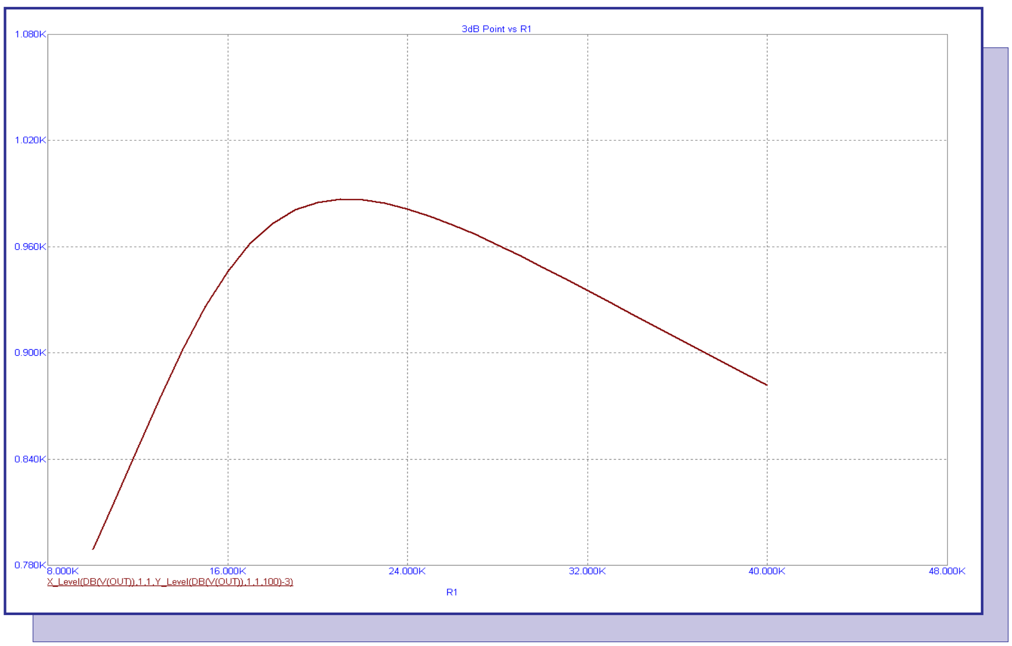
spectrum news

Applications for Micro-Cap™ Users

Winter 2006

News



3dB Point Performance Plot

Featuring:

- RMS Macro
- 3dB Point Performance Plots
- Importing Inductance Values from a Text File

News In Preview

This newsletter's Q and A section describes how to resolve the error that Micro-Cap can't find the file or directory for a TNO file, and answers a question about the use of the RMS operator with a power dissipation expression. The Easily Overlooked Features section describes the use of the Wire Properties dialog box to change the color and width of a wire and to name the node that the wire is associated with.

The first article describes how to create a macro that calculates the root mean square value of an input voltage.

The second article describes how to plot the 3dB point of an AC filter response in a performance plot using a combination of two available performance operators.

The third article describes how to model a time varying inductor whose inductance values are defined within a text file.

Contents

Book Recommendations

General SPICE

- *Computer-Aided Circuit Analysis Using SPICE*, Walter Banzhaf, Prentice Hall 1989. ISBN# 0-13-162579-9

- *Macromodeling with SPICE*, Connelly and Choi, Prentice Hall 1992. ISBN# 0-13-544941-3

- *Inside SPICE-Overcoming the Obstacles of Circuit Simulation*, Ron Kielkowski, McGraw-Hill, First Edition, 1993. ISBN# 0-07-911525-X

- *The SPICE Book,* Andrei Vladimirescu, John Wiley & Sons, Inc., First Edition, 1994. ISBN# 0-471-60926-9

MOSFET Modeling

- *MOSFET Models for SPICE Simulation, William Liu, Including BSIM3v3 and BSIM4*, Wiley-Interscience, First Edition, ISBN# 0-471-39697-4

VLSI Design

- *Introduction to VLSI Circuits and Systems,* John P. Uyemura, John Wiley & Sons Inc, First Edition, 2002 ISBN# 0-471-12704-3

Micro-Cap - Czech

- *Resime Elektronicke Obvody,* Dalibor Biolek, BEN, First Edition, 2004. ISBN# 80-7300-125-X

Micro-Cap - German

- *Schaltungen erfolgreich simulieren mit Micro-Cap V,* Walter Gunther, Franzis', First Edition, 1997. ISBN# 3-7723-4662-6

Micro-Cap - Finnish

- *Elektroniikkasimulaattori,* Timo Haiko, Werner Soderstrom Osakeyhtio, 2002. ISBN# ISBN 951-0-25672-2

Design

- *Microelectronic Circuits High Performance Audio Power Amplifiers,* Ben Duncan, Newnes, First Edition, 1996. ISBN# 0-7506-2629-1

- *Microelectronic Circuits.,* Adel Sedra, Kenneth Smith, Fourth Edition, Oxford, 1998

High Power Electronics

- *Power Electronics,* Mohan, Undeland, Robbins, Second Edition, 1995. ISBN# 0-471-58408-8

- *Modern Power Electronics,* Trzynadlowski, 1998. ISBN# 0-471-15303-6

Switched-Mode Power Supply Simulation

- *SMPS Simulation with SPICE 3,* Steven M. Sandler, McGraw Hill, First Edition, 1997. ISBN# 0-07-913227-8

- *Switch-Mode Power Supply SPICE Simulation Cookbook*, Christophe Basso, McGraw-Hill 2001. This book describes many of the SMPS models supplied with Micro-Cap.

