Galaxy Custom Designer – Quick Startup

- Schematic Editor
- Simulation and Analysis Environment

- CosmoScope
- HSPICE Toolbox for MATLAB

1. Setting Up the Workspace

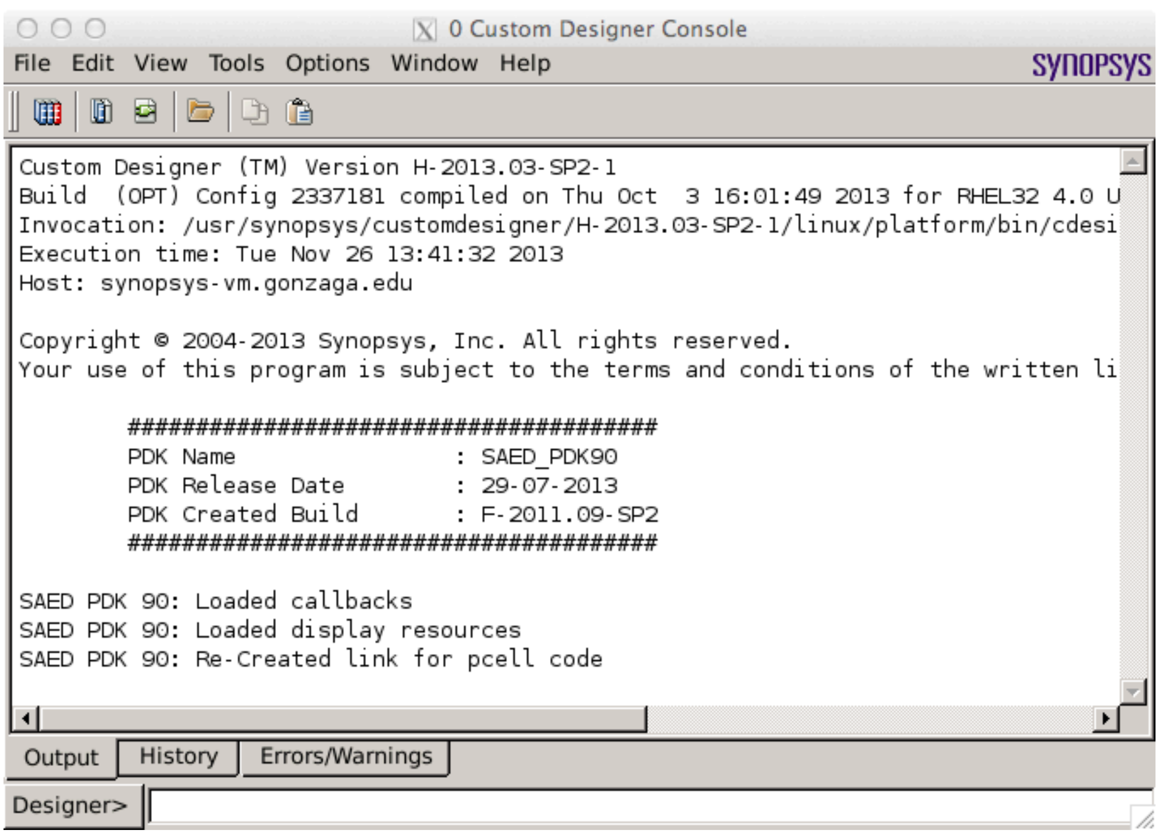
Create a working directory
mkdir ~/ee406/s14/analog/

Enter the workng directory:
cd ~/ee406/s14/analog/

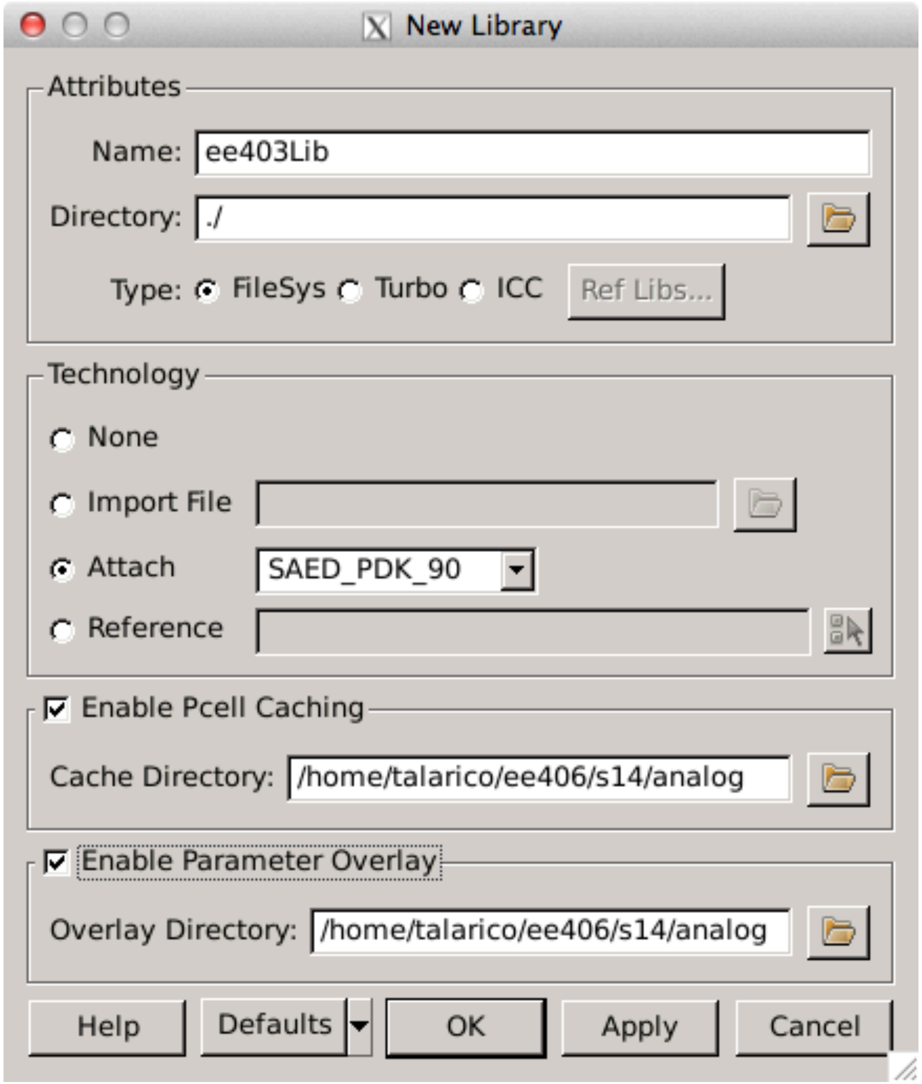
Setup the libraries by creating a file lib.defs in the working directory
The file should look as follows:
INCLUDE $SYNOPSYS_CUSTOM_INSTALL/samples/lib.defs
DEFINE SAED_PDK_90 /usr/synopsys/SAED_PDK90nm/SAED_PDK_90/
DEFINE reference /usr/synopsys/SAED_PDK90nm/reference

2. Running custom designer:

From the working directory:
cdesigner &

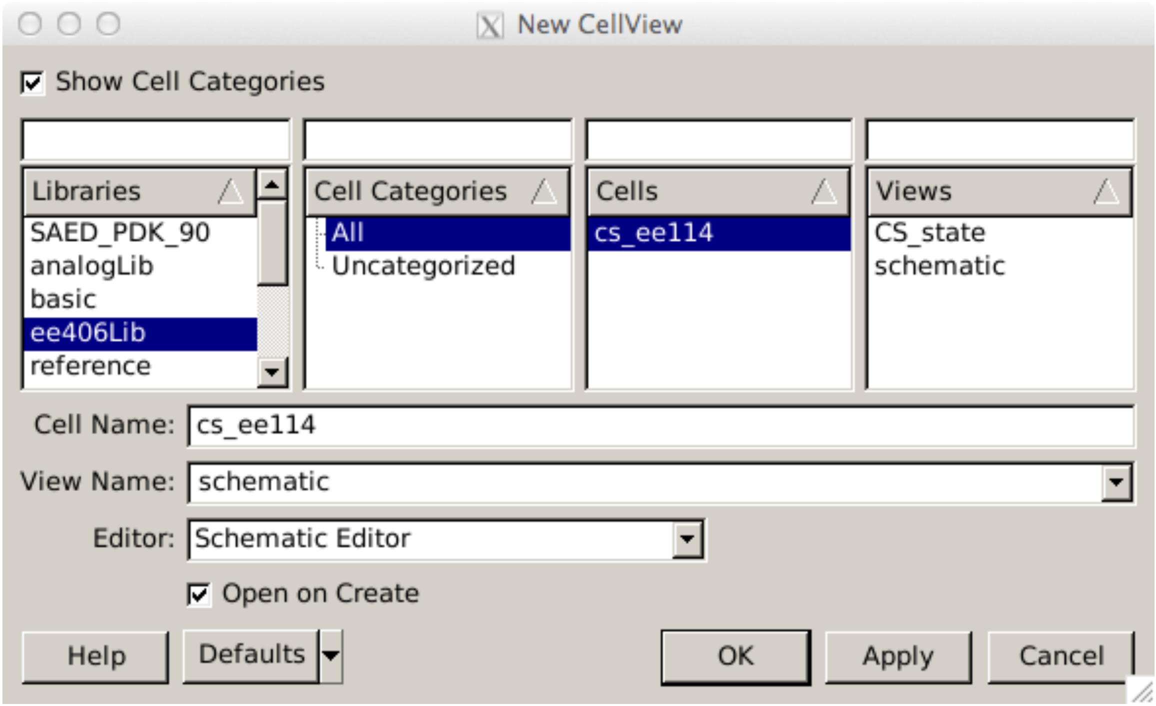


3. Creating a new Library
From the custom designer console:
File → New → Library

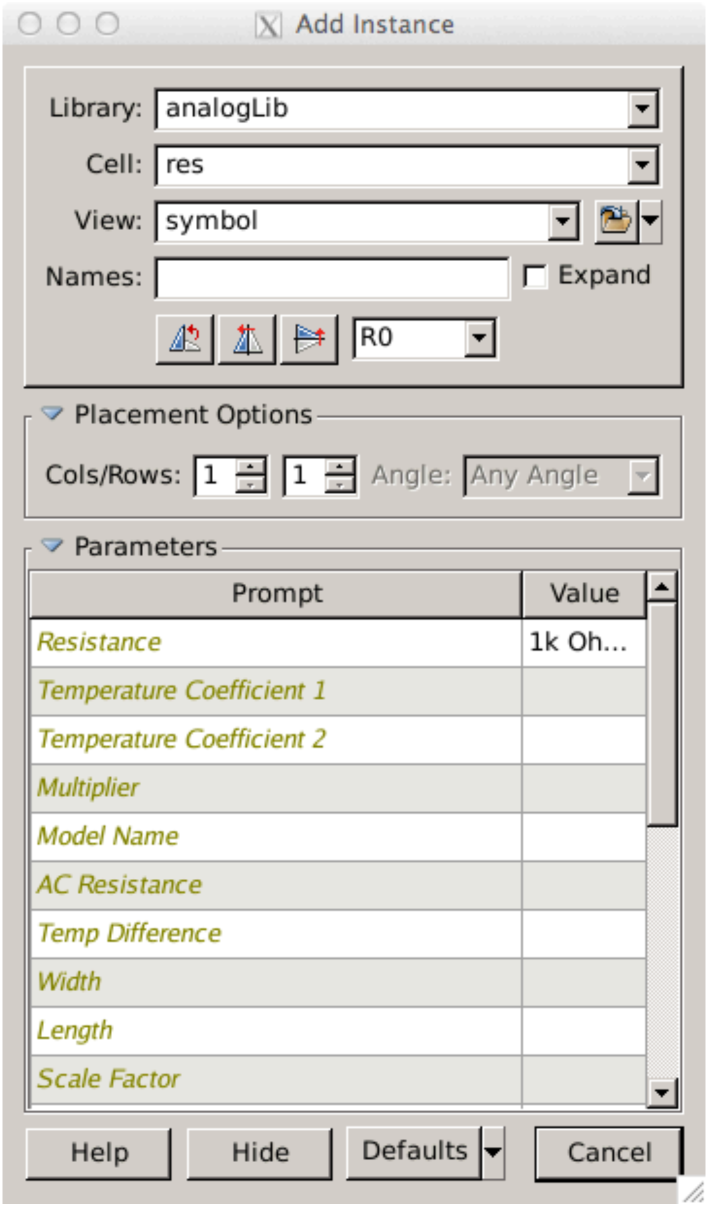


4. Creating a New Cell

From the custom designer console
File → New → Cell



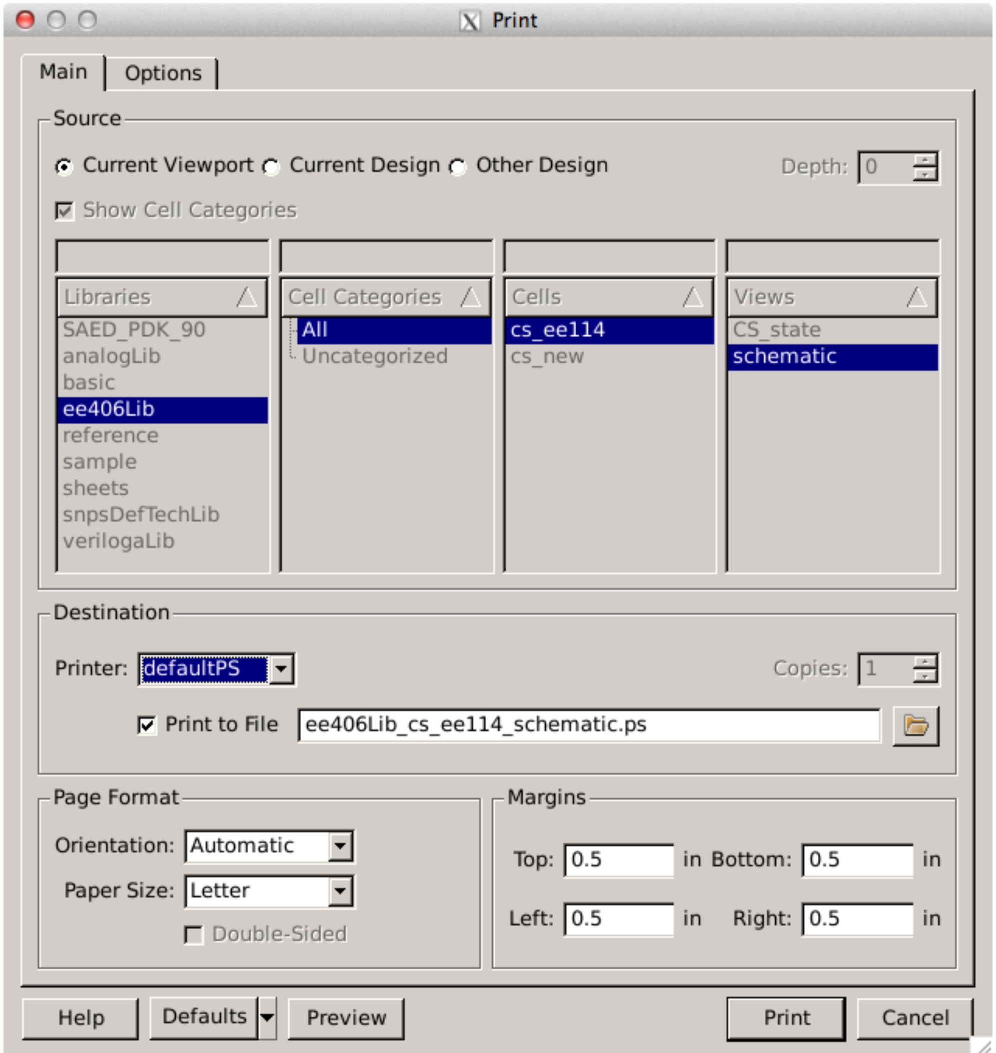
The schematic editor (SE) opens.
Enter the schematic of the design, by adding the various instances, wires, wire names, etc.:

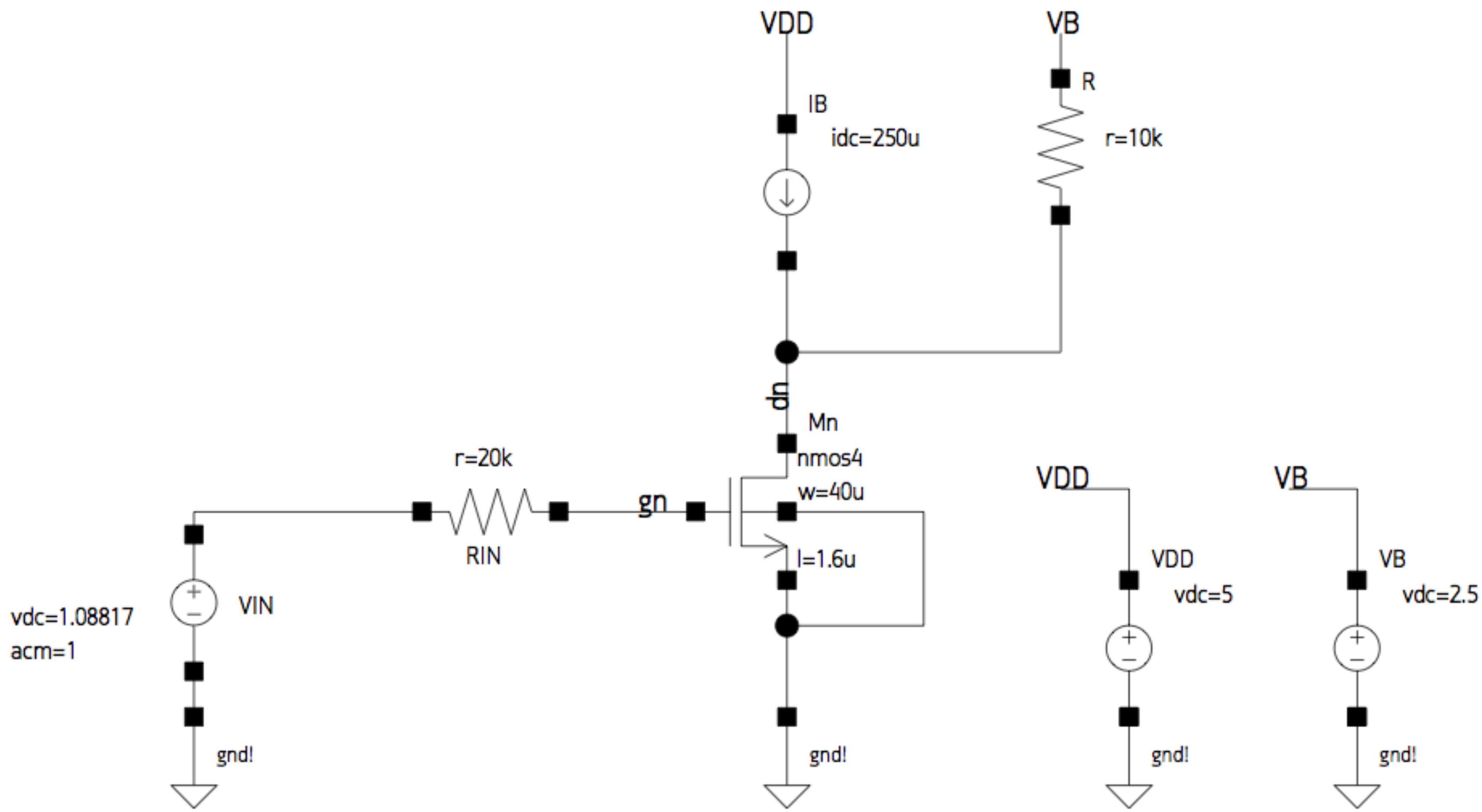


Save the design (Design → Save) often and make sure to check it for possible errors (Design → Check and Save).

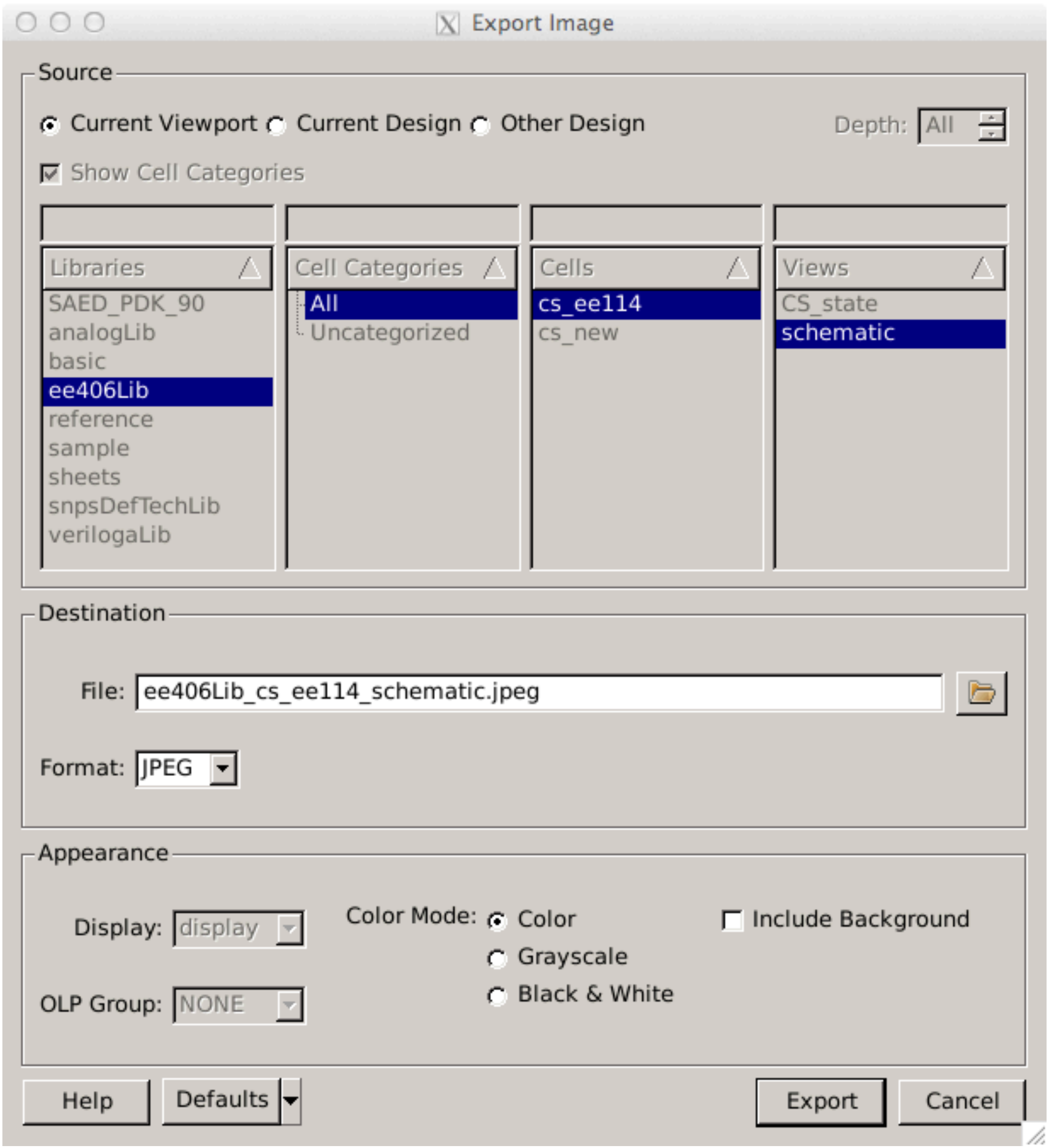
Once the schematic is finished and passes all checks (see the custom explorer console) it can be printed, exported as an image, exported as a netlist, etc.

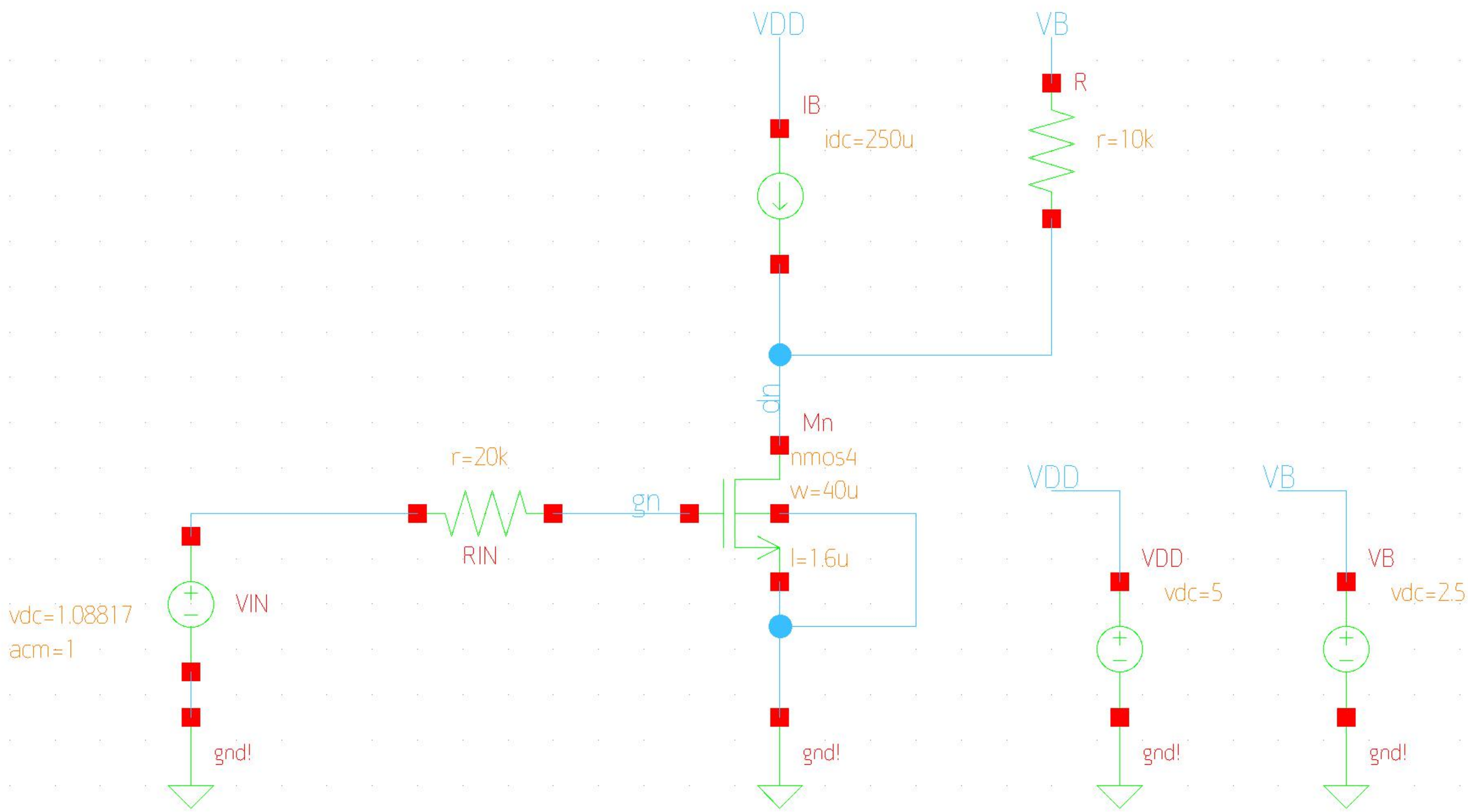
Design → Print Submit (print to file)



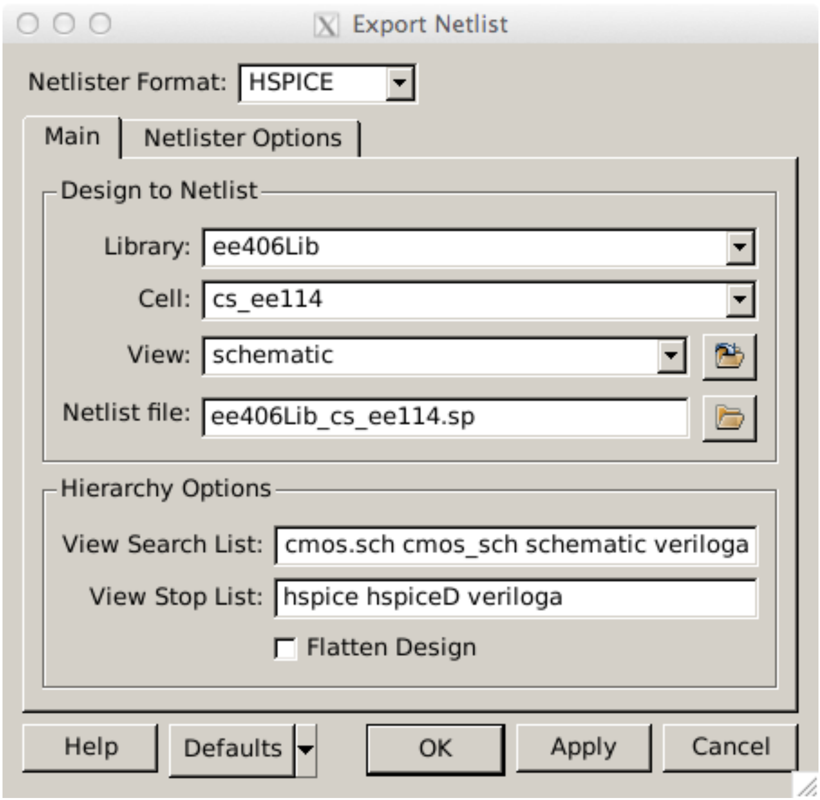

VDD
VB
IB
idc=250u
R
r=10k
dn
Mn
nmos4
w=40u
l=1.6u
r=20k
RIN
gn
vdc=1.08817
acm=1
VIN
gnd!
gnd!
VDD
VDD
vdc=5
gnd!
VB
VB
vdc=2.5
gnd!

Design → Export Image (format jpeg; do not include background)




VDD
VB
IB
idc=250u
R
r=10k
dn
Mn
nmos4
w=40u
l=1.6u
r=20k
RIN
gn
VIN
vdc=1.08817
acm=1
gnd!
gnd!
VDD
VB
VDD
vdc=5
VB
vdc=2.5
gnd!
gnd!

