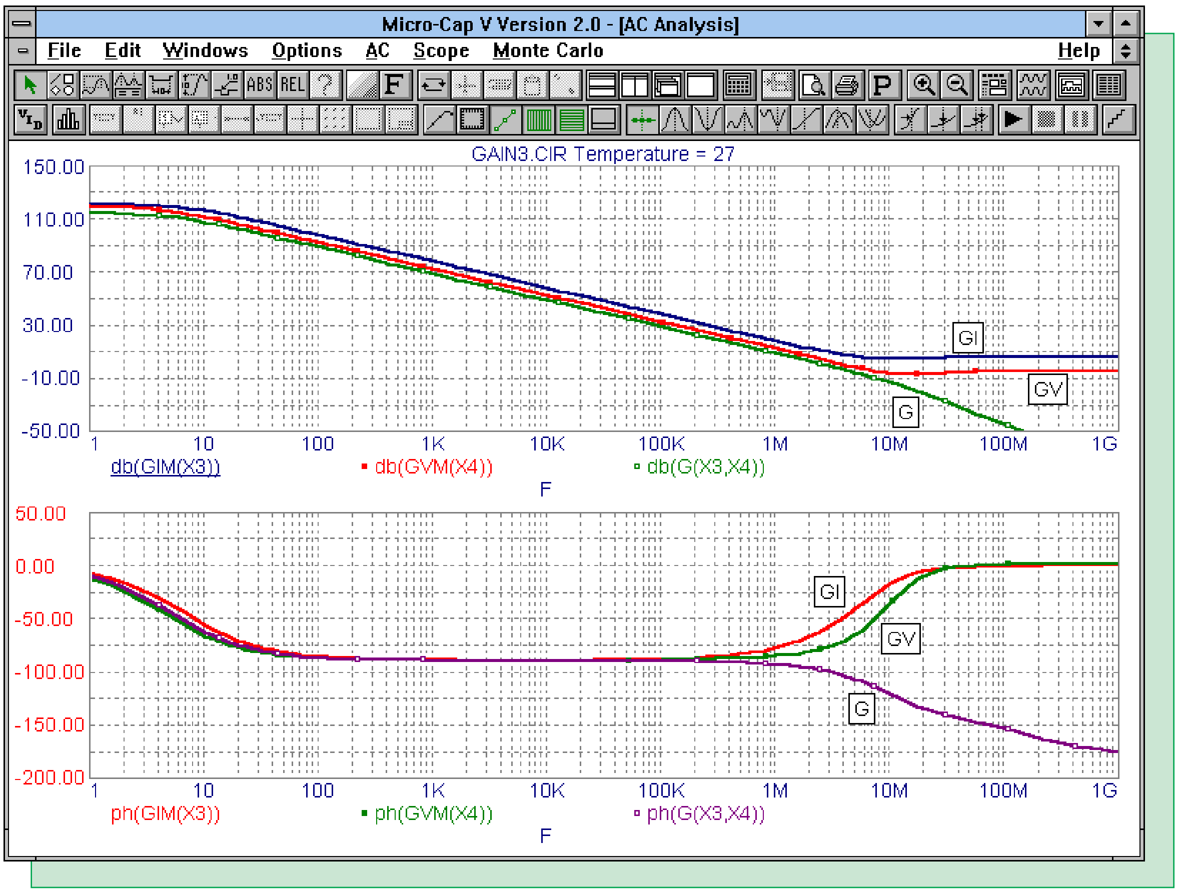
Applications for Micro-Cap™ Users

# Spring 1997

## Open Loop Gain Measurements




Featuring:

- Delay Macro
- Measuring Input Impedance
- Lossy Transmission Line Modeling
- Simulating Loop Gain

## News In Preview

With the introduction of Micro-Cap V Version 2, the Spectrum News will now use that version in all of the application notes. This issue uses some of the features now available in Version 2 such as the table operator and the user functions.

Included in this issue is a useful delay macro, a method for measuring input impedance, and the modeling of lossy transmission lines. Finally, this issue finishes with an article on simulating the open loop gain of a circuit without having to break its loop.

## Contents

## Book Recommendations

- *Electronic Devices and Circuits Using MICRO-CAP II*, Richard H. Berube, Merril 1992. ISBN# 0-02-309160-6
- *Electronic Devices and Circuits Using MICRO-CAP III*, Richard H. Berube, Merril 1993. ISBN# 0-02-309151-7
- *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

German

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

## Micro-Cap V Question and Answer

Caller1: I have just installed Micro-Cap V Version 2 and now when I try to run it, I get a security key not installed error. How do I fix this?

Tech: The security key drivers will not work unless the system is rebooted. For Windows 95 and Windows NT, you want to perform a shutdown and then restart the system. For Windows 3.1, you just need to exit Windows and then type in WIN to restart the Windows system. Micro-Cap V Version 1 needs to be rebooted only when installing under Windows NT.

Caller2: How do I tolerance a resistor for a Monte Carlo analysis?

Tech: The only parameters that may be toleranced are those that are within model statements. Each resistor that you want to tolerance must have a model name defined for its MODEL attribute. Each tolerance value that you want to use may be defined by one model statement each. For example, if you want to place a 5% tolerance on some resistors and a 10% tolerance on others, you would need a minimum of two model statements. The following two model statements could be used:

.model RMOD5 RES (R=1 LOT=5%)
.model RMOD10 RES (R=1 LOT=10%)

All resistors with a 5% tolerance could use RMOD5 as their model statement, and all resistors with a 10% tolerance could use RMOD10 as their model statement. The R parameter is a multiplier which will multiply the value of any resistor that references the model statement. For the RMOD10 model, any resistor that references the model will have its value multiplied by a value within the range .9 to 1.1 when a Monte Carlo analysis is performed.

The LOT statement above defines an absolute tolerance. The model statements above produce resistors with perfect tracking. This means that any resistors that reference the same model statement will use the same variation. For example, if two 10k resistors and one 20k resistor reference the RMOD10 model statement, and the R parameter is toleranced to a value of 1.05, all three resistors will use this value. Their resulting values would be 10.5K, 10.5K, and 21K.