Micro-Cap Questions and Answers

Question: I recently purchased a new computer. I copied the entire Micro-Cap directory over from the older system and then reinstalled the security key driver. The program loads fine, but when I attempt to simulate one of my existing circuits, I get the error message:

No such file or directory D:\MC8\DATA\MyCircuit.TNO

Isn't the TNO file the numeric output file? Why does it need to find it at the beginning of the simulation and how do I resolve this error?

Answer: The TNO file is the numeric output file for transient analysis. At the beginning of the transient simulation, Micro-Cap will create the TNO file to begin writing data to. In this case, the file is not the problem, it is the directory path that Micro-Cap is objecting to when trying to create the file. Most likely, when Micro-Cap was copied from one system to the other, either the partition or the directory that it was located in changed between the source and destination systems. Since the entire Micro-Cap directory was copied over, the new system is using the path references set up for the old system. To remedy this, go to the File menu and select the Paths option. Make sure that the paths specified for the Data field actually exist on the new system. Once this path has been corrected, the error message should no longer appear.

Question: I am trying to determine the continuous effective power at my load resistor. I plot the following expression:

RMS(PD(RL))

but this doesn't produce the results I am looking for. When I plot the next expression:

RMS(V(RL))* RMS(I(RL))

I get the expected curve. Why is there a difference between the two?

Answer: The difference occurs due to the point at which the RMS operator takes effect. In the first expression, the RMS is being calculated based on the instantaneous power of the resistor which would be equivalent to:

RMS(V(RL)*I(RL))

In the second expression, both the current and voltage have their RMS value calculated individually, and then the two RMS values are multiplied together. The second expression is the commonly accepted definition for determining power in AC circuits.

Easily Overlooked Features

This section is designed to highlight one or two features per issue that may be overlooked because they are not made visually obvious with a toolbar button.

Wire Properties Dialog Box
The wires within a schematic can have their properties modified through the Wire Properties dialog box. The Wire Properties dialog box can be invoked by double clicking on a wire when in Select mode. The dialog box will appear as follows:

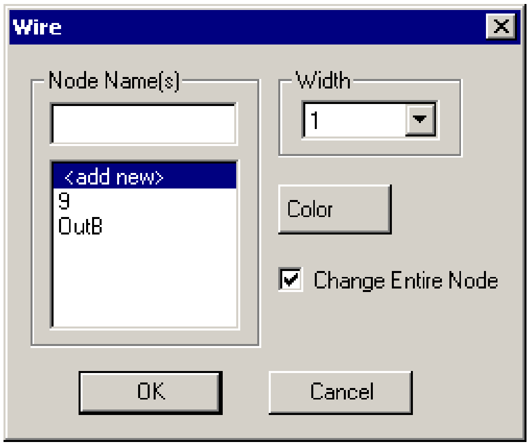


Fig. 1 - Wire Properties dialog box

The Wire Properties dialog box contains the following sections:

Node Name(s): Any wire that is connected to a node will automatically have a node number assigned to it. The node numbers are determined by the order that parts are placed in the schematic so they can change as the circuit evolves. For this reason, with nodes of particular interest, it is recommended that node names be assigned to the nodes. One method is to use the Text mode in the schematic and place a text label on a wire directly. The alternative is to enter a name in this field. The node number will always appear in the list but is not available for editing.

Width: The width of the wire can be set with this field. The width can be set anywhere from one to ten pixels. The default width for the wire is one pixel.

Color: The color of the wire can be selected through this button. The default color of the wires is controlled through the Color/Font page of the Circuit Properties dialog box. If a color is set in this field, it will take priority over the color set in the Circuit Properties dialog box.

Change Entire Node: When enabled, any change in either the Width or the Color settings will be applied to all wires that are connected to the node. If disabled, the Width and Color settings will only be applied to the specific wire segment that was selected.

RMS Macro

The root mean square calculation provides a method to statistically measure the magnitude of a varying waveform. Micro-Cap provides an RMS operator that can be used to plot the root mean square value of any specified waveform. However, the RMS operator can only be used for plotting waveforms in an analysis as it is not available for use in expressions within the schematic. For an instance where a component such as a switch needs to be triggered by an RMS value, circuitry would need to be added to perform the measurement in the schematic. One macro circuit configuration for calculating an RMS value is displayed in Figure 2.

