Lab 1: Schematic and Layout of a NAND gate

In lab 1, our objective is to:

- Get familiar with Cadence environment.
- Draw a schematic of a simple NAND gate and simulate it.
- Draw layout of a NAND gate using cell library, design rule check (DRC), extract, layout versus schematic (LVS) and simulate using extracted version.
- Compare these two simulations.

In this lab procedure, detailed procedures and snapshots are given for the shake of understanding. You are supposed to understand the procedures, so that you can design your own circuit later. Following lab procedures will not be elaborated so much except where necessary. Please refer to this lab procedure in future labs as reference.

Login procedure:

After logging in to the computer in the lab using your existing DoE accounts, double click on "**Sun Systems**" icon to start the remote login server on unix machine. Choose one of the computers from the list and click "**Connec**t". If you are unable to connect, click "Setting" and make sure all the checkmarks are ON. Ask TAs for help if you still have trouble. When you are prompted to enter your username and password, ask TAs for username and initial password. Make sure you **change your password** as soon as you log in for the first time and **remember** it for the rest of the labs. *TAs might not be able to help you if you forget your password.*

PART A: Procedure

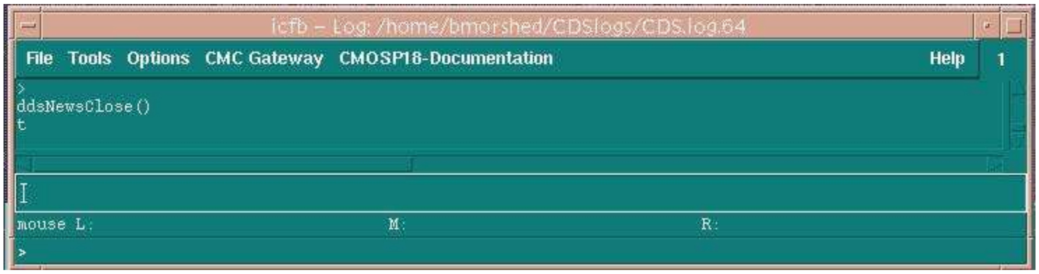
1) Open a console and create a folder in your home directory for Lab 1 "**ELEC4708/lab1**"

```
>mkdir ELEC4708
>cd ELEC4708
>mkdir lab1
>cd lab1
```

2) **Start Cadence** environment with 0.18 technology file.

```
>startCds –t cmosp18 –b icfb
```

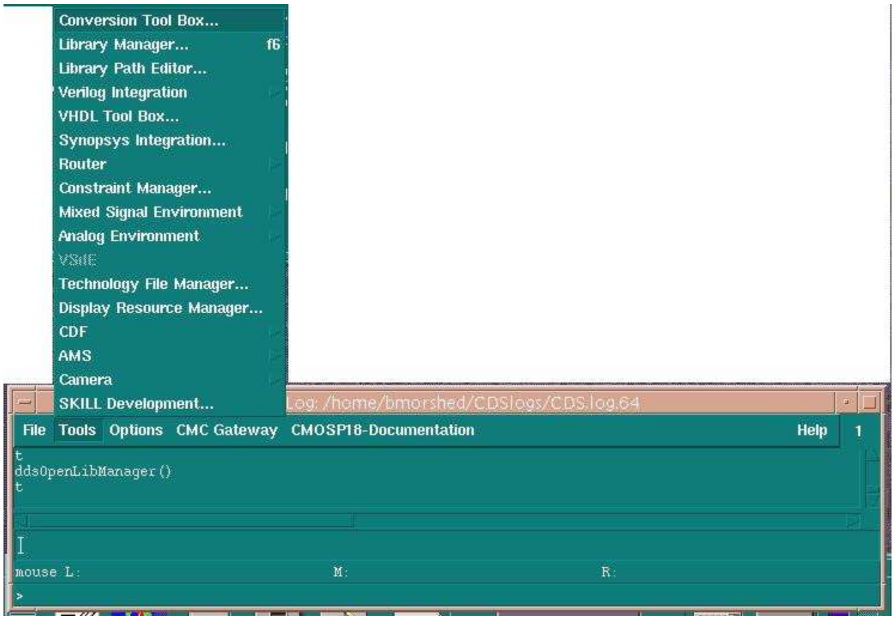
3) Close the "What's New" window. You will have "**icfb**" window open.



```
>
ddsNewsClose()
t
```

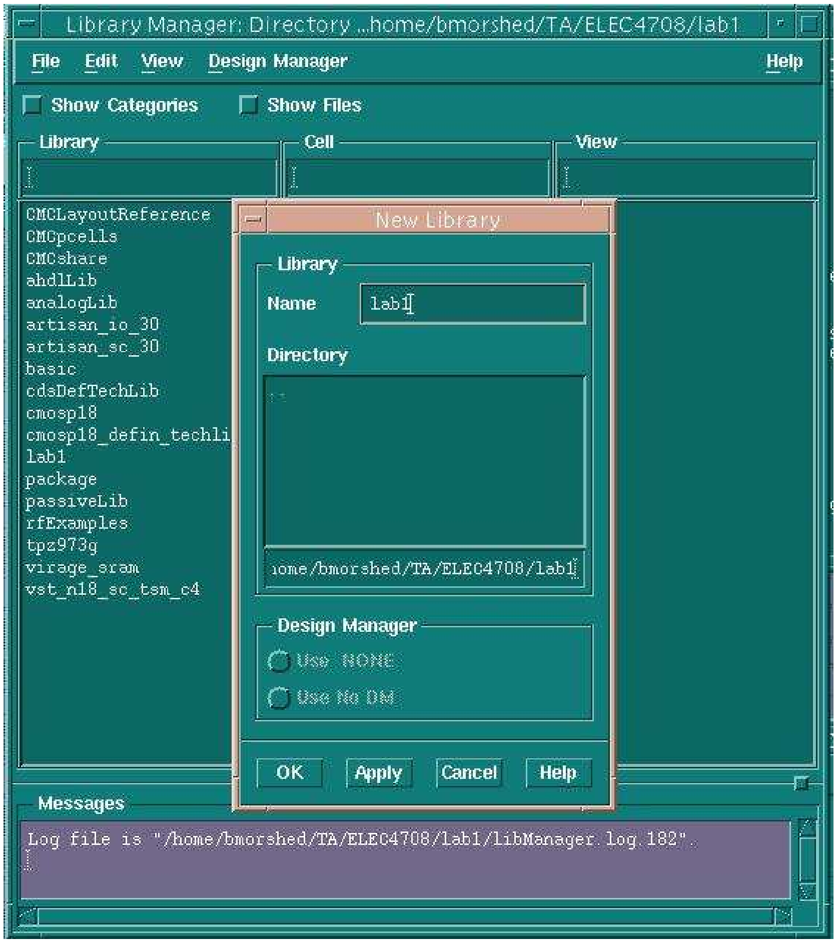
```
mouse L:                    M:                    R:
>
```

4) The next thing we will do is make a library to hold all your work for lab1. Click on Tools -> Library Manager (or simply press F6). Library manager window should pop up.



```
t
ddsOpenLibManager()
t
```

5) Click File -> New -> Library. Type “lab1” in Name field. Click OK.



□ Show Categories □ Show Files

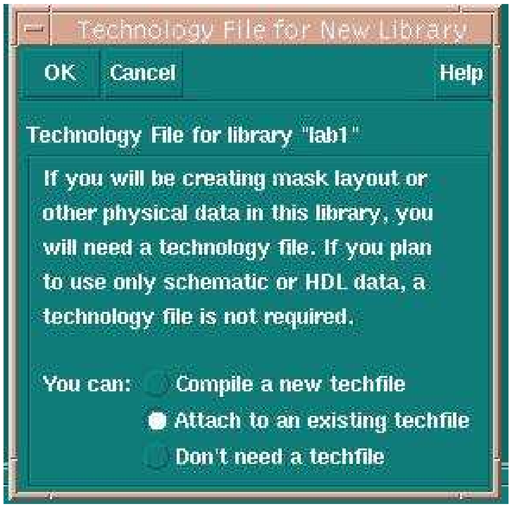
```
CMCLayoutReference
CMCpcells
CMCshare
ahdlLib
analogLib
artisan_io_30
artisan_sc_30
basic
cdsDefTechLib
cmosp18
cmosp18_defin_techli
lab1
package
passiveLib
rfExamples
tpz973g
virage_sram
vst_n18_sc_tsm_c4
```

Name: lab1

Directory: ıome/bmorshed/TA/ELEC4708/lab1

```
Log file is "/home/bmorshed/TA/ELEC4708/lab1/libManager.log.182".
```

6) Select “Attach to an existing techfile”. Click OK.