To have each resistor toleranced individually without having to use multiple model statements, use the DEV statement as follows:

.model RMOD5 RES (R=1 DEV=5%)
.model RMOD10 RES (R=1 DEV=10%)

The DEV value specifies the relative percentage variation. Each resistor will have a private set of parameters that are toleranced individually. A combination of LOT and DEV statements may also be used.

This method is also applicable for capacitors and inductors.

## Delay Macro

The delay macro is designed to provide a simple time delay element in transient analysis or a phase shift in AC analysis. This macro is designed to be used with analog circuitry. There is a digital delay line primitive that offers the same function for digital circuits. This macro is preferred for analog circuits for two reasons: 1) the digital delay line would need I/O models if connected to analog components, which would use more components than the analog macro and force the output of the delay line to the analog voltage equivalent of digital states, and 2) the digital delay line has no AC equivalent whereas this macro will produce a phase shift. The delay macro appears in Figure 1 below.

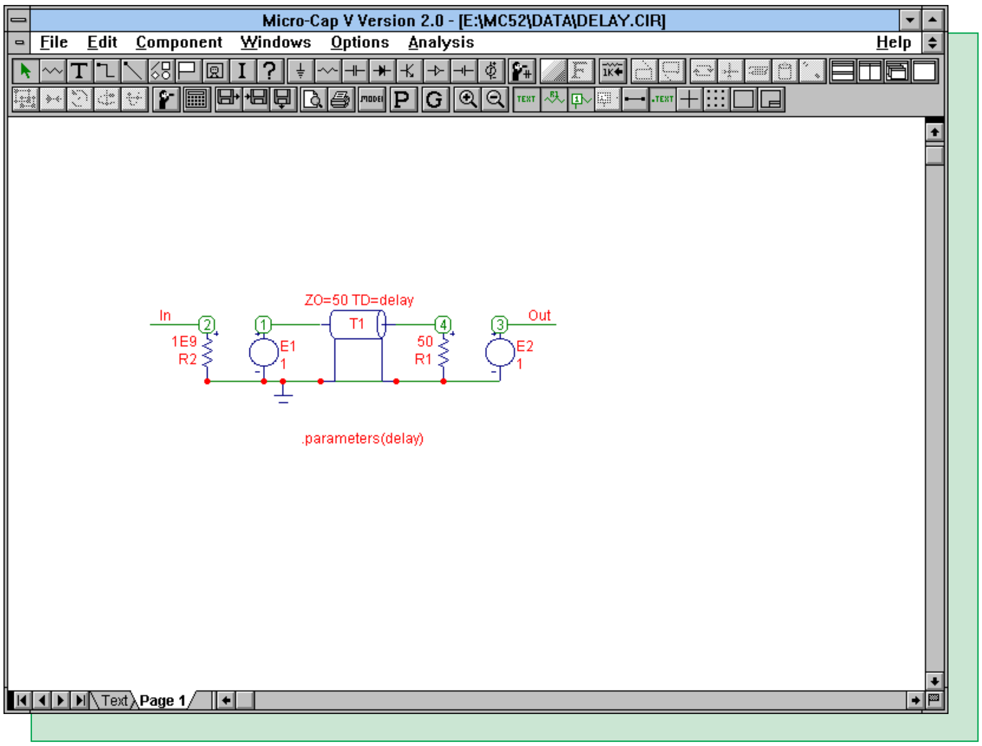



*Fig. 1 - Analog delay macro*

The single parameter in the macro is delay. This parameter defines the time delay that the macro will produce. The macro consists of two resistors, two VofV dependent sources, and an ideal transmission line. The two VofV dependent sources are used as buffering for the transmission line. The ideal transmission line is defined with a characteristic impedance of 50 ohms and a time delay that will be defined by the delay parameter. There is a 50 ohm termination resistor at the output of the transmission line to match the characteristic impedance of the line. This insures that the input waveform is passed cleanly through to the output without any reflections from the line. The 1E9 resistor is used to provide a DC path to ground at the input.

Figure 2 displays the settings in the Component Editor for the Delay macro. The component has been added to the Macros group in the Analog Primitives section. The name has been defined as Delay in order to match up to the macro circuit which is called Delay.cir. The shape used is Dlyline which is an existing shape in the shape library. The definition for the component is Macro. The pins have been defined as analog pins with the names In and Out. These pin names must match exactly with the node names in the macro circuit.

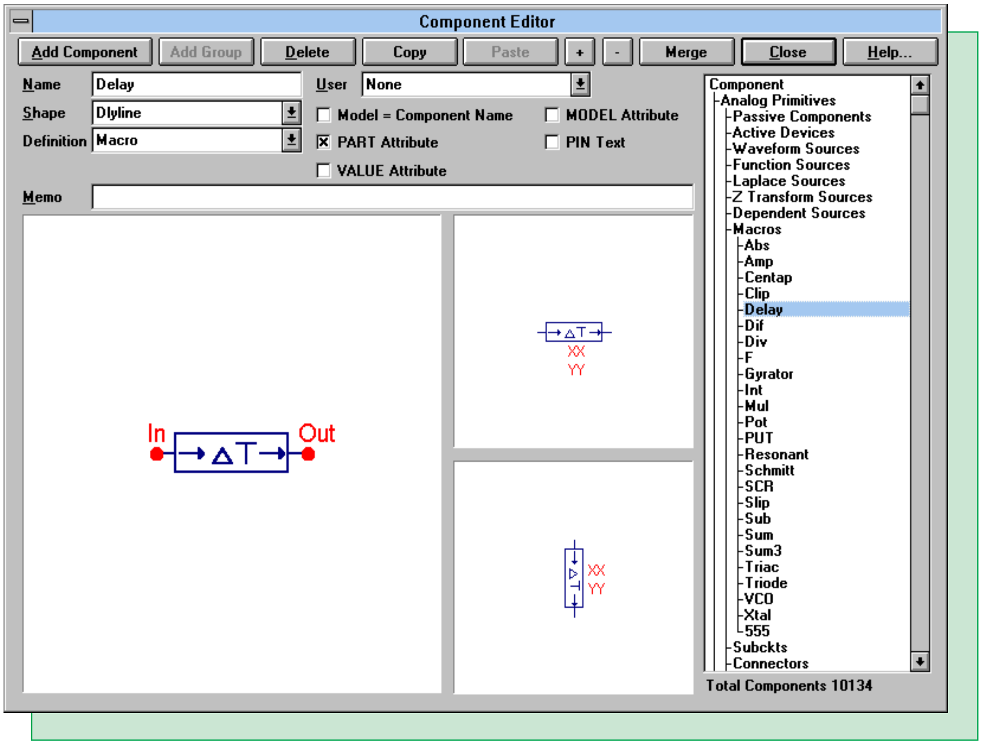



***Fig. 2 - Component Editor settings***

The test circuit for this macro is simply a 1Mhz sine wave going into the input of the Delay macro. The macro has been defined with the VALUE attribute of:

Delay(100n)

to provide a 100ns delay for the sine wave. Figure 3 is a plot of the input and output waveforms of the macro. The input waveform is passed cleanly through the macro with a 100ns shift at the output. Figure 4 is an AC analysis plot of the phase at the output of the macro for the same circuit. The phase values at 1MHz and 10MHz are shown in the cursor information. The 100ns delay is 10% of the period at 1MHz and 100% of the period at 10MHz. The macro produces the correct phase shifts of -36 degrees and -360 degrees at these frequencies.

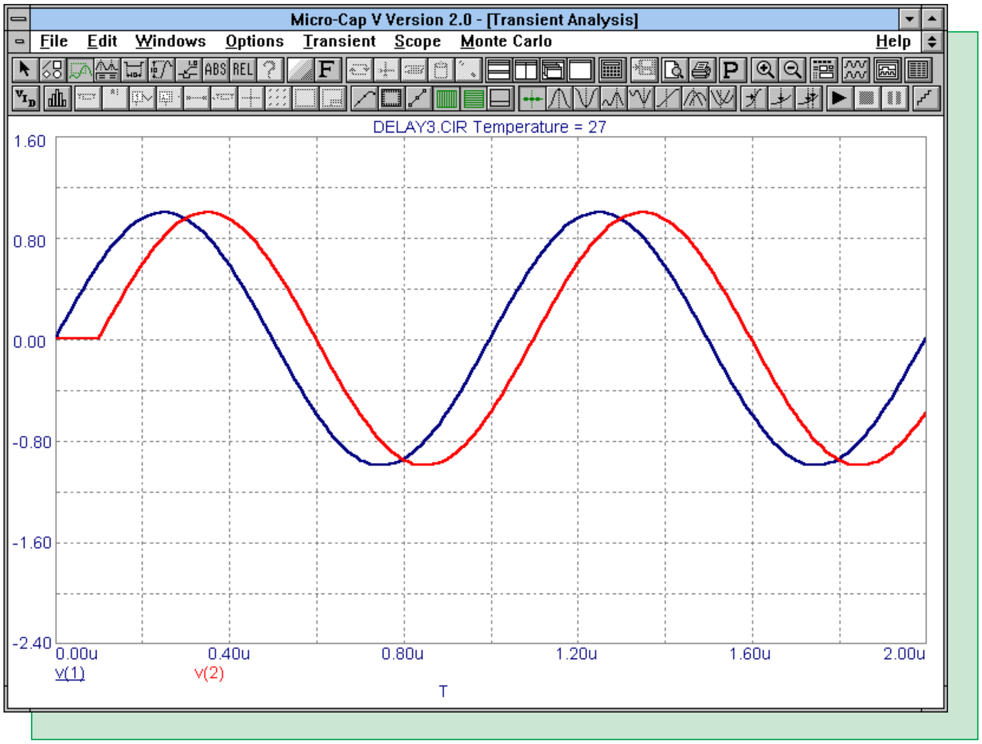



*Fig. 3 - Delay transient analysis results*

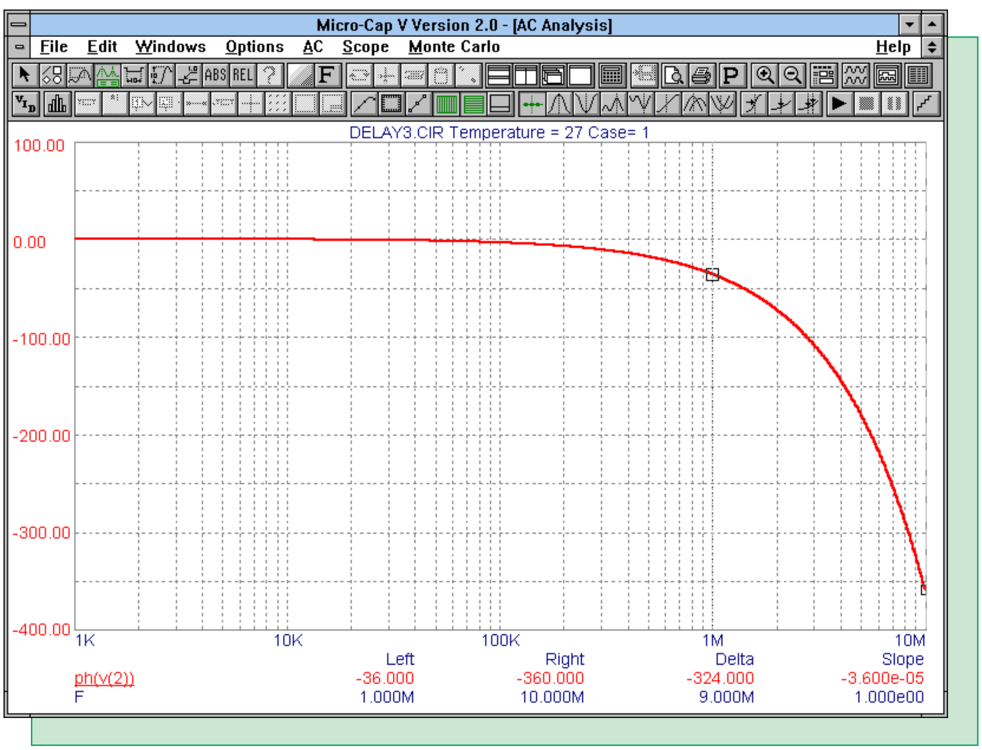



*Fig. 4 - Delay AC analysis results*

## Measuring Input Impedance

In Micro-Cap, there must be an AC source present in the schematic to run a meaningful AC analysis. This means that the circuit must contain one of the following: sine source, pulse source, user source, or the V or I SPICE independent sources. To measure input impedance, the I SPICE independent source will be used. This source is found in the Waveform Sources section of the Analog Primitives and is called 'I'. The current source is placed across the nodes where the impedance is to be measured. Figure 5 displays a test schematic that will measure the input impedance of a transistor circuit.

