

Lab #2. Guideline

In this lab, we will study an RC model to evaluate the propagation delays of a CMOS circuit. The R and C can be calculated from the measured data of simulations. If we can ignore the wiring effects (the resistance and capacitance of interconnecting wires), the R is the overall effective channel resistance of the pull-up (or pull-down) network and the C is the total capacitance at the output node. The output node C consists of two capacitors. One capacitor is due to the diffusion areas (i.e., drain areas) and diffusion peripherals of the driver and the other comes from the input capacitance (i.e., gate capacitance) of the receivers. **(Please, refer to Lab #1 in the Lab book)**

How to calculate unknown parameters:

- Unknown parameters: 4

R_{sn} - Effective channel sheet resistance for nMOS transistors,

R_{sp} - Effective channel sheet resistance for pMOS transistors,

C_g - Gate capacitance per unit area, and

C_d - Drain capacitance per unit area.

Basic equation:

time delay = resistance (R) * capacitance (C), or

time delay = sheet resistance * L (length) / W (width) * unit capacitance * capacitor area,

where sheet resistance (R_s) and unit capacitance (C_d, or C_g) are unknown values,

time delay, transistor's length & width, and capacitor area are measured.

Measured parameters per simulation: 2

T_{plh} (low to high propagation delay), and

T_{phl} (high to low propagation delay).

#of Needed equations to solve 4 unknown values: more than 4.

Thus, minimum # of simulations: more than 3 (2 equations per simulation)

Set up equations:

- 1st simulation without load

Eq. 1. - $T_{plh} = R_{sp} * (L/W) * C_d * (\text{drain areas})$,

Eq. 2. - $T_{phl} = R_{sn} * (L/W) * C_d * (\text{drain areas})$,

- 2nd simulation with load capacitor

Eq. 3. - $T_{plh} = R_{sp} * (L/W) * (C_d * (\text{drain areas}) + \text{additional load cap})$,

Eq. 4. - $T_{phl} = R_{sn} * (L/W) * (C_d * (\text{drain areas}) + \text{additional load cap})$,

- 3rd simulation with inverter load

Eq. 5. - $T_{plh} = R_{sp} * (L/W) * (C_d * (\text{drain areas}) + C_g * (2 * \text{gate areas}))$,

Eq. 6. - $T_{phl} = R_{sn} * (L/W) * (C_d * (\text{drain areas}) + C_g * (2 * \text{gate areas}))$,

where

drain areas (PMOS & NMOS) = $W_{PD} * L_{PD} + W_{ND} * L_{ND}$, and

gate areas (PMOS & NMOS) = $W_{PG} * L_{PG} + W_{NG} * L_{NG}$,

where, W_{PD} is driver PMOS drain width,

L_{PD} is driver PMOS drain length,

W_{ND} is driver NMOS drain width,

L_{ND} is driver NMOS drain length,

W_{PG} is receiver PMOS gate width,

L_{PG} is receiver PMOS gate length,

W_{NG} is receiver NMOS gate width, and

L_{NG} is receiver NMOS gate length.

Procedure Overview

- 1: Drawing an inverter using Cadence Virtuoso

Create the layout of an inverter ($L = 2 \lambda$, $W = 4 \lambda$) Check design rule

Extract inverter's netlist.

- 2: Download 1st spice input file on the Lab web-site.

Add your extracted netlist to 1st spice input file.

This spice input file includes three identical inverter instances and an input pulse. The propagation delay time of only 3rd inverter is measured.

Set up the following equations:

$$T_{plh} = R_{sp} * (L/W) * C_d * (\text{drain areas})$$

$$T_{phl} = R_{sn} * (L/W) * C_d * (\text{drain areas})$$

- 3: Download 2nd spice input file on the Lab web-site.

Add your extracted netlist to 2nd spice input file.

This spice input file includes three identical inverter instances, a capacitor (0.1pF), and an input pulse-The propagation delay time of only 3rd inverter is measured.

