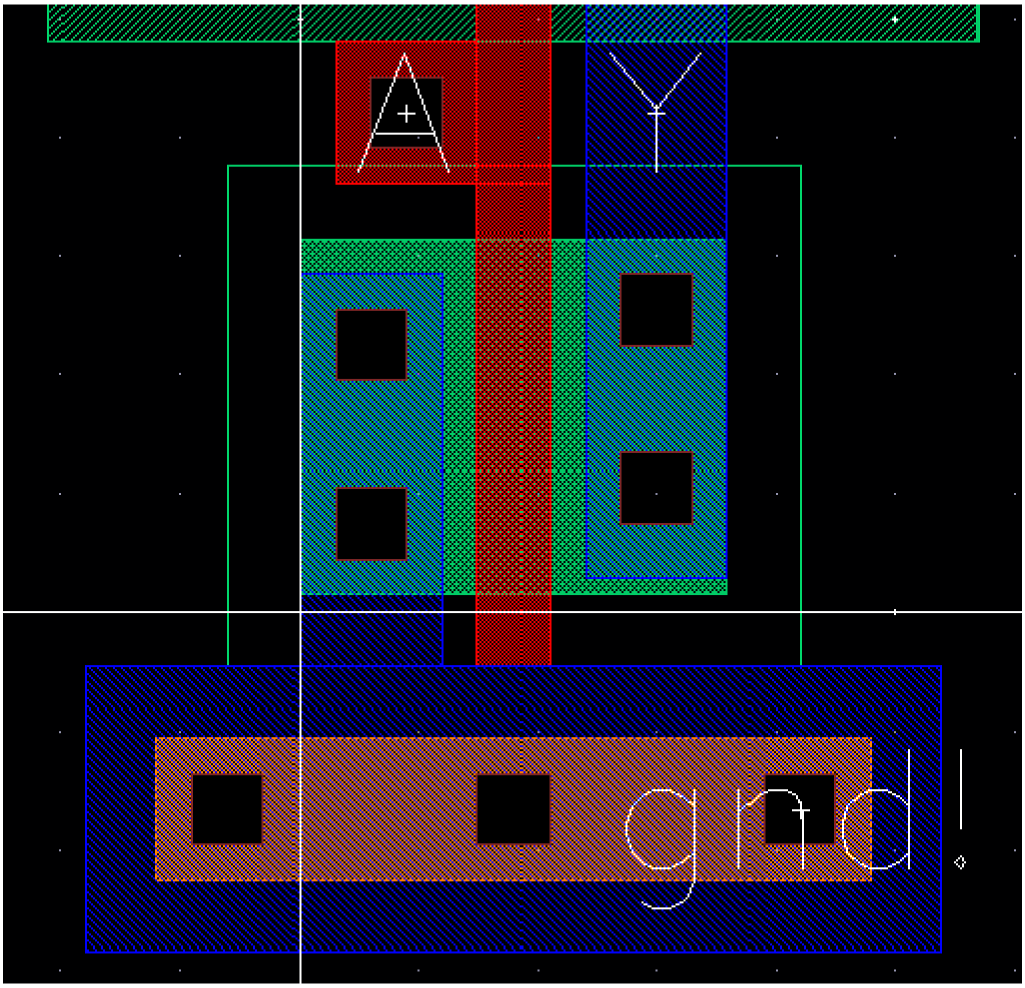
ECE520: Lab 7 Inverter Analysis

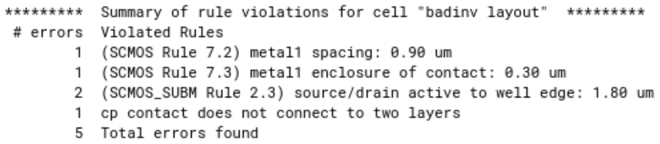
Name: Arijit Sengupta, ID: 001441748

1) Analysing/correcting DRC errors

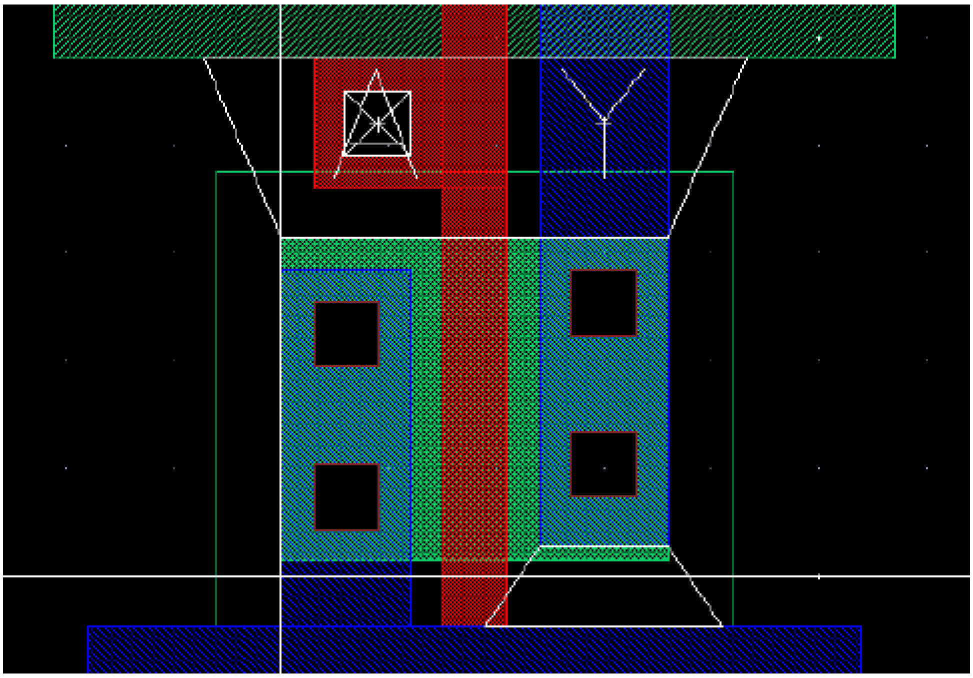
First, we delete the metal1 layer of input A. Next, we will get the nmos too close to above.



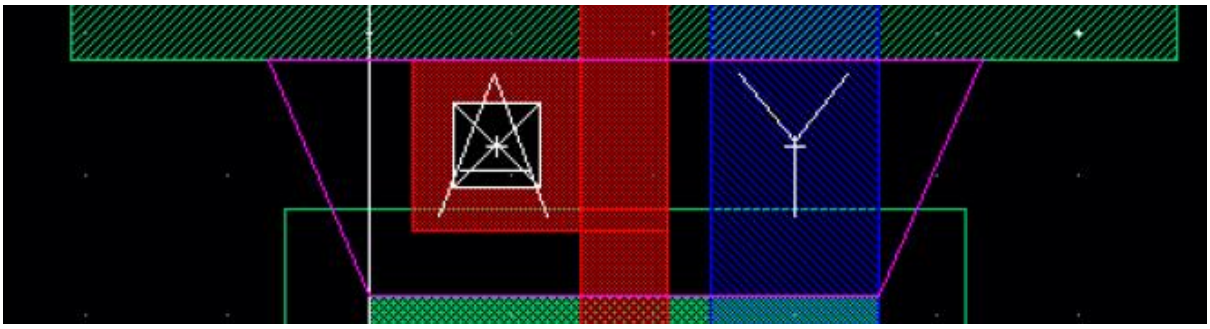
When we verify the design by Verify -> DRC, it shows 5 errors.



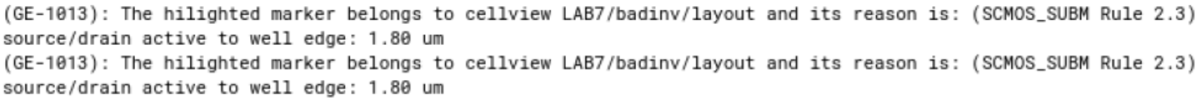
```
*********  Summary of rule violations for cell "badinv layout"  *********
 # errors  Violated Rules
        1  (SCMOS Rule 7.2) metal1 spacing: 0.90 um
        1  (SCMOS Rule 7.3) metal1 enclosure of contact: 0.30 um
        2  (SCMOS_SUBM Rule 2.3) source/drain active to well edge: 1.80 um
        1  cp contact does not connect to two layers
        5  Total errors found
```



The polygon will change its colour to purple, which will highlight which error we are currently analysing.



The following “explanation” will print in the CIW windows:

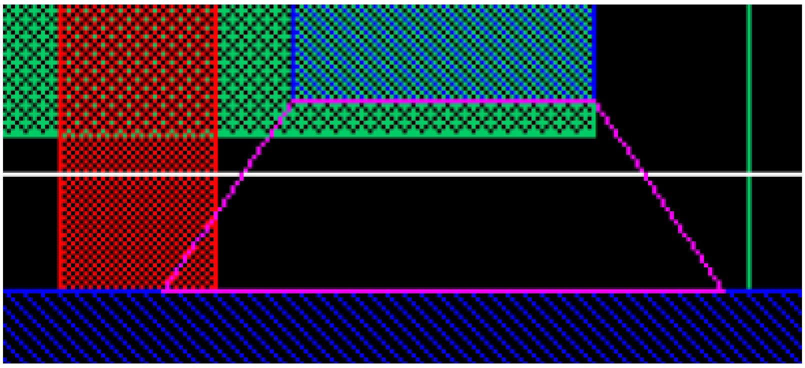


```
(GE-1013): The hilighted marker belongs to cellview LAB7/badinv/layout and its reason is: (SCMOS_SUBM Rule 2.3)
source/drain active to well edge: 1.80 um
(GE-1013): The hilighted marker belongs to cellview LAB7/badinv/layout and its reason is: (SCMOS_SUBM Rule 2.3)
source/drain active to well edge: 1.80 um
```

We violated MOSIS Sub-micron SCMOS technology (SCMOS_SUBM) DRC Rule Number 2.3, which states:

- nactive layer should be at least 1.8µm (6λ) far from the closest nwell.
- By bringing the nmos too close above, the nactive region of the nmos got too close to the nwell of the pmos.