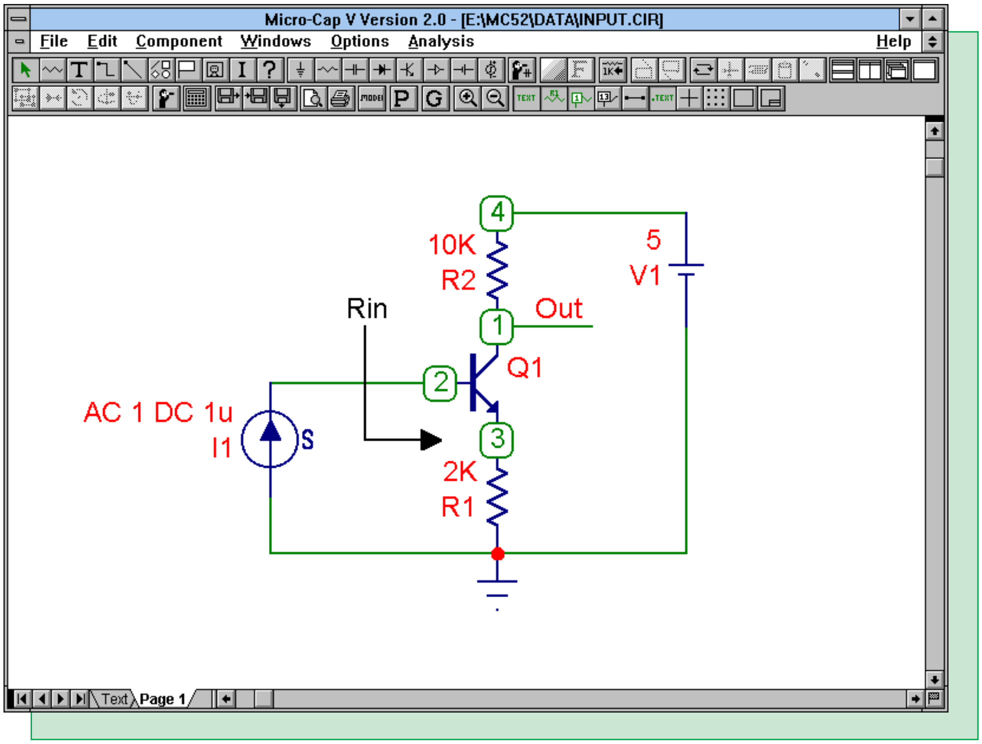



*Fig. 5 - Input impedance test circuit*

In this circuit, the current source is placed between ground and the base of the transistor. Rin marks the impedance that the source is looking into. The VALUE attribute of the current source has been defined as:

AC 1 DC 1u

This provides a DC bias of 1 microamp during the operating point calculation and a one amp small signal current during the AC analysis. The AC current is defined as one so that the impedance will be equivalent to the voltage across the current source. This is calculated from the equation:

V = I * Z

where if I is equal to 1, then:

V = Z

The input impedance of the circuit is the reflected impedance of the emitter multiplied by the beta of the transistor. It can be approximately represented by the following equation:

Rin = (Beta + 1) * Remitter

The value computed from this equation for the input impedance is 402 kohms. The AC analysis results for this circuit are displayed in Figure 6.

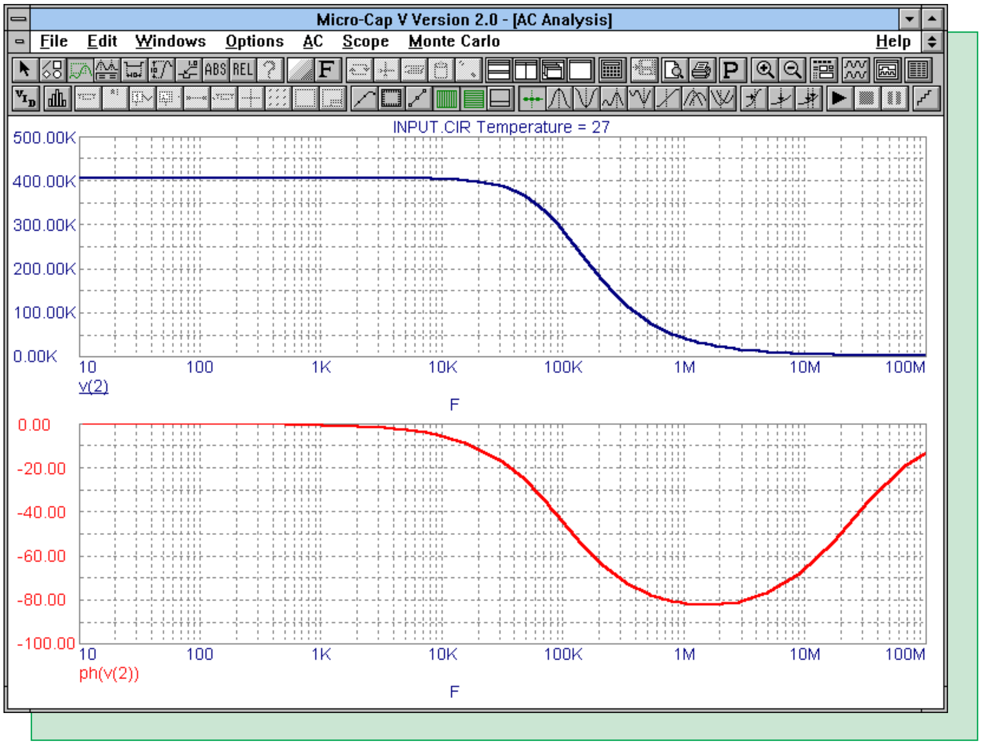



***Fig. 6 - Input impedance magnitude and phase***

Since the impedance is equivalent to the voltage across the source, the waveform V(2) has been chosen to plot the magnitude of the impedance. 2 is the node number at the base. The actual value of the impedance calculated in MC5 is 404.9 kohms which compares favorably with the calculated value of 402 kohms. The second plot is the phase of the impedance, and this waveform is plotted through the expression ph(V(2)).

