Washington University in St. Louis

McKelvey School of Engineering

Spring Semester 2023

CSE463M: Digital Integrated Circuit Design and Architecture

Homework #2

CSE 463M and CSE 563M students:

1) Solve problem 5.1 (inverter design with nMOS and resistor) from Chapter 5 from the textbook. Please show all your work and be neat and circle your answers. (10 points)

2) Solve problem 5.6 (inverter design with nMOS and pMOS) from Chapter 5 from the textbook. Please show all your work and be neat and circle your answers. (10 points)

3) Design suitable circuits in Cadence that will allow you to compute the voltage threshold (Vt), k' and substrate bias coefficient (γ) for an NMOS transistor. The transistor aspect ratios are W=9um and L=9um. Print the circuits that you design to extract these 3 parameters. Simulate the circuit and print the simulated results. Indicate on your plots how you computed the 3 parameters. (30 points)

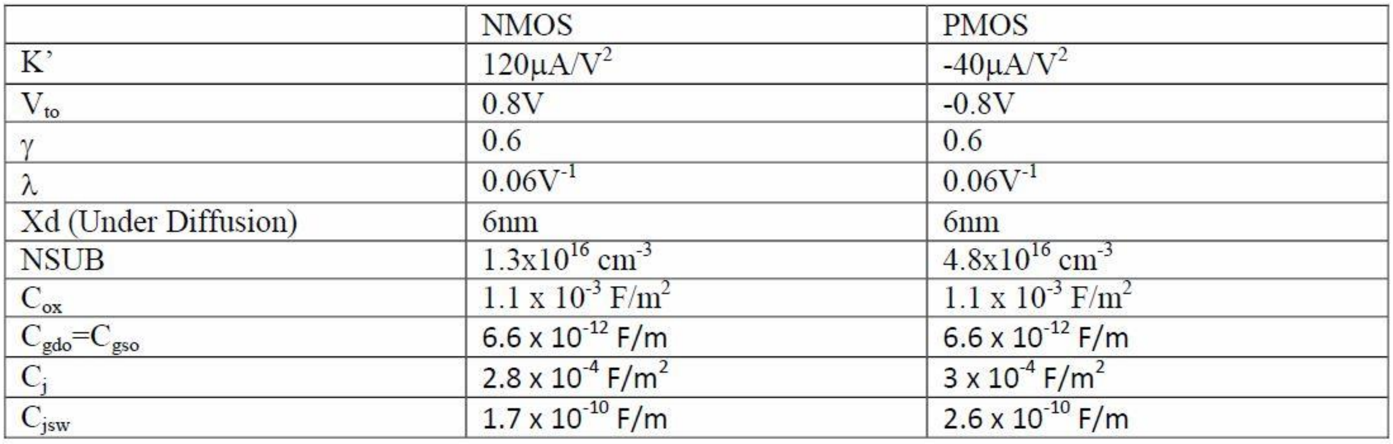
4) Repeat the same exercise for a PMOS device. Print all simulations and circuits that you use in order to determine voltage threshold (Vt), k' and substrate bias coefficient (γ) for the PMOS transistor. (30 points)

5) Inverter design problem (schematic only)
Draw a schematic of a CMOS inverter. The power supply of the circuit is 5V, i.e. the source of the PMOS transistor is connected to 5V. Choose minimum size for both length and width of PMOS and NMOS transistors. Save the circuit and print it out. Perform a DC analysis on the circuit by sweeping the input to the inverter from 0V to 5V. Plot the output of the inverter. Label the threshold of the inverter on the plot. (20 points)

6) Inverter design problem (including layout) (50 points)
Follow the wiki tutorial in order to solve the problems bellow and get familiar with Cadence:

https://eda.engineering.wustl.edu/wiki/index.php/Cadence

Here is transistor data for AMI Semiconductor 0.5 micron process:



	NMOS	PMOS
K'	$120\mu A/V^2$	$-40\mu A/V^2$
V_{to}	0.8V	-0.8V
γ	0.6	0.6
λ	$0.06V^{-1}$	$0.06V^{-1}$
Xd (Under Diffusion)	6nm	6nm
NSUB	$1.3x10^{16}$ cm^{-3}	$4.8x10^{16}$ cm^{-3}
C_{ox}	1.1×10^{-3} F/m^2	1.1×10^{-3} F/m^2
C_{gdo}=C_{gso}	6.6×10^{-12} F/m	6.6×10^{-12} F/m
C_j	2.8×10^{-4} F/m^2	3×10^{-4} F/m^2
C_{jsw}	1.7×10^{-10} F/m	2.6×10^{-10} F/m

Problem set:
6.1) Design a CMOS inverter that will operate from a 5V power supply (i.e. Vdd=5V) and it has a threshold (Vth) of 3V. Show all hand calculations that allowed you to design this circuit. Use the data in the table above for your calculations.

6.2) Design the CMOS inverter from problem 6.1) in Cadence. Print the schematic of the inverter. Simulate the static behavior of the inverter. Make sure that your SPICE simulation indicates that the threshold of the inverter is 3V. If this is not the case, readjust the aspect ratios of your NMOS and/or PMOS in order to have the desired threshold of 3V. Print your SPICE simulation and indicate your Vth on the plot.

6.3) Using SPICE and the calculator within the SPICE simulator to compute VIL and VIH. Plot the results and indicate your results on the plot.

6.4) Draw the layout of the inverter in Cadence. Print the layout of the device.

6.5) Run a Design Rule Check (DRC), extract the layout and LVS the design. Cut and Paste the messages generated in the CIW (the main Cadence window) when the DRC is executed. Print the Layout vs Schematic (LVS) result – make sure it has successfully passed LVS.

CSE 563M students only:

7) Solve problem 5.4 from the textbook. (10 points)

8) Solve problem 5.7 from the textbook. (10 points)

1) - 5.1

(a)

$V_{DD} = 1.1\,V$

$V_{T0} = 0.52\,V$

$V_{OL} = 0.05\,V$

$R = 2\,k\Omega$

$\mu_n C_{ox} = k'_n = 216\,\mu A/V^2$

$$\frac{V_{DD} - V_{OL}}{R_L} = \frac{k'_n}{2}(\frac{W}{L})[2(V_{DD} - V_{T0})V_{OL} - V_{OL}{}^2]$$

$$\frac{1.1 - 0.05}{2000} = \frac{216 \cdot 10^{-6}}{2}(\frac{W}{L})[2(1.1 - 0.52)0.05 - (0.05)^2]$$

$$\boxed{\frac{W}{L} = 87.58759}$$

(b)

$$k_n = k'_n \frac{W}{L}$$

$$V_{IL} = V_{T0} + \frac{1}{k_n R_L} = V_{T0} + \frac{1}{k'_n \frac{W}{L} R_L}$$

$$\boxed{V_{IL} = 0.52 + \frac{1}{216 \cdot 10^{-6} \cdot 87.58759 \cdot 2000} = 0.546\,V}$$

$$V_{IH} = V_{T0} + \sqrt{\frac{8}{3} \cdot \frac{V_{DD}}{k_n R_L}} - \frac{1}{k_n R_L}$$

$$V_{IH} = 0.52 + \sqrt{\frac{8}{3} \cdot \frac{1.1}{216 \cdot 10^{-6} \cdot 87.58759 \cdot 2000}} - \frac{1}{216 \cdot 10^{-6} \cdot 87.58759 \cdot 2000}$$

$$\boxed{V_{IH} = 0.772\,V}$$

(c)

$$\boxed{NM_L = V_{IL} - V_{OL} = 0.546 - 0.05 = 0.496\,V}$$

$$\boxed{NM_H = V_{OH} - V_{IH} = 1.1 - 0.772 = 0.328\,V}$$

2) - 5.6

+ We know the following:

nMOS: $V_{T0,n} = 0.48V, \mu_n C_{ox} = 102\ \mu A/V^2, (\frac{W}{L})_n = 10$

$pMOS: V_{T0,p} = -0.46V, \mu_p C_{ox} = 51.6\ \mu A/V^2, (\frac{W}{L})_p = 19$

$V_{DD} = V_{OH} = 1.2\ V$

$k_n = k'_n(\frac{W}{L})_n = \mu_n C_{ox}(\frac{W}{L})_n = 102 \cdot 10 = 1020\ \mu A/V^2$

$k_p = k'_p(\frac{W}{L})_p = \mu_p C_{ox}(\frac{W}{L})_p = 51.6 \cdot 19 = 980.4\ \mu A/V^2$

$$k_R = \frac{k_n}{k_p} = \frac{1020}{980.4} = 1.04$$

+ To get V_{IL}:

$$V_{IL} = \frac{2V_{out} + V_{T0,p} - V_{DD} + k_R V_{T0,n}}{1 + k_R}$$

$$V_{IL} = \frac{2V_{out} - 0.46 - 1.2 + 1.04 \cdot 0.48}{1 + 1.04} = 0.98V_{out} - 0.569$$

+ Substitute above equation to KCL equation:

$$\frac{k_n}{2}(V_{in} - V_{T0,n})^2 = \frac{k_p}{2}[2(V_{DD} - V_{in} - |V_{T0,p}|)(V_{DD} - V_{out}) - (V_{DD} - V_{out})^2]$$

+ Since, $V_{in} = V_{IL}$

$$\frac{1020 \cdot 10^{-6}}{2}(0.98V_{out} - 0.569 - 0.48)^2$$

$$= \frac{980.4 \cdot 10^{-6}}{2}[2(1.2 - (0.98V_{out} - 0.569) - 0.46)(1.2 - V_{out}) - (1.2 - V_{out})^2]$$

+ The expression yields a second order polynomial as follows:

$$0.000019212{V_{out}}^2 + 0.000211234V_{out} - 0.00027292 = 0$$

$V_{out} = -12.1628\ or\ 1.16796$

$V_{out} = 1.16796(V_{out} > 0)$

$V_{IL} = 0.98V_{out} - 0.569 = 0.98 \cdot 1.16796 - 0.569 = 0.575\ V$

+ To get V_{IH} in terms of the output voltage:

$$V_{IH} = \frac{V_{DD} + V_{T0,p} + k_R\left(2V_{out} + V_{T0,n}\right)}{1 + k_R}$$

$$= \frac{1.2 - 0.46 + 1.04(2V_{out} + 0.48)}{1 + 1.04} = 1.0196V_{out} + 0.6074$$

+ Substitute above equation to KCL equation:

$$\frac{k_n}{2}\left[2\left(V_{in} - V_{T0,n}\right)V_{out} - {V_{out}}^2\right] = \frac{k_p}{2}\left(V_{DD} - V_{in} - \left|V_{T0,p}\right|\right)^2$$

+ Since, $V_{in} = V_{IH}$

$$\frac{1020 \cdot 10^{-6}}{2}\left[2(1.0196V_{out} + 0.6074 - 0.48)V_{out} - {V_{out}}^2\right]$$

$$= \frac{980.4 \cdot 10^{-6}}{2}(1.2 - 1.0196V_{out} - 0.6074 - 0.46)^2$$

$$0.0000203878{V_{out}}^2 + 0.000262497V_{out} - 8.61907 \cdot 10^{-6} = 0$$

$$V_{out} = -12.9079 \; or \; 0.0327516$$

$$V_{out} = 0.0327516(V_{out} > 0)$$

$$V_{IH} = 1.0196V_{out} + 0.6074 = 1.0196 \cdot 0.0327516 + 0.6074 = 0.64\,V$$

+ Finally, we can find the noise margin:

$$\boxed{NM_L = V_{IL} - V_{OL} = 0.575 - 0 = 0.575\,V}$$

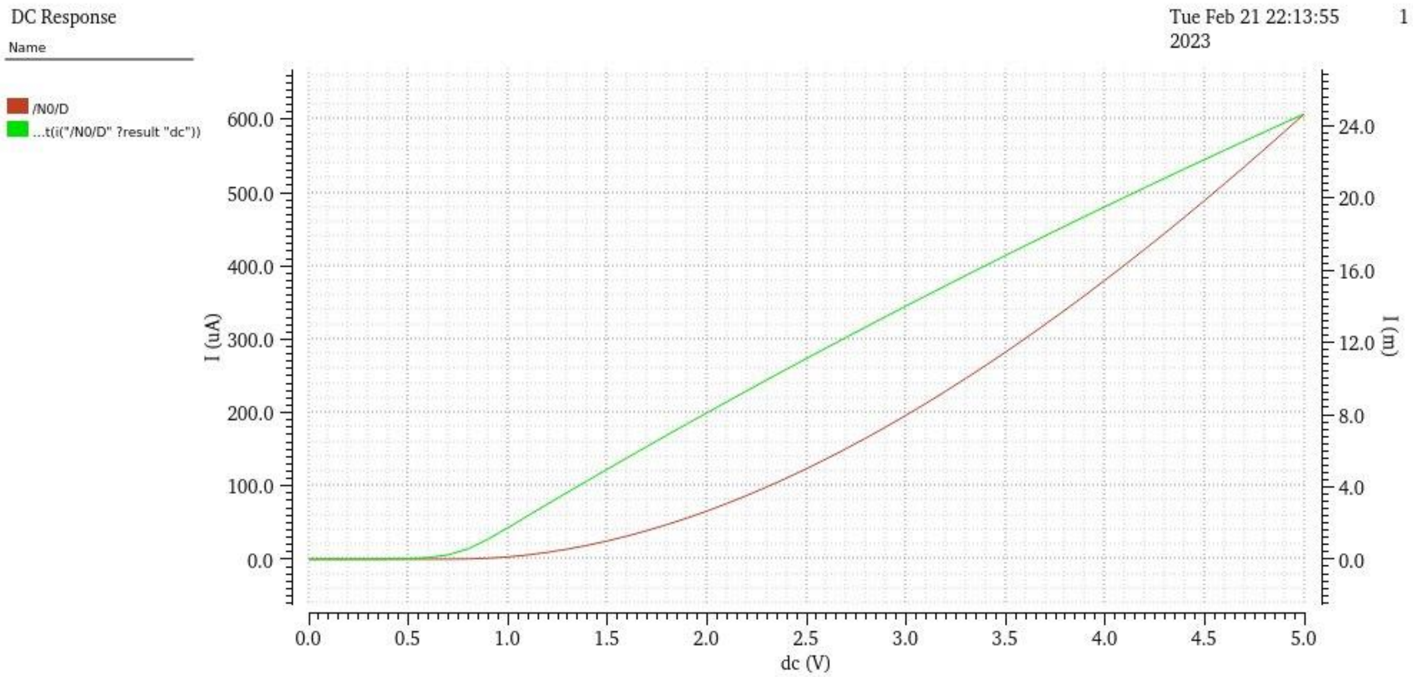
$$\boxed{NM_H = V_{OH} - V_{IH} = 1.2 - 0.64 = 0.58\,V}$$