Set up the following equations:

$$T_{plh} = R_{sp} * (L/W) * (C_d * (\text{drain areas}) + C_{load})$$

$$T_{phl} = R_{sn} * (L/W) * (C_d * (\text{drain areas}) + C_{load})$$

- 4: Download 3rd spice input file on the Lab web-site.

Add your extracted netlist to 3rd spice input file.

The spice input file has four identical inverter instances and an input pulse. The propagation delay time of only 3rd inverter is measured.

Set up the following equations:

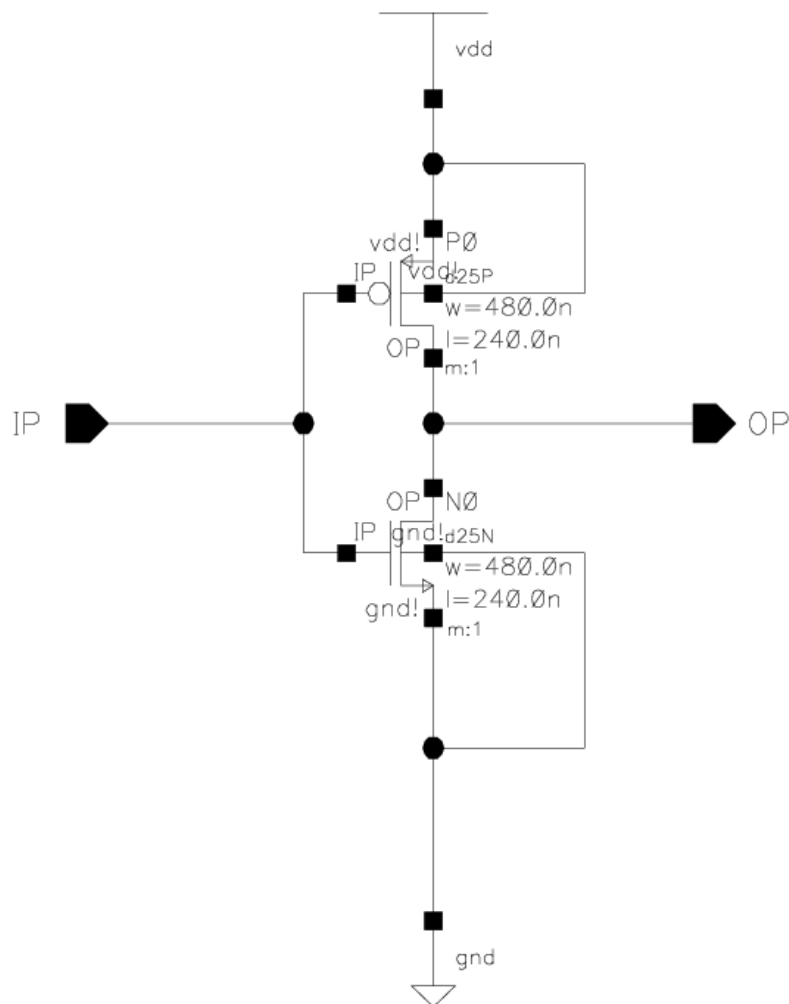
$$T_{plh} = R_{sp} * (L/W) * (C_d * (\text{drain areas}) + C_g * (2 * L * W))$$

$$T_{phl} = R_{sn} * (L/W) * (C_d * (\text{drain areas}) + C_g * (2 * L * W))$$

5. Solve all the equations to find C_g , C_d , R_{sp} and R_{sn} .

Step 1: Drawing an Inverter using Compose (W=300nm, L=50nm)
- refer to Lab #1.

1. Make a New Library (named “**lab2**”)
2. Make a Cell (named “**inverter**”) under lab2 library.
3. Draw the following schematic.



Step 2: Drawing an Inverter using Virtuoso (W=300nm, L=50nm)

- Creating a New Cellview under lab2 library:

- Create a new cellview under the lab2 library.

