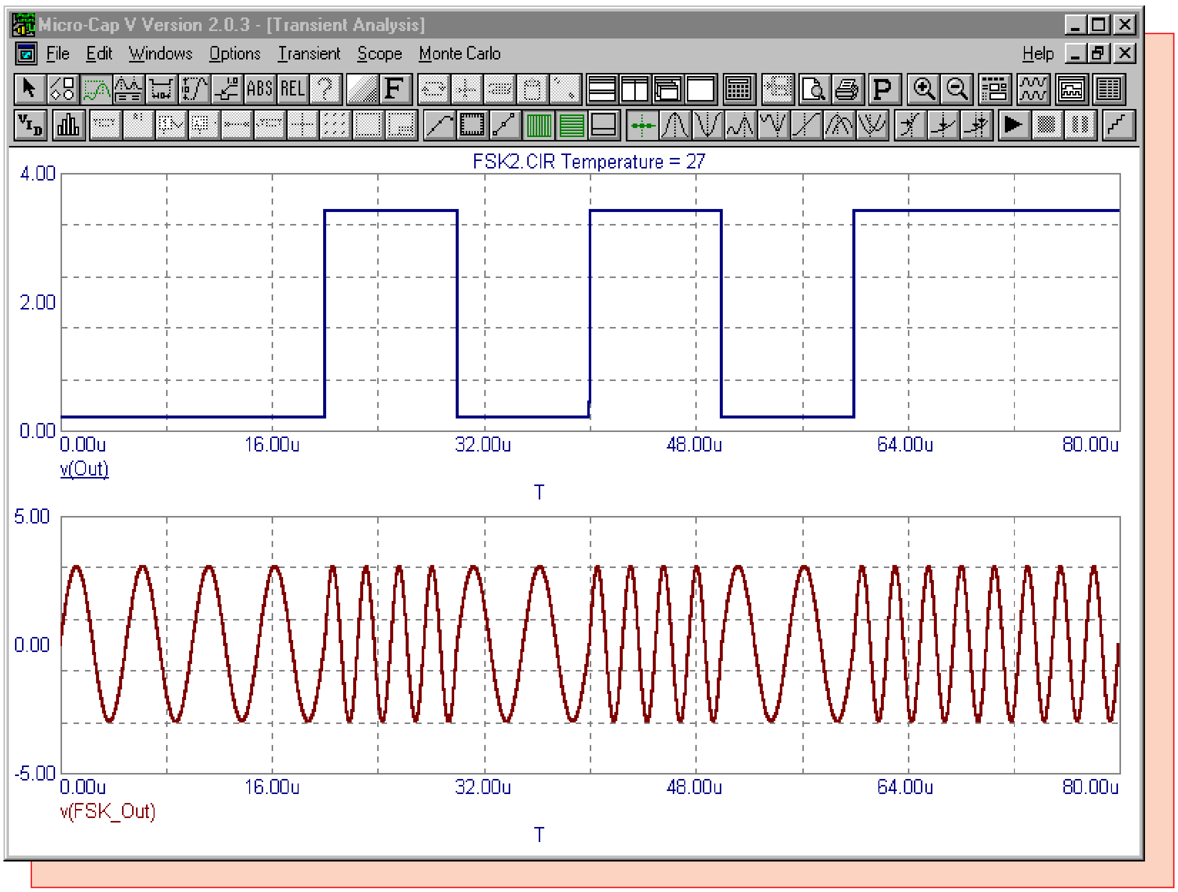
spectrum news

Applications for Micro-Cap™ Users

Summer 1998

FSK Modulation



Featuring:

- White and Pink Noise Source Macros
- FSK Modulator Macro
- AC Analysis Convergence Technique
- User Defined Functions

News In Preview

This issue features an article that describes the creation of white and pink noise sources for use in an AC analysis noise simulation. The second article describes the creation of an FSK modulator macro. This macro converts a binary input into an FSK modulated waveform which is used in communication circuits. The third article desribes a simple technique that is extremely effective when encountering a convergence problem during the operating point calculation of an AC analysis. The final article describes and gives an example of the user defined function capability available in Micro-Cap.

Contents

Book Recommendations

Micro-Cap / 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
- *Semiconductor Device Modeling with SPICE*, Paolo Antognetti and Giuseppe Massobrio McGraw-Hill, Second Edition, 1993. ISBN# 0-07-002107-4
- *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
- *SMPS Simulation with SPICE 3,* Steven M. Sandler, McGraw Hill, First Edition, 1997. ISBN# 0-07-913227-8
- *MOSFET Modeling with SPICE Principles and Practice,* Daniel Foty, Prentice Hall, First Edition, 1997. ISBN# 0-13-227935-5

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

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

Micro-Cap V Question and Answer

Question: I am running an AC analysis on one of my circuits. I have a sine source in the schematic that provides the AC stimulus. However, when I run the simulation, the output node that I am looking at is producing a constant 0 which is in error. In the AC analysis limits, I have disabled the Operating Point option and my State Variables option is set to 'Zero'. What is going wrong?

Answer: The definition of AC analysis is that it is a linear, small signal analysis. Micro-Cap constructs a small signal representation of the circuit for the analysis. Typically, Micro-Cap obtains the small signal models by linearizing the devices about the operating point calculations. However, if the Operating Point option is disabled, then it will use the settings from the State Variables option. Since it is set to Zero in this case, the circuit is creating a small signal model based on all the node voltages and inductor currents being zero which is producing the incorrect results. Either enable the Operating Point option or change the State Variables option to Read or Leave. If using the Read option, there must be a previously saved state variables file for the circuit, and if using the Leave option, there must be the correct values available in the State Variables Editor. The combination of disabled Operating Point and Zero for the State Variables Option should never be present for AC analysis.

Question: I am using an integrator macro in my circuit. The input to this macro is a battery, and I would expect to see a linear waveform with a slope equivalent to the battery voltage as my output. The output I am receiving is a DC waveform at 100KV. What is happening?

Answer: In the Transient Analysis Limits, disable the Operating Point option. This will start the transient analysis with all node voltages and inductor currents at zero instead of starting at the DC operating point calculation. The operating point calculation is skewing the output of the integrator because by definition the integrator needs to know the value of the previous data. When the operating point is disabled, it can use the zero values as its previous data. The operating point should be disabled for transient analysis when the INT or DIF macros are used in a schematic.

Question: I have just upgraded to Micro-Cap V from Micro-Cap IV. In Micro-Cap IV, I had created macros for many of my circuits. Is there a way I can easily import these macros into Micro-Cap V, so that I don't have to regenerate all of this information?

Answer: Yes, Micro-Cap V has the capability to merge previous versions of the Component Editor and Shape Editor information into the most recent version. First of all, the actual macro circuits (*.CIR files) will need to be copied into your DATA subdirectory, directory where your circuits will be run from, or any directory that is referenced by the MC5DATA environmental variable. The macro circuits will need to be loaded into Micro-Cap V and saved to update them to the present format before they may be used in a simulation. Simply click on File and Open, then double click on the appropriate circuit. Click on File and Save. Then click on File and Close, and the macro circuit is converted. Next, click on the Windows menu and choose Component Editor. In the Component Editor, there is a Merge command button. Click on this button and a dialog box will open that lets you pick which file to merge into the current version. The file that needs to be merged from MC4 is called CS.MC4, or you may also merge previous MC5 COMP*.MC5 files. All components that are not in the current version will be merged into it. Close and save the Component Editor. The macros will be ready for use. Macro information from MC3 and earlier versions will need to be regenerated in the Component and Shape Editors.

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 an icon or a menu item.

Multiple Y Scales

Micro-Cap has the capability of having multiple Y scales for a single analysis plot group. With multiple Y scales, it is easy to compare diverse waveforms on the same plot. For example, a current and a voltage waveform may each be analyzed in the same plot with appropriate scales for each waveform. This lets both waveforms be visually compared without having to zoom or scale the plot. To enable multiple Y scales, simply go to the desired analysis, click on the Scope menu and disable (no check mark) the Same Y Scales option. Upon running the analysis, a separate Y scale will be produced for each unique entry in the Y Range text field. If two or more waveforms share the same Y Range value, then only one scale will be created for those waveforms. The color of the shared scale would be the color of the first waveform in the Y Expression text field column that shares the scale.