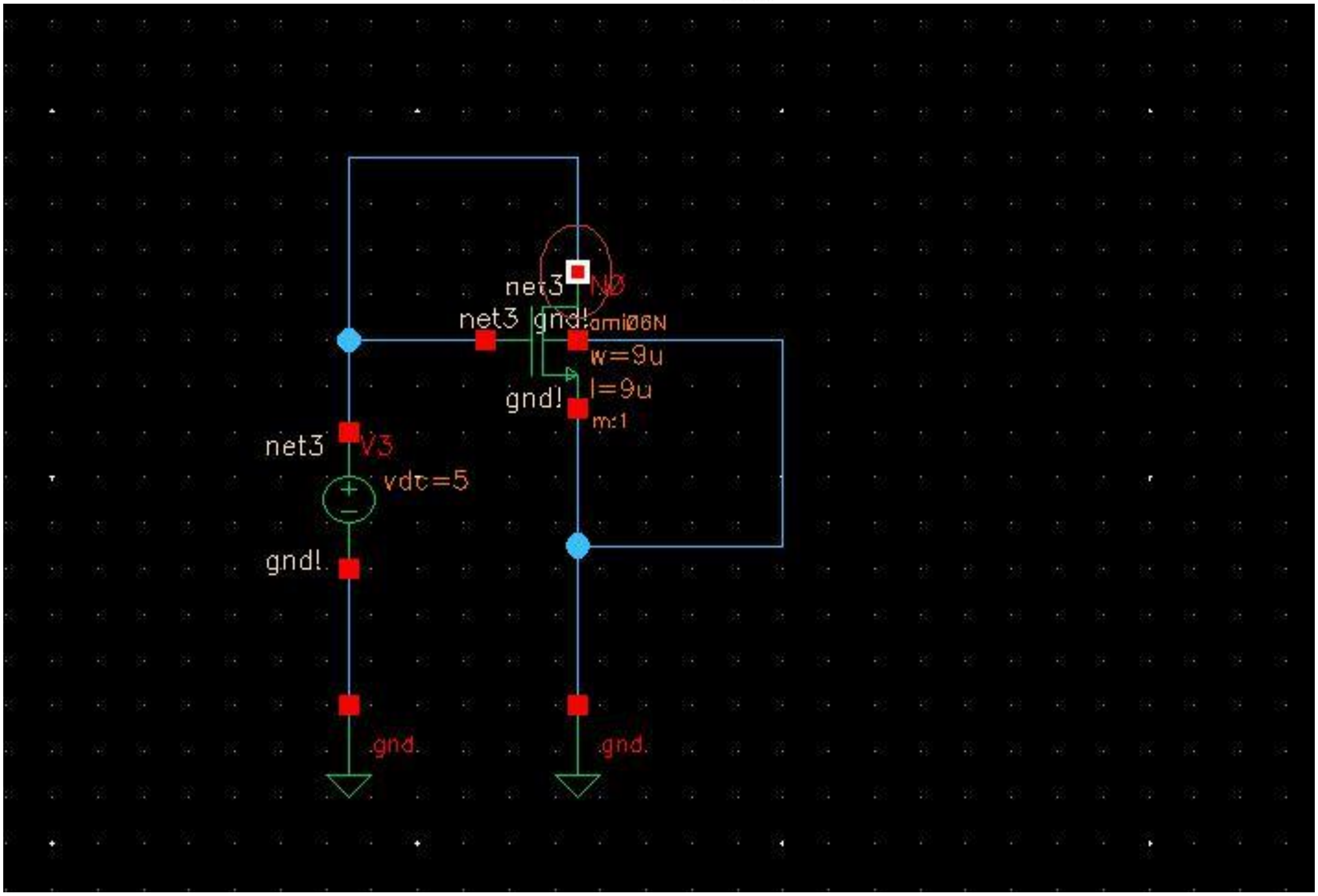
+ For V_{th}:

$$V_{th} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}(V_{DD} - |V_{T0,p}|)}{1 + \sqrt{\frac{1}{k_R}}}$$

$$\boxed{V_{th} = \frac{0.48 + \sqrt{\frac{1}{1.04}}(1.2 - 0.46)}{1 + \sqrt{\frac{1}{1.04}}} = 0.60871V}$$

3)

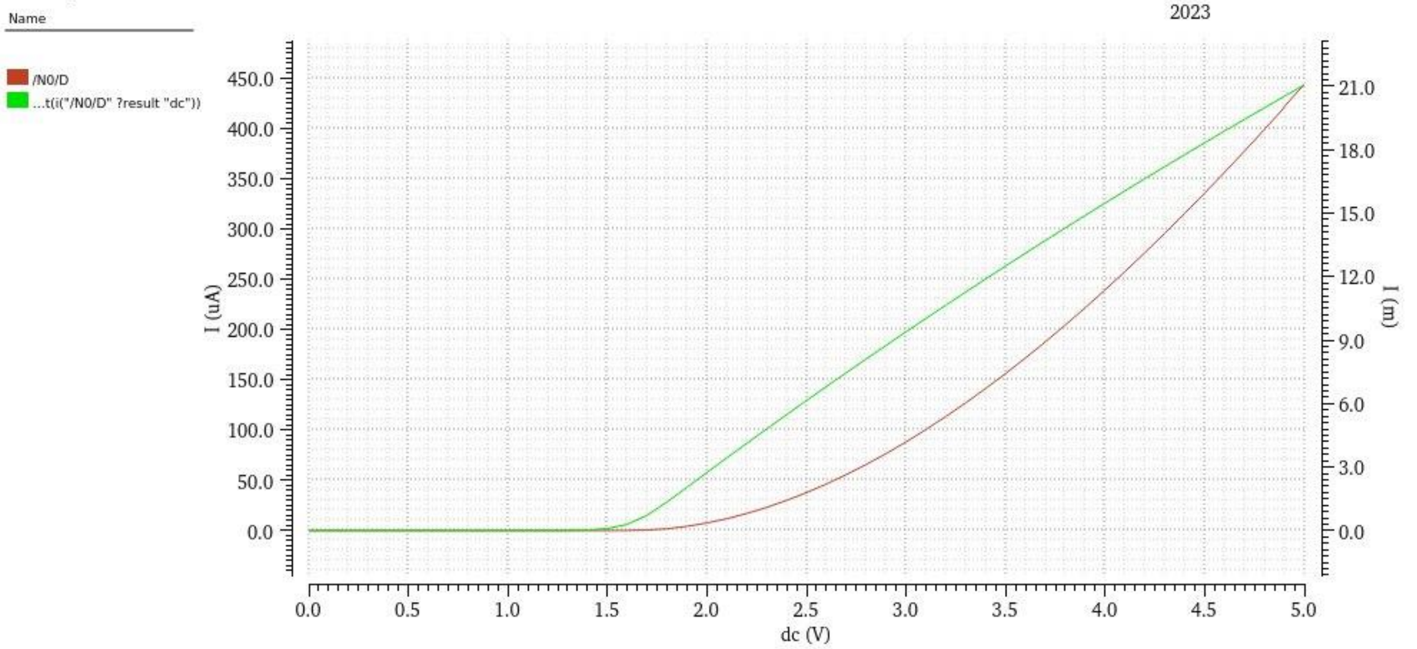

DC Response
Name
🟥 /N0/D
🟩 ...t(i("/N0/D" ?result "dc"))
Tue Feb 21 22:13:55
2023
1
I (uA)
600.0
500.0
400.0
300.0
200.0
100.0
0.0
I (m)
24.0
20.0
16.0
12.0
8.0
4.0
0.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
dc (V)

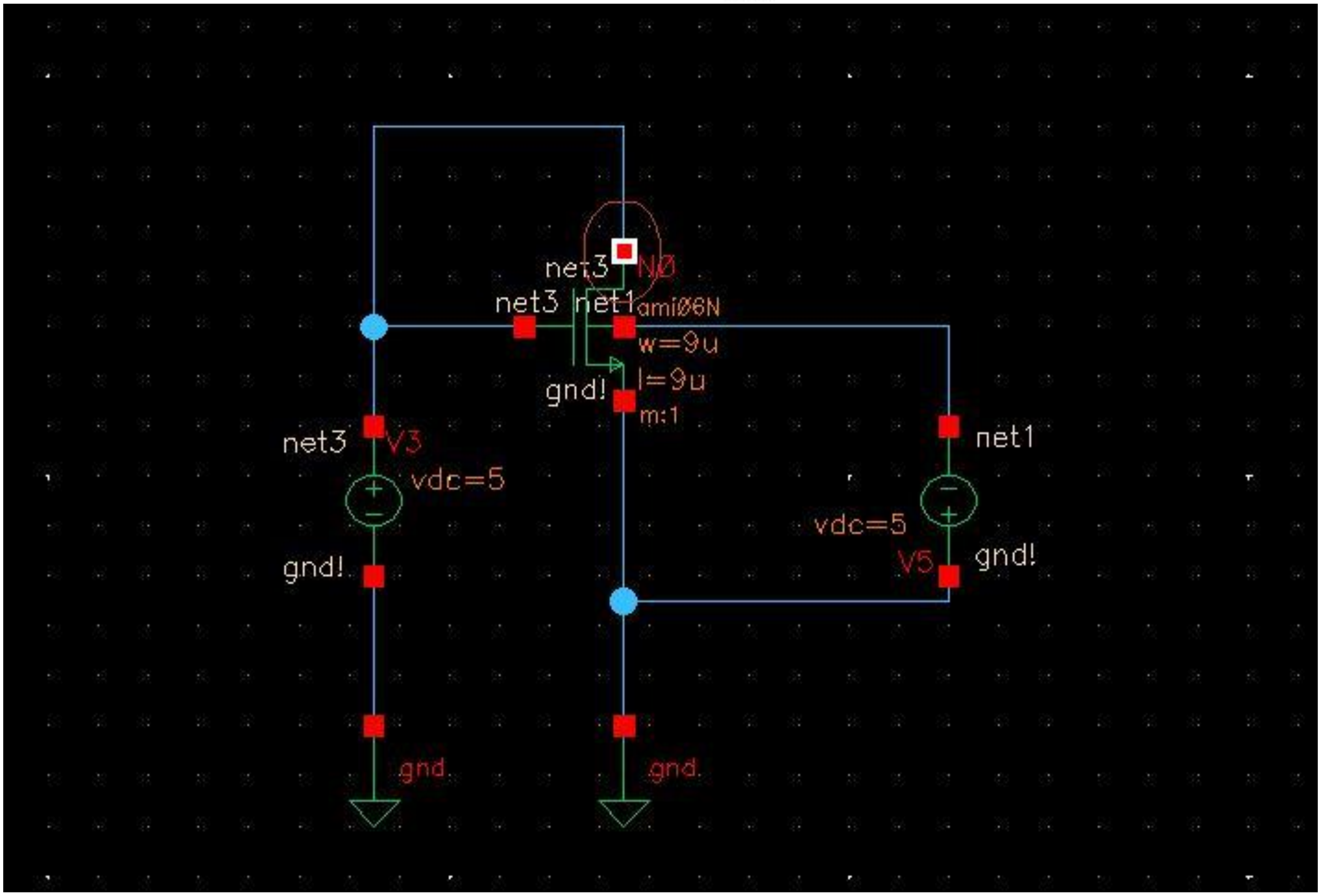

net3
N0
net3
gnd!
ami06N
w=9u
gnd!
l=9u
m:1
net3
V3
vdc=5
gnd!
gnd
gnd

DC Response

Tue Feb 21 22:19:55 2023

1





+ for V_{T0}:
We can get Vt0 from the graph above.

$\boxed{V_{T0} = 0.6\ V}$

+ for k_n:

$$\sqrt{\frac{k_n}{2}} = \frac{\sqrt{I_{D1}} - \sqrt{I_{D2}}}{V_{GS1} - V_{GS2}}$$

We can pick 2 points of green line from above graph.
(2.0V, 200 uA)
(4.0V, 480 uA)

$$\sqrt{\frac{k_n}{2}} = \frac{\sqrt{480} - \sqrt{200}}{4 - 2}$$

$$\sqrt{\frac{k_n}{2}} = 0.00388338$$

$\boxed{k_n = 0.0000301613\ AV^2}$

+ for γ:

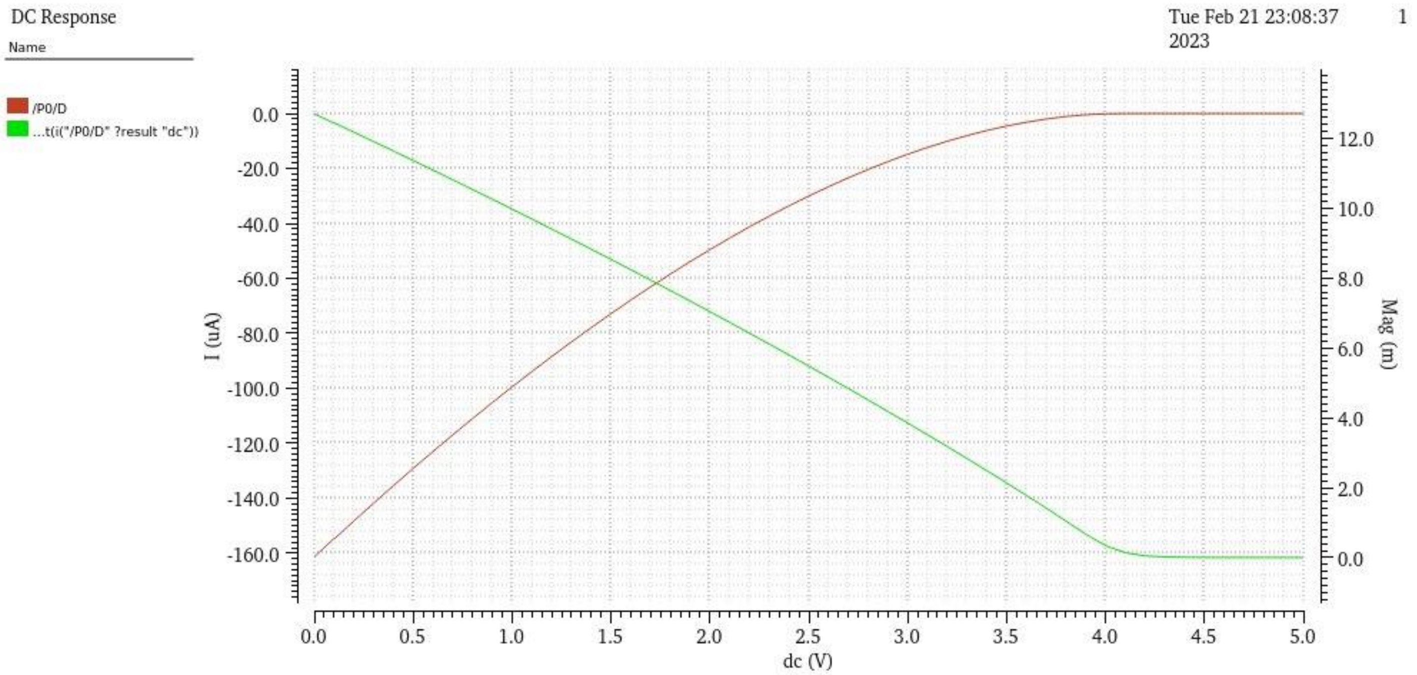
$$\gamma = \frac{V_T - V_{T0}}{\sqrt{|-2\Phi_F + V_{SB}|} - \sqrt{|-2\Phi_F|}}$$

