start and end the wire. For the starting point, select the top edge of the *nmos* gate. For the ending point, select the bottom edge of the *pmos* gate.

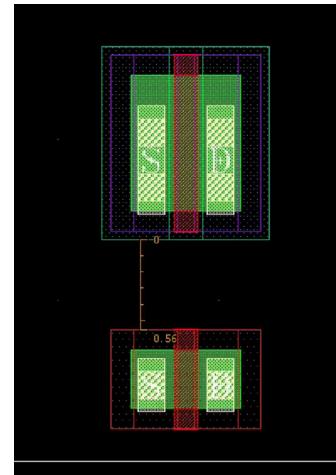


Figure 4: Using the ruler.

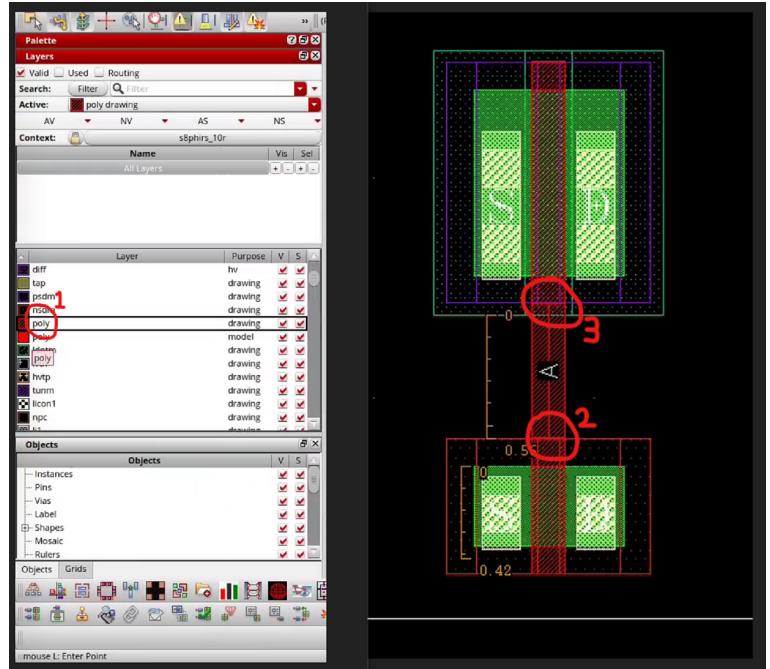


Figure 5: Poly-Silicon wire creation.

- R** It is possible to make the wire to break under a 90 degree angle if you use the left mouse button multiple times. However, at advanced technology nodes, it is generally harder to manufacture such kind of break lines due to the small size of the wires.

Step 4: Placing the Poly-silicon to Metal1 Contact (Via)

- To draw the poly-silicon to metal 1 contact, multiple vias are necessary as there are multiple layers to traverse. Press **o** and select $PY1L_c$ as the Via Definition and then press *Hide*.
- Place your via on the poly-silicon wire as shown in Fig. 6.
- This via will travel between the poly layer and the local interconnect layer. We must place one more via to get to the metal. Repeat the previous step with the $L1M1_c$ Definition and place it directly on top of the previous via. 6.