See Schematic Entry with Composer to find out how to create a new library.
In the Command Interface Window (CIW):

- a) Select **File -> New -> Cellview**.
- b) Choose the library under which you would like to create the new cell view.
- c) Enter Cell Name: **My_layout** (for the tutorial)
- d) Choose **Virtuoso** as the **Tool**. View name should be **layout**.
- e) Click OK.
A blank virtuoso window should open.

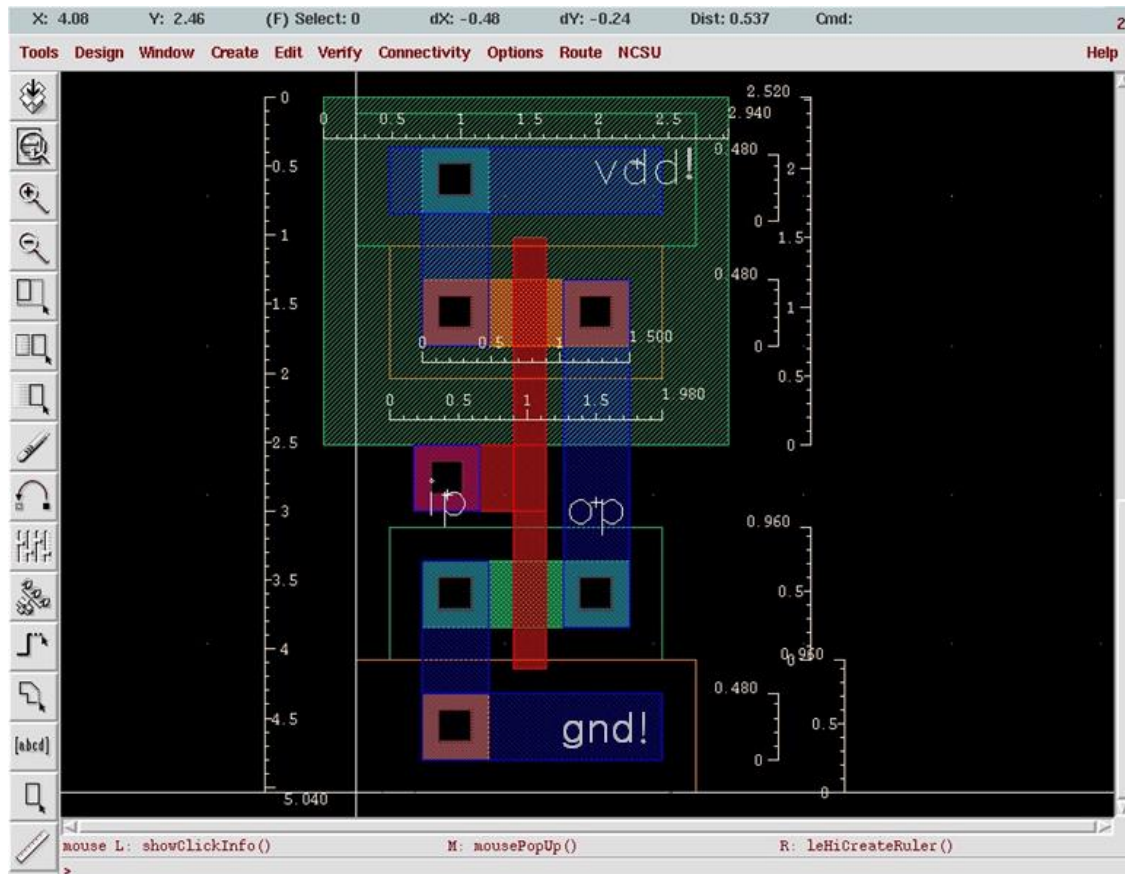
- Setting Display Parameters:

- a) **Options -> Display**.
- b) Set the following options:
 1. X Snap Spacing: 0.0025
 2. Y Snap Spacing: 0.0025

- Click "Save To" to set them permanently.