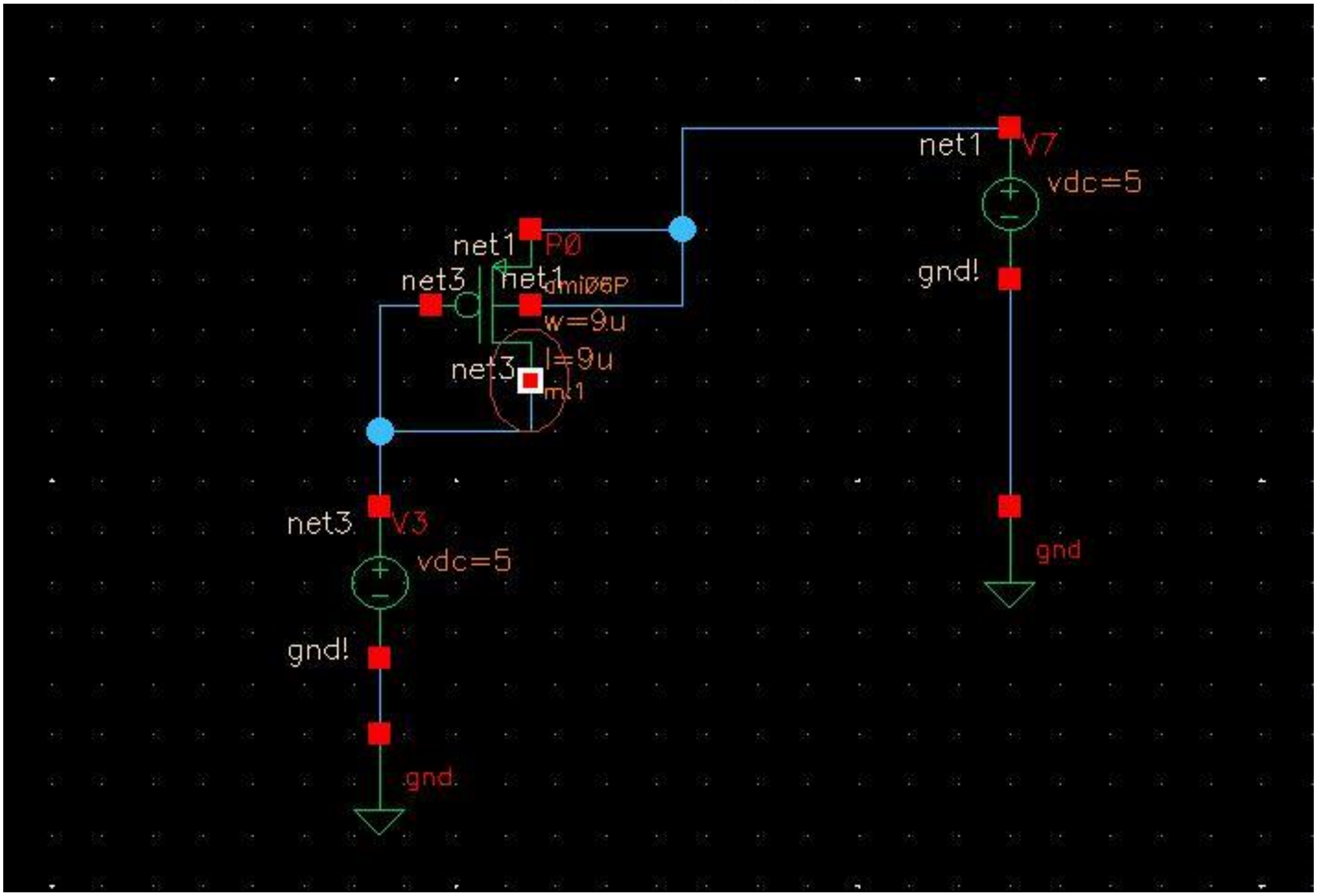
$V_T = 1.5\ V$
$V_{T0} = 0.6\ V$
$\Phi_F = -0.375$
$V_{SB} = 5\ V$

$$\gamma = \frac{1.5 - 0.6}{\sqrt{|-2 \cdot -0.375 + 5|} - \sqrt{|-2 \cdot -0.375|}}$$

$\boxed{\gamma = 0.587509}$

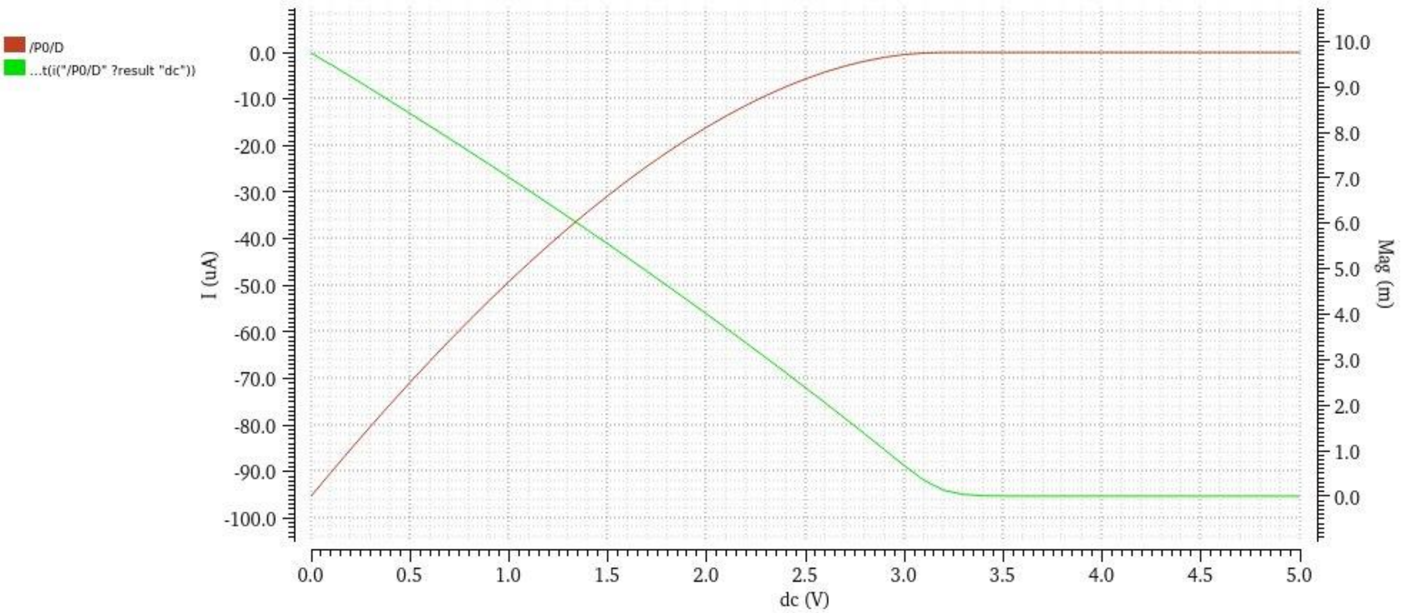
4)

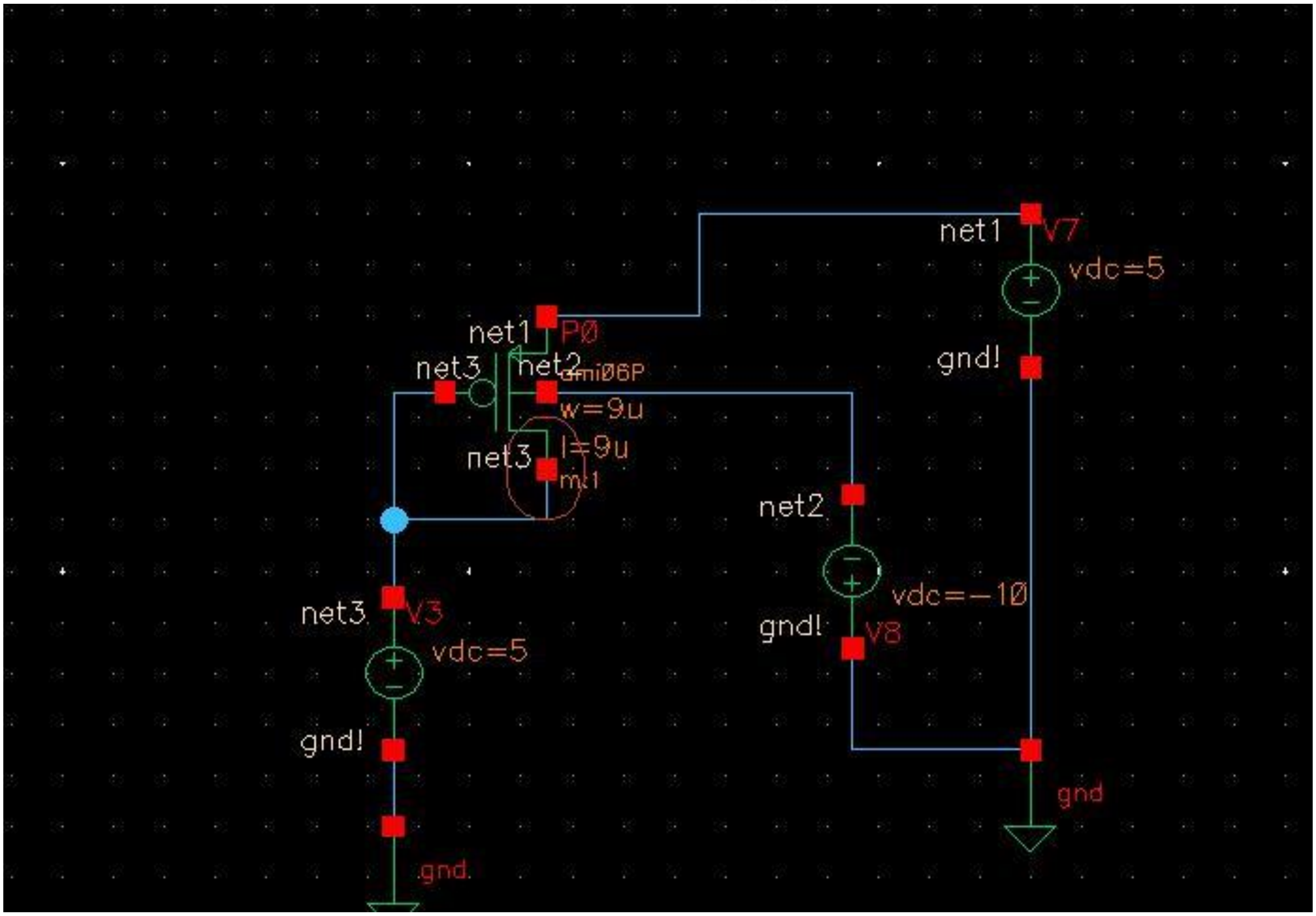

DC Response
Tue Feb 21 23:08:37 2023
1
Name
🟥 /P0/D
🟩 ...t(i("/P0/D" ?result "dc"))
I (uA)
0.0
-20.0
-40.0
-60.0
-80.0
-100.0
-120.0
-140.0
-160.0
Mag (m)
12.0
10.0
8.0
6.0
4.0
2.0
0.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
dc (V)


net1
V7
vdc=5
gnd!
gnd
net1
P0
net3
net1
ami06P
w=9u
net3
l=9u
m:1
net3
V3
vdc=5
gnd!
gnd

DC Response

Tue Feb 21 23:14:31 2023





+ for V_{T0}:

We can get Vt0 from the graph above.

$$\boxed{V_{T0} = -0.8\, V, (becasue\ we\ interpret\ above\ graph\ by\ minus\ 5V, 4.2 - 5)}$$

+ for k_p:

$$\sqrt{\frac{k_p}{2}} = \frac{\sqrt{I_{D1}} - \sqrt{I_{D2}}}{V_{GS1} - V_{GS2}}$$

We can pick 2 points from above graph.

(-3V, -72 uA)

(-2.5V, -92 uA)

$$\sqrt{\frac{k_p}{2}} = \frac{\sqrt{72} - \sqrt{92}}{(-3) - (2.5)}$$

$$\sqrt{\frac{k_p}{2}} = 0.201160$$

$$\boxed{k_p = 0.0809307\ AV^2}$$

+ for γ:

$$\gamma = \frac{V_T - V_{T0}}{\sqrt{\left|-2\Phi_F + V_{SB}\right|} - \sqrt{|-2\Phi_F|}}$$