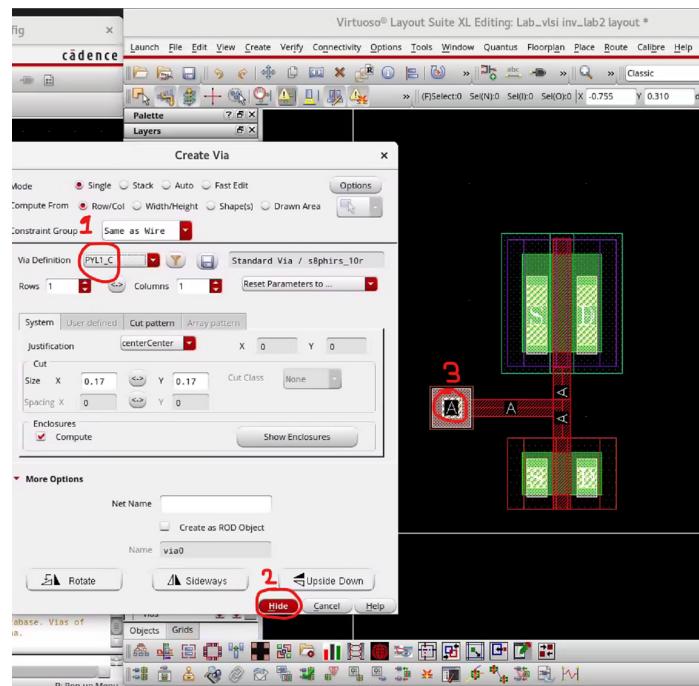


Figure 6: Poly-silicon-Metal1 via creation.

Step 5: Creating the Metal 1 Wire

Now, you need to connect the drain of your *nmos* and *pmos* together to define your inverter output. To do so:

- Select the first level local interconnect layer on the drawing layers panel on the left by choosing *l1-drawing*. 7.
- Press **p** and use the left mouse button to start and end the wire. For the starting point, select the top edge of the *nmos* drain metal 1 area. For the ending point, select the top edge of the *pmos* drain metal 1 area.

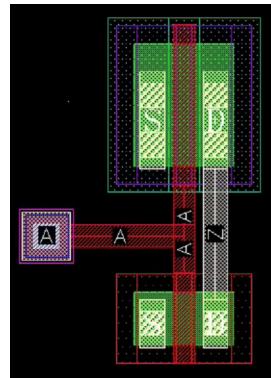


Figure 7: Local interconnect wire creation.

Step 6: Creating the Metal 1 VDD and GND Supply Lines

Now, you need to connect the drain of your *nmos* and *pmos* together to define your inverter output. To do so:

- Select MET1-drawing as a layer. Press **r** and draw a rectangle of random dimensions, as depicted in Fig. 9. Then select the metal 1 rectangle, click on **q** and change its width and height to $4.6\mu m$ and $0.29\mu m$ respectively.



You will learn later that integrated circuits are composed of many rows where each row contains standard cells (INV, MUX, etc). Therefore, all of your standard cells need in this lab and the next ones need to have the same height. For this technology, you need to ensure your supply line have an height of $0.29\mu m$. The standard cell height being $2.72\mu m$, you need to check that the spacing between both metal supply lines is $2.43\mu m$, as depicted in Fig. 8. In addition, the width of your cell needs to be a multiple of $4.6\mu m$. Since the inverter is thinner than this, we chose to use $4.6\mu m$.

- Place the metal 1 rectangle on top of the *pmos* transistor, with a distance of $0\mu m$ from the *NWELL* layer as shown Fig. 9.
- Place a similar rectangle on the bottom of the *nmos*. To do so, press **c** (for copy), select the first metal 1 rectangle and move the second rectangle on the bottom, with a distance of $0.06\mu m$ from the *nsdm* (red) layer..

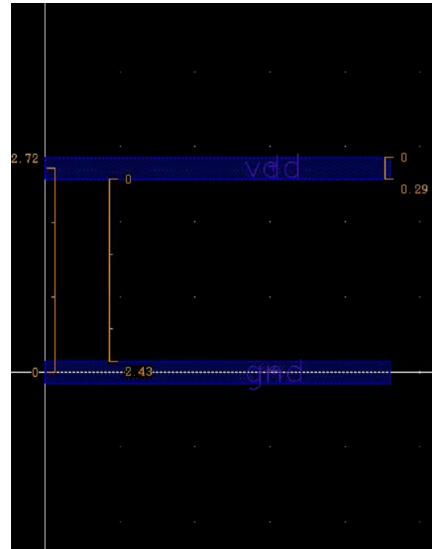


Figure 8:
Standard
cell height
requirements.

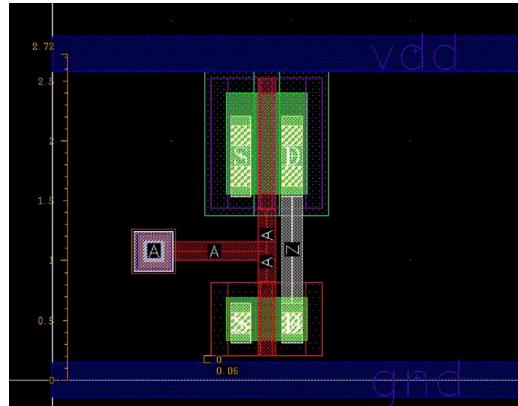


Figure 9: Metal 1 rectangle creation and placement for the power supply lines.