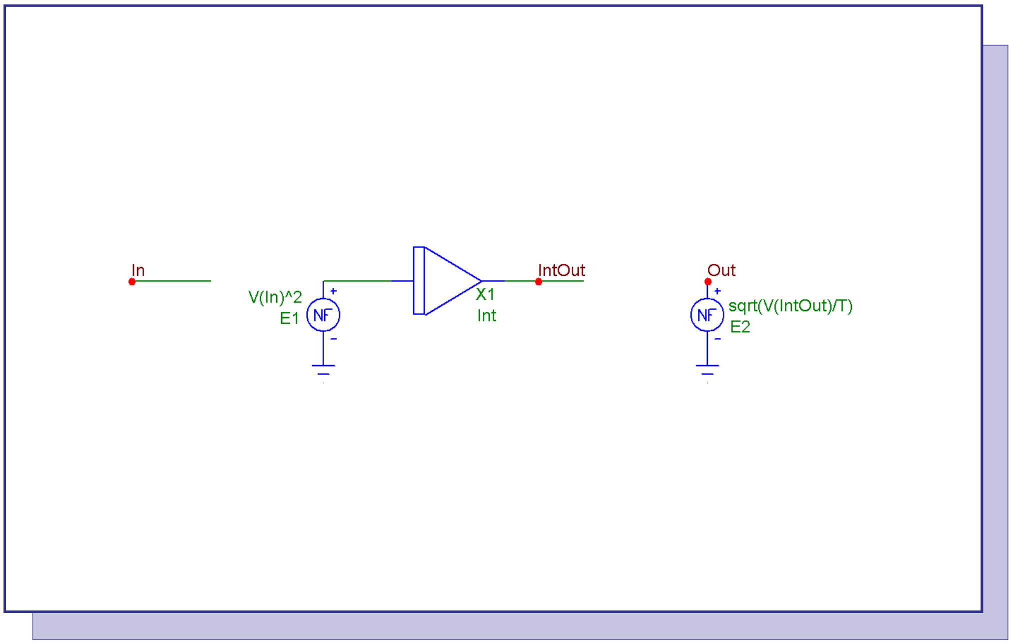


Fig. 2 - RMS Macro Circuit

For a continuous function over the time range T1 to T2, the equation for the root mean square calculation is as follows:

$$y_{\mathrm{rms}} = \sqrt{\frac{1}{T_2 - T_1}\int_{T_1}^{T_2} f(t)^2\,dt}$$

The macro circuit is designed to model this equation. The nonlinear function voltage (NFV) source, E1, squares the voltage at node In. The squared voltage is then fed into an Int macro which performs the integration calculation. For the RMS macro, the T1 parameter from the above equation will be defined as 0 which is the typical value most simulations start with. Since this macro produces a running root mean square value, the T2 parameter will be the time that is currently being analyzed. The output of the integrator is then referenced by the E2 NFV source in the following equation:

sqrt(V(IntOut)/T)

This expression takes the output voltage of the integrator and divides by the current time of the simulation. The T variable can be used in place of T2 - T1 due to the assumption of T1 being 0. Finally, a square root operator is applied to produce the final RMS value. *Due to both the divide by T and the integration within the RMS circuit, any transient simulation using this macro should have its Operating Point option disabled in the Transient Analysis Limits dialog box.*

For circuits, one of the common uses of the RMS calculation is to calculate the power dissipation in individual components when the current and voltage are varying functions. An example circuit using the RMS macro appears in Figure 3 in which the macro is used to calculate the RMS power of the load resistor.

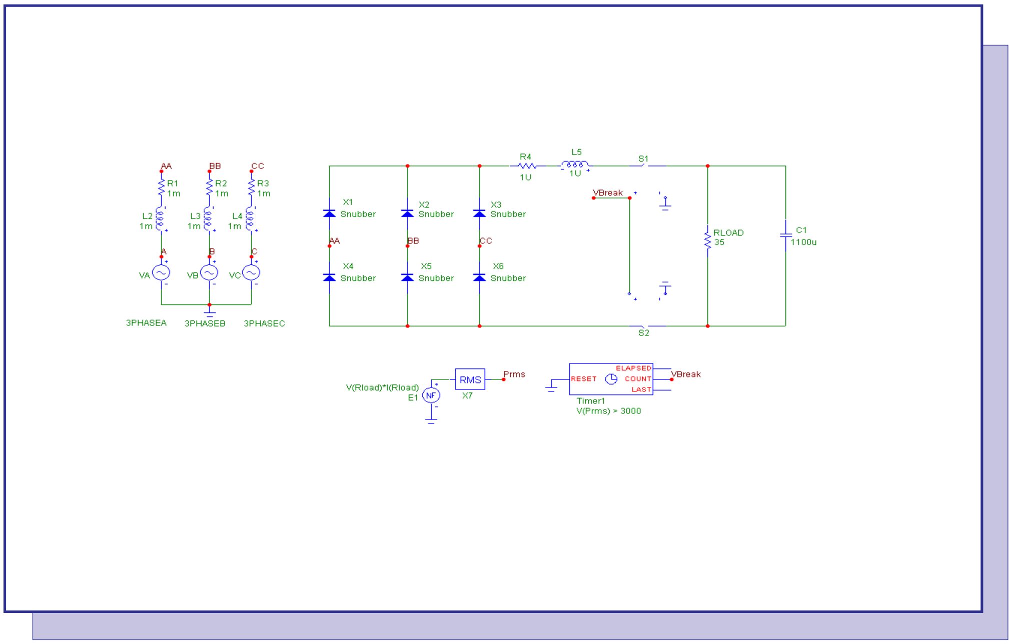


Fig. 3 - RMS Macro Example Circuit

The circuit contains a typical three phase power supply. A six pulse diode rectifier is used as an AC to DC power converter to produce a DC voltage at the load of the circuit. The power through the load resistor will peak during the initial transient of this circuit. To help protect the load, a switch has been connected on each side between the load and the six pulse diode rectifier. These switches will be triggered based on the RMS power dissipated by the load resistor. The E1 NFV source calculates the instantaneous dissipative power of the resistor with the following expression:

V(Rload)*I(Rload)