- Layout an Inverter:

- You should FreePDK45™ 45nm design rule
(<http://www.eda.ncsu.edu/wiki/FreePDK45:Contents>) for 45nm 1.1V processing
(min. length = 50nm).
- The inverter consists of three parts -- p-transistor, n-transistor, and connections.
- This inverter is probably almost minimum size; therefore, the n- and p-transistors have the same W/L.



- Layout of P-transistor with $L=50\text{nm}$ and $W=300\text{nm}$.
In this tutorial, we are using the Nwell process. Thus, the substrate will be p-substrate. We will create a pmos transistor first. To do that we need an Nwell in which the pmos transistor will be formed.
 - Draw the well
 - a) Select the **n-well** layer from the **LSW** window
 - b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) Using your mouse, draw the n-well on the cellview to be **2.94 wide by 2.52 tall**.
 - Draw the p- and n-select regions for the p transistor
 - a) Select the **pselect** layer from the **LSW** window; we will draw the pselect enclosing the transistor
 - b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) Using your mouse, draw the pselect on the cellview; **1.98 wide and 0.96 tall; its left and right edges should be 0.48 away from well edges.** The pselect should be placed within the n-well, even if the size should vary. (you can use the **Edit->move** command to move the layer)
 - Draw Diffusions
 - a) Select the **active** layer from the **LSW** window; we will draw the active region of the p-device

- b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) Using your mouse, draw the pactive on the cellview to be **1.5 wide by 0.48 tall** - it should be enclosed by the pselect by 0.24 and by the nwell by at least 0.72
- **Draw Poly**
 - a) Select the **poly** layer from the **LSW** window
 - b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) Using your mouse, draw the poly on the cellview to be ~~0.24 long by 1.32 wide.~~
The poly should be placed at the center of the p-island. The poly should extend over the p-island by 0.30 μm .
- **Place Contacts**
 - a) Select the **contact** layer from the **LSW** window
 - b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) ~~Using your mouse, draw the contact on the cellview to be 0.24 long by 0.24 wide.~~
The contact should be placed at both sides of the pactive. Repeat the above process for the second contact.
- **Layout of N-transistor with $L=0.24\ \mu\text{m}$ and $W=0.48\ \mu\text{m}$.**
 - **Draw the nselect**
 - a) Select the **nselect** layer from the **LSW** window
 - b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) ~~Using your mouse, draw the nselect on the cellview to be 1.98 wide by 0.92 tall~~
 - **Draw the N active region**
 - a) Select the **nactive** layer from the **LSW** window
 - b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) ~~Using your mouse, draw the nactive on the cellview to be 1.50 wide by 0.48 tall; it should be enclosed by the nselect by 0.24 around its edges. Also, it needs to be at least 0.72 away from the n-well (= 6 x lambda; MOSIS rule 2.3 for DEEP).~~
 - **Extend Poly**
 - a) ~~Extend the **poly** layer created for the p-transistor over the n-island, making sure that the poly extension over n-island is 0.30. (you can use the **Edit->Stretch** command to stretch the poly)~~
~~The poly should be placed at the center of the n-island.~~
 - **Place Contacts**
 - a) Select the **contact** layer from the **LSW** window