## Lossy Transmission Line Modeling

MC5 uses the SPICE3 lossy transmission line model. This model can easily simulate RLC transmission lines if the R, L, and C parameters are known. However, transmission line information often appears only with nominal impedance, velocity of propagation, and attenuation at different frequencies. It is possible to create a subcircuit model using these parameters that will vary the attenuation of the transmission line according to a frequency parameter that is set by the user. This model is only applicable in transient analysis, as it is only valid for one frequency in AC analysis due to the fact that the resistance in the model is based on a fixed, user defined frequency.

The following table displays the specifications for an RG6A/U cable type.

| RG cable type | Velocity of prop., % | Nominal Impedance ohms | Attenuation (dB/100ft) frequency, MHz | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | 10 | 50 | 100 | 200 | 400 | 1000 | 3000 |
| 6A/U | 66.0 | 75.0 | 0.80 | 1.40 | 2.90 | 4.30 | 6.40 | 11.00 | --- |

The equations below convert the RG6A/U specifications into the appropriate R, L, and C parameters.

$C = 1/(V * Z_o)$
$L = Z_o^2 * C$
$R = atten * 2 * Z_o$

where V is the velocity of propagation, $Z_o$ is the nominal impedance, and atten is the attenuation at a fixed frequency. The atten value from the specification table needs to be divided by 8.686 to convert it into the needed units of nepers/ft. The value for the speed of light used in this instance is 9.82E8 ft/s. The subcircuit produced is shown below:

```
* RG-6A/U cable type: 75ohms, 66% velocity of propogation
* default 2.9dB/100ft @ 100MHz
.subckt RG6AU AP AN BP BN
+ params: freq=100MEG length=1
+ atten={table(freq, 10Meg,.008, 50Meg,.014, 100Meg,.029,
+       200Meg,.043, 400Meg,.064, 1G,.11)}

T1 AP AN BP BN RG6A

.model RG6A TRN (len={length} r={(atten/8.686)*2*75}
+             c=20.6p     l=115.6n)
.ends
```

The subcircuit model consists of a single lossy transmission line model. This subcircuit has two parameters: freq and length. The default frequency is 100MHz and the default length is 1 foot. The length value is used directly in the lossy transmission line's len parameter. The freq parameter is filtered through a table operator that contains data pairs of frequency versus attenuation. The table operator looks at the input, in this case the freq parameter, and will find the corresponding output from the data pairs. Information between specified data points will be interpolated. Note that the output values in the table statement are the attenuation values divided by 100 so that it will correspond to a length value of 1 foot. The value produced by the table operator is used to calculate the resistance of the transmission line. Note that the resistance is being divided by 8.686 in the model to produce the appropriate units of nepers/ft.

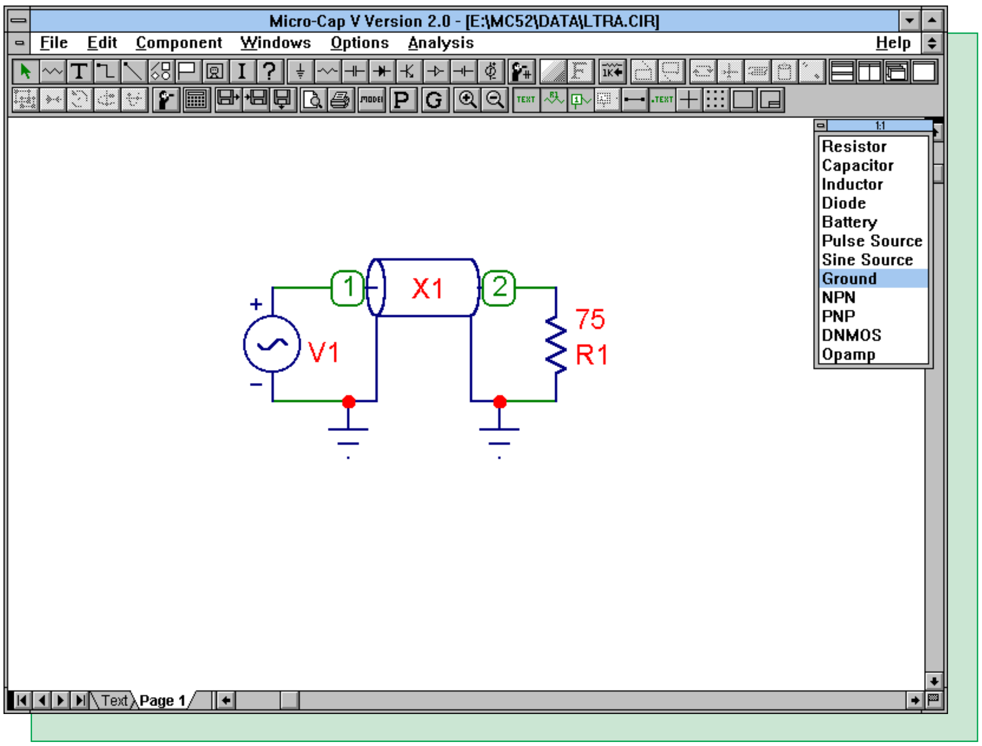



***Fig. 7 - Transmission line circuit***

The test circuit for the lossy transmission line model appears in Figure 7 above. A 10MHz sine wave is input into the RG6A/U transmission line with a matching load resistance of 75 ohms. The attributes for the transmission line subcircuit were defined as follows:

PART = X1
NAME = RG6AU
FILE =
PARAMS: = length = 100 freq = 10Meg
TEXT: =

The subcircuit has been defined as a transmission line of 100ft operating at 10MHz.

The transient analysis results for the circuit appear in Figure 8 below. The calculated delay for this transmission line was 154.3ns and was derived from the equation:

$d/v = t$

where d is the length of the line, v is the velocity of propagation, and t is the time delay. From the table, the attenuation at 10MHz should be 0.8. The simulation produces an attenuation value of 0.83 and a delay of 154.3ns.

