York University
Dept. of Electrical Engineering and Computer Science

A Tutorial for Cadence used in Course EECS3611

*Originally developed by Yang Zhao in 2017 and revised by Zhongxia Shang in 2020.

*This tutorial is based on Cadence User Guide and TSMC cmosp18 180nm PDK documents

Table of Contents

Introduction

This tutorial will introduce the use of IC616 series tool (One generation of Cadence) for simulating circuits in our course. Cadence is a suite of tools for IC design. It allows for schematic capture, simulation, layout and post-layout verification of analog and digital designs. We will be using a portion of the analog design flow, which can handle up to 200,000 devices. **Composer** is the first tool we will use to enter/capture the schematics of circuits that we will design and the test benches with which we will simulate the circuits. Then we will use the **Analog Design Environment** (ADE) to configure a simulator, in this case **Spectre** is used to simulate our circuit with device models that represent the transistors in our circuit. Device models are decided by the adopted process which is provided by the foundry, the manufacture that will fabricate our circuit. The process we will use is **TSMC 180nm**.

This tutorial will explain how to set Cadence up on the CMC Server. An overview of the work flow in Cadence is shown in Fig. 1. Starting from near the top of the figure we will use the Command Interface Window (CIW) to start the schematic composer. Then we will capture a schematic of an op-amp and create a symbol for it. After that we will setup a test bench to test the op-amp to optimize our design parameters.

We will open ADE to configure and run our simulation in Spectre. Then we will plot our results. A circuit simulator is a tool that you need to use to design your circuits efficiently. The simulator is good at solving thousands of operating points. A good way to use the simulator is to first understand the circuit and then sweep the simulation over the entire area where the solution lies. We can run DC simulation to get the DC operational point of our circuits and adjust the design parameters like length and width of the MOSFET to take all the MOSFET into saturation region in our op-amp design to get the maximum amplifying performance. After the DC simulation we will have a set of parameters. Then we can run the AC simulation to observe the circuit's frequency response to further optimize the parameter to get the required bandwidth. Also Transient simulation is an important part to get the driven capability, output-swing and any distortion of our circuit. There are also a lot of other simulations available in our ADE like noise simulation which will present the noise performance of the designed circuit. Due to the time constraint those simulations won't be covered by this tutorial. If you are interested in other simulations, all of them can be found in Cadence User Guide. The simulator is not your brain; it can't understand anything about the circuit. If you are just collecting data and not thinking about it, then you are wasting time. The simulator isn't a pen and paper calculation. If you are running single point calculations and aiming for exact agreement with hand calculations you will be frustrated. Instead of running single point simulations and stabbing in the dark, you should think about sweeping some parameter to collect data with trends that you can think about.

Now we will start our schematic design of a single stage single input common source op-amp circuit with source degeneration which is shown in Fig. 2.

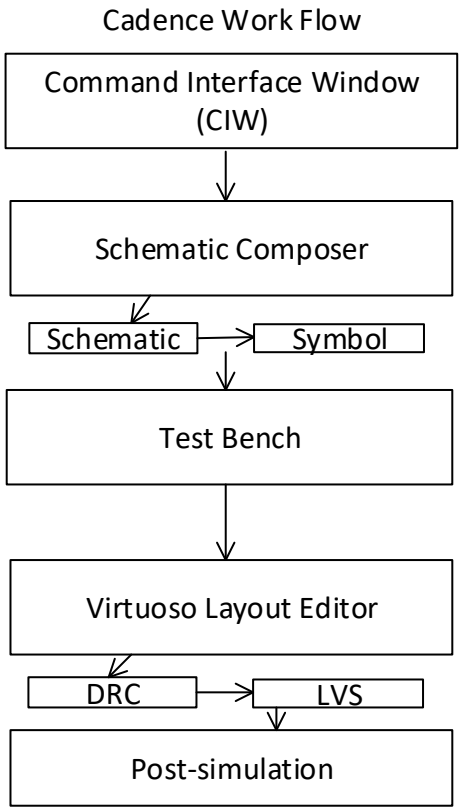


Fig. 1 Cadence Design Flow

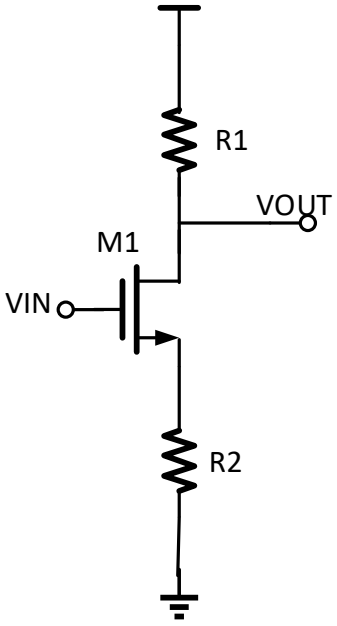


Fig. 2 Common source stage with source degeneration circuit.

Shorthand Conventions

The following are shorthand conventions used in this tutorial

1. When navigating a series of submenus, the tutorial will abbreviate the instruction "Start in the Command Interface Window (CIW) and click on Tools, then click on Library Manager" to Click on CIW->Tools->Library Manager.

2. Cadence has many keyboard shortcuts. When the tutorial writes a letter of a command in parentheses it means that letter is the short cut. Capitalization is significant. For example, i is the shortcut for (i)nstantiate.

3. All commands are in italic format.

Getting Help within Cadence

Click on Help within a Cadence window. This will try and start an instance of the Cadence document server cdsdoc.

The easiest way to use cdsdoc is to click on Search and enter specific terms that you are interested in. Feel free to browse, in order to see the scope of Cadence.

Troubleshooting

This is a very brief section on trouble shooting. Keep in mind that this is an exercise to configure and run some simple simulations in Cadence. The problems are computer or programming problems. Remember the adage, "Stop the bus. Get out of the bus. Close the doors. Open the doors. Get in the bus. Start the bus." So at any step if you think you have worked Cadence into a weird state. Save your work. Copy it to a new cell, new saved state etc. Exit Cadence if you think it is that hosed. (This is very possible). Go back to the last known good position and start again.

Sometimes, cadence is crashed due to improper operation. If it can't be closed in GUI, you should kill it in command line. To kill it you should find the process ID in you terminal by typing:

ps –C virtuoso

It will show you like following information:

PID TTY TIME CMD

2060 pts/6 00:00:02 virtuoso

The number from the column PID is your cadence process ID.

Then you can end it by typing:

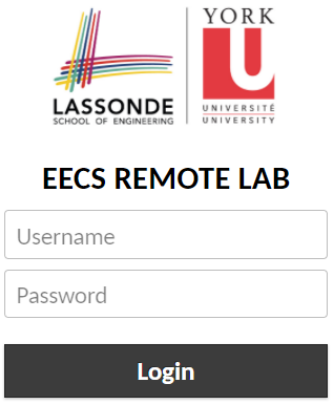
kill 2060

Then you can restart it.

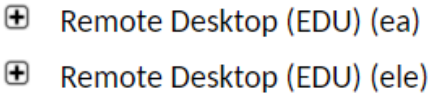
Remote Access

First you have to connect remotely using https://remotelab.eecs.yorku.ca/#/

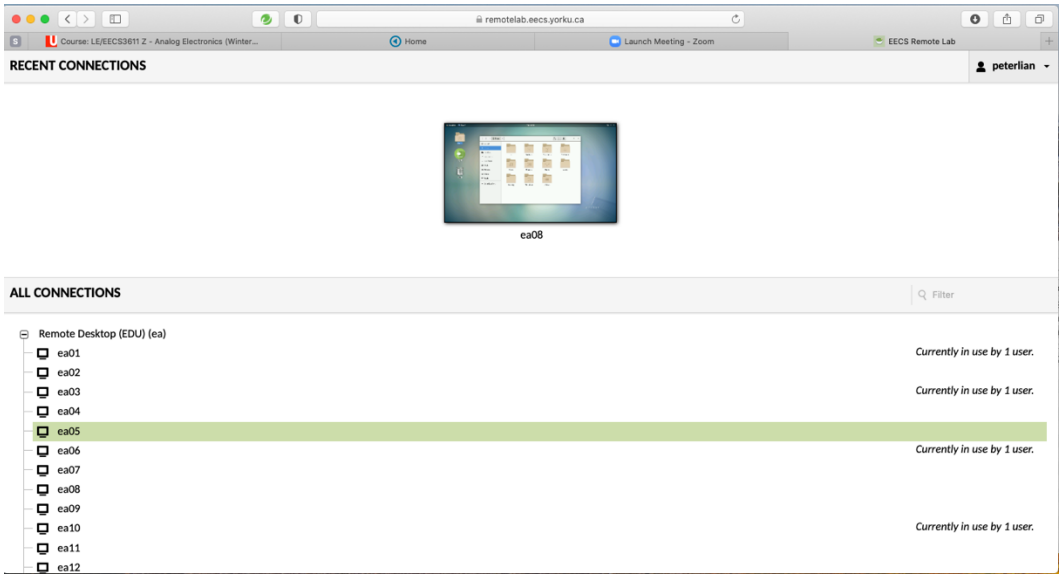
If you have not already setup your remotelab access you can check this link:
https://wiki.eecs.yorku.ca/dept/tdb/services:remotelab



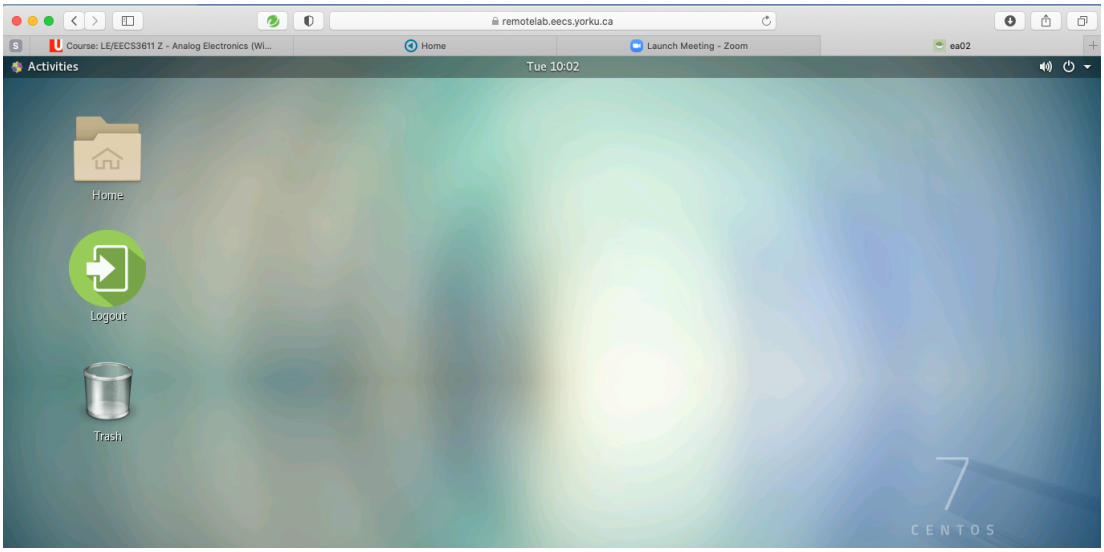
In this window enter your eecs username and password. Once you entered the system, you can choose one of the "ele" or "ea" systems. Click on "+" sign as shown below.



You will see a list of computers for access. You may want to select one with less users to access as illustrated below.



Once you click on one of the machine, e.g. "ea02", it will lead to the following window.



The ea systems are on linux and the ele systems may be on windows. So, to switch to linux just click on the switch to linux icon from shortcut folder.

There are other ways to connect to cadence using your windows machine directly (Like using MobaXterm or VNC) but you may lose connection repeatedly and the remotelab system will give you a better experience.

