Pspice Tutorial

Class: Power Electronic 2

(EE563)

By Colorado State University Student

Minh Anh Nguyen

Five years ago, during my Circuit Theory class, professor Gary Robinson had introduced and applied the Pspice software to solve the lab questions. It was the first time I heard and work with the Pspice software. But Dr. Robinson never explained in detail how to create and run a circuit in Pspice. And unlike every others student in my Circuit Theory, I didn't take EE100, which offer for freshman student. This class was explained the basis of circuit and Pspice program in detail. Since, I didn't have any knowledge of basis of Pspice program, I struggled try to get all my labs done. Others take an hour to complete their lab but I have spend more than four hours to complete my lab, sometime my answer still not correct. When the semester was over, I was so happy because I don't have to work with Pspice any more. However, I make a promised to myself that one day I would learn and understand Pspice program. And since my field study in Electrical engineering is Computer Concentration, which mean I only focus on study Digital and software classes. I never had a chance to learn and understand the Pspice software until today. So in this paper, I will try my very best to explain to you the concept of the Pspice. Here is the outline of the Pspice that I will cover in this paper.

1. What is the Pspice program/tool?
    a. History of Pspice
    b. When is the Pspice created?
2. What is the Pspice student version?
    a. Explain what is the Pspice student version
    b. When should we use the Pspice student version
    c. Show step by steps how to load and create the Pspice file at CSU.
       What are the files contain in the Pspice folder
    d. Show how to open the Pspice file
    e. Show how to creating an input file
    f. What is schematic editor
    g. Show step by steps how to draw a circuit
        i. Getting the parts
        ii. Placing the parts
        iii. Connecting the circuit or wiring components
        iv. Changing the name of the part
        v. Change the value of the part

vi. Making sure you have a GND

vii. Voltage and current bubbles

h. Show step by steps how to save a circuit

i. Show step bys step how to probe

i. Before you do the probe

ii. Show how to using a probe

iii. To start the probe

iv. Graphing

v. Adding/deleting trace

vi. Doing math

vii. Labeling

viii. Find points

ix. Saving

j. Show step by steps how to read the output file

k. Show step by steps how to do analysis menu

i. Ac sweep

ii. DC sweep

iii. Bias point detail

iv. Parametric

v. Sensitivity

vi. Temperature

vii. Digital setup

viii. Transient

l. Show step by steps how to check if the Pspice is not run

m. Show step by steps how to plot the circuit

n. Use some example from the homework and the appendix B to some how to create a Pspice file, run and read the output file.

o. Compare the Pspice to the Mathcad result

3. What is the Pspice professional version?

a. Explain what is the Pspice professional version

b. When should we use the Pspice professional version

c. Show step by steps how to load and create the Pspice file at CSU.
d. What are the files must contain in the Pspice professional version?
e. Show how to open the Pspice file
f. Show step by steps how to save and run a circuit
g. Show step by steps how to read the output file
h. Show step by steps how to check if the Pspice is not run
i. Show step by steps how to plot the circuit
j. Use a design of a hearing aids circuit to show to create a professional Pspice file, run and read the output file.
k. Explain the types of source
    i. Voltage source
        1. VDC
        2. VAC
        3. Vsin
        4. Vpuls
        5. PWL
    ii. Current source
l. Show step by steps how to do analysis menu
    i. Ac sweep
    ii. DC sweep
    iii. Bias point detail
    iv. Parametric
    v. Sensitivity
    vi. Temperature
    vii. Digital setup
    viii. Transient analysis
    ix. Distortion analysis
    x. Linear AC analysis
    xi. Non-linear DC analysis
    xii. Step ceiling
    xiii. Transfer function

xiv. Phase margined

m. How to specify the circuit topology and analysis?

n. Data statement to specify the circuit components and topology?

o. How to using a sub-circuit?

   Defining a sub-circuit?

p. Explain the types of the following

   i. Resistor
   ii. Capacitor C and inductor L
   iii. Mutual inductor
   iv. Ideal transformer
   v. Sinusoidal sources
   vi. Voltage controlled
   vii. Current controlled
   viii. Operation amplifier and other elements
   ix. Diode

q. Compare the Pspice result with the MathCAD results

4. Conclusion

   a. Reference

First I want to tell you a little bit about the history of Pspice; SPICE is an analog circuit simulator and stands for Simulation Program with Integrated Circuit Emphasis that was developed in the late 1970's for IC analysis and design at the University of California at Berkeley. Spice is widely used industry and a new BSEE graduate is expected to be familiar with the program. PSpice is one of the many commercial SPICE derivatives, and has been developed by MicroSim Corporation. Pspice is a version of the original Simulation program with integrated circuit Emphasis program that have been adapted for PC. PSpice is the program, which carries out the actual simulation of the circuit. Normally, one describes a circuit (using the PSpice language) on a text editor. PSpice simulates the circuit, and calculates its electrical characteristics. If we need a graphical output, PSpice can transfer its data to the Probe program for graphing purposes. Also Pspice is a simulation program that models the behavior of a circuit. And Pspice is a Product of the OrCAD Corporation and the student version we are using is “freeware”. At Colorado State University Pspice SV (student version) is install on an ENS server.

Next I want to explain what is student version of Pspice and why Electrical engineering students are only introduced to the student Pspice version. For all the freshman, sophomore and junior Electrical Engineering students attending at Colorado State University can only apply the student Pspice version to solve their labs or home work problems. Because the student version of Pspice is intended for use by college students and its free. The student Pspice version had circuit simulation limited. Here are some of the limited of the student Pspice version.

The circuit can only have 64 nodes, 10 transistors, 65 digital primitive devices and 10 transmission lines in total (ideal or non-ideal) and 4 pair wise couple transmission lines. For additional limits are the sample library includes 39 analog and 134 digital parts. The device characterization in the pspice model editor is limited to diodes. Stimulus generation in the pspice stimulus editor is limited to sine waves (analog) and clock (digital). Circuit optimization with the Pspice optimization is limited to one goal, one parameter and one constraint. You cannot create CSDF format data files. You can only display simulation data from simulations performed with the student version of the simulator. For the Schematics, you can place a maximum of 50 parts on a schematic design and can only draw on size A sheets.   For the Capture, the Pspice libraries are the only ones included. The standard capture libraries are not included. Import facilities, netlisters and accessories that are not relevant to Pspice are not included. You cannot save a design that contains more than 60 parts, (you can view or create larger designs, but you can not save them). You cannot save a library that contains more than 15 parts.

Colorado State University (CSU) also provided a professional version of Pspice for the senior and graduated Electrical engineering students. And the license are limited, Colorado State University (CSU) only had 15 licenses for professional Pspice version. Therefore, only 15 peoples can create and run the circuit at the same time and cannot open the Professional version window longer than 15 minute.   The professional version of Pspice is intended for used by professional, industry or company. The Circuit simulation is not limited to circuits. You can draw on both size A and B sheets and can save a design that contains more than 60 parts. You can save a library that contains more than 15 parts.

In order to run Pspice software at Colorado State University (CSU), you must setup the Pspice software into your own account (U drive). Otherwise you cannot create and run a circuit on Pspice software. Colorado State University provided both student and professional version of Pspice on ENS server. Here are the steps how to setup both student and professional version of Pspice into your own account.

First, you must uses window NT explorer to locate the folder

Second, if you are not a senior Electrical engineering, then use the student version. To load the students Pspice version following the line below:

s:\Application\Pspice_SV_91\CSU_setup\Pspice

Open the window NT explorer and locate the "s" folder

Click on the "s" folder and click on "Application"

Click on the "Pspice_SV_91" and click on "CSU_setup"

Once you see a Pspice folder, copy the Pspice folder and its sub_files into your u drive, by using right mouse button to drag and drop.

If you are a senior or graduate Electrical Engineering, then use Professional version. Before load the professional version make sure you have a student Pspice version in your own account (U drive). To load the professional version, again using window NT explorer and locate the folder and following the line below:

s:\Application\Pspice_pro_92\CSU_setup\Pspice

Open the window NT explorer and locate the "s" folder

Click on the "s" folder and click on "Application"

Click on the "Pspice_pro_92" and click on "CSU_setup"

Once you see a Pspice folder, copy the Pspice folder and its sub files into your u drive, using right mouse button to drag and drop.

After you setup the Pspice into your account, make sure that your Pspice folder must contain all the files such as Pspice SV, Pspiceev.ini, userlib and Backup. If one of the files in your folder is missing, then you cannot create and run the circuit using your Pspice software. The figure below is all files must contain in your student version

Where Pspice SV is schematic file and you open it to create the circuit.

Here are the file must contain in Professional version

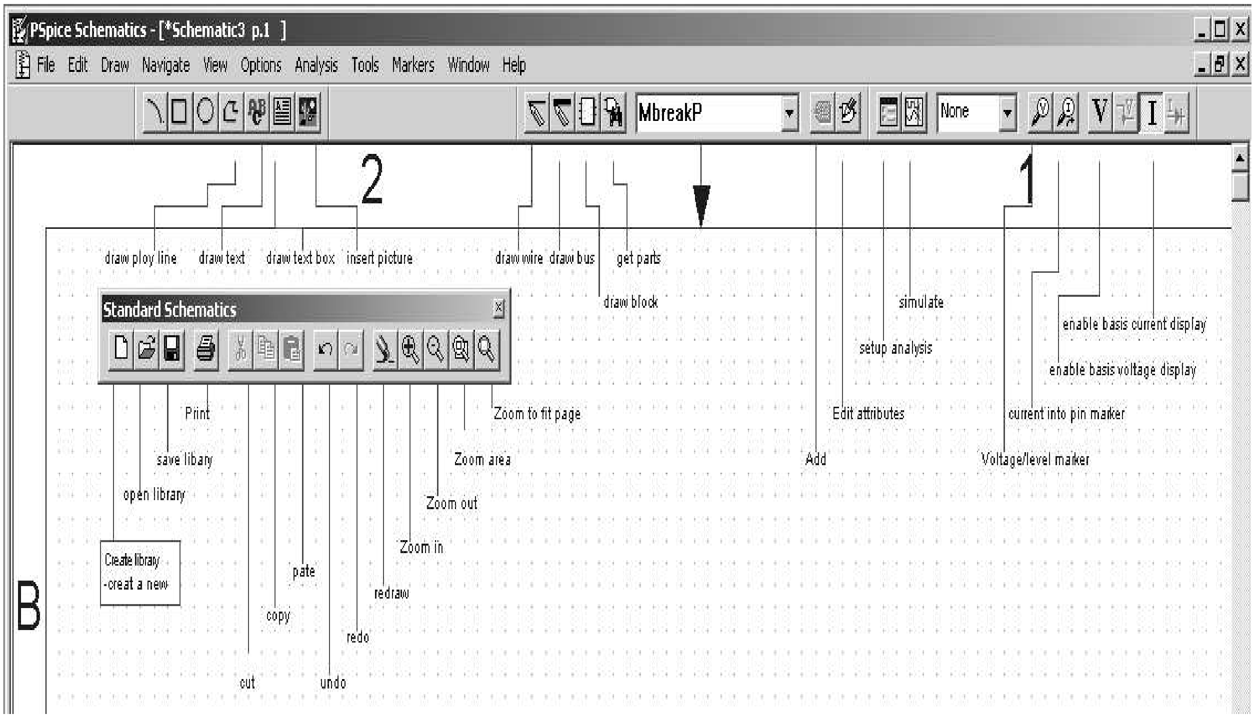
-Pspice pro, Pspice92.ini and all the files of Pspice SV

This figure is shown all the files must contain in professional version

Where Pspice_pro is schematic file and you double click on Pspice button to open the schematic page to create your own circuit.

After you setup and load the student or professional version of Pspice, you can open the schematic and design your own circuit. Here are some steps to open the schematic

To start, double click on the Pspice SV icon and Pspice schematic will open. The figure below is schematic



Where create library: create a new library

Open library is open an existing library.

Save library is save the current stimulus library

Print is prints the active window

Cut is removes the selected object and places it on the clipboard

Copy is copy the selected objected to the clipboard.

Paste is paste the contents of the clipboard at the cursor.

Undo is undoes the last command performed.

Redo is redo the last command performed.

Redraw is redraw a line

Zoom in is a zoom in on a specified point

Zoom out is zooming out from a specified point

Zoom fit is zooming to show all traces and label

Zoom area zoom in on a selected area of graph

Voltage marker level is places a voltage level marker on the schematic page

Current into pin marker is place a current into pin marker on the schematic page

New simulation profile is open the new simulation dialog box

Edit attributes is open the edit digit transition dialog box, which you use to modify the values and setting of the selected stimuli.

Add is adds a new point or transition to the stimuli at the cursor location.

Enable bias current display is display the current at point/ node in the circuit

Enable bias voltage display is display the voltage at point/ node in the circuit

Get part is get the components/part to design a circuit

Draw line is draw a line from one part to another

Draw bus is draw a bus from one part to another

Draw a bock is draw a block on the schematic page

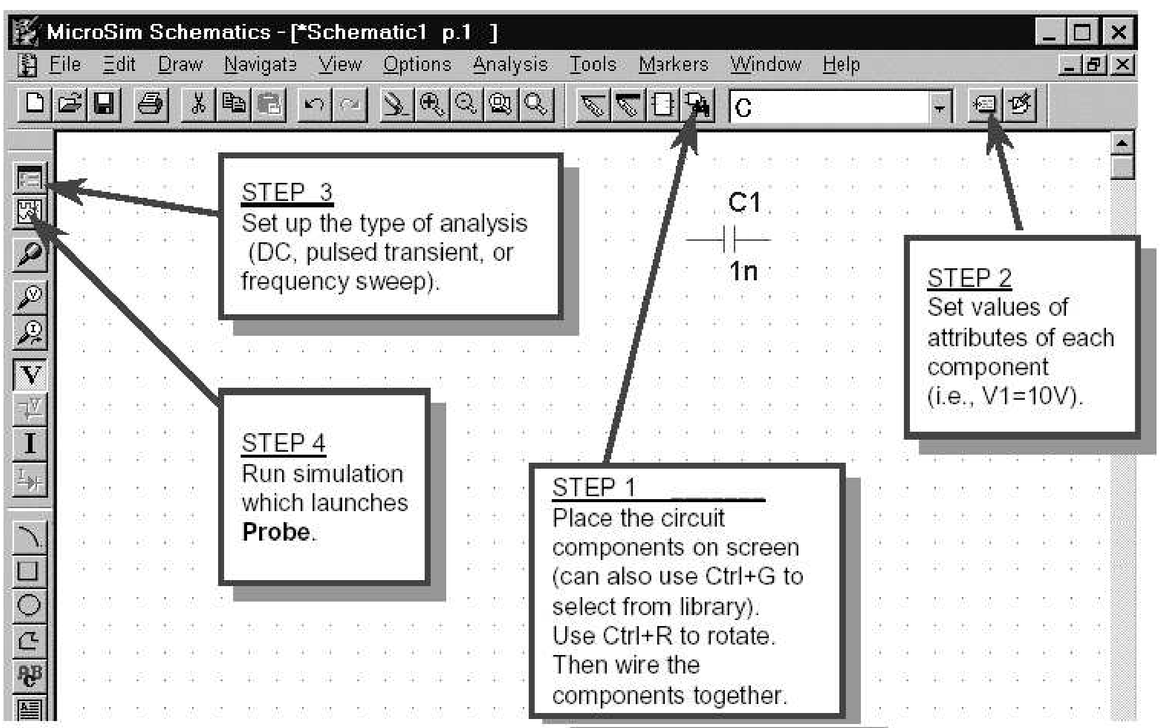
Draw text is draw a text on the schematic page

Insert a picture is insert a picture into a schematic page

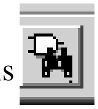
Draw a text line is draw a line to write text

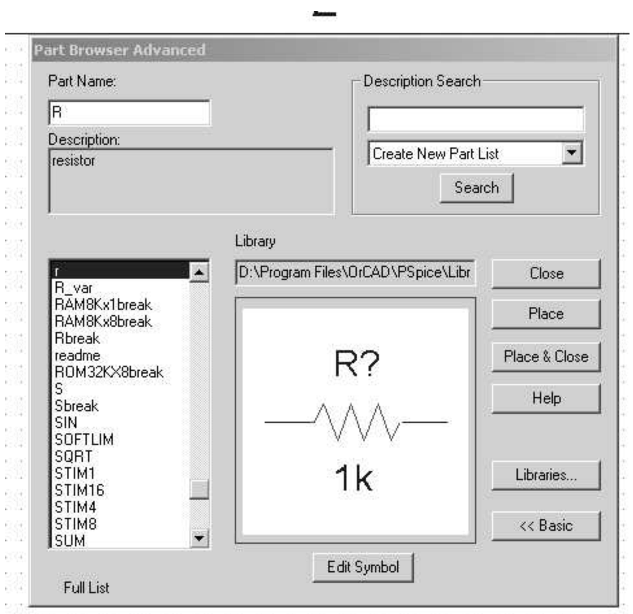
Draw a ploy line is draw a ploy line into schematic page

Use the Pspice schematic page above to create your circuit diagram (a process called schematic capture). The following is some major steps that may help you to start your design.



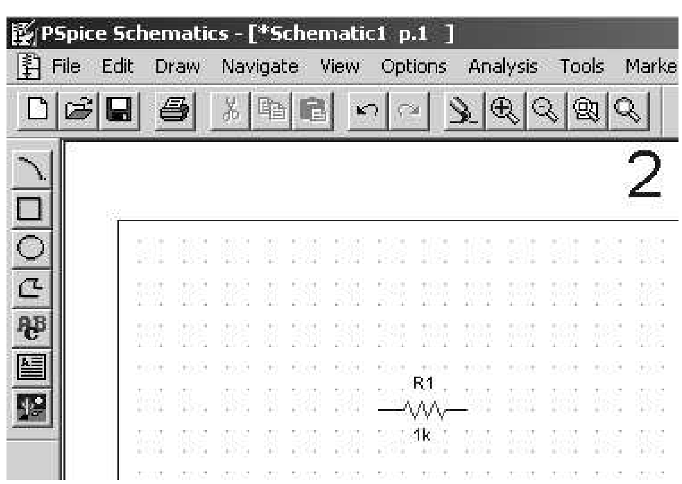
After you open the schematic and ready to design your circuit. Here are some steps or tool that you need to create your circuit:

To Getting the parts or components for your circuit, either on your toolbar click on this  icon or on your menu click on Draw/Get new part or on your keyboard click on Ctrl-G. And then Part Brower advanced window is open like the figure show below.

