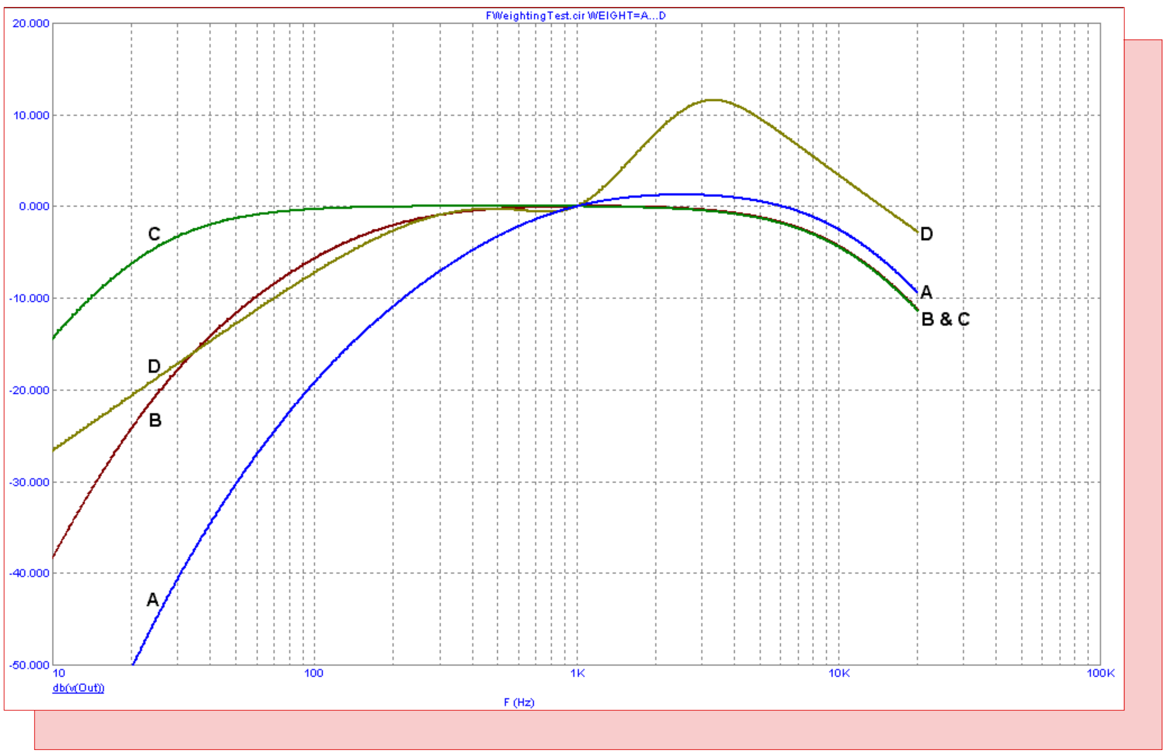
spectrum news

Applications for Micro-Cap™ Users

Fall 2008

News



Frequency Weighting Filter Macro

Featuring:

- Optimizing for Phase Margin
- RTD Macro
- Frequency Weighting Filter Macro

News In Preview

This newsletter's Q and A section describes the merits of installing the license manager as a service on the server rather than an application. The Easily Overlooked Feature section describes how to export waveform data in the CSV file format.

The first article describes how to use the optimizer within Micro-Cap to optimize a stable phase margin for an opamp circuit.

The second article describes how to model a resistance temperature detector which is commonly used for temperature measurements.

The third article describes how to model a frequency weighting filter macro that simulates the A-, B-, C-, and D-weighting specifications of sound sensitivities for AC analysis simulations.

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, 1993. ISBN# 0-07-911525-X

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

MOSFET Modeling

• *MOSFET Models for SPICE Simulation, William Liu, Including BSIM3v3 and BSIM4*, Wiley-Interscience, 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# 951-0-25672-2

Design

• *High Performance Audio Power Amplifiers,* Ben Duncan, Newnes, 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, 1997. ISBN# 0-07-913227-8

• *Switch-Mode Power Supplies Spice Simulations and Practical Designs*, Christophe Basso, McGraw-Hill 2008. This book describes many of the SMPS models supplied with Micro-Cap.

Micro-Cap Questions and Answers

Question: We have a network license for multiple copies of Micro-Cap. The key is plugged into our Windows server, and the license manager has been installed. The clients can run Micro-Cap fine most of the time, but we intermittently run into problems where none of the clients seem to be able to find the license. I believe I have tracked down the trigger to this problem. When I installed the license manager, I did it through my administrative account on the server. Now it seems that whenever my account has been logged out or the server has been rebooted, none of the clients can access the license. I can fix the problem by just logging into my account again. Is there a way to get the license manager accessible to the client systems without having to login into my server account every time?

Answer: The license manager can be installed as either an application or as a service. If it is installed as an application, then its availability is dependent on the user whose account it was installed into being logged into the server. If that user is not logged in, then the application will not be running.

The way to get around this is to install the license manager as a service. For Windows servers, the LMSetup.exe executable file provides the option to install either as an application or a service during the basic install procedure. During the LMSetup installation, one of the screens should state:

Do you want to install HASP License Manager as an application or as a service?

Service is the recommended option for installation. When installed as a service, the license manager software will always be running whenever the server is on. It does not matter if any user is logged in or not.