Step 7: Finishing the Metal Connections

Connect the source of the *nmos* and *pmos* to the appropriate power supply lines through the local interconnect layer. To do so:

- Press **p** and draw a line from the source of the *pmos* to the *vdd* power line. Double-click to confirm the line.
- Place a via that will connect the local interconnect and the metal 1 layers.
- Repeat the previous steps to connect the source of the *nmos* to *gnd*.

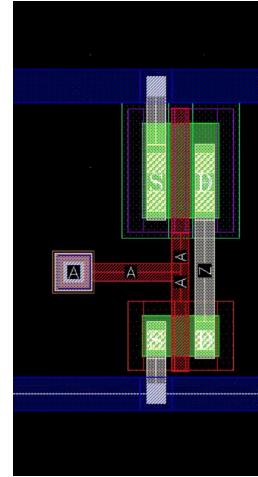


Figure 10: Connecting the terminals to the supply metal lines.

Step 8: Creating the Bulk Taps

- Now, we need to create to connect the bulk of the *nmos* and *pmos* to *VDD* and *GND* respectively.
- Extend some local interconnect lines to the left from the *vdd* and ground pins on the *pmos* and *nmos*.
- At the end of each line, place a via that connects the tap and local interconnect layers (*TPL1_C*).
- The *VDD* tap will need to be surrounded by a rectangle of the *nsdm* layer, and a rectangle of the *nwell* layer.
- The *GND* tap will need to be surrounded by a rectangle of the *psdm* layer.

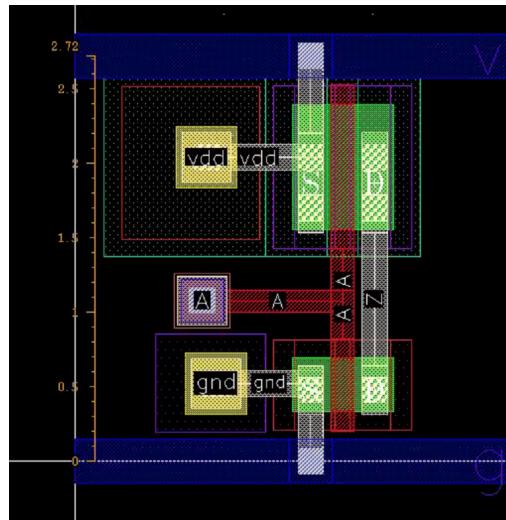


Figure 11: Creating the bulk taps.

Step 9: Creating the Pins/Ports

Now, we need to create the pins so the LVS tool will be able to compare the schematic and the layout connectivity.

- We need a pin for VDD, GND, A, and Z (Z will also need a via to metal first).
- To create a pin, go to Create -> Pin in the toolbar.
- Assure the name is the same as the net it will be place on, select rectangle, and assure the io and signal types are correct. (In this example, "A" is an input signal.)
- Click *Hide*, and draw the pin rectangle over the targeted via (assure met1 - pin in the selected layer)
- Click on *Hide* and place them on your layout.
- In this specific case with the "A" pin, there is only a metal via beneath the pin. You will need to place a metal 1 - drawing layer rectangle beneath the pin for it to be recognized. 12

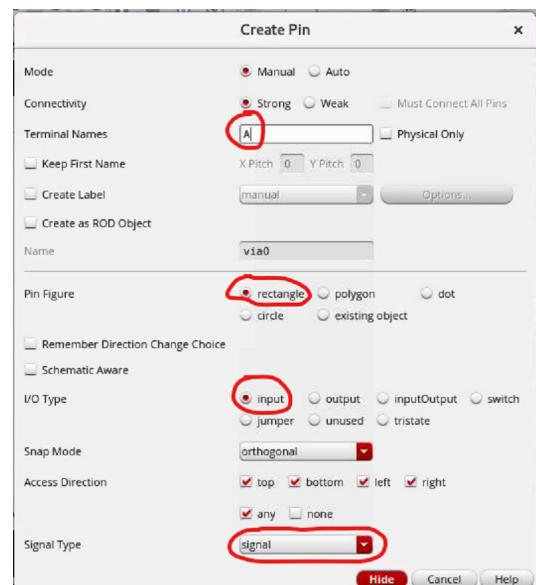


Figure 12: Creating Pins.