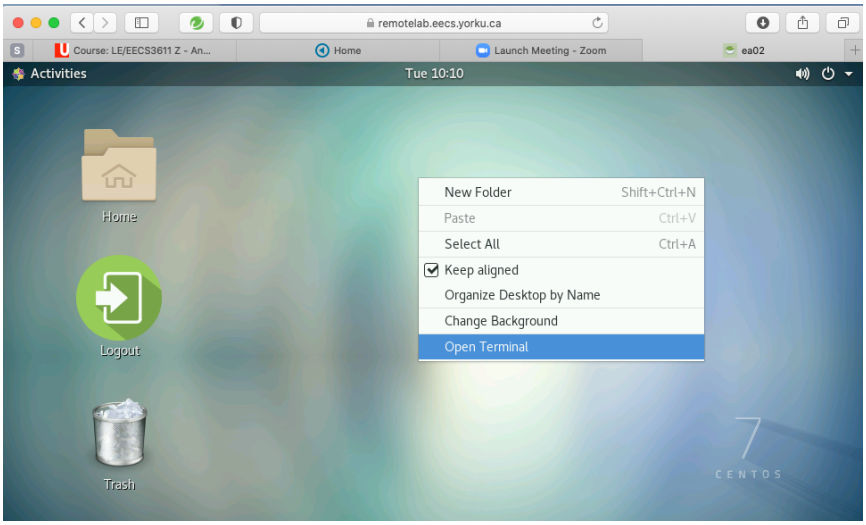
Environment Setup

Setup Cadence on CMC server

1) Login to a Linux machine by Right-click mouse to open a terminal like Fig. 3:

3) execute the command: **touch .3611**

4) **close terminal window and reopen a new terminal.**



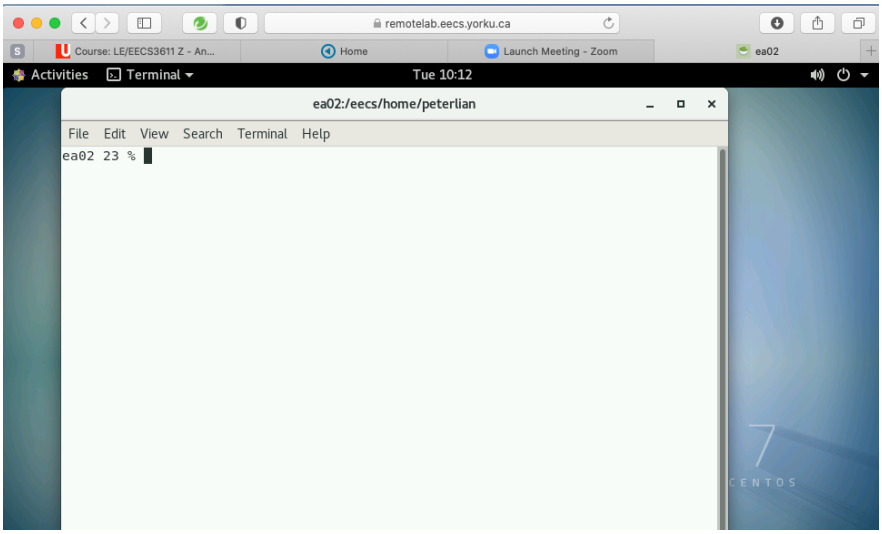


Fig.3 Open a terminal in Linux

5) Create your design directory (It's very important to create a separate design folder for Cadence running as in this way, otherwise you will messed everything up if you run Cadence in your home directory):

mkdir eecs3611

Note you can use command *pwd* to show your current directory, and use *ls* to view the files in the current directory.

6) Navigate into this directory: (After the first time you do not need to repeat the previous steps.)
cd eecs3611

You can create the directory or navigate to it using the Linux interface by creating a new folder in your home directory and entering that folder and open the terminal inside that folder.

7) Complete the setup by running the following command.

tsmc180nm.setup

in your terminal firstly.

8) To Run Cadence, then type:

virtuoso &

Then the CIW windows pop-up as shown in Fig. 4 (It may take some time so be patient)

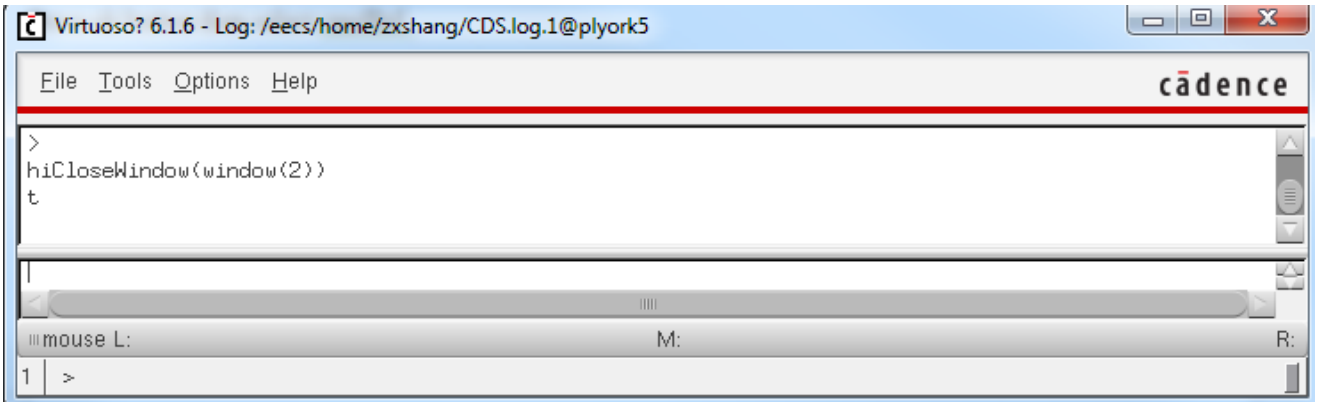


Fig. 4 Cadence CIW window

Open Library Manager

First, we need to open the library manager for our design as shown in Fig. 5.

CIW->Tools->Library Manager

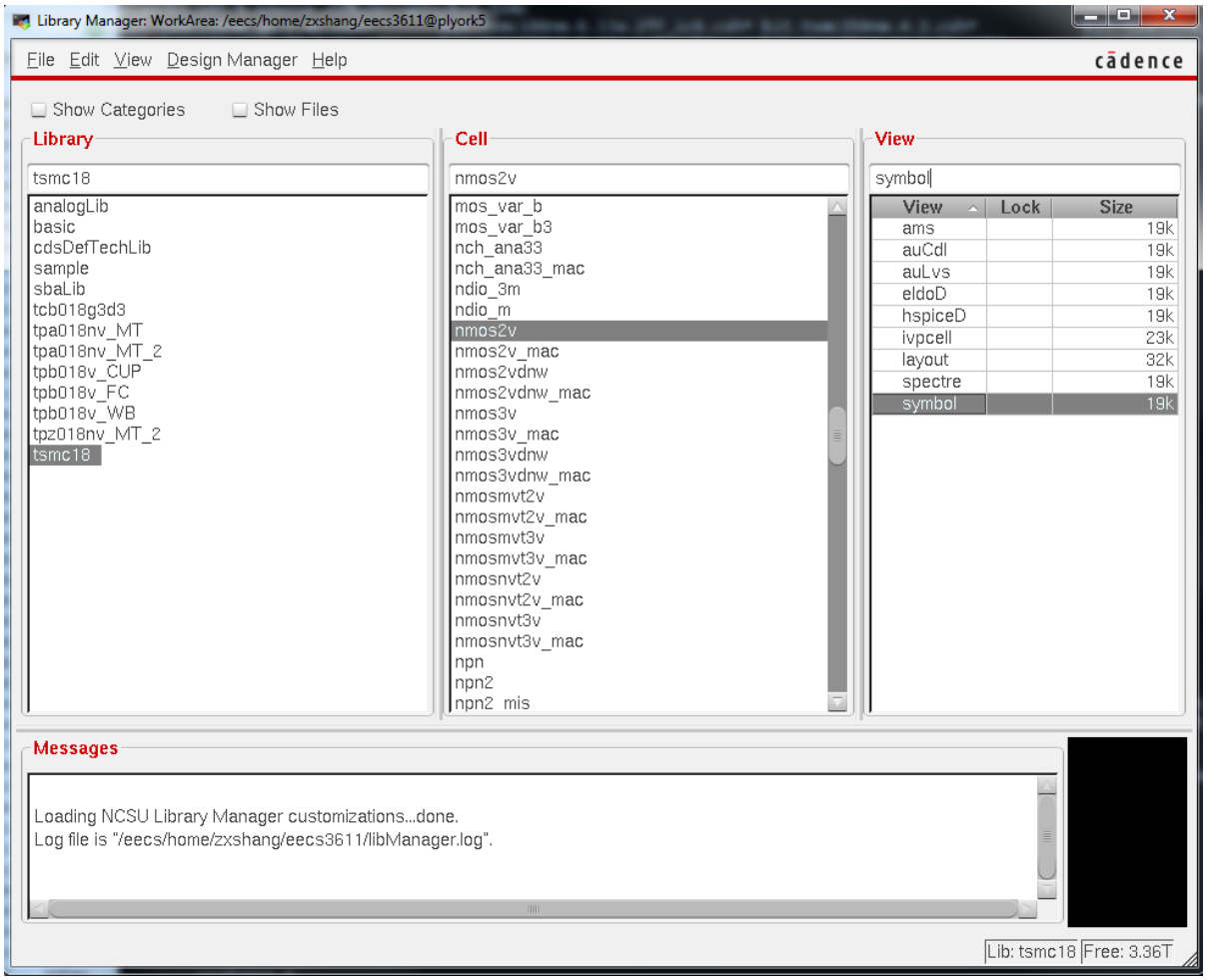


Fig. 5 Library manager

From this library manager we can find it's organized as library, cell and view three columns. In the library column, several created library are provide to us. In our case, we will use the analogLib which is used in creating the test bench and our process lib which is tsmc18. In the cell column, you can type nmos2v (N-type MOSFET) and rnwell (N-type well resistor) to find the necessary devices in our circuit. In creation, we will call the symbol view of these two devices.

Create Library

CIW->File->New ->Library

A window pops up shown in Fig. 6. Type a name in the name box and press Ok button at the left bottom of this window. Then select "attached to an existing technology file".

Press ok button then select tsmc18 as shown in Fig. 7.

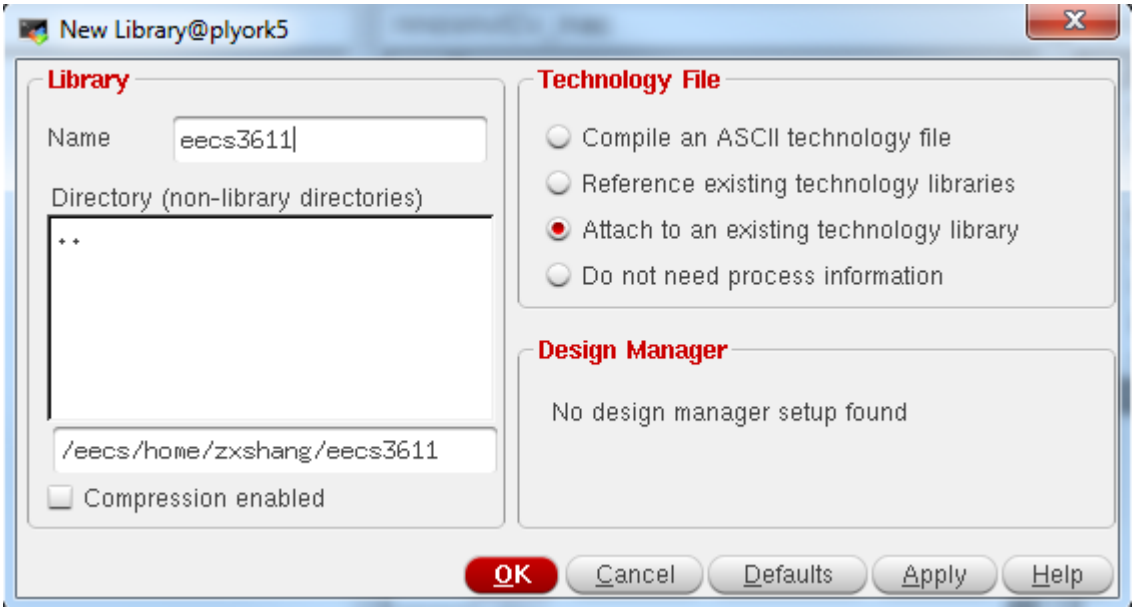


Fig. 6. Create new library.