The value of the instantaneous power is then used as the input into the RMS macro. The RMS power of the load resistor is equivalent to the RMS macro's output voltage at node Prms. Since the switches should stay open once they are triggered, they do not reference the voltage at the output of the RMS macro directly. Instead a Timer component is used to trigger the switches. The Timer component has its INPUTEXPR attribute defined as:

V(Prms) > 3000

which will count each instance that the voltage at node Prms exceeds 3000 volts. The two switches share the same model statement which is defined as follows:

.MODEL LOADSW VSWITCH (RON=1m ROFF=1e6 VON=.2 VOFF=.8)

When the input voltage to the switch is below .2V, the switch is on, and when it is above .8V, the switch is off. The input voltage to the switches is the voltage at the Count pin of the timer. The Count pin keeps track of the number of events that occur defined by the INPUTEXPR attribute. This pin will initially be set to zero volts at the beginning of the simulation so that the switches are on. The first time that the voltage at Prms becomes greater than 3000V, the Count pin voltage is incremented to one volt which will turn off the switches. Since the Reset pin of the Timer is grounded, the voltage at node Count will only increase or stay the same so the switch will remain open for the remainder of the simulation. The transient analysis of this circuit is shown in Figure 4. *Again note that the Operating Point option has been disabled for the RMS macro to operate correctly.*

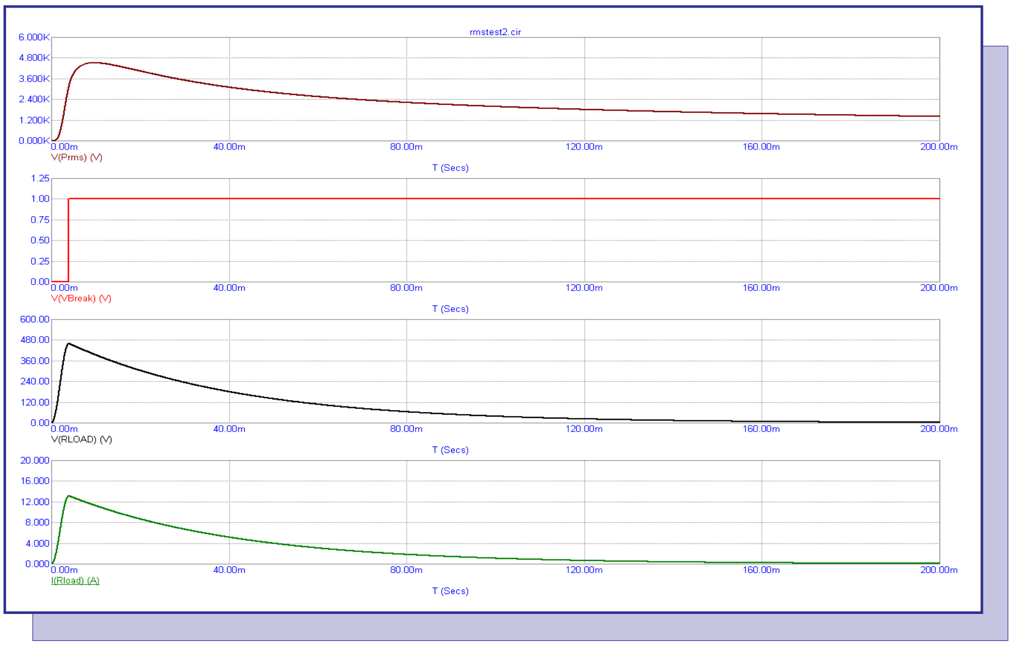


Fig. 4 - RMS Macro Example Analysis

The top plot displays the output voltage of the RMS macro. The second plot displays the voltage at the Count pin of the Timer. Note that when the RMS power exceeds the 3000 limit set in the Timer that the voltage is incremented from zero to one thus turning off the switches. The bottom two plots show the voltage and the current through the load resistor respectively.

3dB Point Performance Plots

The 3dB reference point is a common reference frequency that is used when evaluating the frequency response of a filter. This reference point is defined as the frequency at which the gain drops to .7071 (1/sqrt(2)) of the passband gain. The 3dB point can be used to determine the bandwidth, cutoff frequency, and half power frequency among other measurements depending on the type of filter being simulated.

For a single simulation waveform, the 3dB point can be determined by plotting the dB waveform of the output in AC analysis. When the analysis finishes, enter Cursor mode, and move one of the cursors until it is 3dB below the value specified in the passband gain in order to view the 3dB point frequency. However, when using stepping or Monte Carlo to create multiple waveform branches, determining the 3dB point with this method can be time consuming. The performance plot capability of Micro-Cap provides a quick way to view the 3dB points for all of the branches in a single plot. The performance plots are designed to extract circuit performance measurements from curves generated during an analysis. It will extract one measurement from each branch and then plot that measurement versus the parameter that varied during the simulation. Since there is not a performance operator designed for directly determining the 3dB point, this article describes how to use a combination of two performance operators to produce the appropriate plot.

The schematic in Figure 5 was created through the Active Filter Designer available within Micro-Cap. It is a low pass Butterworth filter that uses the Sallen-Key circuit configuration for both stages. The filter is designed to have a passband gain of 0dB. The opamp models within the filter have their Level parameters set to 1 so that the opamp model used is a simple voltage-controlled current source with a finite output resistance and open loop gain.