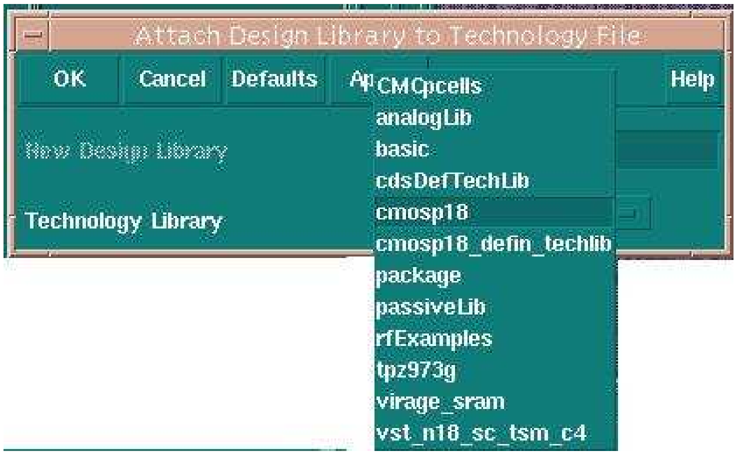


Technology File for library "lab1"

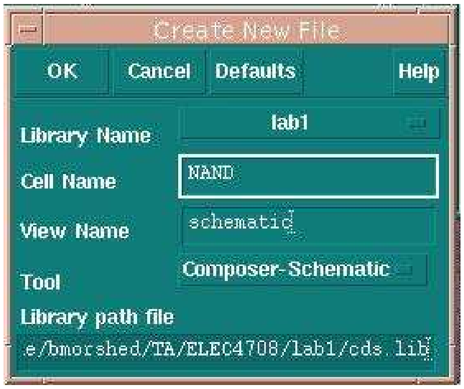
If you will be creating mask layout or other physical data in this library, you will need a technology file. If you plan to use only schematic or HDL data, a technology file is not required.

You can: ○ Compile a new techfile
● Attach to an existing techfile
○ Don't need a techfile

7) Select “**cmosp18**” from the drop down menu of the Technology Library. Click OK.



8) Highlight the new library you have just created, i.e. “lab1”, and click File -> New -> Cell View. Type “**NAND**” in Cell Name. Keep the default values in other fields as shown. Click OK.



Library Name lab1

Cell Name NAND

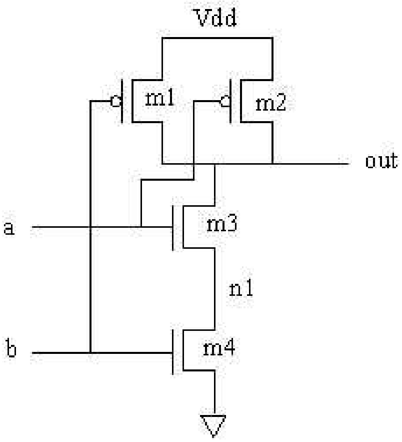
View Name schematic

Tool Composer-Schematic

Library path file
e/bmorshed/TA/ELEC4708/lab1/cds.lib

9) A blank **Virtuoso Schemetic Editing** window will open. Move your cursor through the icons on the left side and pop-up descriptions for each will show up. The next thing we

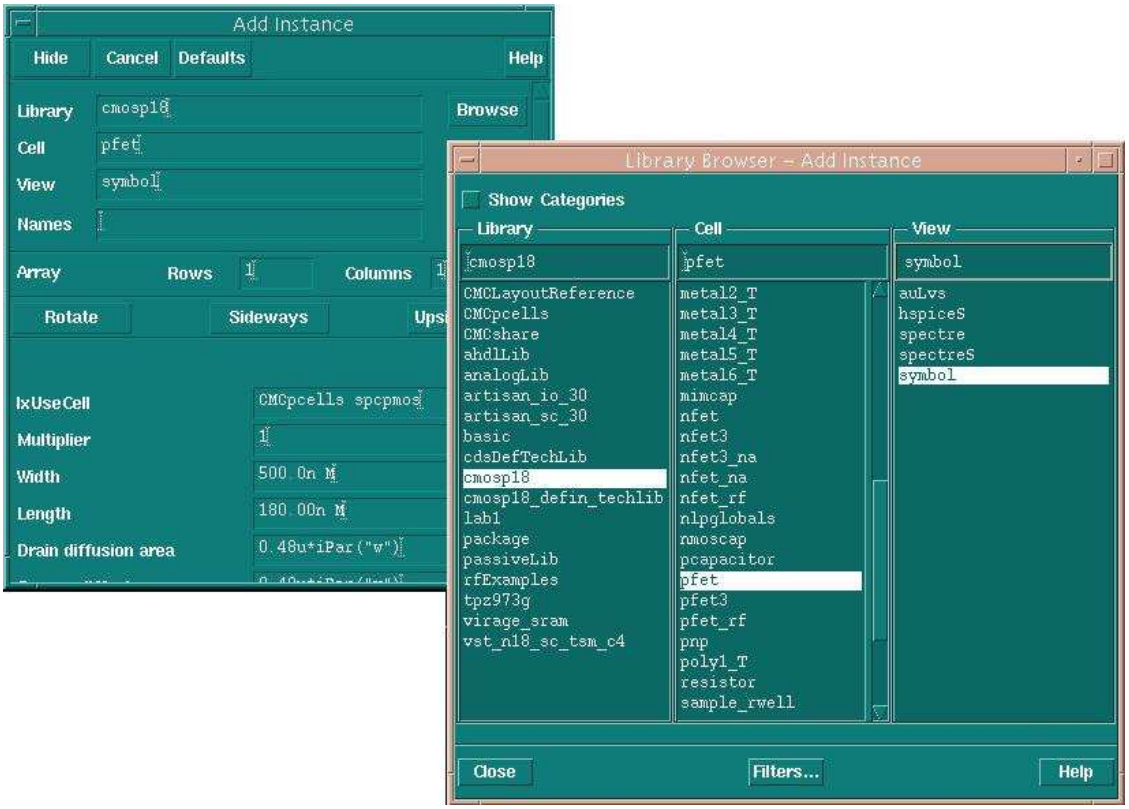
will do is draw the NAND gate using pfet and nfet. We will also add 2 input pins, 1 output pin, 1 VDD pin and 1 GND pin. A circuit diagram of NAND gate is given here.



10) To **add an instance** in your schematic, you can click on Instance icon, or click on Add -> Instance, or simply type "**i**" from the keyboard. Add Instance dialog box will show up.

11) Click Browse beside Library. Library Manager will pop up.

12) In Library, select "**cmosp18**", in cell select "**pfet**", in view select "**symbol**". Then click Close.



Library cmosp18
Cell pfet
View symbol
Names
Array Rows 1 Columns 1
lxUseCell CMCpcells spcpmos
Multiplier 1
Width 500.0n M
Length 180.00n M
Drain diffusion area 0.48u*iPar("w")

□ Show Categories

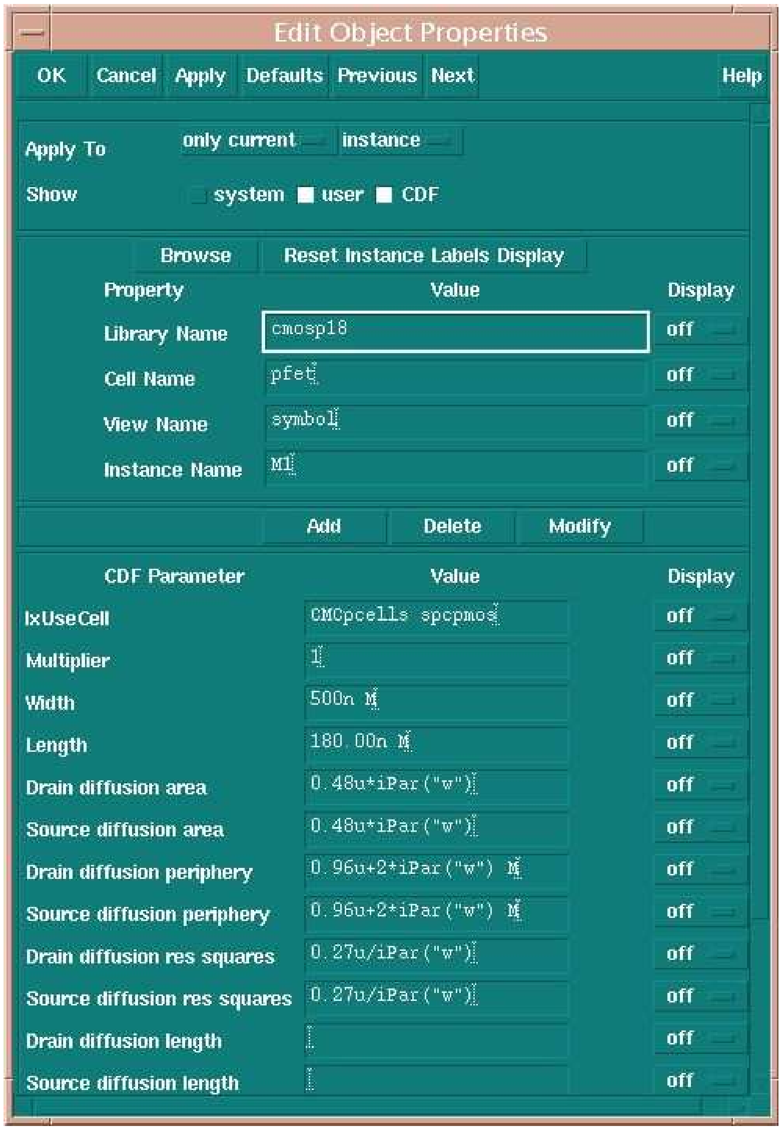
Library: cmosp18
CMCLayoutReference
CMCpcells
CMCshare
ahdlLib
analogLib
artisan_io_30
artisan_sc_30
basic
cdsDefTechLib
cmosp18
cmosp18_defin_techlib
lab1
package
passiveLib
rfExamples
tpz973g
virage_sram
vst_n18_sc_tsm_c4