Figure 1 displays the transient analysis results when the file PRLC.CIR is simulated over 1us. In this analysis, a voltage waveform, V(1), with a scale of 8V to -2V and a current waveform, I(L1), with a scale of 400mA to -100mA are both plotted in the same graph.

The default state for Micro-Cap is for single scales. To change the default state for all future circuits, click on the Options menu and go to Preferences. In the Preferences dialog box, click on Analysis/Performance Defaults and disable the Same Y Scales option. All new circuits would then have multiple scales as their default.

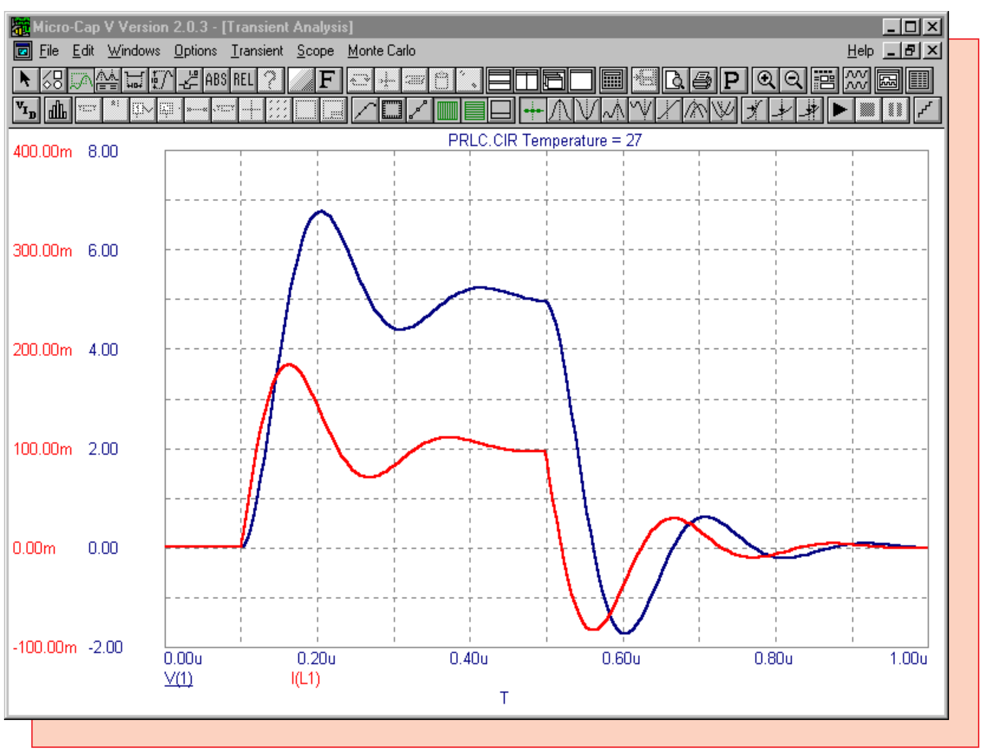


Fig. 1 - Multiple Y Scales

White and Pink Noise Source Macros

In Micro-Cap, the only components that contribute noise for a noise analysis are resistors, diodes, BJTs, MOSFETs, JFETs, and GaAsFETs. For some components, such as an operational amplifier which has a specified input noise voltage and current, a noise source will need to be added into the schematic to model this appropriately. In order to insert a noise source directly into a schematic, a macro circuit will need to be created to model the source. The following article entails the creation of four macros to model a white noise source and a pink noise source for both current and voltage. These macros are only applicable during a noise analysis.

White Noise Source
White noise has equal energy over frequency. For example, the amplitude of the sound will be the same from 400Hz to 500Hz as it will be from 30100Hz to 30200Hz. White noise is considered a bright noise and is effective in masking other sounds.

The circuit in Figure 2 is the white noise current source macro, and the circuit in Figure 3 is the white noise voltage source macro. NMAG is the only parameter for each of these macros. It is used to define the magnitude of the noise current or voltage. These macros consist of a resistor, a battery, and a function source. The resistor is used to create a noise current. The noise current that is produced from a resistor is defined by the equation:

I = sqrt(4*K*T / R)

Rearranging this equation to determine the resistance needed to produce the specified noise current, I=NMAG, produces the equation:

R=(4*K*T) / (NMAG*NMAG)

where K is Boltzmann's constant, 1.38e-23, and T is the temperature in Kelvin. In these macros, the temperature has been arbitrarily defined as 300K since that is the default temperature of an analysis. The resistor is connected to a battery that has a value of 0V. The battery is present for two reasons. The first reason is to complete a valid circuit loop with the resistor that the noise current can flow through, and the second reason is that the battery will measure the noise current from the resistor. The battery has been given a 0V value so that it may act as a short and have no effect on the resistor. The combination of the resistor and battery must be isolated from any other circuit elements.

The final element in each macro is the function source. The function source transfers the noise created by the resistor to the desired spot in the circuit and is the output of the macro. The current noise source uses the NFI source and the voltage noise source uses the NFV source. This is the only difference between the two macros. Both of the function sources have been defined as I(V1) to set the value of the sources to the value of the current through the battery which is also the noise current generated by the resistor. The function source has essentially taken the noise current produced from the resistor and has produced an equivalent noise current or noise voltage across its nodes.

The plot in Figure 4 displays the noise produced by the white noise voltage source during a noise analysis. The parameter NMAG has been defined as 100n in this instance. As can be seen in the plot, a noise of 100nV/sqrt(Hz) is present through the entire frequency range. This is to be expected due to the equal energy over frequency characteristic of white noise.