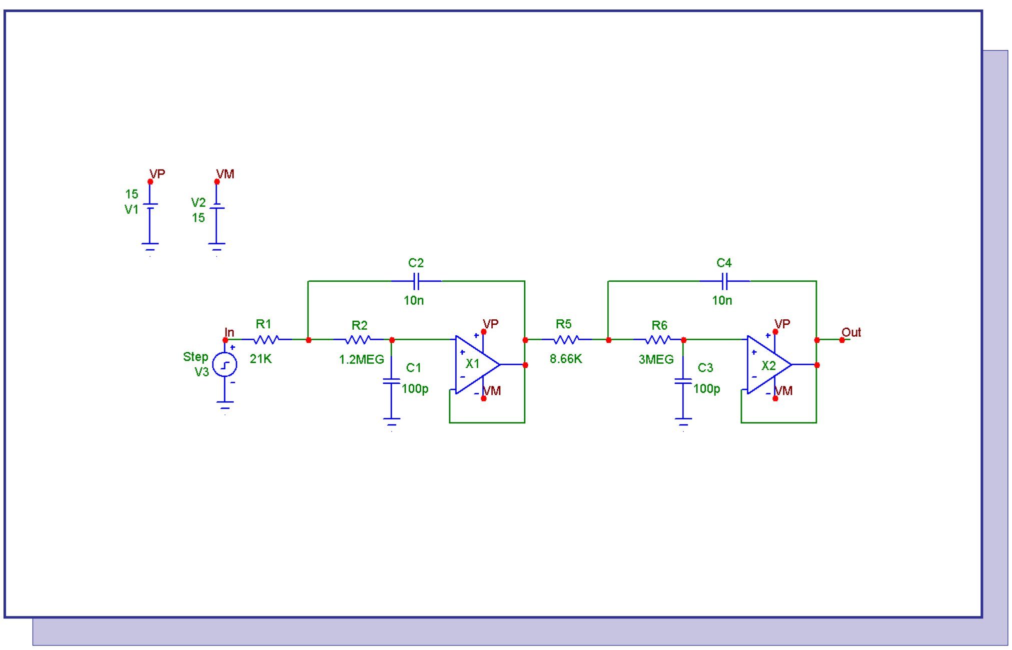


Fig. 5 - Low Pass Butterworth Filter

The AC frequency response for this filter is shown in Figure 6. The waveform being plotted is the decibel magnitude of the voltage at node Out defined as DB(V(OUT)). For this example, the resistor, R1, at the input to the first stage is being stepped from 10kohm to 40kohm in 1000 ohm increments which produces a total of 31 stepped waveforms. Once the simulation finishes, a performance plot can be created to harvest the 3dB data.

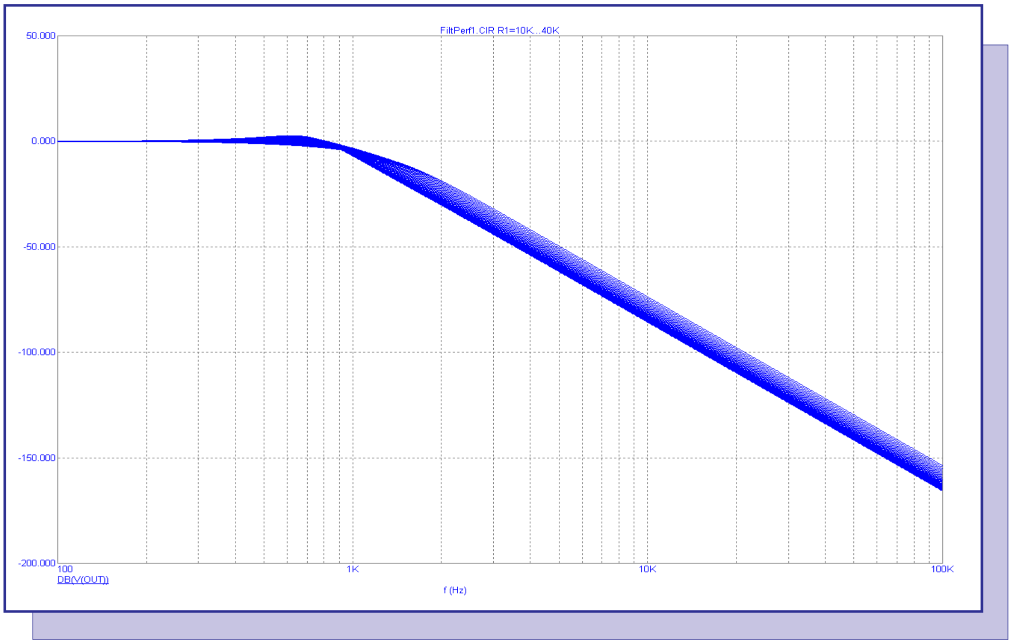


Fig. 6 - Low Pass Butterworth AC Simulation

The performance plot can be created by going to the AC menu, clicking on Performance Windows, and then selecting Add Performance Window. A shortcut to this command can be accessed by right clicking on the DB(V(OUT)) expression in the AC plot, and selecting Add Performance Window on the menu that appears. To extract the 3dB information, the performance plot will use an expression which is a combination of the X_Level and Y_Level performance operators. The X_Level operator has the following syntax:

X_Level(Expression,Boolean,N,Y Level)

The X_Level operator returns an X value at a specified Y value. The Expression field needs to be one of the expressions plotted during the simulation. The Boolean field defines a boolean expression that determines the portion of the specified expression that the performance operator will take its measurement from. The N field is an integer field that defines which instance will be found and measured. The Y Level field specifies the Y value at which the X value will be measured. The Y_Level operator has the following syntax:

Y_Level(Expression,Boolean,N,X Level)

The Y_Level operator acts in the same manner as the X_Level operator except that a Y value is returned at a specified X value. The only field that is different is the X Level field which is used to specify the X value at which the Y value will be measured.