- b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) ~~Using your mouse, draw the contact on the cellview to be 0.24 long by 0.24 wide.~~
~~The contact should be placed at both sides of the p-island. Repeat the above process for the second contact.~~
- **Layout necessary connections**
 - **Draw the well-contact**
 - a) Select the **nselect** layer from the **LSW** window; we will draw the nselect enclosing the substrate (vdd) contact for the P transistor.
 - b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) ~~Using the mouse, draw the nselect on the cellview; at least 0.96 tall and wide, as it will have to enclose the contact n-active by at least 0.24, which in turn enclose the contact by at least 0.12.~~
~~The nselect abuts directly to the pselect of the P transistor, but they should not overlap. Also, both selects should be placed within the n-well.~~
 - d) Select the **nactive** layer from the **LSW** window; we will draw the body contact region of the p device.
 - e) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - f) ~~Using your mouse, draw the nactive on the cellview to be at least 0.48 wide by 0.48 tall~~
 - g) Add a contact in the center of the well-contact island.
 - **Draw the Substrate-contact**
 - a) Select the **pselect** layer from the **LSW** window; we will draw the nselect enclosing the substrate (vss = ground) contact for the N transistor
 - b) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - c) ~~Using the mouse, draw the pselect on the cellview; at least 0.96 tall and wide, as it will have to enclose the contact p-active by at least 0.24, which in turn enclose the contact by at least 0.12.~~
~~The nselect abuts directly to the nselect of the N transistor, but they should not overlap.~~
 - d) Select the **pactive** layer from the **LSW** window
 - e) Select the **Create->Rectangle** (or choose the Rectangle icon from the side toolbar).
 - f) Using your mouse, draw the n-island on the cellview to be 0.48 wide by 0.48 tall; it must be enclosed by the pselect by 0.24.
 The p-island should be placed at least 0.48 μm ($= 4 \times \lambda$, MOSIS rule 2.5 DEEP) below the n-transistor.
 - g) Add contacts in the center of the substrate-contact island.
 - **Metal Connections**
 - a) Connect the source of p-transistor to the well-contact using the metal 1 layer.
 - b) Connect the source of n-transistor to the substrate-contact using the metal 1 layer.
 - c) Add a contact to the gate (poly)

- Add Pins:

You need to add pins for the **input** (named ip), **output** (named op), **vdd**, and **vss** in order to pass **DRC**.

Creating I/O Pins

Once you have finished creating the layout, the next step is to add the I/O pins of your circuit. It is necessary to add the **vdd!** and **vss!** connections to your circuit for the purpose of DRC. The following is a procedure to add I/O pins to your circuit:

Before you start, you have to have a **layout** (view name: layout) in your library.

- From your Layout window:

1. Choose **Create->Pin...** from the menu. **The Create Pin form will appear.**
2. If the form is titled "**Create Shape Pin**", choose "**sym pin**" under the **Mode** option.
3. Enter a **TerminalName**(the name of your pin).

- **NOTE:** For **HSPICE**, there is a node name limit of **16 characters**.

4. Make sure that the "**Display Pin Name**" option is selected.
5. Specify the "**I/O Type**" as either **input**, **output**, or **inputoutput**.
(**vdd!** and **gnd!** are **"jumper" type**)
6. Specify the Pin Type as either **Metal1_T**, **Metal2_T**,... depending on which is the top layer at the place that the pin is to be inserted.
7. Specify the **Pin Width** to the desired pin width (the pin is square).
8. Move the mouse to specify where the pin and the label should be placed.
9. Repeat the above process (1-7) for all the pins in your circuit.

- Design Rule Check (DRC)

DRC is used to check that all process-specific design rules (such as spacing) have been met.

Design Rule Check (DRC)

There are process-specific design rules that describe how close layers can be placed together and what the sizes of the areas can be. These rules are given the minimum requirement to avoid a catastrophic failure of your circuit due to fabrication faults. You can use the [MOSIS SCMOS design rules](http://www.mosis.org/Technical/Designrules/scmos/scmos-main.html#tech-codes) (<http://www.mosis.org/Technical/Designrules/scmos/scmos-main.html#tech-codes>) as a guideline. The design rules are different for different processes.

The following is a procedure to perform design rule check (DRC) for a layout. DRC outputs any violations of the design rules for your technology process. This step is important because the violation of any design rules would cause the fabricated chip to not function as desired.

Before you start, you have to have a **layout** (view name: layout) in your library.