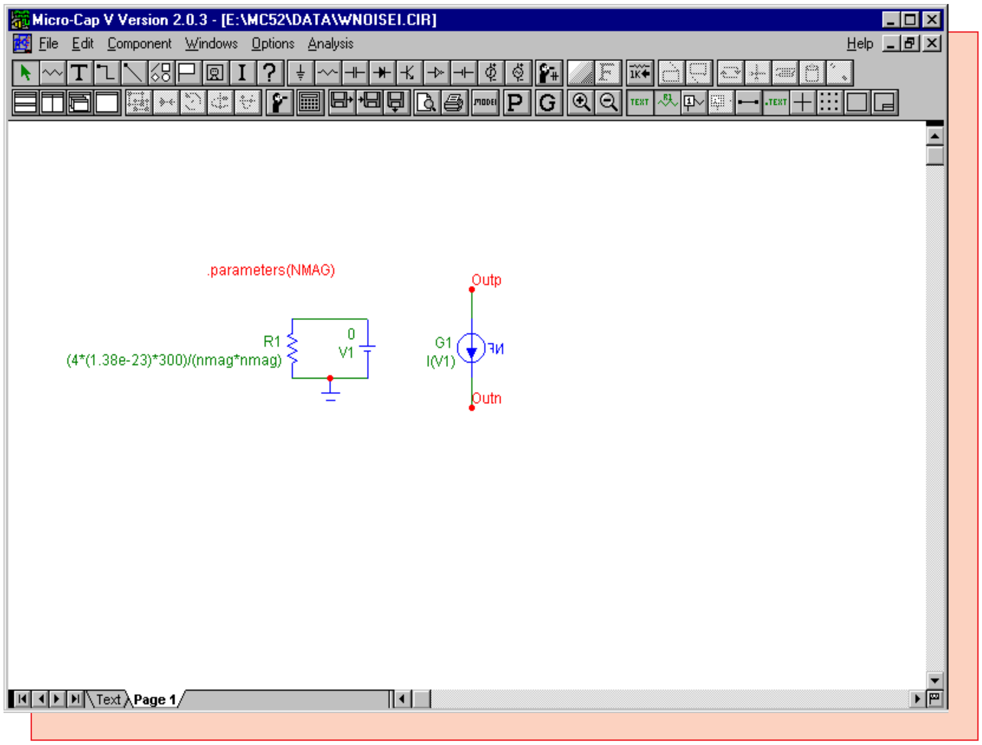


Fig. 2 - White Noise Current Source Macro

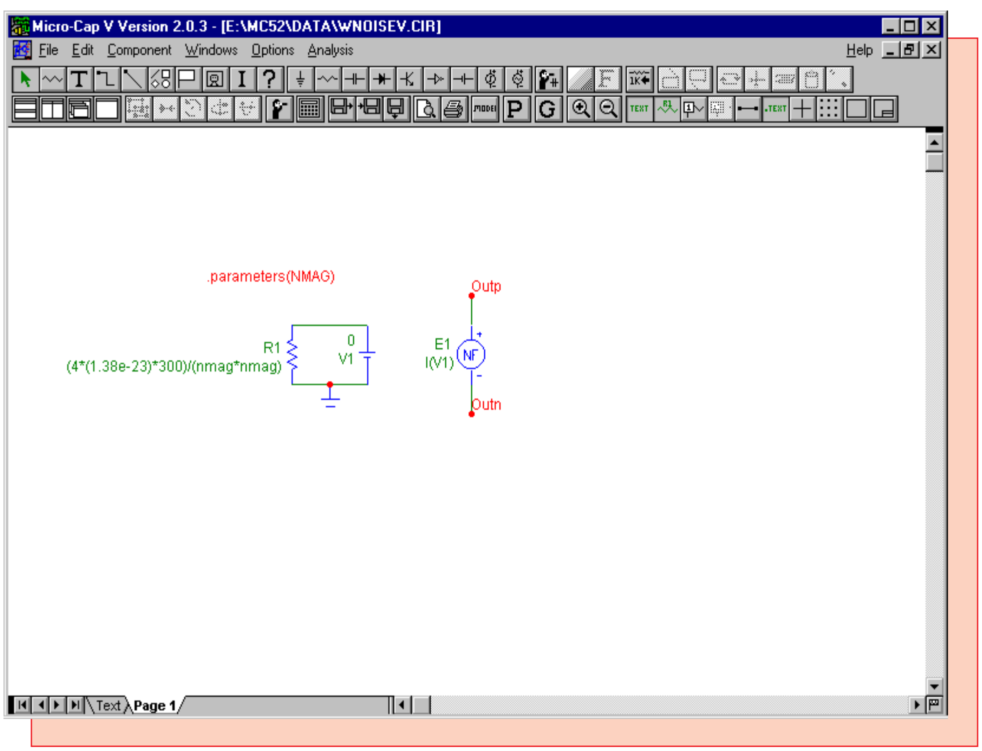


Fig. 3 - White Noise Voltage Source Macro

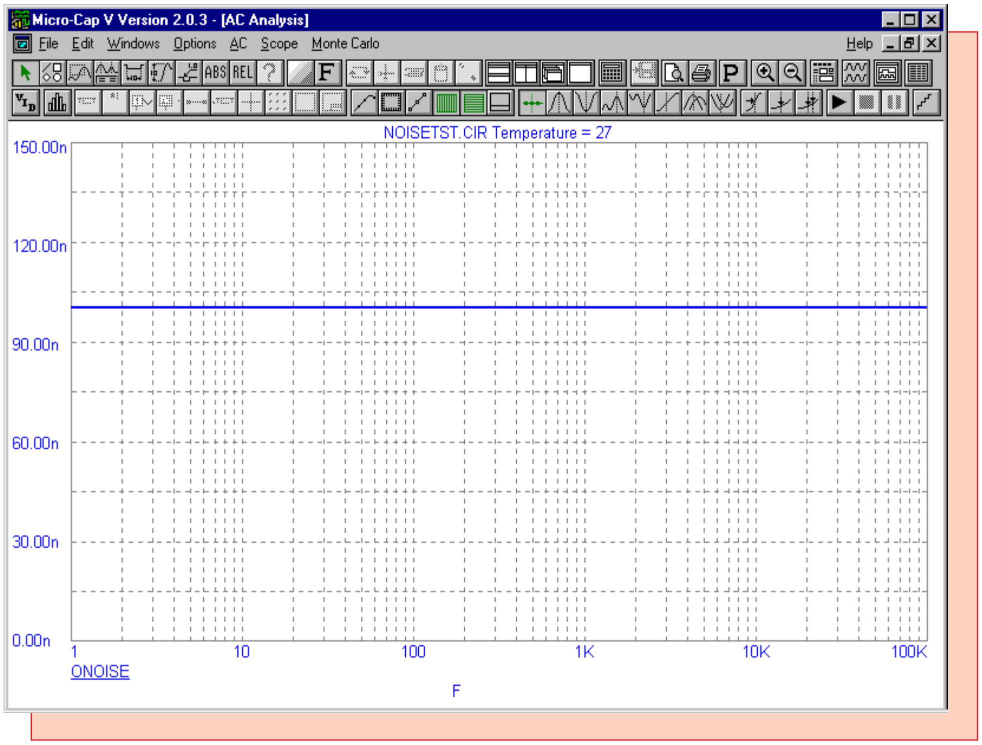


Fig. 4 - White Noise Output

Pink Noise Source

Pink noise has equal energy per octave of frequency. For example, the amplitude of the sound will be the same from 100Hz to 200Hz as it is from 200Hz to 400Hz or 10000Hz to 20000Hz. Pink noise has a more natural sound than white noise and is often used as a test signal to tune equalizers or audio spectrum analyzers. Pink noise can be created by low pass filtering or band pass filtering white noise.

The circuit in Figure 5 is the pink noise current source, and the circuit in Figure 6 is the pink noise voltage source. The three parameters in these macros are NMAG, LF, and HF. NMAG defines the magnitude of the noise current or voltage at the mid-band magnitude. LF defines the low -3dB frequency of the band pass filter, and HF defines the high -3dB frequency of the band pass filter. These macros consist of the white noise current source in conjunction with a band pass filter. The R1, V1, and G1 components create an exact replica of the white noise current source. The white noise current is filtered through the equivalent of a parallel RLC filter. G2 is a GIofV dependent source and is defined with a VALUE attribute of:

FiltOut 0 1

This source models a 1 ohm noiseless resistor. The input to the dependent source is the voltage at node FiltOut which is also the voltage across itself. The output of the source is a current equal to the voltage at FiltOut. Since I and V are the same, R=V/I calculates to be R=1. The GIofV dependent source needs to be used instead of a resistor because the resistor would add unwanted noise to the macro.