Step 10: Labeling

Finally all the ports need to be labeled.

- Select the MET1-pin layer and go to: Create -> Label... (or press l).
- Write in a pin name (or all of them separated by spaces). They need to match the pins of your schematic.
- Set the label size to something resonable (.3)
- Click on *Hide* and place them on your layout.
- Assure the name is the same as the net it will be place on, select rectangle, and assure the io and signal types are correct. (In this example, "A" is an input signal.)
- Click *Hide*, and draw the pin rectangle over the targeted via (assure met1 - pin in the selected layer)
- Click on *Hide* and place them on your layout.
- Your layout should be (mostly) complete now and look like Fig. 13. In the next part, you will verify it and correct the potential errors.

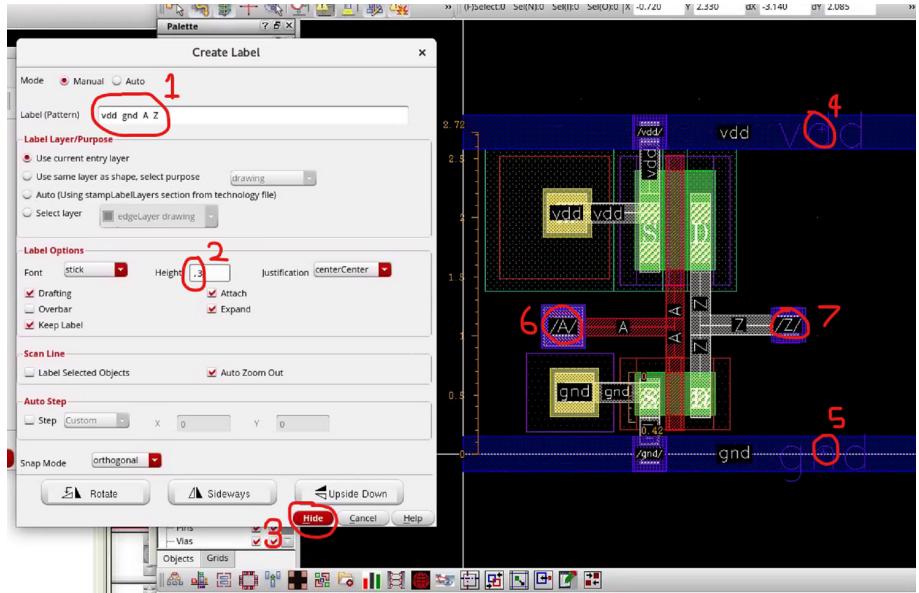


Figure 13: Final inverter layout.

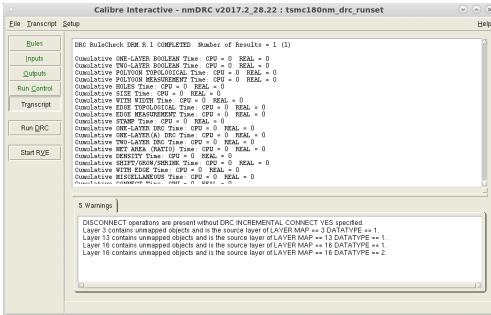


Figure 14: Calibre nmDRC main window.

4.2 CMOS Inverter Layout Verification

In this part, you will learn how to perform all the necessary layout verification steps (DRC and LVS) and how to extract the parasitic values from your layout (PEX).