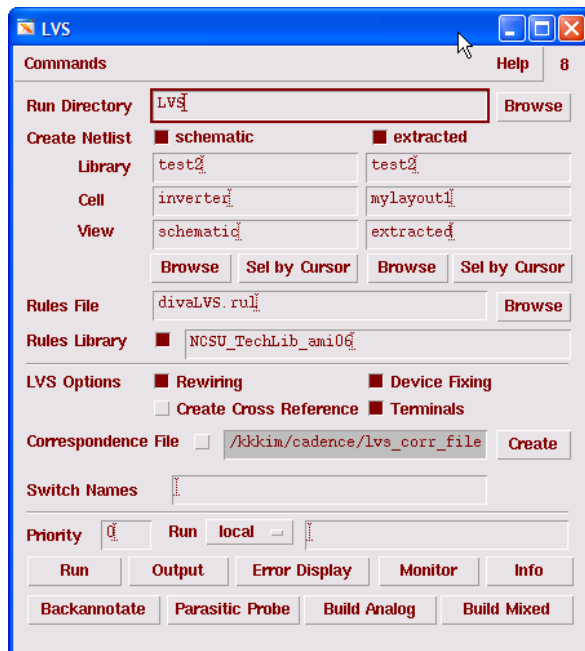
- From your Layout window:
 1. Choose **Verify -> DRC** from the menu. The Verify DRC form will appear.
 2. Set the Switch Names field. This switch name depends on the process you're running.
 - For the TSMC 0.25 um and other processes: Leave as default.
 3. Click **OK** to run DRC.
 - If your design has violated any design rules, DRC will reports the errors in the CIW.
 - Errors are indicated by the markers (white color) on the circuit.
 - You may then proceed to correcting the errors according to the design rules.

Extracting the Layout

Extract your layout by choosing **Verify → Extract**. In the new Extractor window note that the rules file **divaEXT.rul**. This is another rules file of the 'NCSU_TechLib_tsmc25d' technology library which is used to extract the layout. Press **Ok** to extract your layout. Another view (like schematic, layout, and symbol) of the cell inverter by the name **extracted** is created. You can verify this in the Library Manager. After running this process, you should get '**0 errors**' in the CIW window.

Step 3: Layout-versus-Schematic Check (LVS Check)

In Layout Editor(Extracted Layout), Click **Verify -> LVS**

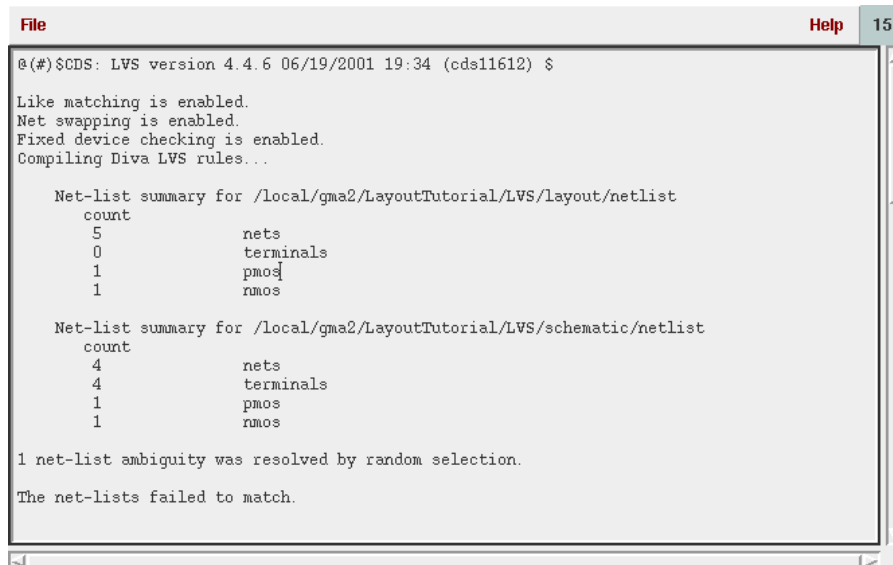


LVS stands for Layout Versus Schematic; it verifies that your layout and schematic are topologically equivalent. Before Do LVS, you need to Create a schematic. We use default value of transistor.