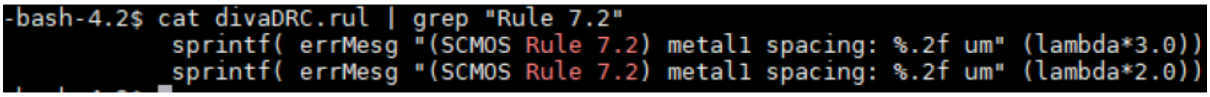
Looking at the other 4 errors, we single-click on the bottom polygon; it will turn purple and will give this explanation in CIW:



This violates DRC Rule Number 7.2, which states that metal1 spacing must be at least 0.9µm (3λ).

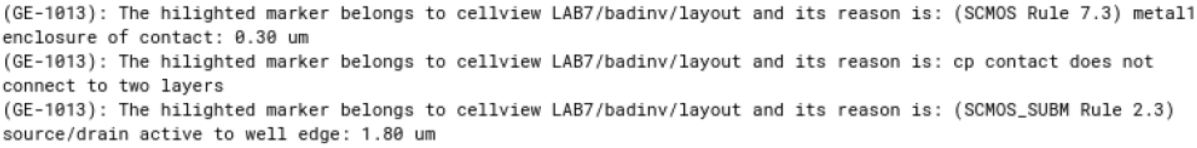
It can be checked in the divaDRC.rul file from the terminal by:

cat divaDRC.rul | grep “Rule7.2”



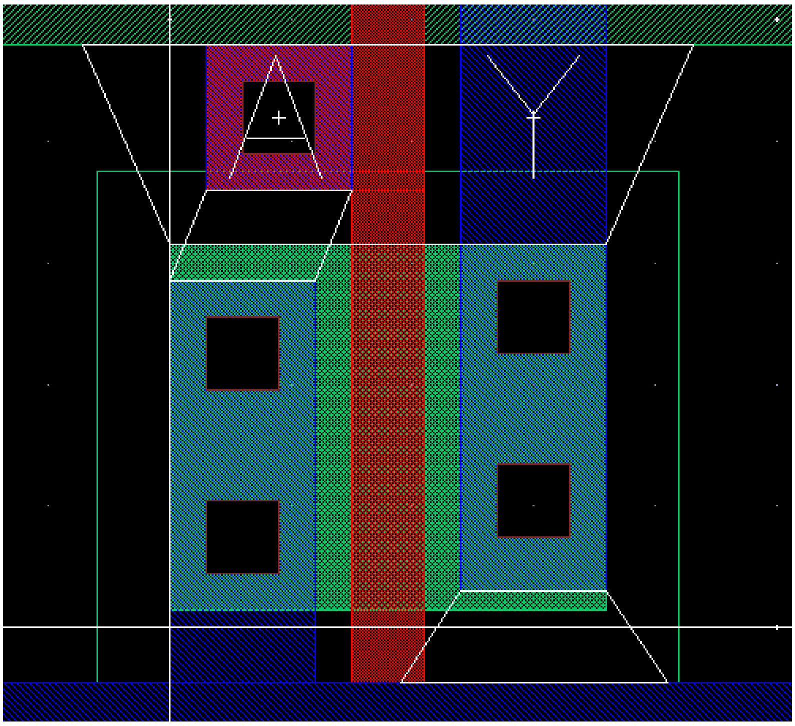
```
-bash-4.2$ cat divaDRC.rul | grep "Rule 7.2"
            sprintf( errMesg "(SCMOS Rule 7.2) metal1 spacing: %.2f um" (lambda*3.0))
            sprintf( errMesg "(SCMOS Rule 7.2) metal1 spacing: %.2f um" (lambda*2.0))
```

The other three errors are all bunched up in the area where we have the A input.

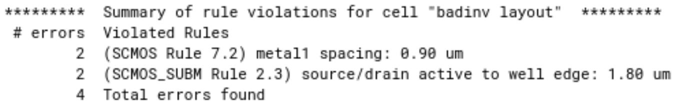


```
(GE-1013): The hilighted marker belongs to cellview LAB7/badinv/layout and its reason is: (SCMOS Rule 7.3) metal1
enclosure of contact: 0.30 um
(GE-1013): The hilighted marker belongs to cellview LAB7/badinv/layout and its reason is: cp contact does not
connect to two layers
(GE-1013): The hilighted marker belongs to cellview LAB7/badinv/layout and its reason is: (SCMOS_SUBM Rule 2.3)
source/drain active to well edge: 1.80 um
```

It is telling to put a **metal1** around the **cp** (poly contact) that at least surrounds the **cp** by 1λ on every side, so we do that and rerun the DRC.

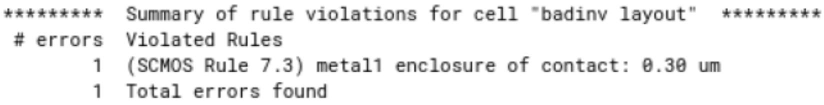


Doing that, we end up with 4 errors instead of 5.

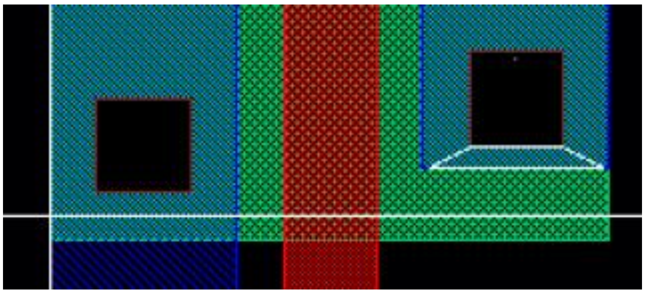


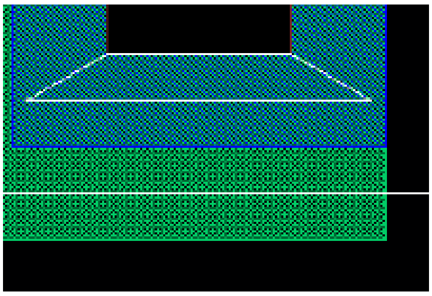
```
*********  Summary of rule violations for cell "badinv layout"  *********
 # errors  Violated Rules
        2  (SCMOS Rule 7.2) metal1 spacing: 0.90 um
        2  (SCMOS_SUBM Rule 2.3) source/drain active to well edge: 1.80 um
        4  Total errors found
```

Now, we hit s (stretch) and stretch the nmos below by λ/2 and run DRC again. We get 1 error.



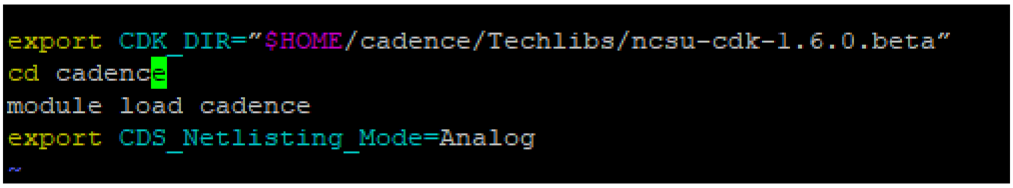
```
*********  Summary of rule violations for cell "badinv layout"  *********
 # errors  Violated Rules
        1  (SCMOS Rule 7.3) metal1 enclosure of contact: 0.30 um
        1  Total errors found
```