For the AC simulation from the low pass Butterworth filter example circuit, the X_Level and Y_Level operators will be combined as follows in order to measure the 3dB point frequency versus the resistance that was varied.

X_Level(DB(V(OUT)),1,1,Y_Level(DB(V(OUT)),1,1,100)-3)

The X_Level operator is measuring the DB(V(OUT)) expression. The Boolean field has been set to 1 which means that the operator can work over the entire simulation, and the N field has been set to 1 so that the first instance of the measurement will be returned. The Y Level field for the X_Level operator is defined as:

Y_Level(DB(V(OUT)),1,1,100)-3

This Y_Level specification will return the Y value of DB(V(OUT)) at a frequency of 100Hz which provides a measure of the passband gain for this filter response. This passband gain value is then subtracted by three to produce the 3dB value at which the frequency will be measured. The final output of the full expression is the frequency value at which the output gain is 3dB below the passband gain which is the definition of the 3dB point frequency. When using this technique, each step of the AC simulation would need to have a common frequency at which the passband gain can be measured. The result of this performance plot expression is displayed in Figure 7 which shows the 3dB point frequency values being plotted versus the resistance values of the R1 resistor for the Butterworth filter example.

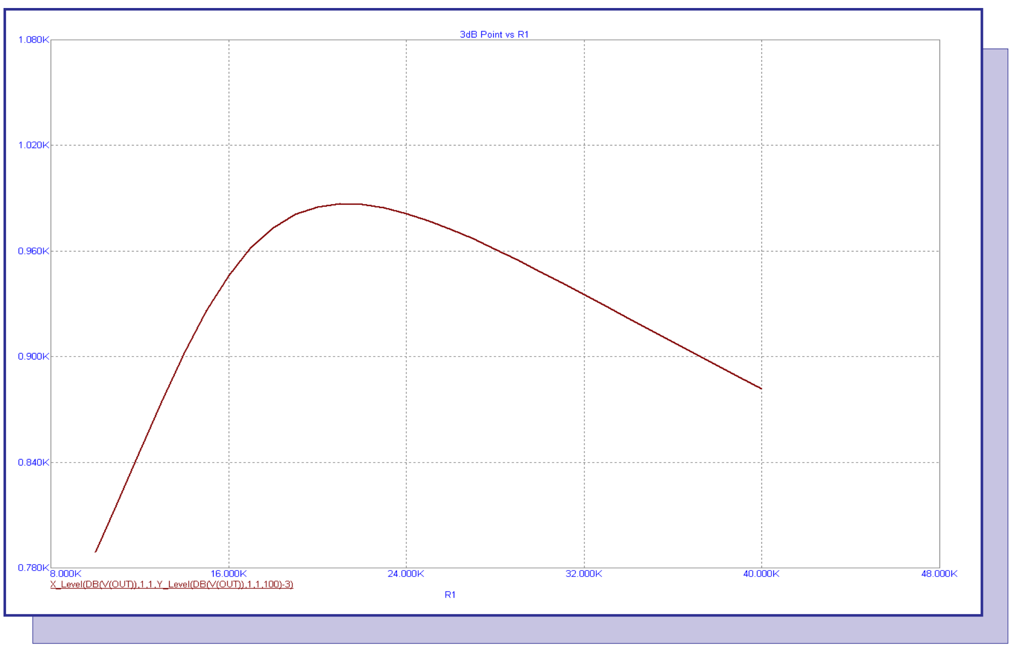


Fig. 7 - 3dB Performance Plot

Importing Inductance Values from a Text File

Time varying inductances can be a critical tool in simulation for such uses as modeling the EMF of a rotor rotation, creating a dynamic tuning control, and simulating many types of magnetic applications. Modeling a time varying inductance in Micro-Cap can be done through either a time based equation or by importing tabular values. Since in many applications, it would be difficult to derive an equation to represent the varying inductance versus time, importing tabular values is the more common method when simulating such an inductor. Often these tabular values have been created in a text file by a third party program. This article describes how to import inductance values from a text file that contains a table of time versus inductance and how to use these imported values to model a time varying inductor.

The file format that the time versus inductance table needs to be in is the USR (User Source) file format. The USR format is essentially a table of comma delimited values with some basic header information at the top of the file. Many third party programs will create the table data within an Excel file. The conversion process from an Excel file to a USR file is described in an article in the Fall 2002 newsletter issue called "Converting an Excel File to a User Source File".

An example schematic that displays the method for modeling such a time varying inductance is shown in Figure 8. There are actually two circuits within the schematic. Both are simple RLC circuits with a pulse source at the input. The top RLC circuit contains a constant, ideal inductor. The bottom RLC circuit contains a time varying inductor model.