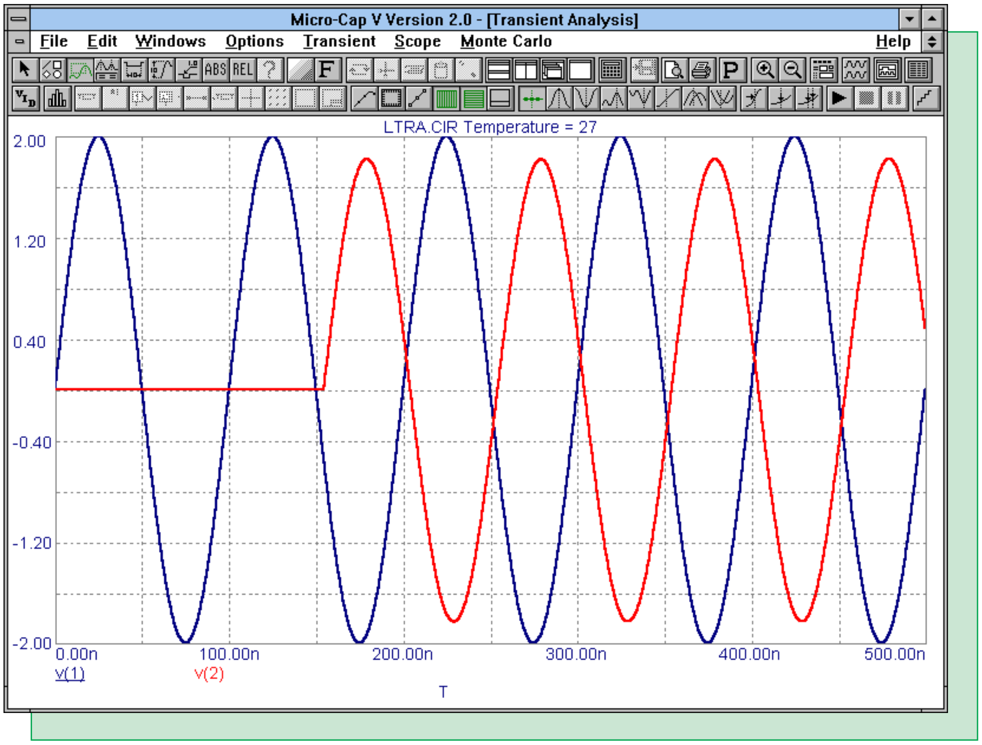



***Fig. 8 - Transmission line analysis***

## Simulating Loop Gain

The problem when trying to simulate loop gain is that in opening up the loop to make the proper measurements, the DC bias point of the circuit will be altered. Since the circuit is linearized around the DC bias point in AC analysis, this will throw off the results of the entire simulation. One technique can make these measurements without opening up the loop. This technique is taken from an article by Dr. R. D. Middlebrook which appears in the International Journal of Electronics, volume 38, number 4, 1975.

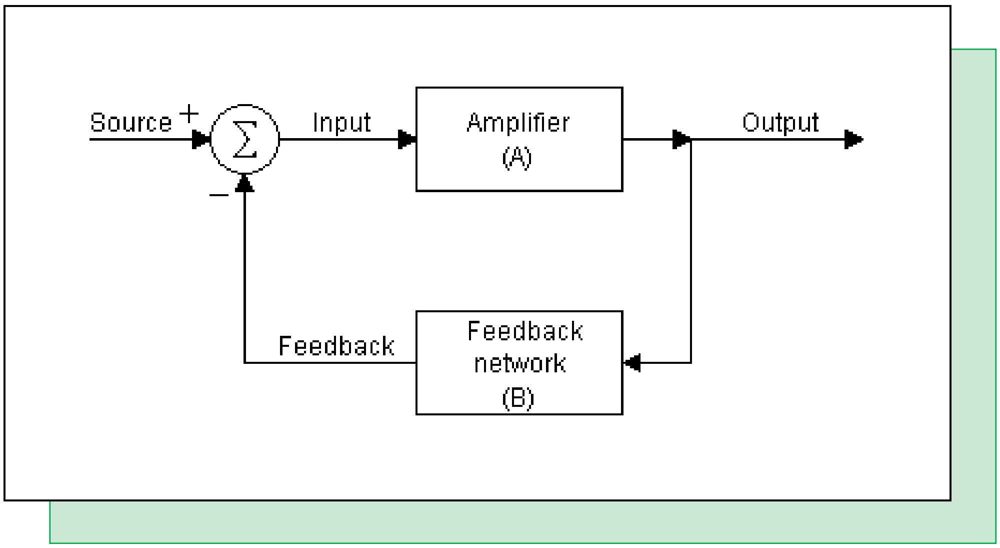



***Fig. 9 - Basic feedback amplifier***

Figure 9 displays the block diagram for a feedback amplifier. The gain may be calculated with either voltage or current. The current and voltage designators for the block diagram are as follows: source ($i_s$ and $v_s$), input ($i_i$ and $v_i$), feedback ($i_f$ and $v_f$), and output ($i_o$ and $v_o$). The open loop voltage gain and the open loop current gain are defined as:

$A_v = v_o/v_s$
$A_i = i_o/i_s$

The system voltage gain and system current gain are defined as:

$A_{vf} = A_v/(1+A_v*B)$
$A_{if} = A_i/(1+A_i*B)$

where B is the feedback transfer ratio. The quantity A*B is the loop gain. A positive loop gain implies that the feedback is negative. A negative loop gain would mean that the feedback is positive which may lead to oscillations in the circuit. The loop gain is also equivalent to the following:

$G_v = v_f/v_i$
$G_i = i_f/i_i$

Now it is not necessary to break the loop in order to measure these gains. The loop may be opened in its feedback path and the appropriate test signal injected. Injecting a current into the signal path will split the current into its feedback and input currents. This ratio can then be measured to produce a current loop gain. The voltage gain may also be measured through the same technique by placing a voltage source in the loop. The gain measurement setups are displayed in the figure below.