4.2.1 DRC Verification

Even if you followed the previous steps for the layout of your inverter, you will probable get some DRC errors. (*Hint:* 3 of these errors can be solved by extending the *diff*, *poly*, and *l1l* layers of the nfet downwards $.06\mu m$) The goal of this part is for you to understand those errors and to correct those by yourself. To perform the DRC step:

1. Launch the Calibre DRC tool: from the layout view: *Calibre -> Run nmDRC....*
2. The Calibre nmDRC window now appears (Fig. 14). Normally, you should not have to modify anything on the nmDRC window except for setting the runset file to:

```
/uusoc/facility/cad_common/skywater-src-nda/s8/V2.0.1/DRC/Calibre/s8_drc_runset
```

3. You can take a quick look at the different DRC settings to understand better how the tool work. For instance, if you click on *Rules*, you can see what calibre rule file is selected to check your layout, as well as the different layers derivations.

4. Then Click on Run DRC, wait a little bit and the potential DRC errors will be displayed. As shown in Fig. 15, a new window should appear. On the left pane are listed all the different rules. You can click on *Results* to order them. If some rules have a red cross in front of them, it means your design does not respect some of the foundry rules (the goal is then to only have green symbol in front of every rules, otherwise, the foundry will not

be able to manufacture your design). The rule description tells you what the rule require to be enforced. In this example, it says that the minimum metal 1 area should be at least $0.202\mu m^2$. The error occurrence tells you how many occurrence of the same error you have in your design. The other pane tells you where the error is located. If you want to directly locate the error on your layout view, you can double click on the error number in the error occurrence pane (1 2 3 4 in Fig. 15). Once this error is corrected, you can go back to the DRC results window, select another error, locate it and so on and so forth.

5. Once you have corrected some errors, go back to the Calibre nmDRC window, click on Run DRC and check again how many error you get. You need to repeat this step until there is no DRC error at all in your design. You don't have to correct all the errors at once before running DRC again.



It is often a good practice to correct some errors and run the tool again to see your progress (sometimes, by correcting some errors you might create new ones so it's better to go step by step).

When closing the DRC window, you will be asked if you want to save your changes to the runset file. Select *No* and do the same when closing the LVS and PEX tools in the rest of the lab.

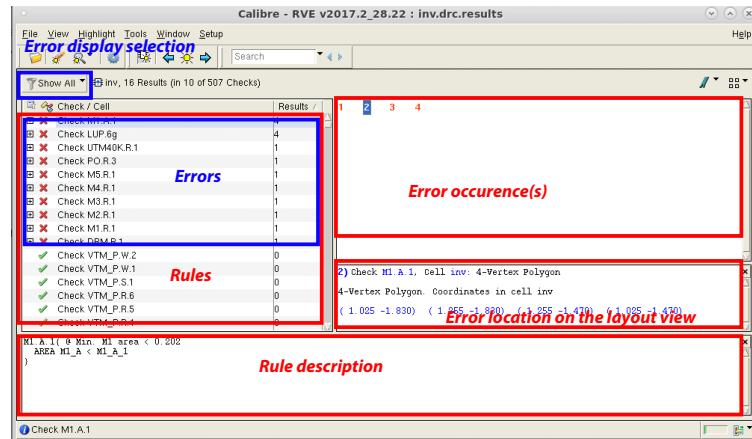


Figure 15: DRC result window

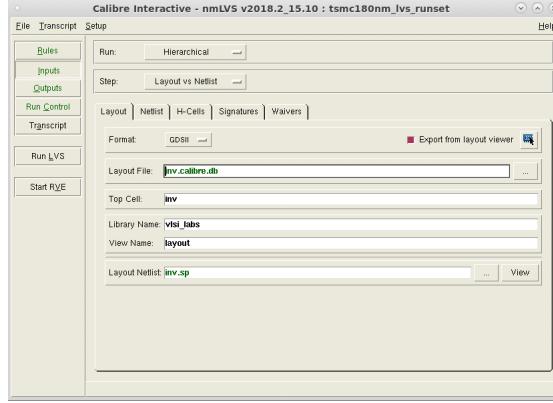


Figure 16: Calibre nmLVS main window.