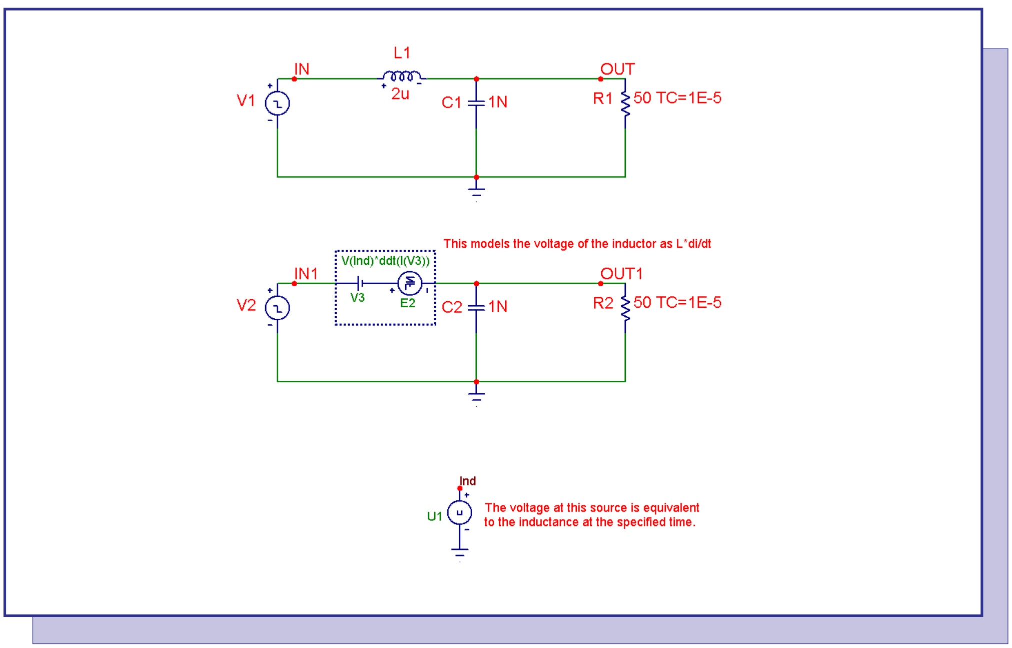


Fig. 8 - Time Varying Inductor Example

The first step in modeling a time varying inductor is to create the associated USR file that contains the inductance values. For this example, the USR file that will be used is called Winter2006.USR, and it contains the following information:

```
[Main]
FileType=USR
Version=2.00
Program=Micro-Cap

[Menu]
WaveformMenu=Ind vs T

[Waveform]
Label=Ind vs T
MainX=T
LabelX=T
LabelY=Ind
Format=SimpleNoX
Data Point Count=5
0,1e-6
1e-6,1e-6
2e-6,2e-6
3e-6,2e-6
4e-6,5e-6
```

The header for the User Source file consists of three sections. The [Main] section defines the format version that is being used. This section should appear exactly as it is shown here for all User Source files.

The [Menu] section defines the names of the waveforms that are present in the file. Since there is only one waveform in this file, there is only one WaveformMenu tag defined. In this case, the waveform is called 'Ind vs T'. Micro-Cap uses these tags to display the names of the waveforms within the file in both the Attribute dialog box and in the Analysis Limits dialog box.

Each waveform in the file must have a [Waveform] section defined for it. This section defines the names of the data columns and declares the format that the data is expressed in. The Label tag must match one of the WaveformMenu tags in the [Menu] section. This tag contains the string that is to be defined in the Expression attribute of the User Source when the source is placed in a schematic. The Format tag defines the format that the data is in. In this case, SimpleNoX states that there will be only two columns of data with the left column being time and the right column voltage (T,Y). Since the format is SimpleNoX, both the MainX and LabelX tags will be defined as T for time, and the LabelY tag is defined as Ind for this waveform. The Data Point Count tag declares the number of data points the waveform consists of. For this example, five data points have been defined for the inductance values. The waveform will be linearly interpolated between data points. The actual data points for the waveform should immediately follow the [Waveform] section. The entire header can just be copy and pasted over to a new file while just modifying the appropriate lines.

The USR file is imported into the schematic through the use of the User Source designated U1. The U1 source has its relevant attributes defined as follows:

FILE = Winter2006.USR
EXPRESSION = Ind vs T
REPEAT = 1

The output of this source is the voltage at node Ind. The voltage at this node will be equivalent to the value of the inductance that is to be modeled. There are a couple of ways to model the actual inductor in the schematic. The method chosen here provides the most robust capability in terms of convergence. The inductor is modeled with a combination of a battery and an NFV (nonlinear function voltage) source. The V3 battery is defined with a value of zero volts. Its only purpose is to measure the current through the branch. The E2 NFV source models the actual inductor by simulating the voltage drop that would occur if the inductor was present. The source has its VALUE attribute defined as:

V(Ind)*ddt(I(V3))

This expression simulates the basic L * di/dt voltage characteristic of an inductor by multiplying the voltage at node Ind by the derivative of the current through the V3 source with respect to time.

A transient analysis simulation of this schematic is displayed in Figure 9. The top plot in the simulation contains three waveforms. V(In) is the voltage at the input pulse source of the ideal RLC circuit. The pulse sources for both RLC circuits share the same specifications so the voltage waveform at node In1 would be the same. V(Out) is the output voltage waveform of the RLC circuit containing the ideal inductor. V(Out1) is the output voltage waveform of the RLC circuit containing the time varying inductor model. The bottom plot in the simulation contains the voltage at node Ind which shows the time varying inductance value the V(Out1) waveform is being affected by.