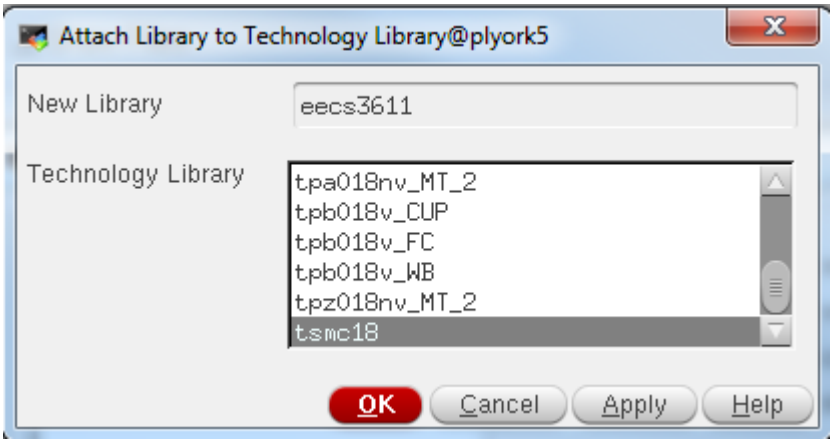


Fig. 7 Tech selection

Create Schematic and Symbol

Create Schematic

In the library manager library column, select the library eecs3611 we created previously. Then

File->new->cell view

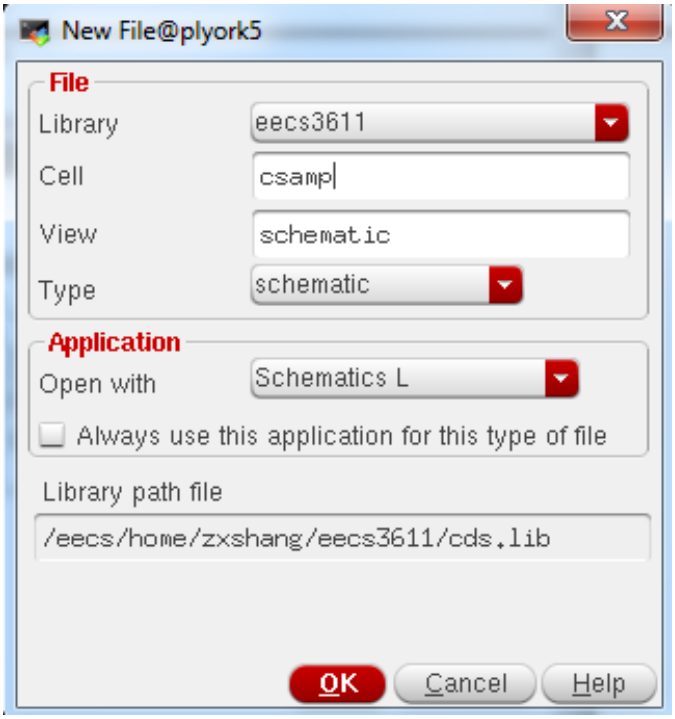


Fig.8 Create new schematic

Type your circuit name in the cell box. In my case, I use csamp as the name as shown in Fig. 8.

Then the composer will pop up automatically as shown in Fig. 9, also you can double click the cell name in the library manager to open it.

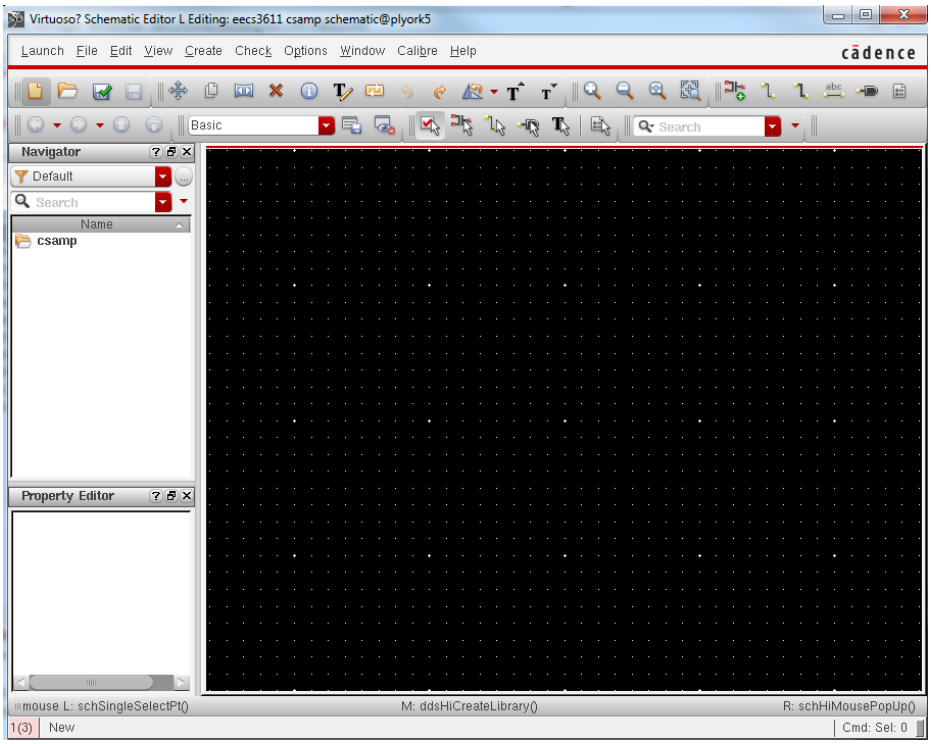


Fig. 9 Schematic composer

Before we start, here are some useful commands that we will use later. (i)nstantiate creates a new object in the schematic. (p) creates pins to allow signals to leave the schematic. (c)opy allows you to copy objects. (m)ove with a lower case m moves objects and drags their wiring with them. (M)ove with an uppercase m moves objects after disconnecting their wires. (l) allows you to create labels for wires so that they don't get stuck with the default names, like net994875. (q)uery opens a dialog box that shows an object properties. (w)ire allows you to create wires. After you type w, left click to start a wire, left click to make a corner and double left click to end the wire. Some commands like copy, move and wire have dialog boxes with more options that can be hidden. (F3) displays these dialog boxes.

1. Instantiate the resistor:

Press i the add instance window will show up, you can press the browse button aside the library box and select library tsmc18, find rnwell cell and select its symbol view as shown in Fig. 10.

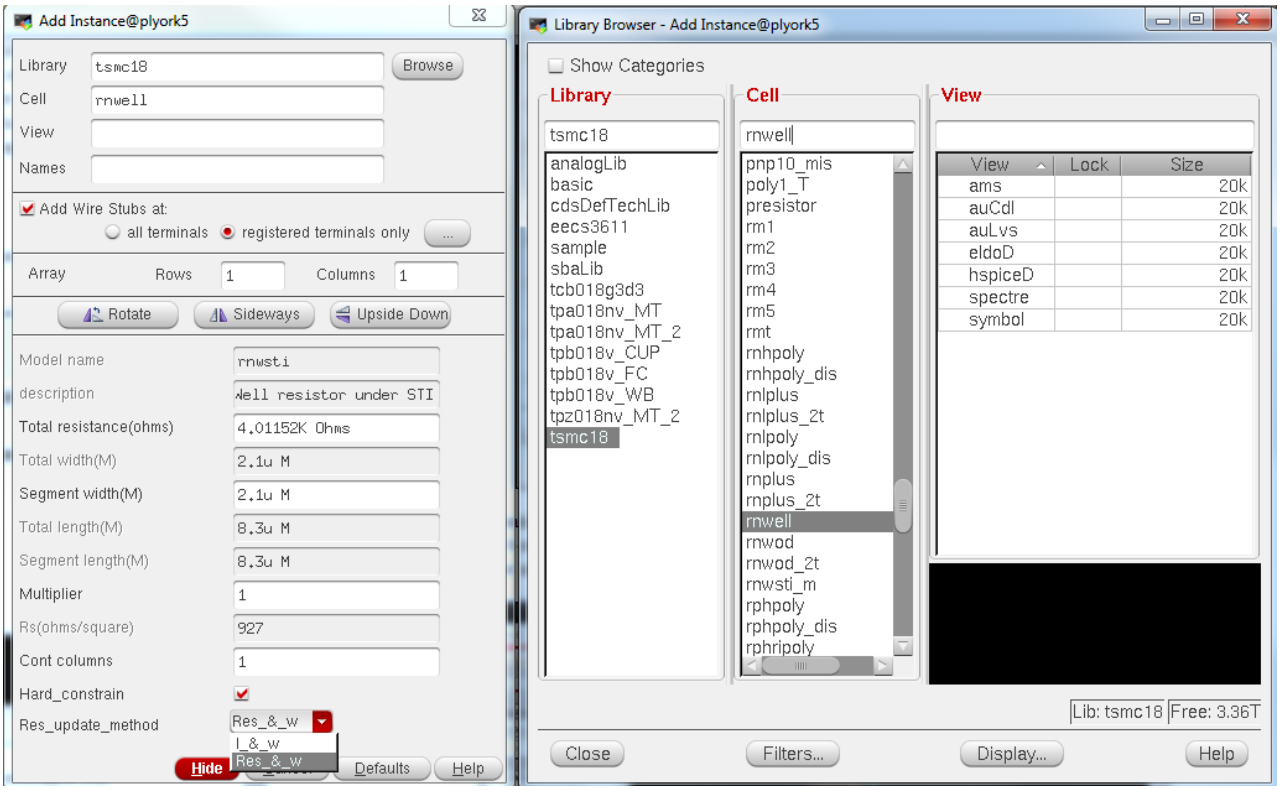


Fig. 10 Instantiate the nwell resistor

Change the checkbox “Res_update_method” to “Res_&_W” you can set the resistance by typing a value in the resistance box otherwise you can only change it by setting the width and length.

Press Hide, place the device in the schematic composer workspace.

You can insert the NMOS using the same way by finding the nmos2v cell in tsmc18. In the property window of the NMOS, you can find the “l”, “w”, “total_width”, “number of fingers” and “multiplier”. The total MOS numbers is the product of “number of fingers” and “multiplier”, so the total width (not the “total_width” in the window) should be the product of the “w” and the total MOS numbers. For the convenience of our design, first we can set the width of all fingers as variable “w”, “number of fingers” and “Multiplier” as 1and keep the default length as shown in Fig. 11.

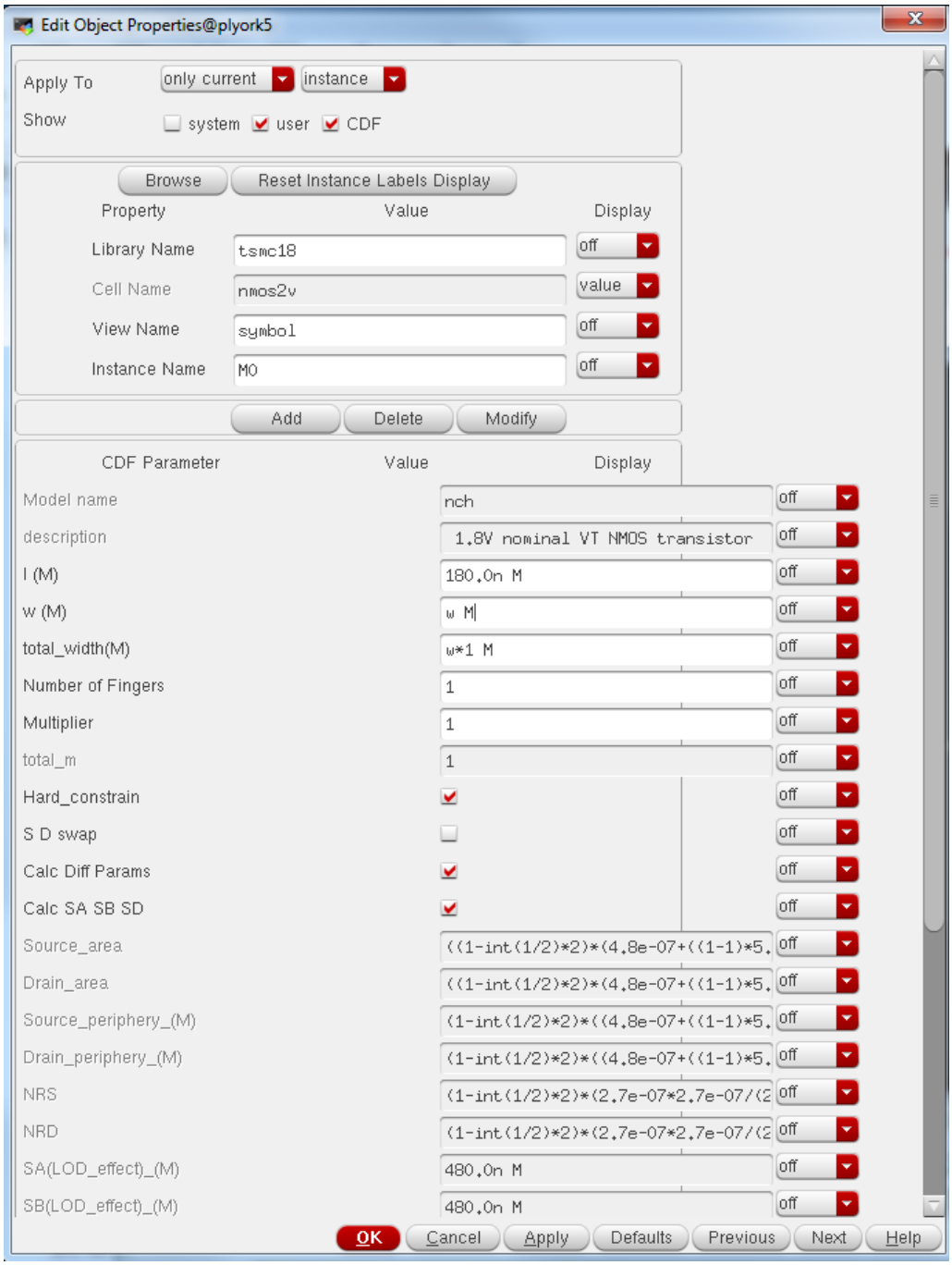


Fig. 11 Property window of the NMOS transistor

After the devices placement, you can press w to connect them by wires.