Easily Overlooked Features

This section is designed to highlight one or two features per issue that may be overlooked among all the capabilities of Micro-Cap.

Exporting Waveform Data in CSV Format

The CSV (Comma Separated Value) file format is a common format for exchanging data between different programs. Microsoft Excel is one of the more popular programs that uses CSV files. Micro-Cap has the capability to export waveform data in CSV format which can then be easily imported into programs such as Excel.

A CSV file can be created through the Save Curves page of the Analysis Properties dialog box. Once the simulation has finished, invoke the Analysis Properties dialog box and click on the Save Curves tab. The page appears as below.

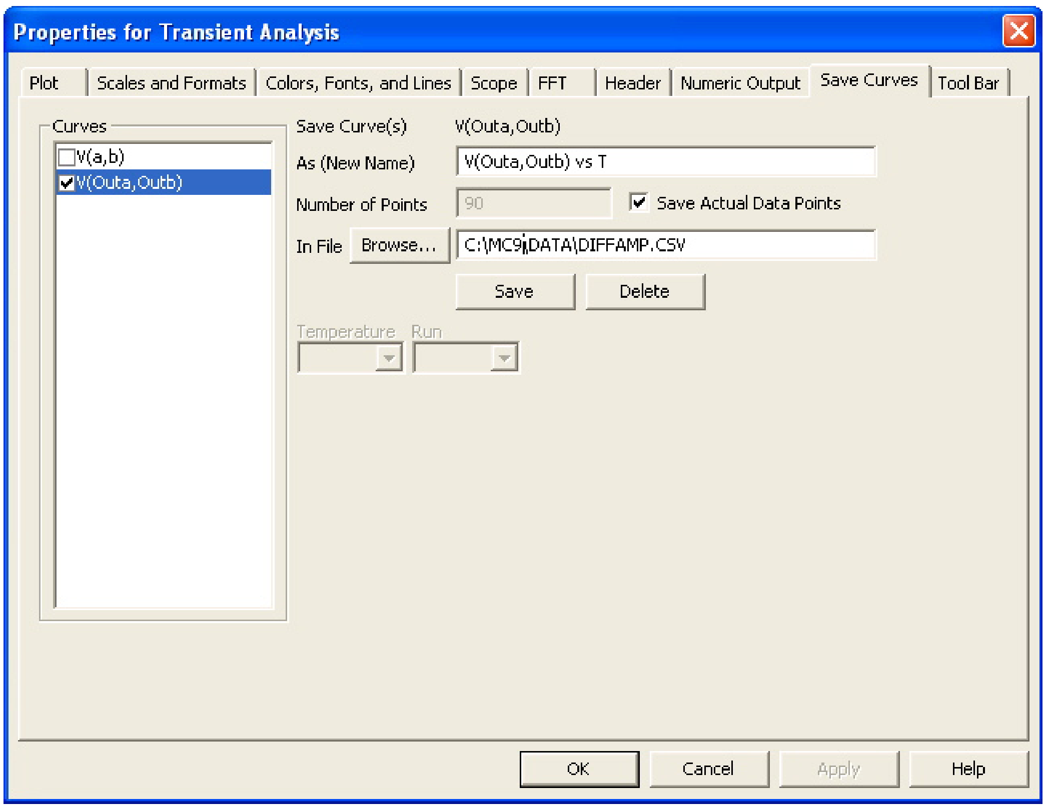


Fig. 1 - Analysis Properties - Save Curves page

The list on the left hand side of the page displays the waveforms that can be exported. A check mark in the box to the left of the waveform enables that waveform for export. The As (New Name) field lets you set the title of the waveform in the header of the CSV file. If Save Actual Data Points is enabled, all of the data points calculated during the simulation will be exported into the file. Otherwise, it will interpolate the waveform using the number of data points specified in the Number of Points field.

The In File field lets you specify the file name that will be created when the Save button is clicked. This page can create either User Source files or CSV files. For CSV files, the extension of the file name must be CSV for that format to be used. Clicking the Save button creates the file.

Optimizing for Phase Margin

The phase margin measurement of an opamp circuit is one method of determining the stability of the circuit. At a phase margin of zero, an opamp circuit becomes unstable. Even at small phase margin values, problems can occur such as peaking in the frequency response or ringing in a step response. Typically a phase margin value of 45 degrees provides a safe margin to produce a stable opamp circuit.

The optimizer that is built into Micro-Cap is a great tool for optimizing circuit measurements such as phase margin. The optimizer provides a simple method to determine the value of specific elements in the circuit to produce a stable phase margin.

One common cause of low phase margin occurs when the opamp is driving a capacitive load. The output resistance of the opamp combines with the load capacitance to create an additional pole in the circuit's transfer function. There are numerous methods to stabilize a circuit that has a capacitive load. The method used in this article will be to add an out of the loop isolation resistor between the output of the opamp and the capacitive load. The Micro-Cap optimizer will be used on the circuit below to calculate the resistance value of the isolation resistor in order to produce a stable phase margin.