Cell: pfet
metal2_T
metal3_T
metal4_T
metal5_T
metal6_T
mimcap
nfet
nfet3
nfet3_na
nfet_na
nfet_rf
nlpglobals
nmoscap
pcapacitor
pfet
pfet3
pfet_rf
pnp
poly1_T
resistor
sample_rwell

View: symbol
auLvs
hspiceS
spectre
spectreS
symbol

13) Click Hide and place the instance in your design. You can place multiple instances of same item. When you are done with placing all the instances of that item, press Esc to get rid of it.

14) You can access the **object property** window by selecting a pfet and pressing “**q**”. Edit Object Properties window will pop up and will show properties of that instance. You can change length or width if required. For now, we will use defaults widths for pfets and nfets. However, you can change widths later to enhance performance. Note that the default length value is set to the minimum allowed by the technology.



Apply To: only current instance

Show: system ■ user ■ CDF

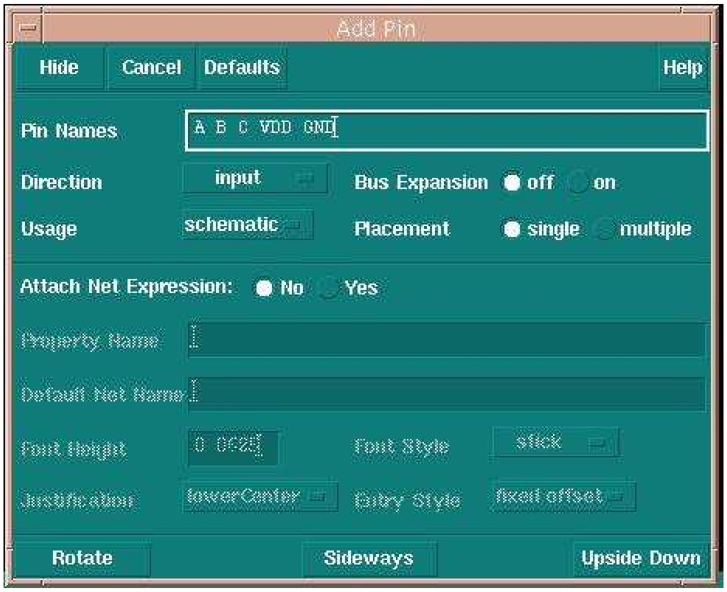
Property	Value	Display
Library Name	cmosp18	off
Cell Name	pfet	off
View Name	symbol	off
Instance Name	M1	off

CDF Parameter	Value	Display
lxUseCell	CMCpcells spcpmos	off
Multiplier	1	off
Width	500n M	off
Length	180.00n M	off
Drain diffusion area	0.48u*iPar("w")	off
Source diffusion area	0.48u*iPar("w")	off
Drain diffusion periphery	0.96u+2*iPar("w") M	off
Source diffusion periphery	0.96u+2*iPar("w") M	off
Drain diffusion res squares	0.27u/iPar("w")	off
Source diffusion res squares	0.27u/iPar("w")	off
Drain diffusion length		off
Source diffusion length		off

15) Repeat steps (10) to (14) for “**nfet**”s from the same library and place them in your design.

16) To **add pins**, click on Pin from tools button on the left side, or click Add -> Pin, or simply type “**p**” from the keyboard. Add Pin dialog box will open.

17) Type the names of the pins you want to add in sequence, leaving a blank space in between. DO NOT click Hide button. You will need to change Direction property for some of the pins.



Pin Names: A B C VDD GND

Direction: input

Bus Expansion: ● off ○ on

Usage: schematic

Placement: ● single ○ multiple

Attach Net Expression: ● No ○ Yes

Property Name:

Default Net Name:

Font Height: 0.0625

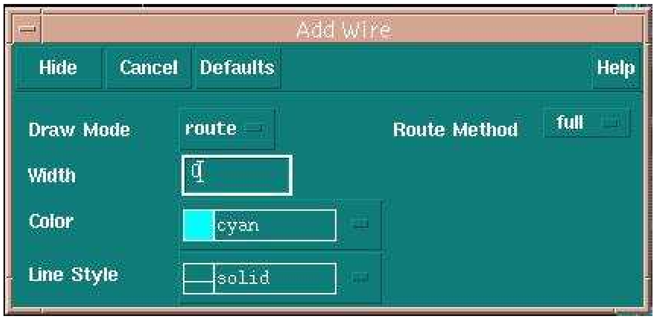
Font Style: stick

Justification: lowerCenter

Entry Style: fixed offset

18) Click on the schematic where you want to place pin A. For pin **A and B**, direction should be **input**, for pin **C**, direction should be **output**, for pin **VDD and GND**, direction should be **inputOutput**. Now click "Cancel" of the Add Pin dialog box.

19) To **add wires**, click on Wire (narrow) from tools button on the left side, or click Add -> Wire (narrow), or simply type "**w**" from the keyboard. Add Wire dialog box will open.



Draw Mode: route

Route Method: full

Width: 0

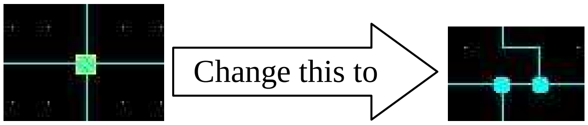
Color: cyan

Line Style: solid

20) Click hide, and connect the pins with FETs properly to form a NAND gate. Notice that you have to connect the substrate of pfets and nfets properly.

21) Click on "**Check and Save**", the first icon on the left hand side toolbar, or click Design -> Check and Save, or simply type "X" from the keyboard to save the file.

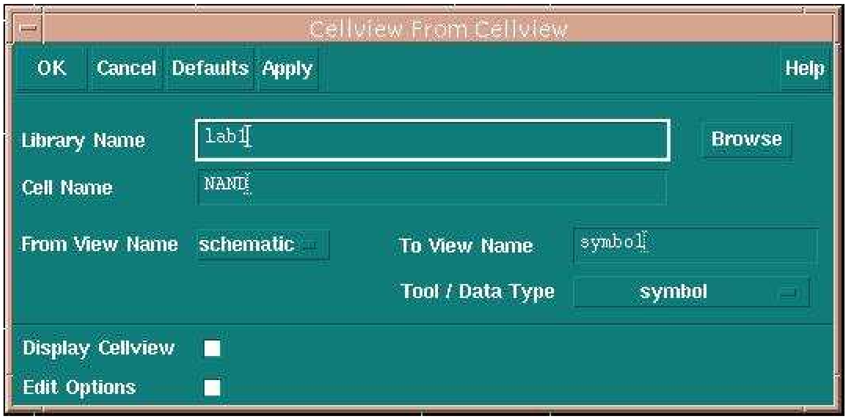
22) You might get errors and warnings at this point. Note that Virtuoso will produce warning if a junction has more than 3 wires connected. To avoid this, you can reroute one wire to form 2 junctions instead of 1.



23) Get rid of all errors and warnings. Ask TAs if you need help. Next thing we will do is create a symbol for our NAND gate.

24) Click **Design -> Create Cellview -> From Cellview**. "Cellview From Cellview" dialog box will open.

25) Keep the default values as shown and click OK.