Design → Export Netlist



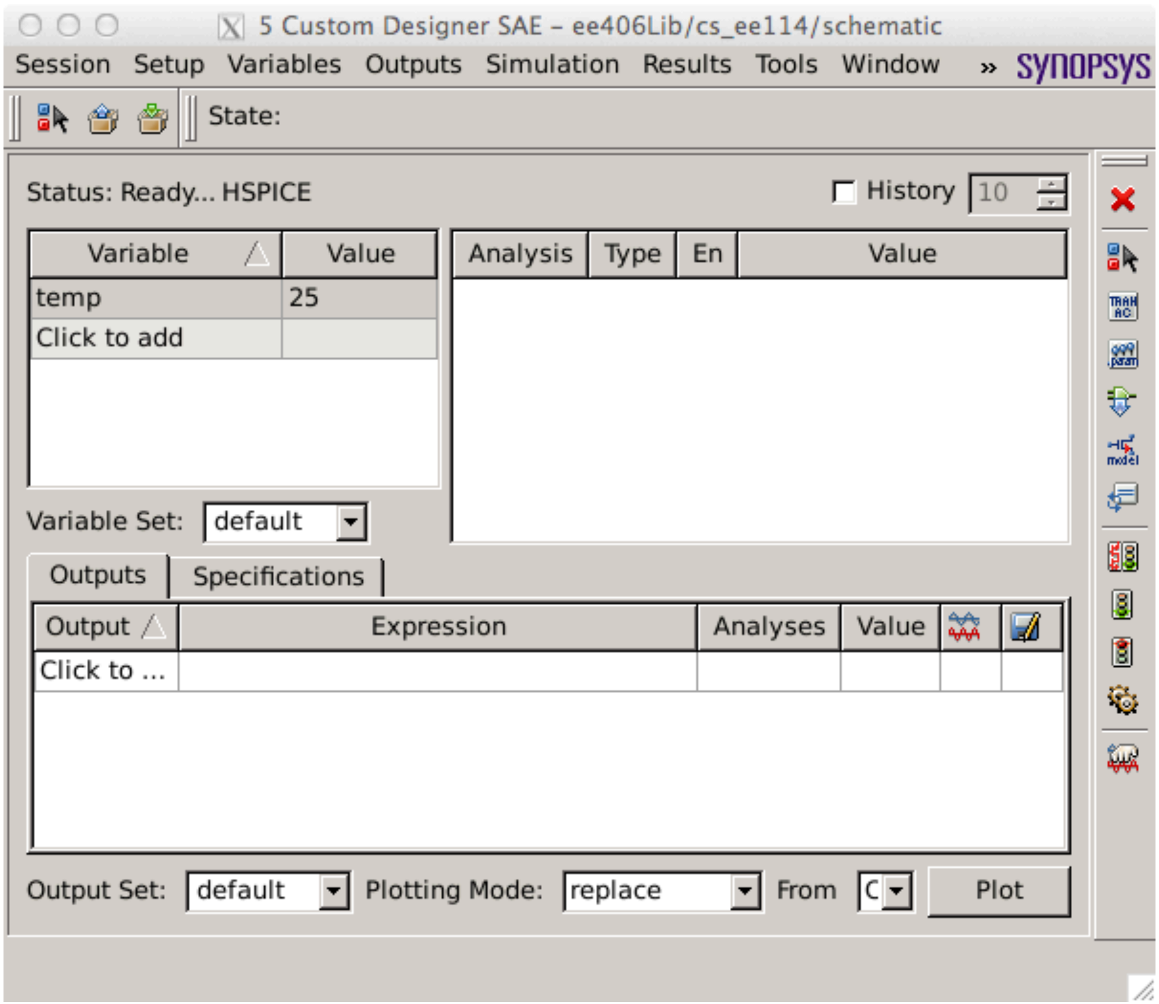
cat ee406Lib_cs_ee114.sp

```
*Custom Designer (TM) Version H-2013.03-SP2-1
*Mon Nov 25 10:58:00 2013

.GLOBAL gnd!
********************************************************************************
* Library          : ee406Lib
* Cell             : cs_ee114
* View             : schematic
* View Search List : hspice hspiceD cmos.sch cmos_sch schematic veriloga
* View Stop List   : hspice hspiceD veriloga
********************************************************************************
.subckt cs_ee114
ib vdd dn dc=250u
rin gn in r=20k
r vb dn r=10k
mn dn gn gnd! gnd! nmos4 w=40u l=1.6u
vb vb gnd! dc=2.5
vdd vdd gnd! dc=5
vin in gnd! dc=1.08817 ac=1
.ends cs_ee114
```

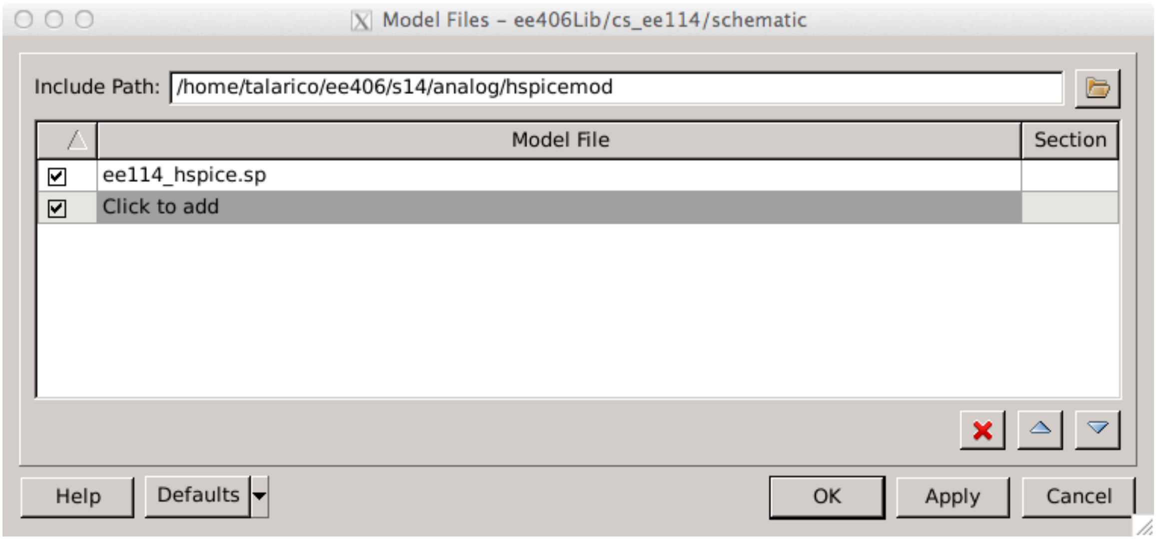
5. Running the Simulation and Analysis Environment (SAE)

From the schematic Editor invoke Tools → SAE.
The SAE window opens.



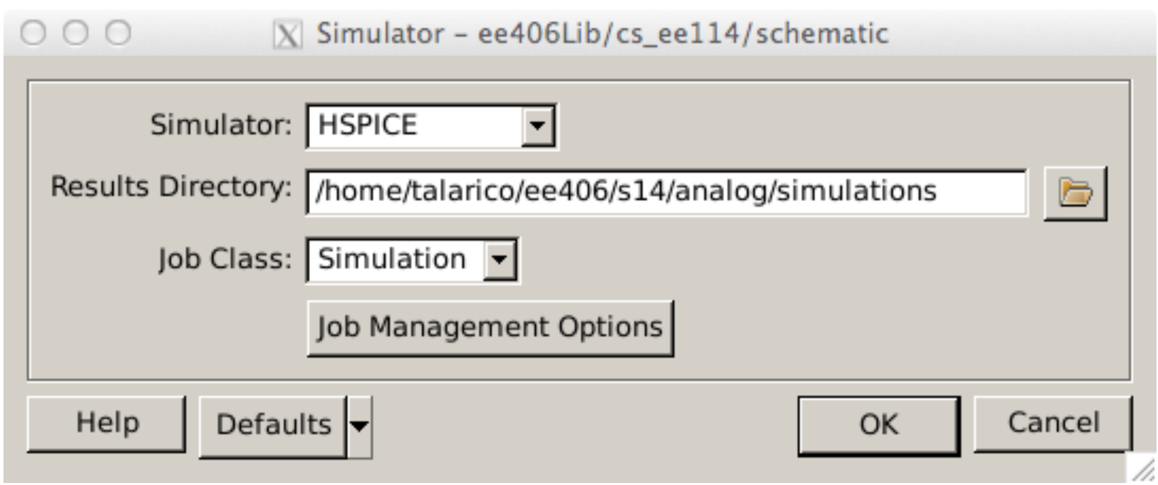
Set up the model files to be used by the simulator

Setup → Model files



Setup where to save the simulation:

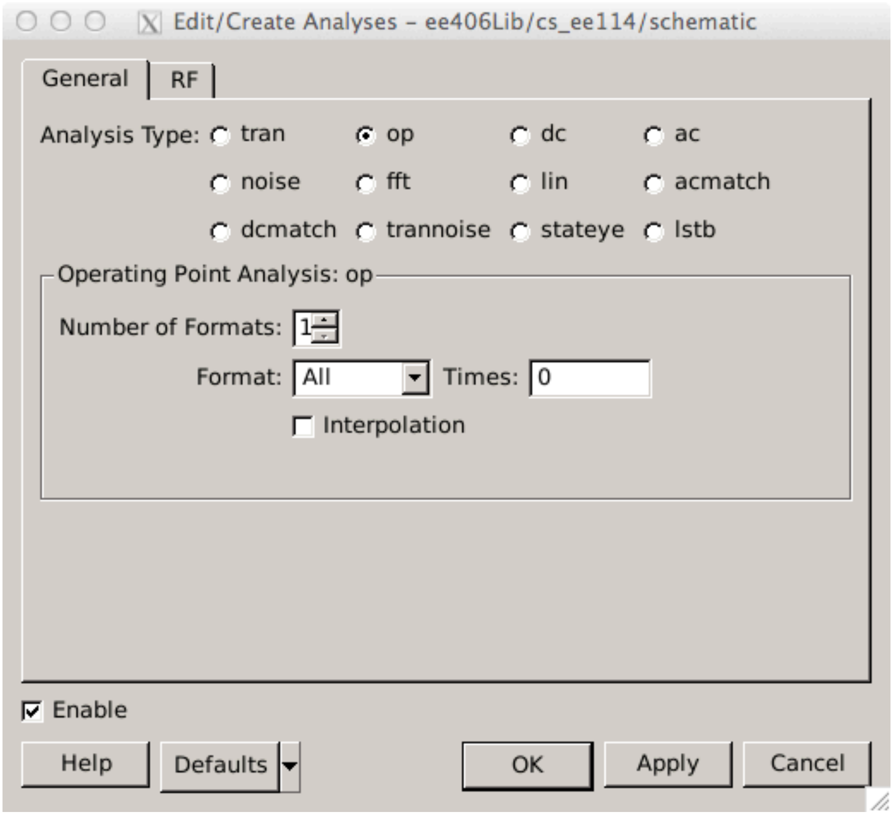
Setup → Simulator

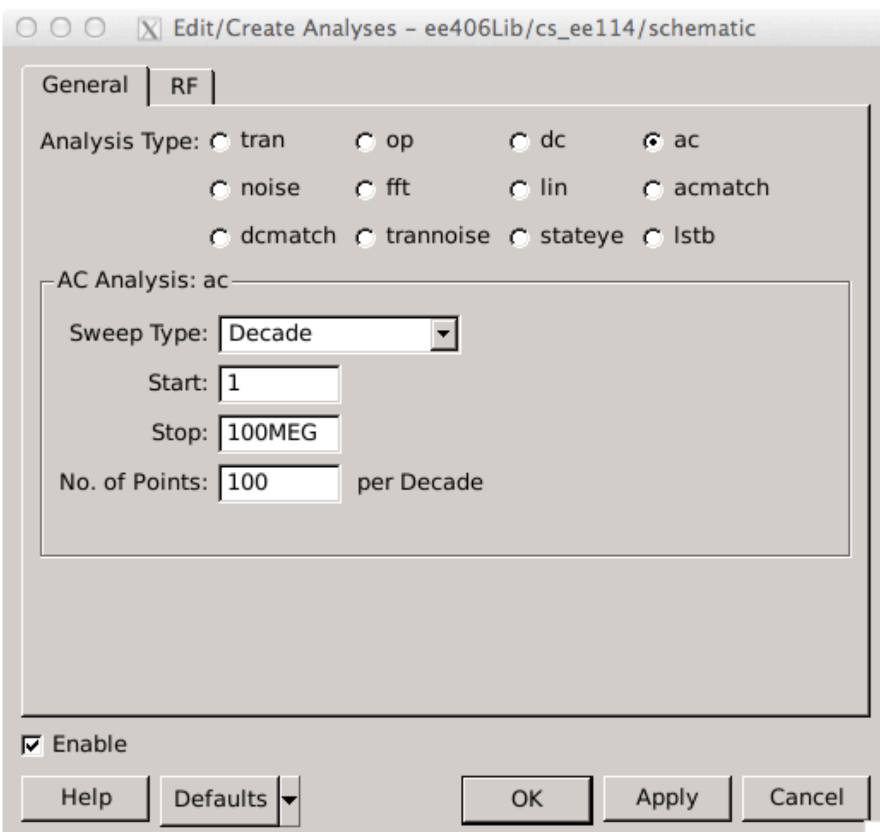


Save the location of the result directory in simu.xml (Defaults → Save)

Setup the analysis to run:

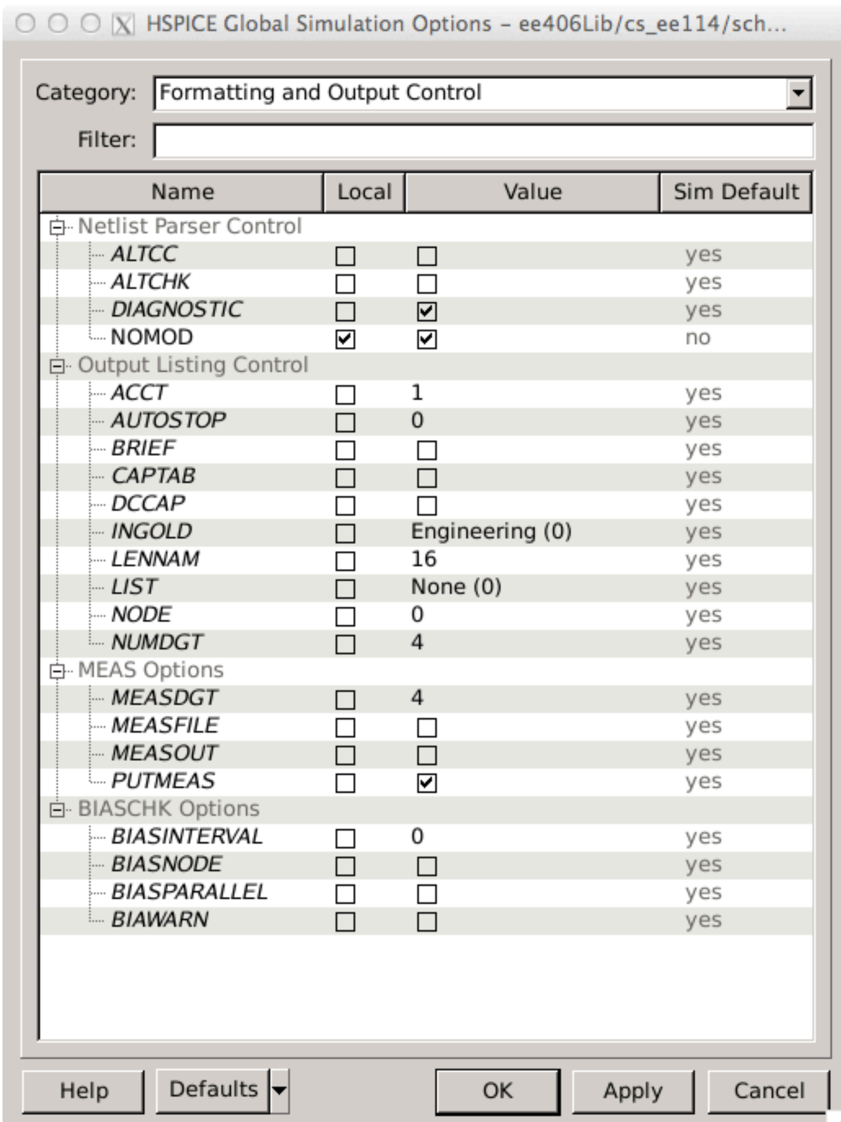
Setup → Analysis


Edit/Create Analyses - ee406Lib/cs_ee114/schematic
General
RF
Analysis Type:
tran
op
dc
ac
noise
fft
lin
acmatch
dcmatch
trannoise
stateye
lstb
Operating Point Analysis: op
Number of Formats:
1
Format:
All
Times:
0
Interpolation
Enable
Help
Defaults
OK
Apply
Cancel