2) Extracting the inverter layout

We add the line **export CDS_Netlisting_Mode=Analog** to the ~/.bash_profile

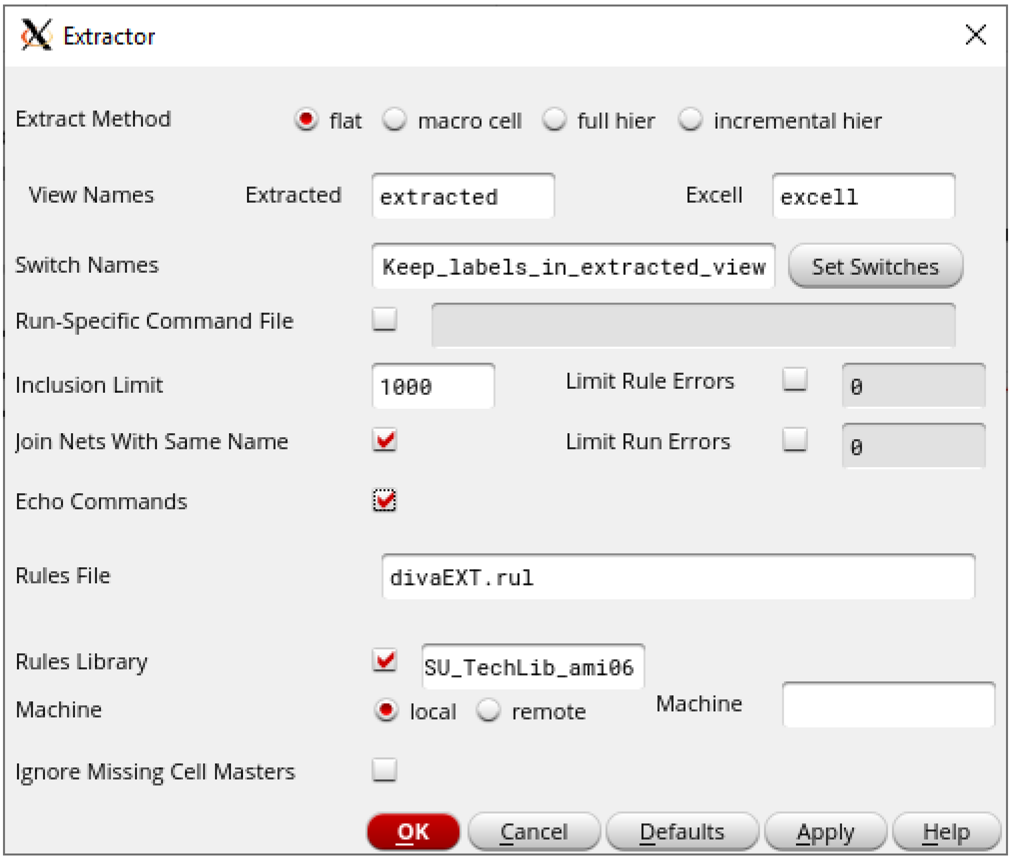


```
export CDK_DIR=”$HOME/cadence/Techlibs/ncsu-cdk-1.6.0.beta”
cd cadence
module load cadence
export CDS_Netlisting_Mode=Analog
~
```

We draw pins for vdd!, gnd!, A, Y, setting A to input, Y to output. vdd! and gnd! both to input/output.

Our goal is to determine whether the schematic of the inverter represents the same circuit as the layout of the same inverter cell, we just drew.

- The first step in doing so is to extract the layout. Go to Verify -> Extract
- Click Set Switches and choose Extract_parasitic_caps Keep_labels_in_extracted_view
- Set Rules file to divaEXT.rul



Extractor

Extract Method ● flat ○ macro cell ○ full hier ○ incremental hier

View Names Extracted extracted Excell excell

Switch Names Keep_labels_in_extracted_view Set Switches

Run-Specific Command File □

Inclusion Limit 1000 Limit Rule Errors □ 0

Join Nets With Same Name ☑ Limit Run Errors □ 0

Echo Commands ☑

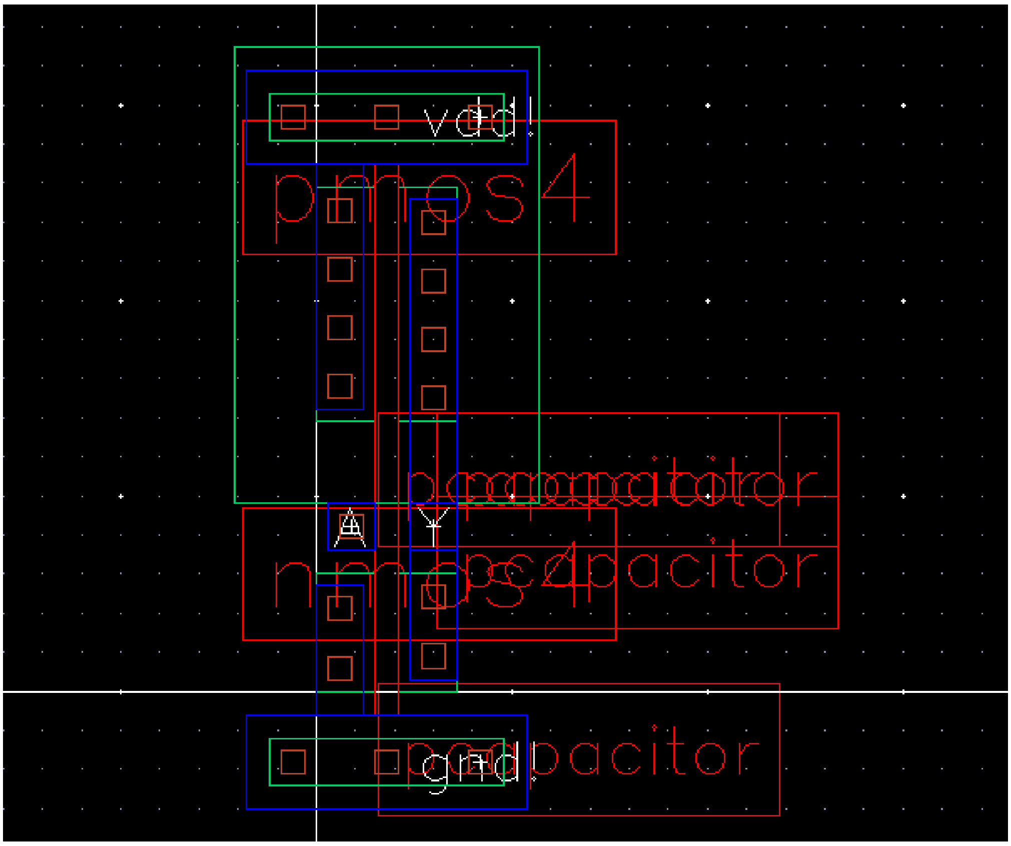
Rules File divaEXT.rul

Rules Library ☑ SU_TechLib_ami06

Machine ● local ○ remote Machine

Ignore Missing Cell Masters □

With these options, the extracted layout will be saved in the extracted view.



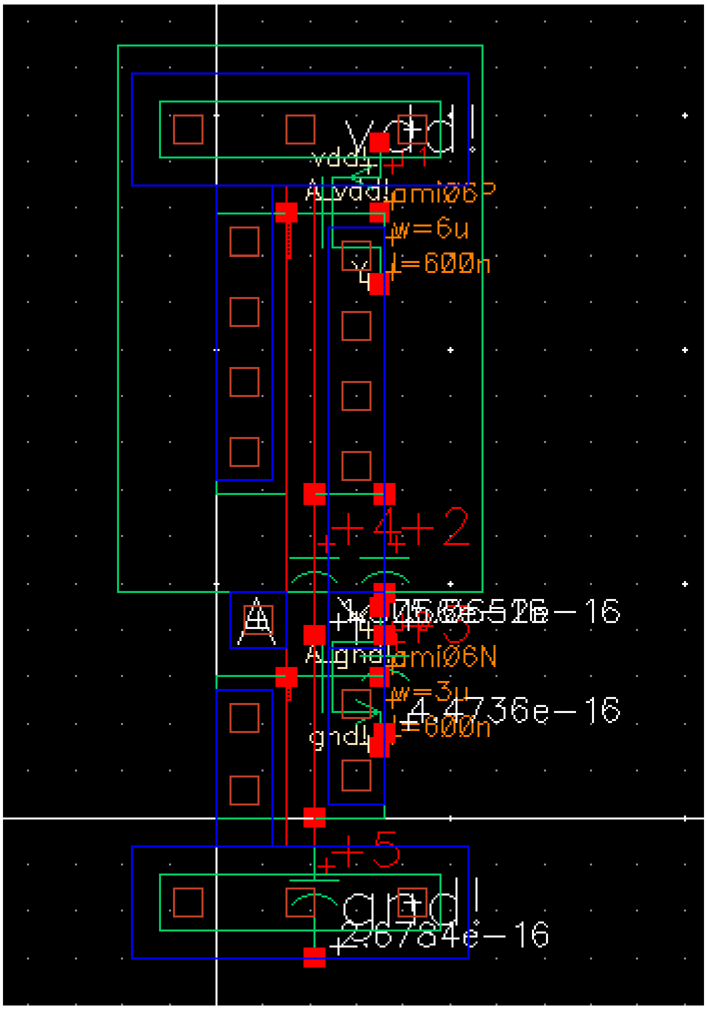
The values of the parasitic capacitors are:

- pcapacitor (instance “+2”, library “NCSU_Analog_...) – **506.52aF**
- pcapacitor (instance “+3”, library “NCSU_Analog_...) – **447.36aF**
- pcapacitor (instance “+4”, library “NCSU_Analog_...) – **150.66aF**
- pcapacitor (instance “+5”, library “NCSU_Analog_...) – **267.84aF**

According to the extractor, this layout represents 6 parts, all from the same NCSU_Analog_Parts library.