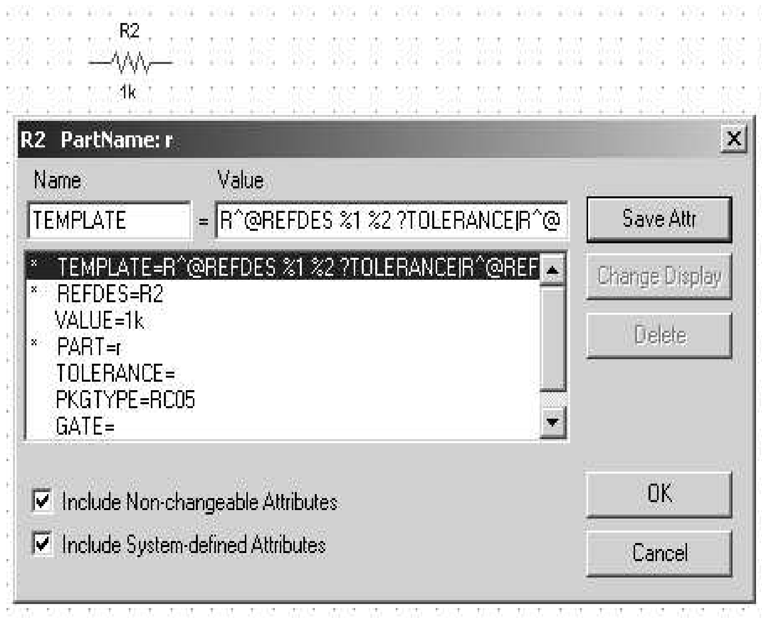


When the Part Brower advanced window is open, on the "part name" type the name of components that you need or want to use to design your circuit. For example, you need a resistor then on the Part name type "R" and click on "r" to make sure that the component, which you chose to design your circuit.

Once you chose the right component to create your circuit, the next step is placing the parts. To placing the parts, double click on the part that you want and place it on the schematic, like the figure below



Next, if you wish to change the name and the value of the part, then simply do the Double click on the part, the part turn to red color and a window will open. The figure below is shown how to change the name and value of the part.



Once, the part name window is open, click on the VALUE to change the value and click on the REFDES to change the name of the parts.

When change the value of part, please be careful with the unit. If you are type a wrong unit, then you will get a wrong result for your circuit. Pspice recognize the units of time, second, millisecond, microsecond, nanosecond, picoseconds are write with abbreviation s,ms,us,ps. Unit of resistor, K-ohms and Meg-ohms are write with abbreviation K and Meg. Note that 10M is evidently 10 micro-ohms.

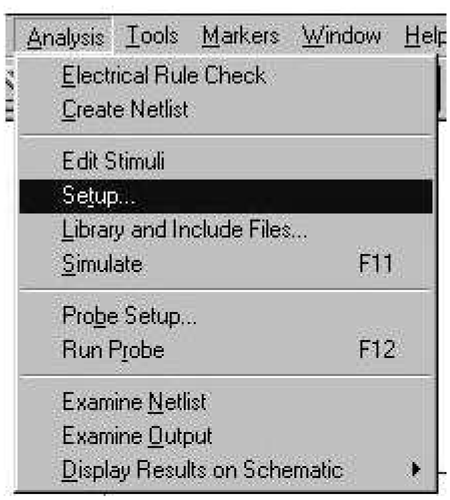
When you have all the components that you need to design your circuit. Then you have to connecting all the parts together. To connecting the parts, either on your toolbar clicks on this  icon or on your menu click on Draw/Wire or on your keyboard do Ctrl-W and you simply connecting one component to another.

To makes the circuit look nice, less area and easy to connecting all the components in your design, you can Flip or Rotate part. To do Flip/Rotate part, click on the part that you want to Flip or Rotate, and the part is turn red. Then either on your menu selects on the Edit and then selects on Flip or Rotate or on your keyboard uses Ctrl-F to flit the part and Ctrl-R to rotate the part.

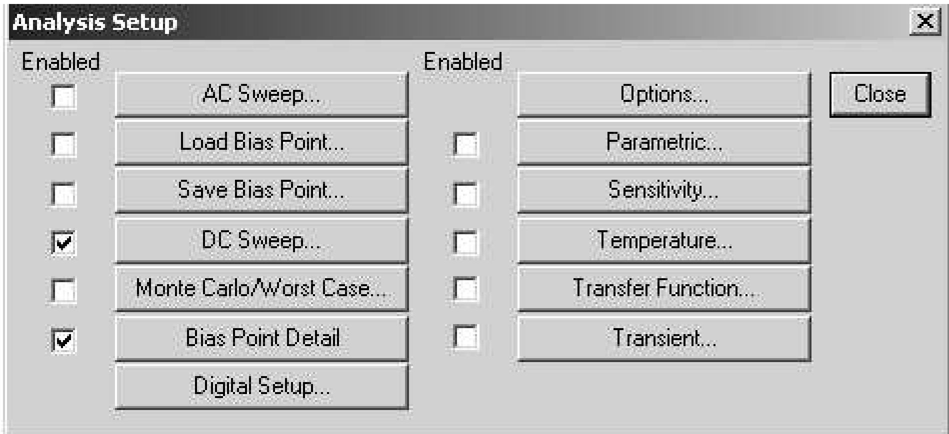
During your process of design a circuit, if you do or don't want any part and you don't like to go through some hard step described above, then you can do Cut/copy/pate part. To Cut/copy/pate part, click on the part which you want to cut or copy, the part turned into red color. Then either on your menu click on cut or copy or pate, you also can use the Ctrl-X, Ctrl-C, and Ctrl-V from the keyboard.

To make your circuit easy to read, follow and trace, you can label all the nodes on your circuit. Labels can be edited by selecting and Double-click on them. Letter and number are fine; most other characters will be cause error. e.g. r' won't work but r- is ok. But parts that normally don't have labels, like wire, may be labels as well.

After you complete your design on the schematic, then do the analysis (calculated, plot output, ect) of your design. To do the analysis, either on the toolbar clicks on this  button or on the menu click on Analysis/setup. The figure below is shown how to use the menu to do the analysis setup.



Once, you click on the "analysis/setup" it's bringing up a setup dialog box. This dialog offers numerous sub-dialog boxes to set up various types of data collection. The figure below is shown analysis setup dialog box.



AC sweeps: To setup to collect data for a frequency response

Bias point detail: check by default

Parametric: to analyze using multiple parameters for components such as multiple values for R.

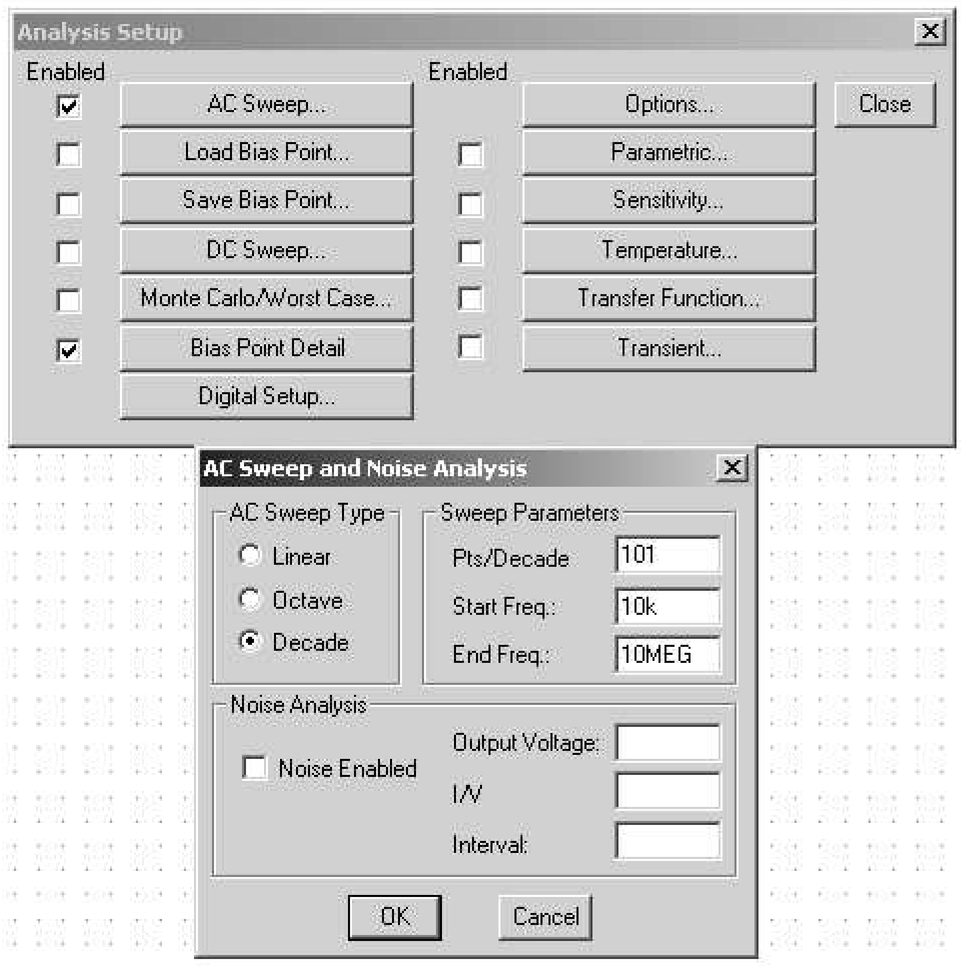
Transfer function: the input source must be current or voltage source

Transient: the first order of business is to enter something in the steps ceiling field.

To plots and find the transfer function of the circuit. Then apply on AC sweep. Because the AC sweep allows you to plot magnitude versus frequency for different inputs in your circuit. In the AC sweep menu you have the choice of three types of analysis: Linear, Octave and Decade. These three choices describe the X-axis scaling which will be produced in probe. For example, if you choose decade then a sample of your X-axis might be 10Hz, 1kHz, 100kHz, 10MHz, etc.... Therefore if you want to see how your circuit reacts over a very large range of frequencies choose the decade option. You now have to specify at how many points you want PSpice to calculate frequencies, and what the start and end frequency will be. That is, over what range of frequencies do you want to simulate your circuit.

In the AC sweep you also have the option of **Noise enable** in which PSpice will simulate noise for you either on the output or the input of the circuit. These noise calculations are performed at each frequency step and can be plotted in probe. The two types of noise are: V (ONOISE) for noise on the outputs and V (INOISE) for noise on the input source.

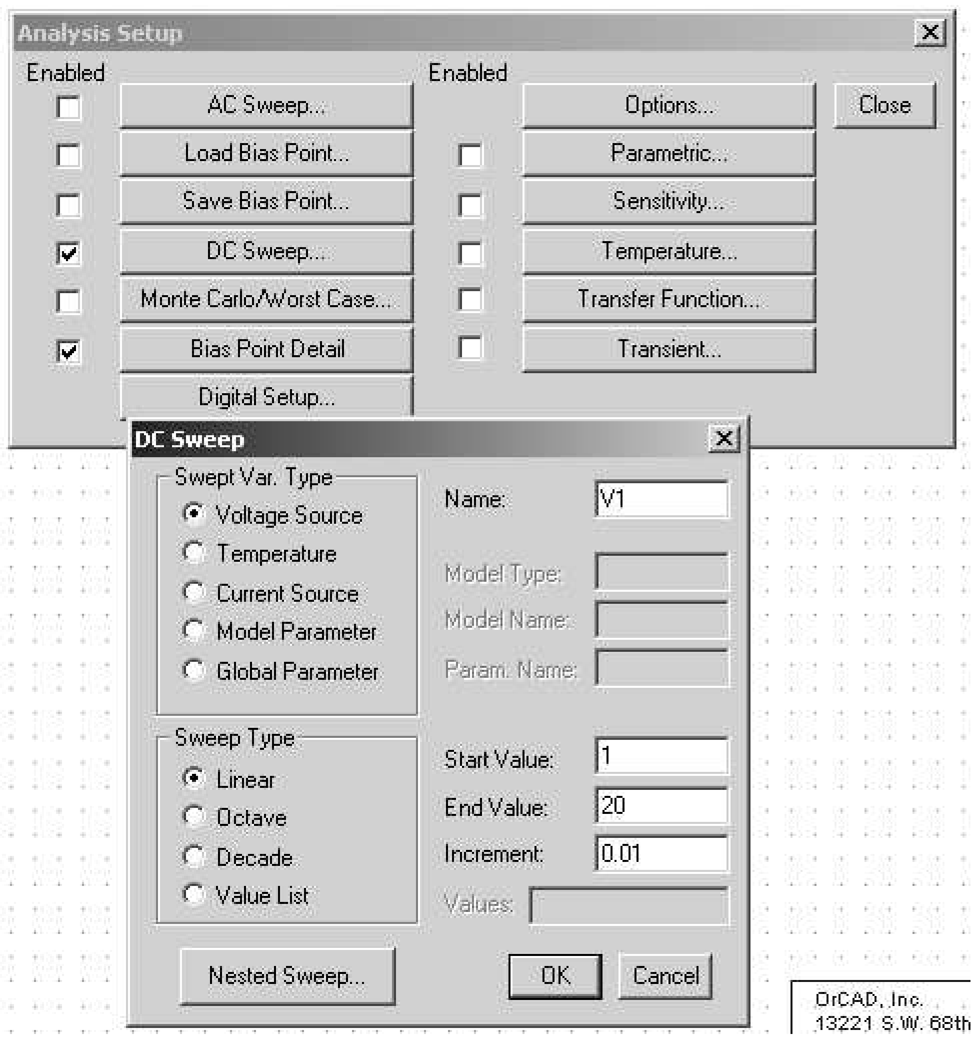
To use input noise you need to tell PSpice where you consider the 'input' in your circuit to be, for example, if your voltage source is labeled 'V1'. Finally you need to specify in what interval you want the noise to be calculated (note: the default interval for spice is zero, i.e.: no noise will be calculated). The figure below is shown the AC sweep.



To find the DC conditions for voltage, current, temperature and parameter global of the circuit. The uses the DC sweep. Because the DC sweep allows you to do various different sweeps of your circuit to see how it responds to various conditions.

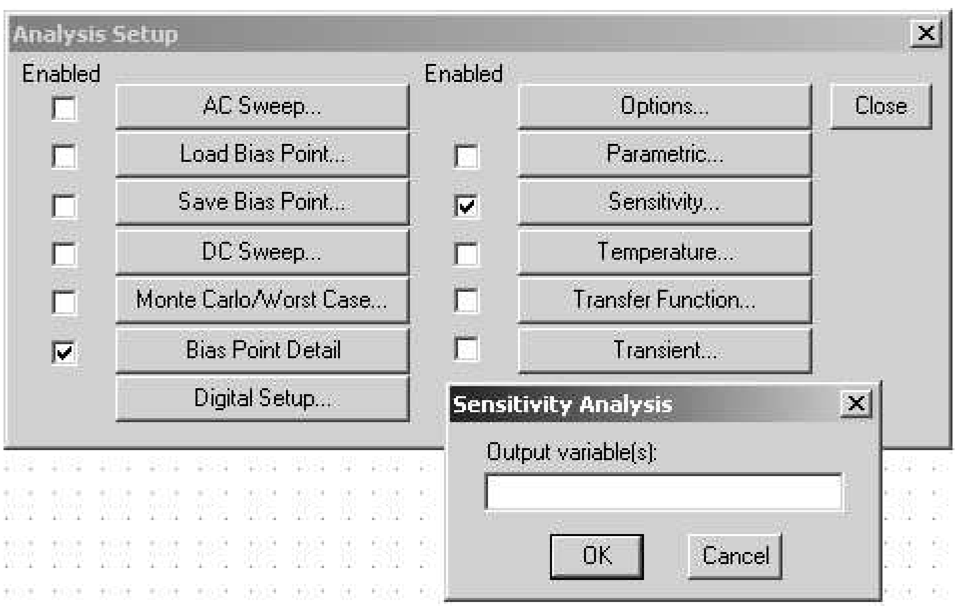
In order to simulate your circuit uses DC sweep, you need to specify a start value, an end value, and the number of points you wish to calculate. For example you can sweep your circuit over a voltage range from 0 to 12 volts. The main two sweeps that will be most important to us at this stage are the voltage sweep and the current sweep. For these two, you need to indicate to PSpice what component you wish to sweep, for example V1 or V2. An excellent feature of the

DC sweep in PSpice, is the ability to do a **nested sweep**. A nested sweep allows you to run two simultaneous sweeps to see how changes in two different DC sources will affect your circuit. Once you've filled in the main sweep menu, click on the nested sweep button and choose the second type of source to sweep and name it, also specifying the start and end values. (Note: In some versions of PSpice you need to click on **enable nested sweep**). Again you can choose Linear, Octave or Decade, but also you can indicate your own list of values, example: 1V 10V 20V. **DO NOT** separated the values with commas. The figure below is shown how to setup the DC Sweep.

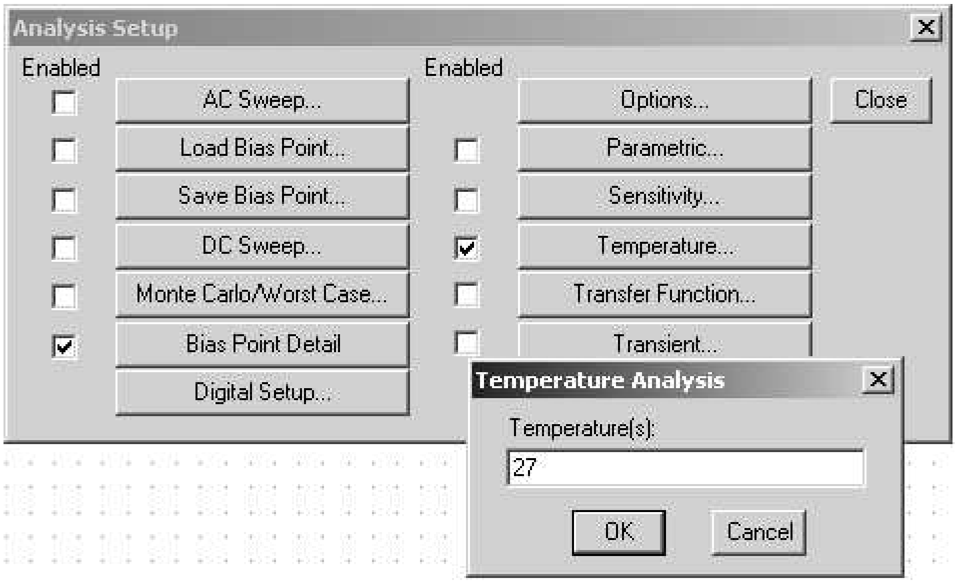


In Pspice the Bias point detail is always set as default. The Bias point detail is a simple, but incredibly useful sweep. It will not launch Probe and so give you nothing to plot. But after simulated the circuit, clicking on **enable bias current display** I or **enable bias voltage display** V, this will indicate the voltage and current at certain points within the circuit.