Click Run in LVS Form



Click Output in above LVS form



Why netlists fail to match ? Lets add a rectangle metal in layout.

Close the extracted view, Close Output Form,

In LSW select metal1 layer, draw the rectangle on the top of PMOS

Save the Design Design--->Save

Do Extraction on the layout,

Open the extracted view do LVS, This time the Output Form reports the netlists were matched.(click Output in LVS Form)

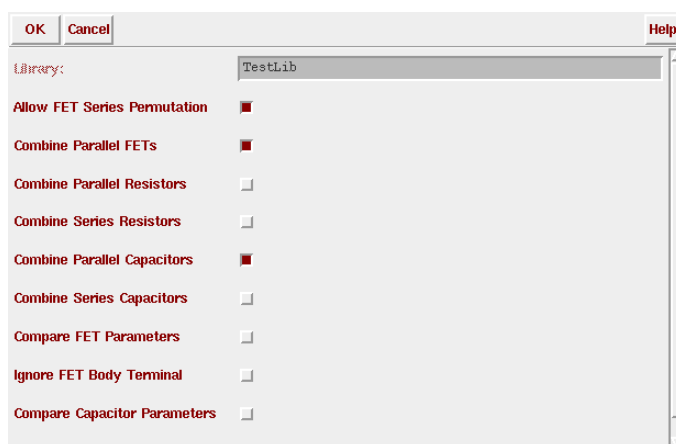
We still notice difference between the Pmos width of Schematic and Extracted View

Can we use LVS to report this ?

In the Extracted View Window, select NCSU--->Modify LVS Rules

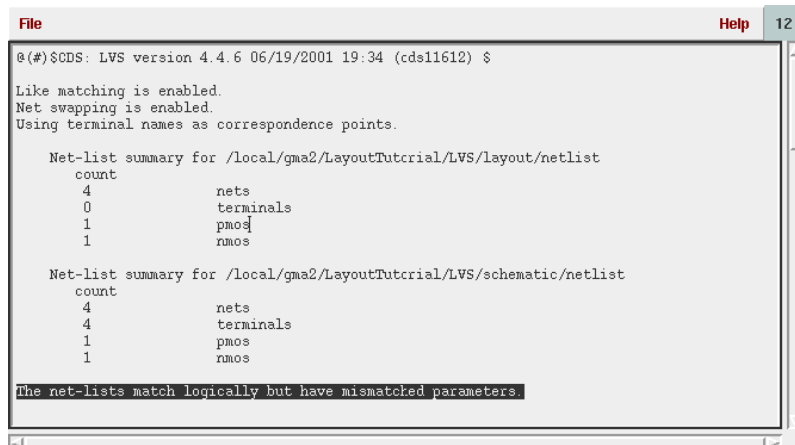
Select Compare FET Parameters, as the name implies LVS will compare PMOS parameter

Press OK



After do LVS, click in LVS Form.

LVS reports Mismatch Parameter.



```
File Help 12
@(#)SCDS: LVS version 4.4.6 06/19/2001 19:34 (cds11612) $
Like matching is enabled.
Net swapping is enabled.
Using terminal names as correspondence points.

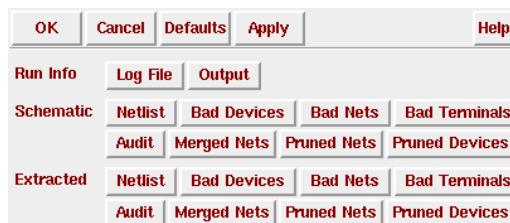
Net-list summary for /local/gma2/LayoutTutorial/LVS/layout/netlist
count
4      nets
0      terminals
1      pmos
1      rmos

Net-list summary for /local/gma2/LayoutTutorial/LVS/schematic/netlist
count
4      nets
4      terminals
1      pmos
1      rmos

The net-lists match logically but have mismatched parameters.
```

Do you have any ideas about terminal mismatch?