4.2.2 LVS Verification

To perform the LVS step:

1. Launch the Calibre LVS tool: from the layout view: *Calibre -> Run nmLVS....*
2. When the *Load Runset File* window appears, browse and select:
3. The Calibre nmLVS window now appears (Fig. 16). As for the DRC, you should not have to modify anything on the nmLVS window except setting the LVS runset file to:

```
/uusoc/facility/cad_common/skywater-src-nda/s8/V2.0.1/LVS/Calibre/s8_lvs_runset
```

4. Then Click on Run LVS, wait a little bit and the potential LVS errors will be displayed. A new LVS result window should appear. If your layout is correct, you should see the green smiley and icons as shown in Fig. 17. If not, it means you have some errors. On the left pane are listed the different errors as follows:

- **Extraction results** will list the pin related errors. For instance, if you define the same pin on two different nets, or if your design is missing the power supply or the ground pin, errors will be reported.
- **Comparison results** will list the mismatches between your schematic and your layout. For instance, if you forget to connect the source of the *nmos* of your inverter to the ground, an error will be reported.

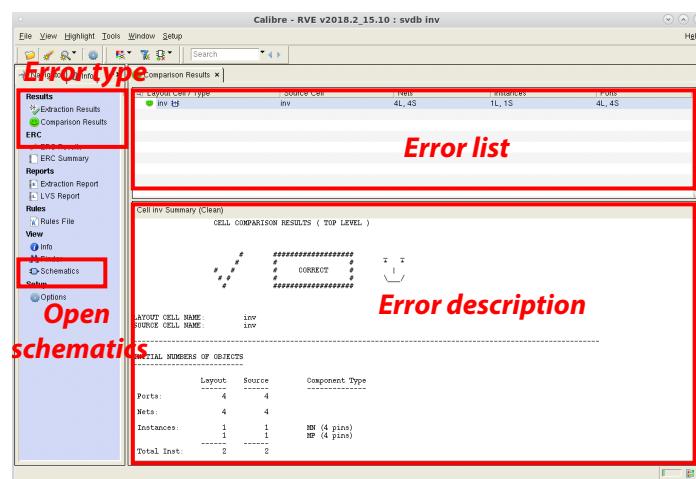


Figure 17: LVS result window.

To help you correct some LVS errors, Fig. 18 shows an inverter design with some errors. In case you have some extraction errors, click on *Extraction Results* and you will see the different errors in the error list. In Fig. 18 (a), the error is caused by an absence of the power supply pin. To display the comparison errors, click on *Comparison Results*. In Fig. 18 (b), there are two errors: the nets A and Z can not be found in the layout since the pins have not been specified either. In some cases, an error could be caused by a bad connection (i.e. the input and output of the inverter could be connected together by a metal line). In the error description panel, you can click on the net names (A and Z in this case) and the associated net will be highlighted on either the source (schematic) or layout of your design.



Sometimes, it can be helpful to click on *Schematics* on the LVS result window. It will open both the schematic of your design and the equivalent schematic of your layout. From here, you can visualize the difference and correct them.

As before, it is a good practice to correct some errors and run the tool again to see your progress.