$V_T = -1.8\, V$

$V_{T0} = -0.8\, V$

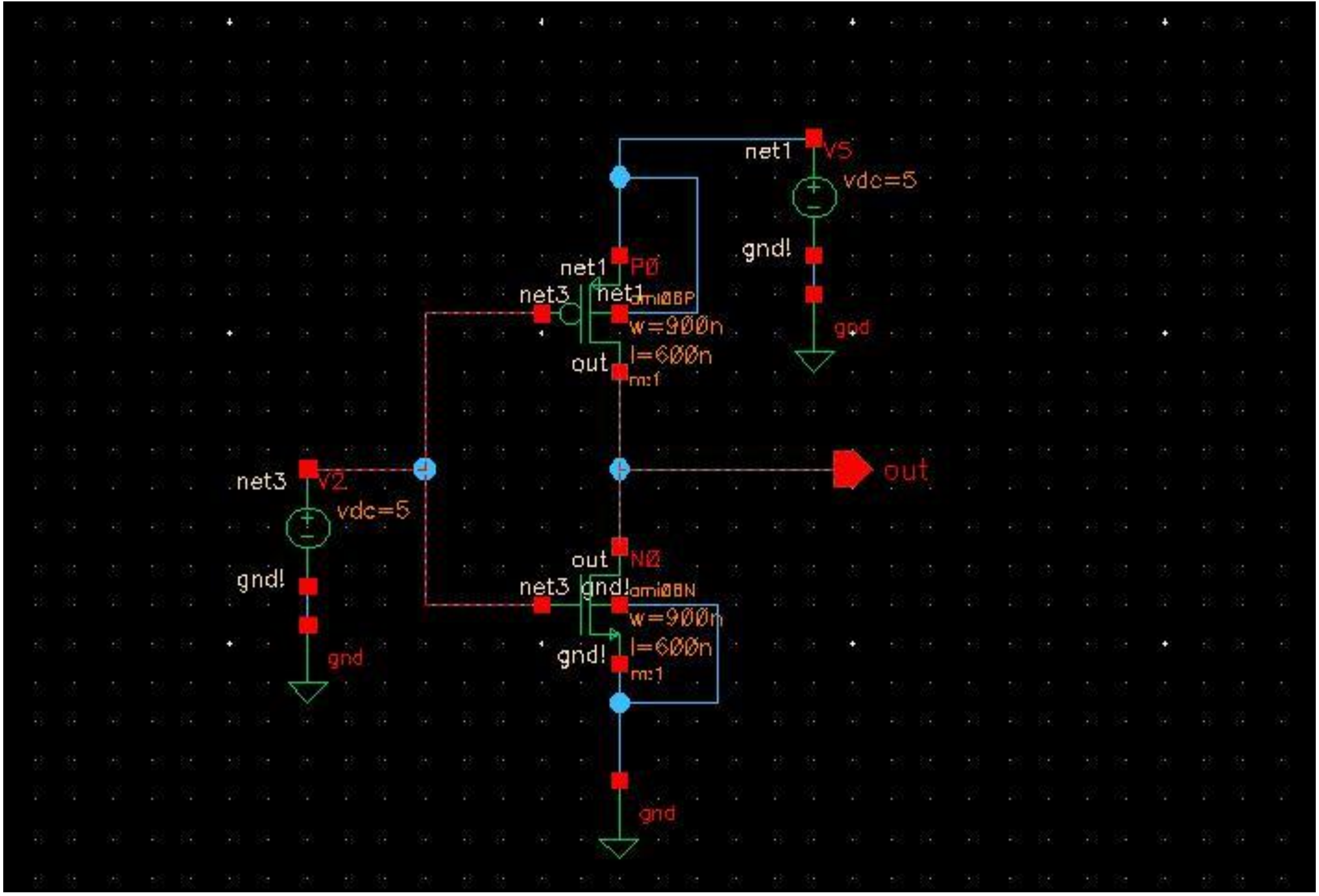
$\Phi_F = 0.375$

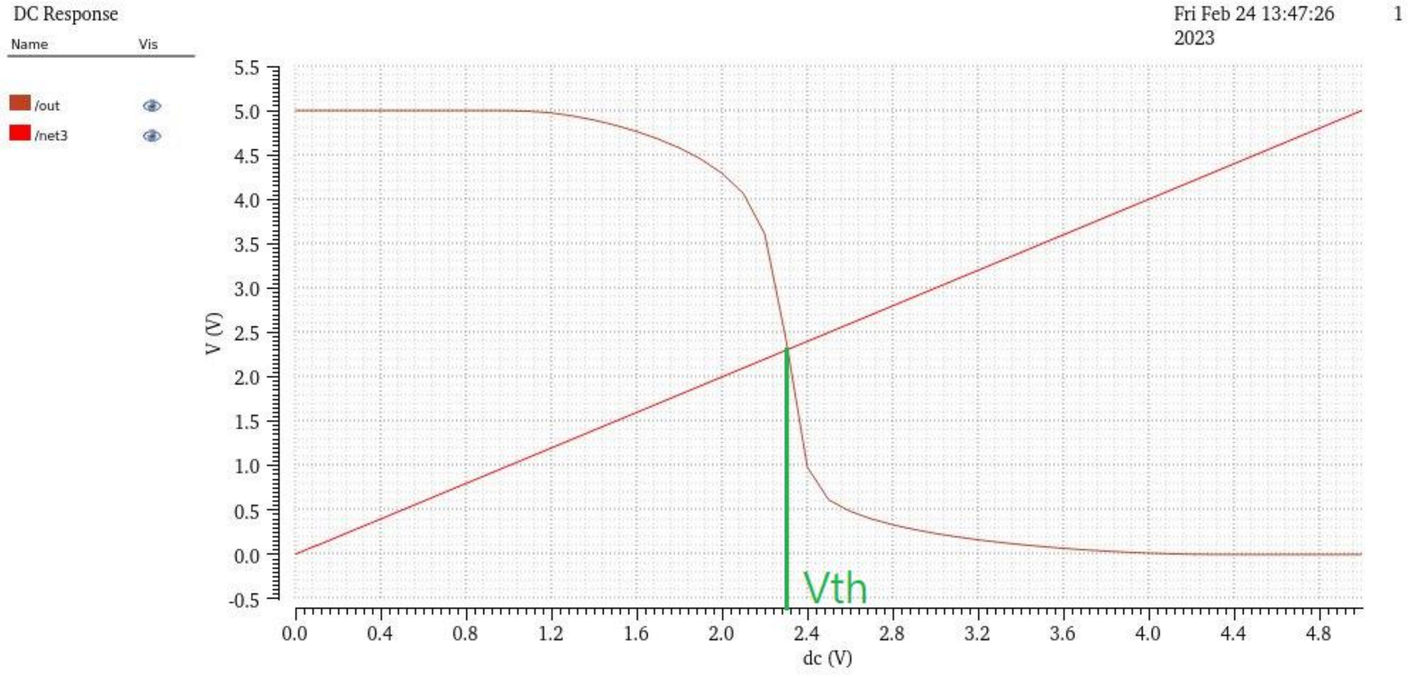
$V_{SB} = -5\, V$

$$\gamma = \frac{-1.8 - 0.8}{\sqrt{\left|-2 \cdot 0.375 - 5\right|} - \sqrt{|-2 \cdot 0.375|}}$$

$$\boxed{\gamma = -1.69725}$$

5)

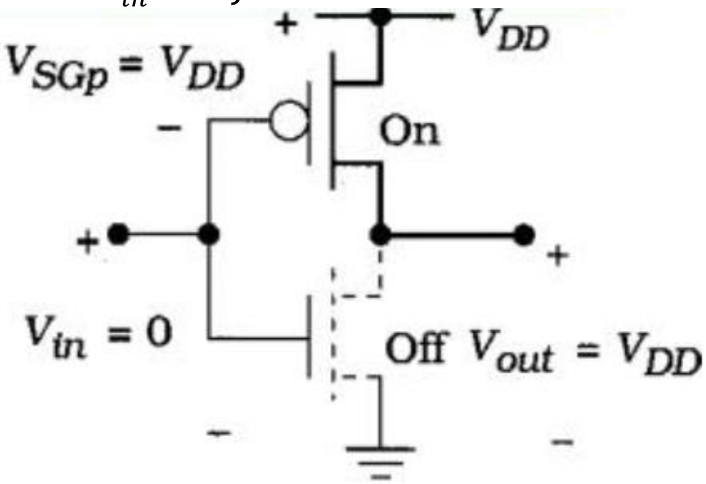

net1
V5
vdc=5
gnd!
gnd
net1
P0
net3
net1
ami06P
w=900n
l=600n
out
m:1
net3
V2
vdc=5
gnd!
gnd
out
out
N0
net3
gnd!
ami06N
w=900n
gnd!
l=600n
m:1
gnd


DC Response
Fri Feb 24 13:47:26 2023
1
Name
Vis
/out
/net3
V (V)
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
-0.5
0.0
0.4
0.8
1.2
1.6
2.0
2.4
2.8
3.2
3.6
4.0
4.4
4.8
dc (V)
Vth

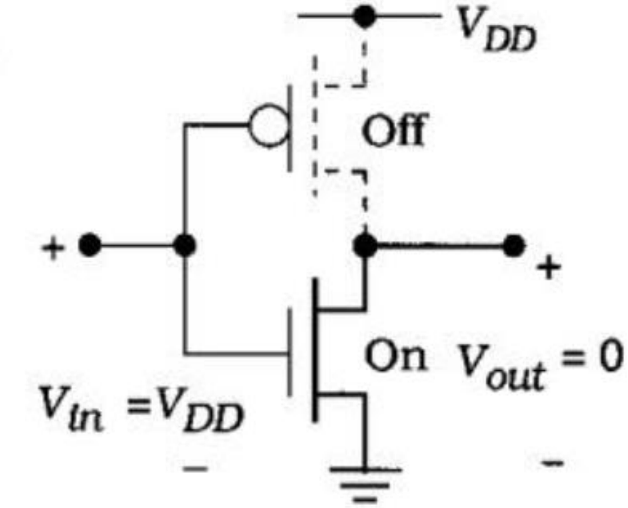
6-1)

Using PMOS and NMOS to implement CMOS inverter.
Upper part is PMOS and lower part is NMOS.
Gate is connected to Vin and Vout is connected to GND.

When $V_{in} = 0$, PMOS is ON and NMOS if OFF like below.



When $V_{in} = VDD$, PMOS is OFF and NMOS if ON like below.



+ For V_{OH}:
$V_{OH} = VDD$

+ For V_{OL}:
$V_{OL} = 0$

+ Using $V_{th} = 3V$, we can calculate k_R

$$V_{th} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}(V_{DD} - |V_{T0,p}|)}{1 + \sqrt{\frac{1}{k_R}}}$$

$$3 = \frac{0.8 + \sqrt{\frac{1}{k_R}}(5 - 0.8)}{1 + \sqrt{\frac{1}{k_R}}}$$

$k_R = 0.298$

+ For low voltage, V_{IL}:

$$V_{IL} = \frac{2V_{out} + V_{T0,p} - V_{DD} + k_R V_{T0,n}}{1 + k_R}$$

$$V_{IL} = \frac{2V_{out} - 0.8 - 5 + 0.298 \cdot 0.8}{1 + 0.298} = \frac{2V_{out} - 5.5616}{1.298} = 1.54V_{out} - 4.28$$

+ Substitute above equation to KCL equation:

$$\frac{k_n}{2}(V_{in} - V_{T0,n})^2 = \frac{k_p}{2}[2(V_{DD} - V_{in} - |V_{T0,p}|)(V_{DD} - V_{out}) - (V_{DD} - V_{out})^2]$$

$$k_R(V_{in} - V_{T0,n})^2 = [2(V_{DD} - V_{in} - |V_{T0,p}|)(V_{DD} - V_{out}) - (V_{DD} - V_{out})^2]$$

+ Since, $V_{in} = V_{IL}$

$$0.298(1.54V_{out} - 4.28 - 0.8)^2 = [2(5 - (1.54V_{out} - 4.28) - 0.8)(5 - V_{out}) - (5 - V_{out})^2]$$

+ The expression yields a second order polynomial as follows:

$-1.37326{V_{out}}^2 + 17.6974V_{out} - 52.1097 = 0$

$V_{out} = 4.5532 \; or \; 8.33389$

$V_{out} = 4.5532$

$$V_{IL} = 1.54V_{out} - 4.28 = 1.54 \cdot 4.5532 - 4.28 = 2.73\,V$$

+ For high voltage, V_{IH}:

$$V_{IH} = \frac{V_{DD} + V_{T0,p} + k_R(2V_{out} + V_{T0,n})}{1 + k_R}$$

$$V_{IH} = \frac{5 - 0.8 + 0.298(2V_{out} + 0.8)}{1 + 0.298} = \frac{0.596V_{out} + 4.4384}{1.298} = 0.46V_{out} + 3.41$$

+ Substitute above equation to KCL equation:

$$\frac{k_n}{2}\left[2\left(V_{in} - V_{T0,n}\right)V_{out} - {V_{out}}^2\right] = \frac{k_p}{2}\left(V_{DD} - V_{in} - \left|V_{T0,p}\right|\right)^2$$

$$k_R\left[2\left(V_{in} - V_{T0,n}\right)V_{out} - {V_{out}}^2\right] = \left(V_{DD} - V_{in} - \left|V_{T0,p}\right|\right)^2$$

+ Since, $V_{in} = V_{IH}$

$$0.298\left[2(0.46V_{out} + 3.41 - 0.8)V_{out} - {V_{out}}^2\right] = (5 - (0.46V_{out} + 3.41) - 0.8)^2$$

$$-0.23544{V_{out}}^2 + 2.28236V_{out} - 0.6241 = 0$$

$$V_{out} = 0.281627 \; or \; 9.41239$$

$$V_{out} = 0.281627$$

$$V_{IH} = 0.46V_{out} + 3.41 = 0.46 \cdot 0.281627 + 3.41 = 3.54\,V$$

+ Finally, we can find the noise margin:

$$\boxed{NM_L = V_{IL} - V_{OL} = 2.73 - 0 = 2.73\,V}$$

$$\boxed{NM_H = V_{OH} - V_{IH} = 5 - 3.54 = 1.46\,V}$$

6-2)

$$V_{th} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}(V_{DD} - |V_{T0,p}|)}{1 + \sqrt{\frac{1}{k_R}}}$$

$$3 = \frac{0.8 + \sqrt{\frac{1}{k_R}}(5 - 0.8)}{1 + \sqrt{\frac{1}{k_R}}}$$

$$k_R = 0.297521$$

+ I’m going to adjust NMOS and to make the calculation easy, so I set PMOS W/L = 1.

$$k_R = \frac{k_n}{k_p} = \frac{k'_n(\frac{W}{L})_n}{k'_p(\frac{W}{L})_p} = \frac{120(\frac{W}{L})_n}{40(\frac{W}{L})_p} = \frac{3(\frac{W}{L})_n}{(\frac{W}{L})_p}$$

$$0.297521 = 3(\frac{W}{L})_n$$

$$(\frac{W}{L})_n = 0.099$$

$$L_n = \frac{W_n}{0.099}$$

$$L_n = 9150$$

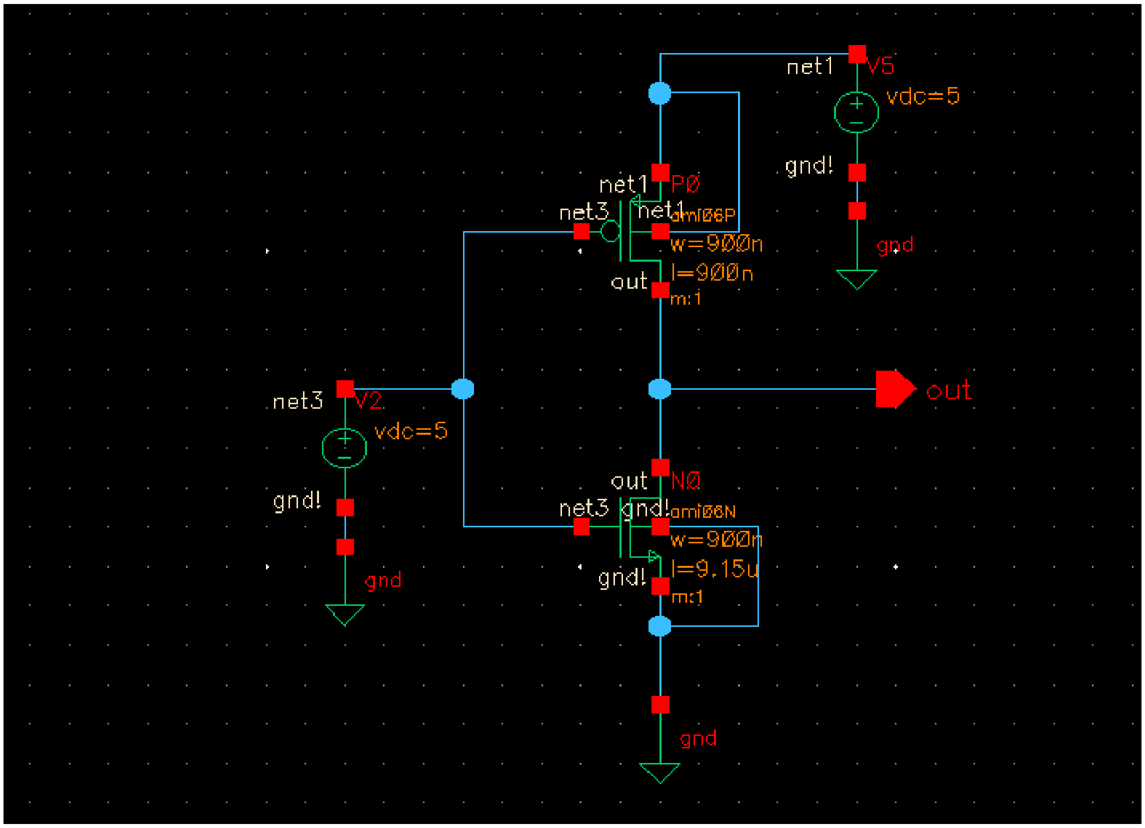
+ Now, I found out all the values.