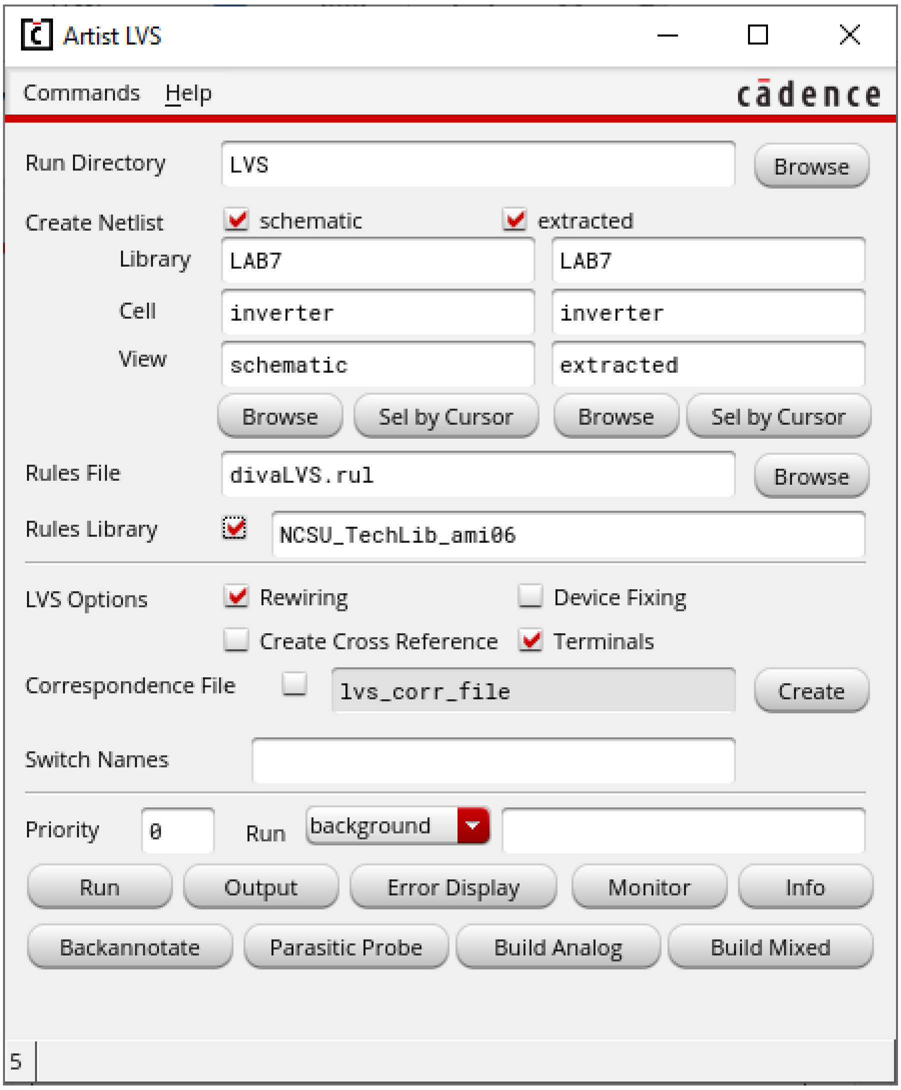
- pmos4: which is our pmos transistor on top
- nmos4: which is our nmos transistor on the bottom
- pcapacitor: which is a cell that exists in the NCSU_Analog_Parts library, with the sole intention to be able to model the parasitic capacitors in a circuit (layout).

Hitting SHIFT+F we switch to this detailed view:



3) LVS: Layout vs. Schematic

The tool that facilitates this comparison is called LVS (Layout vs. Schematic).



Run Directory: LVS

Create Netlist: ☑ schematic ☑ extracted

	schematic	extracted
Library	LAB7	LAB7
Cell	inverter	inverter
View	schematic	extracted

Rules File: divaLVS.rul

Rules Library: ☑ NCSU_TechLib_ami06

LVS Options: ☑ Rewiring □ Device Fixing □ Create Cross Reference ☑ Terminals

Correspondence File: □ lvs_corr_file

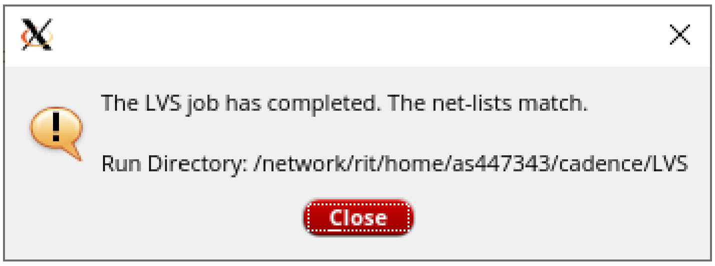
Switch Names:

Priority: 0 Run: background

5

When Run, it returns the following success message:

```
LVS job is now started...
The LVS job has completed. The net-lists match.

Run Directory: /network/rit/home/as447343/cadence/LVS
```

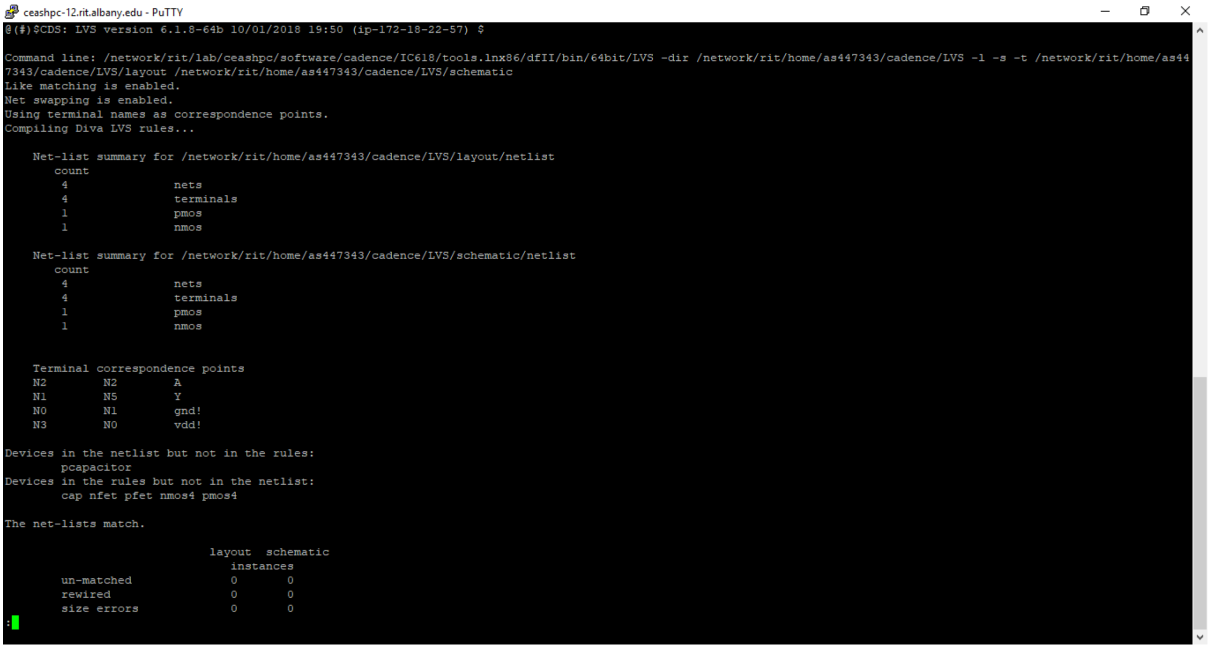


The LVS job has completed. The net-lists match.

Run Directory: /network/rit/home/as447343/cadence/LVS

We go to the ~/cadence/LVS directory and look at the si.out file.

- cd ~/cadence/LVScat si.out | less

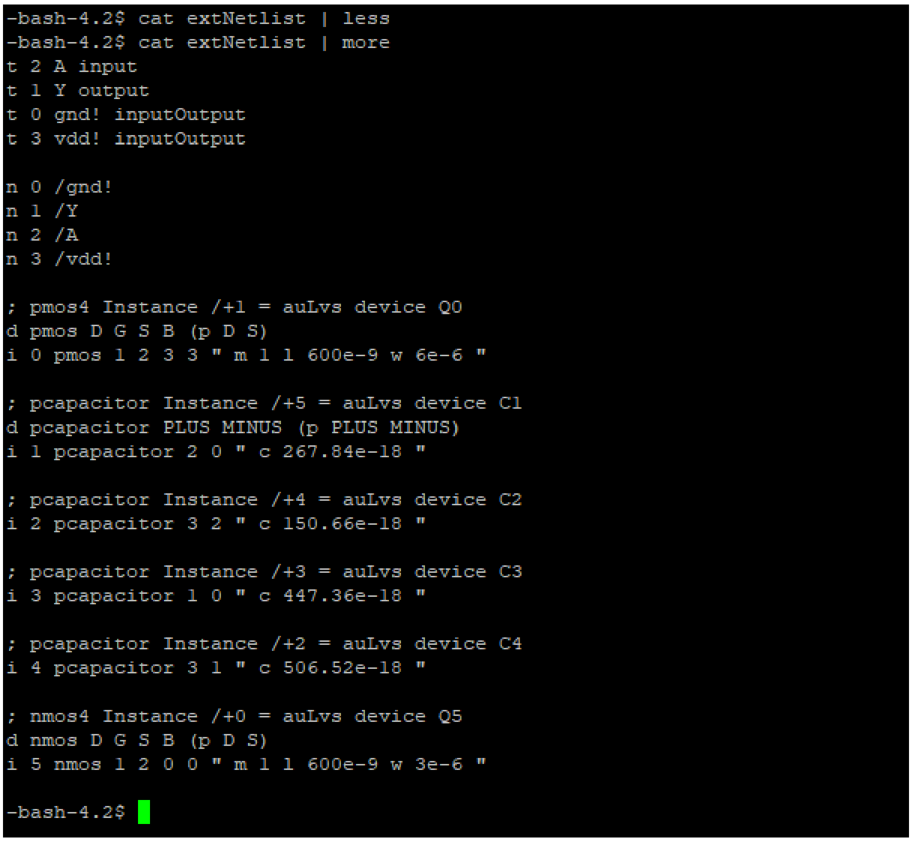


```
@(#)$CDS: LVS version 6.1.8-64b 10/01/2018 19:50 (ip-172-18-22-57) $

Command line: /network/rit/lab/ceashpc/software/cadence/IC618/tools.lnx86/dfII/bin/64bit/LVS -dir /network/rit/home/as447343/cadence/LVS -l -s -t /network/rit/home/as44
7343/cadence/LVS/layout /network/rit/home/as447343/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 /network/rit/home/as447343/cadence/LVS/layout/netlist
       count
        4               nets
        4               terminals
        1               pmos
        1               nmos

    Net-list summary for /network/rit/home/as447343/cadence/LVS/schematic/netlist
       count
        4               nets
        4               terminals
        1               pmos
        1               nmos


    Terminal correspondence points
    N2        N2        A
    N1        N5        Y
    N0        N1        gnd!
    N3        N0        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
:
```

This is the report from LVS.