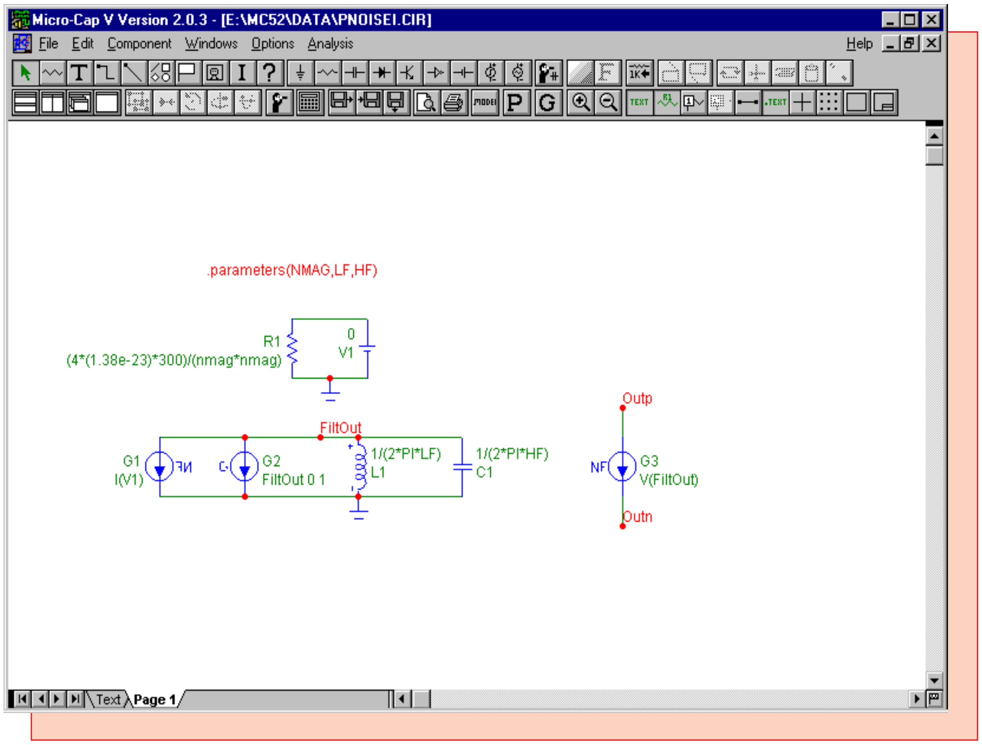


Fig. 5 - Pink Noise Current Source Macro

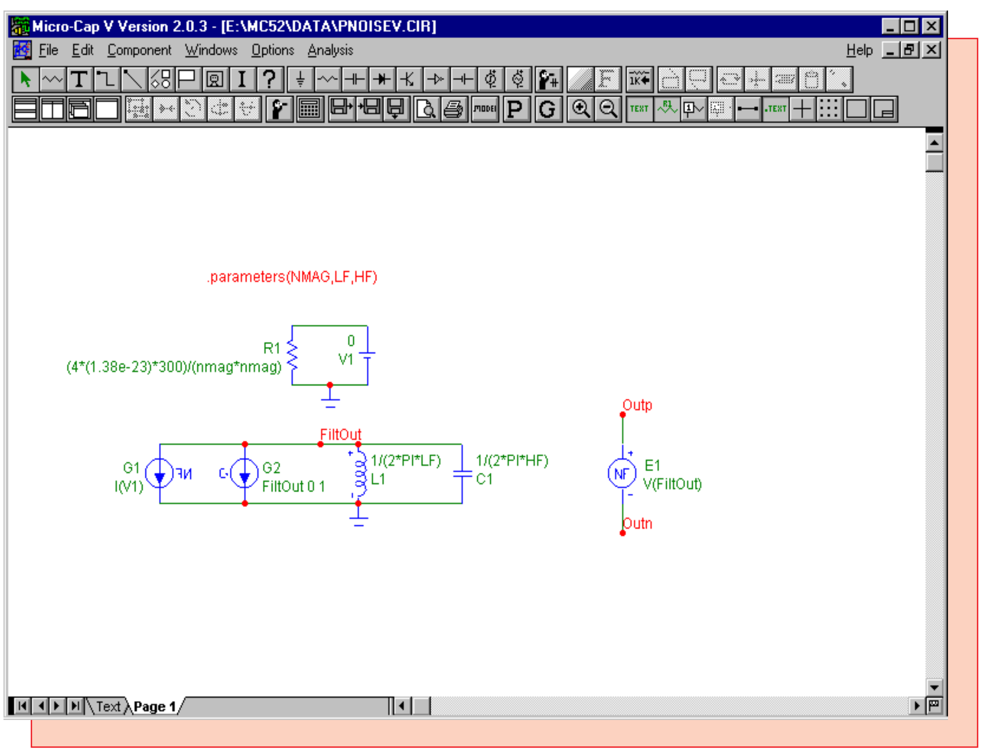


Fig. 6 - Pink Noise Voltage Source Macro

The inductor and the capacitor are then defined so that they will have a reactance of 1 ohm at the specified frequencies, LF and HF, respectively. The inductor has its VALUE attribute defined as:

1/(2*PI*LF)

to set up a -3dB point at the LF frequency, and the capacitor has its VALUE attribute defined as:

1/(2*PI*HF)

to set up a -3dB point at the HF frequency. The only difference between the current and voltage macros is the type of function source used for the output. The current source macro uses the NFI source, and the voltage source macro uses the NFV source. Both function sources have their VALUE attribute defined as V(FiltOut). The function sources are transferring the band pass filtered noise of the resistor to produce an equivalent noise current or noise voltage at the output of the macro.

The plot in Figure 7 displays the noise analysis of a pink noise voltage source. The macro was defined with NMAG at 100n, LF at 10Hz, and LH at 20KHz. As can be seen in the plot, the -3dB points occur at approximately 10Hz and 20KHz. The mid band magnitude of the noise is at 100nV/sqrt(Hz).

These pink source macros use band pass filtering of the noise, but in some cases, low pass filtering may need to be used. To change the macro to a low pass filter, delete the inductor from the macro circuit, and eliminate the LF parameter from the .parameters statement. Another method is to define the LF parameter with an extremely small value to essentially act as a low pass filter.

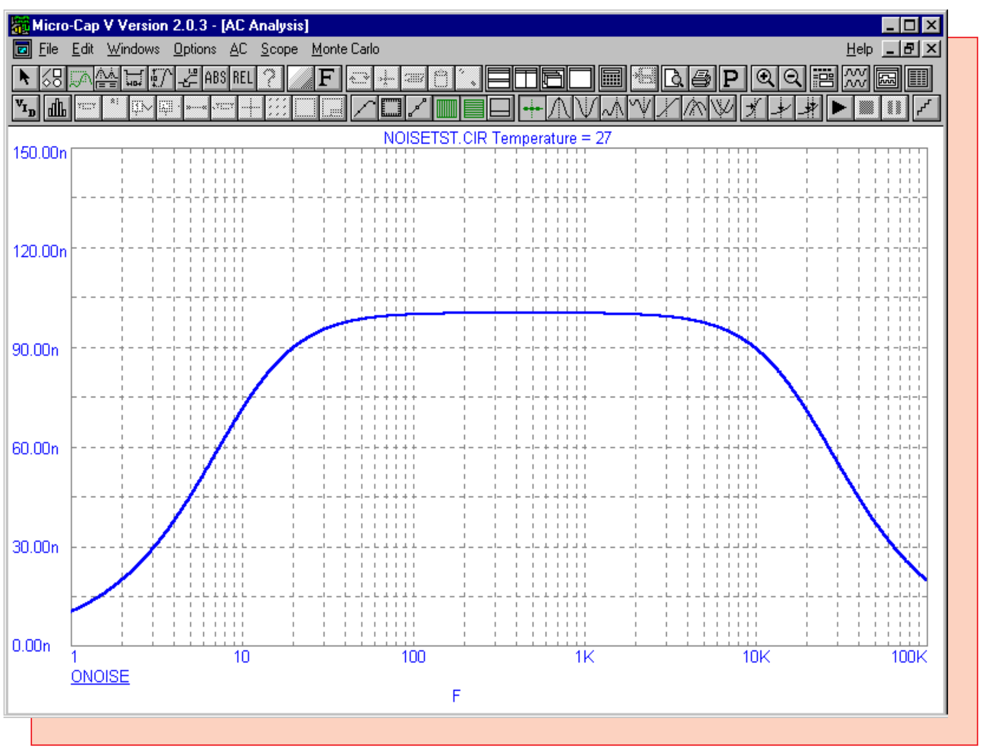


Fig. 7 - Pink Noise Output

FSK Modulator Macro

In the previous issue of the newsletter, Spring 1998, a PSK modulator macro was introduced that provided a phase modulation technique for superimposing a binary PCM (Pulse-Code Modulation) signal onto a carrier signal. A second modulation technique for this procedure is frequency modulation. Frequency modulation, also known as frequency shift keying (FSK) modulation, produces a waveform with the following characteristics:

$$V_{FSK}(t) = A * \cos((wo + k) * t)$$

where A is a fixed amplitude, and the frequency will vary according to the binary PCM waveform. For example, if the length of a single bit of the binary PCM waveform is 10us, an FSK modulator may produce a waveform where the frequency is 200KHz when the binary signal is at a zero state and 400KHz when the binary signal is at a one state.

The macro circuit for an FSK modulator appears in Figure 8. This macro can accept either a digital or an analog waveform as its binary PCM input signal and will produce the corresponding FSK modulated waveform at its output. The FSK macro accepts four parameters: WMAG, NC0, NC1, and TB. WMAG defines the magnitude of the output waveform. NC0 defines the number of cycles of the output waveform that will occur in the duration of a single zero bit of the input waveform. NC1 defines the number of cycles of the output waveform that will occur in the duration of a single one bit of the input waveform. TB defines the duration of a single bit in seconds.