The last step is the creation of pins for your circuit by pressing p then the window in Fig. 12 shows up:

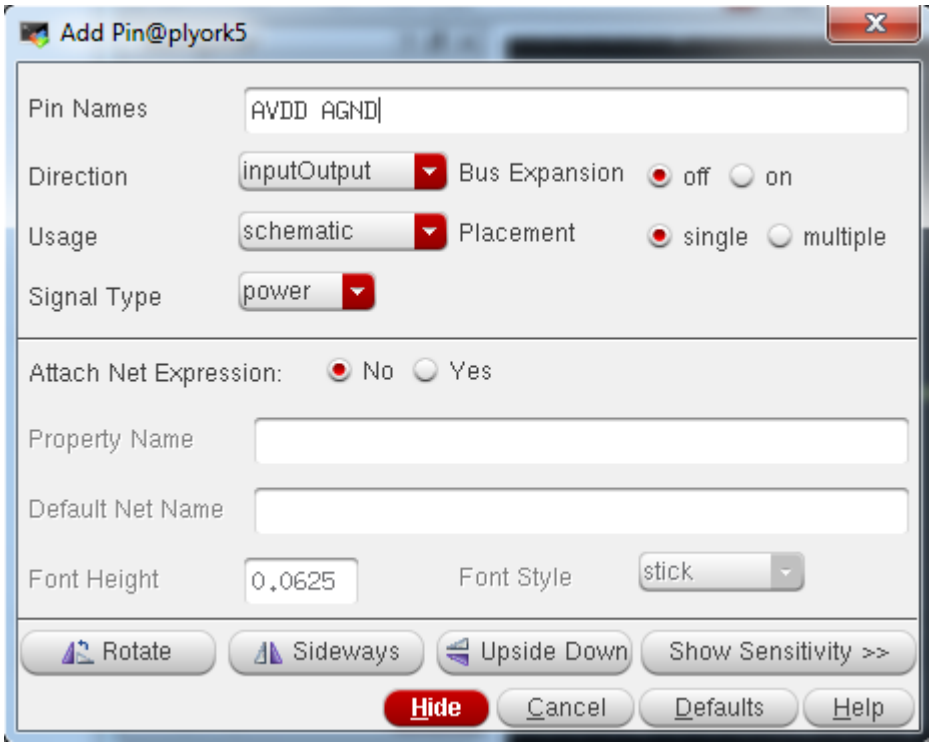


Fig. 12 Create pins

You can sequentially insert multi pins with same directions; like in the up figure you can separate different pins in the pin names box by space. For the power and ground terminal, the direction should be inputOutput. Input terminal should be input and output terminal should be output.

To soft connect on wire with another wire or pin, you can press “L” to label it with same name with the target wire or pin, like Fig. 13:

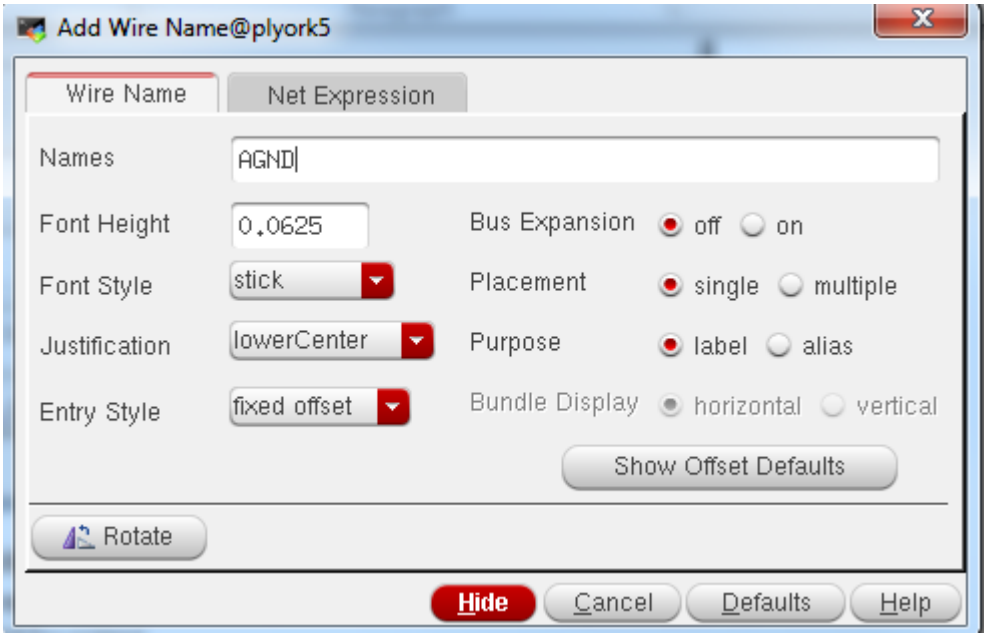


Fig. 13 Create label

Finally, your circuit should be like Fig. 14 (remember to save your circuit schematic by pressing check and save as circled by red):

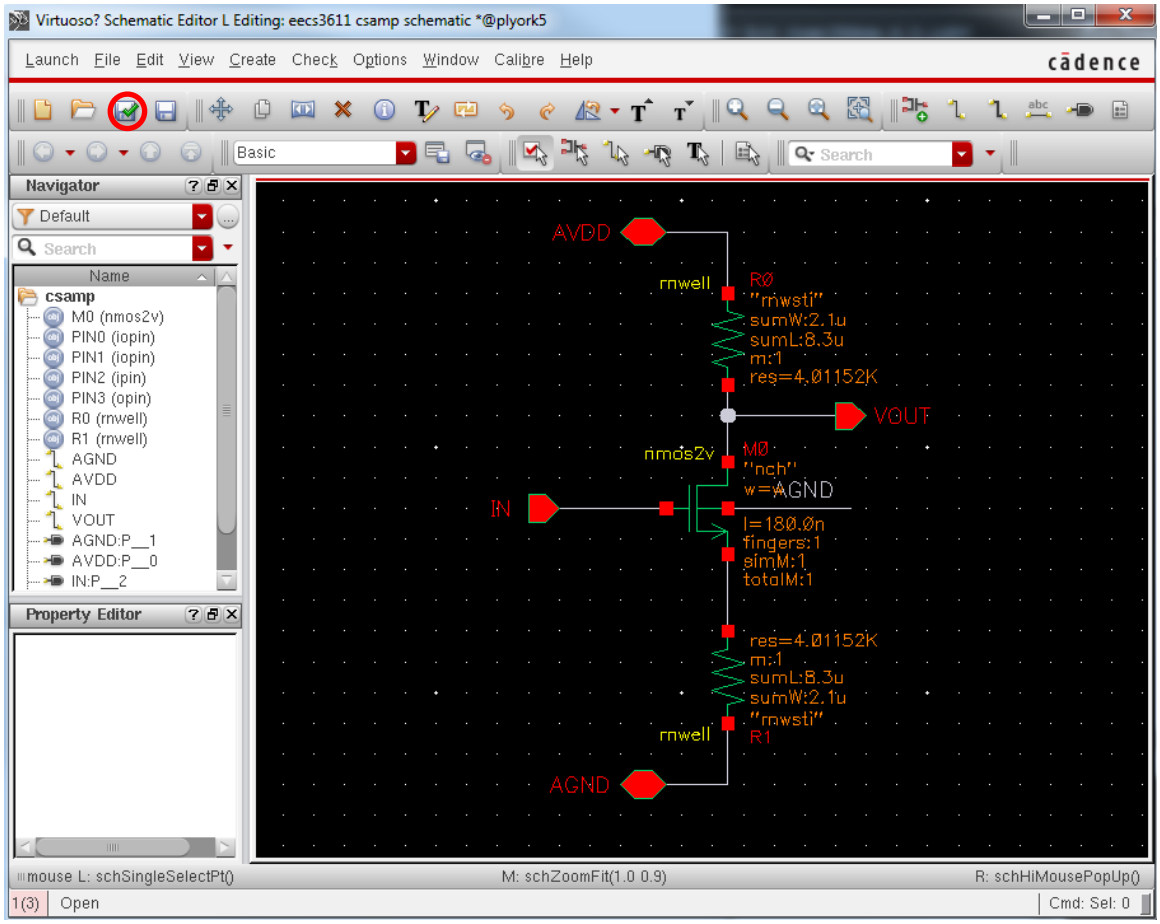


Fig. 14 Complete schematic

Create Symbol

From the Virtuoso Schematic Editor window, by the step:

Create->Cellview->FromCellview

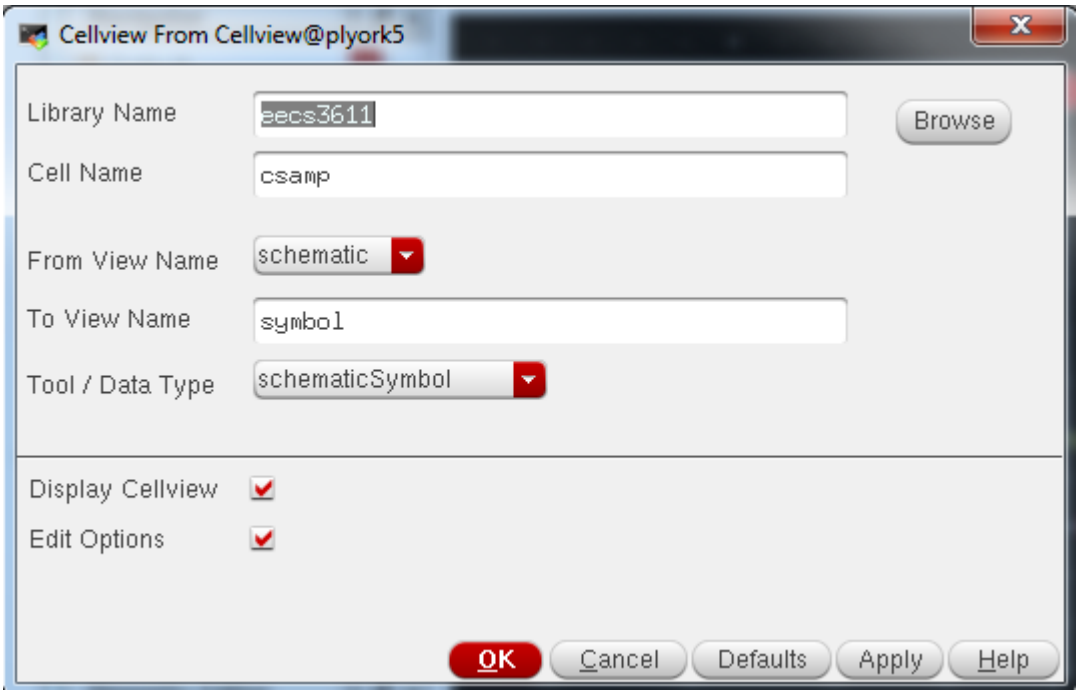


Fig. 15 Create Cellview

Make sure the box “To View Name” is symbol. Then press OK for twice and the automatically created symbol will be as shown in Fig. 16:

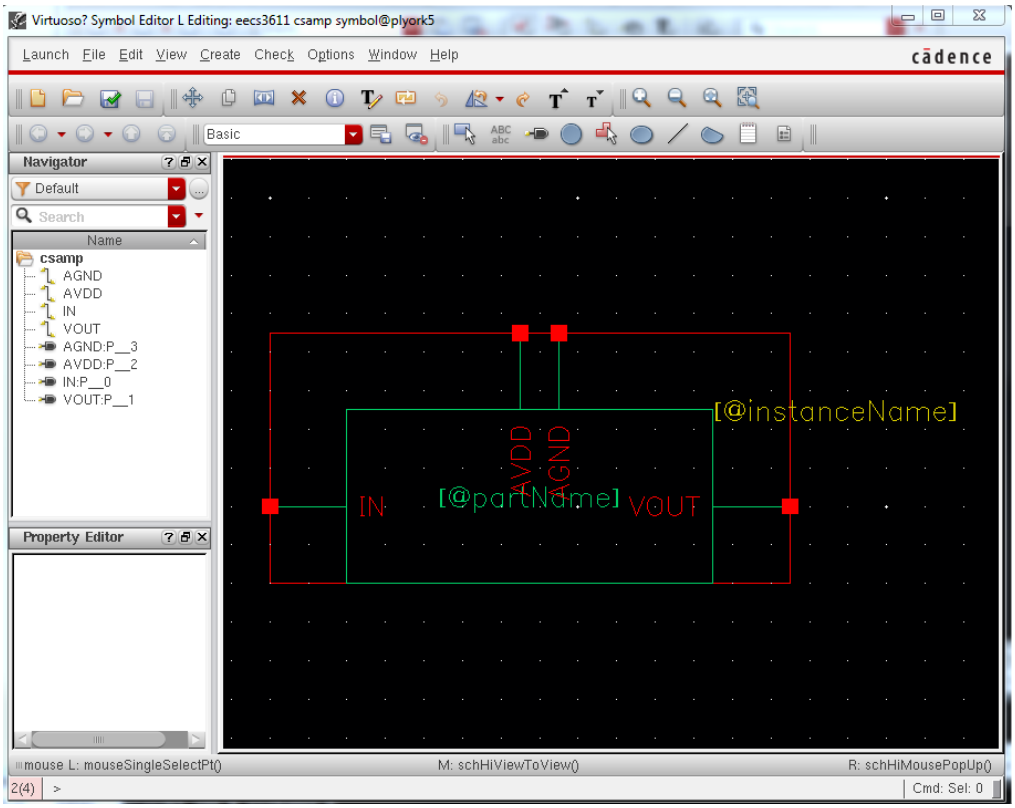


Fig. 16 Automatically created symbol

To create a typical amp symbol, we can delete the rectangular green and red box then create a triangle green box by Creat->Shape->Line. Then we can move the pins by select and press m. During the movement, by pressing r we can rotate the pins' directions. Then we should add the selection box by Creat->Selection box->automatic. Here is a sample symbol:

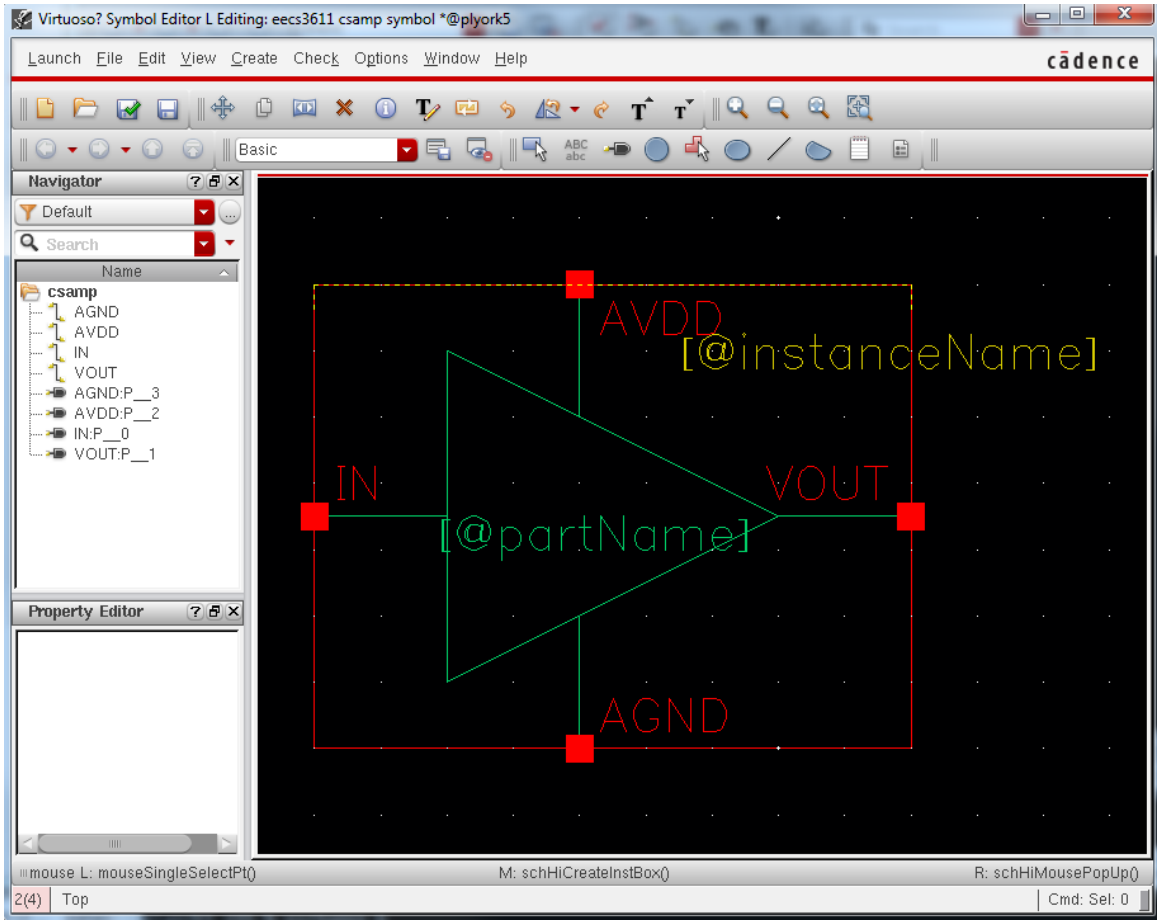


Fig. 17 Complete symbol

Then press the check and save and close this window.

After the upper two procedures, you will find there are two view names appears in the view column of the cell csamp from the library manager which indicates we have finished the creation of schematic and symbol for our circuit.

ADE Simulation

Create Test Bench

Create a new cell in our library eecs3611:

Select this library, then File->new->cellview

Fill the cellview name as sim_csamp.

Insert the csamp from our library and a DC power source vdc, a sinusoidal input voltage source vsin, an output load 1pF capacitor cap and the ground gnd from analogLib. Connect them by wires, and then our test bench will be like Fig. 18:

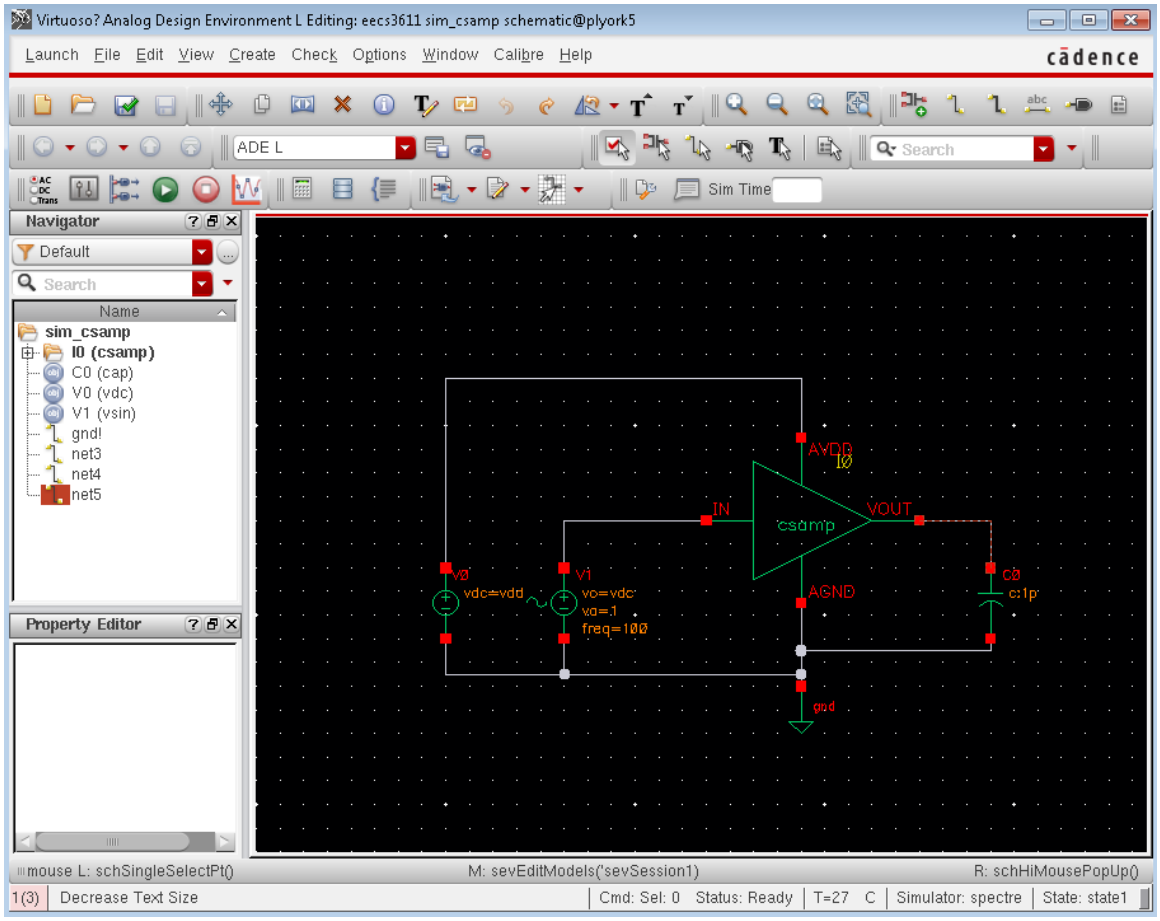


Fig. 18 Complete testbench

You can set the dc voltage of vdc as a variable “vdd” that we can change it later to change the power supply. For the input sinusoidal setting we have a property window shown in Fig. 1. We should notice that different simulation apply the different parameters. The DC voltage box is only used in dc sweep, so we can fill it as a variable “vdc”. AC magnitude and AC phase are only used in AC simulation for frequency sweep which means for all frequency the input keeps the same amplitude. Delay time, Offset voltage, Amplitude, Initial phase and Frequency are used for transient simulation. Default value of Delay time is 0. Set the offset voltage as “vdc”, amplitude as 1V and frequency as 100Hz.