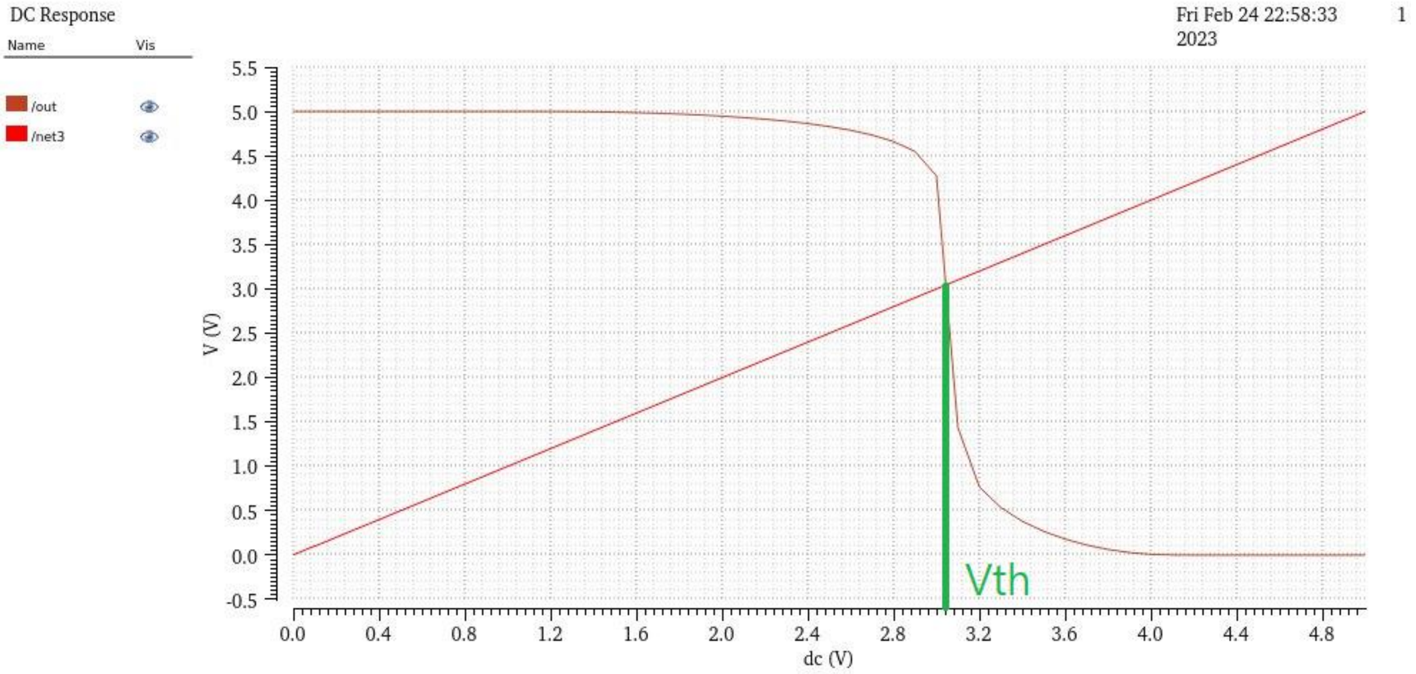
PMOS width = 900n

PMOS length= 900n

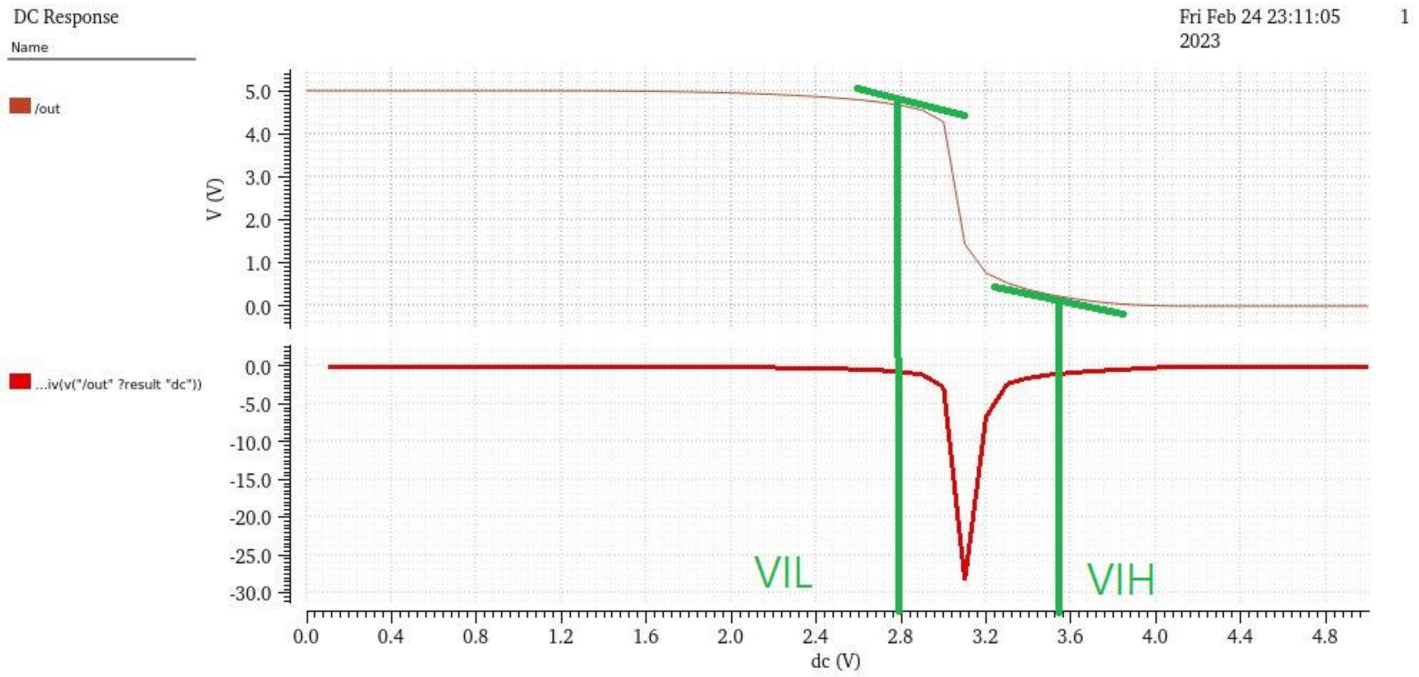
NMOS width = 900n

NMOS length= 9150n


net1
V5
vdc=5
gnd!
gnd
net1
PØ
net3
net1
ami06P
w=9ØØn
l=9ØØn
out
m:1
out
net3
V2
vdc=5
gnd!
gnd
out
NØ
net3
gnd!
ami06N
w=9ØØn
l=9.15u
gnd!
m:1
gnd


DC Response
Fri Feb 24 22:58:33 2023
1
Name
Vis
/out
/net3
V (V)
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
-0.5
0.0
0.4
0.8
1.2
1.6
2.0
2.4
2.8
3.2
3.6
4.0
4.4
4.8
dc (V)
Vth

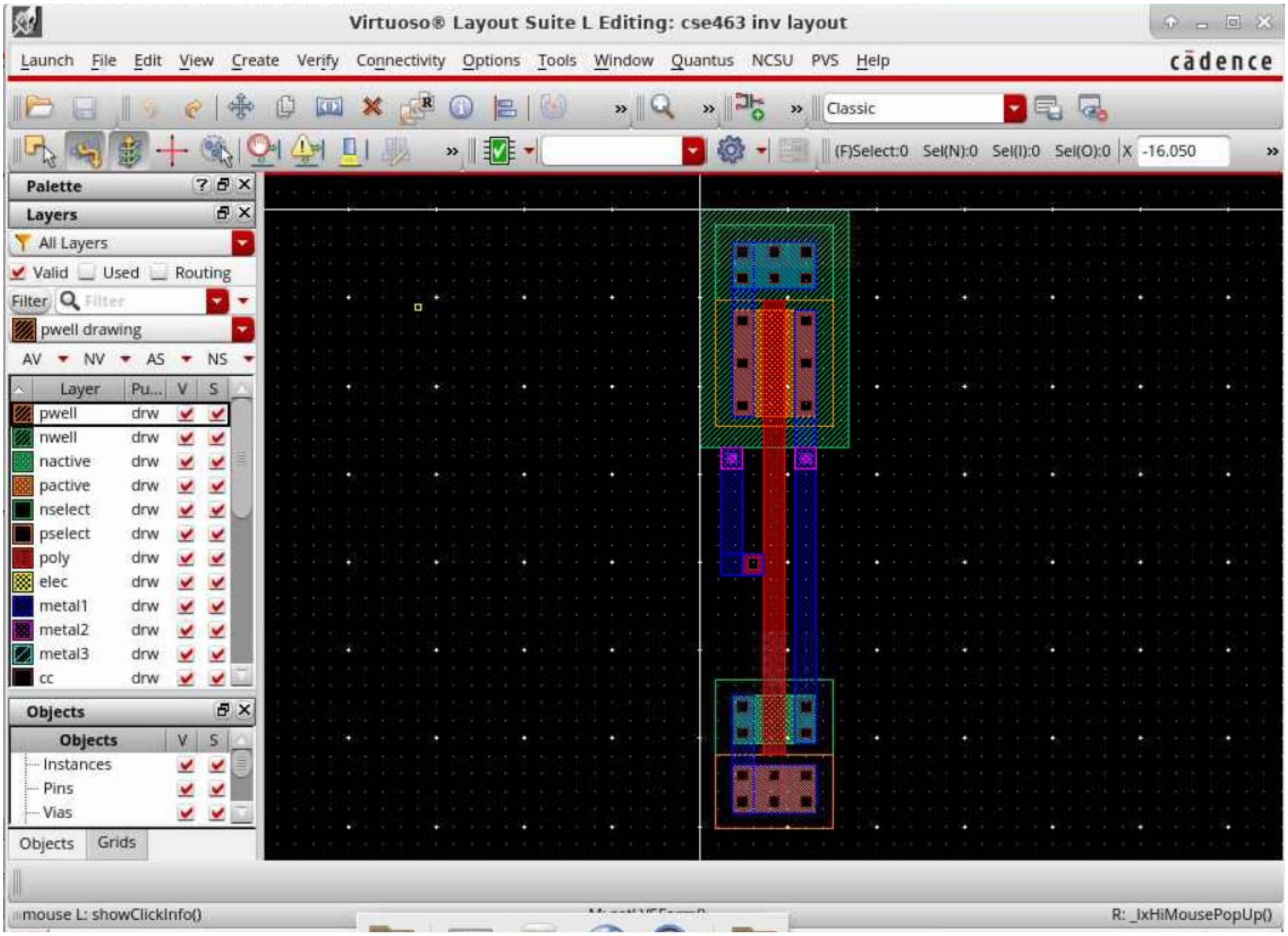
6-3)



+ To find the VIL and VIH, get the derivative of a plot of ‘out’. And then find the slope -1.

+ The result shows that VIL and VIH have the same values that I calculated on 6.1 question!

6-4)


Virtuoso® Layout Suite L Editing: cse463 inv layout
Launch File Edit View Create Verify Connectivity Options Tools Window Quantus NCSU PVS Help
cādence
Classic
Palette
Layers
All Layers
Objects
Grids

(F)Select:0 Sel(N):0 Sel(I):0 Sel(O):0 X -16.050

☑ Valid □ Used □ Routing

Filter: pwell drawing

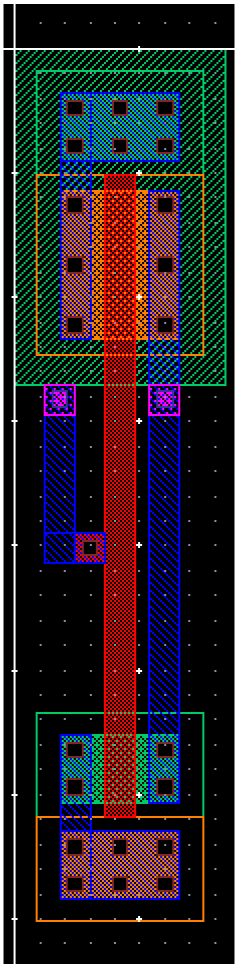
AV NV AS NS

Layer	Pu...	V	S
pwell	drw	✓	✓
nwell	drw	✓	✓
nactive	drw	✓	✓
pactive	drw	✓	✓
nselect	drw	✓	✓
pselect	drw	✓	✓
poly	drw	✓	✓
elec	drw	✓	✓
metal1	drw	✓	✓
metal2	drw	✓	✓
metal3	drw	✓	✓
cc	drw	✓	✓

Objects	V	S
Instances	✓	✓
Pins	✓	✓
Vias	✓	✓

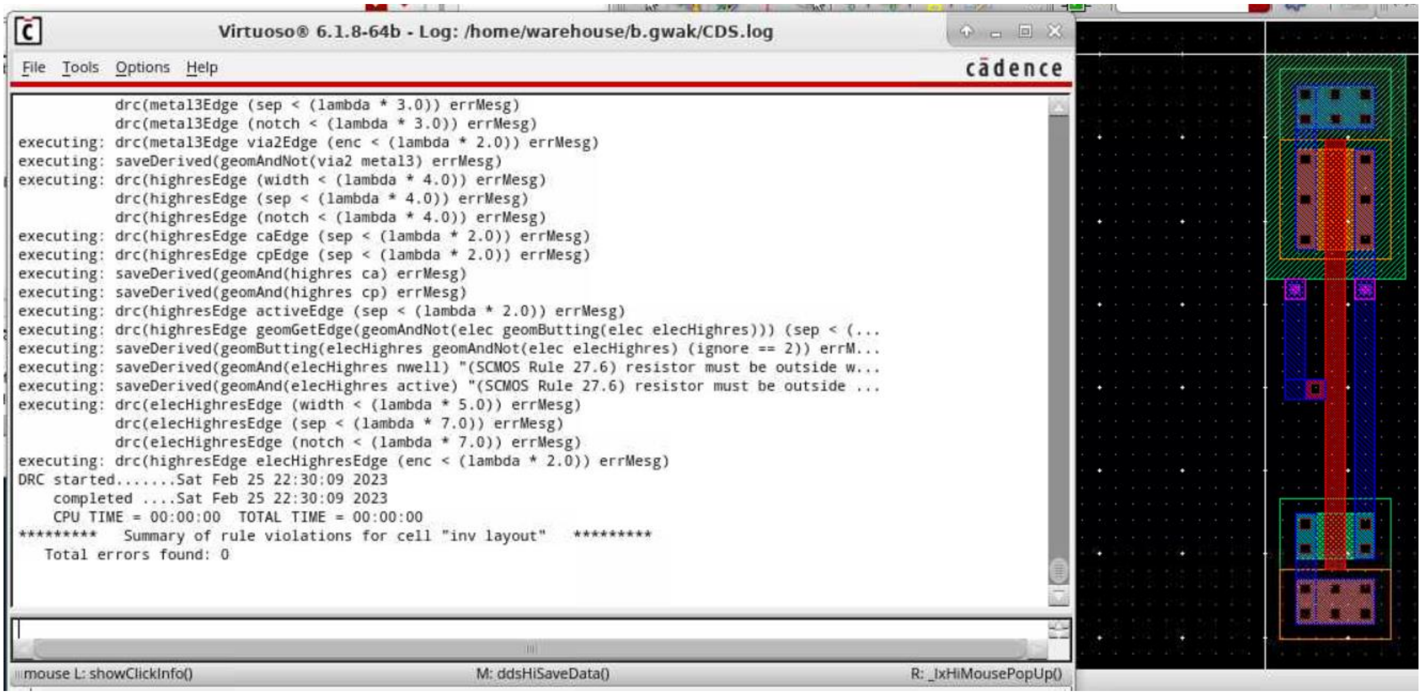
mouse L: showClickInfo()