Note that when the time varying inductance value is specified as 2uH between 2us and 3us, the voltage waveform at node Out1 matches the voltage waveform at node Out which has a constant inductance of 2uH over the course of the simulation. For a second transient analysis simulation, the ideal inductor in the schematic has been defined with the value of 5uH. The resultant analysis is displayed in Figure 10. Now the two output voltages match during the fifth cycle of the simulation when the time varying inductance is specified as 5uH also.

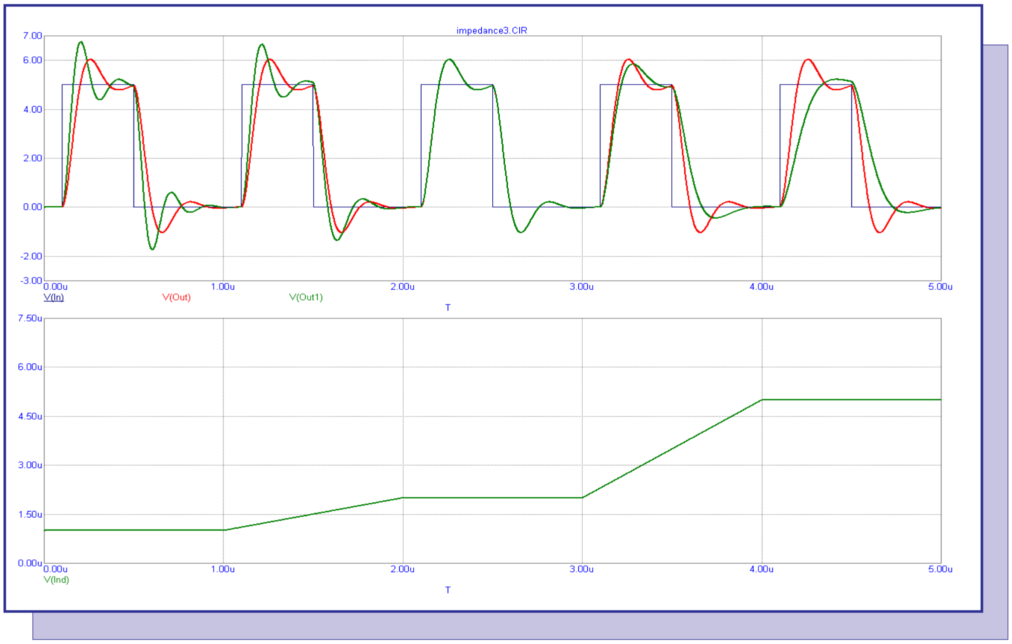


Fig. 9 - Nonlinear Inductance Analysis (Ideal Inductance = 2uH)

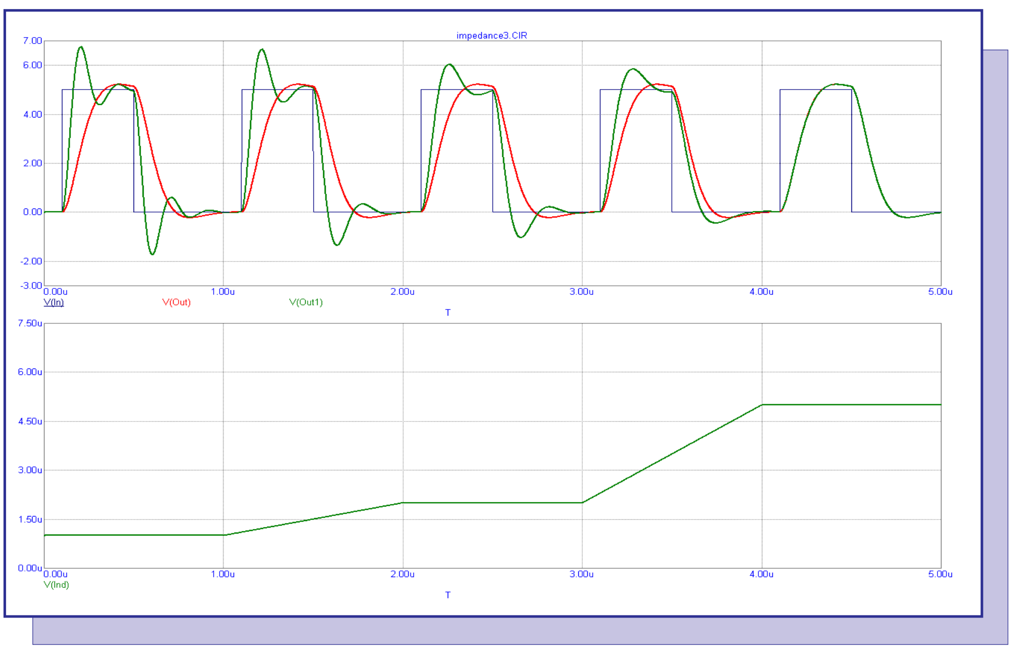


Fig. 10 - Nonlinear Inductance Analysis (Ideal Inductance = 5uH)

Product Sheet

Latest Version numbers

Micro-Cap 8 .... Version 8.1.1
Micro-Cap 7 .... Version 7.2.4
Micro-Cap 6 .... Version 6.3.3
Micro-Cap V .... Version 2.1.2

Spectrum's numbers

Sales .... (408) 738-4387
Technical Support .... (408) 738-4389
FAX .... (408) 738-4702
Email sales .... sales@spectrum-soft.com
Email support .... support@spectrum-soft.com
Web Site .... http://www.spectrum-soft.com
User Group .... micro-cap-subscribe@yahoogroups.com