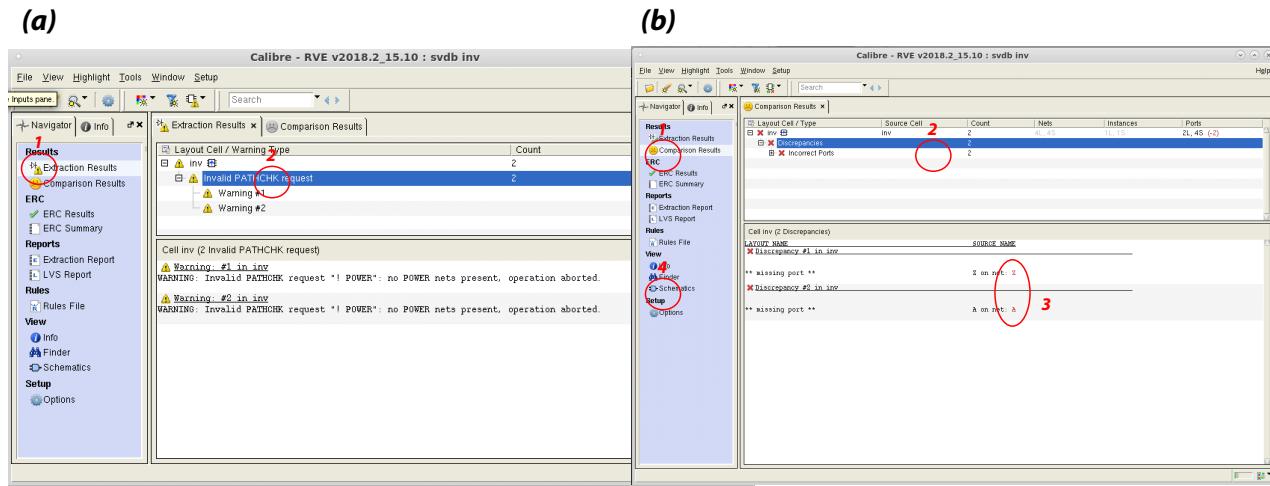


Figure 18: LVS result window for: (a) Extraction results; (b) Connectivity results.

Checkpoint

Please call an assistant and show him that your inverter design pass both the DRC and LVS with no errors.

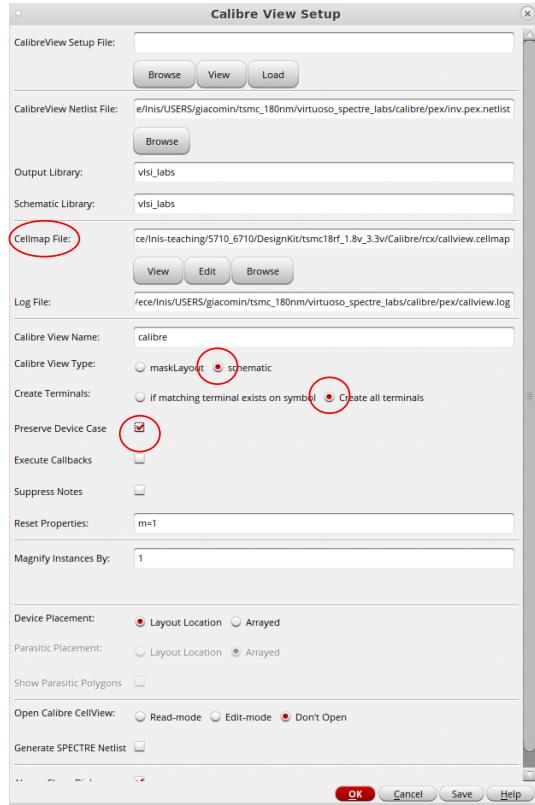


Figure 19: PEX options.

4.2.3 Parasitic Extraction (PEX)

- ! The PEX step can only be done if your design is LVS compliant.

To perform the PEX step:

1. Launch the Calibre PEX tool: from the layout view: *Calibre -> Run PEX....*
2. The Calibre PEX window now appears. Again, you normally should not have to modify any settings except setting the runset file to:

```
/uusoc/facility/cad_common/skywater-src-nda/s8/V2.0.1/PEX/xRC/Calibre_PEX_runset
```

3. Click on Run PEX.
4. Then, a window will appear as depicted in Fig 19. Verify that the options (the important ones are highlighted in red) are the same as in Fig. 19 (some fields, such as the library settings can of course differ depending on the name you chose).
5. Click on *OK* at the bottom of the window.
6. You should get a pop-up window as shown in Fig. 20 with no warnings.
7. Once this is done, go back to the Library Manager window. You should be able to see a new view for your inverter cell in your working library named **calibre**. When you open it, you can see the resistances and capacitances modeling the parasitics of your layout.



Do not forget to check and save your *calibre* view (you might get some warnings but those are fine) before closing it. Otherwise, you will encounter an error when doing your post PEX simulation.

Checkpoint

Please call an assistant and show him that the *calibre* view for your inverter has been properly generated.



Figure 20: Successful PEX extraction.

4.3 CMOS Inverter Post Layout Simulation

With technology scaling, the RC delay is now a big issue due to the increased parasitics. This is why it is important to perform a post layout simulation which takes into account the parasitics (after the PEX step).