R: _lxHiMousePopUp()



6-5)

+ FOR DRC check:



```
              drc(metal3Edge (sep < (lambda * 3.0)) errMesg)
              drc(metal3Edge (notch < (lambda * 3.0)) errMesg)
executing: drc(metal3Edge via2Edge (enc < (lambda * 2.0)) errMesg)
executing: saveDerived(geomAndNot(via2 metal3) errMesg)
executing: drc(highresEdge (width < (lambda * 4.0)) errMesg)
              drc(highresEdge (sep < (lambda * 4.0)) errMesg)
              drc(highresEdge (notch < (lambda * 4.0)) errMesg)
executing: drc(highresEdge caEdge (sep < (lambda * 2.0)) errMesg)
executing: drc(highresEdge cpEdge (sep < (lambda * 2.0)) errMesg)
executing: saveDerived(geomAnd(highres ca) errMesg)
executing: saveDerived(geomAnd(highres cp) errMesg)
executing: drc(highresEdge activeEdge (sep < (lambda * 2.0)) errMesg)
executing: drc(highresEdge geomGetEdge(geomAndNot(elec geomButting(elec elecHighres))) (sep < (...
executing: saveDerived(geomButting(elecHighres geomAndNot(elec elecHighres) (ignore == 2)) errM...
executing: saveDerived(geomAnd(elecHighres nwell) "(SCMOS Rule 27.6) resistor must be outside w...
executing: saveDerived(geomAnd(elecHighres active) "(SCMOS Rule 27.6) resistor must be outside ...
executing: drc(elecHighresEdge (width < (lambda * 5.0)) errMesg)
              drc(elecHighresEdge (sep < (lambda * 7.0)) errMesg)
              drc(elecHighresEdge (notch < (lambda * 7.0)) errMesg)
executing: drc(highresEdge elecHighresEdge (enc < (lambda * 2.0)) errMesg)
DRC started.......Sat Feb 25 22:30:09 2023
     completed ....Sat Feb 25 22:30:09 2023
     CPU TIME = 00:00:00  TOTAL TIME = 00:00:00
*********   Summary of rule violations for cell "inv layout"   *********
    Total errors found: 0
```

```
DRC started at Sat Feb 25 22:30:09 2023

Validating hierarchy instantiation for:
library: cse463
cell:    inv
view:    layout
Rules come from library NCSU_TechLib_ami06.
Rules path is divaDRC.rul.
Inclusion limit is set to 1000.
Running layout DRC analysis
Flat mode
Full checking.
executing: nodrc = geomOr("nodrc")
executing: gwell = geomOr(geomAndNot(("gwell" "drawing") nodrc))
executing: nwell = geomOr(geomAndNot(("nwell" "drawing") nodrc))
executing: pwell = geomOr(geomAndNot(("pwell" "drawing") nodrc))
executing: nactive = geomOr(geomAndNot(("nactive" "drawing") nodrc))
executing: pactive = geomOr(geomAndNot(("pactive" "drawing") nodrc))
executing: active = geomOr(geomAndNot(("active" "drawing") nodrc)
nactive pactive)
executing: gselect = geomOr(geomAndNot(("gselect" "drawing") nodrc))
executing: nselect = geomOr(geomAndNot(("nselect" "drawing") nodrc))
```

```
executing: pselect = geomOr(geomAndNot(("pselect" "drawing") nodrc))
executing: poly = geomOr(geomAndNot(("poly" "drawing") nodrc))
executing: metal1 = geomOr(geomAndNot(("metal1" "drawing") nodrc))
executing: cc = geomOr(geomAndNot(("cc" "drawing") nodrc))
executing: metal2 = geomOr(geomAndNot(("metal2" "drawing") nodrc))
executing: via = geomOr(geomAndNot(("via" "drawing") nodrc))
executing: glass = geomOr(geomAndNot(("glass" "drawing") nodrc))
executing: pad = geomOr(geomAndNot(("pad" "drawing") nodrc))
executing: res_id = geomOr("res_id")
executing: metal3 = geomOr(geomAndNot(("metal3" "drawing") nodrc))
executing: via2 = geomOr(geomAndNot(("via2" "drawing") nodrc))
executing: highres = geomOr(geomAndNot(("highres" "drawing") nodrc))
executing: elec = geomOr(geomAndNot(("elec" "drawing") nodrc))
executing: ce = geomOr(geomOr(geomAndNot(("ce" "drawing") nodrc))
geomAnd(cc elec))
executing: cp = geomOr(geomOr(geomAndNot(("cp" "drawing") nodrc))
geomAnd(cc geomAndNot(poly ce)))
executing: ca = geomOr(geomOr(geomAndNot(("ca" "drawing") nodrc))
geomAnd(cc geomAndNot(active ...
executing: nActive = geomAnd(active nselect)
executing: pActive = geomAnd(active pselect)
executing: nBulk = geomOr(nwell)
executing: pBulk = geomOr(geomNot(nwell) geomAndNot(pwell nwell))
executing: nOhmic = geomAnd(nActive nwell)
           nNotOhmic = geomAndNot(nActive nwell)
executing: pOhmic = geomAndNot(pActive nwell)
           pNotOhmic = geomAnd(pActive nwell)
executing: nDiff = geomAndNot(nNotOhmic geomOr(poly elec))
executing: pDiff = geomAndNot(pNotOhmic geomOr(poly elec))
executing: nDiffContact = geomAnd(ca nDiff)
executing: pDiffContact = geomAnd(ca pDiff)
executing: nOhmicContact = geomAnd(ca nOhmic)
executing: pOhmicContact = geomAnd(ca pOhmic)
executing: Gate = geomAnd(geomOr(nNotOhmic pNotOhmic) poly)
executing: fieldPoly = geomAvoiding(poly Gate)
executing: elecGate = geomAnd(geomOr(nNotOhmic pNotOhmic) elec)
executing: fieldElec = geomAvoiding(elec elecGate)
executing: CapacitorElec = geomInside(elec poly)
executing: TransistorElec = geomOverlap(elec geomNot(poly))
executing: fieldPoly = geomAndNot(fieldPoly res_id)
executing: poly = geomAndNot(poly res_id)
```

```
executing: fieldElec = geomAndNot(fieldElec geomOr(res_id highres))
executing: elecRes = geomButting(geomAnd(res_id elec) fieldElec (keep
== 2))
executing: elecHighres = geomButting(geomAnd(highres elec) fieldElec
(keep == 2))
executing: elec = geomAndNot(elec geomOr(res_id highres))
executing: nBulk = geomAndNot(nBulk res_id)
executing: nwell = geomAndNot(nwell res_id)
executing: geomConnect((via nOhmicContact nOhmic nwell nBulk metal1)
(via pOhmicContact pOhmic ...
executing: dubiousData(("gwell" "drawing") "Improperly formed shape
- gwell")
executing: dubiousData(("nwell" "drawing") "Improperly formed shape
- nwell")
executing: dubiousData(("pwell" "drawing") "Improperly formed shape
- pwell")
executing: dubiousData(("active" "drawing") "Improperly formed shape
- active, nactive or pacti...
executing:  dubiousData(("gselect"  "drawing")  "Improperly  formed
shape - gselect")
executing:  dubiousData(("nselect"  "drawing")  "Improperly  formed
shape - nselect")
executing:  dubiousData(("pselect"  "drawing")  "Improperly  formed
shape - pselect")
executing: dubiousData(("poly" "drawing") "Improperly formed shape -
poly")
executing: dubiousData(("metal1" "drawing") "Improperly formed shape
- metal1")
executing: dubiousData(("ca" "drawing") "Improperly formed shape -
ca")
executing: dubiousData(("cp" "drawing") "Improperly formed shape -
cp")
executing: dubiousData(("metal2" "drawing") "Improperly formed shape
- metal2")
executing: dubiousData(("via" "drawing") "Improperly formed shape -
via")
executing: dubiousData(("glass" "drawing") "Improperly formed shape
- glass")
executing:  saveDerived(geomGetNon45(gwell)  "Non-Manhattan  shape  -
gwell")
```

```
executing: saveDerived(geomGetNon45(nwell) "Non-Manhattan shape -
nwell")
executing: saveDerived(geomGetNon45(pwell) "Non-Manhattan shape -
pwell")
executing: saveDerived(geomGetNon45(active) "Non-Manhattan shape -
active, nactive or pactive")
executing: saveDerived(geomGetNon45(gselect) "Non-Manhattan shape -
gselect")
executing: saveDerived(geomGetNon45(nselect) "Non-Manhattan shape -
nselect")
executing: saveDerived(geomGetNon45(pselect) "Non-Manhattan shape -
pselect")
executing: saveDerived(geomGetNon45(poly) "Non-Manhattan shape -
poly")
executing: saveDerived(geomGetNon45(metal1) "Non-Manhattan shape -
metal1")
executing: saveDerived(geomGetNon45(ca) "Non-Manhattan shape - ca")
executing: saveDerived(geomGetNon45(cp) "Non-Manhattan shape - cp")
executing: saveDerived(geomGetNon45(metal2) "Non-Manhattan shape -
metal2")
executing: saveDerived(geomGetNon45(via) "Non-Manhattan shape -
via")
executing: saveDerived(geomGetNon45(glass) "Non-Manhattan shape -
glass")
executing: offGrid(gwell gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(nwell gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(pwell gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(active gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(gselect gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(nselect gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(pselect gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(poly gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(metal1 gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(ca gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(cp gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(metal2 gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(via gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(glass gridRes "(SCMOS Inst) Edge not on grid")
executing: dubiousData(("metal3" "drawing") "Improperly formed shape
- metal3")
```

```
executing: dubiousData(("via2" "drawing") "Improperly formed shape -
via2")
executing: saveDerived(geomGetNon45(metal3) "Non-Manhattan shape -
metal3")
executing: saveDerived(geomGetNon45(via2) "Non-Manhattan shape -
via2")
executing: offGrid(metal3 gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(via2 gridRes "(SCMOS Inst) Edge not on grid")
executing: dubiousData(("elec" "drawing") "Improperly formed shape -
elec")
executing: dubiousData(("ce" "drawing") "Improperly formed shape -
ce")
executing: saveDerived(geomGetNon45(elec) "Non-Manhattan shape -
elec")
executing: saveDerived(geomGetNon45(ce) "Non-Manhattan shape - ce")
executing: offGrid(elec gridRes "(SCMOS Inst) Edge not on grid")
executing: offGrid(ce gridRes "(SCMOS Inst) Edge not on grid")
executing: nwellEdge = geomGetEdge(nwell)
executing: pwellEdge = geomGetEdge(pwell)
executing: activeEdge = geomGetEdge(active)
executing: nselectEdge = geomGetEdge(nselect)
executing: pselectEdge = geomGetEdge(pselect)
executing: polyEdge = geomGetEdge(poly)
executing: metal1Edge = geomGetEdge(metal1)
executing: caEdge = geomGetEdge(ca)
executing: cpEdge = geomGetEdge(cp)
executing: metal2Edge = geomGetEdge(metal2)
executing: viaEdge = geomGetEdge(via)
executing: glassEdge = geomGetEdge(glass)
executing: highresEdge = geomGetEdge(highres)
executing: ceEdge = geomGetEdge(ce)
executing: metal3Edge = geomGetEdge(metal3)
executing: via2Edge = geomGetEdge(via2)
executing: nBulkEdge = geomGetEdge(nBulk)
executing: pBulkEdge = geomGetEdge(pBulk)
executing: nOhmicEdge = geomGetEdge(nOhmic)
executing: pOhmicEdge = geomGetEdge(pOhmic)
executing: nNotOhmicEdge = geomGetEdge(nNotOhmic)
executing: pNotOhmicEdge = geomGetEdge(pNotOhmic)
executing: GateEdge = geomGetEdge(Gate)
executing: fieldPolyEdge = geomGetEdge(fieldPoly)
```

```
executing: CapacitorElecEdge = geomGetEdge(CapacitorElec)
executing: TransistorElecEdge = geomGetEdge(TransistorElec)
executing: elecHighresEdge = geomGetEdge(elecRes coincident elec)
executing: saveDerived(geomAndNot(active geomOr(nselect pselect))
"(DBM Rule 1.1) Active must b...
executing: saveDerived(geomAnd(poly nOhmic) "(DBM Rule 2.0) Poly
cannot overlap ohmic diffusion")
executing: saveDerived(geomAnd(poly pOhmic) "(DBM Rule 2.0) Poly
cannot overlap ohmic diffusion")
executing: saveDerived(geomAnd(elec nOhmic) "(DBM Rule 2.1) Elec
cannot overlap ohmic diffusion")
executing: saveDerived(geomAnd(elec pOhmic) "(DBM Rule 2.1) Elec
cannot overlap ohmic diffusion")
executing: saveDerived(geomAnd(pactive nselect) "(DBM Rule 3.1)
Pactive and Nselect may not ove...
executing: saveDerived(geomAnd(nactive pselect) "(DBM Rule 3.2)
Nactive and Pselect may not ove...
executing: saveDerived(geomAnd(active elec) "(DBM Rule 4.0, AMI
0.6um) Elec and active may not ...
executing: drc(nwellEdge (width < (lambda * 12.0)) errMesg)
           drc(nwellEdge (notch < (lambda * 6.0)) errMesg)
executing: drc(pwellEdge (width < (lambda * 12.0)) errMesg)
           drc(pwellEdge (notch < (lambda * 6.0)) errMesg)
executing: drc(nwell (sep < (lambda * 18.0)) diffNet errMesg)
           drc(nwell (sep < (lambda * 6.0)) sameNet errMesg)
executing: drc(pwell (sep < (lambda * 18.0)) diffNet errMesg)
           drc(pwell (sep < (lambda * 6.0)) sameNet errMesg)
executing: saveDerived(geomAnd(nwell pwell) "(SCMOS Rule 1 note) n-
wells and p-wells may not ov...
executing: drc(activeEdge (width < (lambda * 3.0)) errMesg)
           drc(activeEdge (sep < (lambda * 3.0)) errMesg)
           drc(activeEdge (notch < (lambda * 3.0)) errMesg)
executing: drc(nNotOhmicEdge nBulkEdge (sep < (lambda * 6.0))
errMesg)
executing: drc(pNotOhmicEdge pBulkEdge (sep < (lambda * 6.0))
errMesg)
executing: drc(pBulkEdge nNotOhmicEdge (enc < (lambda * 6.0))
errMesg)
executing: drc(nBulkEdge pNotOhmicEdge (enc < (lambda * 6.0))
errMesg)
executing: drc(nBulkEdge nOhmicEdge (enc < (lambda * 3.0)) errMesg)
```

```
executing: drc(nOhmicEdge pBulkEdge (sep < (lambda * 3.0)) errMesg)
executing: drc(pOhmicEdge nBulkEdge (sep < (lambda * 3.0)) errMesg)
executing: drc(pBulkEdge pOhmicEdge (enc < (lambda * 3.0)) errMesg)
executing: drc(nNotOhmicEdge pOhmicEdge ((0 < sep) < (lambda * 4.0))
errMesg)
executing: drc(pNotOhmicEdge nOhmicEdge ((0 < sep) < (lambda * 4.0))
errMesg)
executing: drc(polyEdge (width < (lambda * 2.0)) errMesg)
           drc(polyEdge (sep < (lambda * 3.0)) errMesg)
           drc(polyEdge (notch < (lambda * 3.0)) errMesg)
executing: drc(polyEdge activeEdge (enc < (lambda * 2.0)) errMesg)
           drc(activeEdge polyEdge (enc < (lambda * 3.0)) errMesg)
           drc(polyEdge activeEdge (sep < (lambda * 1.0)) errMesg)
executing: drc(nselectEdge geomGetEdge(polyEdge inside pNotOhmic)
(sep < (lambda * 3.0)) (app >...
executing: drc(nselectEdge geomGetEdge(polyEdge inside nNotOhmic)
(enc < (lambda * 3.0)) (app >...
executing: drc(pselectEdge geomGetEdge(polyEdge inside nNotOhmic)
(sep < (lambda * 3.0)) (app >...
executing: drc(pselectEdge geomGetEdge(polyEdge inside pNotOhmic)
(enc < (lambda * 3.0)) (app >...
executing: drc(geomOr(nselectEdge pselectEdge) activeEdge (sep <
(lambda * 2.0)) errMesg)
           drc(geomOr(nselectEdge pselectEdge) activeEdge (enc <
(lambda * 2.0)) errMesg)
executing: drc(nselectEdge caEdge (sep < (lambda * 1.0)) errMesg)
           drc(nselectEdge caEdge (enc < (lambda * 1.0)) errMesg)
executing: drc(pselectEdge caEdge (sep < (lambda * 1.0)) errMesg)
           drc(pselectEdge caEdge (enc < (lambda * 1.0)) errMesg)
executing: saveDerived(geomButting(geomAnd(ca nselect) geomAnd(ca
pselect)) errMesg)
executing: drc(nselectEdge (width < (lambda * 2.0)) errMesg)
           drc(nselectEdge (sep < (lambda * 2.0)) errMesg)
           drc(nselectEdge (notch < (lambda * 2.0)) errMesg)
executing: drc(pselectEdge (width < (lambda * 2.0)) errMesg)
           drc(pselectEdge (sep < (lambda * 2.0)) errMesg)
           drc(pselectEdge (notch < (lambda * 2.0)) errMesg)
executing: saveDerived(geomAnd(nselect pselect) errMesg)
executing: drc(cpEdge (width < (lambda * 2.0)) errMesg)
           drc(cpEdge (sep < (lambda * 3.0)) errMesg)
           drc(cpEdge (notch < (lambda * 3.0)) errMesg)
```