Now, we look at the extracted netlist.

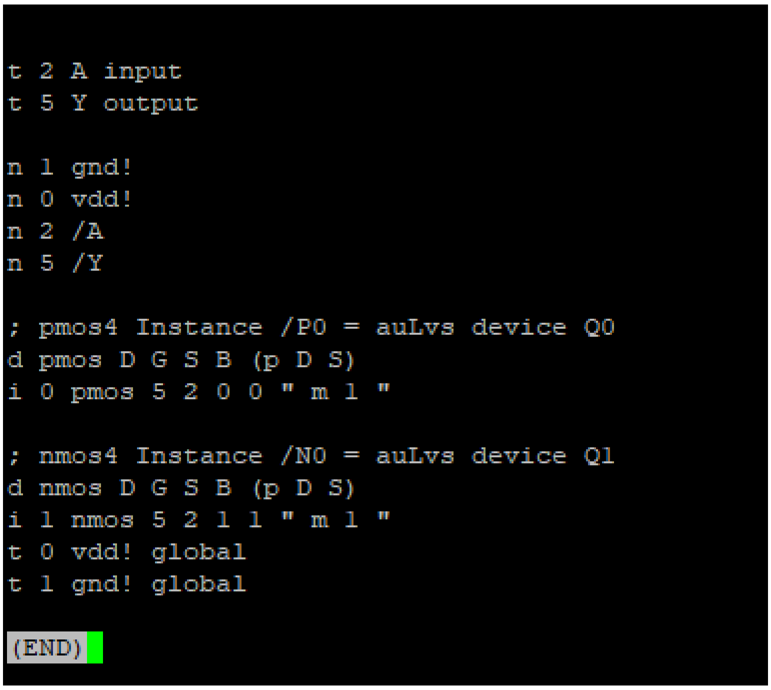
- cd ~/cadence/LVS/layout
- cat extNetlist | more



```
-bash-4.2$ cat extNetlist | less
-bash-4.2$ cat extNetlist | more
t 2 A input
t 1 Y output
t 0 gnd! inputOutput
t 3 vdd! inputOutput

n 0 /gnd!
n 1 /Y
n 2 /A
n 3 /vdd!

; pmos4 Instance /+1 = auLvs device Q0
d pmos D G S B (p D S)
i 0 pmos 1 2 3 3 " m 1 l 600e-9 w 6e-6 "

; pcapacitor Instance /+5 = auLvs device C1
d pcapacitor PLUS MINUS (p PLUS MINUS)
i 1 pcapacitor 2 0 " c 267.84e-18 "

; pcapacitor Instance /+4 = auLvs device C2
i 2 pcapacitor 3 2 " c 150.66e-18 "

; pcapacitor Instance /+3 = auLvs device C3
i 3 pcapacitor 1 0 " c 447.36e-18 "

; pcapacitor Instance /+2 = auLvs device C4
i 4 pcapacitor 3 1 " c 506.52e-18 "

; nmos4 Instance /+0 = auLvs device Q5
d nmos D G S B (p D S)
i 5 nmos 1 2 0 0 " m 1 l 600e-9 w 3e-6 "

-bash-4.2$
```

Similarly, we look at the netlist of the schematic using these commands:

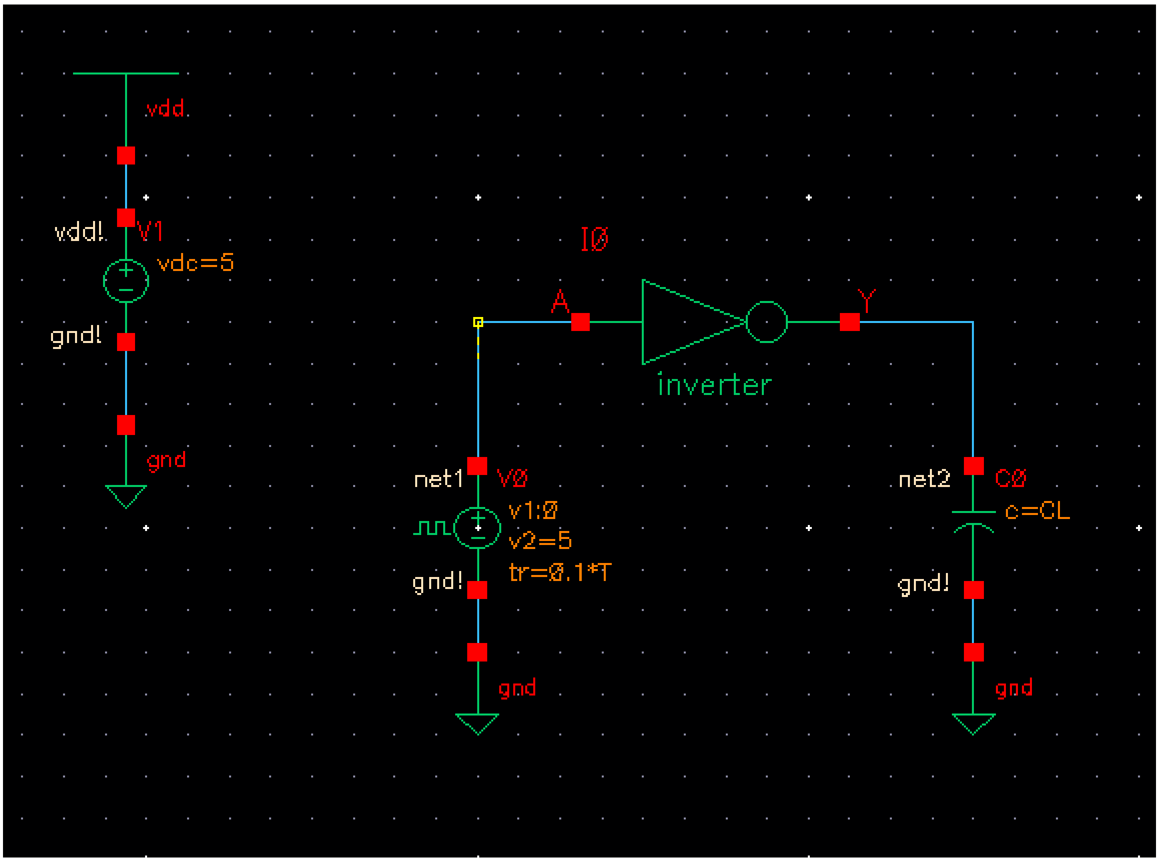
- ➢ cd ~/cadence/LVS/schematic
- ➢ cat extNetlist | less



```
t 2 A input
t 5 Y output

n 1 gnd!
n 0 vdd!
n 2 /A
n 5 /Y

; pmos4 Instance /P0 = auLvs device Q0
d pmos D G S B (p D S)
i 0 pmos 5 2 0 0 " m 1 "

; nmos4 Instance /N0 = auLvs device Q1
d nmos D G S B (p D S)
i 1 nmos 5 2 1 1 " m 1 "
t 0 vdd! global
t 1 gnd! global

(END)
```

4) Analyzing the inverter using ADE L

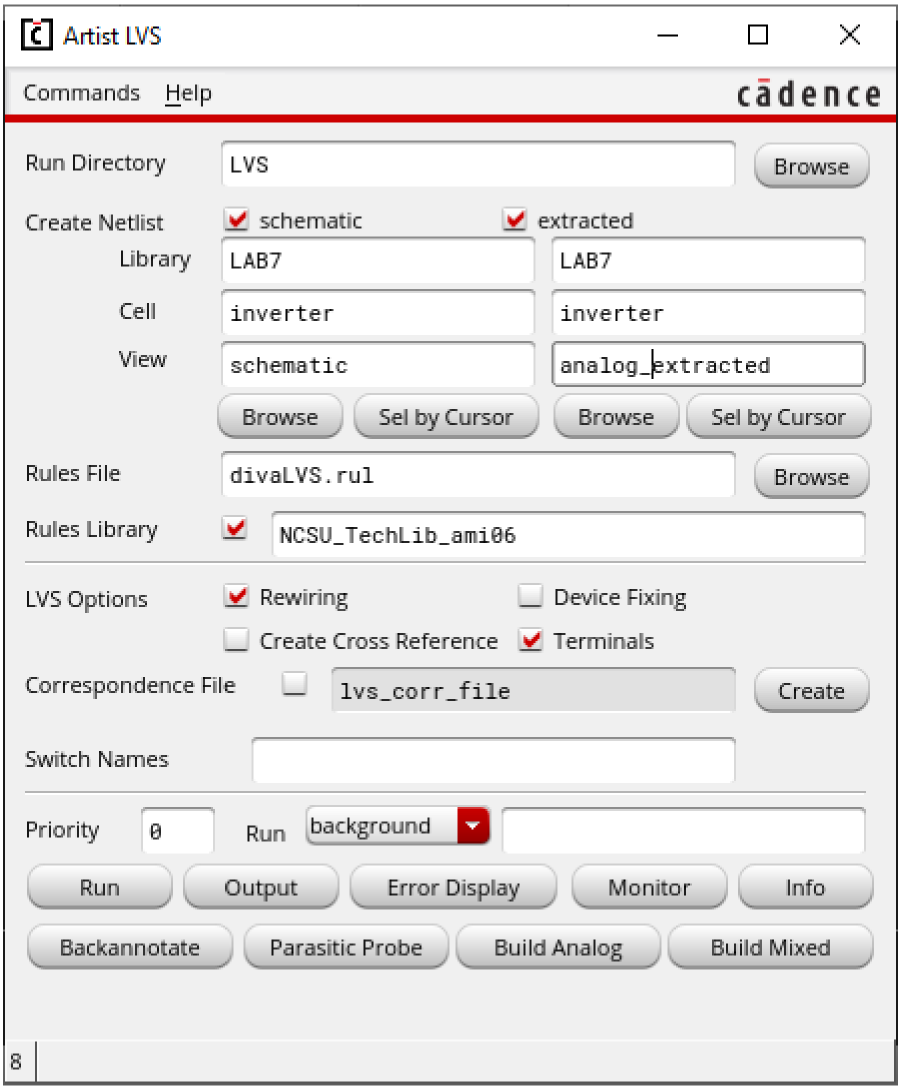
The TBinverter is the schematic view (called cmos_sch) of the test bench that includes other schematics and symbols.