Library Name: lab1

Cell Name: NAND

From View Name: schematic

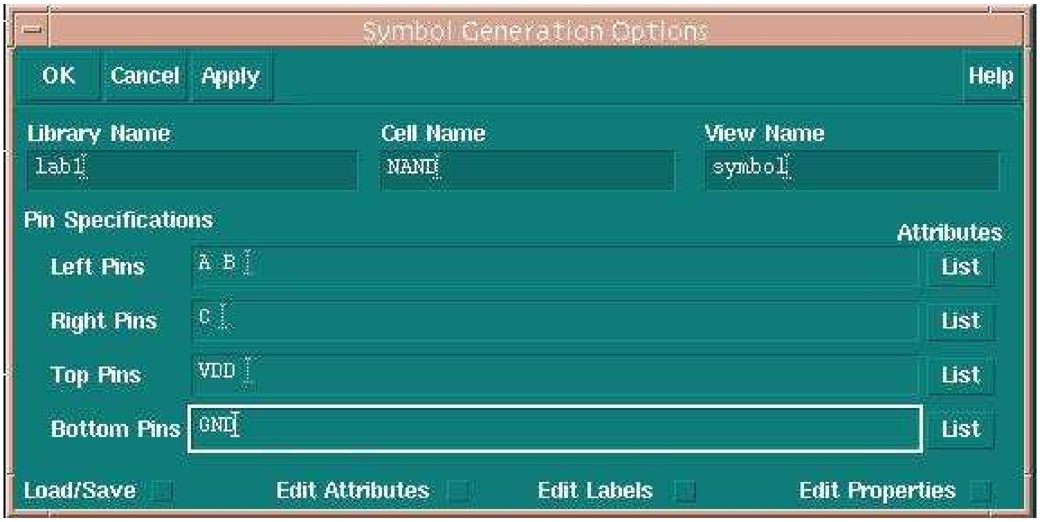
To View Name: symbol

Tool / Data Type: symbol

Display Cellview □

Edit Options □

26) Symbol Generation Options dialog box will open. Cut and paste GND from Top Pins field to Bottom Pins field. The dialog box should look like this. Click OK.



Library Name: lab1

Cell Name: NAND

View Name: symbol

Pin Specifications

Attributes

Left Pins: A B

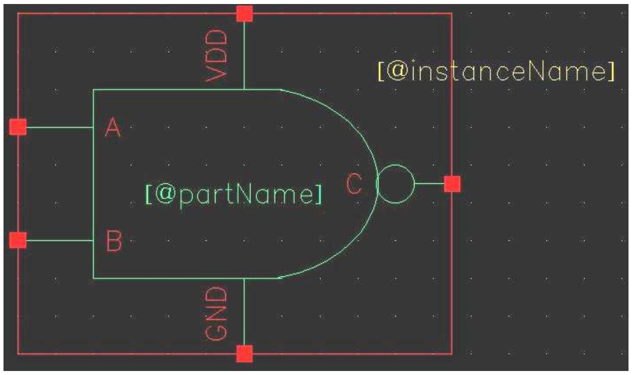
Right Pins: C

Top Pins: VDD

Bottom Pins: GND

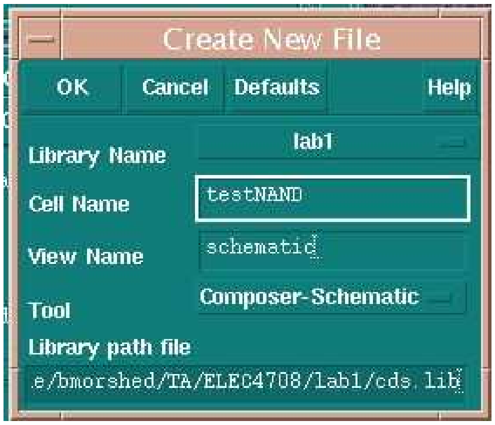
Load/Save □ Edit Attributes □ Edit Labels □ Edit Properties □

27) Virtuoso Symbol Editing window will open. You can keep the default rectangular shape or change it to your own as you like. To change shape, select and delete rectangular box, click on Line icon in the left side toolbox, select shape and line width as necessary and draw a new shape. You can also move the pins around if you like. This step is optional.



28) Save and close the symbol window and schematic window. You will also need to create an **inverter schematic and symbol** using the same procedure (step 8 to 27). Name your inverter as **inv0**. Next we will make a testbench to simulate our NAND gate.

29) With "icfb" window highlighted, press F6 to open the library manager. Select lab1 from the library field. Click File -> New -> Cell view. In the Create New File dialog box, enter **testNAND** in the Cell Name field. Keep other default values. Click OK.



Library Name lab1

Cell Name testNAND

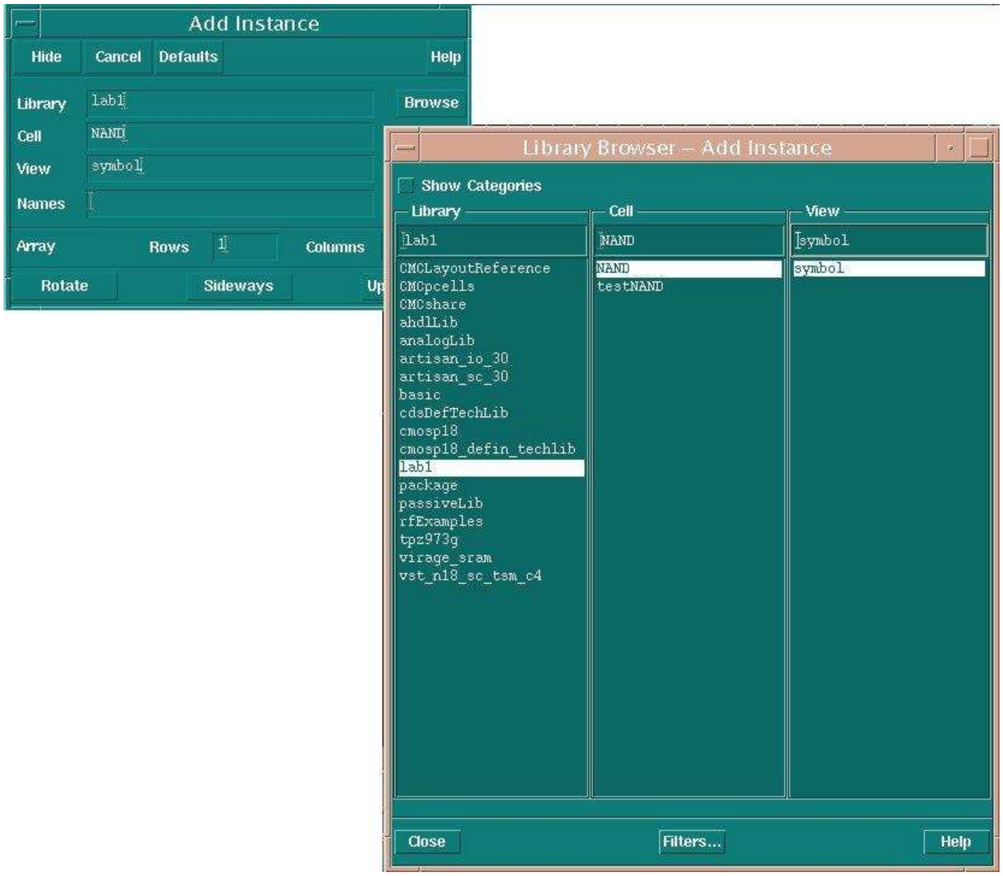
View Name schematic

Tool Composer-Schematic

Library path file

.e/bmorshed/TA/ELEC4708/lab1/cds.lib

30) A blank Virtuoso Schematic Editing window will open. Press "i" to Add Instance. Click Browse. From Library Browser, select lab1 from Library, NAND from Cell, symbol from view. (From now on, this selection sequence will be given as: **lab1 -> NAND -> symbol**) Click Close. Place the NAND gate in your blank Virtuoso window. Press Esc.