Edit/Create Analyses - ee406Lib/cs_ee114/schematic
General
RF
Analysis Type:
tran
op
dc
ac
noise
fft
lin
acmatch
dcmatch
trannoise
stateye
lstb
AC Analysis: ac
Sweep Type:
Decade
Start:
1
Stop:
100MEG
No. of Points:
100
per Decade
Enable
Help
Defaults
OK
Apply
Cancel

Set up your favorite HSPICE options:

Simulation → options

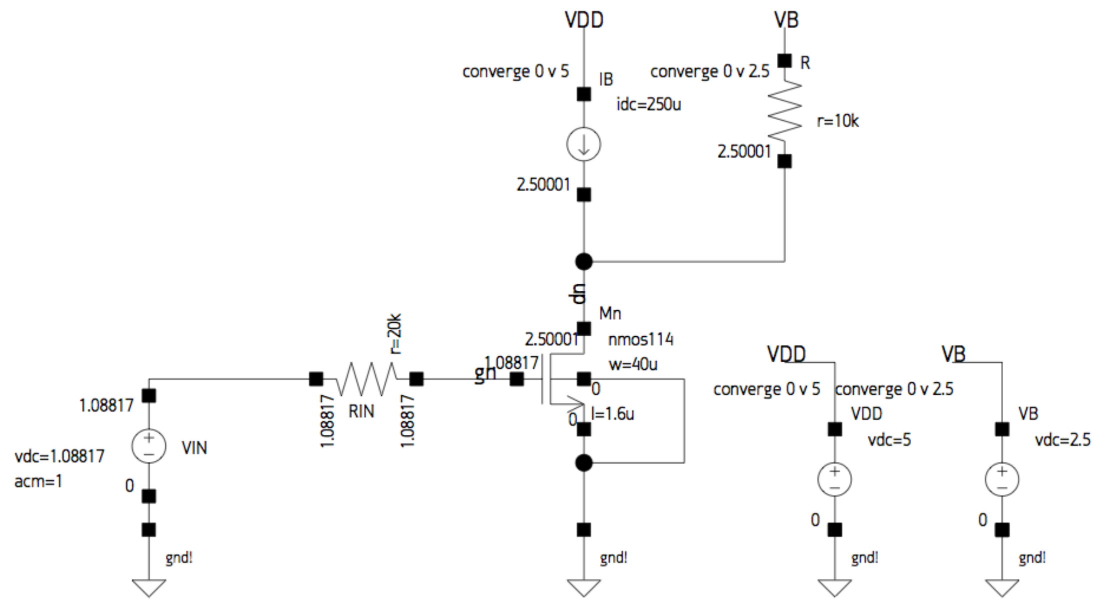


.OPTION NOMOD = 1
This option suppresses the printout of model parameters.

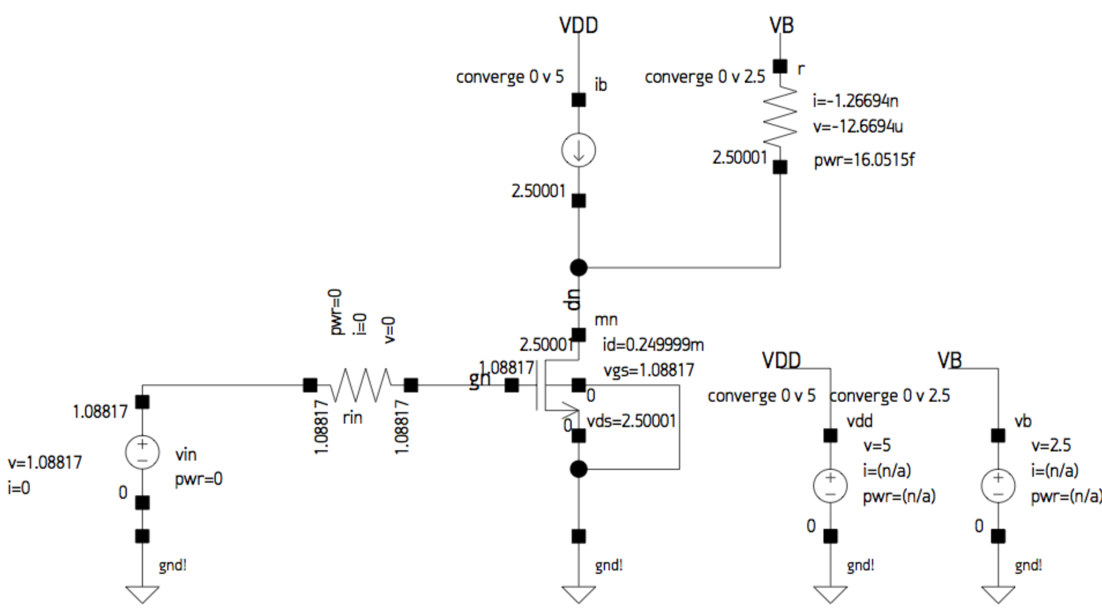
Run the simulation
Simulation → Netlist and Run

Analyzing the results of the simulation:

Annotating the schematic with the DC voltages:
Results → Annotate → DC node Voltages

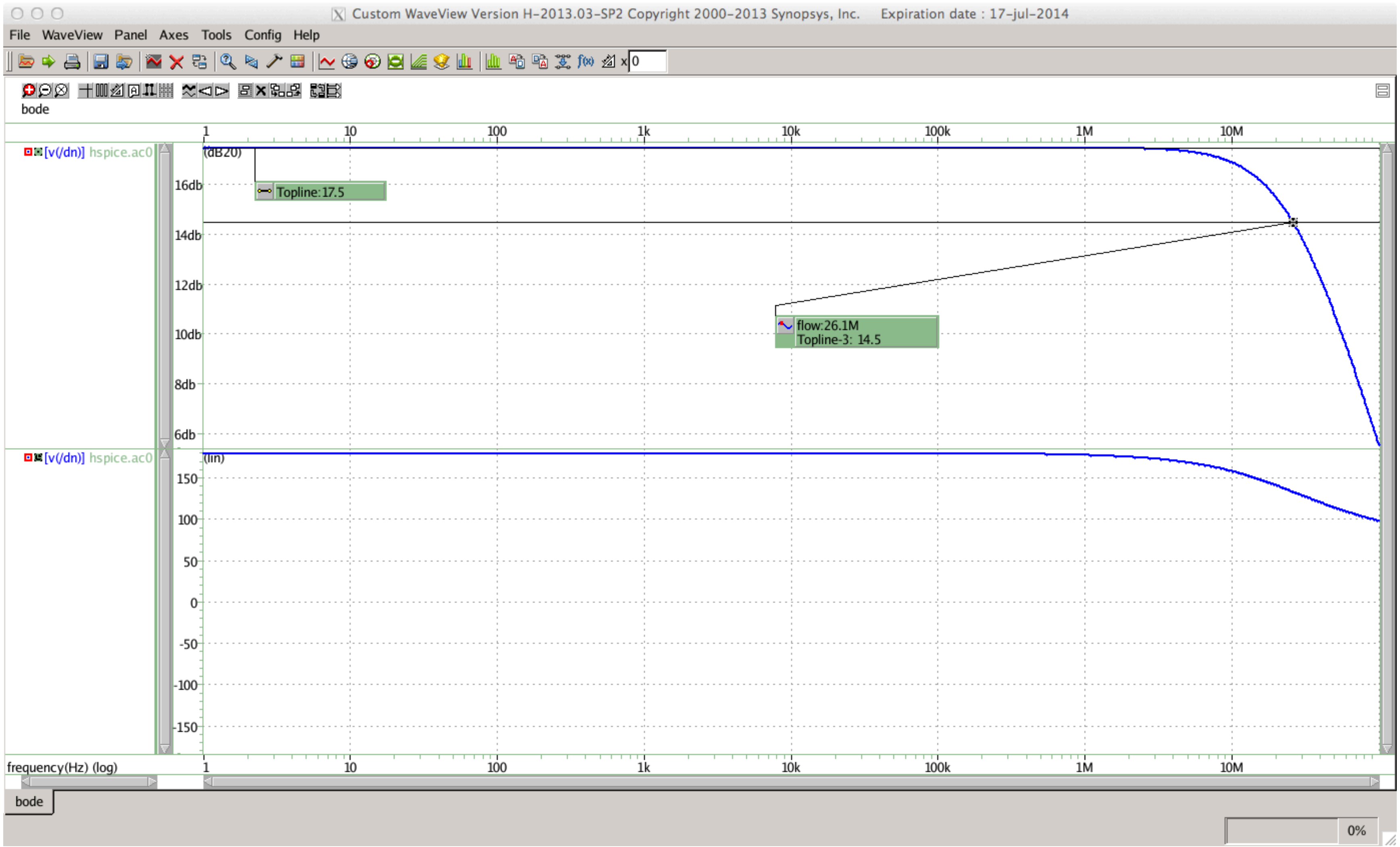


Annotating the schematic with the DC Operating point values:
Results → Annotate → DC Operating Point



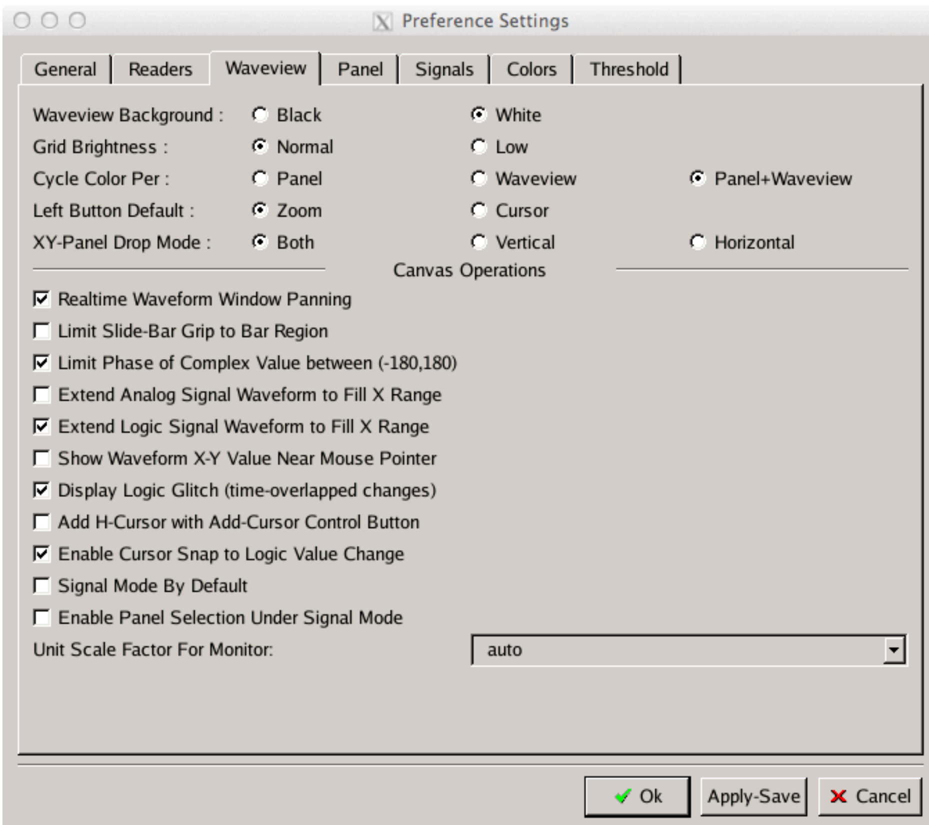
Plotting the frequency response of the amplifier
Results → Plot Signal → AC Bode
Click on the node of interest in the schematic

Custom waveView opens

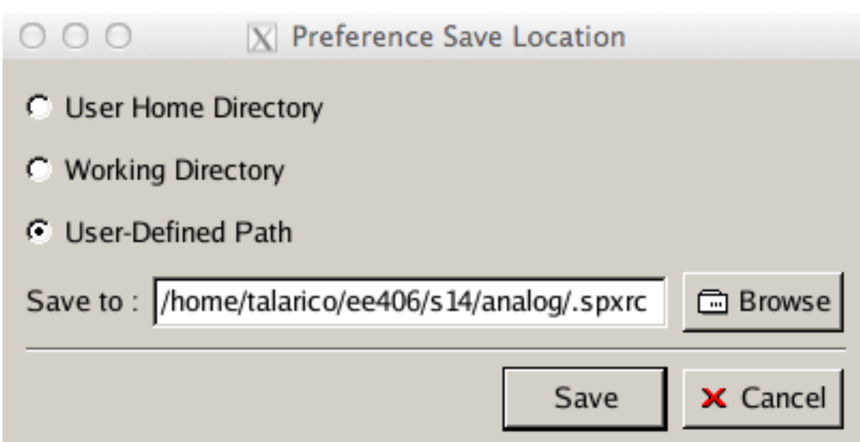

Custom WaveView Version H-2013.03-SP2 Copyright 2000-2013 Synopsys, Inc. Expiration date : 17-jul-2014
File WaveView Panel Axes Tools Config Help
bode
[v(/dn)] hspice.ac0
(dB20)
Topline:17.5
flow:26.1M
Topline-3: 14.5
16db
14db
12db
10db
8db
6db
(lin)
150
100
50
0
-50
-100
-150
frequency(Hz) (log)
1
10
100
1k
10k
100k
1M
10M
0%

To change the default color of the background:

Config → Preferences

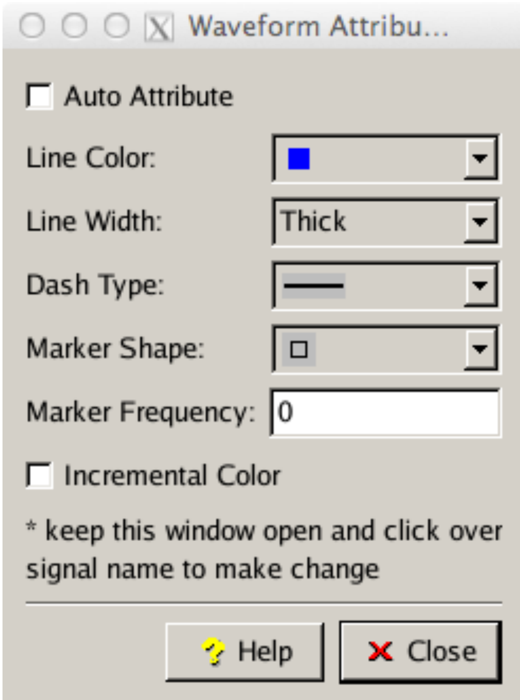


To save the preferences for future use click on Apply-Save:



To change the thickness and color of waveforms:

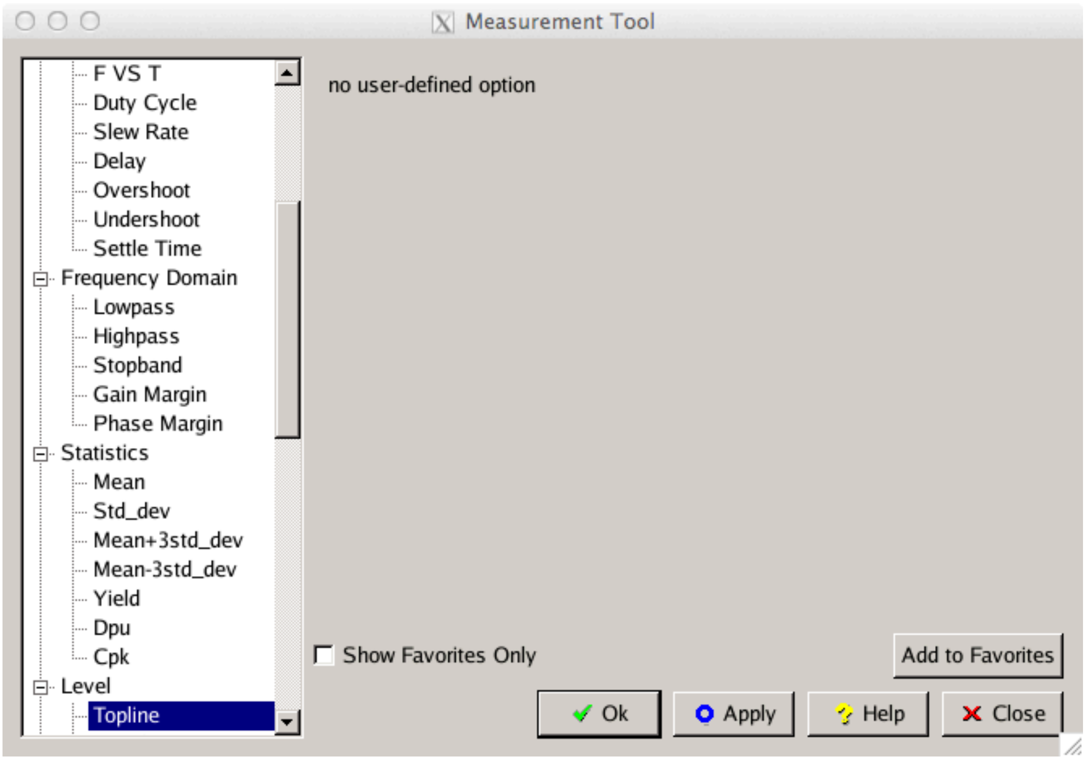
Config → Waveform Attributes

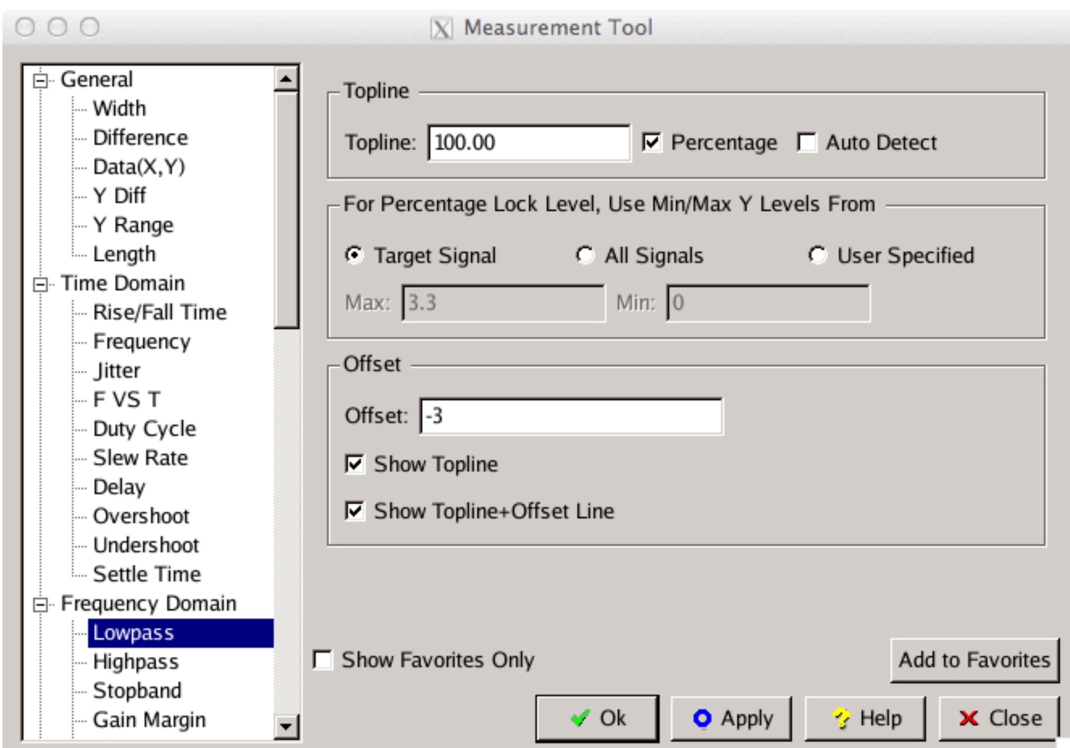


Make sure to click over the signal names to make the change happen.

Measuring the gain and the BW of the designed amplifier:

Tools → Measurements

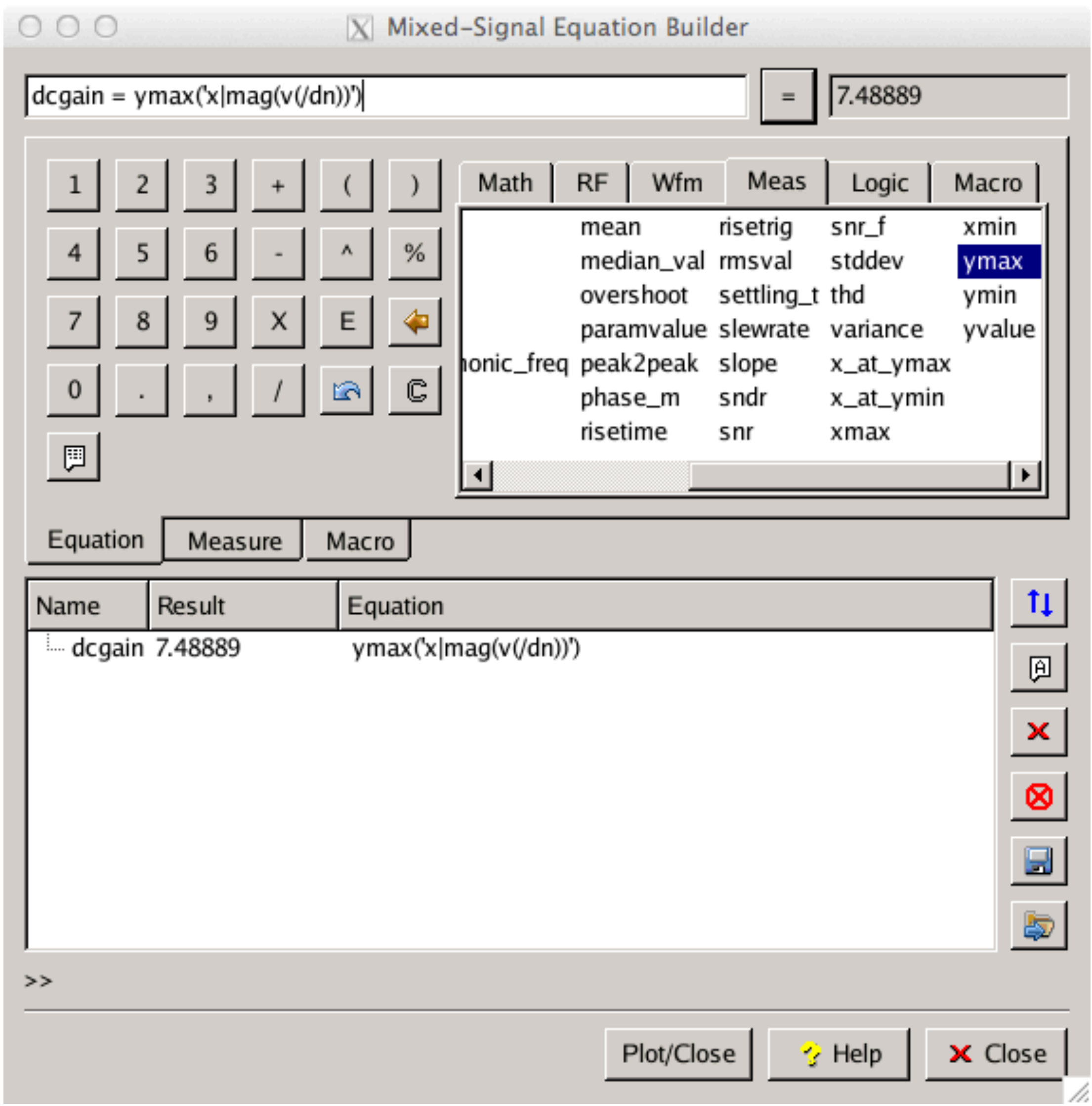


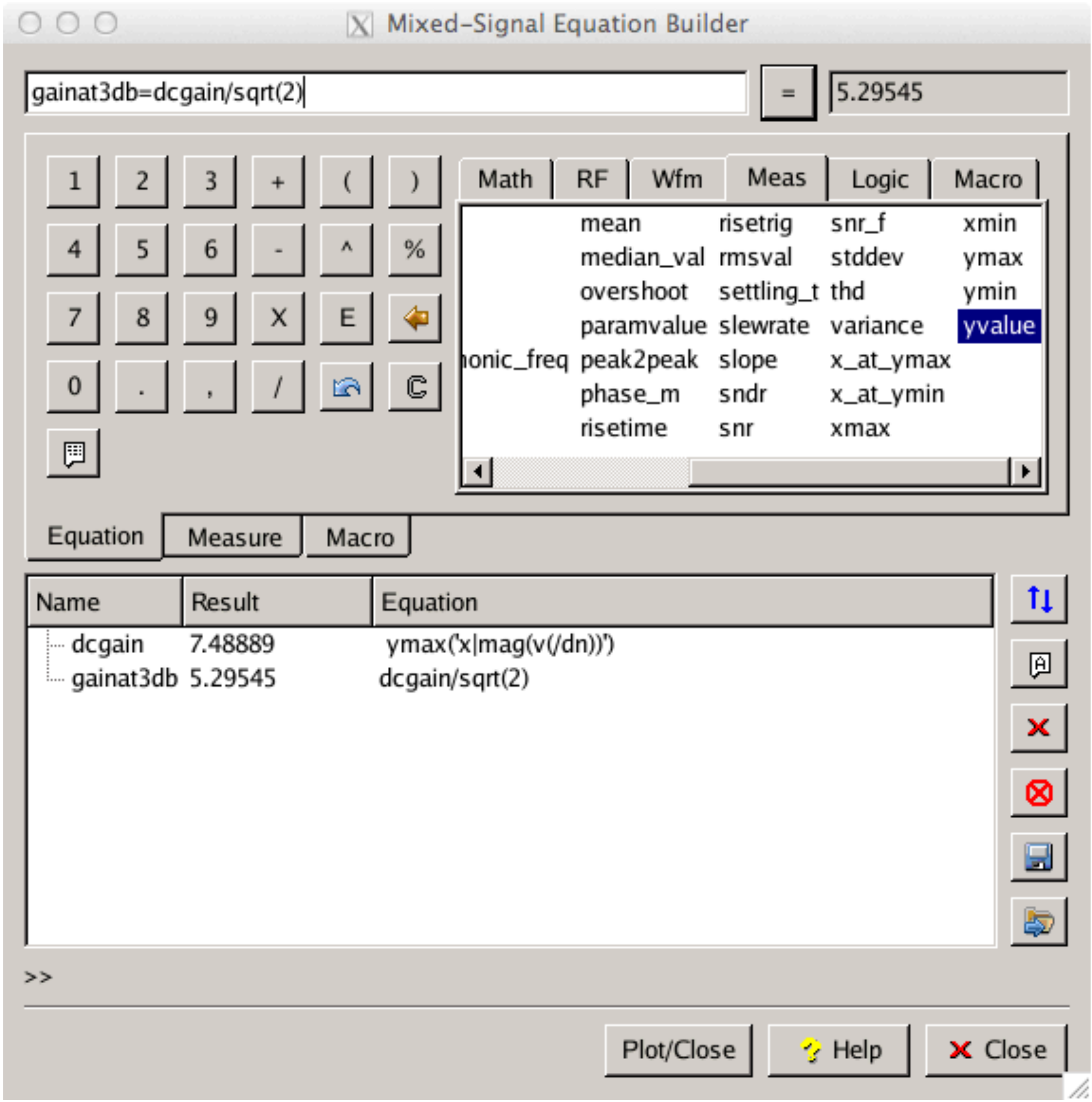


Computing the BW using the equation Builder of Custom Explorer WaveView

Results → Plot Signal → AC Magnitude
Click on the node of interest in the schematic
Custom waveView opens

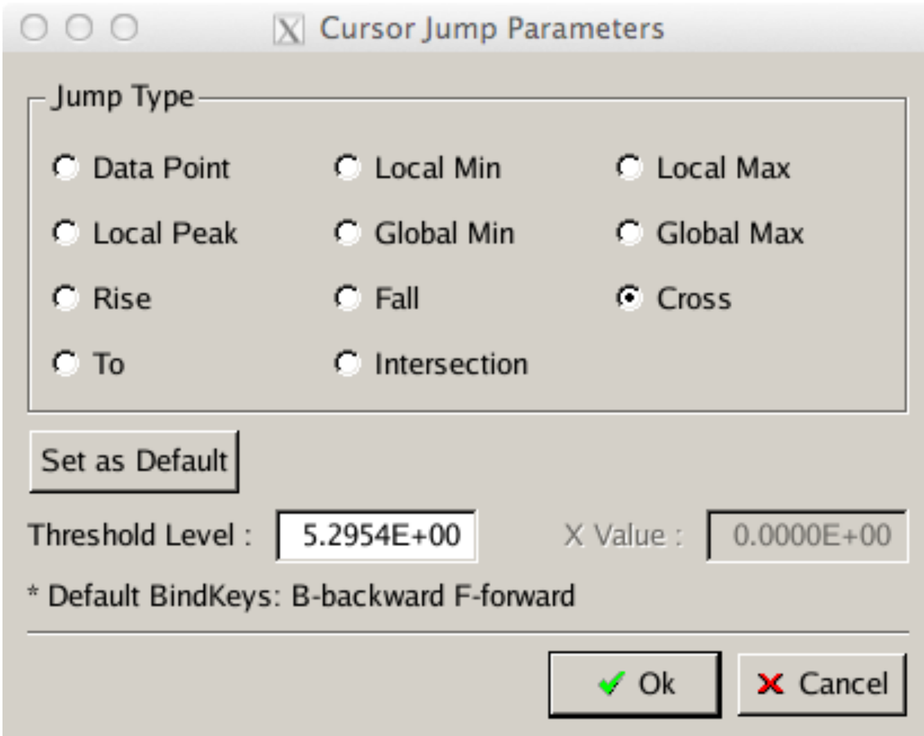
Tools → Equation Builder
Right Click on the signal name → Add to Equation