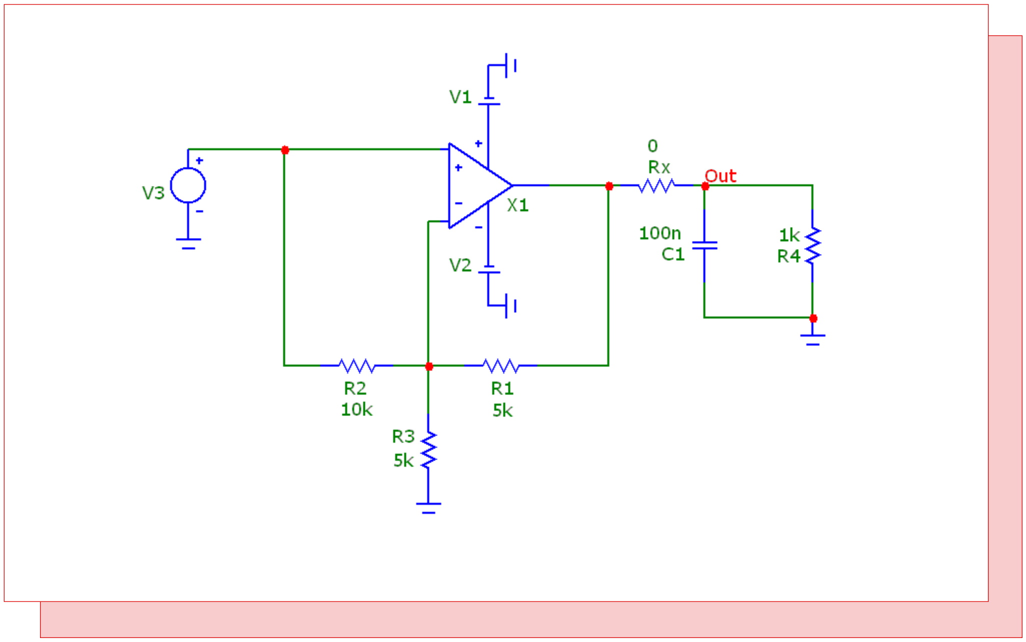


Fig. 2 - Low phase margin opamp circuit

The circuit is a basic noninverting opamp circuit that has a gain of two. The opamp model used was created just for this example. It uses the Level 2 opamp model with an open loop gain of 100k and an output resistance of 100 ohms. The opamp is driving a load that consists of a 1000 ohm resistor in parallel with a 100nF capacitor. Note that the isolation resistor, Rx, is present in the circuit even though its value has been set at zero. The optimizer will only operate on circuit elements that are already in the schematic. Since the isolation resistor is set to zero, it will initially operate as a short circuit prior to the optimizer setting its value. The unoptimized frequency response of this circuit is shown in Figure 3.

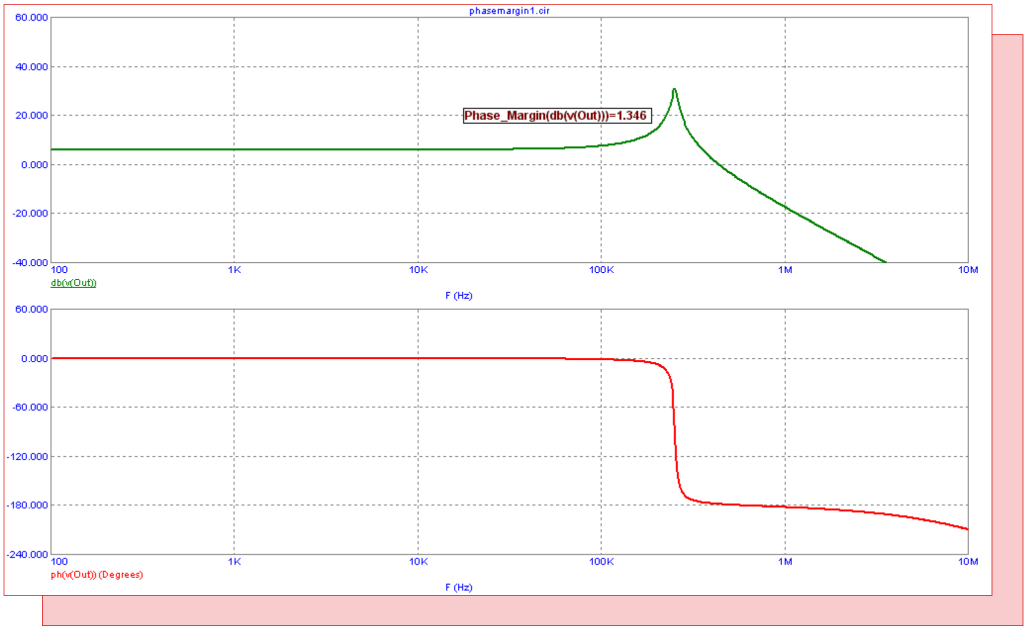


Fig. 3 - AC analysis showing a phase margin of 1.3 degrees

The phase margin is measured by determining the phase at the point that the gain curve crosses zero. This phase value then has 180 degrees added to it to produce the phase margin value. The performance tag in the top plot displays the phase margin of the circuit. To use the Phase_Margin performance operator both the gain in dB and the phase of the desired output must be plotted. Prior to opimization, the phase margin is barely above zero at 1.3 degrees. Peaking can be seen in the gain curve as a result of the low phase margin. To ensure stability with this circuit, the optimizer will be used to increase the phase margin to 45 degrees by calculating a value for the isolation resistor. The settings for the Optimizer are shown in Figure 4.