Library lab1

Cell NAND

View symbol

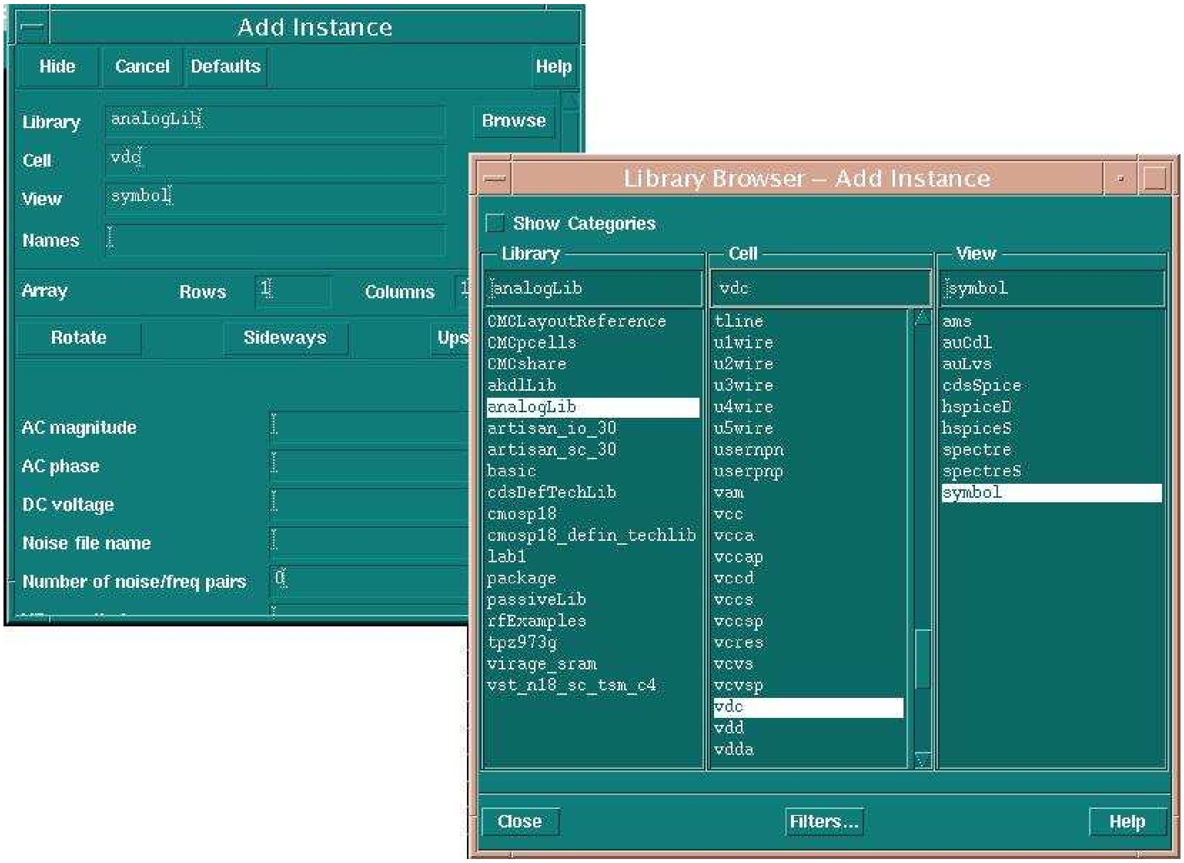
Names

Array Rows 1 Columns

□ Show Categories

Library	Cell	View
lab1	NAND	symbol
CMCLayoutReference	NAND	symbol
CMCpcells	testNAND	
CMCshare		
ahdlLib		
analogLib		
artisan_io_30		
artisan_sc_30		
basic		
cdsDefTechLib		
cmosp18		
cmosp18_defin_techlib		
lab1		
package		
passiveLib		
rfExamples		
tpz973g		
virage_sram		
vst_n18_sc_tsm_c4		

31) Press “i” again to Add Instance. Click Browse. From Library Browser, select **analogLib -> vdc -> symbol**. Click Close. Place the “vdc” in your design. Press Esc.



Library analogLib
Cell vdc
View symbol
Names
Array Rows 1 Columns 1
AC magnitude
AC phase
DC voltage
Noise file name
Number of noise/freq pairs 0

□ Show Categories

Library: analogLib
CMCLayoutReference, CMCpcells, CMCshare, ahdlLib, analogLib, artisan_io_30, artisan_sc_30, basic, cdsDefTechLib, cmosp18, cmosp18_defin_techlib, lab1, package, passiveLib, rfExamples, tpz973g, virage_sram, vst_n18_sc_tsm_c4

Cell: vdc
tline, u1wire, u2wire, u3wire, u4wire, u5wire, usernpn, userpnp, vam, vcc, vcca, vccap, vccd, vccs, vccsp, vcres, vcvs, vcvsp, vdc, vdd, vdda

View: symbol
ams, auCdl, auLvs, cdsSpice, hspiceD, hspiceS, spectre, spectreS, symbol

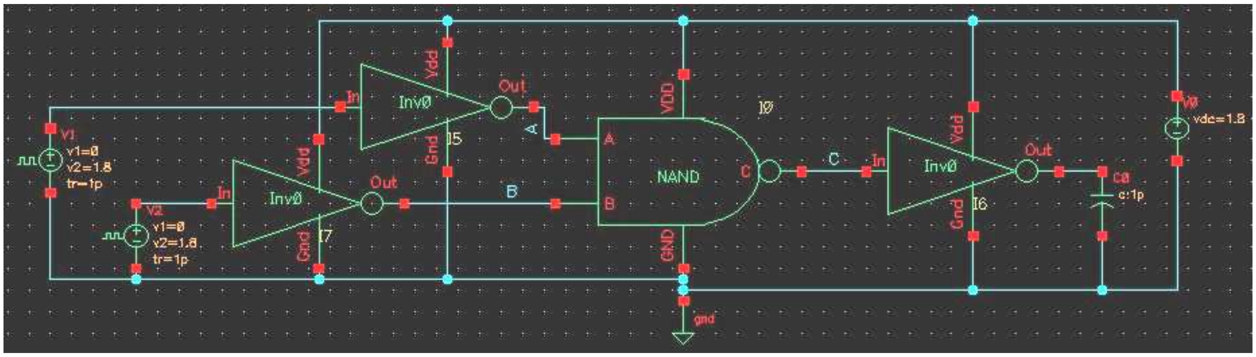
32) Press “i” again to Add Instance. Click Browse. From Library Browser, select **analogLib -> gnd -> symbol**. Click Close. Place the “gnd” in your design. Press Esc.

33) Press “i” again to Add Instance. Click Browse. From Library Browser, select **analogLib -> vpulse -> symbol**. They will serve as input signals for input A and B (through inverters). Click Close. Place 2 (two) “vpulse” in your design. Press Esc. Note that to complete this lab, you might need to use **vpwl** as input signals. Ask TAs if you need help.

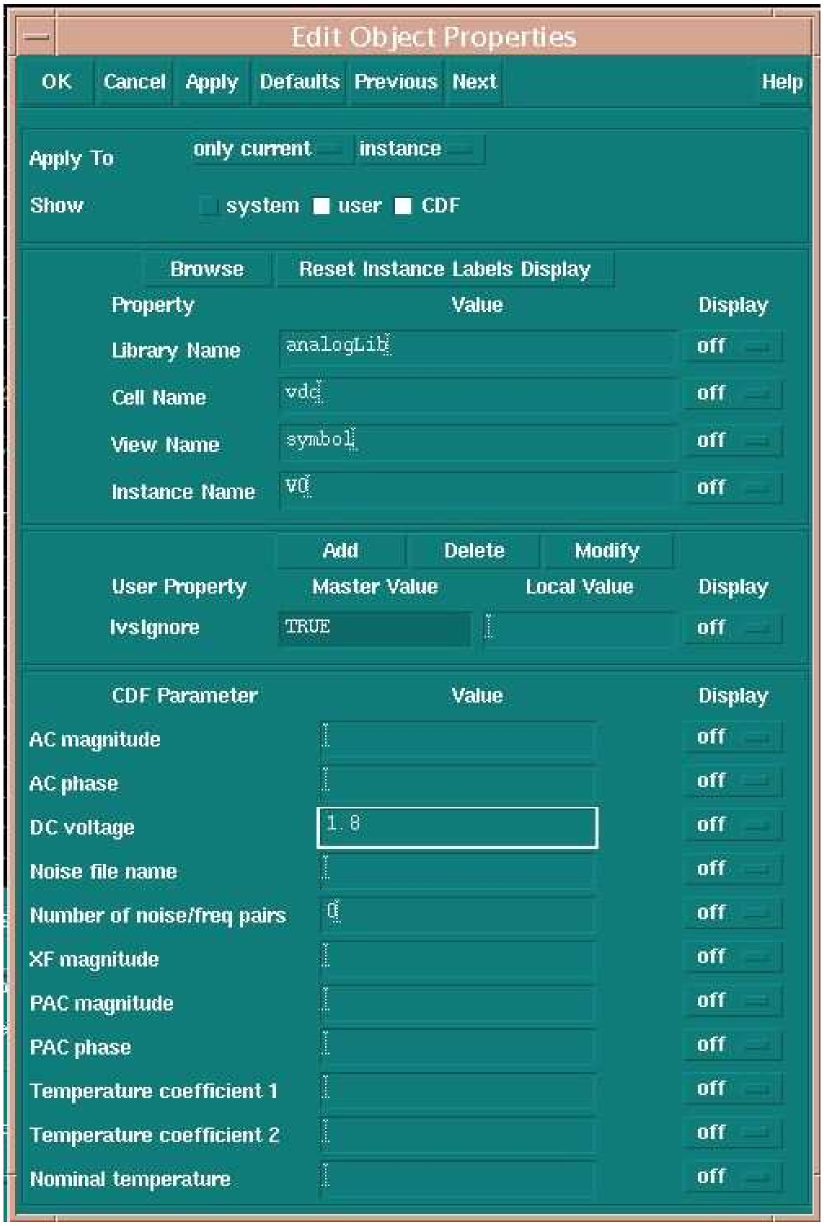
34) Press “i” again to Add Instance. Click Browse. From Library Browser, select **analogLib -> cap -> symbol**. This will serve as output load for output inverter. Click Close. Place the “cap” in your design. Press Esc.