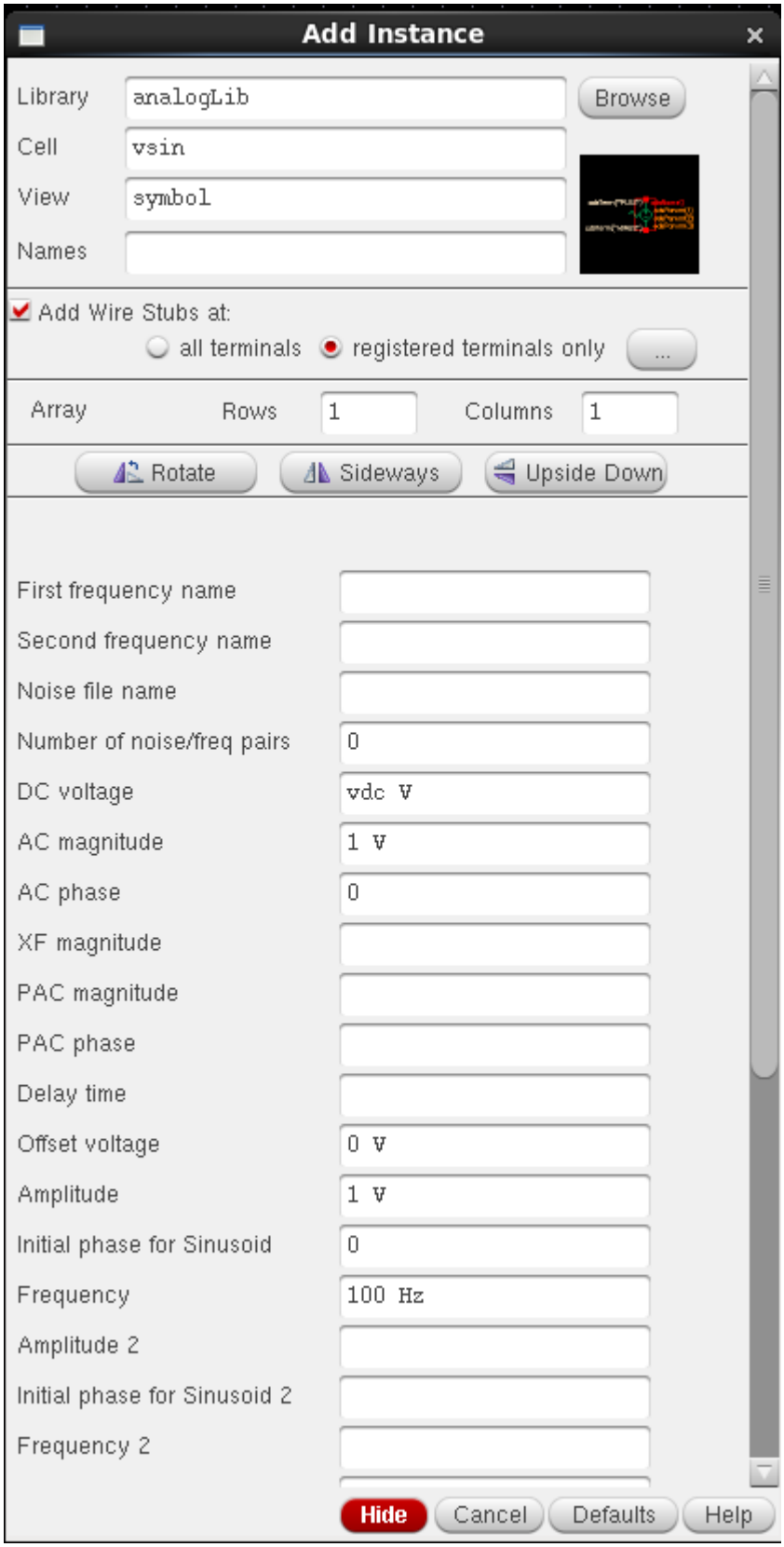
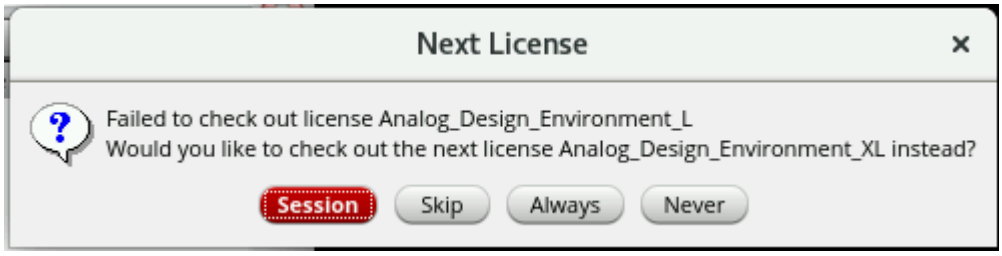


Fig. 19 Property setting of input signal.

ADE Simulation

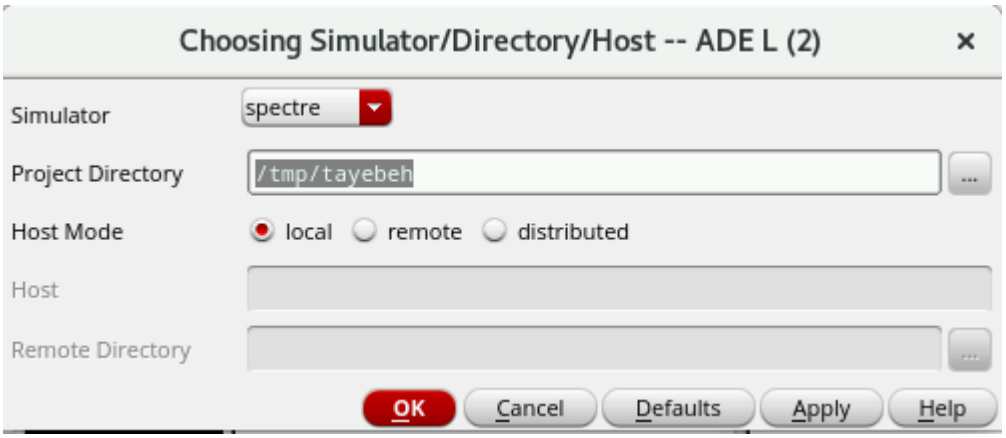
From the test bench schematic window, Launch->ADE L then ADE window will show up.

You may get this message first. This is fine. Just click session and continue.



When you run a simulation in cadence a lot of log files will be generated, and you may get some error as you do not have enough space in your home directory. Therefore, you need to point to a temporary directory to save these files. To do that click on

Setup->Simulator/Directory/Host



In this window you need to change the project directory. You should make a directory in /tmp in your name and then point to that directory in this step. Make sure to delete the directory from /tmp when you got your results. You need to delete the data from /tmp to avoid using up this space as this is a shared space.

In the ADE window, first we should copy the variables from the test bench schematic by Variables->Copy from cellview.

Set vdd, vdc and w as 2, 1 and 1.2u separately.

The device model can be checked through Setup->Model

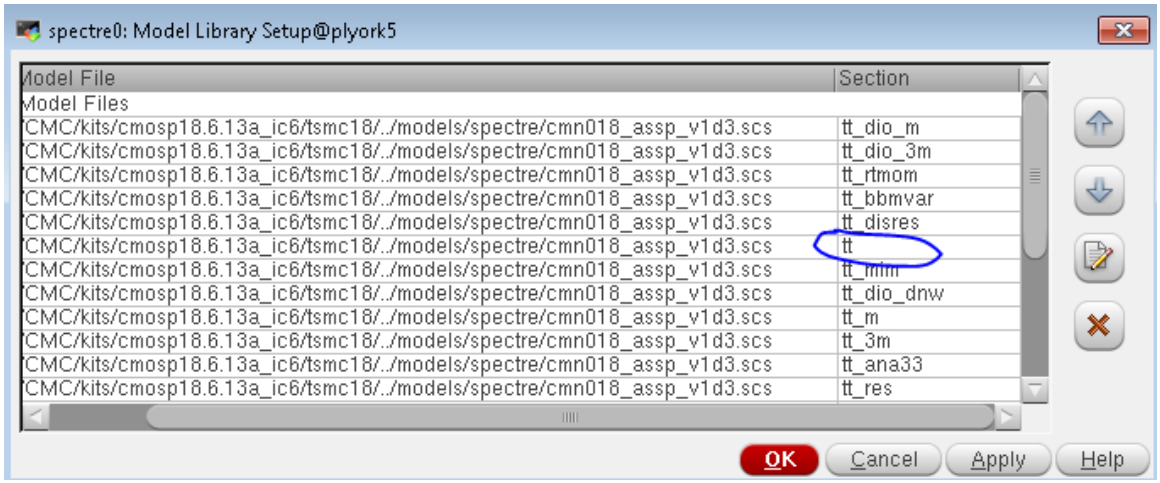


Fig. 20 Model library setup

As circled, the model for the MOSFETs is set to tt corner, you can double click it to change the setting.

Different Corners are shown in the following table:

Corner	NMOS	PMOS
tt	typical	typical
ff	fast	fast
ss	slow	slow
sf	slow	fast
fs	fast	slow

Since our circuits just employed the NMOS so, FF and FS or SS and SF should give the same simulation results.

Add simulations by Analyses->Choose->tran, since our input sinusoidal transient signal frequency is 100Hz, we can set the stop time as 20ms to observe two entire cycle.

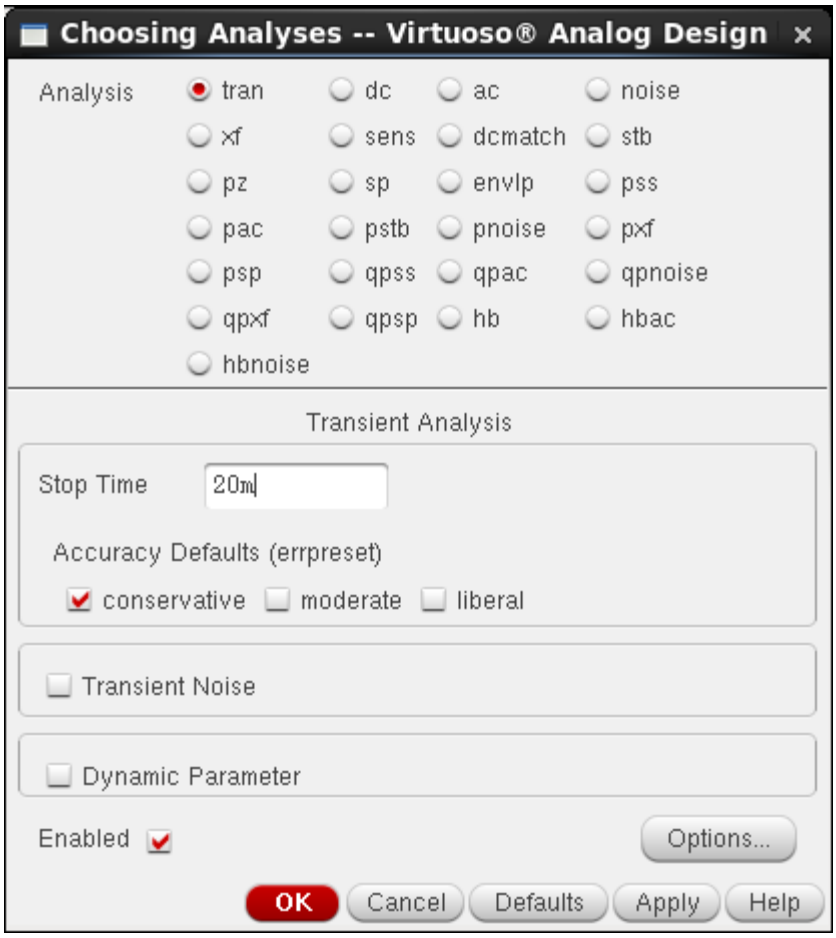


Fig. 21 Transient simulation setup

For the dc simulation, use the setting as shown below.

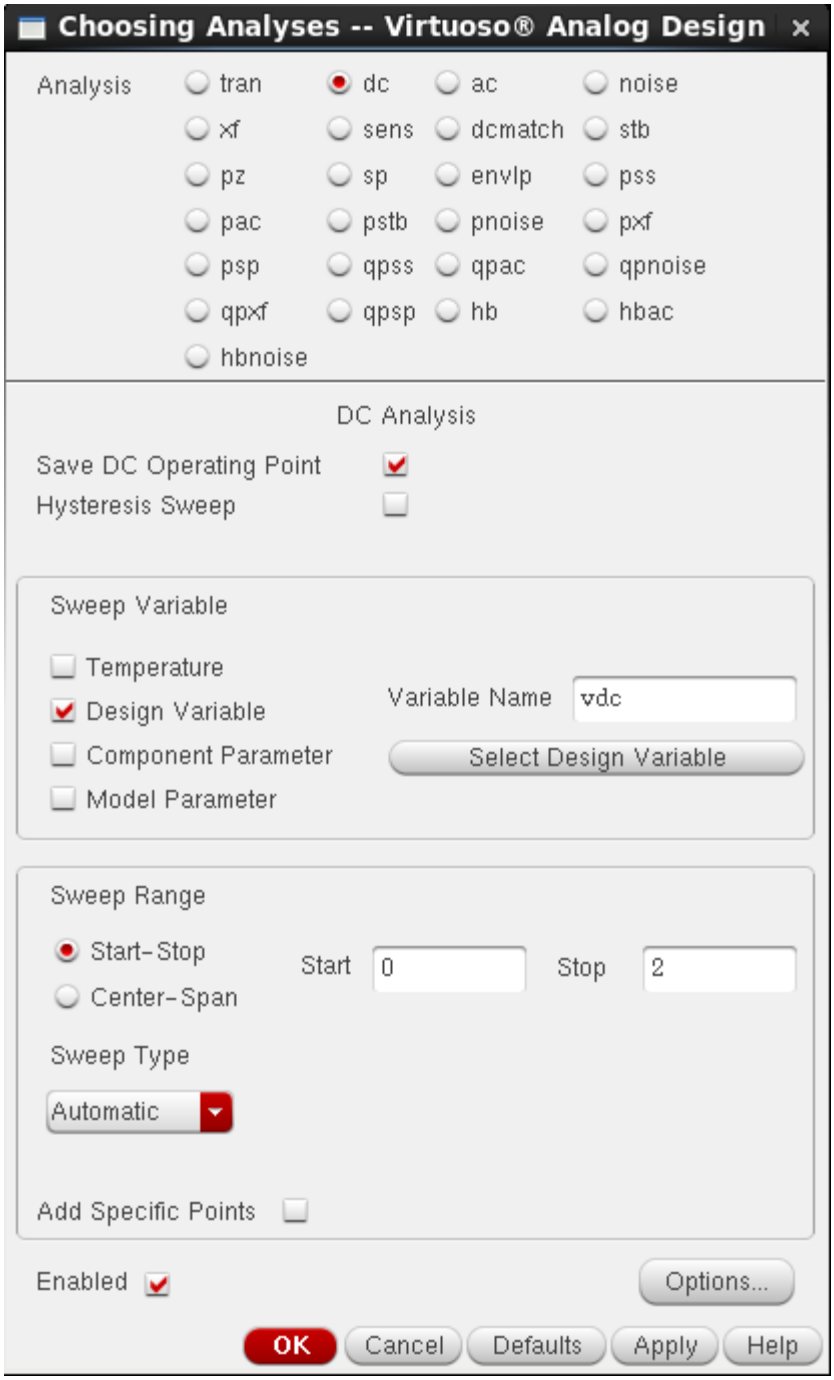


Fig. 22 DC simulation setup

AC simulation settings are shown here. Notice the stop frequency is 10M not 10m.