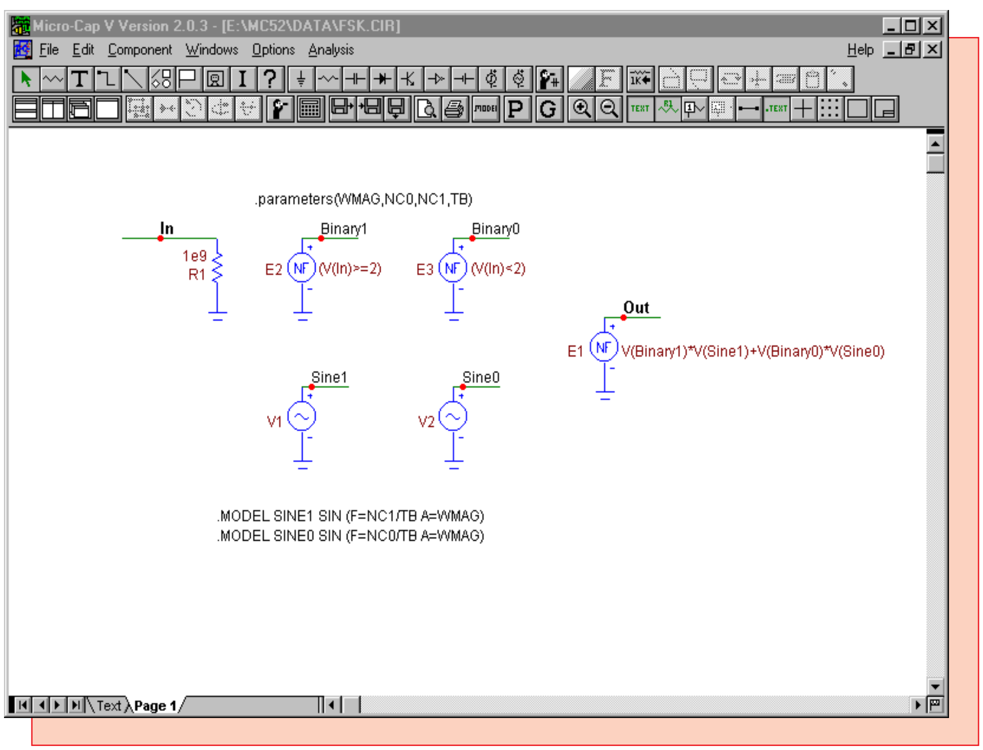


Fig. 8 - FSK Modulator Macro Circuit

The macro consists of a resistor, two sine sources, and three nonlinear function voltage sources. The resistor is present at the input node for two reasons. The first reason is to provide a DC path to ground for the In node so that any element may be connected to it. The second reason is that the NFV sources are only able to work with analog references. If the input waveform happens to be a digital waveform, the resistor will force Micro-Cap to convert it to its equivalent analog voltage for use with the nonlinear function voltage sources, E2 and E3. The value of the resistor is set to 1e9 ohms so that it will not have a loading effect. The NFV source, E2, has its VALUE attribute defined as:

(V(In) >= 2)

This equation will convert the input waveform into a signal that has a value of either 1V or 0V. If the voltage at node In is greater than or equal to 2V, than the source will produce an output of 1V, if it is less than 2V, then the output of the source will be 0V. The 2V threshold is arbitrary. In this case, it was setup to handle a standard TTL digital waveform. The threshold should be approximately in the middle of the one and zero state voltages of the input waveform. The NFV source, E3, has its VALUE attribute defined as:

(V(In) < 2)

This source acts in the same manner as the E2 source. However, this source will produce a 1V output whenever the input signal is less than 2V and a 0V output whenever the input signal is greater than or equal to 2V. These two NFV sources will produce a pair of inverted pulse waveforms that correspond to the input PCM signal.

The two sine sources, V1 and V2, have been defined using the following model statements:

```
.MODEL SINE1 SIN (F=NC1/TB A=WMAG)
.MODEL SINE0 SIN (F=NC0/TB A=WMAG)
```

The V1 source, with the SINE1 model statement, models the output waveform when the input signal is at a one state. The V2 source, with the SINE0 model statement, models the output waveform when the input signal is at a zero state. The amplitude of each source is defined by the parameter WMAG. The frequency of the sources are determined by the ratios of the NC1 and NC0 parameters to the TB parameter. NC1/TB determines the frequency of the modulated waveform at the one state of the input PCM signal, and NC0/TB determines the frequency of the modulated waveform at the zero state of the input PCM signal.

The NFV source, E1, produces the modulated output waveform from a combination of the four other sources. Its VALUE attribute is defined as:

V(Binary1)*V(Sine1)+V(Binary0)*V(Sine0)

The E1 source takes the product of the voltages at nodes Binary1 and Sine1 and the product of the voltages at nodes Binary0 and Sine0 and sums them together. Since the voltage at either Binary1 or Binary0 must be zero at any given time, the E1 actually alternates between the two sine source voltages depending on the input waveform. If the input signal is greater or equal to 2V, then the output will be the voltage at Sine1, and if the input signal is less than 2V, the output will be the voltage at Sine0.

Figure 9 displays the Component Editor settings for the FSK macro. The FSK shape was created in the Shape Editor, and the component has been defined as Macro. The two pins are named In and Out which correspond exactly to the node names in the FSK macro circuit.

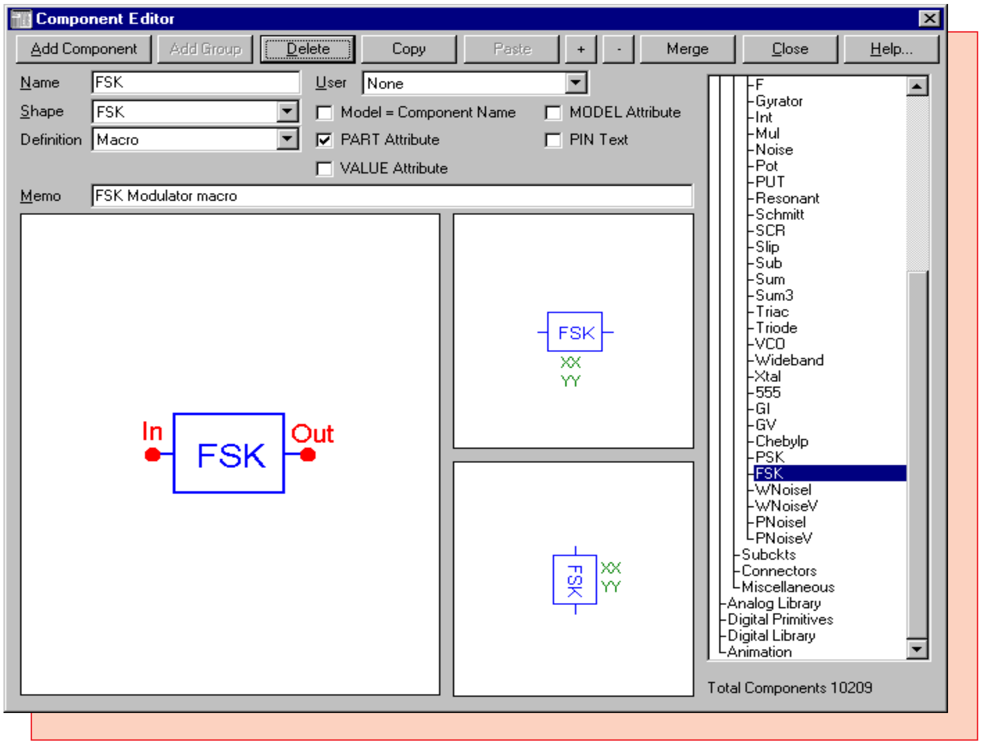


Fig. 9 - Component Editor Settings

The test circuit for the FSK macro is displayed in Figure 10. The circuit consists of a simple two bit shift register that feeds the FSK macro. The two D flip-flops are defined with zero gate delays. The clock input to the flip-flops has a period of 10us. The preset inputs have been wired to a Fixed Digital component which produces a constant one state on those pins. The clear inputs have a short zero state pulse of 100ns to initialize the Q outputs of the flip-flops to zero at the beginning of the simulation. The FSK macro has its VALUE attribute defined as:

FSK(3,2,4,10u)

This will produce an FSK modulated waveform with a magnitude of 3V. The frequency when the binary input signal is at a zero state will be 200KHz, and the frequency when the binary input signal is at a one state will be 400KHz.