35) Press “i” again to Add Instance. Click Browse. From Library Browser, select **lab1 -> Inv0 -> symbol**. Place three inverters in you design. Two of the inverters between vpulses (or vpwl) and input A and B. Third inverter between C and output capacitor.

36) Select wire (narrow) and connect all the symbols properly. If you need to **pan around** use the **cursor arrow keys**. Use “**[**“ or “**]**” to **zoom in or out** by factor of 2. Use “**z**” to **zoom to window**. Your design should be similar to this one.



37) Now we will setup the properties of some instances. Select V0 (vdc) and press “q” to access the Edit Object Properties dialog box. In the DC Voltage field, enter **1.8 V** and click OK. Note that standard supply voltage for CMOS 0.18u technology is 1.8V.



Apply To only current instance

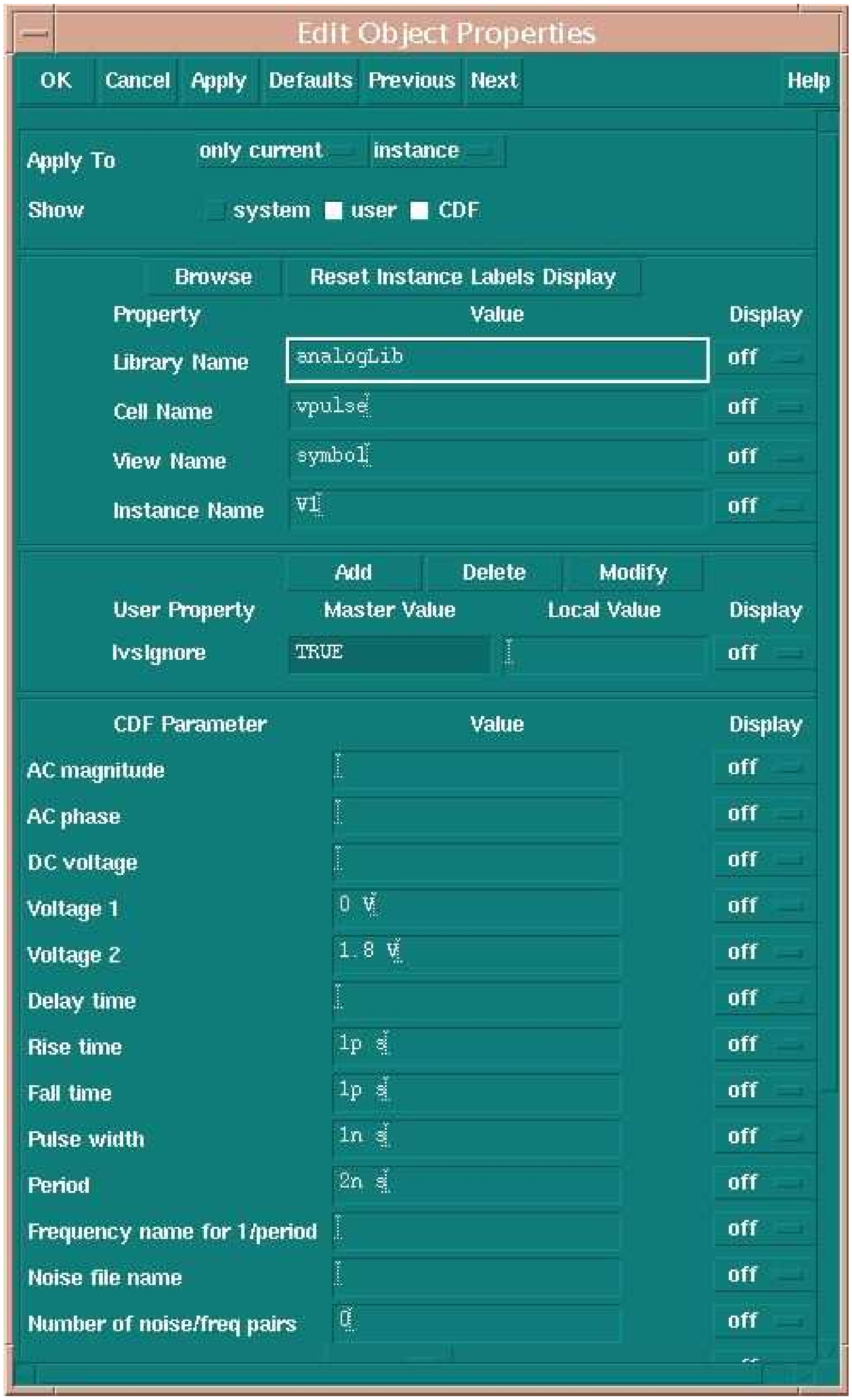
Show □ system ■ user ■ CDF

Property	Value	Display
Library Name	analogLib	off
Cell Name	vdc	off
View Name	symbol	off
Instance Name	V0	off

User Property	Master Value	Local Value	Display
lvsIgnore	TRUE		off

CDF Parameter	Value	Display
AC magnitude		off
AC phase		off
DC voltage	1.8	off
Noise file name		off
Number of noise/freq pairs	0	off
XF magnitude		off
PAC magnitude		off
PAC phase		off
Temperature coefficient 1		off
Temperature coefficient 2		off
Nominal temperature		off

38) Select V1 (vpulse) and press “q” to access the Edit Object Properties dialog box. In the Voltage 2 field, enter 1.8 V; Rise time field, enter 1p; Fall time field, enter 1p; Pulse width field, enter 1n; Period field, enter 2n and click OK. If you are using vpwl, you should enter time-voltage pairs for your desired pulse. *Note that you can also enter a variable or expression in any field and later assign a value to that variable during simulation. For example, you can enter “T” in Period field, “T/2” in Pulse width. Then during simulation, specify T=2n. Variables give you flexibility of sweeping values.*



Apply To: only current instance

Show: □ system ■ user ■ CDF

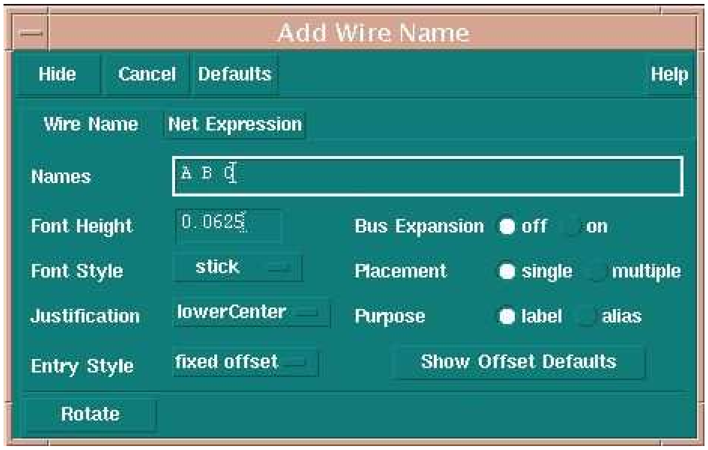
Property	Value	Display
Library Name	analogLib	off
Cell Name	vpulse	off
View Name	symbol	off
Instance Name	V1	off

User Property	Master Value	Local Value	Display
lvsIgnore	TRUE		off

CDF Parameter	Value	Display
AC magnitude		off
AC phase		off
DC voltage		off
Voltage 1	0 V	off
Voltage 2	1.8 V	off
Delay time		off
Rise time	1p s	off
Fall time	1p s	off
Pulse width	1n s	off
Period	2n s	off
Frequency name for 1/period		off
Noise file name		off
Number of noise/freq pairs	0	off

39) Select V2 (vpulse) and press “q” to access the Edit Object Properties dialog box. In the Voltage 2 field, enter 1.8 V; Rise time field, enter 1p; Fall time field, enter 1p; Pulse width field, enter 2n; Period field, enter 4n and click OK.

40) Click the icon name “**Wire Name**” on the left had side toolbar, or click Add -> Wire Name, or press “**l**” (small L). The Add Wire Name dialog box will open. Type “A B C” in the Names field. Click Hide. Click the wire connected to input pin **A** of the NAND gate first, then the wire connected to input pin **B**, then the wire connected to output pin **C**.