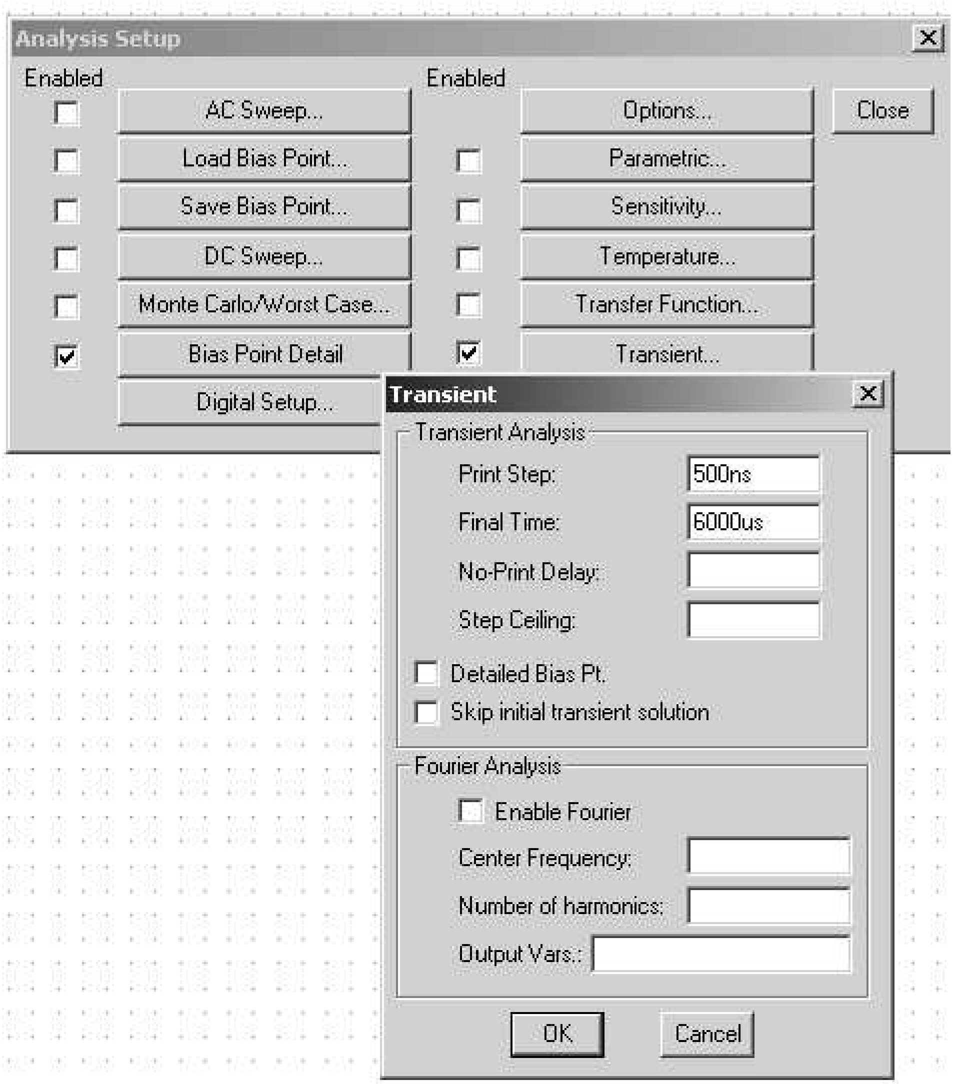
To see the resistors, independent voltage and current source, diode and bipolar transistors. Sensitivity causes a DC sensitivity analysis to be performed in which one or more output variables may be specified. You would use the sensitivity setting for discovering the maximum range of circuit performance and the causes of extreme operation. These techniques are used to identify effective changes to improve the quality of circuit operation (for example, which components need to have tight tolerance and which can be lower quality and less expensive). This isn't as important for us in the lab, but some day when you are constructing real circuits that need to function under various conditions this will be useful. The figure below is shown the sensitivity



If you want to see whether or not your circuit design is depend on the temperature. Then apply the temperature to check your circuit. Because the temperature option allows you to specify a temperature, or a list of temperatures (do not included commas between temperature values) for which PSpice will simulate your circuit. For a list of temperatures that simulation is done for each specified temperature. On your analysis setup click on the temperature and set the temperature value that wish to check. The figure below is shown how to set the temperature value.



To Plots and find the value of voltage or current versus time of the circuit then use transient. Because the transient analysis is probably the most important analysis you can run in PSpice, and it computes various values of your circuit over time. In transient analysis the two very important parameters are print step and final time. The transient analysis portion of SPICE computes the transient output variables as a function of time over a user specified time interval. The initial conditions are automatically determined by a dc analysis. All sources which are not time dependent (for example, power supplies) are set to their dc value. For large-signal sinusoidal simulations, a Fourier analysis of the output waveform can be specified to obtain the frequency domain Fourier coefficients. The ratio of final time and print step determines how many calculations PSpice must make to plot a wave forms. PSpice always defaults the start time to zero seconds and going until it reaches the user defined final time. It is incredibly important that you think about what print step you should use before running the simulation, if you make the print step too small the probe screen will be cluttered with unnecessary points making it hard to read, and taking extreme amounts of time for PSpice to calculate. However, at the opposite side of that coin is the problem that if you set the print step too high you might miss important phenomenon that are occurring over very short periods of time in the circuit. Therefore play with step time to see what works best for your circuit. You can set a step ceiling, which will limit the size of each interval, thus increasing calculation speed. Another handy feature is the Fourier analysis, which allows you to specify your fundamental frequency and the number of harmonics you wish to see on the plot. PSpice defaults to the 9th harmonic unless you specify otherwise, but this still will allow you to decompose a square wave to see it's components with sufficient detail. The figure below is shown how to setup Transient analysis.

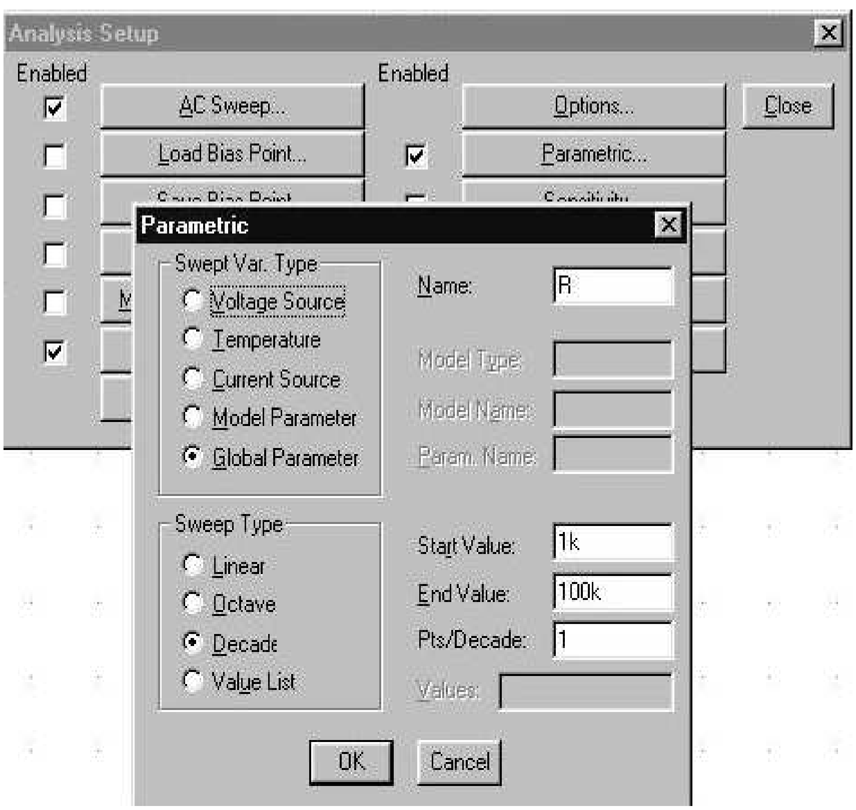


Sometime you want plots and calculated different values for a certain components on your circuit. Also plots and calculated different values on the same graph but you don't want to go back and change values again and again. You can apply the Parameter. For Example: setting up multiple values of resistor. First, Place the PARAM part on the drawing then Double -click on the part to bring up its dialog box. Under NAME1=Assign the unique name for parameter list.

(Up to 3 components can be assigned parameter lists with this one PARAM parts.) Under Value1= assign the default value, close the dialog box. Double-click on the resistor value at the resistor. Replace the value with parameter name that you assigned under Name= in PARAM dialog and enclose the name in curly braces.

Second, do the analysis set up. From menu select **analysis/setup/Parametric**. And Select **Global parameter**. Then Fill in *Model name* (the resistor "values") without the curly braces this time. If you want to use the discrete resistor value, select *value list* and fill in values in the **Values** box like 50k, 100k and 200k.

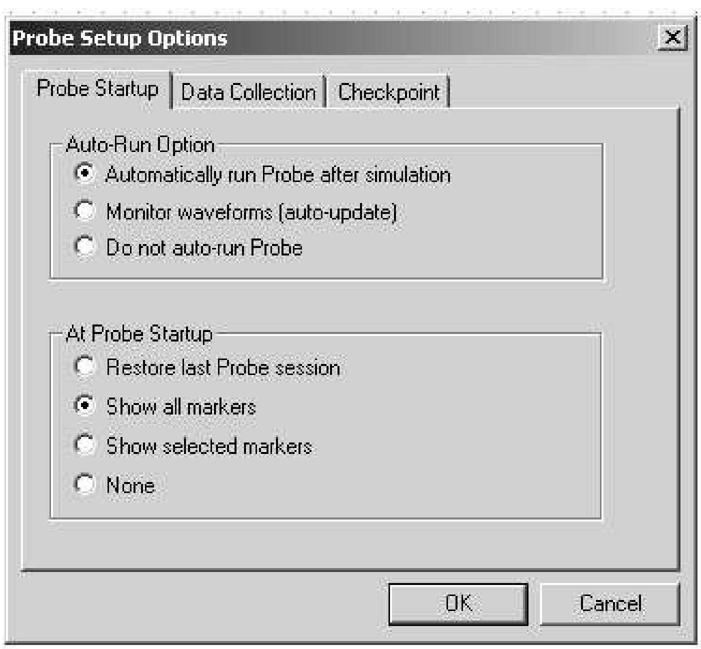
Third, Run the simulation, when you go to **Microseism probe**, you will be presented with a list of parameters and can choose to plot any or all of them. The figure below is shown how to setup the Parametric.



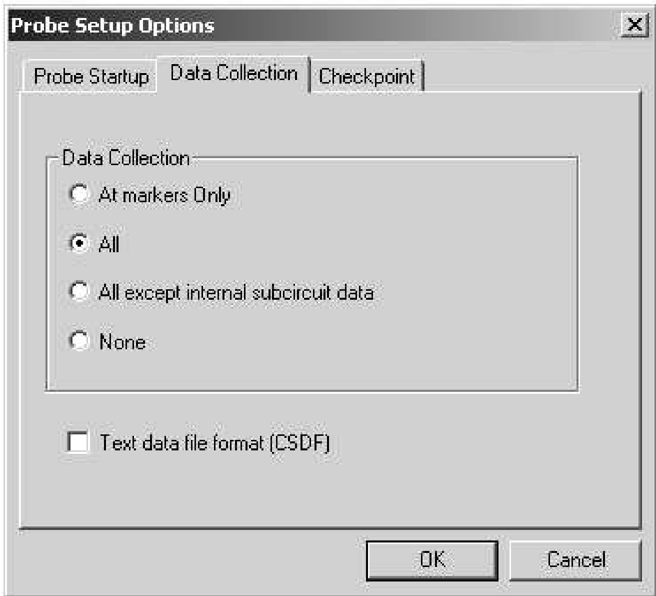
Once you have your circuit properly drawn and analysis setup then save your circuit. To save your probe you need to go into the tools menu and click display, this will open up a menu, which will allow you to name the probe file and choose where to save it. You can also open previously saved plots from here as well. Remember at the end of the name file you must have ".sch" otherwise the circuit cannot run.

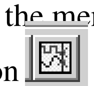
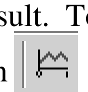
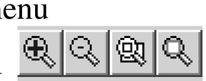
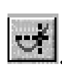
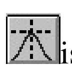
After, your circuit is saved. You should check make sure there must not be any floating parts on your page (i.e. unattached devices). You should make sure that all parts have the values that you want. There are no extra Wires and it is very important that you have a ground on your circuit. Also make sure that you have don analysis set. If all these steps are check and correct then you can run the probe.

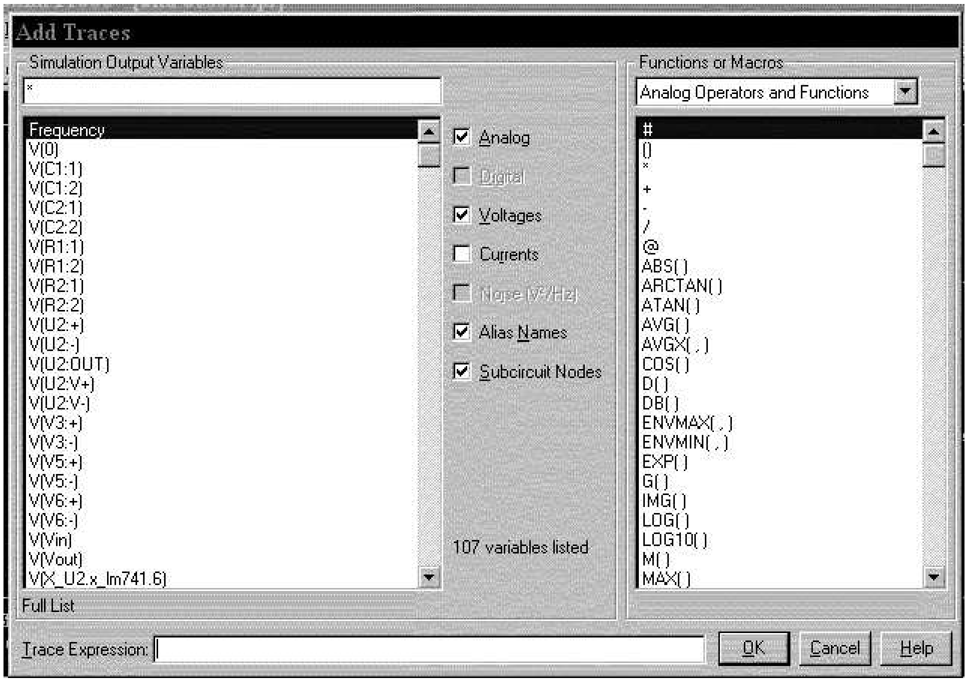
Probe is a program, which can take the data from an electrical circuit, and provide a variety of graphical displays on the users request. Also Probe is a graphical results analyzer. To tells Pspice to run probe is after performing the specified circuit analysis. Here are some steps how to setup the probe. From the menu select Analysis/probe setup and the Probe setup options window open. The figure below is shown how to setup the probe



On the Probe Setup Option selected "data collection" tab and select "All" then click "Ok". Other options allow to limited size of .dat file. The figure below is shown how to setup probe



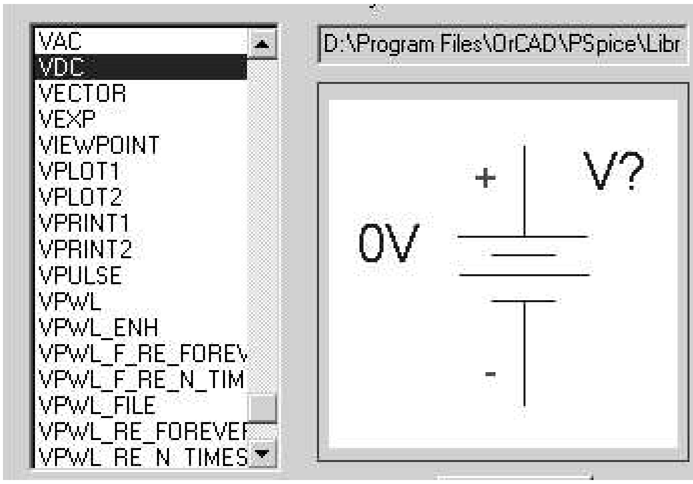
To run the simulation, either from the menu selected analysis/simulation or from keyboard uses F11 or from toolbar click on  . When your plot is done you can do some nice stuff to the plot. You can add trace, add y-axis, label in your plot result. To use Add Trace, from the menu selected Trace/add Trace or insert or from toolbar click on  button. And to reduces the trace shown by use voltage markers. If you want to add Y-axis, from menu select plot/add y-axis or from keyboard use Ctrl-Y. For display manipulation function, from menu select View/Fit/In/Out/Area or from keyboard uses Ctrl N,I,O,A or from toolbar click on  buttons. For rescale axes, Double click on any axes and plot XY axis setting. You can label your graph, from menu select plot/label/text or from the toolbar click  button. To finding point on your graph, there are Cursor buttons that allow you to find the maximum or minimum or just a point on the line. These are located on the toolbar (to the right). Select which curve you want to look at and then select "Toggle Cursor" . Then you can find the max, min, the slope, or the relative max or min (is find relative max). To doing math, in Add Traces, there are functions that can be perform, these will add/subtract (or whatever you chose) the lines together. The figure below is all the math function that you can uses.



To do the math, select the first output then either on your keyboard or on the right side, click the function that you wish to perform. There are many functions here that may or may not be useful. If you want to know how to use them, you can use PSpice's Help Menu. It is interesting to note that you can plot the phase of a value by using IP(xx), where xx is the name of the source you wish to see the phase for.

Until this moment, I assume that you understand all the sources in Pspice and how setup them. Now I want to explain all the voltage sources such as VDC, VAC, Vsin, VSRC, VPWL on the Pspice.

- VDC: is your basic direct current voltage source that simulates a simple battery and allows you to specify the voltage value. The figure below is shown the VDC



- VAC: The ac small-signal portion of SPICE computes the ac output variables as a function of frequency. The program first computes the dc operating point of the circuit and determines linear zed, small-signal models for all of the nonlinear devices in the circuit. The resultant linear circuit is then analyzed over a user-specified range of frequencies. The desired output of an ac small-signal analysis is usually a transfer function (voltage gain, Tran impedance, etc). If the circuit has only one ac input, it is convenient to set that input to unity and zero phases, so that output variables have the same value as the transfer function of the output variable with respect to the input.

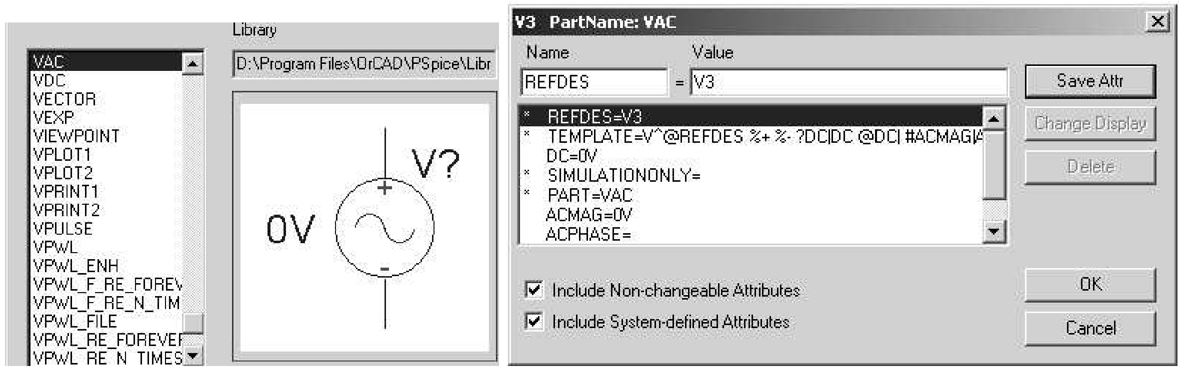
A few things to note about the alternating current source, first PSpice takes it to be a sine source, so if you want to simulate a cosine wave you need to add (or subtract) a 90° phase shift. There are three values which PSpice will allow you to alter, these being:

ACMAG, which is the RMS value of the voltage.

DC, which is the DC, offset voltage.

ACPHASE, which is the phase angle of the voltage

Note that the phase angle if left unspecified will be set by default to 0°. The figure below is show VAC source.