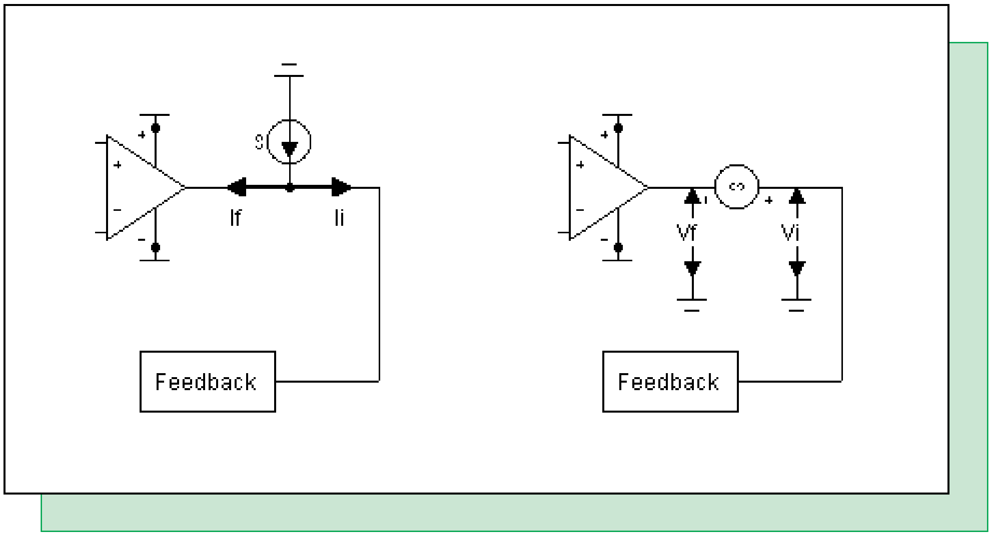



***Fig. 10 - The $G_i$ and $G_v$ setup***

For the actual measurements, macros will be created for the voltage and current injections in order to measure the voltages and currents with user functions. Both the voltage loop gain and the current loop gain must be taken into account to measure the total loop gain. They are related through the following equation:

$G + 1 = (G_v + 1) \| (G_i + 1)$

where

$x \| y = (x * y)/(x + y)$

The above equation may be reduced to:

$G = (G_i * G_v - 1)/(G_i + G_v + 2)$

As is normal in parallel calculations, the lower of the current or voltage gain will be the one that dominates. The feedback amplifier circuit that is to be analyzed is shown in Figure 11. The circuit is shown in its closed loop measurement configuration.

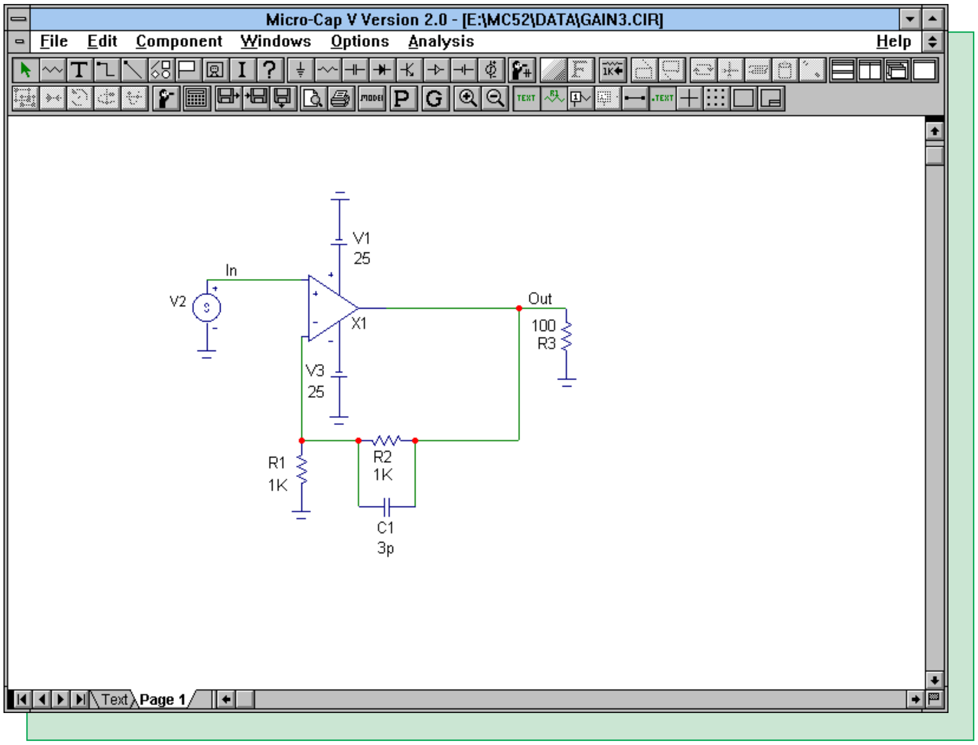



*Fig. 11 - Feedback amplifier circuit*

## Macros

Two macros need to be created for open loop measurement.  These macros will inject the test signal and provide the means for measuring the loop gain.  One macro will measure the open loop current gain and the other will measure the open loop voltage gain.

Figure 12 displays the GI macro which will measure the open loop current gain.  The macro consists of two batteries and the independent SPICE current source called I.  The current source is defined as 'AC 1' so that it will provide a 1A small signal current in the AC analysis.  The two batteries are used to measure the current in each direction.  They are given a voltage of 0 so that they don't affect simulation results.  The VII battery measures the $I_i$ current and the VIF battery measures the $I_f$ current.

Figure 13 displays the GV macro which will measure the open loop voltage gain.  The macro consists of two batteries and the independent SPICE voltage source called V.  The voltage source is defined as 'AC 1' so that it will provide a 1V small signal voltage in the AC analysis.  The two batteries provide fixed nodes (VIM and VFM) within the macro.  This is done so that the user functions may reference a fixed node in its equation because the input and output pins of the macro always change to what they are connected to which would make a global .define more complex.  The batteries are again given a voltage of 0 to not affect the simulation.