Figure 11 displays the results of an 80us transient analysis simulation. The top waveform, V(Out), is the binary input to the FSK macro. The bottom waveform, V(FSK_Out), is the modulated waveform. Note the change in frequencies when the input is at a zero or one state. As can be seen in the figure, the modulated waveform produces two cycles for a single zero bit and four cycles for a single one bit.

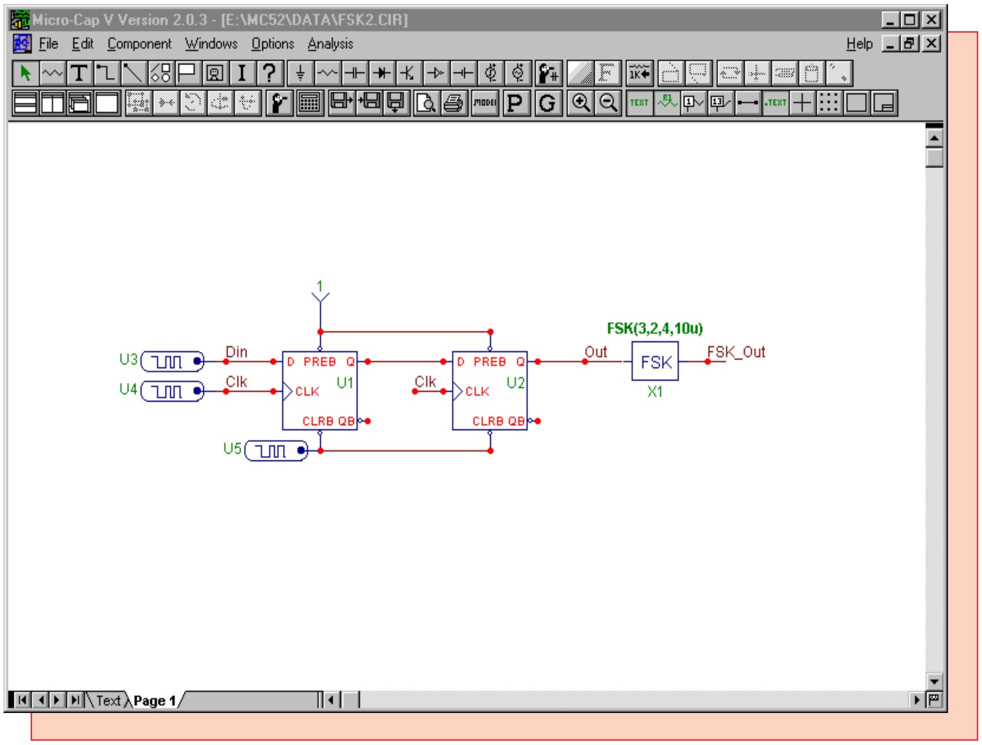


Fig. 10 - FSK Macro Test Circuit

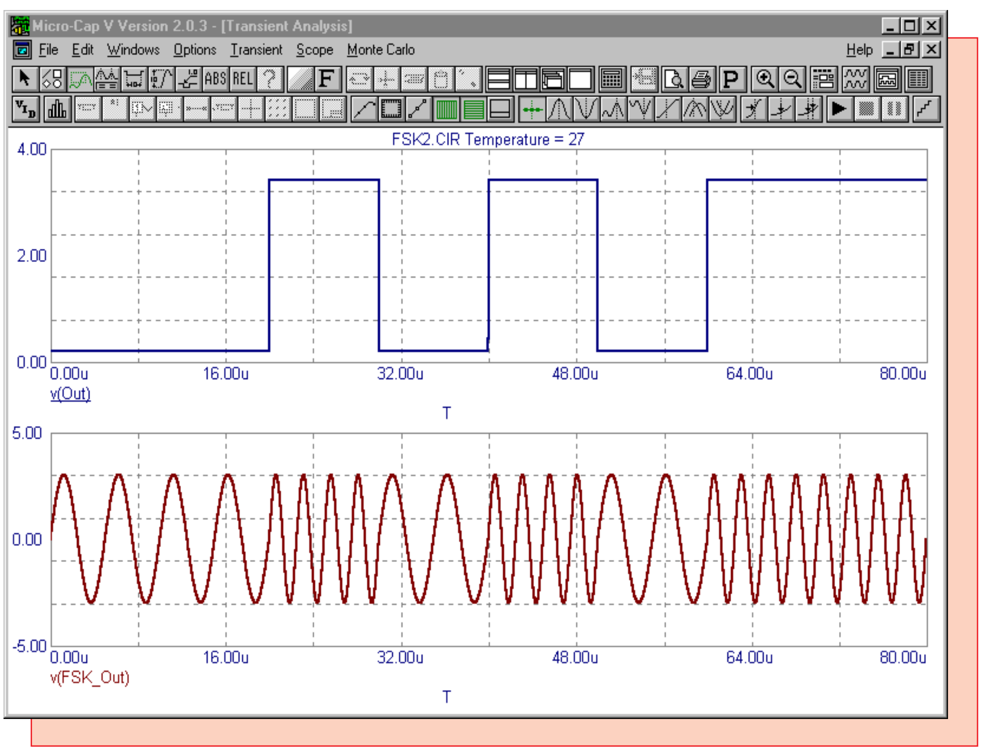


Fig. 11 - FSK Macro Test Transient Analysis

AC Analysis Convergence Technique

AC analysis is by definition a small signal, linear analysis. The circuit is linearized for AC simulation through an operating point calculation or by reading in State Variables. In most cases, the operating point calculation will be used. However, a circuit may have trouble converging during the operating point calculation or the user may want to run the AC analysis at a point different than the operating point results. Common types of circuits that may have convergence problems with an operating point calculation are integrators, oscillators, or circuits with multi-stable states. When encountering a convergence error in AC, a simple technique involving the reading in of State Variables can resolve most problems. This technique gathers steady state information from a transient analysis and uses it in linearizing the circuit for AC analysis.

This convergence technique will be demonstrated on the circuit in Figure 12. The circuit consists of an operational amplifier in a gain configuration. The opamp used is a INA118 precision, low power instrumentation amplifier from Burr-Brown. The input to the opamp is a sine source with 10Kohm resistors to ground at each lead. The opamp is being supplied with +/-15V at its power supplies, and the external gain resistor for the opamp has been defined as 20 ohms. If an AC analysis is run with the Operating Point option in the AC analysis limits enabled and the State Variables option set to Zero, which are the default options, then Micro-Cap will return with the error:

Failed to converge in specified number of iterations at time=0.

This error means that Micro-Cap was unable to converge on a solution during the operating point calculation before the number of iterations specified by ITL1 was reached. ITL1 may be changed by editing its value in the Global Settings under the Options menu while in the schematic editor.

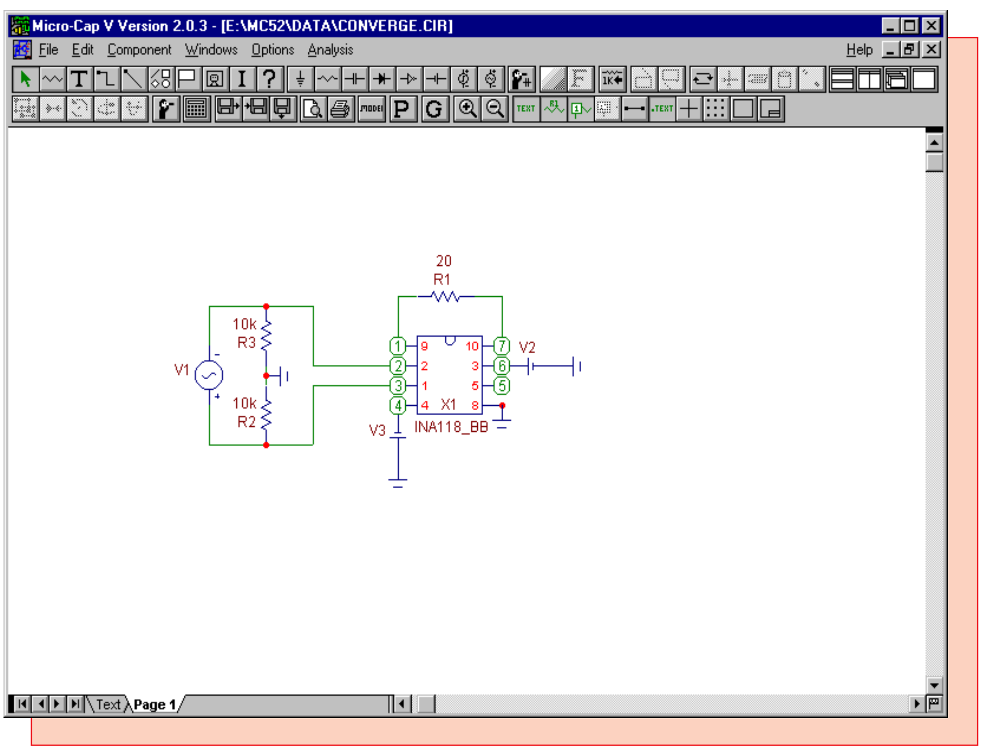


Fig. 12 - AC Convergence Circuit

Upon receiving the error, the steps to achieve convergence for this circuit are as follows:

1) Return to the schematic. The circuit must now be set up so that all sources are only DC. It is possible to replace sources for this portion, such as replacing a sine source with a battery, but note that editing the circuit may change the node number order. The best method is to edit the parameters of the sources. For example, set the F parameter of a sine source to 0, or set the VONE parameter of a pulse source to the value of the VZERO parameter. This will create DC sources out of these components. Any time-based user equation that is used in the circuit must have its time variable changed to 0. There is also no need to change these values back for the AC analysis as they are not used at all for that simulation. In the example circuit, the V1 sine source has its frequency changed to 0.

2) Enter transient analysis. In the transient analysis limits, disable the operating point. Set the time range long enough for any initial transients to dissipate. In this case, the time range has been set to 10ms. Run the transient simulation. The results appear in Figure 13. Notice how the V(5) waveform has settled into its steady state value.

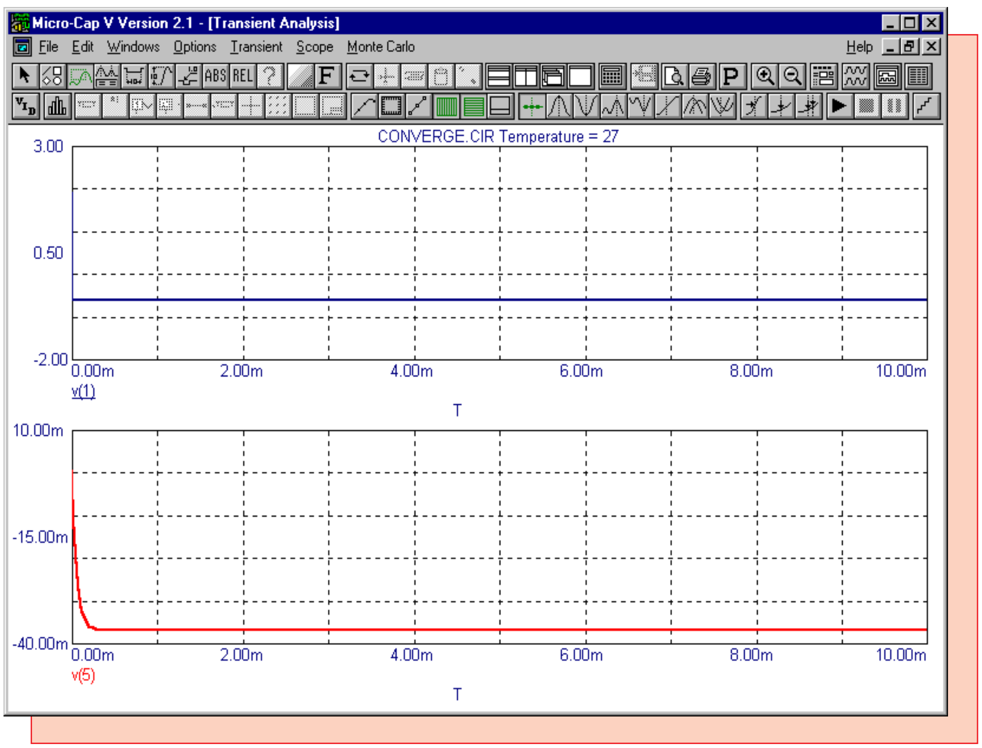


Fig. 13 - Transient Analysis of the Convergence Circuit

3) Once the simulation has finished and it is apparent that the simulation has reached its steady state value, enter the State Variables Editor. The State Variables Editor is used to review, edit, or save the values of the state variables for the node voltages, inductor currents, and digital states. For transient analysis, the state variables contain the information from the last data point calculated. The State Variables Editor may be accessed by hitting F12 or by choosing State Variables Editor under the Transient menu. Once in the editor, click on the Write command button. This creates a file on the hard drive with the extension .TOP which contains all of the information from the State Variables Editor. Close the State Variables Editor.

4) Exit transient analysis. Enter AC analysis. In the AC analysis limits, disable the Operating Point option, and set the State Variables option to Read. Run AC analysis. Instead of calculating an operating point, Micro-Cap will look for a file with the same name as the circuit name and the extension .TOP. It will import the data found in this file and linearize the circuit around this data. In this case, the data from the .TOP file is the data derived from the steady state information gathered in the transient analysis. The resulting AC analysis appears in Figure 14.

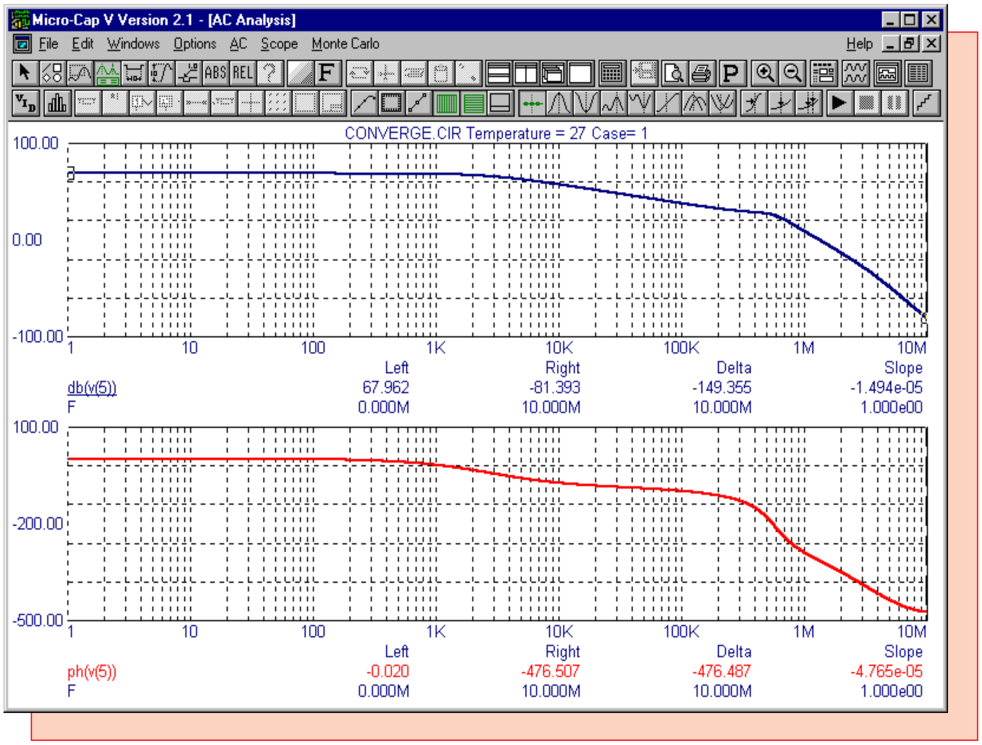


Fig. 14 - AC Analysis of Convergence Circuit

The gain for this opamp is defined by the equation:

$1 + 50k/R_G$

where R_G is the value of the gain resistor. With a 20 ohm gain resistor, the gain comes out to be 2501 or 67.962dB. The waveform db(v(5)), which is the output of the opamp, matches exactly with this result at low frequencies as shown in the cursor display in Figure 14.

This technique is successful because the transient analysis is able to calculate the necessary DC steady state information that the operating point calculation is failing on and supply it to the AC analysis through the Read operation. It can also be used to run an AC simulation on a single circuit in different states such as in saturation or linear operation without the time-consuming step of modifying the circuit. Simply run transient analysis till it reaches the desired state, save the data, and import it into AC analysis for simulation.