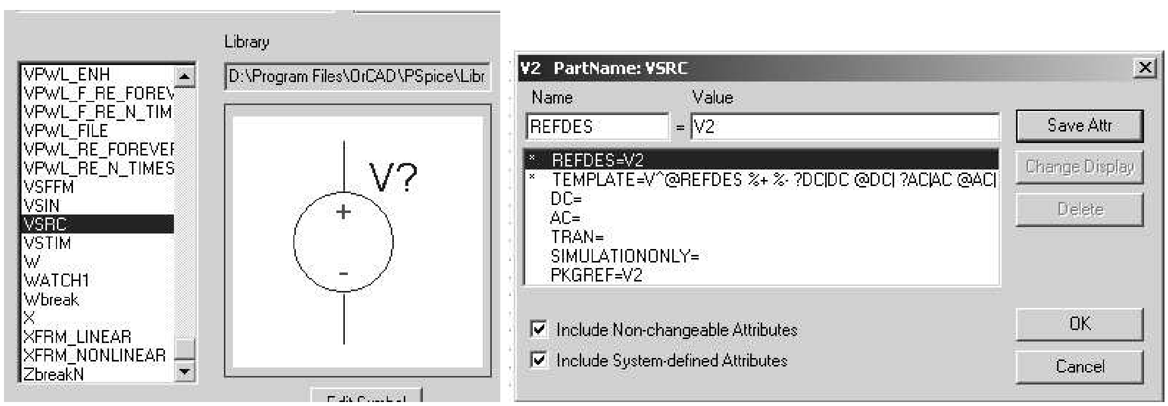
- VSRC is a simple voltage source

DC: Only for DC bias point analysis

AC: AC sweep simulation; the amplitude of the sine wave

Tran: for Transient analysis, voltage

The figure below is shown how to get the VSRC.



VSIN is sinusoidal voltage source. The SIN type of source is actually a damped sine with time delay, phase shift and a DC offset. If you want to run a transient analysis you need to use the VSIN see how AC will affect your circuit over time. Do not use this type of source for a phase or frequency sweep analysis; VAC would be appropriate for that.

DC: Only for DC bias point analysis or DC component of sin wave

AC: AC sweep simulation; the amplitude of the sine wave or the value of sin wave

VOFF: for Transient analysis, voltage offset or the DC offset value. It should be set to zero if you need a pure sinusoid.

VAMPL: for Transient analysis, voltage amplitude and it is the under-damped amplitude of the sinusoid; i.e., the peak value measured from zero if there were no DC offset value

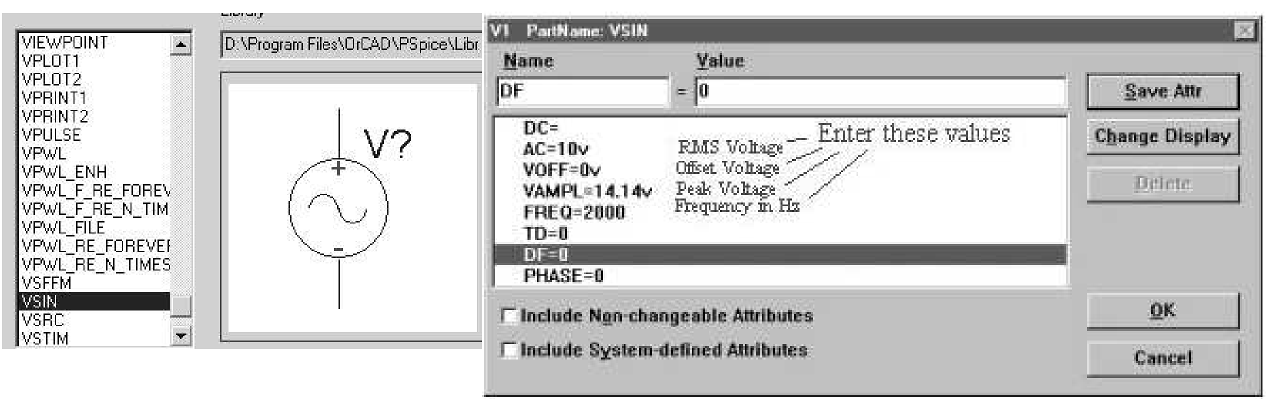
FREQ: for Transient analysis, frequency and is the frequency in Hz of the sinusoid

TD: for Transient analysis, Time delay for starting sine the wave and is the time delay in seconds. Set this to zero for the normal sinusoid.

DF: for Transient analysis, Distortion factor for the negative going amplitude of the sine wave. Or is the damping factor. Also set this to zero for the normal sinusoid.

Phase: is the phase advance in degrees. Set this to 90 if you need a cosine waveform.

The normal usage of this source type is to set **Voff**, **TD** and **DF** to zero, as this will give you a 'nice' sine wave. The Figure below is shown how to get Vsin and setup Vsin



-- VPULSE (Pulsed voltage source). The VPULSE is often used for a transient simulation of a circuit where we want to make it act like a square wave source. It should never be used in a frequency response study because PSpice assumes it is in the time domain, and therefore your probe plot will give you inaccurate results.

DC: Only for DC bias point analysis or the DC component of the wave

AC: AC sweep simulation; the amplitude of the sine wave and the AC component of the wave

V1: the voltage at the bottom of the pulse and is the value when the pulse is not "on." So for a square wave, the value when the wave is 'low'. This can be zero or negative as required. For a pulsed current source, the units would be "amps" instead of "volts."

V2: the voltage at the top of the pulse and the value when the pulse is fully turned 'on'. This can also be zero or negative. (Obviously, V1 and V2 should not be equal.) Again, the units would be "amps" if this were a current pulse.

FREQ: for Transient analysis, frequency

TD: for Transient analysis, Time delay between t=0 and the start of the pulse and the time delay. The default units are seconds. The time delay may be zero, but not negative.

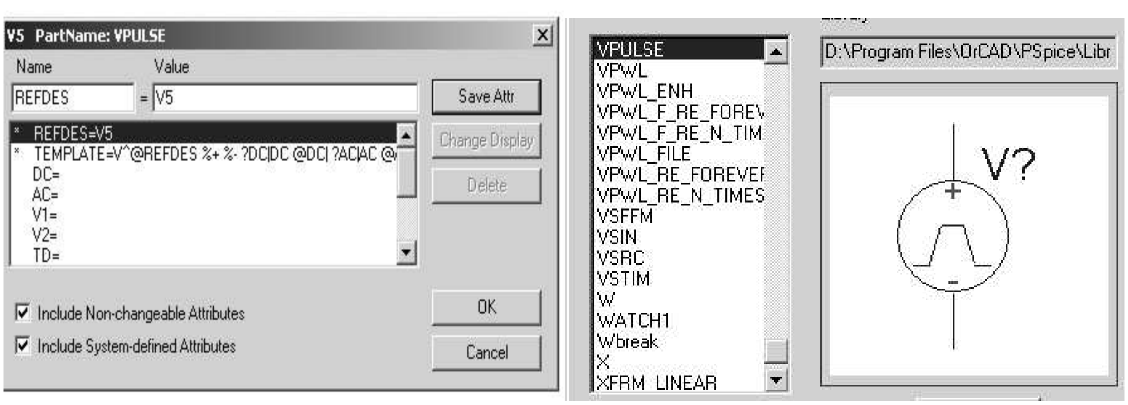
TR: rise time of the pulse and the rise time of the pulse. PSpice allows this value to be zero, but zero rise time may cause convergence problems in some transient analysis simulations. The default units are seconds.

TF: the fall time of the pulse or the fall time in seconds of the pulse.

PW: is the pulse width, width of the top of the pulse. This is the time in seconds that the pulse is fully on.

PER: is the period and is the total time in seconds of the pulse.

This is a very important source for us because we do a lot of work on with the square wave on the wave generator to see how various components and circuits respond to it. The figure below is the VPULSE source.

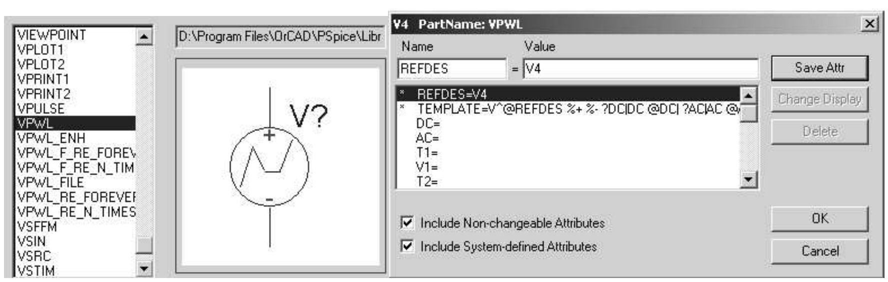


- VPWL (Piecewise linear voltage source): The PWL source is a Piece Wise Linear function that you can use to create a wave form consisting of straight line segments drawn by linear interpolation between points that you define. Since you can use as many points as you want, you can create a very complex waveform this source type can be a voltage source or a current source. The syntax for this source type is flexible and has several optional parameters. The required parameters are two-dimensional points consisting of a time value and a voltage (or current) value. There can be many of these data pairs, but the time values must be in ascending order, and the intervals between time values need not be regular. The two optional parameters are "DC" and "AC." The use of an AC parameter with this source is not very well documented and because this source is intended for use with a transient analysis any AC value would be ignored.

DC: Only for DC bias point analysis

AC: AC sweep simulation; the amplitude of the sine wave

The other parameters consist of time and voltage pairs, T1, V1, T2, V2, ect that represents points, which are connected by straight line from the waveform. The figure below is shown the VPWL.

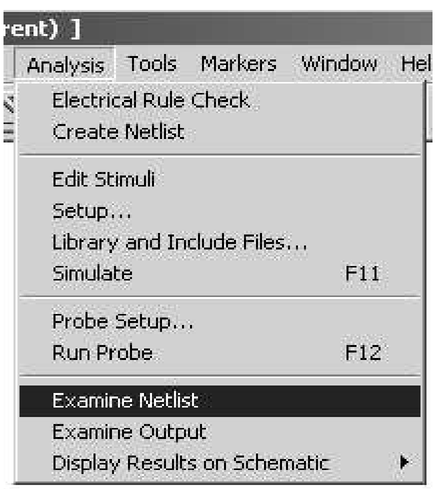


For any of the previous discussed voltage sources, there exist the exact source except that they produce current. There is one thing that should be mentioned; current sources in PSpice get a little confusing. For those current sources whose circuit symbol has an arrow, you have to point the arrow in the direction of conventionally flowing current. This applies to all current sources, including AC and DC. Therefore placing the current source in the circuit backwards with seemingly incorrect polarities will give the correct results.

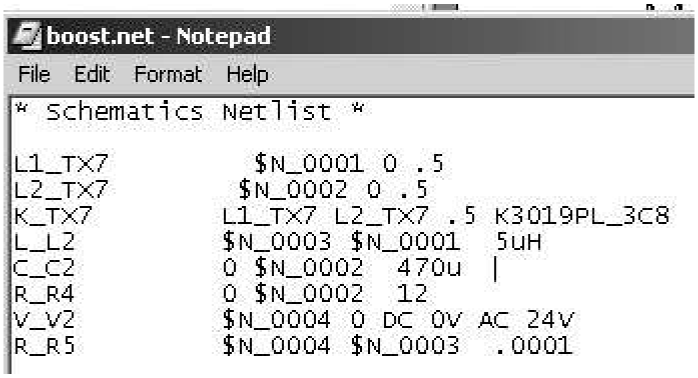
An interesting little feature under the **markers** menu is the ability to add markers to your circuit so you can see where the current and voltage have imaginary values in the circuit, and the phase of your source at any point in the circuit.

The dc analysis portion of SPICE determines the dc operating point of the circuit with inductors shorted and capacitors opened. A dc analysis is automatically performed prior to a transient analysis to determine the transient initial conditions, and prior to an ac small-signal analysis to determine the linear zed, small-signal models for nonlinear devices. If requested, the dc small-signal value of a transfer function (ratio of output variable to input source), input resistance, and output resistance will also be computed as a part of the dc solution.

It is a simple text editor where you will type your netlist and save it as a file under the extension *.cir. The netlist is a description of the circuit on paper, without a diagram. You don't need to bother about this. To Check or create your own netlist, from your menu select analysis and either select create netlist or Examine netlist. The figure below is shown how to select the netlist file.



After you run a simulation of your circuit, Pspice created it own netlist. To open your netlist file simply click on the Examine Netlist and the notepad will open. The file had your circuit file name.net like the figure show below. In this file described all the values and parts of your circuit however, not the node of your circuit. The figure below is an example of netlist file for a boost circuit.



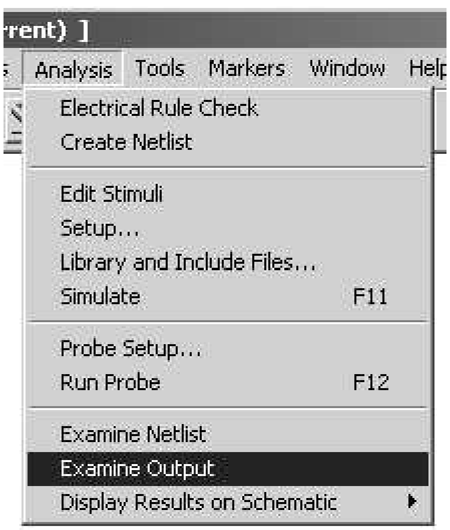
boost.net - Notepad

File Edit Format Help

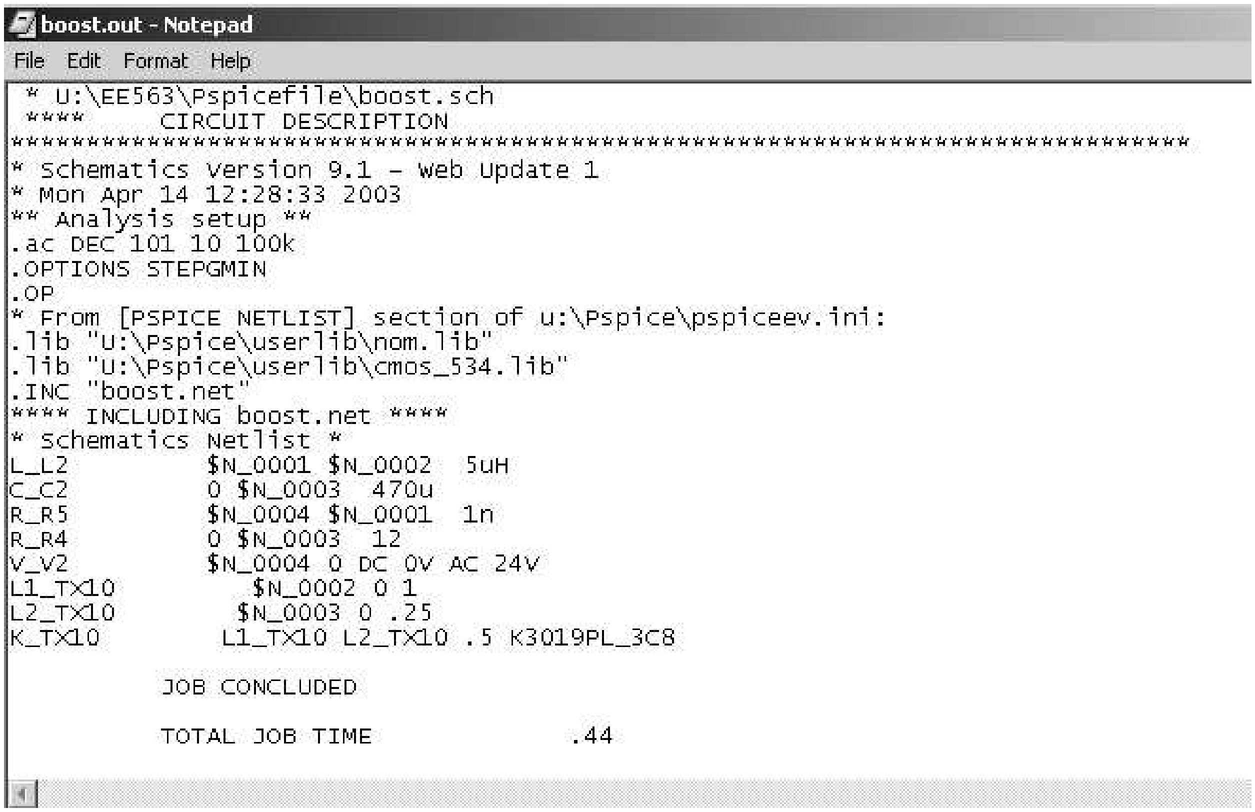
```
* Schematics Netlist *

L1_TX7          $N_0001 0 .5
L2_TX7         $N_0002 0 .5
K_TX7          L1_TX7 L2_TX7 .5 K3019PL_3C8
L_L2           $N_0003 $N_0001  5uH
C_C2           0 $N_0002  470u  |
R_R4           0 $N_0002  12
V_V2           $N_0004 0 DC 0V AC 24V
R_R5           $N_0004 $N_0003  .0001
```

To check all values, nodes and parts of your circuit, on your menu select analysis and click on the examine output. The figure below is shown how to open the examine output file.



After you selected on Examine output, the notepad file will open. The notepad file had your circuit name *.out. This file described all the nodes, values, parts and circuit description of your design. And when you run a simulation of your circuit, what temperature that you set for your circuit. At what frequency, which you run simulate of your circuit. Also shown how long it takes to complete simulation of your circuit. The figure below is the example of the output file for a boost circuit.



boost.out - Notepad

File Edit Format Help

```
 * U:\EE563\Pspicefile\boost.sch
 ****     CIRCUIT  DESCRIPTION
******************************************************************************
* Schematics Version 9.1 - Web Update 1
* Mon Apr 14 12:28:33 2003
** Analysis setup **
.ac DEC 101 10 100k
.OPTIONS STEPGMIN
.OP
* From [PSPICE NETLIST] section of u:\Pspice\pspiceev.ini:
.lib "U:\Pspice\userlib\nom.lib"
.lib "U:\Pspice\userlib\cmos_534.lib"
.INC "boost.net"
**** INCLUDING boost.net ****
* Schematics Netlist *
L_L2          $N_0001 $N_0002  5uH
C_C2          0 $N_0003  470u
R_R5          $N_0004 $N_0001  1n
R_R4          0 $N_0003  12
V_V2          $N_0004 0 DC 0V AC 24V
L1_TX10          $N_0002 0 1
L2_TX10         $N_0003 0 .25
K_TX10         L1_TX10 L2_TX10 .5 K3019PL_3C8

          JOB CONCLUDED

          TOTAL JOB TIME            .44
```

After spend all this time to explain how to create and run a simulation of a circuit by using Pspice software. I think it is time to apply all the knowledge and understand of Pspice into action. Let design SEPIC, Boost, buck and Buck-boost converter and then run a simulation of these circuit. The circuits and values are given in the Fundamental of Power Electronics 2nd edition textbook. Uses Pspice Student Version, which installed or setup in your own account (u drive) to design or create these circuits.