Names A B C

Font Height 0.0625

Bus Expansion ● off ○ on

Font Style stick

Placement ● single ○ multiple

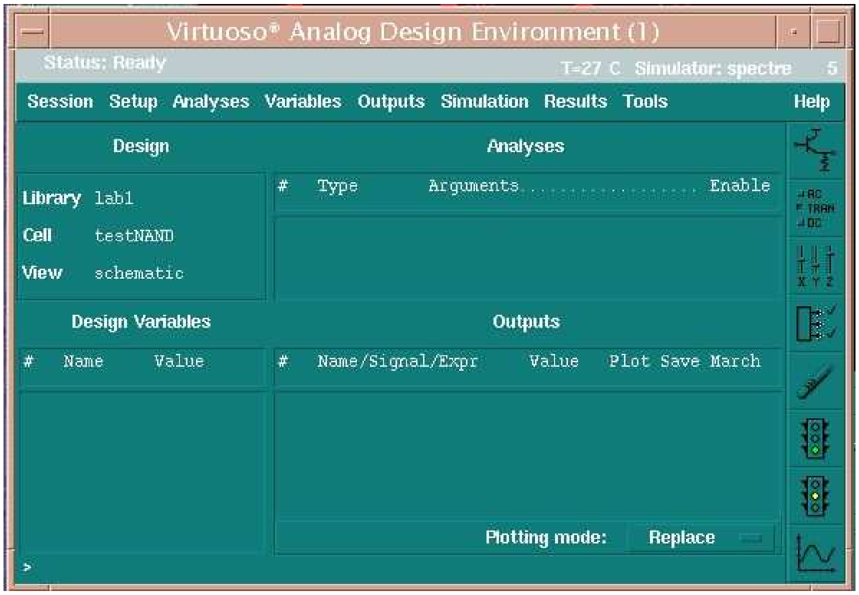
Justification lowerCenter

Purpose ● label ○ alias

Entry Style fixed offset

41) Look at the picture in step (36). Each component of your design should show similar values.

42) To simulate our design, we will use **Virtuoso Analog Design Environment**. Click **Tools -> Analog Environment** to access the dialog box for Virtuoso Analog Design Environment.



Status: Ready T=27 C Simulator: spectre 5

Library lab1

Cell testNAND

View schematic

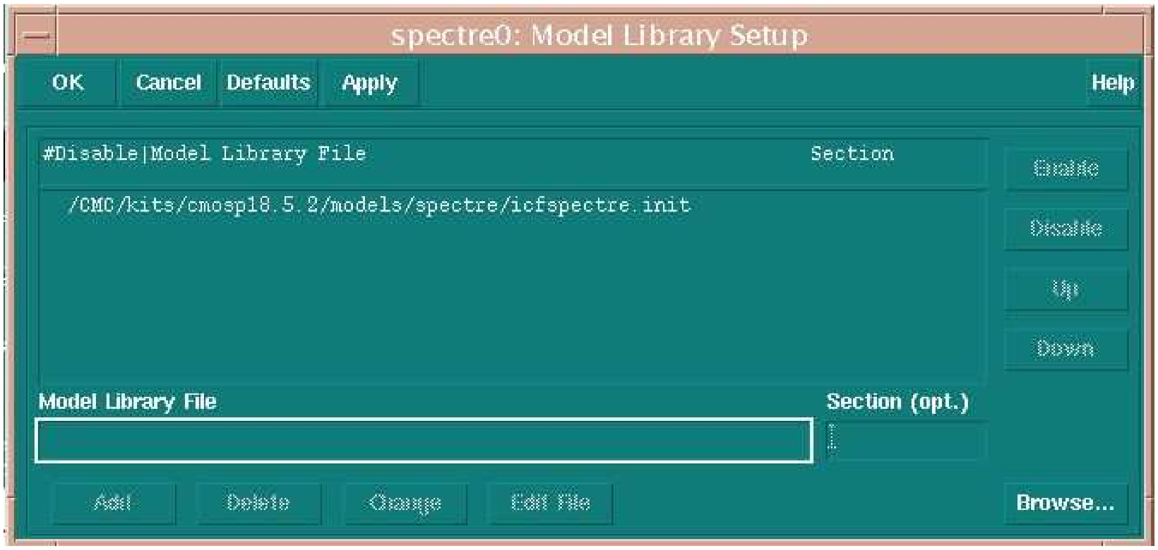
#	Type	Arguments	Enable

#	Name	Value

#	Name/Signal/Expr	Value	Plot	Save	March

Plotting mode: Replace

43) Click **Setup -> Model Libraries**. In the Model Library Setup window, Enter this line to Model Library File: **/CMC/kits/cmosp18.5.2/models/spectre/icfspectre.init**. Click OK. Then click **Add** to add this model in your dialog box. Click OK.

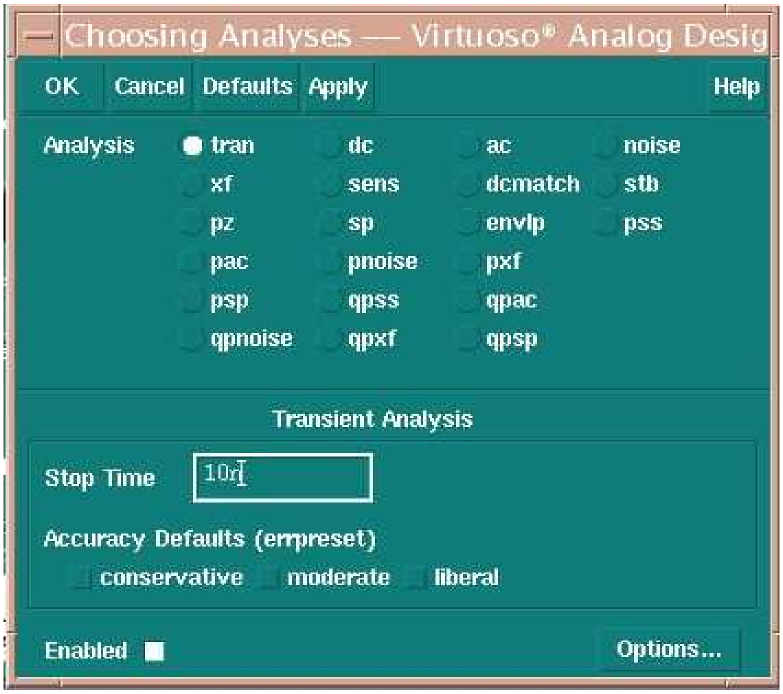

spectre0: Model Library Setup

```
#Disable|Model Library File                                Section
  /CMC/kits/cmosp18.5.2/models/spectre/icfspectre.init
```

Model Library File

Section (opt.)

44) Click **Analysis -> Choose**. Choosing Analysis dialog box will show up. Select Analysis type as **tran** and enter **10n** in Stop Time field and select **Enabled**. Click OK.


Choosing Analyses — Virtuoso® Analog Desig

Analysis ● tran

Transient Analysis

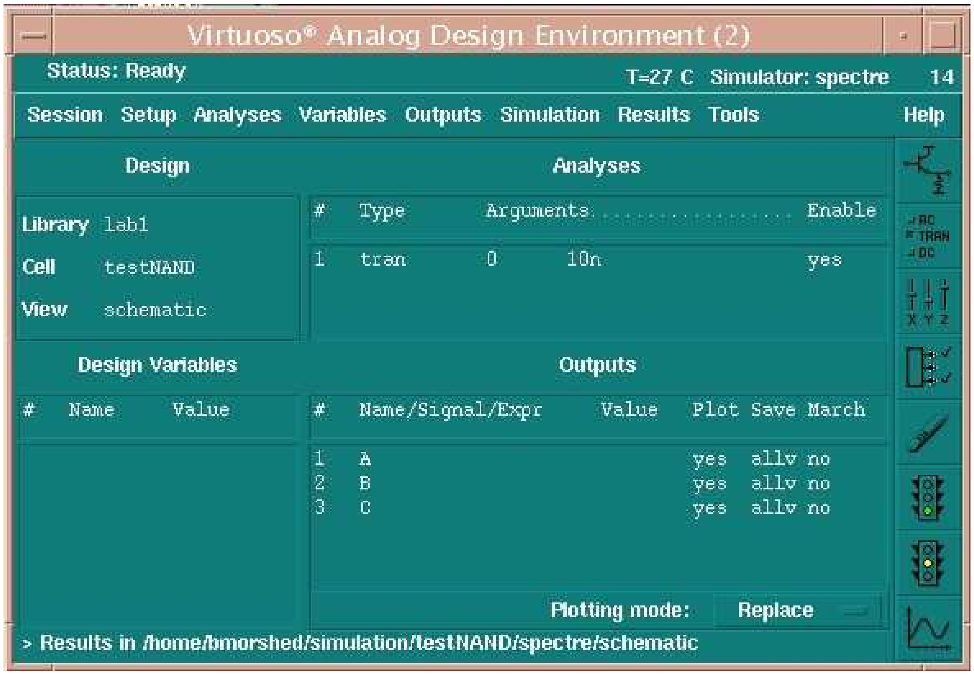
Stop Time 10n

Accuracy Defaults (errpreset)

□ conservative □ moderate □ liberal

Enabled ■

45) Click **Outputs -> To Be Plotted -> Select On Schematic**. Click the input wire connected to **A**, the input wire connected to **B** and the output wire connected to **C**. Now your Virtuoso Analog Design Environment should look like this.