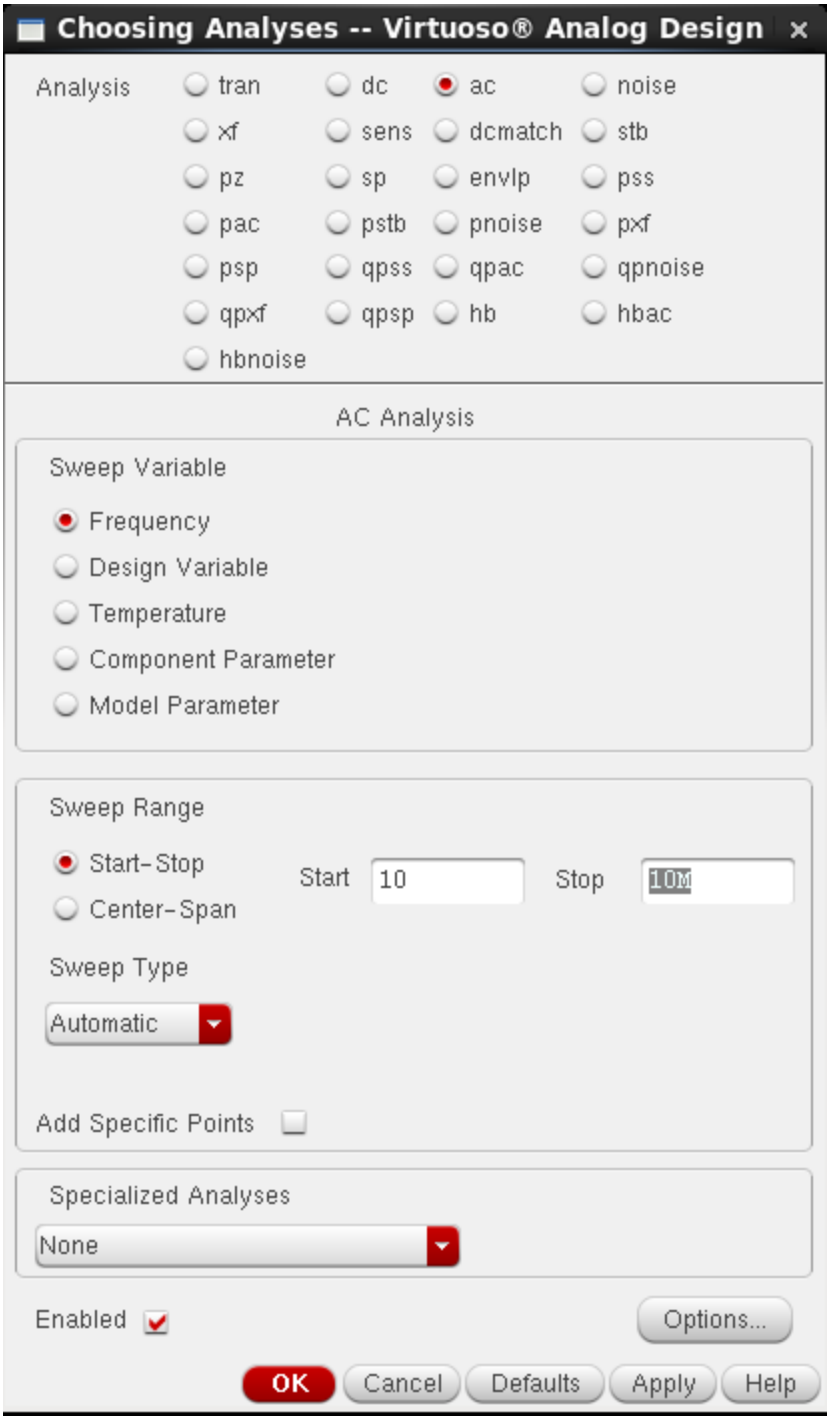


Fig. 23 AC simulation setup

Setting the output plot after the simulation through outputs->To Be Plotted->Select on Schematic. After the signal selection you can press ESC to abandon the command. It should be noticed that all the command in cadence is continuous available until you cancel it by press ESC or switch to other command.

Finally, our ADE setting will be similar with the Fig. 24. By pressing the green "netlist and run" arrow in the on the side menu, simulator will run the transient, DC and AC simulation separately and present all the results in visualization window.

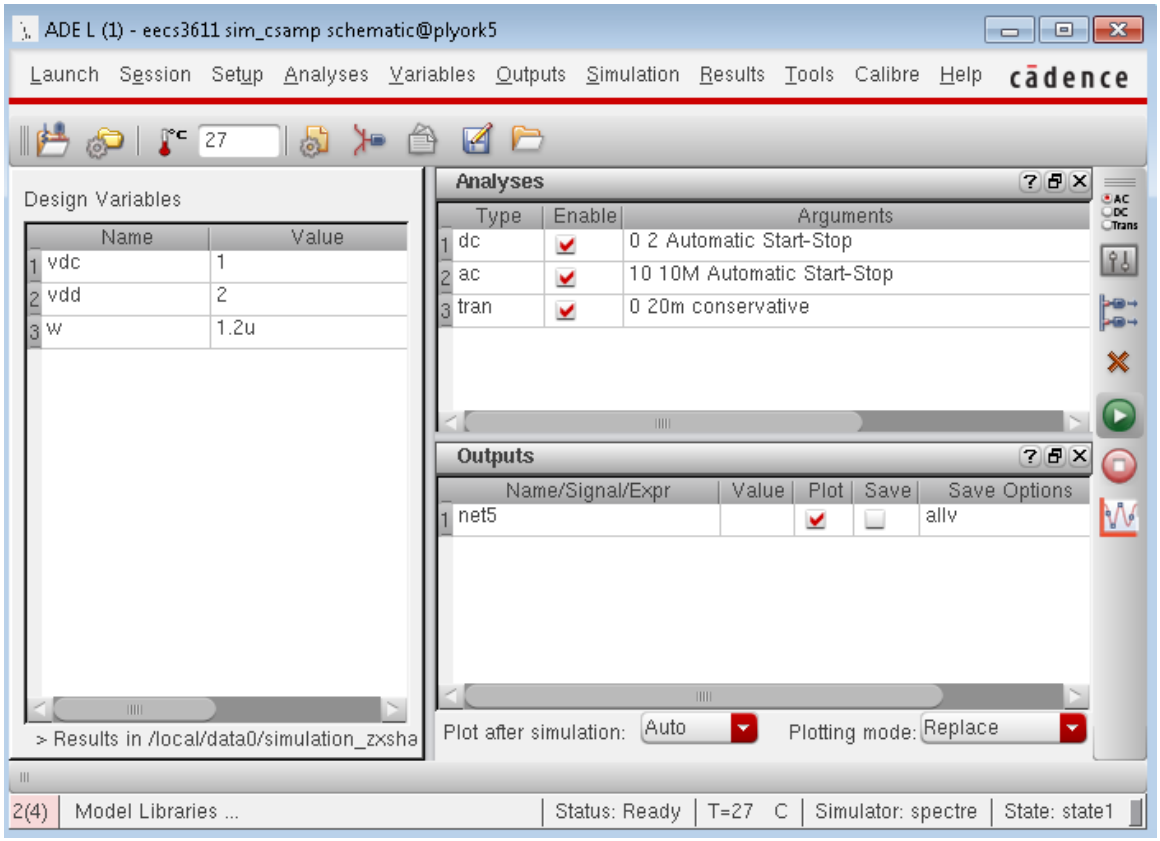


Fig. 24 Complete ADE

****

The following steps should be taken ONLY IF there is an error reported during the simulation:

****

Copy the scripts from CMC server to your project directory

cp /CMC/scripts/ kit.tsmc180nm.6.13a_ic6.csh ~/eecs3611

In the eecs3611 directory, open the copied file

gedit kit.tsmc180nm.6.13a_ic6.csh

The following figure pops up and you will need to change the circled part as shown in Fig. 25, which will call a newer version of the simulator in cadence and that should work properly.

```
#!/usr/bin/tcsh

if( $?CMC_HOME == 0) then
        setenv CMC_HOME /CMC
endif

set VEND=TSMC
set KIT=cmosp18
set KIT_LIB=tsmc_180nm_libs
set VER=6.13a
set VERCDS=ic6

setenv CMCcdsTech cmosp18.6.13a_ic6
setenv CMCcdsTechBase cmosp18
setenv CMCcdsTechVersion 13a

set cDir=`pwd | tr -d "\n"`
if( $cDir == $HOME ) then
        echo ""
        echo "HEY!"
        echo "DONT EVEN THINK ABOUT RUNNING THIS IN YOUR HOME DIR."
        echo "Go create some sub-dir"
        echo "     - A friendly reminder from your local sys-admin"
        echo ""
        exit
endif

setenv CDS_AUTO_64BIT NONE
echo "Setting up environment for $VEND $KIT $VER ... "
source $CMC_HOME/scripts/cadence.ic06.16.110.csh
source $CMC_HOME/scripts/cadence.mmsim15.10.801.csh
source $CMC_HOME/scripts/synopsys.hspice.2014.09.csh
source $CMC_HOME/scripts/cadence.incisive14.20.023.csh
source $CMC_HOME/scripts/mentor.calibre.2017.4_19.14.csh

#set CMC_TECH=${CMC_HOME}/kits/${KIT}.${VER}
set CMC_TECH=${CMC_HOME}/kits/${KIT}.${VER}_${VERCDS}
set CMC_TECH_LIB=${CMC_HOME}/kits/${KIT_LIB}
#set CMC_TECH_LIB=${CMC_HOME}/kits/${KIT_LIB}
#set CMC_TECH=/fab/fabwork/p18kit2018/IC6/test1/cmosp18.6.13a_ic6
```

Fig. 25 Change of the script

After the change of the script, close cadence and do the following command in the terminal within the eecs3611 directory

source kit.tsmc180nm.6.13a_ic6.csh

This will source the changed script rather than the original one provided by CMC and you can start cadence again.

****

The results at typical/typical corner are shown in Fig. 26. From the DC response result, we can see the maximum slope is almost at vdc=1V which means the largest small signal gain appears at this point, that is the reason why I set the

vdc=1 in the ADE variable value. Generally the common mode of input signal is set as half of the power source. From the AC response we can find the gain is 0.524 which is the expected results of a common source stage amp, the gain of which should be less than 1. However the transient response tells us the output swing is saturated due to the too high common mode of output (DC component of your output). This issue is caused by the improper value of load resistor and the regulating resistor. We can decrease the load resistor R1 and regulating resistor R2 to increase the output swing.