The Find section specifies the parameters to optimize. In this case, the part name of the isolation resistor, Rx, has been entered. Rx has been given a range from 0 to 100 to find an optimized value within. The That section specifies the performance criteria that the Find parameters will be optimized for. The only criteria for this circuit is to set the phase margin to 45 degrees. The performance function:

Phase_Margin(db(v(Out)))

is used. The Equates entry tells the optimizer to find the result that produces the closest fit to the specified value of 45. A click on the Optimize button initiates the optimization. The optimizer calculates that a value of 5.673 ohms for the Rx resistor will produce the desired 45 degree phase margin. Hitting the Apply button modifies the schematic to use the calculated values.

The optimized frequency response is displayed in Figure 5. The phase margin is now at a stable value of 45 degrees. The peaking in the frequency response has also been greatly mitigated.

References:
1) http://www.analog.com/Analog_Root/static/techSupport/designTools/interactiveTools/stability/stability.html#applet

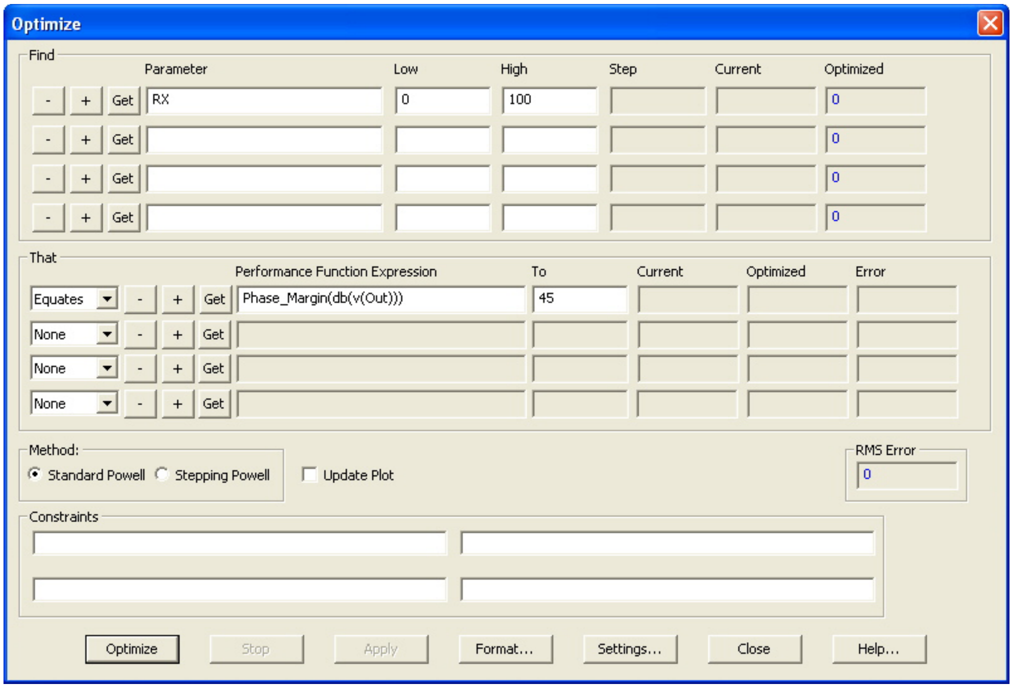


Fig. 4 - Optimizer dialog box settings

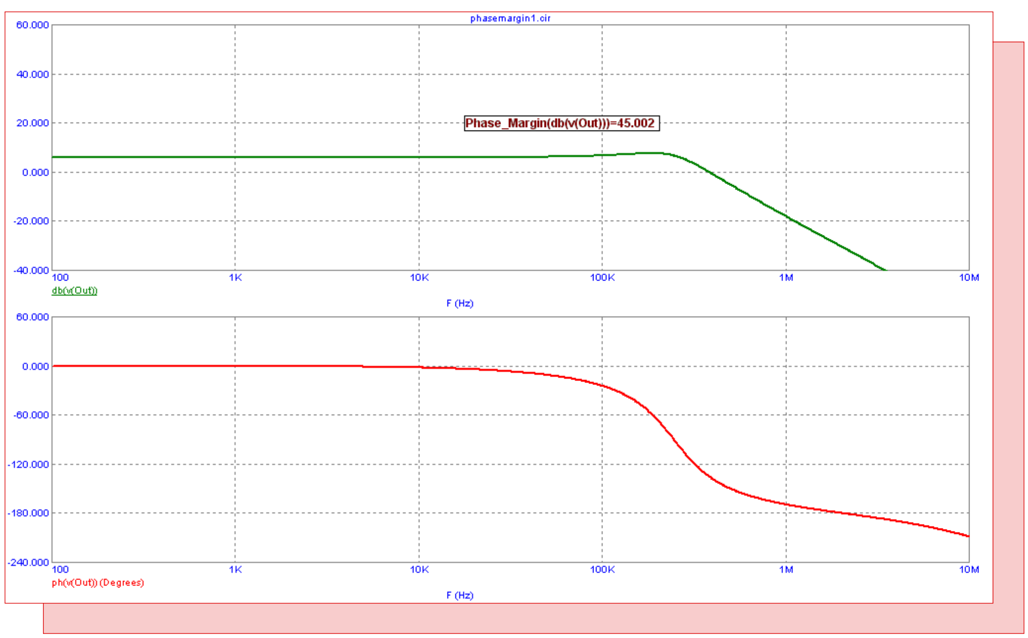


Fig. 5 - AC analysis showing a phase margin of 45 degrees

RTD Macro

The resistance temperature detector (RTD) is a sensor that is commonly used to make temperature measurements. The RTD can be made out of a variety of metals or alloys although platinum is the typical material used. The RTD provides a stable, accurate temperature measurement as the resistance of the RTD increases in a nearly linear fashion as the temperature increases. The response of the resistance versus temperature is based upon the Callendar - Van Duesen equation which is as follows:

$$R(Temp) = Rnom * (1 + A * Temp + B * Temp^2 + C * Temp^3 * (100-Temp))$$

For the popular Pt100 RTD, the above variables are defined as:

Rnom = 100 ohms
A = 3.9083m
B = -5.775e-7
C = -4.183p

where the value of the C coefficient is only applicable when the temperature is below 0C. Above 0C, the C coefficient has a value of zero. The schematic for the RTD macro appears in the figure below.

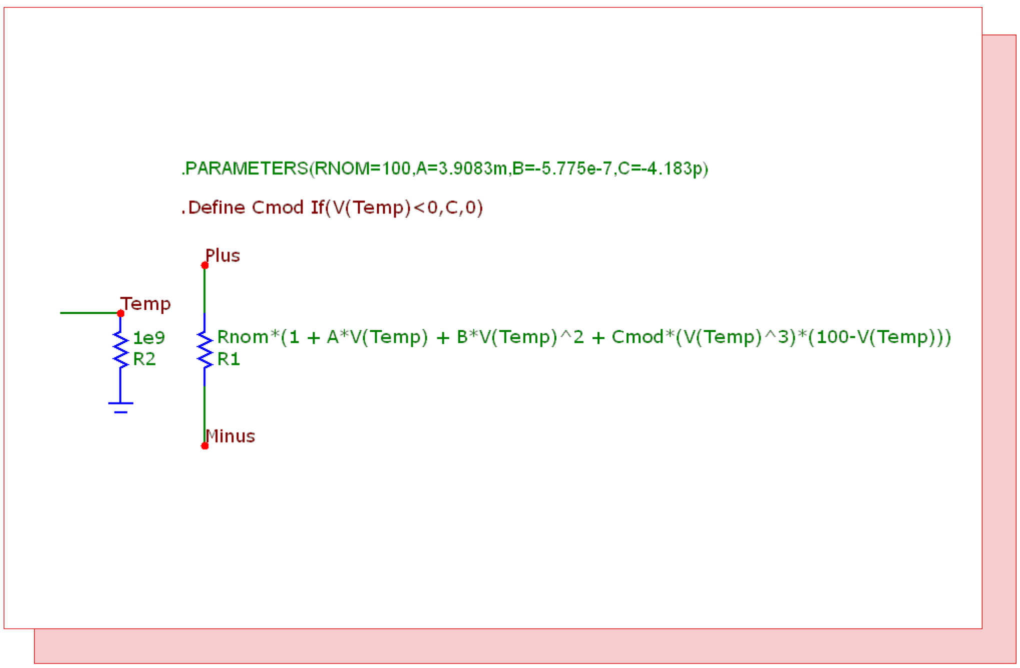


Fig. 6 - RTD macro circuit

The macro circuit has four parameters that are passed to it: RNOM, A, B, and C. The RNOM parameter is the nominal resistance at 0C. The A parameter is the linear temperature coefficient. The B parameter is the quadratic temperature coefficient. The C paramater is the quartic temperature coefficient. The default parameters settings have been defined with the values specified for the Pt100 RTD device.

The macro circuit consists of two resistors. The R2 resistor simply provides a DC path to ground for the Temp node. The voltage at the Temp node is equivalent to the temperature that the RTD component is measuring using a direct 1V:1C ratio. The measured temperature for the RTD was setup in this manner to provide the user the means of individually controlling the temperature of the device without affecting the other components in the schematic.

The R1 resistor models the RTD resistance using the Callendar - Van Duesen equation. Its RESISTANCE attribute is defined as:

Rnom*(1 + A*V(Temp) + B*V(Temp)^2 + Cmod*(V(Temp)^3)*(100-V(Temp)))

This expression has one slight modification versus the previously shown Callendar - Van Duesen equation. Rather then using the C parameter directly, the Cmod parameter is used instead which has been set through the following define statement.

.Define Cmod If(V(Temp)<0,C,0)

If the voltage at node Temp is less than zero, Cmod will be set with the value of the C parameter. If the voltage at node Temp is greater than or equal to zero, Cmod is set to zero. This models the temperature dependency of the C coefficient for the RTD. Note that the voltage at node Temp is used as the temperature reference for the above two equations.