User Defined Functions

Micro-Cap provides the user the capability to create their own functions from the operators and functions currently available in the program. This can be accomplished through the use of the .DEFINE command statement in the following format:

.DEFINE <name[(<p1>[,<p2>][...,<pn>])]>   f([<p1>][,<p2>][...,<pn>])

where f() is some expression involving the parameters p1, p2, ... pn. <name> can be any valid, unique name, and the optional parameters p1 to pn can be used to pass a variety of values through to the expression. Parameter values may consist of constants, PART attribute names, expressions, other .DEFINE parameters, or any other elements that are valid within the defined function. When the circuit is run, the parameter values will be substituted into the expression.

These user defined functions can be either local or global. A local function can be defined by typing the .DEFINE statement into the text area or into the schematic area and would only be available for that circuit. A global function is defined by typing the .DEFINE statement into the DEF.MC5 file which is accessed by clicking on the User Definitions option under the Options menu. This procedure makes the function available for all circuit simulations. For either method, the function will be available for use in the top level schematic or in the analysis of the circuit.

A common use of the user defined function capability would be to create a function for any often used expressions. The circuit in Figure 15 has two functions defined in its schematic as:

.define Power(R1) V(R1)*I(R1)
.define Energy(R1) SUM(V(R1)*I(R1),T)

The first expression calculates the power through any two node element such as a resistor, capacitor, inductor, diode, or source. The second expression calculates the energy through any two node element. While R1 as the parameter name seems to imply the use of a resistor, any PART attribute of a two node element may be passed through to the function. The Power(R1) function is available for schematic or analysis use. The Energy(R1) function is only available for use in transient analysis due to the presence of the SUM operator and the T variable.

The transient analysis results are displayed in Figure 16. The results are of a 2ms simulation with plots displaying the voltage, power, and energy associated with the load resistor with the PART attribute of RL. The top waveform is the voltage at node Out. The middle waveform is the calculation of the power dissipated in the load resistor, RL, as specified by the function Power(RL). The bottom waveform is the calculation of the energy used by the load resistor, RL, through 2ms as specified by the function Energy(RL).

The power and energy function are simple to use. The R1 parameter that needs to be passed is simply the PART attribute of a valid two node component. Plotting the power and energy through the battery, V1, would be as simple as changing the plot expressions to Power(V1) and Energy(V1).

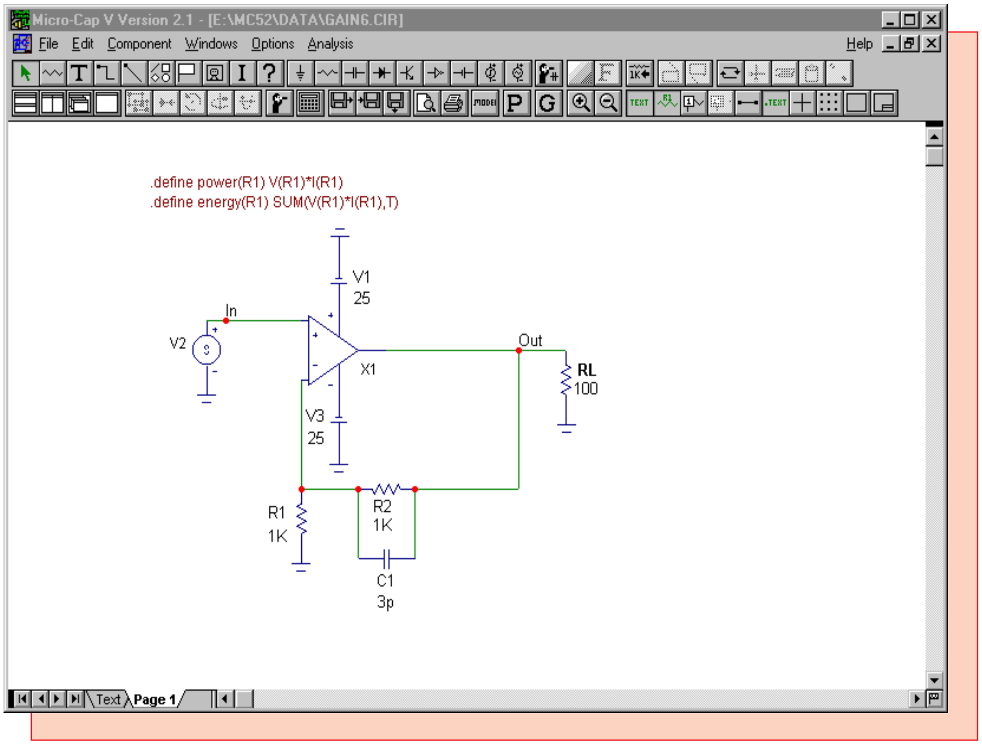


Fig. 15 - Power and Energy Operator Example Circuit

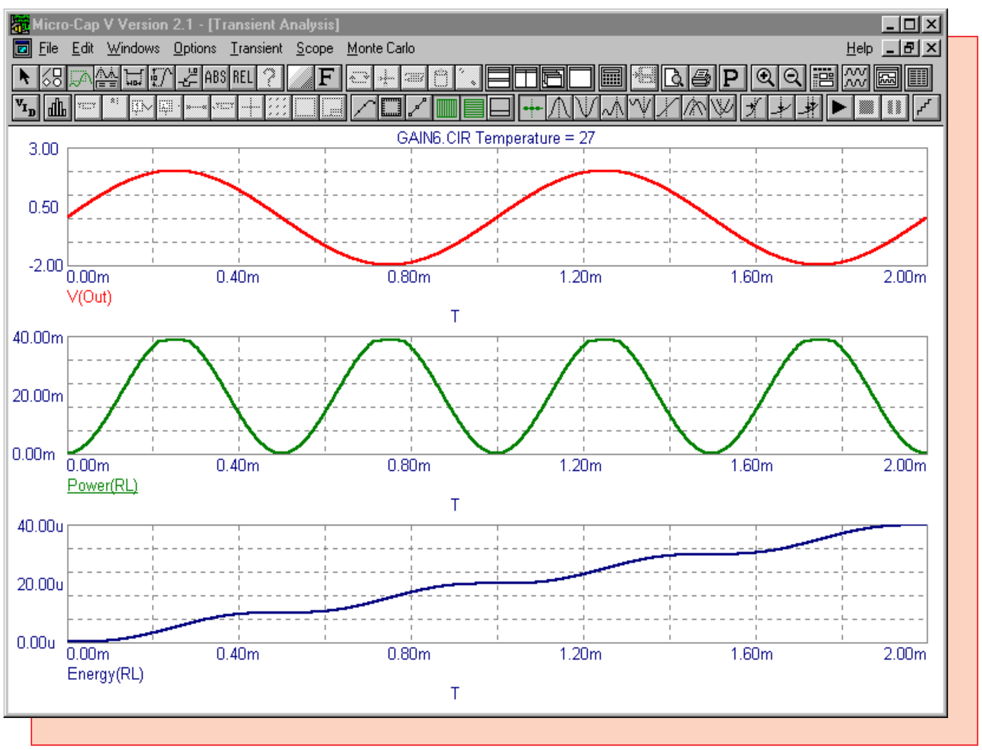


Fig. 16 - Power and Energy Analysis

Product Sheet

Latest Version numbers

Micro-Cap V .... Version 2.1
Micro-Cap IV IBM/NEC/MAC .... Version 3.04

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

Spectrum's Products

- Micro-Cap V $3495.00
- Micro-Cap V LAN (single seat) $3595.00
- Upgrade from MC5 Ver 2 to MC5 Ver 2 LAN $100.00
- Upgrade from MC5 Ver 1 to MC5 Ver 2 .... $500.00
- Upgrade from MC5 Ver 1 to MC5 Ver 2 LAN $600.00
- Upgrade from MC4 to MC5 .... $1000.00
- Upgrade from MC4 to MC5 LAN $1100.00

You may order by phone or mail using VISA, MASTERCARD, or American Express. Purchase orders accepted from recognized companies in the U.S. and Canada. California residents please add sales tax.