1. Go to the library manager and open the schematic view of your inverter transient testbench (*inv_testbench_tran*).
2. Launch ADE L. From the ADE window, open up the state you previously defined in the previous part: click on *Session -> Load State* A new window will appear, simply click on *Ok*.
3. Now, all the settings you previously defined (parameters values, analysis type, display output) should be there.

4. To take into account the parasitics of your layout, we need to tell ADE to consider the parasitics in the simulation. To do so, go to the ADE window and go to: *Setup -> Environment* and add the **calibre** view in first in the *Switch View List*, as shown in Fig 21.
5. Click OK and launch the simulation.
6. Observe the output curves to check that your inverter behaves properly as well as how the delay changed compared the the simulation you ran before considering the parasitics.

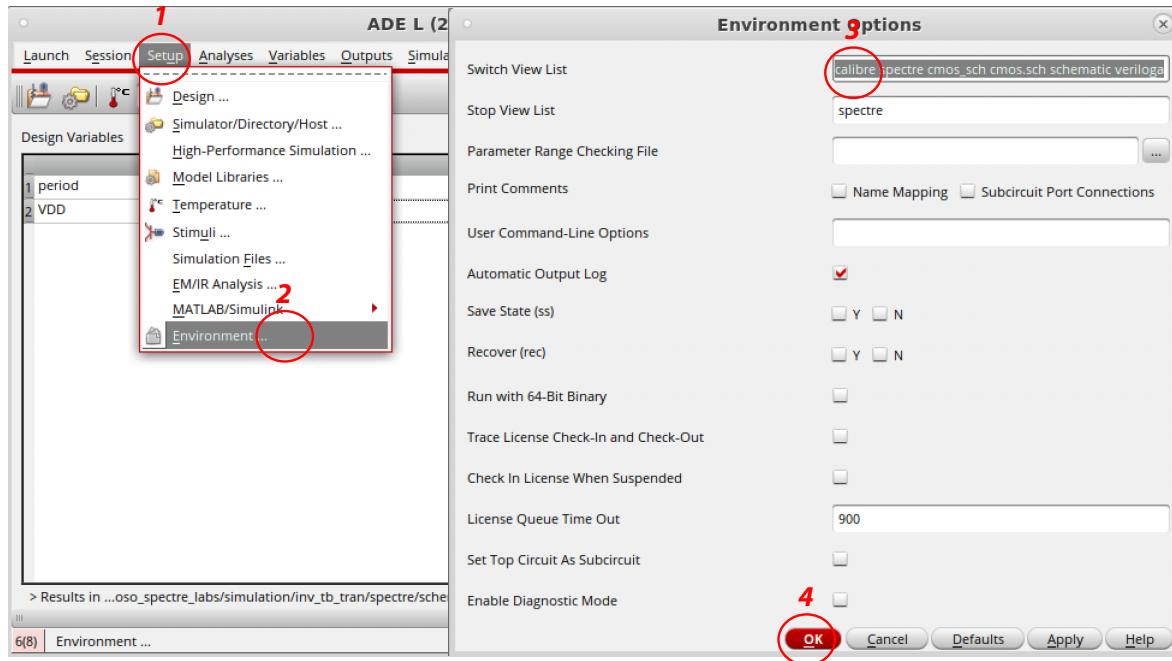


Figure 21: ADE L environment options.

Assignment

Report the rising, falling and propagation delays of your inverter after the PEX step. How did the delay change when compared to the pre-layout simulation? Why?

5 Assignment and Checkpoint Summary

Write a report and answer the assignments asked during the lab, which are summarized below. Do not forget to validate the checkpoints, summarized below as well, by an assistant before the end of the lab.

Assignments

1. Report the rising, falling and propagation delays of your inverter after the PEX step. How did the delay changed when compared to the pre-layout simulation? Why?
2. Provide a screenshot of the DRC result window as well as the LVS result window for the inverter, showing the correctness of your design.

Checkpoints

1. Please call an assistant and show him that your inverter design pass both the DRC and LVS with no errors.
2. Please call an assistant and show him that the *calibre* view for your inverter has been properly generated.