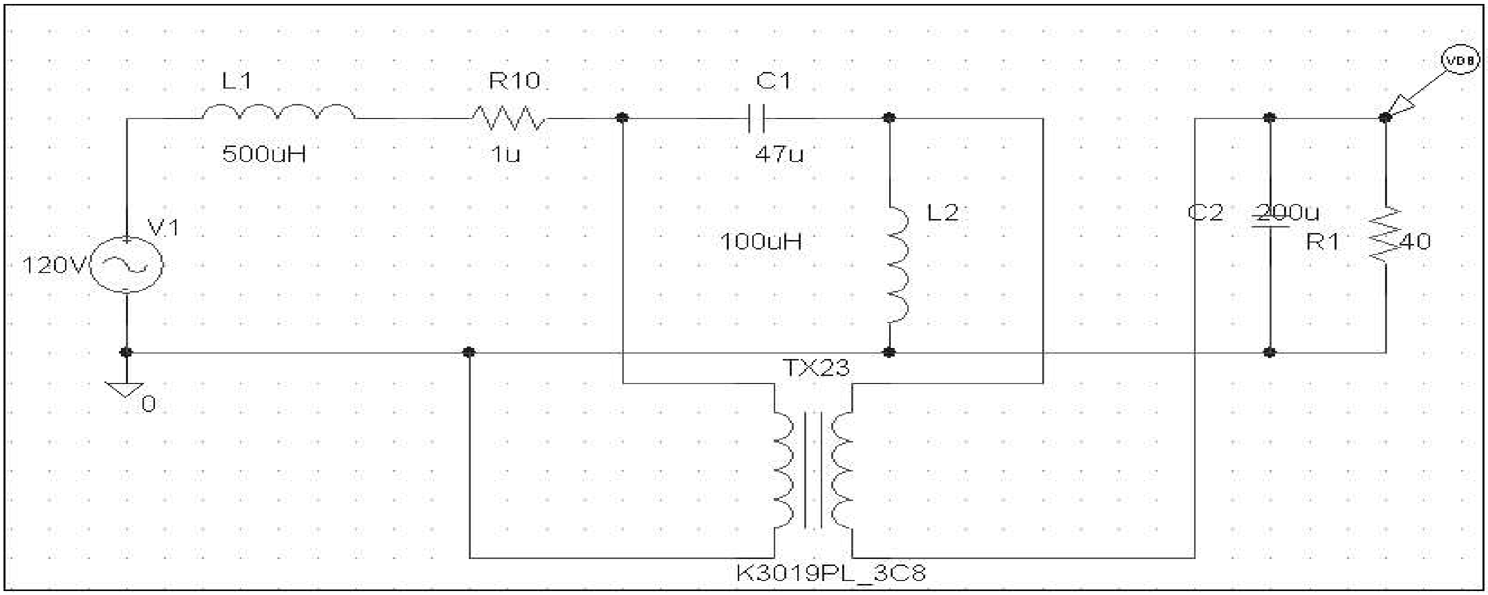
The figure below is a circuit of SEPIC, this circuit and values are given in the Appendix B of the Fundamental of Power Electronics 2nd edition textbook. The values given are:

L1=500uH L2=100uH

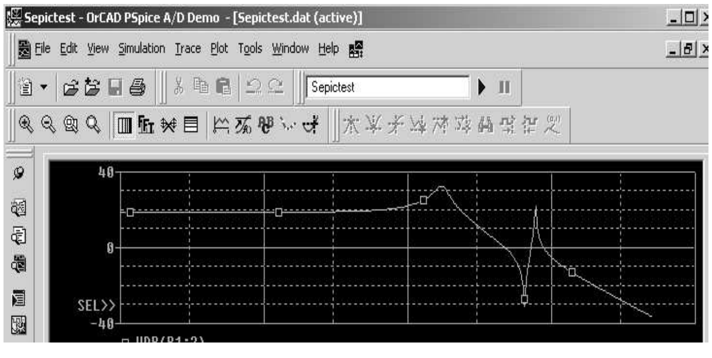
RL=1u RL=40

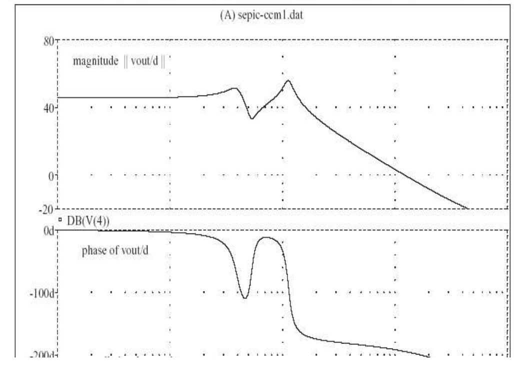
C1=47uF C2=200uF

Plot the transfer function and the phase of this circuit.

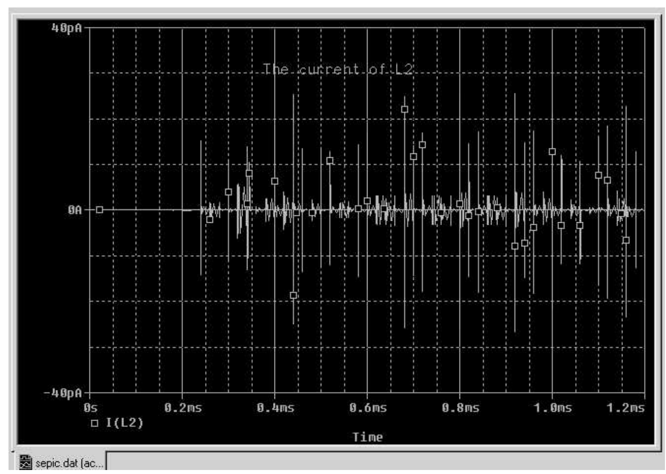


Do to the limited of Pspice student version; we cannot design a sub-circuit to implementation of the combined CCM/DCM averaged switch model. And we cannot apply the library given in the Appendix B of the Fundamental of Power Electronics 2nd edition textbook to solve this circuit. But the question is plots the transfer function and the phase of SEPIC circuit. Therefore, apply what we learn from the class in Chapter 7, 8 and 11, which replace the switch transistor and diode by a coupling induction and the coupling inductor will become our averaged switch network. Now, put this ideal into task, Design a same circuit as the fig. B.13 in the Appendix B of the Fundamental of Power Electronics 2nd edition textbook but replace the CCM/DCM sub-circuit with a coupling inductor. Saved and run the simulation of the circuit the result of the frequency response is shown in the figure below. For this question I only do for the CCM case. As you can see the result of my circuit is the same as the Fig B.14 of in the Appendix B of the Fundamental of Power Electronics 2nd edition textbook.

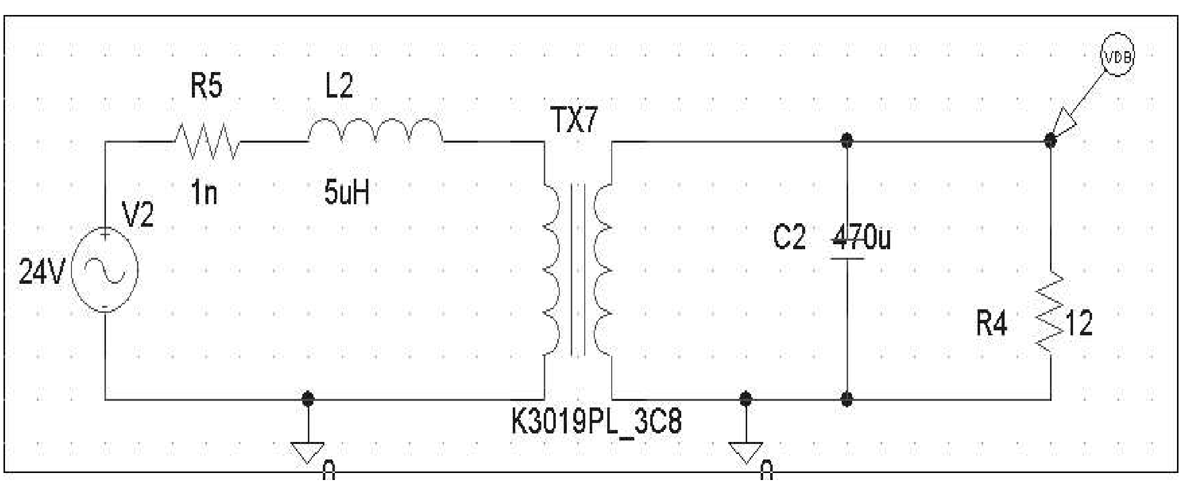




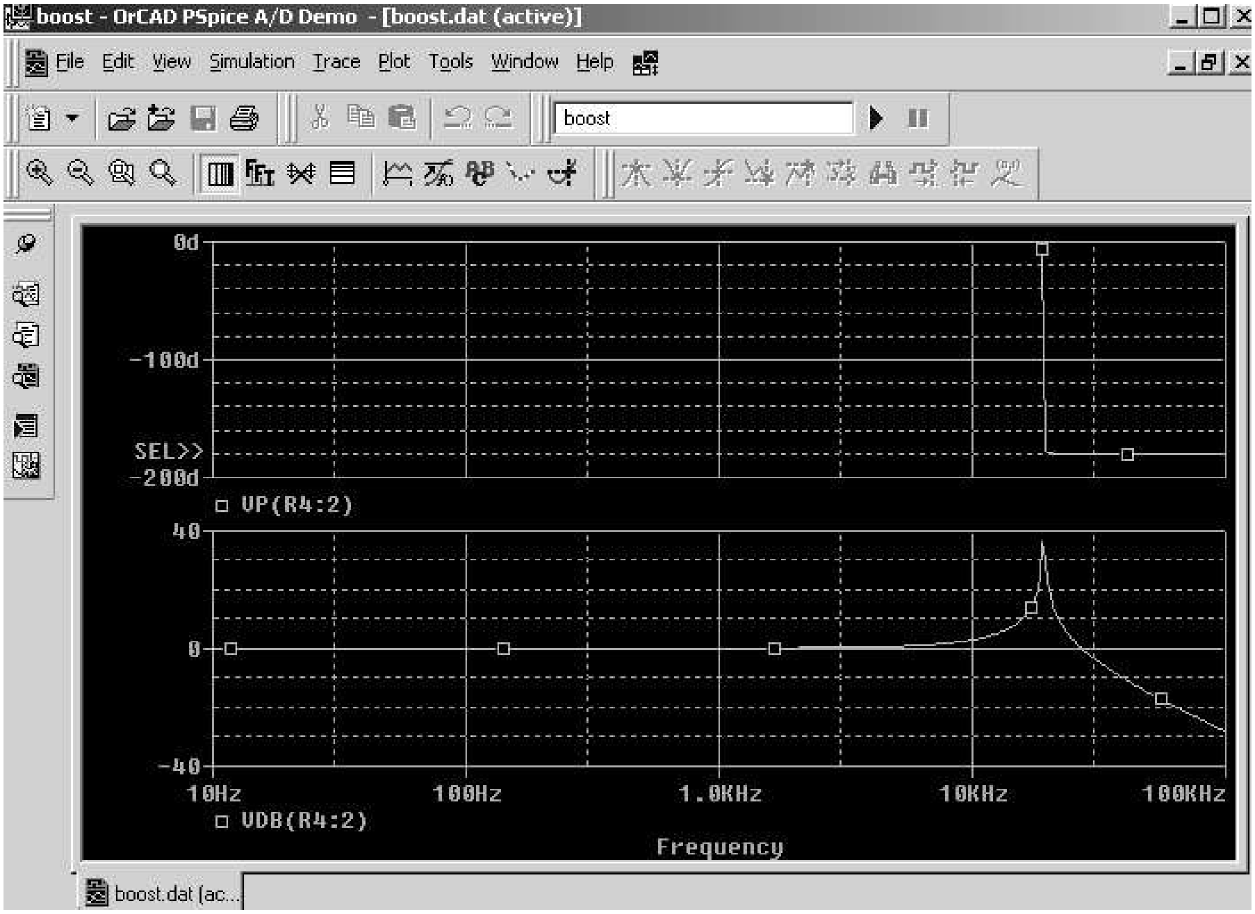
The figure below is the inductor current and the output voltage of Sepic converter



Apply the see idea as above, I design a boost circuit and plot it frequency response. I use the values that given in chapter example 8.2.1 of the Fundamental of Power Electronics 2nd edition textbook. The figure below is boost circuit



The figure below is shown the result of my boost circuit.



To proved that the result of my circuit by using Pspice is correct. I used MathCAD software to calculate and plot the frequency response. The following is the MathCAD calculation and plot of the transfer function and phase of the boost circuit.

$R := 12 \qquad L := 5 \cdot 10^{-6} \qquad C := 470 \cdot 10^{-6} \qquad fs := 100 \cdot 10^{3}$

$Vg := 24 \qquad Vo := 36 \qquad I := 3$

$P := I \cdot (Vo - Vg)$

$$Re := \frac{Vg^2}{P}$$

$$Ts := \frac{1}{fs}$$

$$D := \sqrt{\frac{2 \cdot L}{Re \cdot Ts}}$$

$$M := \frac{Vo}{Vg} \qquad M = 1.5$$

$$Gdo := \frac{2 \cdot Vo}{D} \cdot \frac{M - 1}{2 \cdot M - 1} \qquad Gdo = 72$$

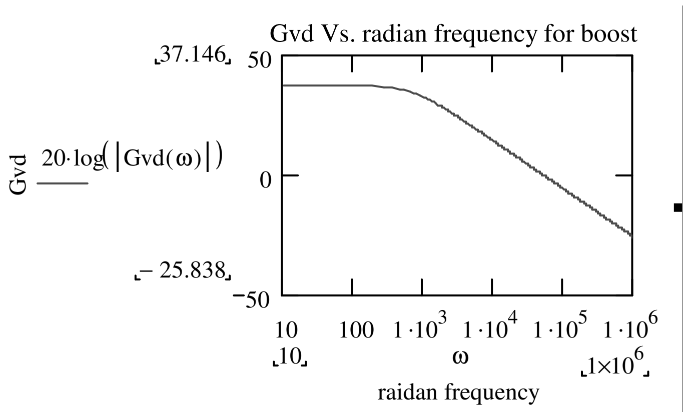
$20 \cdot \log(Gdo) = 37.147$

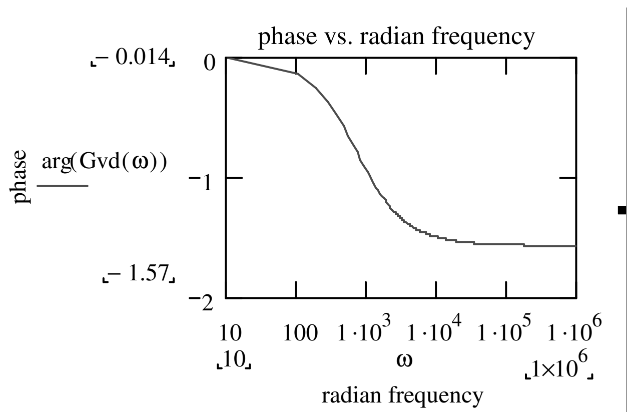
$j := \sqrt{-1}$

$$fp := \frac{2 \cdot M - 1}{2\pi \cdot (M - 1) \cdot R \cdot C} \qquad fp = 112.876$$

$\omega := 10, 100 .. 100 \cdot 10^{4}$

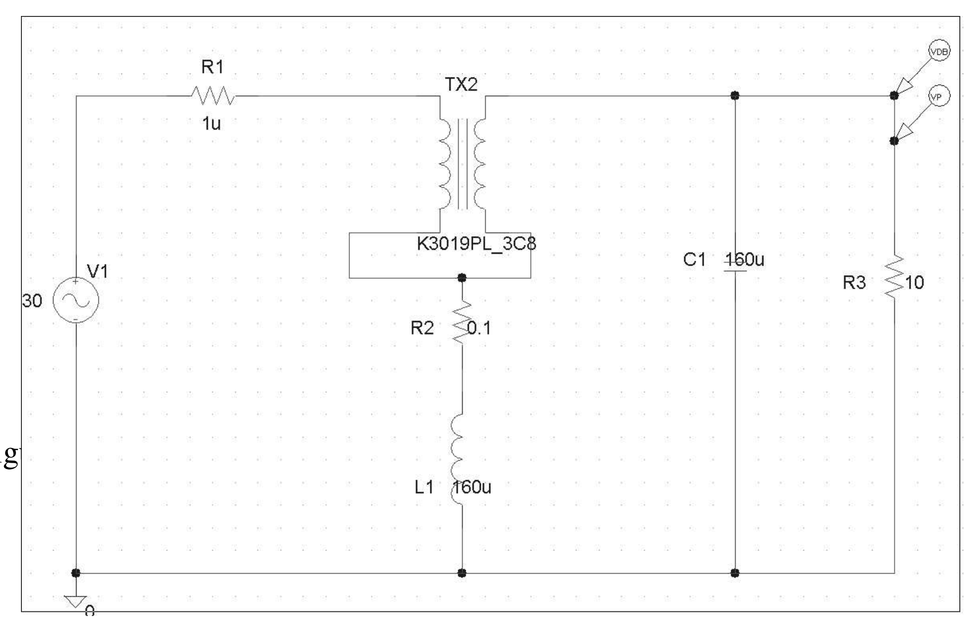
$$Gvd(\omega) := \frac{Gdo}{1 + \frac{j \cdot \omega}{2 \cdot \pi \cdot fp}}$$



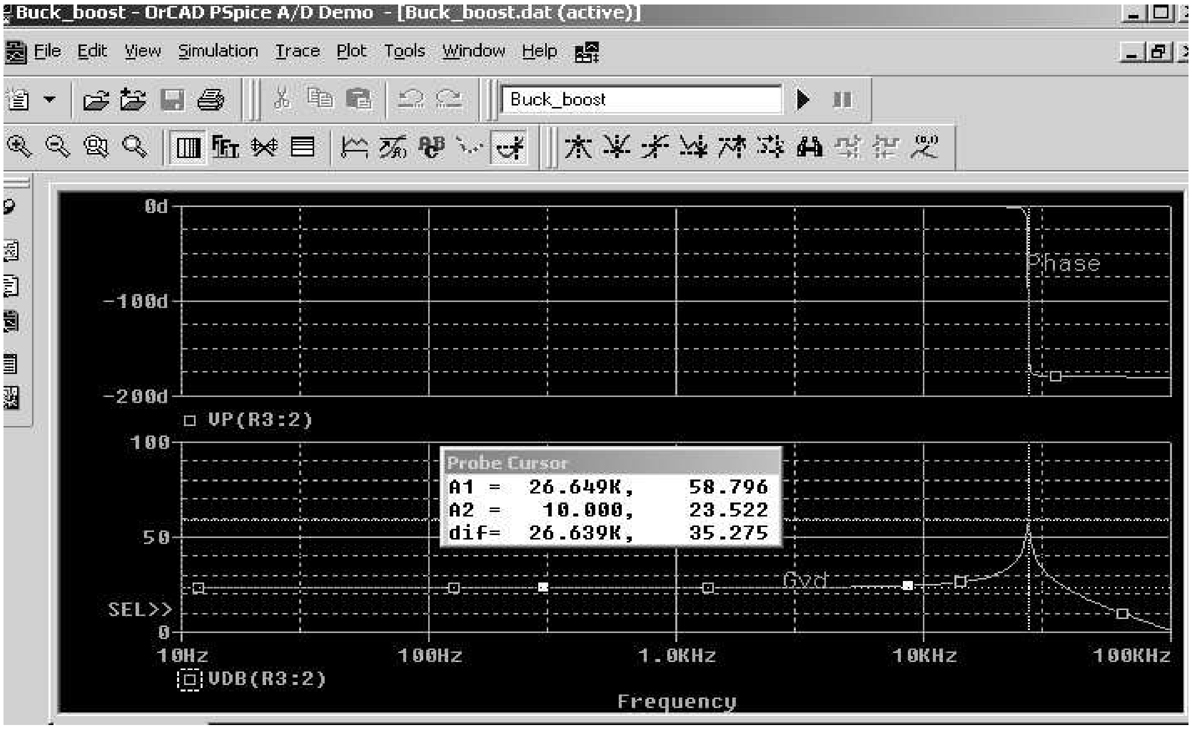


As you can see the answer of the boost circuit that provided by MathCAD and Pspice software are very close. However, the Pspice simulation is plot all the behavior of the circuit and MathCAD only estimate and calculate the value of the circuit. Therefore, the plots not look exactly the same but the answers are very close.