- Parameterize cap to CL.
- Set vdc to 5V DC.
- Set vpulse Period=T.
- Delay = 1ps, Rise time = 0.1*T, Fall time = 0.1*T, Pulse width = 0.4*T.
- Voltage 1 = 0V, Voltage 2 = 5V.

Rise and fall times are 10% of the total period; 80% is the upper and lower part of the wave, each 40% of the period.

Double-clicking the layout view of the inverter cell, we open it and go to Verify -> LVS again. Entering the parameters: Left side will still be the schematic view, but we will set the right side to analog_extracted. This time, we will click Build Analog.



Artist LVS

Run Directory: LVS

Create Netlist: ☑ schematic ☑ extracted

	schematic	extracted
Library	LAB7	LAB7
Cell	inverter	inverter
View	schematic	analog_extracted

Rules File: divaLVS.rul

Rules Library: ☑ NCSU_TechLib_ami06

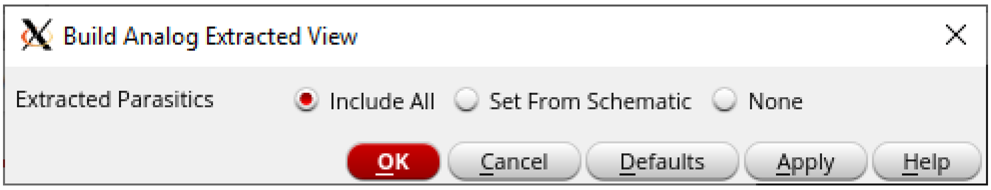
LVS Options: ☑ Rewiring ☐ Device Fixing ☐ Create Cross Reference ☑ Terminals

Correspondence File: ☐ lvs_corr_file

Switch Names:

Priority: 0 Run: background

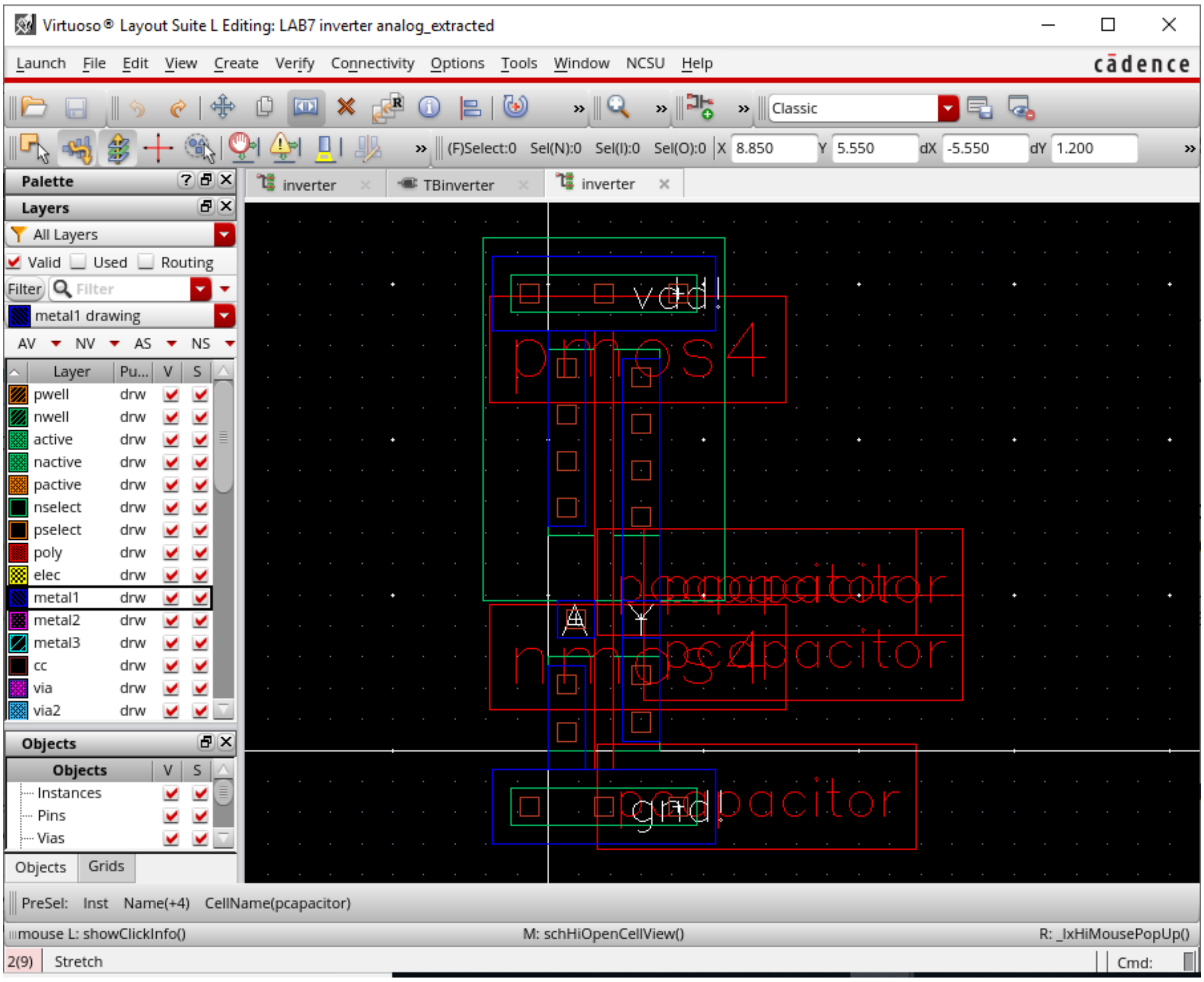
We select Include All for parasitics



Build Analog Extracted View

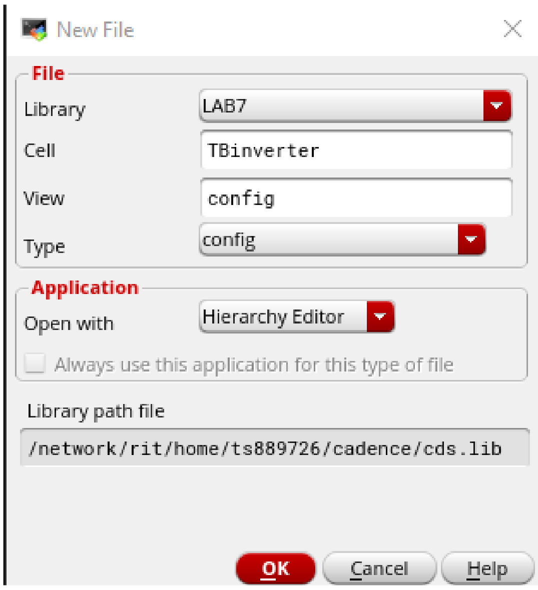
Extracted Parasitics: ● Include All ○ Set From Schematic ○ None

Analog_extracted view is very similar to the extracted view.


Virtuoso® Layout Suite L Editing: LAB7 inverter analog_extracted
Launch File Edit View Create Verify Connectivity Options Tools Window NCSU Help
cadence
Palette
Layers
All Layers
Valid Used Routing
metal1 drawing
pwell nwell active nactive pactive nselect pselect poly elec metal1 metal2 metal3 cc via via2
Objects
Instances Pins Vias
inverter TBinverter inverter
vdd!
pmos4
pcapacitor
A
Y
nmos4
gnd!
PreSel: Inst Name(+4) CellName(pcapacitor)
mouse L: showClickInfo()
M: schHiOpenCellView()
R: _lxHiMousePopUp()
Stretch
Cmd:

The inverter now has 5 different views: analog_extracted, extracted, layout, schematic, and symbol. Close out of the layout view and analog_extracted view.

In Library Manager, TBinverter cell is selected and we create new config view by File -> New -> Cell View. Choose config for View. It will open the config view of TBinverter with the Hierarchy Editor.