The schematic (from Reference 1) below demonstrates the use of the RTD macro. The circuit contains two similar resistor dividers. One divider uses the Pt100 RTD device, X2, while the other uses an ideal 100 ohm resistor. The other half of each divider is a 50kohm resistor. A voltage source, VTemp, has been placed at the Temp input of the RTD macro. This source will be used to define the measured temperature of the RTD. At 0C, the two resistor dividers will produce the same volt-

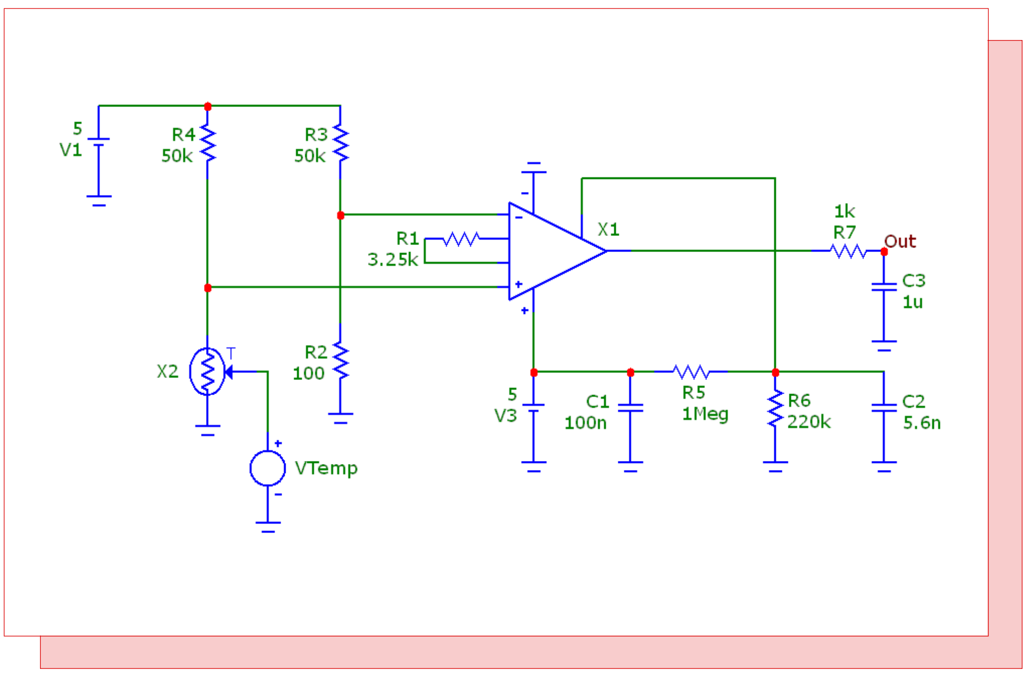


Fig. 7 - RTD interface circuit

age, but at any other temperature, a voltage difference will arise between the two due to the varying resistance of the RTD. The junction of each of these dividers is connected to the inputs of an INA326 instrumentation amplifier. The INA326 amplifies the voltage difference between the two dividers so that it can be used by devices such as ADCs.

A DC simulation is run on the circuit. The VTemp voltage source has its value linearly swept from -200V to 850V. This is equivalent to sweeping the temperature of the RTD from -200C to 850C. The simulation results appear below.

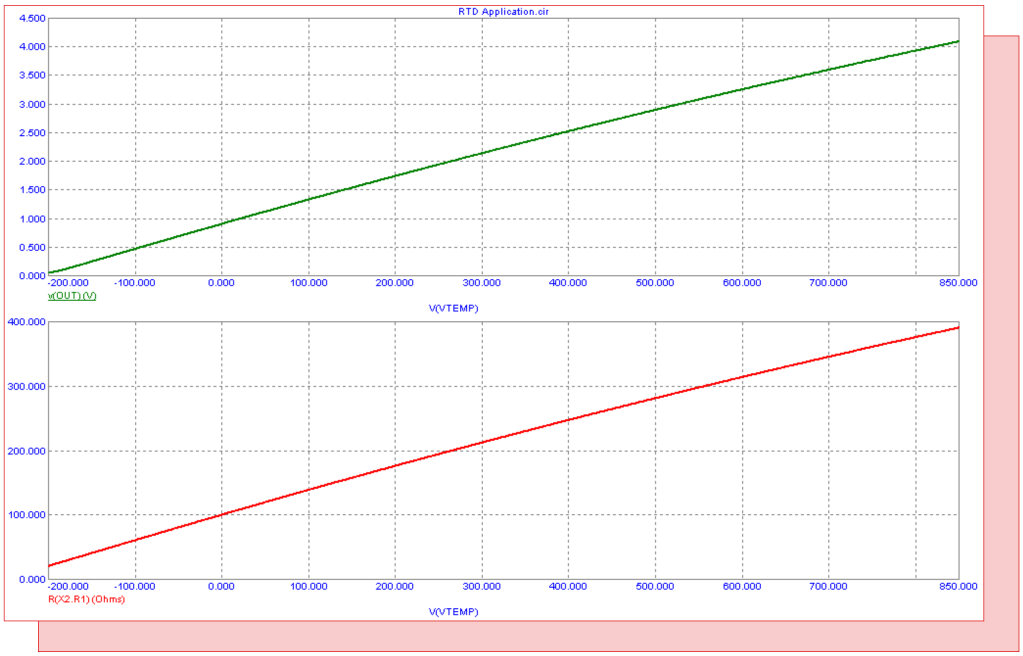


Fig. 8 - RTD interface circuit DC analysis

The top waveform displays the output voltage of the INA326 amplifier. The amplifier has been configured to produce a voltage range of approximately 0V to 4V which is right in line with the range of many ADC components.