Next the Buck-Boost circuit, again use the same idea as the other two circuits. I used student version of Pspice to build a Buck-boost circuit. The figure below is a Buck-Boost circuit.



The fig



Again, in order to proved that the result of my circuit by using Pspice is correct. I used MathCAD software to calculate and plot the frequency response. The following is the MathCAD calculation and plot of the transfer function and phase of the Buck- boost circuit.

$R := 10 \qquad L := 160\cdot10^{-6} \qquad C := 160\cdot10^{-6} \qquad Ts := 10\cdot10^{-6}$

$Vg := 30 \qquad D := 0.6$

Calculate the output voltage

Basic formula

$V := \frac{-D}{(1-D)}\cdot Vg \qquad V = -45$

$I := \frac{-V}{(1-D)\cdot R} \qquad I = 11.25$

$Ggo := \frac{D}{1-D} \qquad Ggo = 1.5$

$20\cdot\log(Ggo) = 3.522$

$Re := \frac{2\cdot L}{D^2\cdot Ts} \qquad Re = 88.889$

$fs := \frac{1}{Ts} \qquad fs = 1\times10^{5}$

$Gdo := \frac{|V|}{D\cdot(1-D)} \qquad Gdo = 187.5$

$20\cdot\log(Gdo) = 45.46$

$\omega z := \frac{(1-D)^2\cdot R}{D\cdot L} \qquad \omega z = 1.667\times10^{4}$

$fz := \frac{\omega z}{2\pi} \qquad fz = 2.653\times10^{3}$

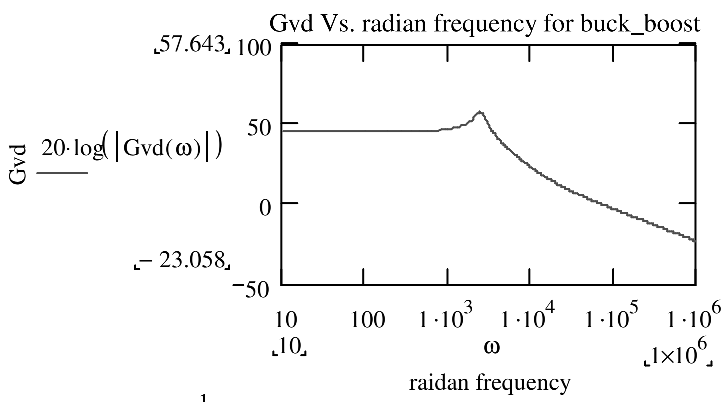
$j := \sqrt{-1}$

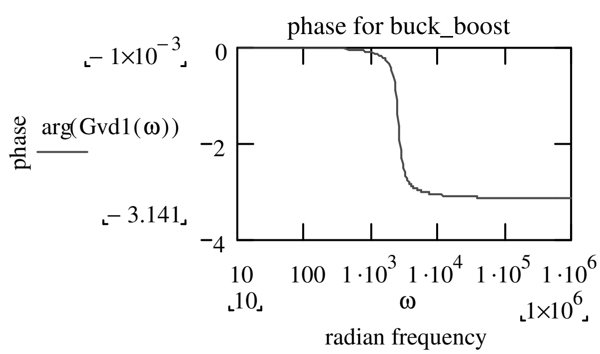
$Q := (1 - D)\cdot R\cdot\sqrt{\dfrac{C}{L}}$ $\qquad Q = 4$

$fo := \dfrac{(1 - D)}{2\pi\cdot\sqrt{L\cdot C}}$ $\qquad fo = 397.887$

$\omega := 10, 100 .. 100\cdot 10^4$

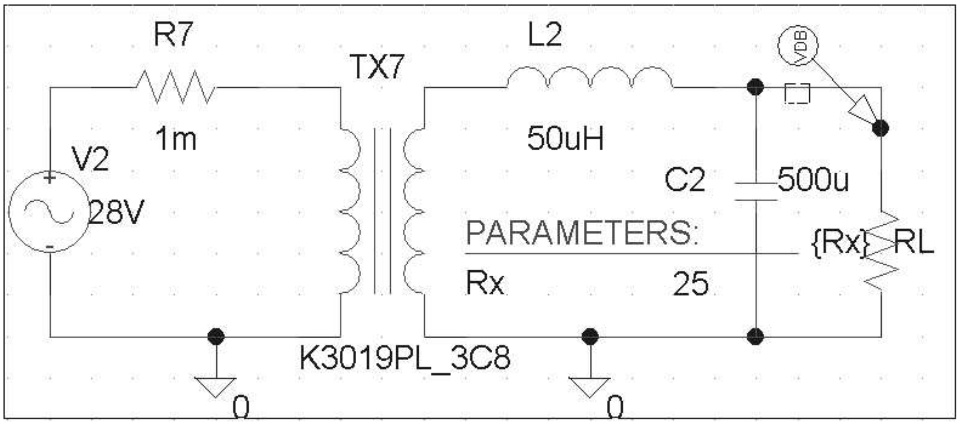
$$Gvd(\omega) := Gdo\cdot\frac{\left(1 - \dfrac{j\cdot\omega}{\omega z}\right)}{\left[1 + \dfrac{j\cdot\omega}{Q\cdot 2\pi fo} + \left(\dfrac{j\cdot\omega}{2\pi fo}\right)^2\right]}$$



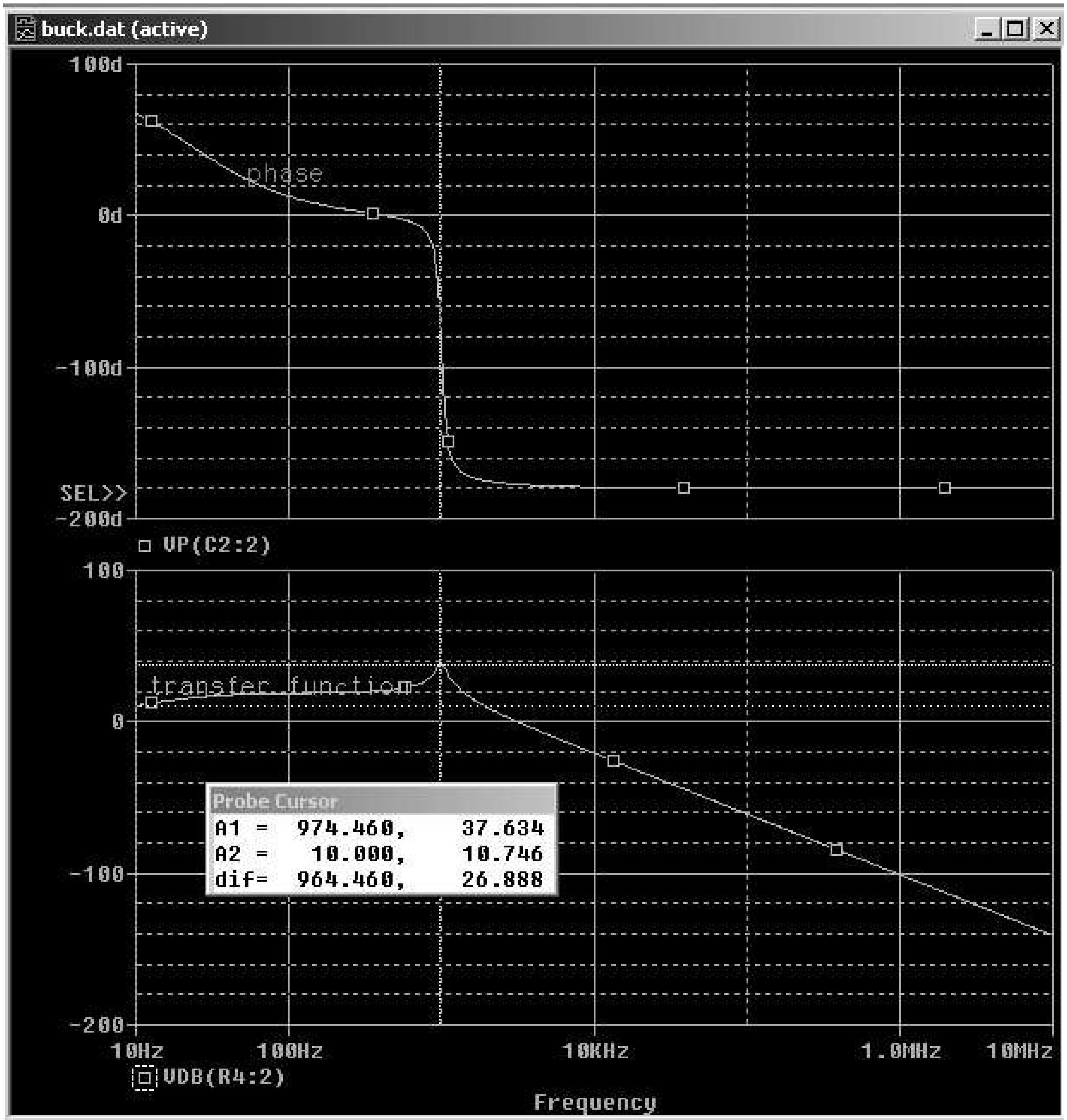


As you can see that the results in both MathCAD and Pspice software are the same, which prove that the result of my buck-boost using Pspice is correct.

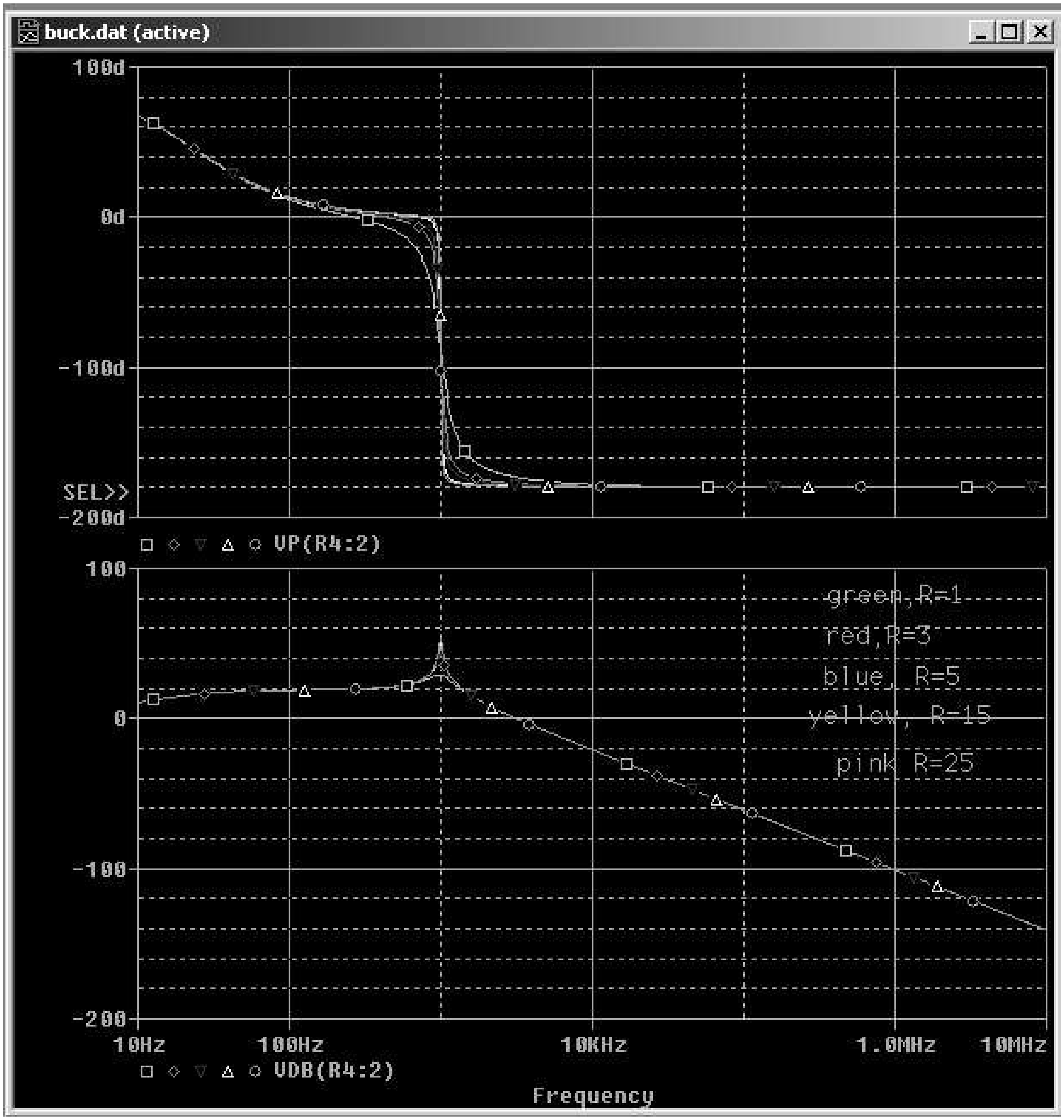
Apply the same idea and buck example in page 355 of the Fundamental of Power Electronics 2nd edition textbook, I design a buck converter and uses the given values in the example to test my design. The figure below is buck converter.



The figure below is the results of buck converter



buck.dat (active)
100d
phase
0d
-100d
SEL>>
-200d
VP(C2:2)
100
transfer function
0
Probe Cursor
A1 = 974.460, 37.634
A2 = 10.000, 10.746
dif= 964.460, 26.888
-100
-200
10Hz
100Hz
10KHz
1.0MHz
10MHz
VDB(R4:2)
Frequency



buck.dat (active)
100d
0d
-100d
SEL>>
-200d
VP(R4:2)
100
0
-100
-200
green, R=1
red, R=3
blue, R=5
yellow, R=15
pink R=25
10Hz
100Hz
10KHz
1.0MHz
10MHz
VDB(R4:2)
Frequency

The calculation of Buck converter

$Vg := 28$ $\quad V := 15$

$fs := 100 \cdot 10^{3}$

$C := 500 \cdot 10^{-6}$

$L := 50 \cdot 10^{-6}$

$R := 3$

$D := \frac{V}{Vg}$ $\quad D = 0.536$

$j := \sqrt{-1}$

$\omega := 1, 10 .. 100 \cdot 10^{3}$

$$Gvd(\omega) := \frac{V}{D} \cdot \frac{1}{1 + j \cdot \omega \cdot \frac{L}{R} + (j \cdot \omega)^{2} \cdot L \cdot C}$$

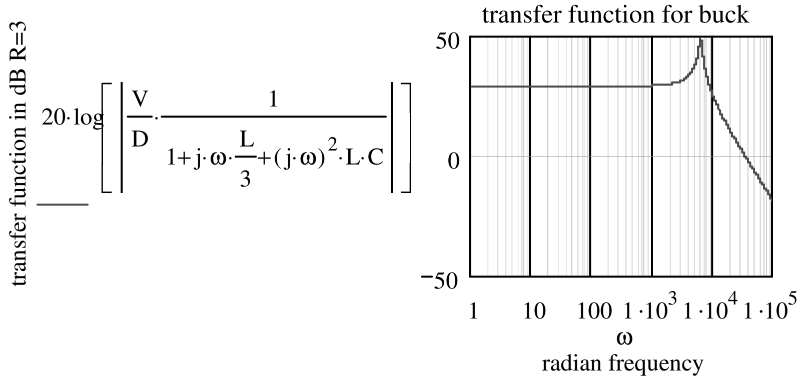
$$Gvd(\omega) := \frac{V}{D} \cdot \frac{1}{1 + j \cdot \omega \cdot \frac{L}{3} + (j \cdot \omega)^{2} \cdot L \cdot C}$$

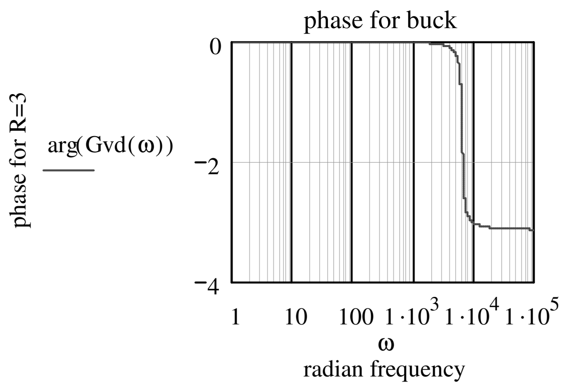
$$Gvd4(\omega) := \frac{V}{D} \cdot \frac{1}{1 + j \cdot \omega \cdot \frac{L}{1} + (j \cdot \omega)^{2} \cdot L \cdot C}$$

$$Gvd1(\omega) := \frac{V}{D} \cdot \frac{1}{1 + j \cdot \omega \cdot \frac{L}{5} + (j \cdot \omega)^{2} \cdot L \cdot C}$$

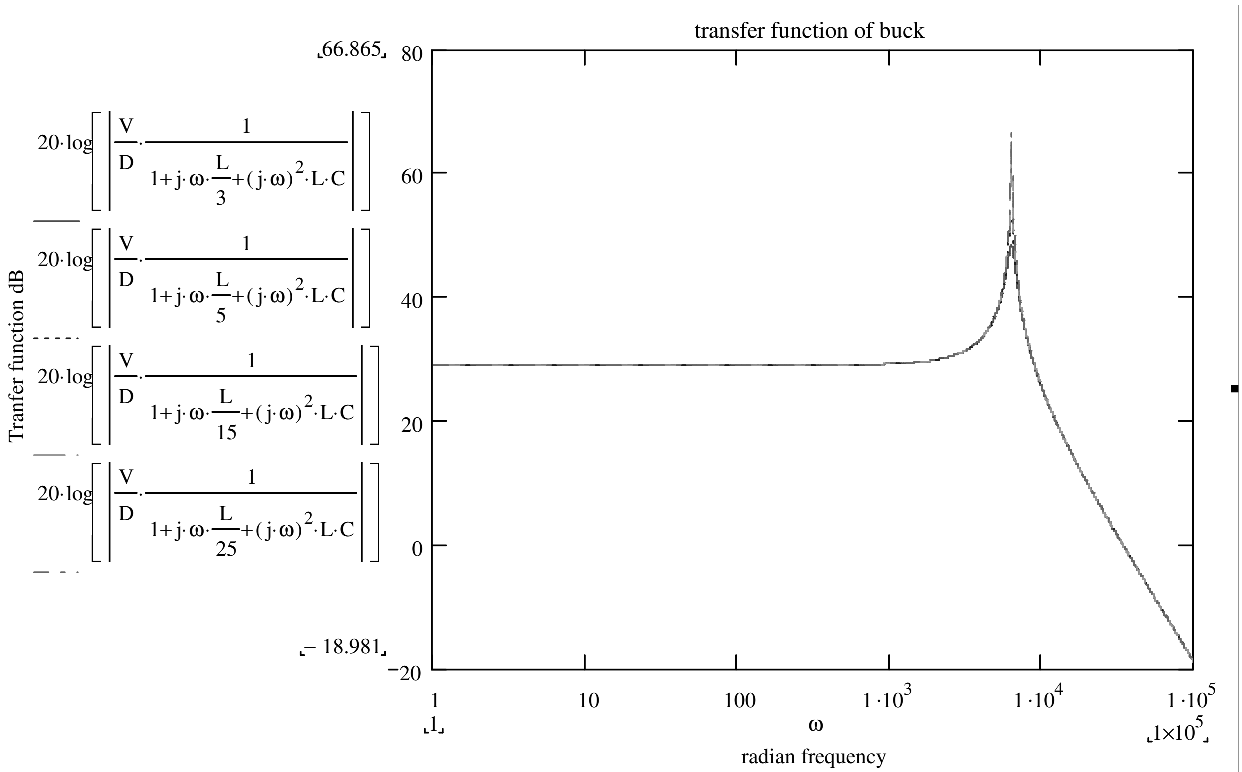
$$Gvd3(\omega) := \frac{V}{D} \cdot \frac{1}{1 + j \cdot \omega \cdot \frac{L}{25} + (j \cdot \omega)^{2} \cdot L \cdot C}$$

$$Gvd2(\omega) := \frac{V}{D} \cdot \frac{1}{1 + j \cdot \omega \cdot \frac{L}{15} + (j \cdot \omega)^{2} \cdot L \cdot C}$$