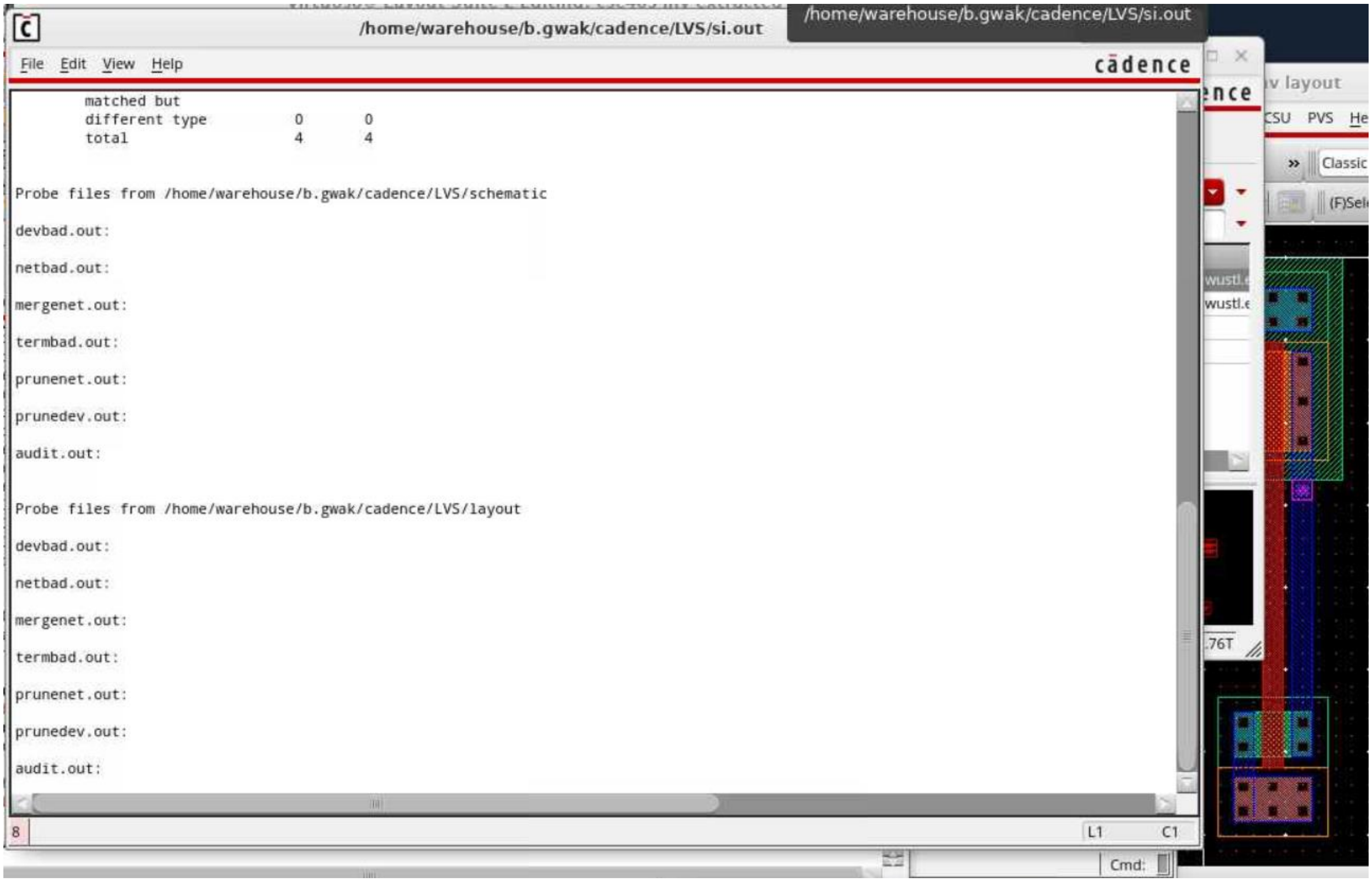
```
executing: drc(cp (area > ((lambda * 2.0 * (lambda * 2.0)) + (lambda
* 0.1 * (lambda * 0.1)))) ...
executing: drc(polyEdge cpEdge (enc < (lambda * 1.0)) errMesg)
           drc(cpEdge polyEdge (sep < (lambda * 5.0)) errMesg)
executing: saveDerived(geomAndNot(cp poly) errMesg)
executing: saveDerived(geomAndNot(cc geomOr(poly elec active))
errMesg)
executing: drc(cpEdge GateEdge (sep < (lambda * 2.0)) errMesg)
executing: drc(cpEdge activeEdge (sep < (lambda * 2.0)) errMesg)
executing: saveDerived(geomAnd(cp active) errMesg)
executing: saveDerived(geomGetLength(drc(cpEdge activeEdge (sep <
(lambda * 3.0))) (length > (l...
executing: drc(caEdge (width < (lambda * 2.0)) errMesg)
           drc(caEdge (sep < (lambda * 3.0)) errMesg)
           drc(caEdge (notch < (lambda * 3.0)) errMesg)
executing: drc(ca (area > ((lambda * 2.0 * (lambda * 2.0)) + (lambda
* 0.1 * (lambda * 0.1)))) ...
executing: drc(activeEdge caEdge (enc < (lambda * 1.0)) errMesg)
           drc(caEdge activeEdge (sep < (lambda * 5.0)) errMesg)
executing: saveDerived(geomAndNot(ca active) errMesg)
executing: drc(caEdge GateEdge (sep < (lambda * 2.0)) errMesg)
executing: saveDerived(geomAnd(ca Gate) errMesg)
executing: drc(caEdge fieldPolyEdge (sep < (lambda * 2.0)) errMesg)
executing: saveDerived(geomAnd(ca fieldPoly) errMesg)
executing: saveDerived(geomGetLength(drc(caEdge fieldPolyEdge (sep <
(lambda * 3.0))) (length >...
executing: drc(caEdge cpEdge (sep < (lambda * 4.0)) errMesg)
executing: saveDerived(geomAnd(ca cp) errMesg)
executing: drc(metal1Edge (width < (lambda * 3.0)) errMesg)
           drc(metal1Edge (sep < (lambda * 3.0)) errMesg)
           drc(metal1Edge (notch < (lambda * 3.0)) errMesg)
executing: drc(metal1Edge cpEdge (enc < (lambda * 1.0)) errMesg)
executing: drc(metal1Edge caEdge (enc < (lambda * 1.0)) errMesg)
executing: saveDerived(geomAndNot(cp metal1) errMesg)
executing: saveDerived(geomAndNot(ca metal1) errMesg)
executing: drc(viaEdge (width < (lambda * 2.0)) errMesg)
           drc(viaEdge (sep < (lambda * 3.0)) errMesg)
executing: drc(via (area > ((lambda * 2.0 * (lambda * 2.0)) + (lambda
* 0.1 * (lambda * 0.1))))...
executing: drc(metal1Edge viaEdge (enc < (lambda * 1.0)) errMesg)
executing: saveDerived(geomAndNot(via metal1) errMesg)
```

```
executing: drc(metal2Edge (width < (lambda * 3.0)) errMesg)
           drc(metal2Edge (sep < (lambda * 3.0)) errMesg)
           drc(metal2Edge (notch < (lambda * 3.0)) errMesg)
executing: drc(metal2Edge viaEdge (enc < (lambda * 1.0)) errMesg)
executing: saveDerived(geomAndNot(via metal2) errMesg)
executing: BondingGlass = geomInside(glass pad)
           ProbeGlass = geomOutside(glass pad)
executing: saveDerived(geomStraddle(glass pad))
executing: BondingPad = geomAndNot(geomSize(BondingGlass 6.0)
geomHoles(BondingGlass))
executing: ProbePad = geomAndNot(geomSize(ProbeGlass 6.0)
geomHoles(ProbeGlass))
executing: Pad = geomOr(BondingPad ProbePad)
executing: BondingPadEdge = geomGetEdge(BondingPad not_over "nodrc")
executing: ProbePadEdge = geomGetEdge(ProbePad not_over "nodrc")
executing: PadEdge = geomGetEdge(Pad not_over "nodrc")
executing: Metal3EdgeNearPad =
geomGetEdge(geomOr(geomGetByLayer("metal3" "glass" 36.0)) not_ov...
executing: Metal2EdgeNearPad =
geomGetEdge(geomOr(geomGetByLayer("metal2" "glass" 36.0)) not_ov...
executing: Metal1EdgeNearPad =
geomGetEdge(geomOr(geomGetByLayer("metal1" "glass" 21.0)) not_ov...
executing: PolyEdgeNearPad =
geomGetEdge(geomOr(geomGetByLayer("poly" "glass" 21.0)) not_over
"...
executing: ActiveEdgeNearPad =
geomGetEdge(geomOr(geomGetByLayer("active" "glass" 21.0)) not_ov...
executing: ElecEdgeNearPad =
geomGetEdge(geomOr(geomGetByLayer("elec" "glass" 21.0)) not_over
"...
executing: drc(BondingPadEdge (width < 60.0) "(SCMOS Rule 10.1)
bonding pad width: 60 um")
executing: drc(ProbePadEdge (width < 20.0) "(SCMOS Rule 10.2) probe
pad width: 20 um")
executing: drc(Metal3EdgeNearPad glassEdge (enc < 6.0) "(SCMOS Rule
10.3) pad enclosure of glas...
executing: saveDerived(geomAndNot(glass metal3) "(SCMOS Rule 10.3)
pad enclosure of glass: 6 um")
executing: drc(PadEdge Metal3EdgeNearPad (sep < 30.0) "(SCMOS Rule
10.4) pad to unrelated metal...
```

```
executing: drc(PadEdge Metal2EdgeNearPad (sep < 30.0) "(SCMOS Rule
10.4) pad to unrelated metal...
executing: drc(PadEdge Metal1EdgeNearPad (sep < 15.0) "(SCMOS Rule
10.5) pad to unrelated metal...
executing: drc(PadEdge PolyEdgeNearPad (sep < 15.0) "(SCMOS Rule
10.5) pad to unrelated poly sp...
executing: drc(PadEdge ActiveEdgeNearPad (sep < 15.0) "(SCMOS Rule
10.5) pad to unrelated activ...
executing: drc(PadEdge ElecEdgeNearPad (sep < 15.0) "(SCMOS Rule
10.5) pad to unrelated elec sp...
executing: drc(CapacitorElecEdge (width < (lambda * 7.0)) errMesg)
           drc(CapacitorElecEdge (sep < (lambda * 3.0)) errMesg)
           drc(CapacitorElecEdge (notch < (lambda * 3.0)) errMesg)
executing: drc(polyEdge CapacitorElecEdge (enc < (lambda * 5.0))
errMesg)
executing: drc(CapacitorElecEdge nBulkEdge (sep < (lambda * 2.0))
errMesg)
           drc(nBulkEdge CapacitorElecEdge (enc < (lambda * 2.0))
errMesg)
executing: drc(CapacitorElecEdge pBulkEdge (sep < (lambda * 2.0))
errMesg)
           drc(pBulkEdge CapacitorElecEdge (enc < (lambda * 2.0))
errMesg)
executing: saveDerived(geomStraddle(CapacitorElec nBulk) errMesg)
executing: saveDerived(geomStraddle(CapacitorElec pBulk) errMesg)
executing: drc(CapacitorElecEdge activeEdge (sep < (lambda * 2.0))
errMesg)
executing: saveDerived(geomAnd(CapacitorElec active) errMesg)
executing: drc(CapacitorElecEdge cpEdge (sep < (lambda * 6.0))
errMesg)
executing: drc(geomGetEdge("elec") geomGetEdge("metal3") (sep <
(lambda * 2.0)) errMesg)
executing: saveDerived(geomOverlap(metal3 elec diffNet) errMesg)
executing: drc(geomGetEdge("elec") geomGetEdge("metal2") (sep <
(lambda * 2.0)) errMesg)
executing: saveDerived(geomOverlap(metal2 elec diffNet) errMesg)
executing: drc(geomGetEdge("elec") geomGetEdge("metal1") (sep <
(lambda * 2.0)) errMesg)
executing: saveDerived(geomOverlap(metal1 elec diffNet) errMesg)
executing: drc(TransistorElecEdge (width < (lambda * 2.0)) errMesg)
           drc(TransistorElecEdge (sep < (lambda * 3.0)) errMesg)
           drc(TransistorElecEdge (notch < (lambda * 3.0)) errMesg)
executing: drc(TransistorElecEdge activeEdge (enc < (lambda * 2.0))
errMesg)
           drc(TransistorElecEdge activeEdge (sep < (lambda * 1.0))
errMesg)
executing: drc(TransistorElecEdge polyEdge (sep < (lambda * 2.0))
errMesg)
           drc(TransistorElecEdge polyEdge (ovlp < (lambda * 2.0))
errMesg)
executing: drc(TransistorElecEdge cpEdge (sep < (lambda * 3.0))
errMesg)
executing: saveDerived(geomAnd(TransistorElec cp) errMesg)
executing: drc(TransistorElecEdge caEdge (sep < (lambda * 3.0))
errMesg)
executing: saveDerived(geomAnd(TransistorElec ca) errMesg)
executing: drc(ceEdge (width < (lambda * 2.0)) errMesg)
           drc(ceEdge (sep < (lambda * 3.0)) errMesg)
           drc(ceEdge (notch < (lambda * 3.0)) errMesg)
executing: drc(ce (area > ((lambda * 2.0 * (lambda * 2.0)) + (lambda
* 0.1 * (lambda * 0.1)))) ...
executing: drc(CapacitorElecEdge ceEdge (enc < (lambda * 3.0))
errMesg)
executing: drc(TransistorElecEdge ceEdge (enc < (lambda * 2.0))
errMesg)
executing: saveDerived(geomAndNot(ce elec) "(SCMOS Rules 13.3,13.4)
electrode enclosure of cont...
executing: drc(ceEdge polyEdge (sep < (lambda * 3.0)) errMesg)
executing: saveDerived(geomOutside(geomAnd(ce poly) CapacitorElec)
errMesg)
executing: drc(ceEdge activeEdge (sep < (lambda * 3.0)) errMesg)
executing: saveDerived(geomAnd(ce active) errMesg)
executing: drc(via2Edge (width < (lambda * 2.0)) errMesg)
           drc(via2Edge (sep < (lambda * 3.0)) errMesg)
executing: drc(via2 (area > ((lambda * 2.0 * (lambda * 2.0)) + (lambda
* 0.1 * (lambda * 0.1)))...
executing: drc(metal2Edge via2Edge (enc < (lambda * 1.0)) errMesg)
executing: saveDerived(geomAndNot(via2 metal2) errMesg)
executing: drc(metal3Edge (width < (lambda * 5.0)) errMesg)
           drc(metal3Edge (sep < (lambda * 3.0)) errMesg)
           drc(metal3Edge (notch < (lambda * 3.0)) errMesg)
executing: drc(metal3Edge via2Edge (enc < (lambda * 2.0)) errMesg)
```

```
executing: saveDerived(geomAndNot(via2 metal3) errMesg)
executing: drc(highresEdge (width < (lambda * 4.0)) errMesg)
           drc(highresEdge (sep < (lambda * 4.0)) errMesg)
           drc(highresEdge (notch < (lambda * 4.0)) errMesg)
executing: drc(highresEdge caEdge (sep < (lambda * 2.0)) errMesg)
executing: drc(highresEdge cpEdge (sep < (lambda * 2.0)) errMesg)
executing: saveDerived(geomAnd(highres ca) errMesg)
executing: saveDerived(geomAnd(highres cp) errMesg)
executing: drc(highresEdge activeEdge (sep < (lambda * 2.0)) errMesg)
executing:      drc(highresEdge       geomGetEdge(geomAndNot(elec
geomButting(elec elecHighres))) (sep < (...
executing:   saveDerived(geomButting(elecHighres   geomAndNot(elec
elecHighres) (ignore == 2)) errM...
executing: saveDerived(geomAnd(elecHighres nwell) "(SCMOS Rule 27.6)
resistor must be outside w...
executing:  saveDerived(geomAnd(elecHighres  active)  "(SCMOS  Rule
27.6) resistor must be outside ...
executing: drc(elecHighresEdge (width < (lambda * 5.0)) errMesg)
           drc(elecHighresEdge (sep < (lambda * 7.0)) errMesg)
           drc(elecHighresEdge (notch < (lambda * 7.0)) errMesg)
executing: drc(highresEdge elecHighresEdge (enc < (lambda * 2.0))
errMesg)
DRC started.......Sat Feb 25 22:30:09 2023
    completed ....Sat Feb 25 22:30:09 2023
    CPU TIME = 00:00:00  TOTAL TIME = 00:00:00
*********     Summary  of  rule  violations  for  cell  "inv  layout"
*********
   Total errors found: 0
```