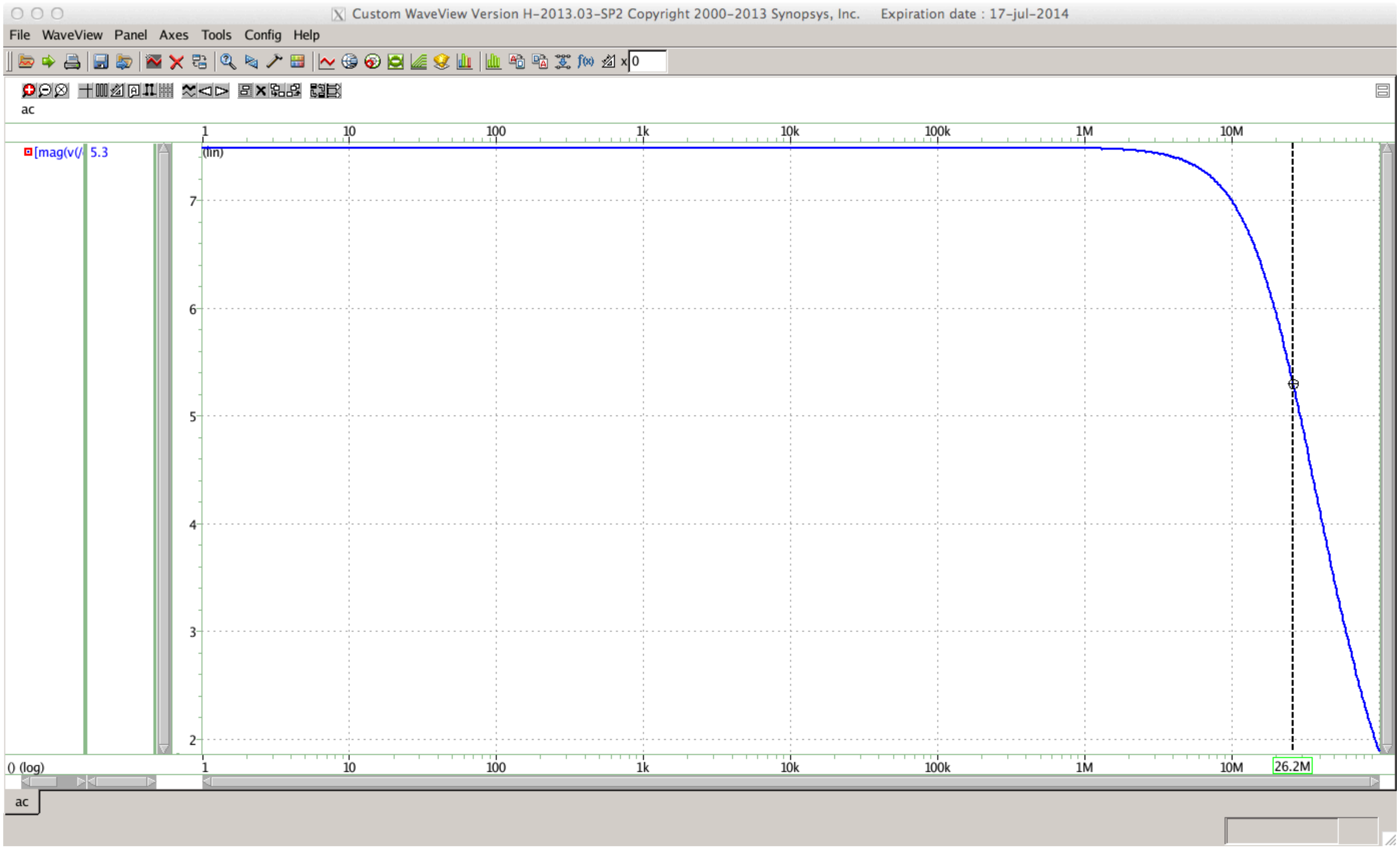




Finally set a cursor at 5.29545 to find out the value of the BW (about 26.2 MHx)

Axes → Cursor → Jump Settings

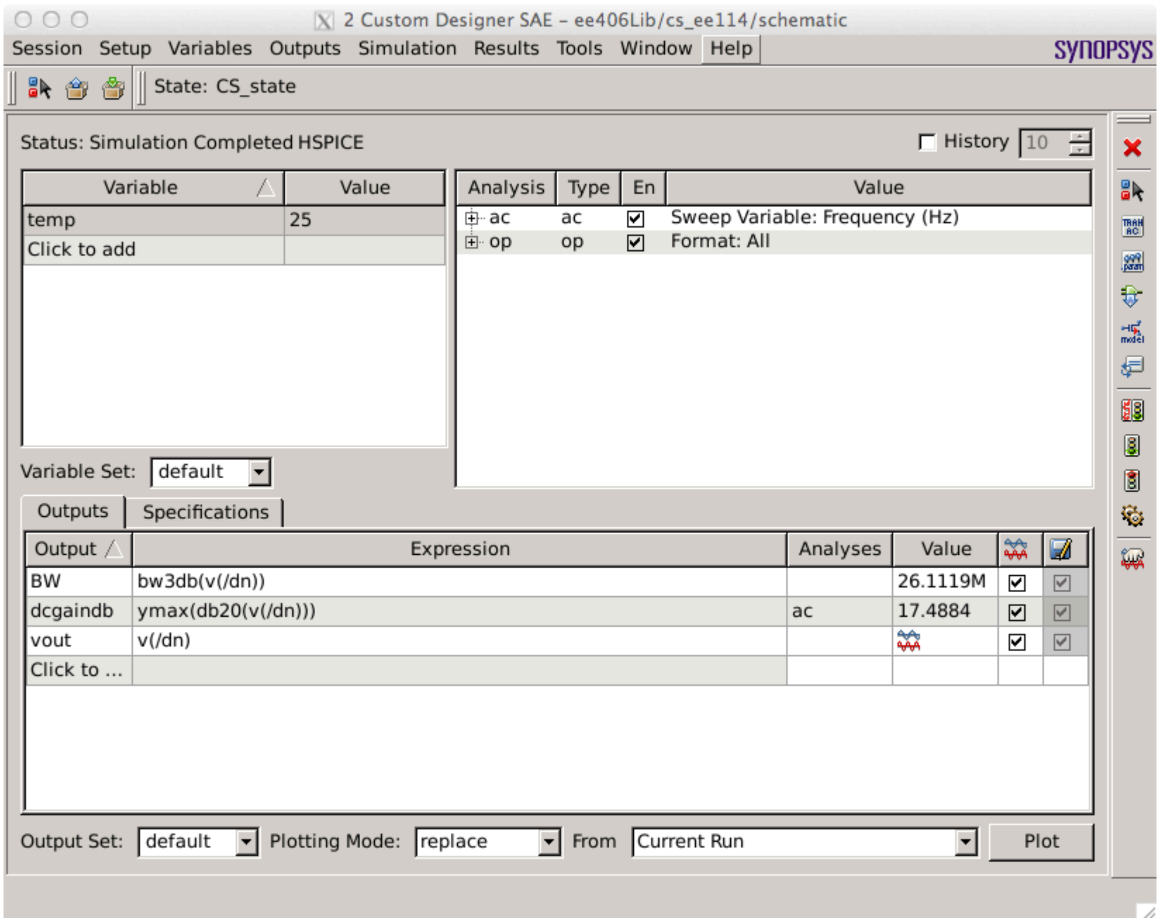

Cursor Jump Parameters
Jump Type
Data Point
Local Min
Local Max
Local Peak
Global Min
Global Max
Rise
Fall
Cross
To
Intersection
Set as Default
Threshold Level :
5.2954E+00
X Value :
0.0000E+00
* Default BindKeys: B-backward F-forward
Ok
Cancel


Custom WaveView Version H-2013.03-SP2 Copyright 2000-2013 Synopsys, Inc. Expiration date : 17-jul-2014
File WaveView Panel Axes Tools Config Help
ac
[mag(v(/ 5.3
(lin)
1 10 100 1k 10k 100k 1M 10M
7
6
5
4
3
2
() (log)
26.2M
ac

Setting up the results to output through the SAE output section

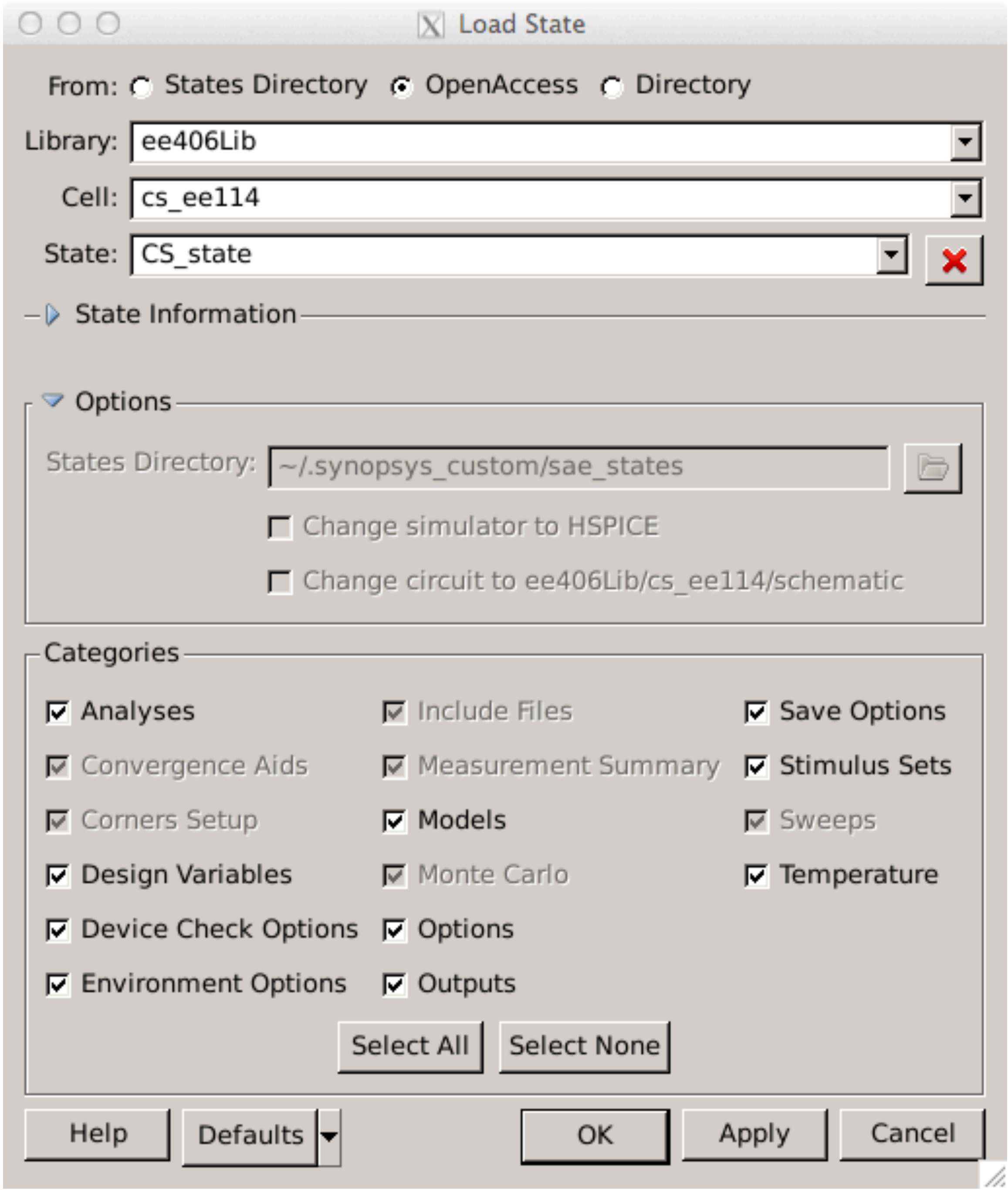
Outputs → Select in design (click on the schematics' net(s) of interest)

Defining names for the observed outputs; creating expressions through the calculator, plotting the results



Saving the simulation and analysis setting

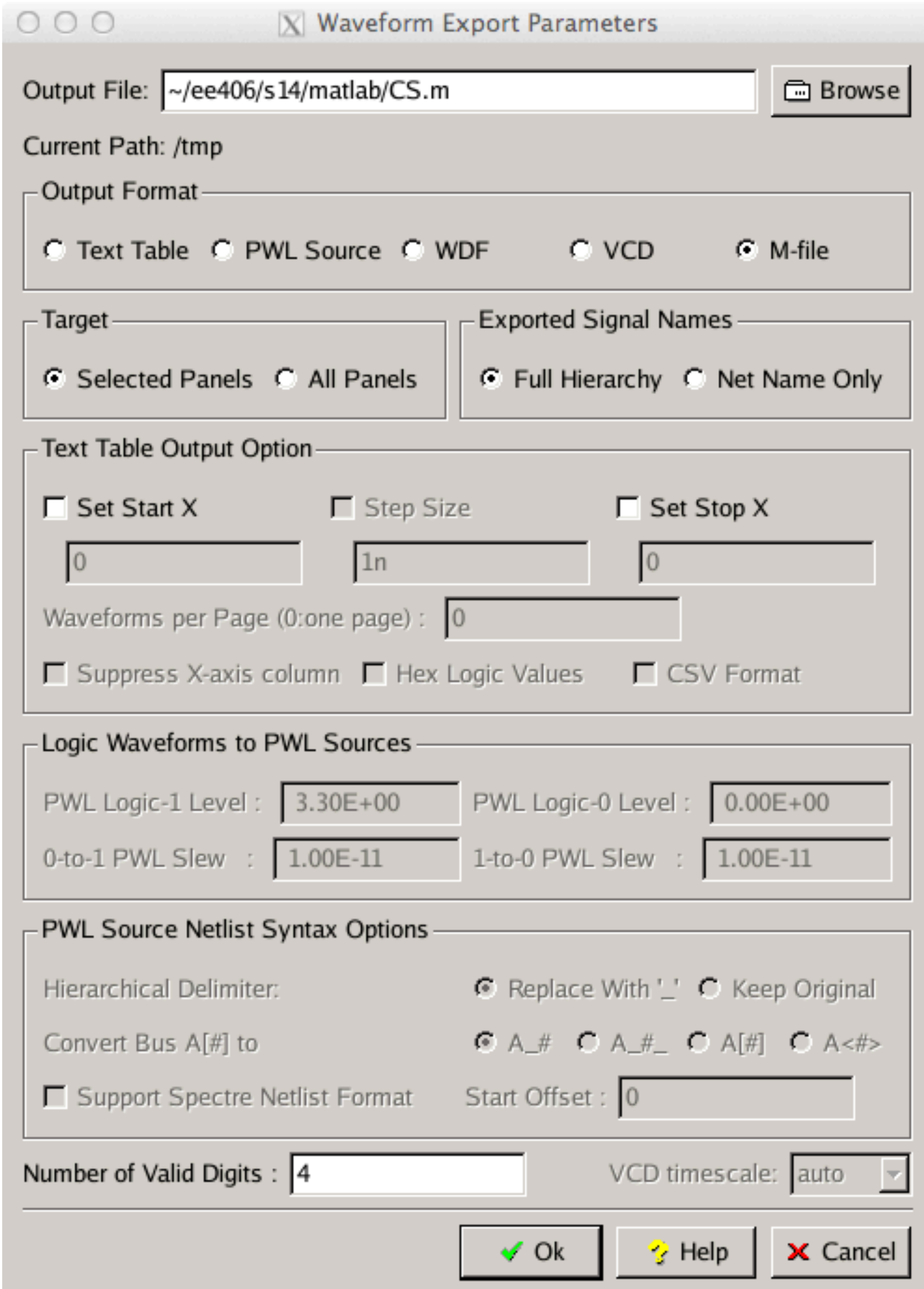
Session → Save State.