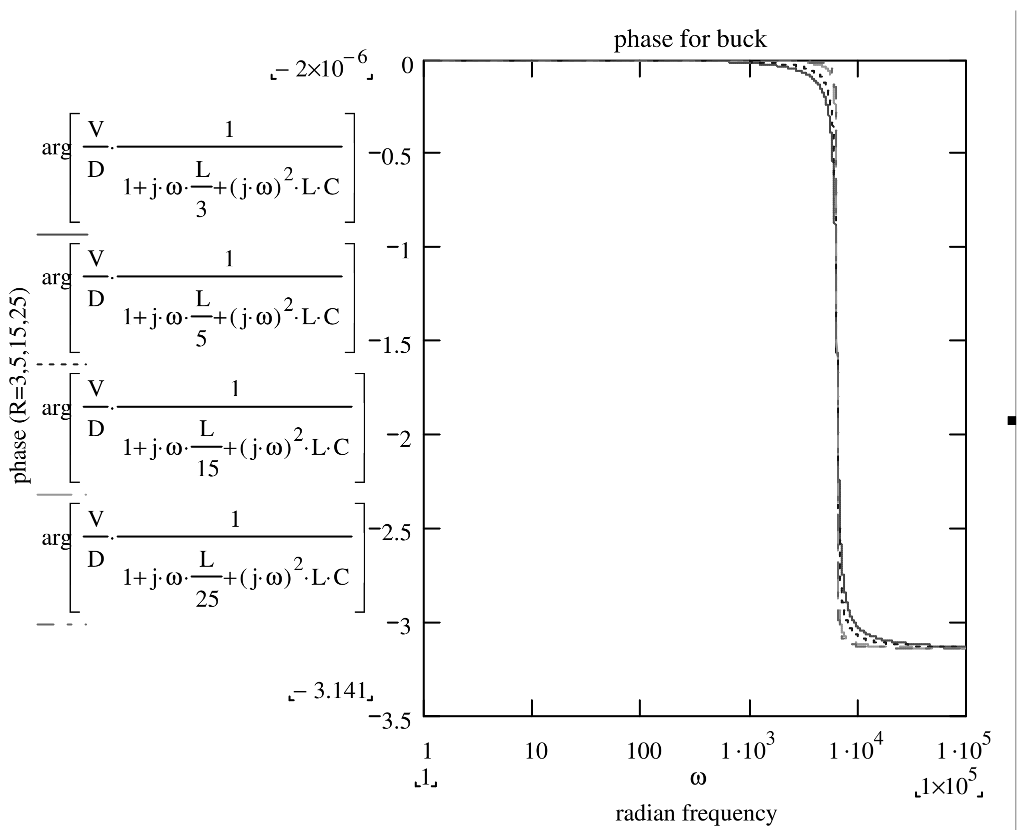




The transfer function and phase plots of buck converter for R = 3, 5, 15 and 25.

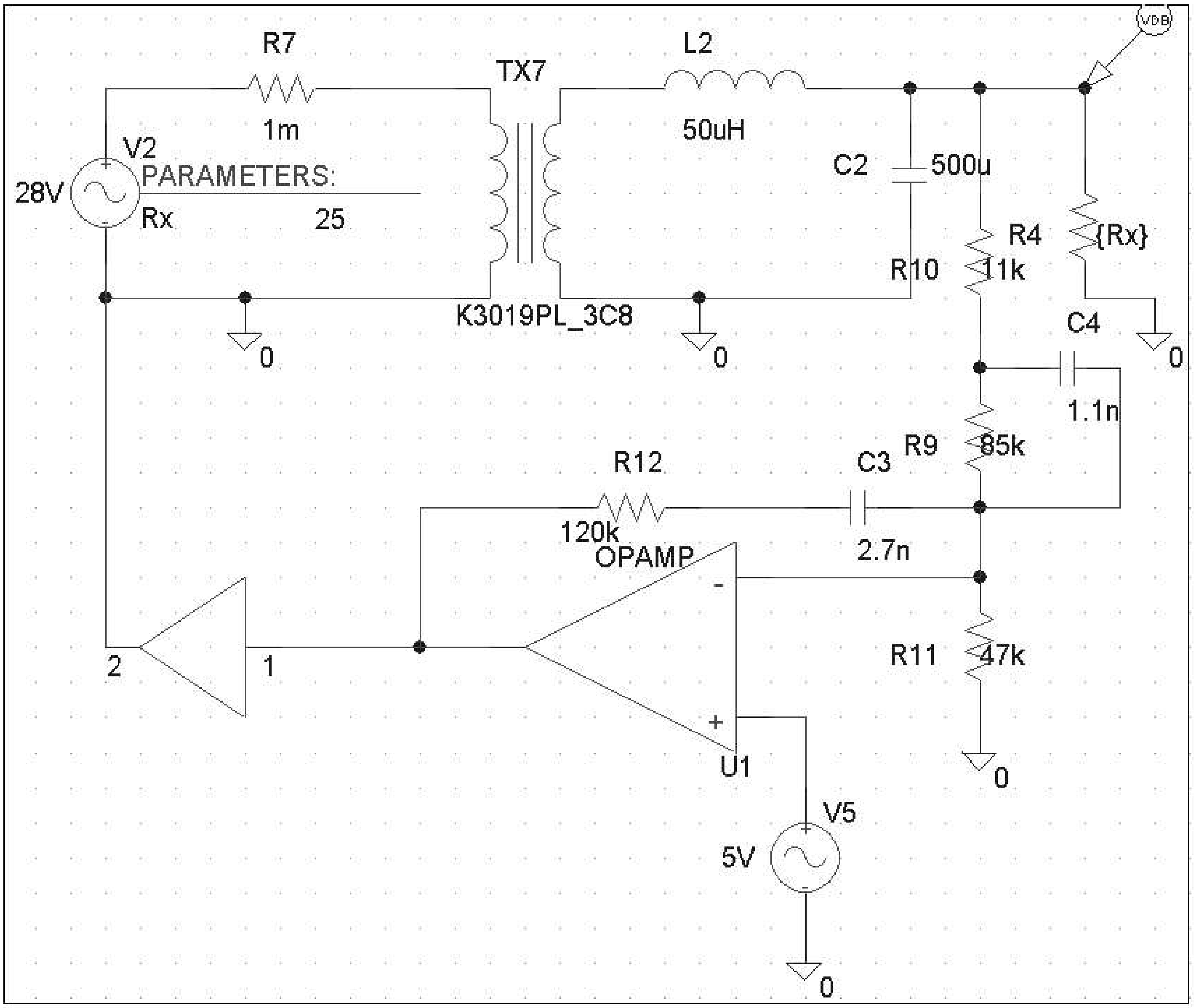


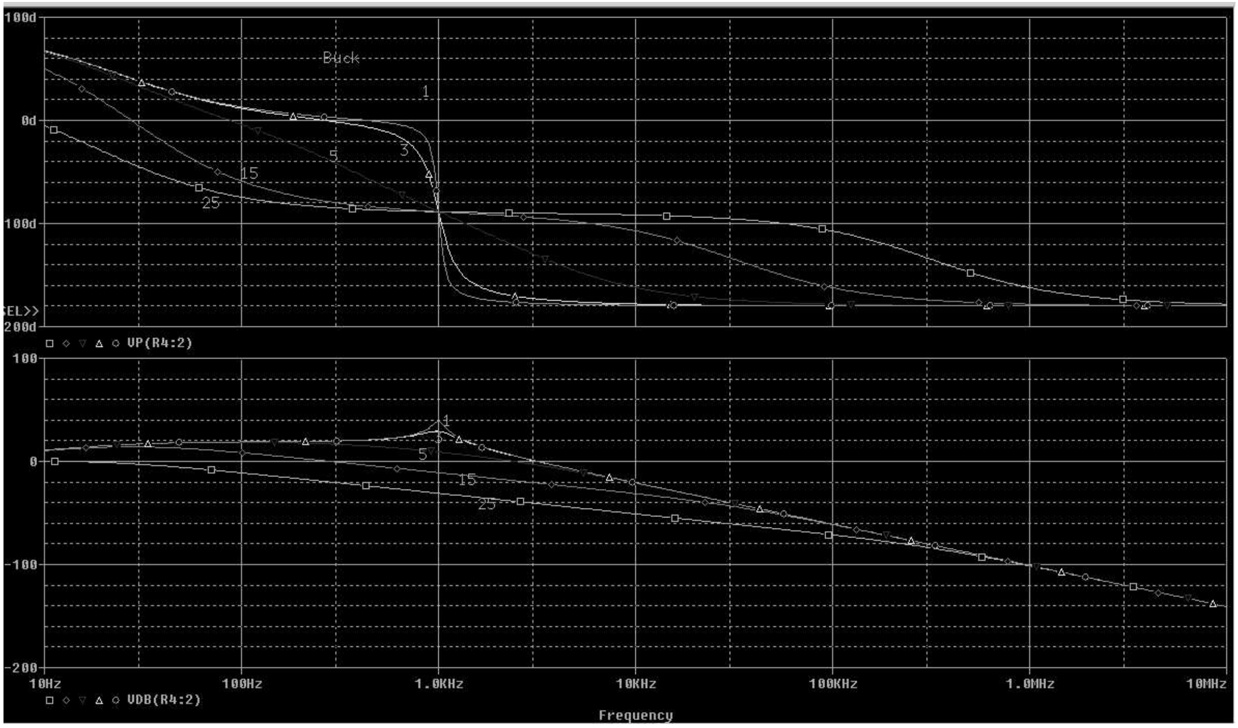
transfer function of buck
Tranfer function dB
20·log(|V/D · 1/(1+j·ω·L/3+(j·ω)^2·L·C)|)
20·log(|V/D · 1/(1+j·ω·L/5+(j·ω)^2·L·C)|)
20·log(|V/D · 1/(1+j·ω·L/15+(j·ω)^2·L·C)|)
20·log(|V/D · 1/(1+j·ω·L/25+(j·ω)^2·L·C)|)
66.865
− 18.981
80
60
40
20
0
−20
1
10
100
1·10^3
1·10^4
1·10^5
1
1×10^5
ω
radian frequency



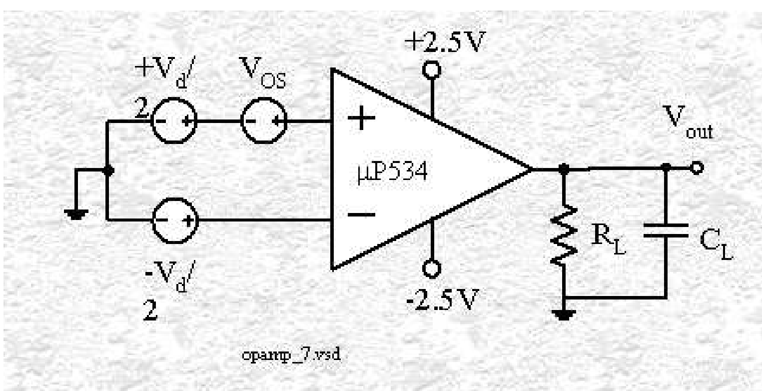
phase for buck
phase (R=3,5,15,25)
arg(V/D · 1/(1+j·ω·L/3+(j·ω)^2·L·C))
arg(V/D · 1/(1+j·ω·L/5+(j·ω)^2·L·C))
arg(V/D · 1/(1+j·ω·L/15+(j·ω)^2·L·C))
arg(V/D · 1/(1+j·ω·L/25+(j·ω)^2·L·C))
− 2×10^−6
− 3.141
0
−0.5
−1
−1.5
−2
−2.5
−3
−3.5
1
10
100
1·10^3
1·10^4
1·10^5
1
1×10^5
ω
radian frequency

I also use the buck voltage regulator example in the Appendix B of the Fundamental of Power Electronics 2nd edition textbook to build the circuit in Pspice software. However, I test all the different values for load resistors. Here is the circuit and it output.





At this moment, you know how to apply the student version of Pspice to build and run simulation of your circuit. And you have some idea what is Pspice professional is. Let use a professional Pspice to design a hearing aid or heart pacemaker. The figure below is a model of your hearing aid circuit.



Since you are a professional design and your job or project is design a hearing aid. As a design engineer have to design a circuit that work and especially meet your customer specifications. If your circuit is not work correctly and not complete the requirements of your customer then your customer will not satisfy with work design. Here are your customer specifications.

–An amplifier is to be realized that meets the following specifications:

1. A differential to single-ended voltage gain Ad > 30,000 (open loop, low-frequency, small-signal) driving an off-chip load resistor RL of 5 kohms.

2. A output voltage swing: >3.5 V peak-to-peak driving RL = 5 kohms.

3. A unity gain bandwidth of > 16 MHz, with RL.

4. The input offset voltage Vos must satisfy VosAd < 100mV for your value of Ad, without RL.

5. The phase margin > 0 (i.e., ac stable) for the unity gain closed-loop configuration, with RL.

6. DC power dissipation and chip area should be minimized.

7. Your circuit should not depend on a precise value of supply voltage or temperature, but should work properly with small variations in either (+/- 5%).

8. Veff should be no smaller than 150 mV, use Veff = 200 mV or larger if power consumption is not a problem.

10. The minimum feature size (lambda) is 0.8 microns.

11. Calculate the **total area of your design** by adding the area of each transistor gate (LxW) to the area of the resistors. The width and length of the transistors and resistors must be multiples of 0.8 microns.

12. The minimum channel length is 1.6 microns and the minimum width is 2.4 microns. Assume the minimum width of a resistor is 3.2 microns, with a sheet resistance of 80 ohms/square.

13. No capacitors are available. However, a MOSFET can be used as a capacitor.

14. All components are to be on chip, except the load R_L and C_L. No voltage or current sources can appear on-chip.

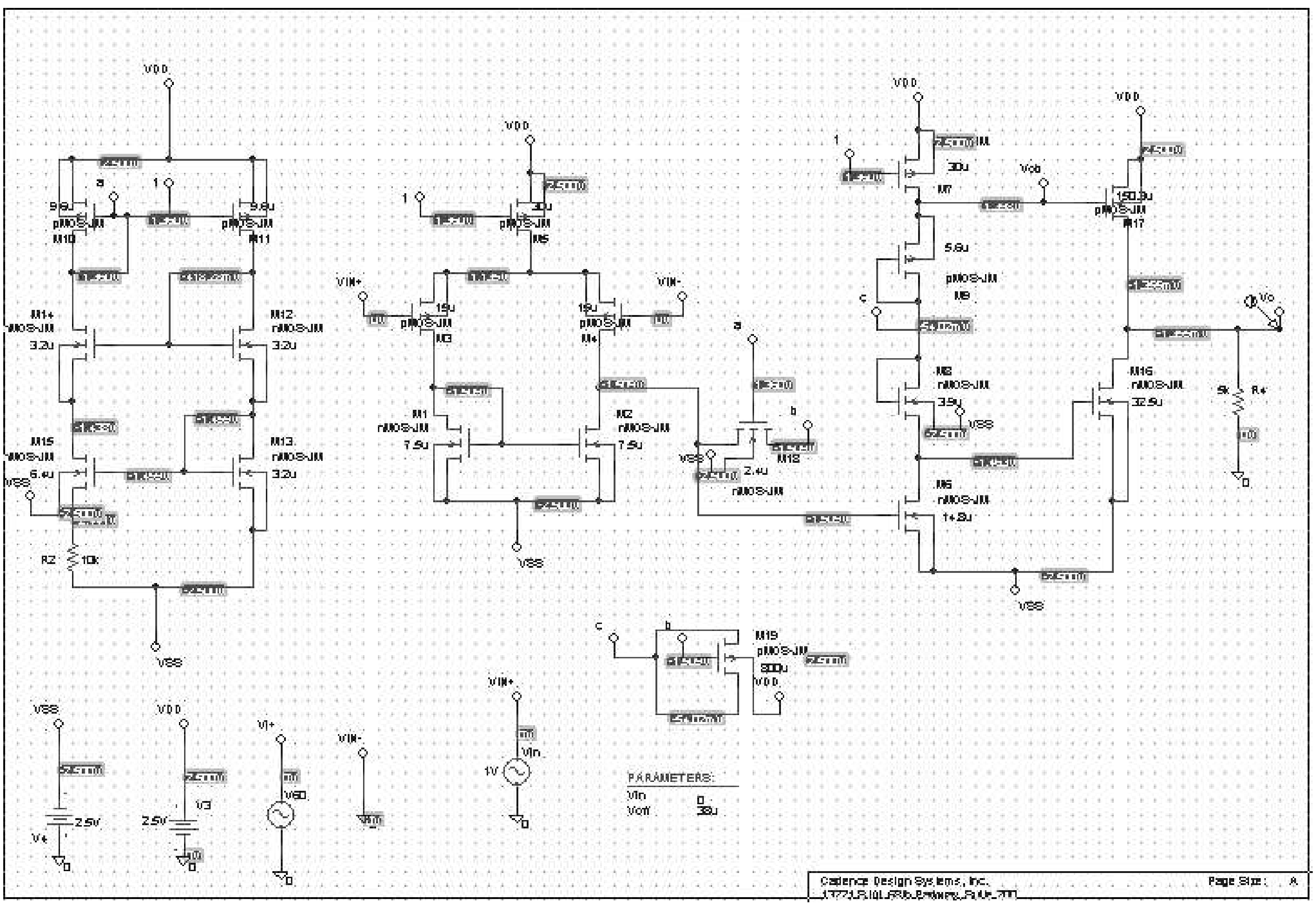
15. Since this is a n-well process, you can tie the well (i.e, body) of a pMOSFET to the source if you want. However, the area of that device is doubled (i.e., 2xLxW).