Next we'll display other information, click Info button in LVS Form



OK	Cancel	Defaults	Apply	Help
Run Info				
Log File		Output		
Schematic	Netlist	Bad Devices	Bad Nets	Bad Terminals
	Audit	Merged Nets	Pruned Nets	Pruned Devices
Extracted	Netlist	Bad Devices	Bad Nets	Bad Terminals
	Audit	Merged Nets	Pruned Nets	Pruned Devices

In the Info Form, click netlist button, you can get the netlist of schematic and layout

Step 4: Hspice Netlist Extraction with Cadence

This tutorial explains how to extract a HSPICE netlist from your cellview from either the layout view.

- From Virtuoso (the layout view):
 - a) Get the extracted view of the layout:
 1. Select **Verify -> Extract**.
 2. Click OK.
 - b) Start Analog Environment:
Select **Tools -> Analog Environment** from the extracted window.
 - c) Go to **Netlist Extraction Procedure** below.

Netlist Extraction Procedure

From Analog Environment: (Note: You must have the extracted view open in the background)

- Select **Setup->Simulator/Directory/Host**.
 - a) Select the **hspiceS** simulator.
 - b) Set your project directory. This will be where all your files will be created.
- Select **Setup->Design**
 - a) Library name: (your library name)
 - b) Cell Name: (your cell name)
 - c) View Name: Extracted OR Schematic, depending on whether you started with layout or schematics
 - d) Open Mode: ReadClick OK to close the window.
- Select **Setup->Environment**.
 - a) Insert "**extracted**" before "**hspiceS**" on the **Switch View List** if you started with layout; if you started with schematic, insert "**schematic**" instead if not already there, but it should be already there.
 - b) Insert "**ivpcell**" before "**hspiceS**" on the **Stop View List**.
 - c) Select the **Include File Syntax** to be **hspice**.
- Create Netlist.
 - a) Select **Simulation -> Netlist -> Create Final**.
 - c) Save the file displayed. **This is the HSPICE netlist of your design.**
- Exit Analog Environment: Select **Session -> Quit**.
- When asked if you wish to **save current state**, click No.

Extracted Hspice Netlist

```
* # FILE NAME:
/PROJ/HPVLSI/KKKIM/LAB/SIMULATION/INVERTER/HSPICES/EXTRACTED/
* NETLIST/INVERTER.C.RAW
* NETLIST OUTPUT FOR HSPICES.
* GENERATED ON MAY 23 05:06:53 2006
```

```
M0 OP IP Vdd! Vdd! D25P L=240E-9 W=480E-9 AD=288.000011209114E-15
+AS=316.799998777498E-15 PD=1.67999996847357E-6 PS=1.7999999499807E-6 M=1
M1 OP IP 0 0 D25N L=240E-9 W=480E-9 AD=288.000011209114E-15
+AS=316.799998777498E-15 PD=1.67999996847357E-6 PS=1.7999999499807E-6 M=1
```

```
.lib "/ECEnet/Apps1/CDS/NCSU1.4/local/models/hspice/nom/nom/d25P" PMOS
```

```
.lib "/ECEnet/Apps1/CDS/NCSU1.4/local/models/hspice/nom/nom/d25N" NMOS
```

```
* INCLUDE FILES
```

```
* END OF NETLIST
.TEMP 25.0000
.OP
.save
.OPTION INGOLD=2 ARTIST=2 PSF=2
+ PROBE=0
.END
```



Copy these lines and Paste
the lines in the frame files.

Download three frame files from
<http://www.ece.neu.edu/courses/eece4525/2016su/eece4525.htm>

And, solve the equations on page 2 in this guideline.