Virtuoso® Analog Design Environment (2)
Session Setup Analyses Variables Outputs Simulation Results Tools Help
Plotting mode: Replace

Status: Ready T=27 C Simulator: spectre 14

Design

Library lab1
Cell testNAND
View schematic

Analyses

#	Type	Arguments		Enable
1	tran	0	10n	yes

Design Variables

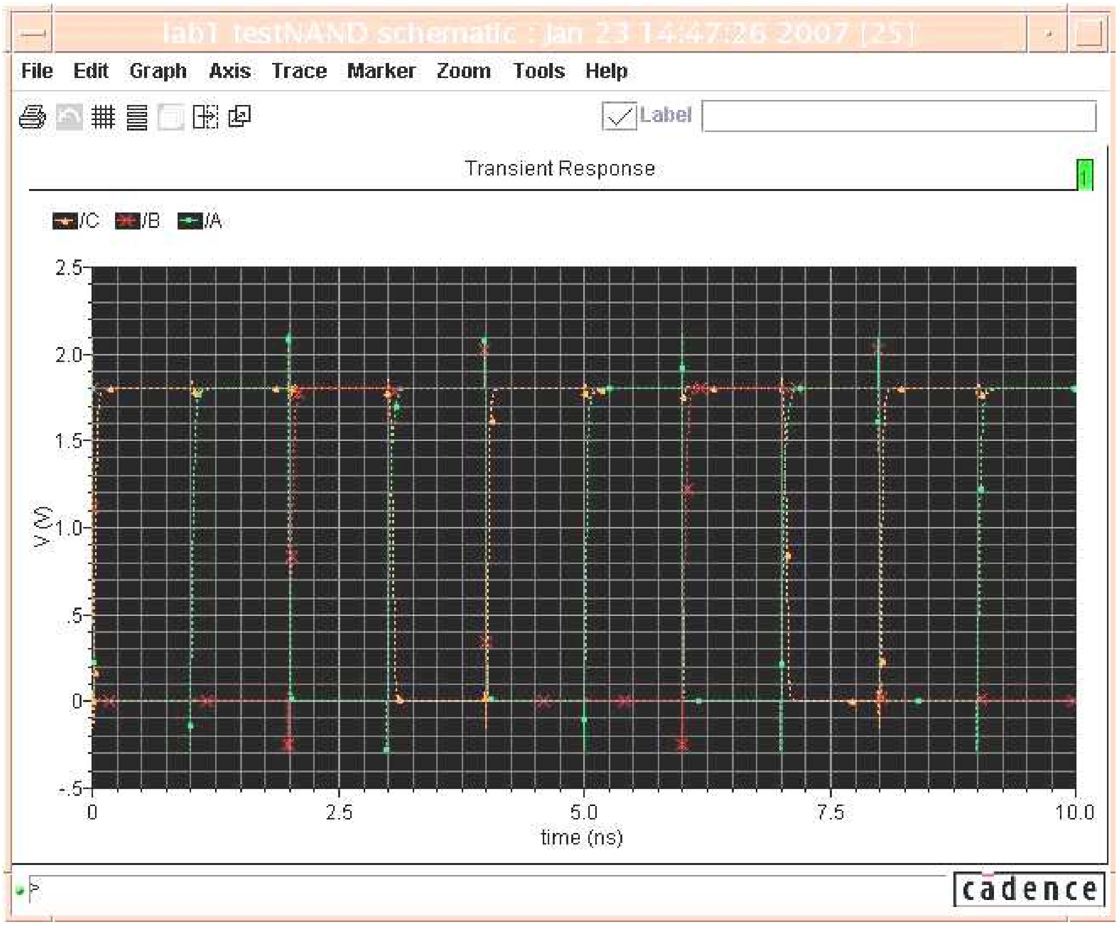
#	Name	Value

Outputs

#	Name/Signal/Expr	Value	Plot	Save	March
1	A		yes	allv	no
2	B		yes	allv	no
3	C		yes	allv	no

> Results in /home/bmorshed/simulation/testNAND/spectre/schematic

46) Click **Simulation -> Netlist and Run**, or the traffic light with green light on the right side. Turn off “Welcome to Spectre”. A message box should open and show log file. If the simulation is successful, input/output waveforms will be shown in a new window.


lab1 testNAND schematic : Jan 23 14:47:26 2007 [25]
File Edit Graph Axis Trace Marker Zoom Tools Help
Label
Transient Response
/C /B /A
2.5
2.0
1.5
1.0
.5
0
-.5
V (V)
0
2.5
5.0
7.5
10.0
time (ns)
cadence

Hints for rest of the lab of part A:

- If you want to find out what is inside a symbol, select it and press "**E**" or "**e**" to **descend** (first one lets you edit, second one only to view as read only). Press "^**e"** when you are done to **ascend** to the top level.
- Change vpulse to **vpwl** if you need to simulate a specific input waveform.
- **Size** your transistors for optimum performance.
- Use variables to sweep a parameter. In Analog Desing Environment, there is an area showing design variables. It is a good practice to assign variables to the parameters that you might want to change. Give meaningful names variables so that you can recognize them later.
- To **sweep** a parameter, select **Tools -> Parametric Analysis**. Enter variable name and range of the variable.
- If you need to find a complex expression, you can use **Output -> Setup** and then click **Open Calculator**. Use **Get Expression** to copy it and provide a name to the expression. Click **Add**.
- You can save your settings of Analog Design Environment by selecting **Session -> Save State**. To load a previously saved state, select **Session -> Load State**.
- Anytime you make any change in schematic window, make sure you save it before you simulate it.
- Use variables for defining widths of MOSFETs. You can assign values for these variables in Analog Design Environment (use Variables -> Copy from Cellview).
- To find delay, you need at least 2 inverter symbols, eg. Inv0 and Inv1, as the inverters in the source and drain are different size. However, a more efficient way to do this is to use the keyword **pPar("varname")**, where varname is the name that will be shown in symbol attributes, accessed in higher level. For example, in your inverter schematic, set value for nMOS and pMOS width as pPar("InvWidth"). Then create the symbol. Now, if you place this inverter symbol in another higher level schematic, when you select the inverter and view it's property (by pressing 'q'), you should notice a new variable at the bottom named InvWidth. You can assign different value for same symbol in different instances.
- For delay measurement, use 2 vpwl as sources and a capacitor as load. Set the timing of the vpwl's properly to get the given waveform at A and B.
- Notice that, pMOS to nMOS width ratios (β) is for equivalent inverter. Consult class notes if necessary.
- Delay measurements should be done by transient analysis. Make sure A and B matches given wave shape (not X and Y). Look closely to your graph for rise times and fall times (You should be able to find all cases as required from this plot) and tabulate them. Find propagation delay (maximum) and contamination delay (minimum) from the table. The schematic might be helpful for explanations.

- For power measurement, you will need vpulse instead of vpwl. Use a variable for period (1/f). Perform a transient analysis. Using small stop time will make simulation faster (use this for rough idea for range of simulation).
- For average power, find average current through the supply node of the vdc connected to NAND gate and multiply it with supply voltage (1.8V). Use calculator to find this expression.
- For parametric analysis, choose your variable name (f) as sweep variable in **Tools -> parametric analysis**. Select the sweep range and step size (linear) as you need. Choose **analysis -> start**. You might have to sweep to a higher frequency (depending on the width you are using) to see the trend.
- To get delay from calculator, Open calculator. Select Delay function. Choose Signal 1 as **VT("/A")**, signal 2 as **VT("/C")**. Threshold values should be **0.9**. Edge type, **rising to falling** for falling delay or vice versa for rising delay. Do transient simulation. Delay value should be shown in ADE window. Remember to use pulse input voltage for Q3 and Q4, as mentioned in instructions. Look at a couple of simulations and verify manually to make sure your function is working properly. You can also sweep a variable for W-NAND and get a plot with Parametric Analysis.
- To find delays manually, you can use transient analysis with different sizes. Make a table of values in this case for t_{pd}, t_{pdr}, t_{pdf} etc. and plot it. You should do this to verify your results.