New File

File

Library: LAB7

Cell: TBinverter

View: config

Type: config

Application

Open with: Hierarchy Editor

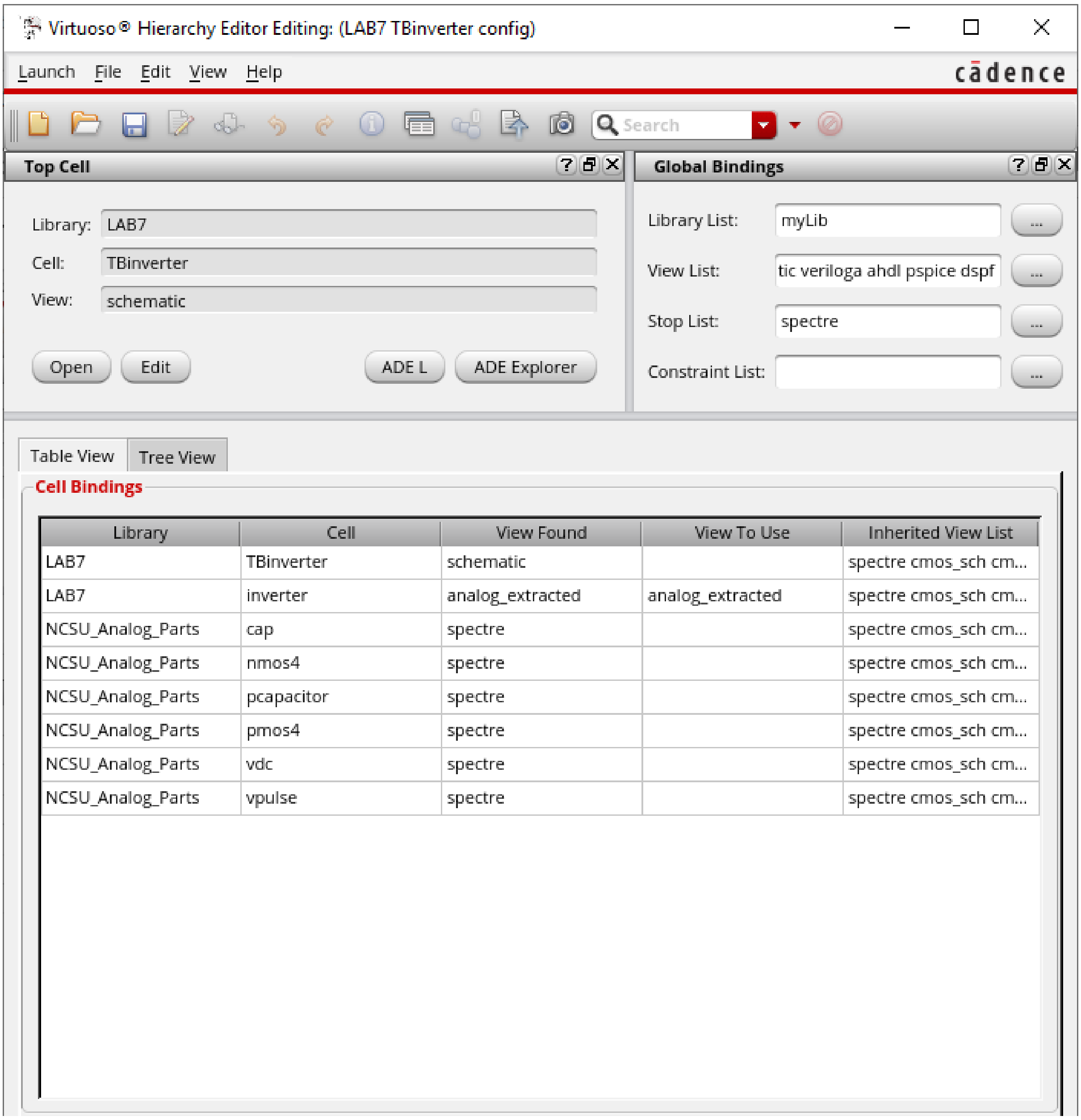
□ Always use this application for this type of file

Library path file

/network/rit/home/ts889726/cadence/cds.lib

When in Hierarchy Editor, we click Use Template. In the Use Template window, Name = spectre is chosen.

In the New Configuration window, View is changed to schematic.



Library: LAB7

Cell: TBinverter

View: schematic

Library List: myLib

View List: tic verilogа ahdl pspice dspf

Stop List: spectre

Constraint List:

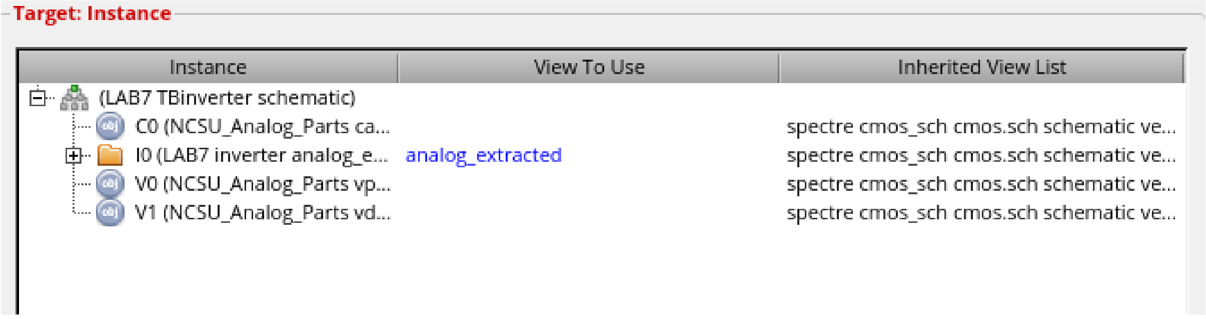
Library	Cell	View Found	View To Use	Inherited View List
LAB7	TBinverter	schematic		spectre cmos_sch cm...
LAB7	inverter	analog_extracted	analog_extracted	spectre cmos_sch cm...
NCSU_Analog_Parts	cap	spectre		spectre cmos_sch cm...
NCSU_Analog_Parts	nmos4	spectre		spectre cmos_sch cm...
NCSU_Analog_Parts	pcapacitor	spectre		spectre cmos_sch cm...
NCSU_Analog_Parts	pmos4	spectre		spectre cmos_sch cm...
NCSU_Analog_Parts	vdc	spectre		spectre cmos_sch cm...
NCSU_Analog_Parts	vpulse	spectre		spectre cmos_sch cm...

TBinverter is the root of the tree, which has a schematic view

- It has an instantiation of our inverter cell, which has a schematic view
- It has instantiations of cap, nmos4, pmos4, vdc, and vpulse cells with a spectre views.

View -> Tree is selected.

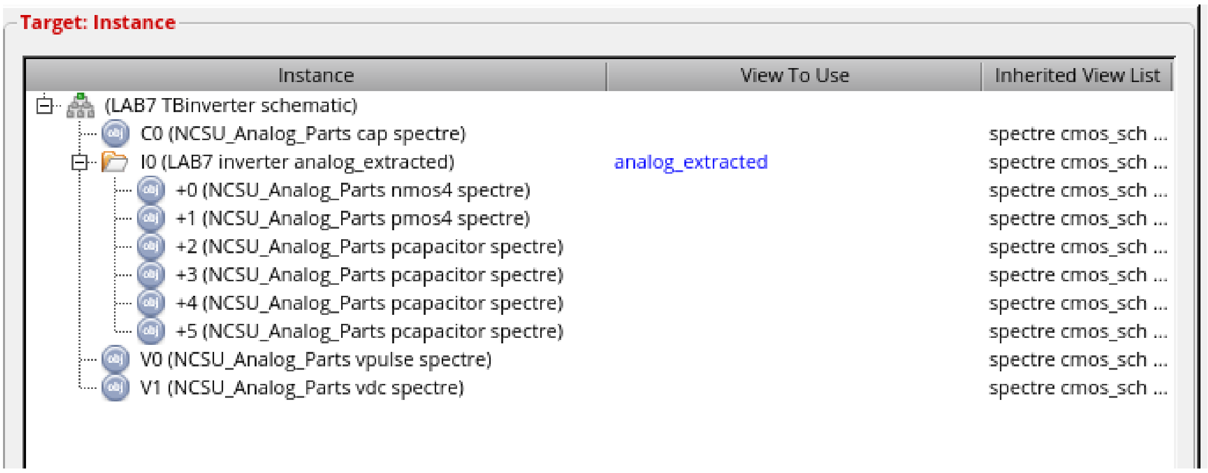
In the Tree View, right clicking the inverter cell in the tree Instance View = analog_extracted is set.



Target: Instance

Instance	View To Use	Inherited View List
(LAB7 TBinverter schematic)		
C0 (NCSU_Analog_Parts ca...		spectre cmos_sch cmos.sch schematic ve...
I0 (LAB7 inverter analog_e...	analog_extracted	spectre cmos_sch cmos.sch schematic ve...
V0 (NCSU_Analog_Parts vp...		spectre cmos_sch cmos.sch schematic ve...
V1 (NCSU_Analog_Parts vd...		spectre cmos_sch cmos.sch schematic ve...

Instead of the schematic view, using the hierarchy editor we use the analog_extracted view of the inverter cell, which we know has 4 pcapacitor cells, in addition to the nmos and pmos cells.