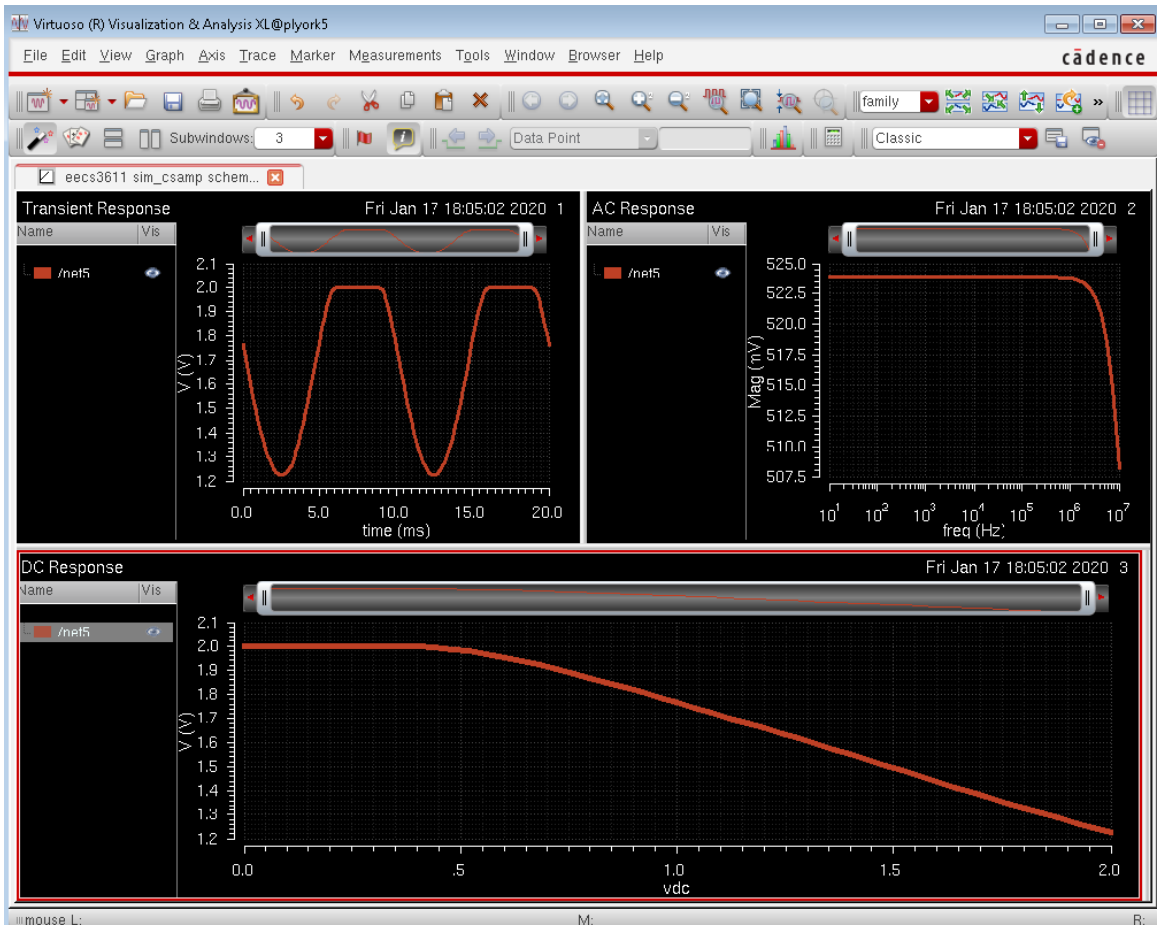


Fig. 26 Simulation results at first try

Actually, the common mode is decided by the following equation:

$$\mathrm{I} = \frac{1}{2}\mu_n C_{ox}\frac{W}{L}(V_{GS} - V_{th})^2$$

$$V_{GS} = V_{DC} - \mathrm{I}R_2$$

$$V_{OUT,CM} = V_{DD} - \mathrm{I}R_1$$

In the tsmc180nm model file (/CMC/kits/cmosp18.6.13a_ic6/models/spectre/cmn018_assp_v1d3.scs), you can find the effective gate oxide thickness T_{ox} is 3.98nm. So we can calculate the C_{ox} by

$$C_{ox} = \frac{\varepsilon_{ox}}{T_{ox}} = \frac{11.9 \times 8.85 \times 10^{-12}}{3.98 \times 10^{-9}} F$$

And the mobility for NMOS in tt corner is found as

$$\mu_n = 305cm^2/(V \cdot s)$$

Also from the PDK specification we can find, the recommended Vdd of our device is 1.8V. So we should modify the variable value vdd and vdc accordingly to 1.8 and 0.9.

During this simulation, NMOS should be operating in saturation region. We can judge it by:

ADE->Results->Print->DC operating point->press e->select amp->select NMOS in our circuit.

We can get the DC operating point like:

vds	1.49255
vdsat	204.683m
vdss	204.683m
vearly	7.48301
vgb	1.1
vgd	-546.272m
vgs	946.273m
vgt	371.296m
vsat_marg	1.28786
vsb	153.727m
vth	574.977m

Fig. 27 Operating points for NMOS

As the equation $V_{ds} \geq V_{gs} - V_{th}$ valid, NMOS is in saturation region.

Finally when we can set vdd=1.8V, vdc=1.1V, sinusoidal amplituide=0.5V, R1=1K and R2=1K, the results are shown in Fig. 28. In order to get the same results at a lower vdc value, you should increase the width of NMOS which will also increase the gain.

To change the y-axis from voltage to dB, select AC response curve, through Tools->Calculator->select wave->select dB20 from the Function panel->evaluate the buffer, you will get the Bode plot of the output signal. Calculator panel is shown in Fig. 29.

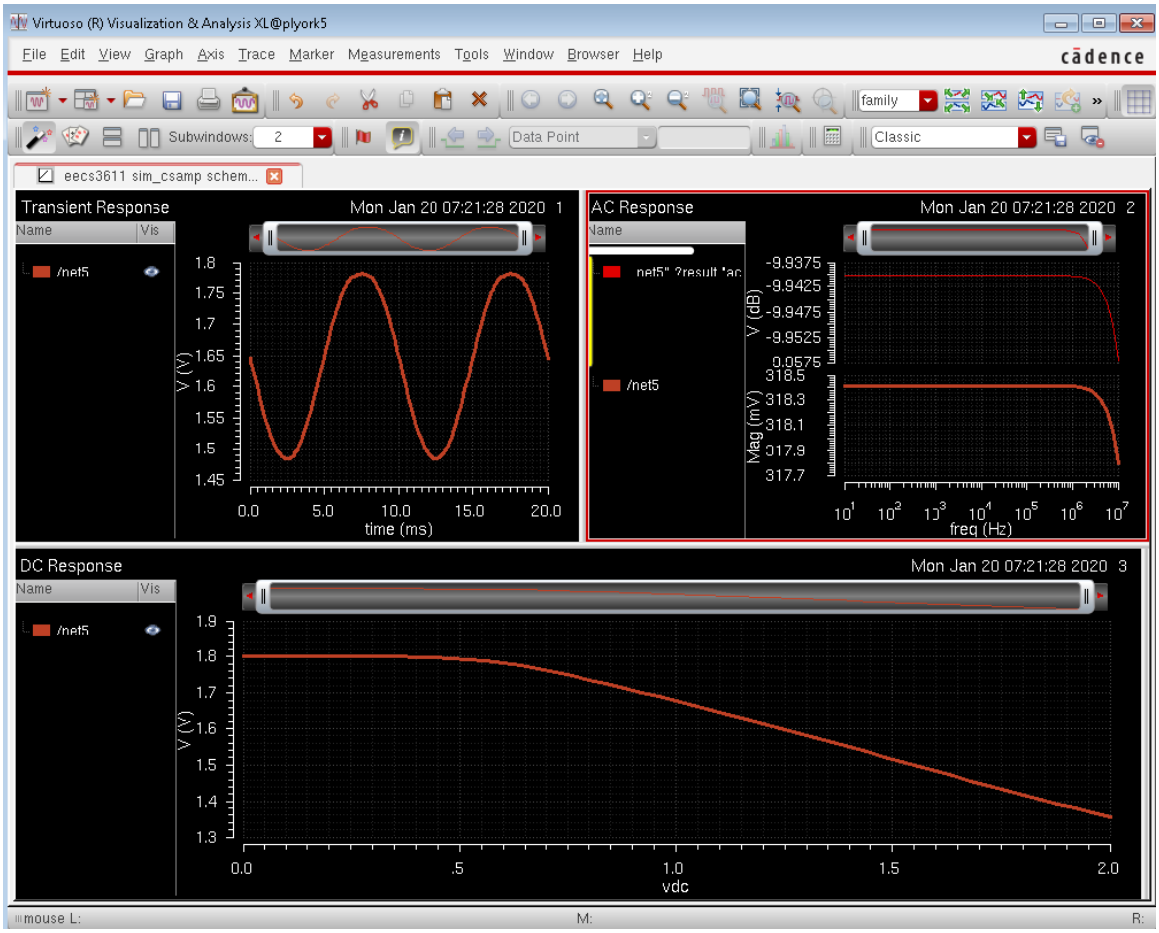


Fig. 28. Final simulation results at tt corner.

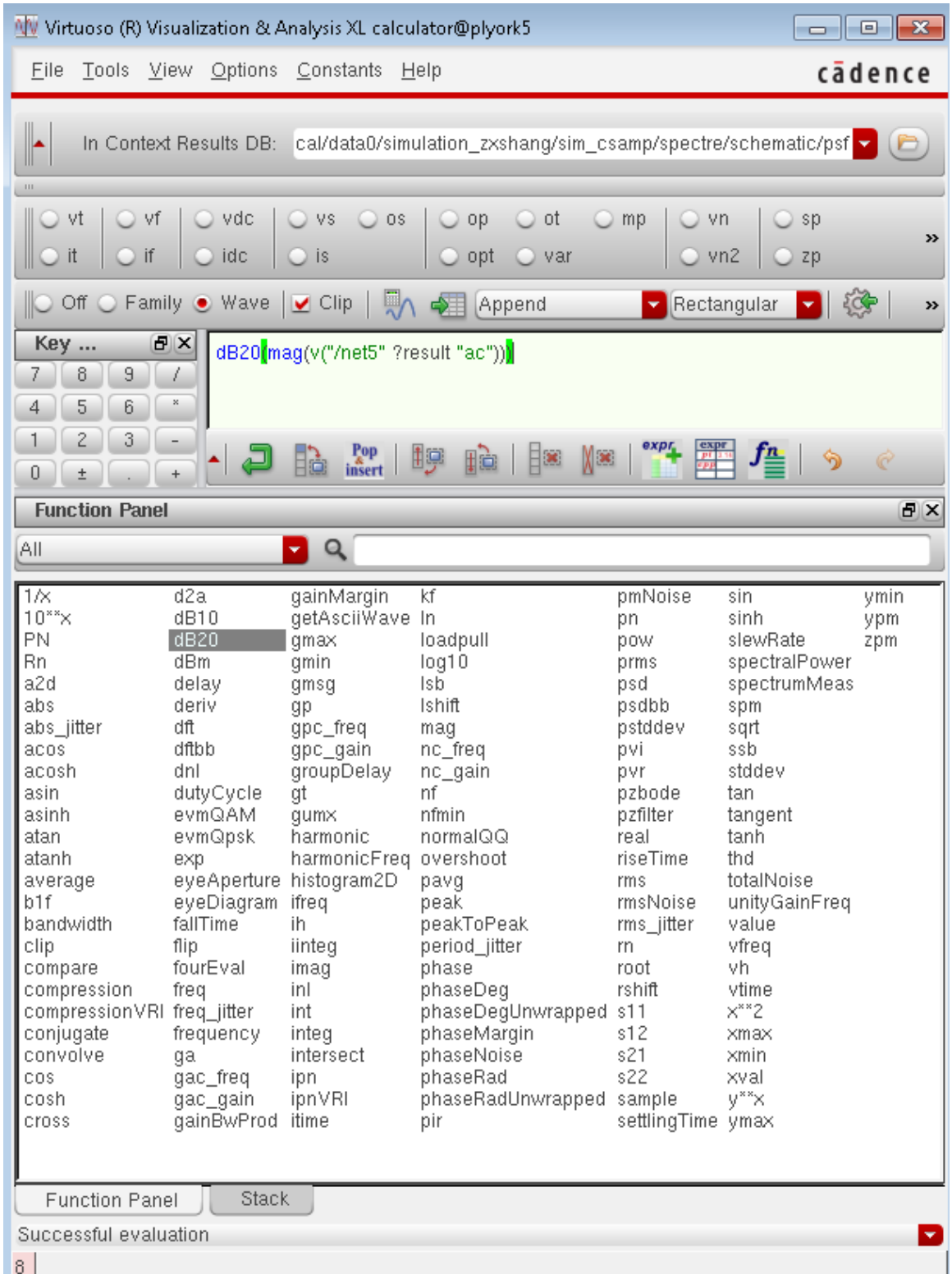


Fig. 29. Calculator panel.