Exporting the Waveforms data for MATLAB

Go to the Custom Explorer WaveView window and click on the signals to export.

File → Export Waveform Data


Waveform Export Parameters
Output File: ~/ee406/s14/matlab/CS.m
Browse
Current Path: /tmp
Output Format
Text Table
PWL Source
WDF
VCD
M-file
Target
Selected Panels
All Panels
Exported Signal Names
Full Hierarchy
Net Name Only
Text Table Output Option
Set Start X
Step Size
Set Stop X
0
1n
0
Waveforms per Page (0:one page) :
0
Suppress X-axis column
Hex Logic Values
CSV Format
Logic Waveforms to PWL Sources
PWL Logic-1 Level : 3.30E+00
PWL Logic-0 Level : 0.00E+00
0-to-1 PWL Slew : 1.00E-11
1-to-0 PWL Slew : 1.00E-11
PWL Source Netlist Syntax Options
Hierarchical Delimiter:
Replace With '_'
Keep Original
Convert Bus A[#] to
A_#
A_#_
A[#]
A<#>
Support Spectre Netlist Format
Start Offset : 0
Number of Valid Digits : 4
VCD timescale: auto
Ok
Help
Cancel

Using CosmosScope to view the simulation results

The spice netlist generated by SAE is saved in:

~/ee406/s14/analog/simulations/ee406Lib/cs_ee114/schematic/HSPICE/nominal/netlist/input.spi

```
*  Generated for: HSPICE
.option search='/home/talarico/ee406/s14/analog/hspicemod'

.option PARHIER = LOCAL
.option NOMOD = 1
.option ARTIST=2 PSF=2
.temp 25
.include 'ee114_hspice.sp'

.GLOBAL gnd!

ib vdd dn dc=250u
r vb dn r=10k
rin gn net18 r=20k
mn dn gn gnd! gnd! nmos114 w=40u l=1.6u
vb vb gnd! dc=2.5
vdd vdd gnd! dc=5
vin net18 gnd! dc=1.08817 ac=1

.ac DEC 100 1 100Meg
.op All 0
.option opfile=1 split_dp=1

.end
```

It is important to understand what the various options mean:

.option PARHIER = LOCAL
Use this option to specify scoping rules

.OPTION NOMOD = 1
This option suppresses the printout of model parameters.

.option ARTIST=2
Enables the Virtuoso® Analog Design Environment if ARTIST=2. This option is generally used together with .OPTION PSF.

.option PSF=2
If .option PSF=2, HSPICE produces ASCII output

.option opfile=1
Use this option to output the operating point information to a file.

- If value is 1, operating point information is output to a file named <design>.dp#.
- If value is 0, the operating point information outputs to stdout

The results of the simulation are put in:

~/ee406/s14/analog/simulations/ee406Lib/cs_ee114/schematic/HSPICE/nominal/results

The HSPICE results produced by SAE are not in the correct format to be viewed with Cosmoscope

In order to use Cosmoscope as viewer we need to overwrite the .option ARTIST=2 PSF=2 with .option POST.

```
…
.option artist=2 psf=2
.option post
…
```

and re-run HSPICE simulation manually.

For convenience name the new "hacked" spice netlist input.sp:

```
.option search='/home/talarico/ee406/s14/analog/hspicemod'

.option PARHIER = LOCAL
.option NOMOD = 1

.option ARTIST=2 PSF=2
.option POST * added by Claudio
.temp 25
.include 'ee114_hspice.sp'
*Custom Designer (TM) Version H-2013.03-SP2-1
*Wed Nov 27 11:43:47 2013

.GLOBAL gnd!
ib vdd dn dc=250u
rin gn net18 r=20k
r vb dn r=10k
mn dn gn gnd! gnd! nmos114 w=40u l=1.6u
vb vb gnd! dc=2.5
vdd vdd gnd! dc=5
vin net18 gnd! dc=1.08817 ac=1
```

```
.ac DEC 100 1 100Meg
.op All 0
.option opfile=1 split_dp=1

.end
```

make a copy of the runSimulation script:

cp runSimulation runSimulationH

and modify it as follows:

hspice -i input.sp -o ../results/Hhspice > ../results/run.log 2>&1

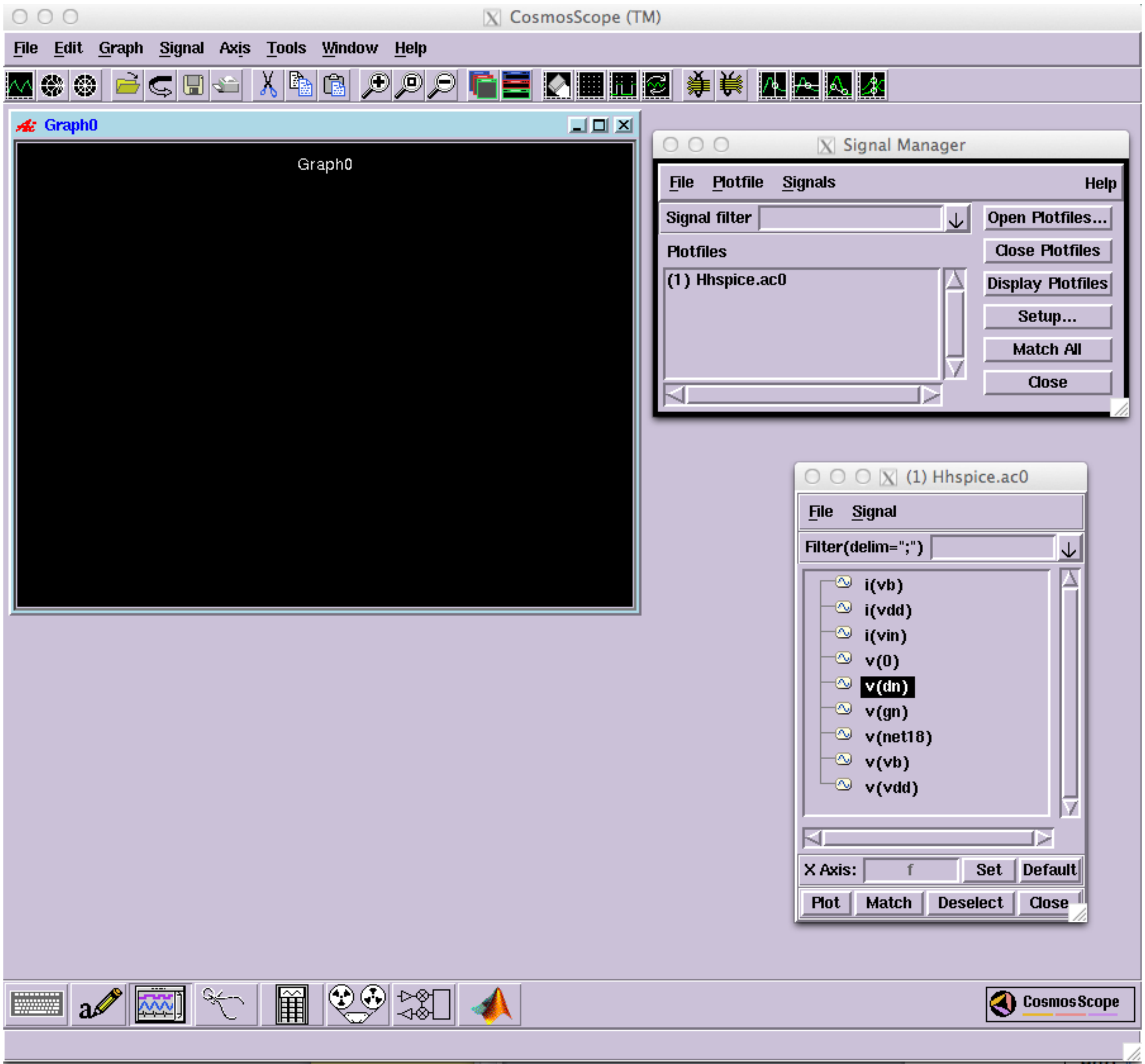
Run the “hacked” spice simulation:
./runSimulationH

and move to the result directory:
cd ../results/

Invoke cosmoScope:

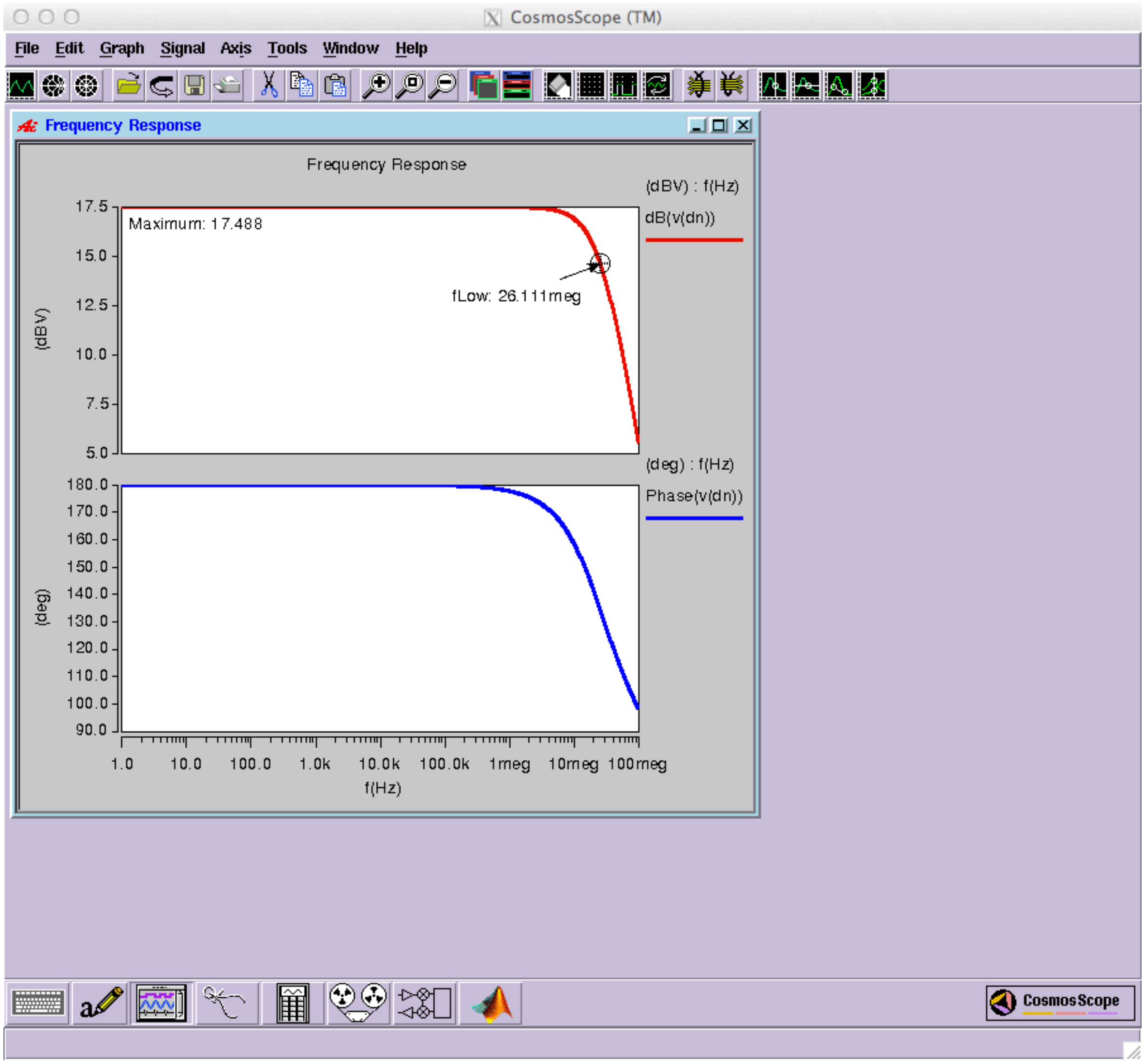
cscope &

File → Open Plot Files
Double click on Hhspice.ac0
Select the signal to plot and click on the plot button.



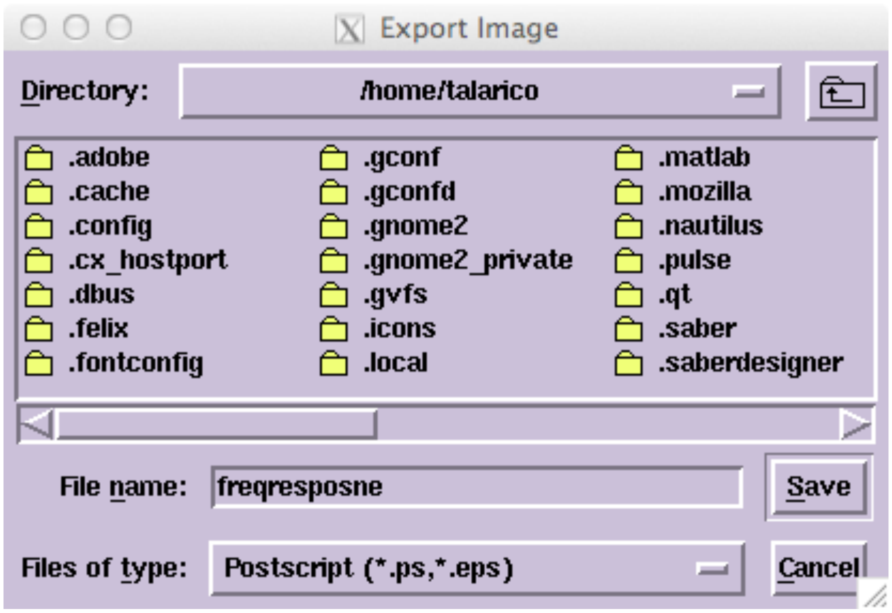
Change the color and thickness of the waveform (right click on the name of the signal and edit the attributes), edit the title of the graph (click on the title and edit it), add measurements to the plot (click on Tools → Measurement Tool and then select the type of measurement you are interested in), etc.