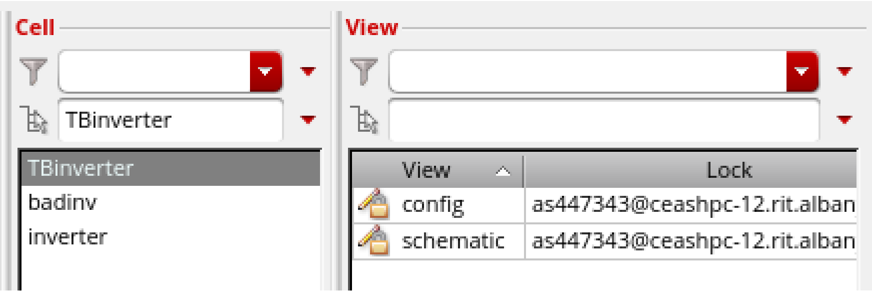


Target: Instance

Instance	View To Use	Inherited View List
(LAB7 TBinverter schematic)		
C0 (NCSU_Analog_Parts cap spectre)		spectre cmos_sch ...
I0 (LAB7 inverter analog_extracted)	analog_extracted	spectre cmos_sch ...
+0 (NCSU_Analog_Parts nmos4 spectre)		spectre cmos_sch ...
+1 (NCSU_Analog_Parts pmos4 spectre)		spectre cmos_sch ...
+2 (NCSU_Analog_Parts pcapacitor spectre)		spectre cmos_sch ...
+3 (NCSU_Analog_Parts pcapacitor spectre)		spectre cmos_sch ...
+4 (NCSU_Analog_Parts pcapacitor spectre)		spectre cmos_sch ...
+5 (NCSU_Analog_Parts pcapacitor spectre)		spectre cmos_sch ...
V0 (NCSU_Analog_Parts vpulse spectre)		spectre cmos_sch ...
V1 (NCSU_Analog_Parts vdc spectre)		spectre cmos_sch ...

The Hierarchy editor is saved and closed. We now have two different views of the TBinverter:

- schematic view is the good-and-old schematic of it with symbols and other parts in it
- config view is the one that allows to substitute different views of different cells.



Cell

TBinverter

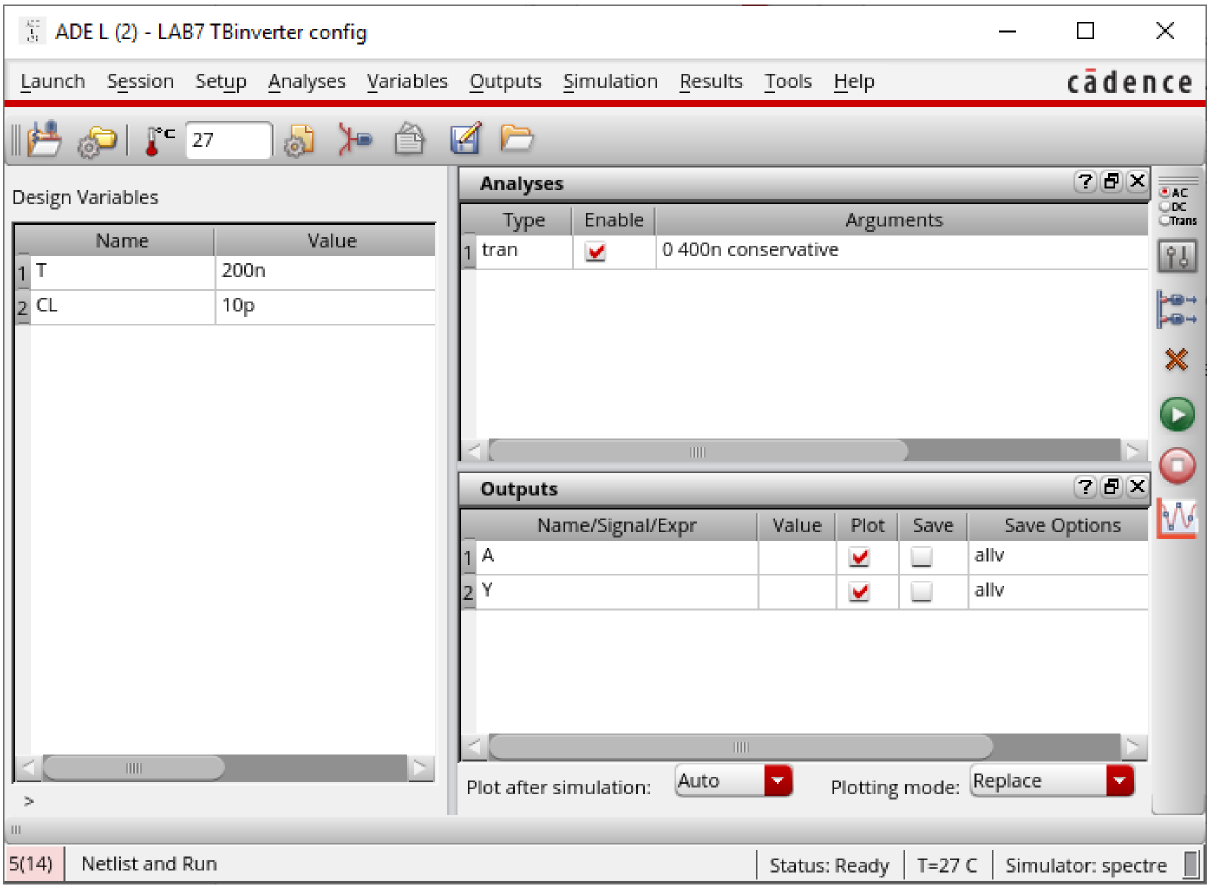
TBinverter
badinv
inverter

View

View	Lock
config	as447343@ceashpc-12.rit.alban
schematic	as447343@ceashpc-12.rit.alban

We launch ADE L and do transient analysis on the TBinverter test bench.

- We choose spectre as the simulator. Enter your transistor models, ami06P.m and ami06N.m.
- We choose Analyses -> Tran, Stop Time= 400n, conservative.
- Outputs -> To Be Plotted . Select A and Y. Enter CL=10p T=200n. Netlist and Run.



ADE L (2) - LAB7 TBinverter config

27

Design Variables

	Name	Value
1	T	200n
2	CL	10p

Analyses

	Type	Enable	Arguments
1	tran	☑	0 400n conservative

Outputs

	Name/Signal/Expr	Value	Plot	Save	Save Options
1	A		☑	□	allv
2	Y		☑	□	allv

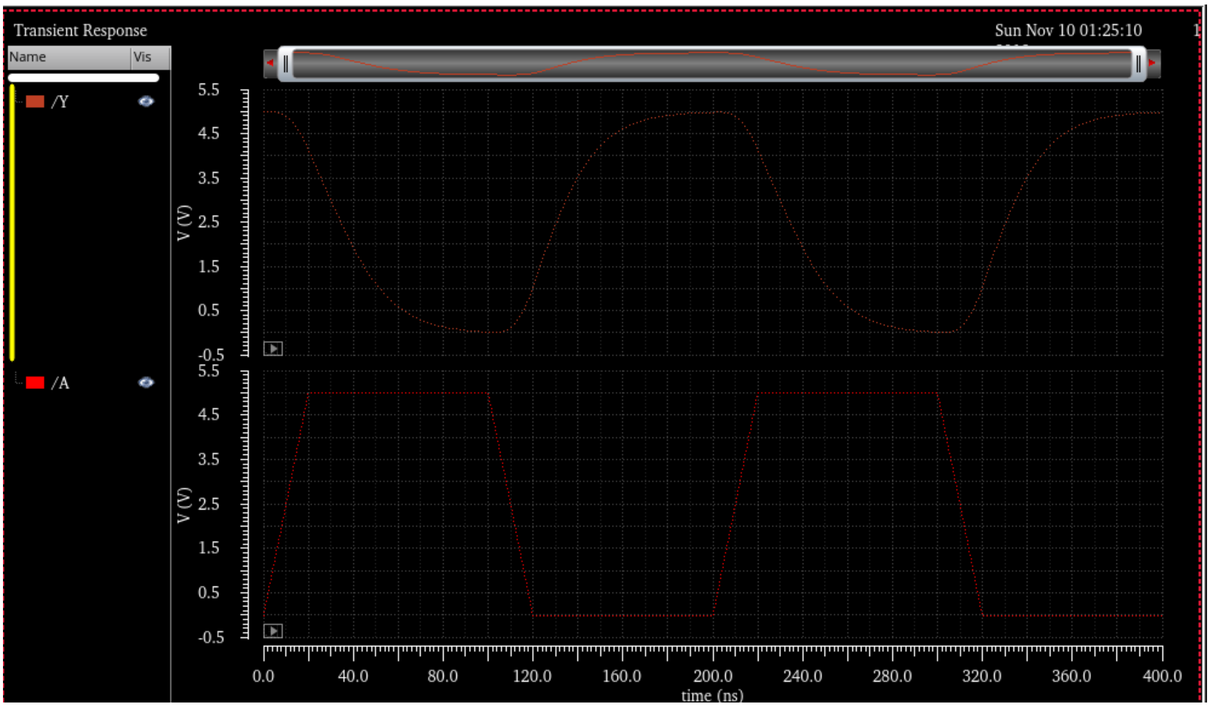
Plot after simulation: Auto

Plotting mode: Replace

5(14) Netlist and Run

Status: Ready T=27 C Simulator: spectre

The parasitic capacitances are very small and we notice no visible effect on the waveforms.


Transient Response
Sun Nov 10 01:25:10
Name
Vis
/Y
/A
V (V)
time (ns)