The bottom waveform displays the resistance of the R1 resistor within the X2 RTD macro. This is a direct plot of the RTD resistance. As can be seen in the plot, at 0C the RTD resistance is at its nominal resistance value of 100 ohms.

Reference:

1) Developing A Precise Pt100 RTD Simulator For SPICE, Thomas Kuehl - http://www.analog-zone.com/acqt_052807.pdf

Frequency Weighting Filter Macro

Sound pressure level is an objective measurement in decibels of the pressure change between the sound wave and the medium it is traveling through. Since human hearing is more sensitive to certain frequencies within the audible frequency range (approximately 20Hz to 20kHz), frequency weighting is used to get a better idea of what sound pressure levels would be perceived as loud by a person. Within the audible frequency range, the human ear is less sensitive to the lower and higher frequencies. Several weighting filters have been developed to compensate for this effect. Some of the common weighting filters are the A-, B-, C-, and D-Weighting curves. The transfer functions for these four curves are well known and can be easily modeled within Micro-Cap through the use of a Laplace source. A macro that models all four of these weighting curves is shown in the figure below.

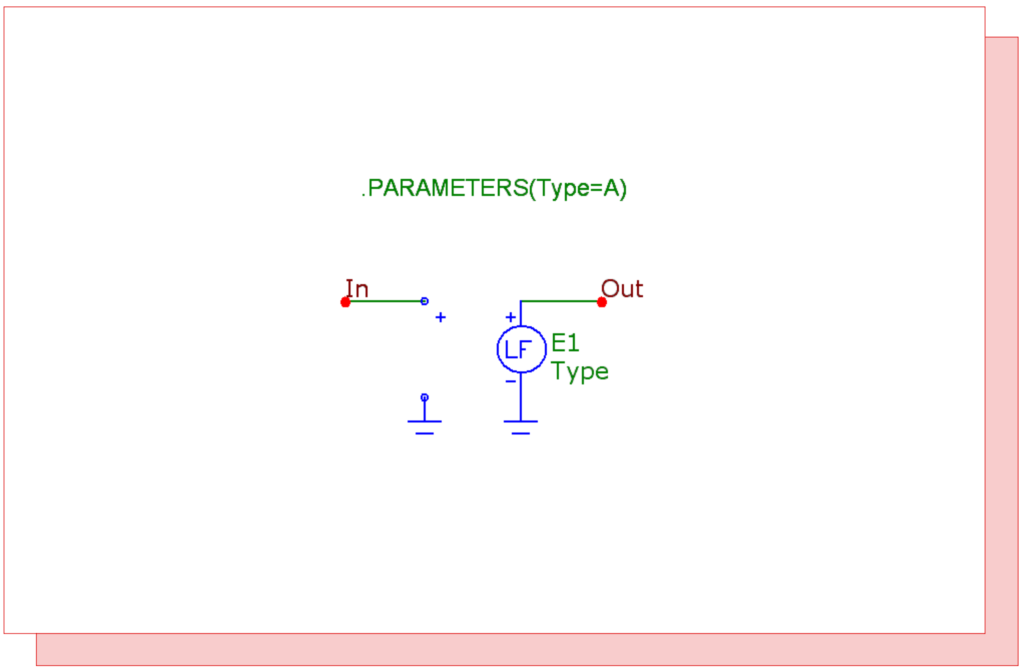


Fig. 9 - Frequency weighting macro

The macro circuit has just a single parameter that is passed through to it. The parameter is named Type, and it must be defined as either A, B, C, or D within the circuit that is using the macro. The default value of Type is A since that is the most commonly used filter of the four.

The macro consists of a single LFVofV (Laplace Formula Voltage of Voltage) component. The LAPLACE attribute for the LFVofV is defined with the Type parameter. The In node is connected to the input of the LFVofV component, and the Out node is connected to the output of the LFVofV component. Basically, the macro models the transfer function that is defined for the LFVofV. In the Text page of the macro circuit are several define statements which set the behavior of the Laplace source depending on the value of the Type parameter. The define statements are used to specify the transfer functions of the frequency weighting curves and are specified as follows:

A-Weighting
.define Ka 7.39705e9
.define A (Ka*s^4)/((s+129.4)^2 * (s+676.7) * (s+4636) * (s+76655)^2)

B-Weighting
.define Kb 5.99185e9
.define B (Kb*s^3)/((s+129.4)^2 * (s+995.9) * (s+76655)^2)

C-Weighting
.define Kc 5.91797e9
.define C (Kc*s^2)/((s+129.4)^2 * (s+76655)^2)

D-Weighting
.define Kd 91104.32
.define D (Kd*s*(s^2+6532*s+4.0975e7))/((s+1776.3) * (s+7288.5) * (s^2+21514*s+3.8836e8))

Now the way that the macro works is that the value of the Type parameter determines which of the above define statements will be used within the Laplace source. For example, if the Type parameter is specified with the value of B within the circuit using the frequency weighting macro, the LAPLACE attribute of the Laplace source will then be defined with the value B. The Laplace source will then use the corresponding define statement specified above for the B variable.