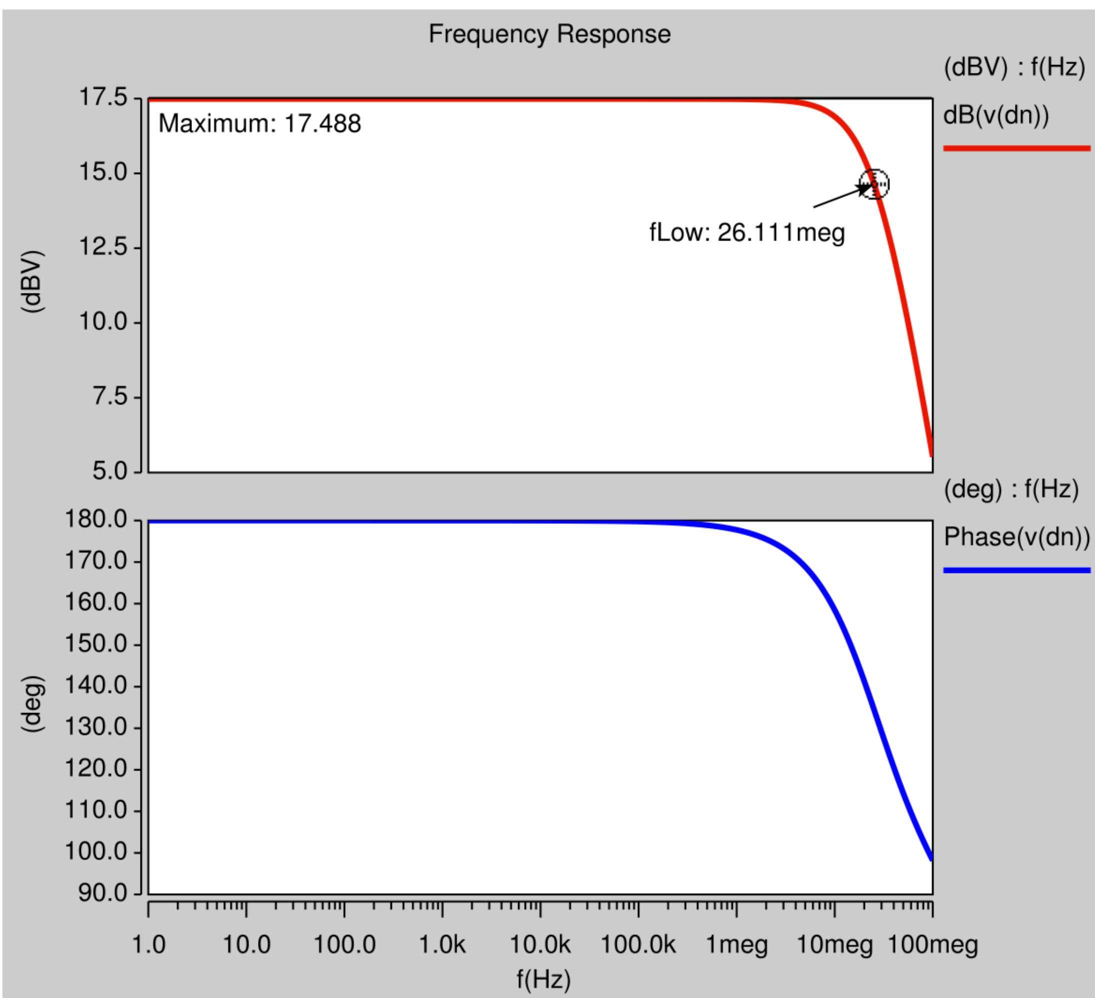
... Make sure to change the background !!
Graph → Color Map → Map 2



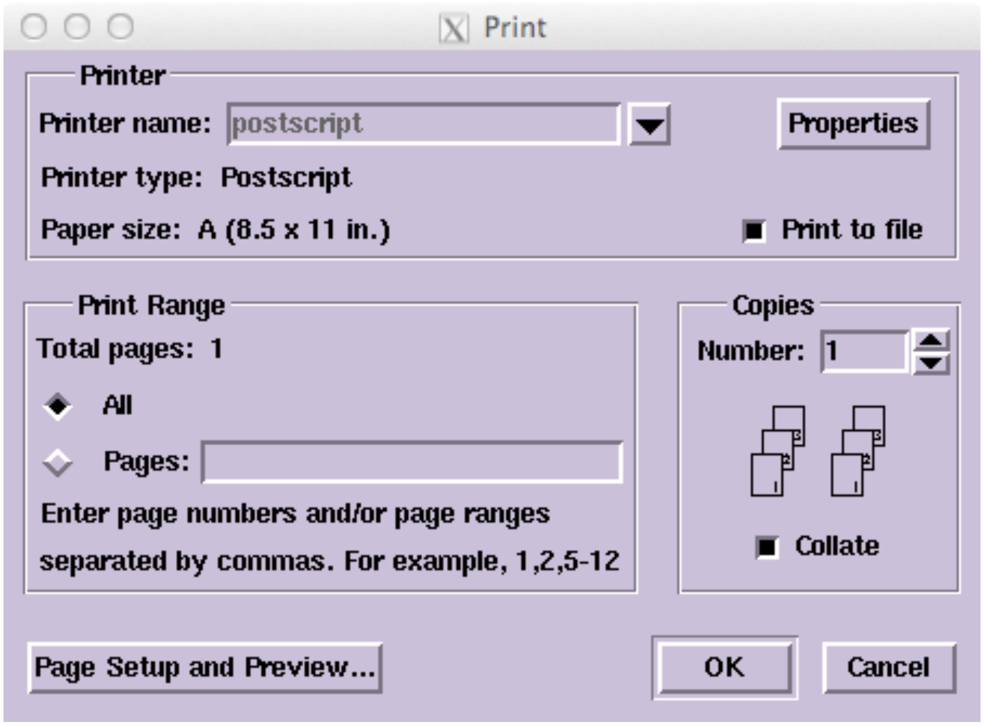
The plots can be printed or exported as images.

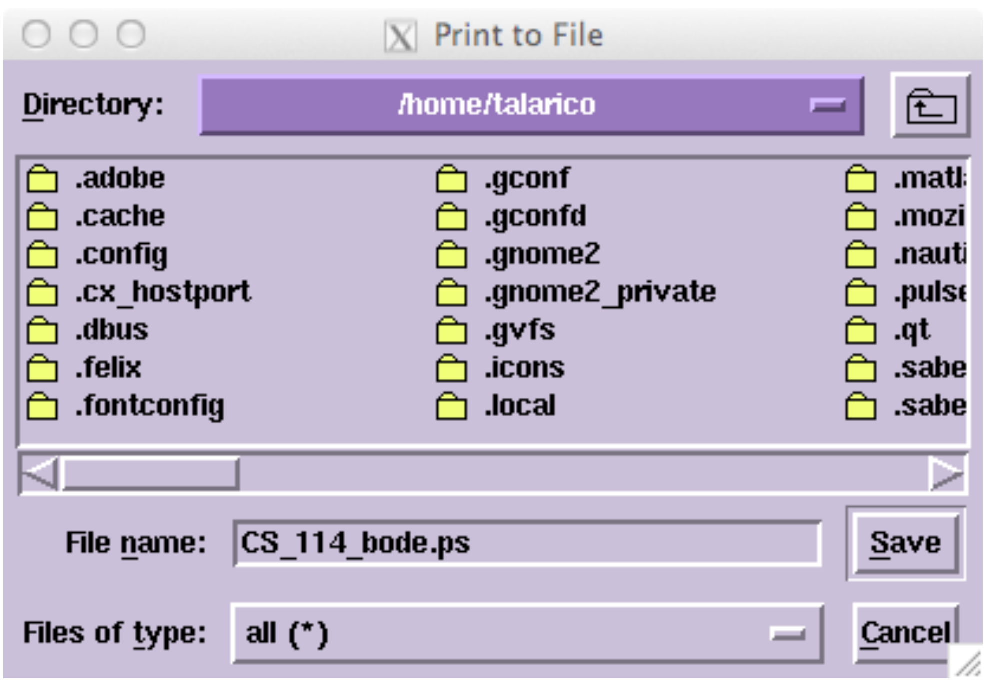
File → Export Image





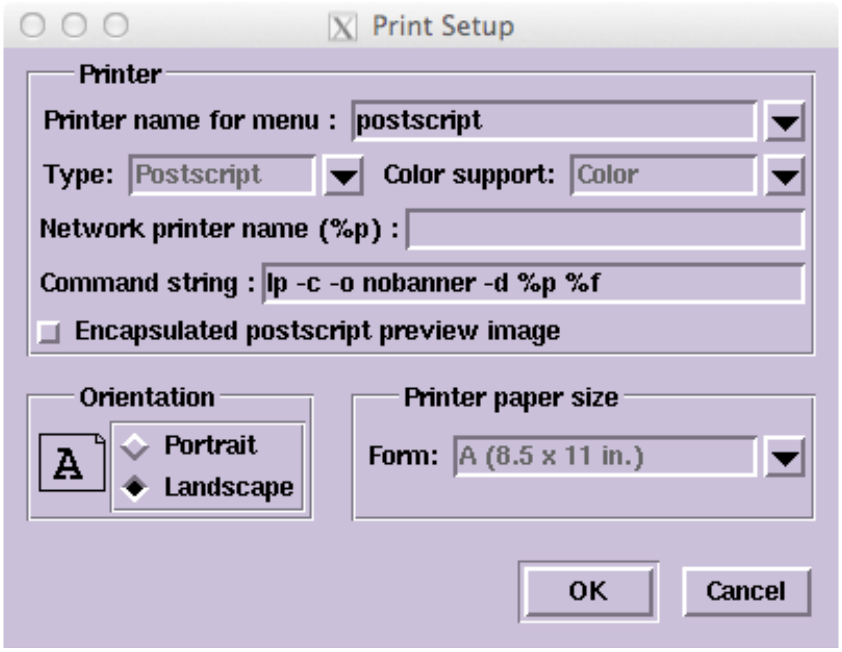
File → Print

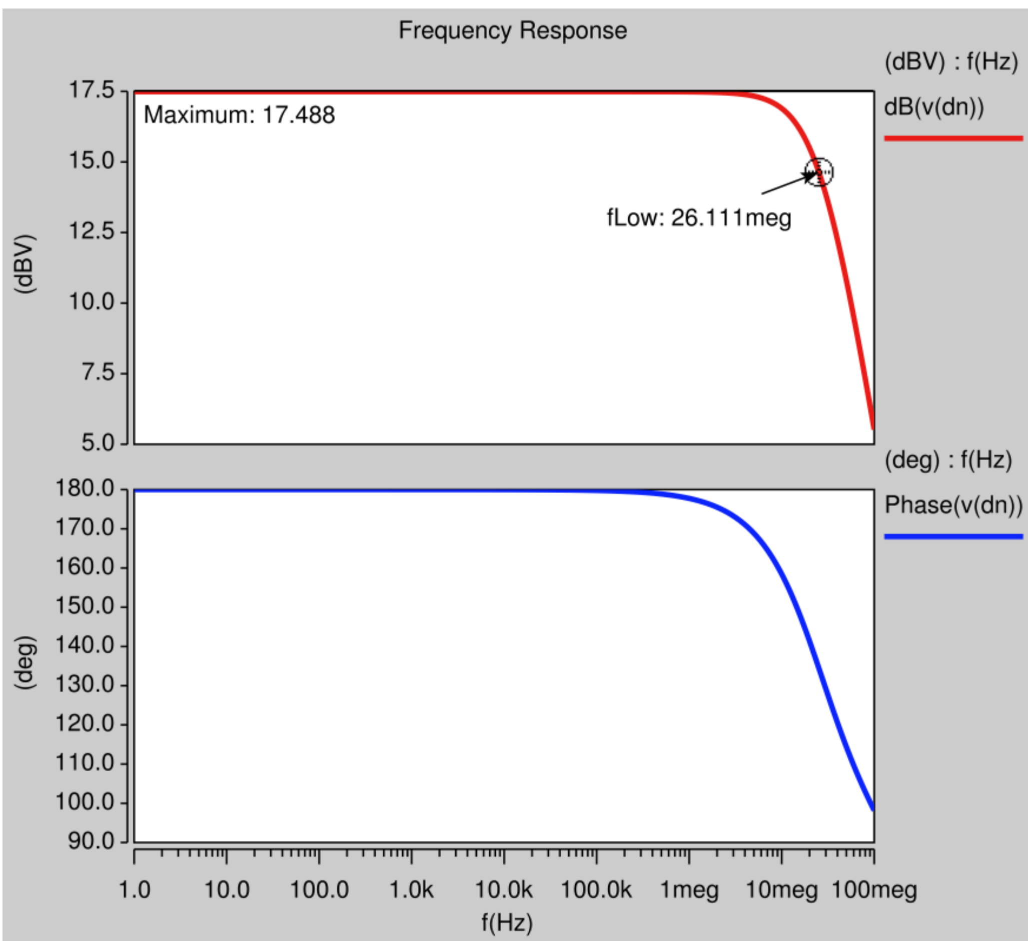




Before Printing make sure the printer is setup correctly:

File → Printer → Setup





The colors can be further customized: Edit → Preferences → Graph

Using Matlab to import HSPICE simulations.

Setup a directory for the matlab files:
mkdir ~/ee406/s14/matlab

We will use the HSPICE toolbox for MATLAB developed by M. Perrott. The HSPICE toolbox for Matlab is a collection of Matlab functions that allow you to manipulate and view signals generated by HSPICE simulations directly.

The toolbox files are located at:
/usr/local/MATLAB/personal/HspiceToolbox

Note: the toolbox requires the HSPICE outputs to be in the format generated through the POST option.

There are many functions in the toolbox, but we will focus mainly on three of them. See hspice_toolbox.pdf for more details.

1. loadsig (…)
   This function loads the circuit signals into MATLAB. It takes the hspice output file, such as DC (*.sw0) or AC (*.ac0) files as the function input. For example, if you simulated a common source amplifier in HSPICE and have CS.ac0 as the output file, the function call is:
   x = loadsig('CS.ac0');

2. lssig(...) - This function lists the signals contained in the file loaded by the loadsig(...) function. Continuing with the above example, this function is called as:
   lssig(x);

3. evalsig(... , ...) - This function actually loads the signals from the HSPICE output file into vectors in MATLAB. For example, if have a signal called v_dn, the function is called as follows:
   vout = evalsig(x, 'v_dn')

Once you have loaded a signal into a MATLAB vector, you can then proceed to do any mathematical manipulations on the data.

MATLAB example:

```
%
% author: C. Talarico
% file: cad.m

clear all; close all; clc;
addpath('/usr/local/MATLAB/personal/HspiceToolbox')
x =
loadsig('/home/talarico/ee406/s14/analog/simulations/ee406Lib/cs_ee114/schem
atic/HSPICE/nominal/results/Hhspice.ac0')
lssig(x)
vout = evalsig(x,'v_dn');
freq = evalsig(x,'HERTZ');
mag = abs(vout);
phase = angle(vout); % phase in rad
magdb = 20*log10(mag);
degree = phase.*180/pi;

subplot(2,1,1);
set(gca, 'fontsize',14);
semilogx(freq*1e-6,magdb,'linewidth',2,'color', 'b', 'linestyle','-');
ylabel('Magnitude [dB]', 'fontsize', 12);
xlabel('freq [MHz]', 'fontsize', 12);
maxdb = max(magdb) + 5;
axis([min(freq*1e-6) max(freq*1e-6) 0 maxdb]);

% compute DC gain and f3dB
format shorte;
gain = magdb(1);
idx = find(magdb < gain - 3, 1, 'first');
f3db = 1e-6*freq(idx);

% annotate the plot with DC gain and f3dB
str0 = sprintf('CS Frequency Response');
str1 = sprintf('DC gain = %.2f [dB]', gain);
str2 = sprintf('f3db = %.2f [MHz]', f3db);
str = {str0, str1, str2}; % distribute the string on three lines
title(str, 'fontsize', 16);

g3db = gain-3;
% annotate the f3db point with "template lines"
% horizontal line
line([min(1e-6*freq) 1e-6*freq(idx)], [g3db g3db], 'linestyle', ':', ...
   'linewidth', 3, 'color', 'r');
```

```
%vertical line
line([f3db f3db], [-20 g3db], 'linestyle', ':', ...
    'linewidth', 3, 'color', 'r');

subplot(2,1,2);
set(gca, 'fontsize',14);
semilogx(1e-6*freq,degree,'linewidth',2,'color', 'm', 'linestyle','-');
ylabel('Angle [degree]', 'fontsize', 14);
xlabel('freq [MHz]', 'fontsize', 14);
```