16. For purposes of calculating MOSFET capacitance, the area and perimeter of the active regions of the transistors should be taken as follows (not counted as part of total area of your design):

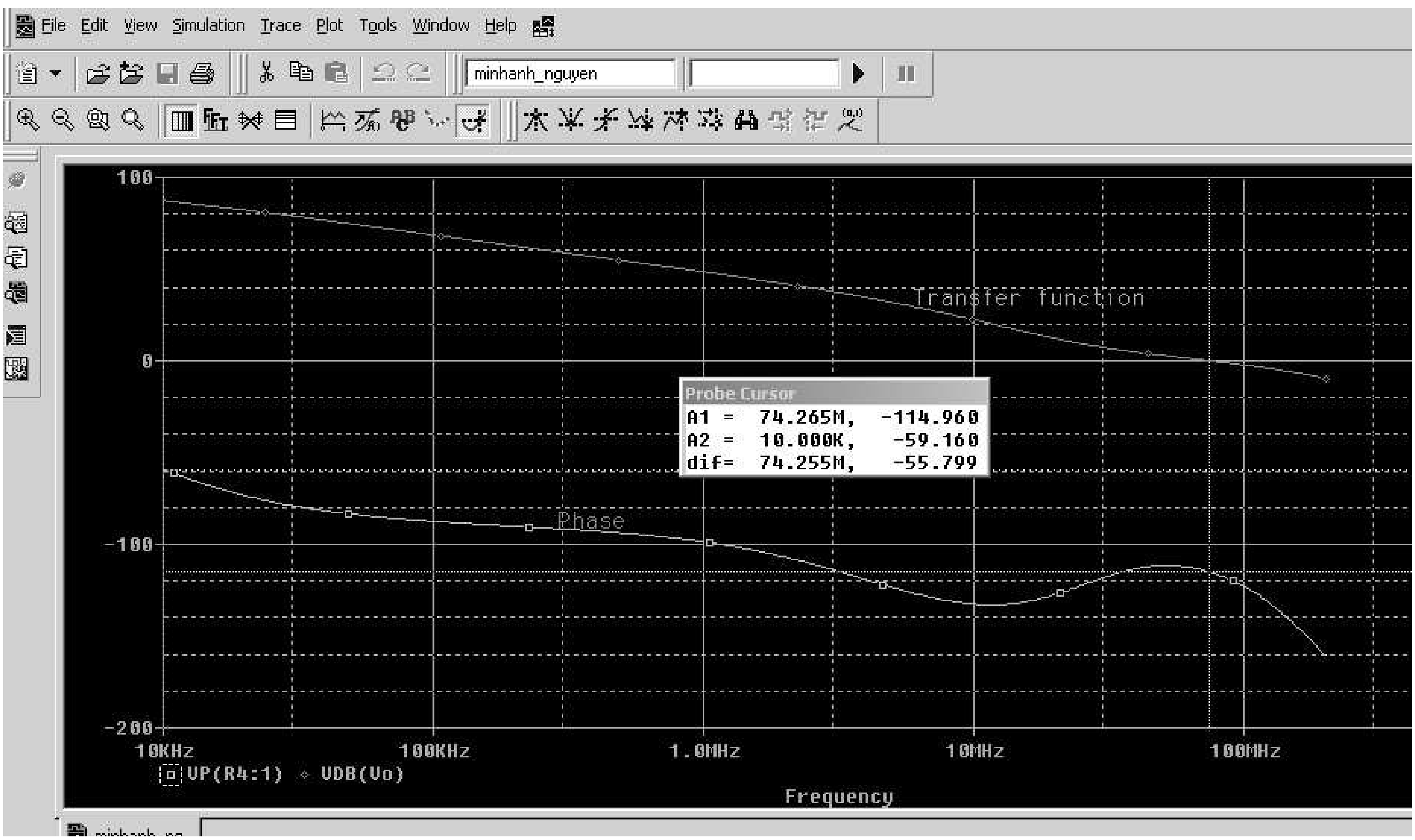
$A_S = A_D = W\text{x}(4.0 \text{ microns})$

$P_S = P_D = W+(8.0 \text{ microns})$

To design a hearing aid by applies only an amplifier, which meets all the specification above. You must use two stages amplifier that you learned in Electronic class (EE332) or analog design circuit class (EE534). Since the gain is bigger than 30000V/V, the differential for the first stage must bigger or equal to 100V/V and the second stage 300V/V. The first stage is a diff amp stage and the second stage is a buffer stage. The buffer stage is applied a push and pull circuit that explain very clearly in topic10 of analog design circuit class note or in any the micro-Electronic circuit textbook. The unity gain bandwidth is greater than 16MHz. To able to make the circuit stable and frequency is bigger than 16Mhz. Use the compensation capacitor or feedback capacitor C in the second stage. The compensation capacitor calculated and apply value should be less than 3pF. The circuit is a stable circuit, when the phase is less than or equal to negative180 degree. The figure below is the complete of two stages ideal model design an operational amplifier hearing aid.



When you have a circuit, which have more than 10 transistors. You must use the Pspice_pro to build your circuit. As you remember at Colorado State University only had 15 licenses, and you can only open the professional window less than 15 minute. You can start build your small circuit on the student version and run the simulation to check make sure your circuit worked. After you completed all the small circuit and it simulation worked, put all your small circuits into a Pspice_pro schematic page, save and run the simulation of the whole circuit. The steps of save and run the simulations are the same as the student version of Pspice. The design above is meets all the specification of the customer required. The figure below is the result of the hearing aid design.



The figure below is the result of Pspice for transistor 10,11,15,and 14 of the circuit

NAME	M_M11	M_M15	M_M14	M_M10
•MODEL	pMOS-JM	nMOS-JM	nMOS-JM	pMOS-JM
•ID	-6.26E-06	5.87E-06	5.87E-06	-5.87E-06
•VGS	-1.14E+00	9.82E-01	1.02E+00	-1.14E+00
•VDS	-2.92E+00	1.00E+00	2.80E+00	-1.14E+00
•VBS	0.00E+00	-5.87E-02	0.00E+00	0.00E+00
•VTH	-9.00E-01	8.17E-01	8.00E-01	-9.00E-01
•VDSAT	-2.40E-01	1.65E-01	2.19E-01	-2.40E-01
•Lin0/Sat1	-1.00E+00	-1.00E+00	-1.00E+00	-1.00E+00
•if	-1.00E+00	-1.00E+00	-1.00E+00	-1.00E+00
•ir	-1.00E+00	-1.00E+00	-1.00E+00	-1.00E+00
•TAU	-1.00E+00	-1.00E+00	-1.00E+00	-1.00E+00
•GM	5.22E-05	7.11E-05	5.35E-05	4.88E-05
•GDS	2.24E-07	4.34E-07	3.83E-07	2.24E-07
•GMB	2.49E-05	2.04E-05	1.60E-05	2.34E-05
•CBD	1.12E-14	6.61E-15	3.04E-15	1.49E-14
•CBS	2.17E-14	9.03E-15	5.44E-15	2.17E-14
•CGSOV	1.92E-15	1.28E-15	6.40E-16	1.92E-15
•CGDOV	1.92E-15	1.28E-15	6.40E-16	1.92E-15
•CGBOV	0.00E+00	0.00E+00	0.00E+00	0.00E+00
•CGS	1.80E-14	1.20E-14	5.99E-15	1.80E-14
•CGD	0.00E+00	0.00E+00	0.00E+00	0.00E+00
•CGB	0.00E+00	0.00E+00	0.00E+00	0.00E+00

Below is The Pspice result for the gain of the hearing aid and the answer is not in DB.

**** SMALL-SIGNAL CHARACTERISTICS

V(Vo)/V_Vin = 4.473E+04

INPUT RESISTANCE AT V_Vin = 1.000E+20

OUTPUT RESISTANCE AT V(Vo) = 4.830E+03

TOTAL POWER DISSIPATION 5.86E-04 WATTS

The calculation of hearing aid use MathCAD software

$$I11 := It\cdot\left(\frac{w11}{w5}\right) \qquad I11 = 8.32\times 10^{-6}$$

$$veff11 := \sqrt{\frac{(2\cdot I11)}{upCox\cdot\frac{w1}{L}}} \qquad veff11 = 0.344$$

$$VDS11 := VDD - veff11 + Vtp0$$

$$VDS11 = 1.256$$

$VGS11 := VDS11$

$VGS11 = 1.256$

$gm11 := \frac{2 \cdot I11}{veff11}$ $\quad gm11 = 4.837 \times 10^{-5}$

$Av1 := -gm3 \cdot \left(\frac{ro2 \cdot ro4}{ro2 + ro4}\right)$ $\quad Av1 = -52.957$

$Av2 := -gm6 \cdot \left(\frac{ro6 \cdot ro7}{ro6 + ro7}\right)$ $\quad Av2 = -112.976$

$Av3 := (gm17 + gm17) \cdot RL$ $\quad Av3 = 7.3$

$AV := Av1 \cdot Av2 \cdot Av3$

$AV = 4.368 \times 10^{4}$

$f3db := \frac{1}{2\pi \cdot Rout \cdot Cc}$ $\quad f3db = 3.976 \times 10^{5}$

$PM := 90 - \frac{1}{\tan\left(\frac{Wta}{weq}\right)}$ $\quad PM = 86.084$

$fta := 16 \cdot 10^{6}$ Hz

$Wta := 2 \cdot \pi \cdot fta$ $\quad Wta = 1.005 \times 10^{8}$

$weq := 4 \cdot \beta \cdot Wta$ $\quad weq = 4.021 \times 10^{8}$

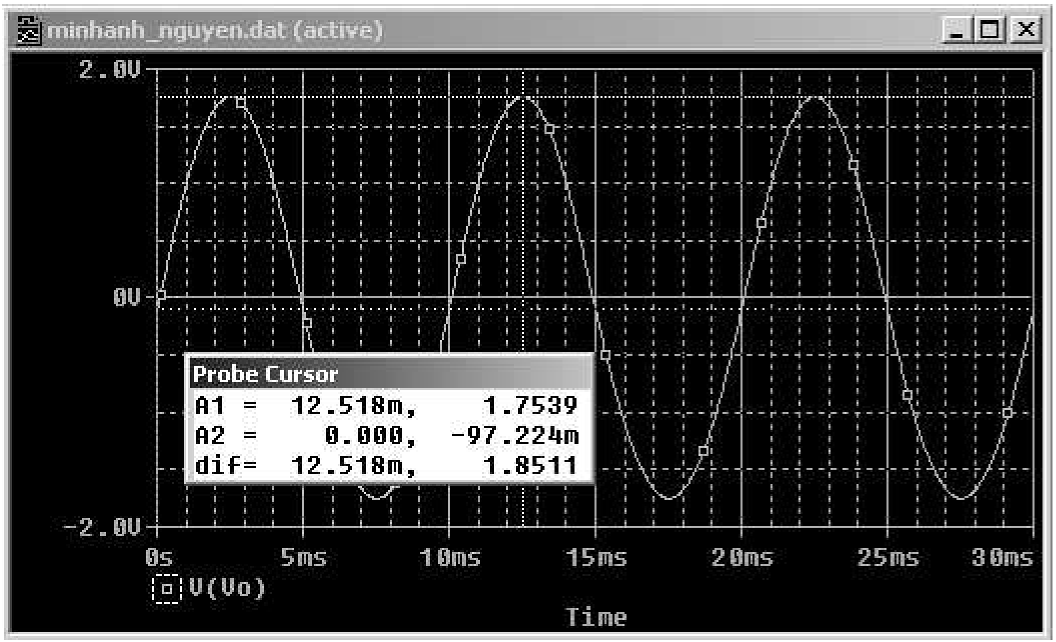
$Rout := \frac{ro16 \cdot ro17}{ro16 + ro17}$

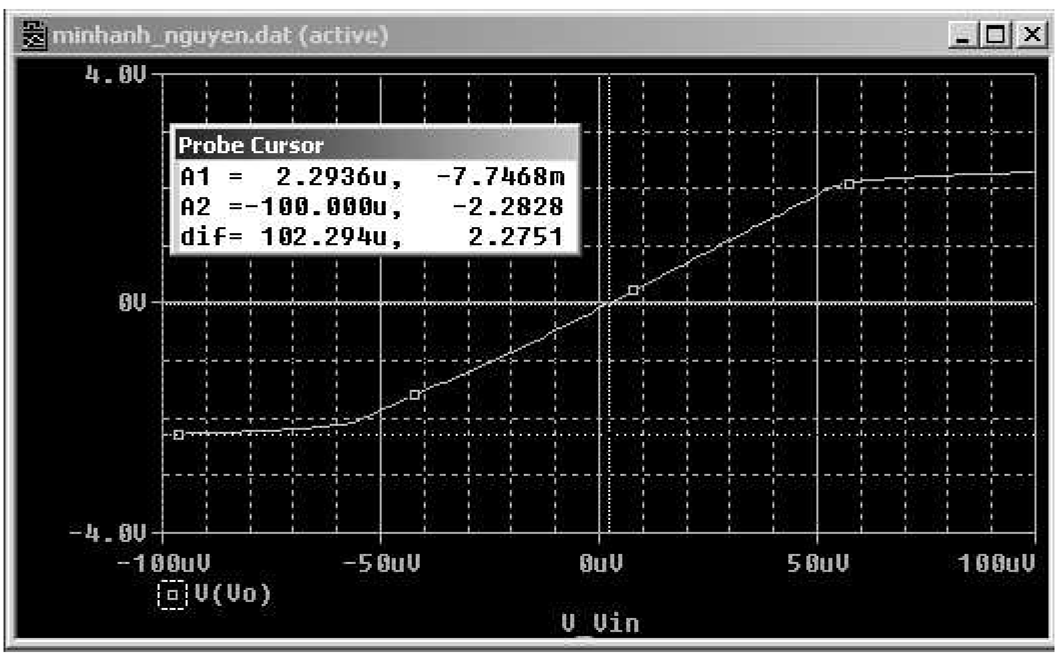
$Rout = 1.142 \times 10^{5}$

$Power := VDD \cdot I - VSS \cdot I$

$Power = 6.25 \times 10^{-4}$

The Output swing voltage and DC Sweep voltage for Hearing aid circuit. The DC sweep uses to determine the offset voltage of the circuit.





This design is more meets all require of the customer. The power is very small and area of the chip is equal to 3.55E-10 m^2. Therefore, this circuit is satisfying the customer.

After spend hours to design a beautiful circuit, sometime the circuit did not run. You don’t know what to check or do with your circuit. And sometime you went mad. Here are some tips that you can do:

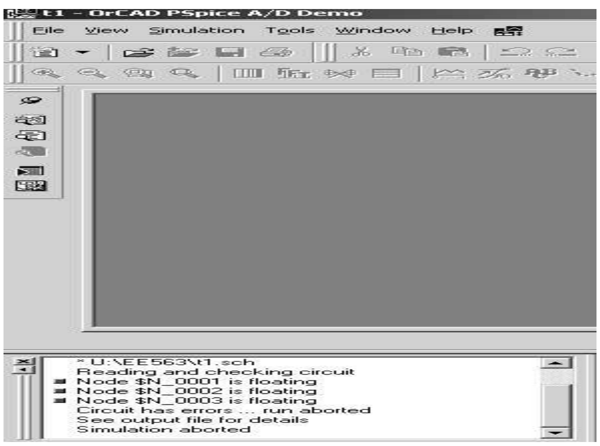
Making Sure You Have a GND: This is very important. You cannot do any simulation on the circuit if you don't have a ground. If you aren't sure where to put it, place it near the negative side of your voltage source.

Make sure Circuit wire correctly**.**

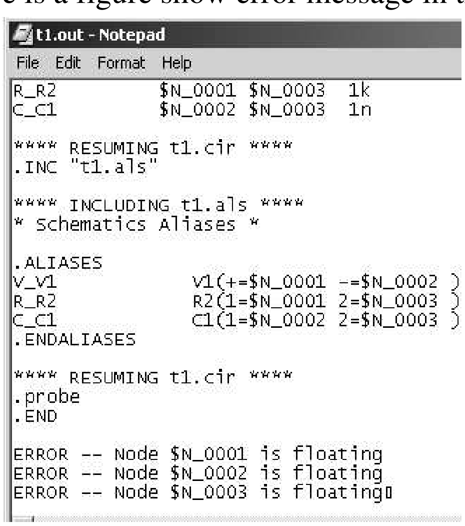
Make all the files that require are contain in your u. Check your setup of Pspice files.

Check the error message in the examine output file

For example, you don’t have GND on your circuit. When you run the circuit, the errors will appearance on your ORCAD Pspice A/D demo window. It is very easy to recognize, because the red button is on and you have no result of graph or value. The figure below is shown the error message on the ORCAD Pspice A/D demo window.



When this happened open your examine output file and read the error message and correct your error. The error message are play at the end of the examine output file. Here is a figure show error message in the examine output file.



t1.out - Notepad

File Edit Format Help

```
R_R2         $N_0001 $N_0003  1k
C_C1         $N_0002 $N_0003  1n

**** RESUMING t1.cir ****
.INC "t1.als"

**** INCLUDING t1.als ****
* Schematics Aliases *

.ALIASES
V_V1            V1(+=$N_0001 -=$N_0002 )
R_R2            R2(1=$N_0001 2=$N_0003 )
C_C1            C1(1=$N_0002 2=$N_0003 )
.ENDALIASES

**** RESUMING t1.cir ****
.probe
.END

ERROR -- Node $N_0001 is floating
ERROR -- Node $N_0002 is floating
ERROR -- Node $N_0003 is floating
```

The uses of a simulation software is an essential requirement to teaching electrical engineering: Simulation technique gives results, when lacking sophisticated hardware for tasks such as: current measurement in a transistor, calculation for Fourier transform of a signal, measurement of magnetic fluxes. Students being what they are (and not what they should), many phenomena are too complex for a theoretical presentation. Simulation allows the presentation of findings when time is short for explanation. Simulation is increasingly used in the industry; familiarizing student with its uses is therefore mandatory.

Since PSpice's strong point is that it helps the user simulate the circuit design graphically on the computer before building a physical circuit. The designer can make any necessary changes on the prototype without modifying any hardware. As soon as the test design is completed. PSpice can help you run a check on it before deciding to commit yourself to building a hard model. PSpice allows you to check the operability of the circuit model in real life simulations to validate its viability. Since all the tests, designs and modifications are made over a terminal, the designer can save a lot of money that would have otherwise been spent on the building of models and modifying them.

Today, I have explained to you what the Pspice software is, and show you step by steps how to create and run a circuit using Pspice software. Also show you step by steps how to calculate the transfer and reasons why the Pspice used in circuit design. I hope that my paper will help you understand more what the Pspice software is. And when someone ask you what is the Pspice software, you can answer him/her question easily. Especially, when you interview for a job.

Reference:

http://www.engr.colostate.edu/EE534/spice/spice_list.htm

http://www.engr.colostate.edu/academic/ece/PSpice/

http://denethor.wlu.ca/pc300/PSpice/pspice_tutorial.html#IIIE

http://rock.uta.edu/dillon/pspice/

http://www.glue.umd.edu/~oramahi/PSPICE-TUTORIAL.pdf

http://www.te.rl.ac.uk/europractice/vendors/cadence_pspice.pdf

http://www.yk.psu.edu/~dec147/eet101/pspqrc.pdf

http://www.stanford.edu/class/ee122/Spice_Decks/pspicedemo.pdf

http://www2.elen.utah.edu/~ee3110/Intro_to_Spice.pdf