+ For LVS check:



/home/warehouse/b.gwak/cadence/LVS/si.out
/home/warehouse/b.gwak/cadence/LVS/si.out
File Edit View Help
cādence
L1 C1
Cmd:

```
      matched but
      different type          0       0
      total                   4       4

Probe files from /home/warehouse/b.gwak/cadence/LVS/schematic

devbad.out:

netbad.out:

mergenet.out:

termbad.out:

prunenet.out:

prunedev.out:

audit.out:

Probe files from /home/warehouse/b.gwak/cadence/LVS/layout

devbad.out:

netbad.out:

mergenet.out:

termbad.out:

prunenet.out:

prunedev.out:

audit.out:
```

```
@(#)$CDS: LVS version 6.1.8-64b 08/09/2022 19:10 (sjfhw317) $

Command                                                     line:
/project/engineering/cadence21/IC618/tools.lnx86/dfII/bin/64bit/LVS
-dir      /home/warehouse/b.gwak/cadence/LVS      -l      -s      -t
/home/warehouse/b.gwak/cadence/LVS/layout
/home/warehouse/b.gwak/cadence/LVS/schematic
Like matching is enabled.
Net swapping is enabled.
Using terminal names as correspondence points.
Compiling Diva LVS rules...

    Net-list                       summary                      for
/home/warehouse/b.gwak/cadence/LVS/layout/netlist
       count
        4               nets
        4               terminals
        1               pmos
        1               nmos
    Net-list                       summary                     for
/home/warehouse/b.gwak/cadence/LVS/schematic/netlist
       count
        4              nets
        4              terminals
        1              pmos
        1              nmos


    Terminal correspondence points
    N0        N0        gnd!
    N3        N4        in
    N2        N2        out
    N1        N1        vdd!

Devices in the netlist but not in the rules:
        pcapacitor
Devices in the rules but not in the netlist:
        cap nfet pfet nmos4 pmos4

The net-lists match.

                             layout  schematic
                                instances
        un-matched              0       0
        rewired                 0       0
        size errors             0       0
        pruned                  0       0
        active                  2       2
        total                   2       2

                                  nets
        un-matched              0       0
        merged                  0       0
        pruned                  0       0
        active                  4       4
        total                   4       4

                                terminals
        un-matched              0       0
        matched but
        different type          0       0
        total                   4       4


Probe files from /home/warehouse/b.gwak/cadence/LVS/schematic

devbad.out:

netbad.out:

mergenet.out:

termbad.out:

prunenet.out:

prunedev.out:

audit.out:


Probe files from /home/warehouse/b.gwak/cadence/LVS/layout

devbad.out:

netbad.out:

mergenet.out:

termbad.out:

prunenet.out:

prunedev.out:

audit.out:
```

7)- 5.4 (For 563M)

+ We know the following:

$V_{T0} = 0.48\,V$

$\mu_n C_{ox} = 102\,\frac{\mu A}{V^2}$

$(\frac{W}{L})_{load} = 3$

$(\frac{W}{L})_{driver} = 9$

$\gamma = \ 0\,V^{\frac{1}{2}}$

$|2\Phi_F| = \ 1.011\,V$

$\lambda = 0$

$V_{DD} = 1.2\,V$

$E_c L_n = 0.45\,V$

+ (a), For V_{OH},V_{OL}:

Consider that load transistor is in saturation.

$$\frac{k_{load}}{2}(V_{DD} - \ V_{OH} - \ V_T)^2 = 0$$

So, $V_{OH} = V_{DD} - \ V_T$

$$\boxed{V_{OH} = 1.2 - 0.48 = 0.72V}$$

Consider the load is in saturation and driver is in linear when input is V_{OH}.

$$\frac{k_{load}}{2}(V_{DD} - \ V_{OL} - \ V_T)^2 = \frac{k_{driver}}{2}\left[2(V_{OH} - V_{T0})V_{OL} - {V_{OL}}^2\right]$$

$$(V_{DD} - \ V_{OL} - \ V_T)^2 = \frac{k_{driver}}{k_{load}}\left[2(V_{OH} - V_{T0})V_{OL} - {V_{OL}}^2\right]$$

Since, $\frac{k_{driver}}{k_{load}} = \frac{\mu_n C_{ox}\left(\frac{W}{L}\right)_{driver}}{\mu_n C_{ox}\left(\frac{W}{L}\right)_{load}} = \frac{\left(\frac{W}{L}\right)_{driver}}{\left(\frac{W}{L}\right)_{load}} = \frac{9}{3} = 3$

$$(1.2 - \ V_{OL} - \ 0.48)^2 = 3 \cdot \left[2(0.72 - 0.48)V_{OL} - {V_{OL}}^2\right]$$

$${V_{OL}}^2 - 1.44\,V_{OL} + 0.5184 = 1.44\,V_{OL} - 3{V_{OL}}^2$$

$$\boxed{V_{OL} = 0.36\,V}$$

+ (b), For noise margin :

Consider load is in saturation and driver is in saturation.

$$\frac{k_{load}}{2}(V_{DD} - \ V_{out} - \ V_{T0})^2 = \frac{k_{driver}}{2}(V_{in} - \ V_{T0})^2$$

$(V_{DD} - V_{out} - V_{T0})^2 = 3(V_{in} - V_{T0})^2$
$(1.2 - V_{out} - 0.48)^2 = 3(V_{in} - 0.48)^2$
$(0.72 - V_{out})^2 = 3(V_{in} - 0.48)^2$ ------- (X)

Take derivation with respect to Vin on both side.

$$\frac{dV_{out}}{dV_{in}} = -1$$

$$2(0.72 - V_{out})\frac{dV_{out}}{dV_{in}} = 6(V_{in} - 0.48)$$

$-2(0.72 - V_{out}) = 6(V_{in} - 0.48)$
$-(0.72 - V_{out}) = 3(V_{in} - 0.48)$
$V_{out} = 3V_{in} - 1.44 + 0.72$
$V_{out} = 3V_{in} - 0.72$

Put $V_{out} = 3V_{in} - 0.72$ back into (X) equation.
$(0.72 - V_{out})^2 = 3(V_{in} - 0.48)^2$ ------- (X)
$(0.72 - 3V_{in} + 0.72)^2 = 3(V_{in} - 0.48)^2$
$V_{in} = V_{IL} = 0.48\,V$

Therefore, noise margin :

$$\boxed{NM_L = V_{IL} - V_{OL} = 0.48 - 0.36 = 0.12\,V}$$

+ (c), For current:

$$I_{load} = \frac{k_{load}}{2}(V_{GS,L} - V_{T,L})^2 = \frac{k_{load}}{2}(V_{DD} - V_{out} - V_{T,L})^2$$

$$= \frac{\mu_n C_{ox}\left(\frac{W}{L}\right)_{load}}{2}(V_{DD} - V_{OL} - V_{T,L})^2$$

$$= \frac{102 \cdot 10^{-6} \cdot 3}{2}(1.2 - 0.36 - 0.48)^2$$

$= 0.0000198288$

$\boxed{= 19.8288\,\mu A}$

8) - 5.7(For 563M)

+ we know following:

nMOS: $V_{T0,n} = 0.48V, \mu_n C_{ox} = 102\,\mu A/V^2, (\frac{W}{L})_n = 10$

$pMOS: V_{T0,p} = -0.46V, \mu_p C_{ox} = 51.6\,\mu A/V^2, (\frac{W}{L})_p = 19$

$E_{C,n}L_n = 0.4V$

$E_{C,p}L_p = 1.8V$

$L_n = L_p = 60\,nm$

+ (a) For $\frac{W_n}{W_p}$:

We use the below equation.

$$V_{th} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}(V_{DD} - |V_{T0,p}|)}{1 + \sqrt{\frac{1}{k_R}}}$$

$$0.5 = \frac{0.48 + \sqrt{\frac{1}{k_R}}(1.2 - 0.46)}{1 + \sqrt{\frac{1}{k_R}}}$$

$k_R = 144$

$$k_R = \frac{k_n}{k_p} = \frac{\mu_n C_{ox}(\frac{W}{L})_n}{\mu_p C_{ox}(\frac{W}{L})_p} = \frac{102(\frac{W}{L})_n}{51.6(\frac{W}{L})_p} = \frac{102W_n}{51.6W_p} = 144$$

$$\boxed{\frac{W_n}{W_p} = 144 \cdot \frac{51.6}{102} = 72.84}$$

+ (b) For V_{th}:

$$k_R = \frac{k_n}{k_p} = \frac{\mu_n C_{ox}(\frac{W}{L})_n}{\mu_p C_{ox}(\frac{W}{L})_p}$$

$$k_R = \frac{k_n}{k_p} = \frac{102 \cdot 10}{51.6 \cdot 19} = 1.04$$

$$\sqrt{\frac{1}{k_R}} = 0.98$$

$$V_{th} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}(V_{DD} - |V_{T0,p}|)}{1 + \sqrt{\frac{1}{k_R}}}$$

$$V_{T0,n} = 0.48 \pm 15\%$$
$$V_{T0,p} = -0.46 \pm 20\%$$

$$V_{T0,n}max = 0.48 + 0.48 \cdot 15\% = 0.552$$
$$V_{T0,n}min = 0.48 - 0.48 \cdot 15\% = 0.408$$
$$V_{T0,p}max = -0.46 + 0.46 \cdot 20\% = -0.368$$
$$V_{T0,p}min = -0.46 - 0.46 \cdot 20\% = -0.552$$

For upper limit, $V_{T0,n}$ is max and $V_{T0,p}$ is min.

$$V_{th,upper} = \frac{0.552 + \sqrt{\frac{1}{k_R}}(V_{DD} - |V_{T0,p}|)}{1 + \sqrt{\frac{1}{k_R}}}$$

$$\boxed{V_{th,upper} = \frac{0.552 + 0.98(1.2 - 0.368)}{1 + 0.98} = 0.69\,V}$$

For lower limit, $V_{T0,n}$ is min and $V_{T0,p}$ is max.

$$V_{th,lower} = \frac{V_{T0,n} + \sqrt{\frac{1}{k_R}}(V_{DD} - |V_{T0,p}|)}{1 + \sqrt{\frac{1}{k_R}}}$$

$$\boxed{V_{th,lower} = \frac{0.408 + 0.98(1.2 - 0.552)}{1 + 0.98} = 0.52\,V}$$