A simple circuit was created to simulate the transfer function curve of each of the frequency weighting filters. The circuit consists of just the frequency weighting macro along with a Voltage Source component which is placed at the input to the macro. The AC magnitude value of the Voltage Source has been set to one. The Type parameter of the macro was defined with the symbolic variable Weight that is set with the following define statement:

.define Weight A

In order to plot all four of the weighting transfer functions in a single plot using just one macro, the Weight symbolic variable will be stepped so that it steps through the values A, B, C, and D. The settings within the Stepping dialog box to step Weight in this manner are shown below. Since the variable is being stepped using text values, the Method used must be List. In the List field, the text strings that are to be stepped are entered in a comma delimited list.

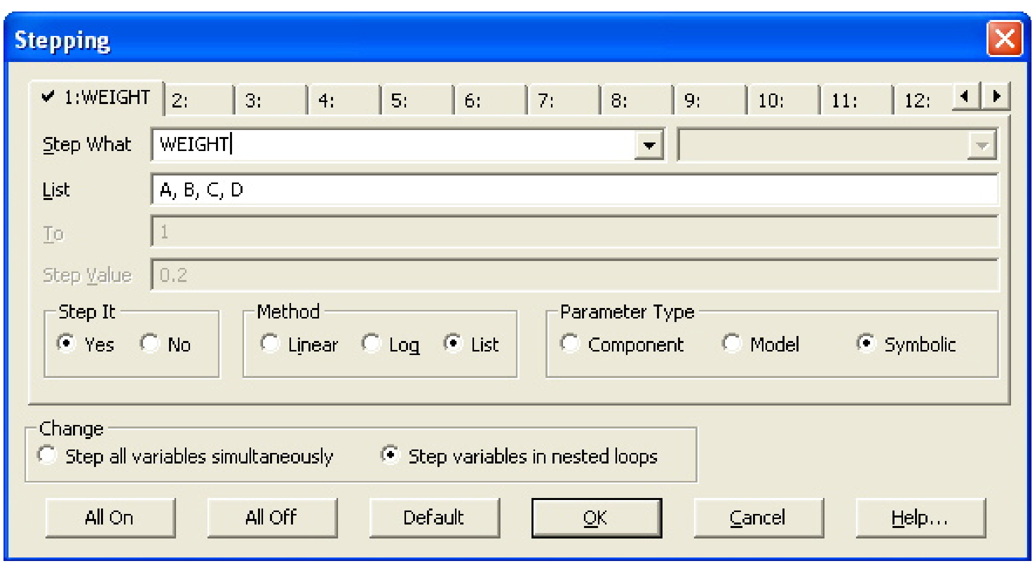


Fig. 10 - Stepping dialog box settings for the Weight variable

The resulting AC analysis is shown below. Each of the four weighting curves available in the macro have been stepped through. The plot curves match up precisely to the transfer functions specified for the weighting curves.

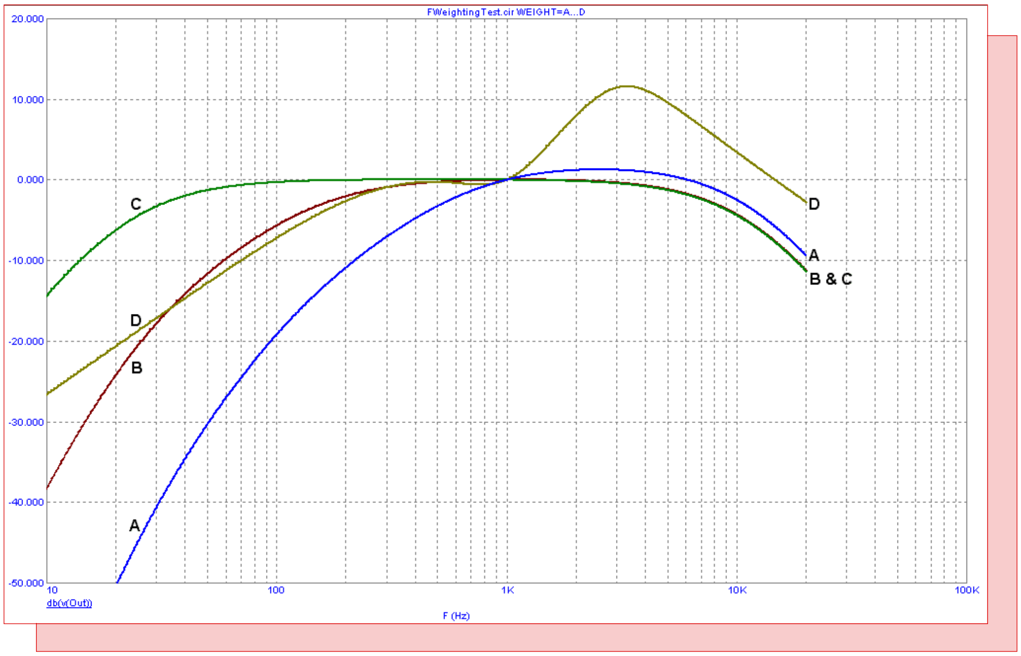


Fig. 11 - A-, B-, C-, and D-Weighting curves in AC analysis

Reference:
1) http://www.ptpart.co.uk/show.php?contentid=70#FrequencyWeighting

Product Sheet

Latest Version numbers

Micro-Cap 9 .... Version 9.0.5
Micro-Cap 8 .... Version 8.1.3
Micro-Cap 7 .... Version 7.2.4

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