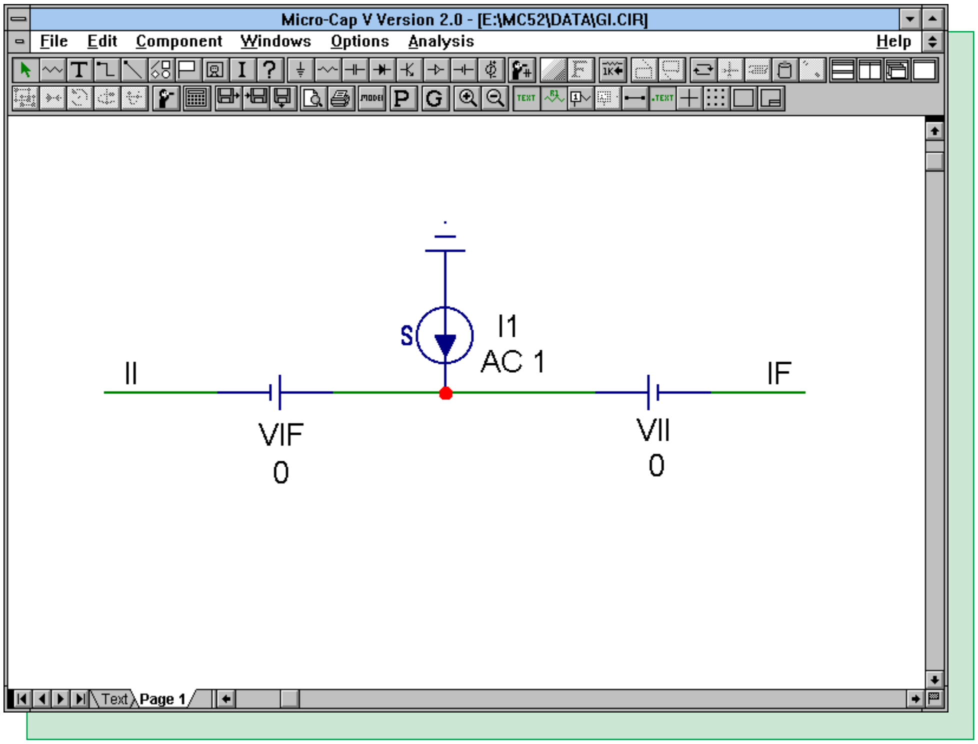



*Fig. 12 - The GI macro*

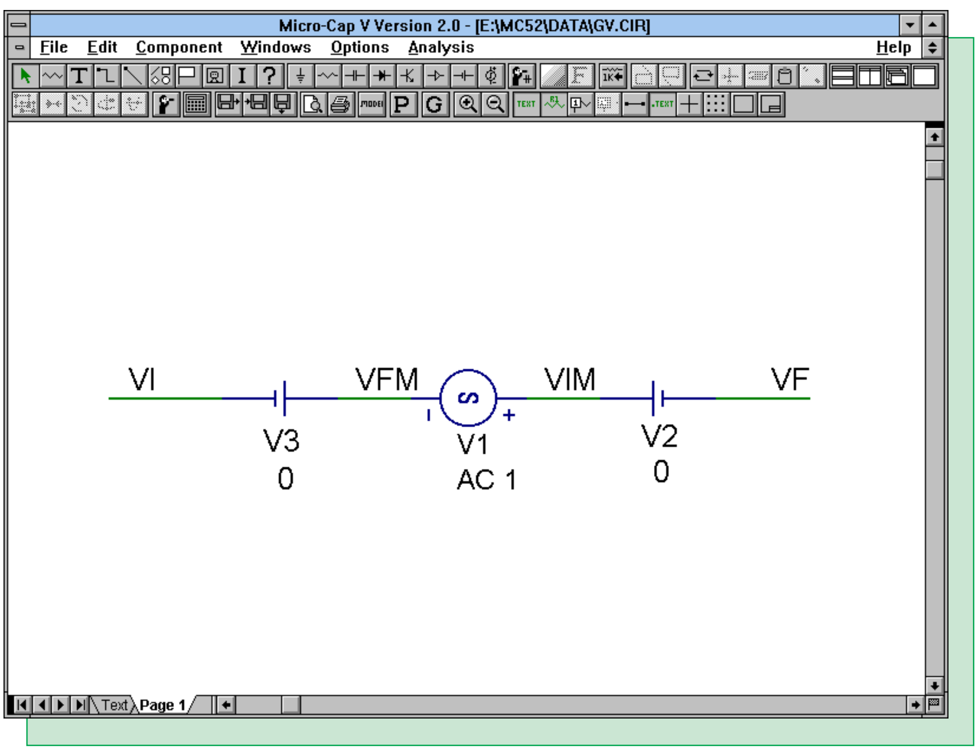



*Fig. 13 - The GV macro*

## User Functions

The user functions provide a means for users to create their own functions and operators by being able to pass parameters through a .define statement. The three user functions created for loop gain are shown below.

```
.define GIM(X) I(X.VIF)/I(X.VII)
.define GVM(X) V(X.VFM)/(-V(X.VIM))
.define G(X1,X2) (GIM(X1)*GVM(X2)-1)/(GIM(X1)+GVM(X2)+2)
```

GIM(X) defines the equation for the loop current gain. The parameter it passes is the PART attribute of the GI macro in the schematic. It divides the current in the VIF battery by the current in the VII battery. Both of these batteries exist inside the GI macro. GVM(X) defines the equation for the loop voltage gain. The parameter it passes is the PART attribute of the GV macro in the schematic. It divides the voltage of the VFM node by the negative voltage of the VIM node. Both of these nodes exist within the GV macro. G(X1,X2) defines the equation for the total open loop gain. It passes two parameters. X1 is the PART attribute for the GI macro in the schematic and X2 is the PART attribute for the GV macro in the schematic.

The user functions within a circuit are local to that one circuit. However, the user functions in the DEF.MC5 file are globally available to all circuits. This file is accessed from the User Definitions item on the Options menu.

## Analysis Example

The schematic that is to be analyzed is shown in Figure 14. This is actually two copies of the circuit from Figure 11. The AC source has been removed from the noninverting inputs and the GI and GV macros have been added to the feedback loops. The circuit on the left measures the open loop current gain and the circuit on the right measures the open loop voltage gain. The opamp that is being analyzed is the OP_27. The circuits need to be analyzed at the same time in order to produce the total gain as the total gain relies on both the open loop current gain and the open loop voltage gain.

The resulting analysis is shown in Figure 15. The top plot contains the open loop magnitude responses for the current gain (GI), the voltage gain (GV), and the total gain (G). The bottom plot contains the open loop phase responses for the current gain (GI), the voltage gain (GV), and the total gain (G). Note that the user functions have been used to plot these curves. The PART attribute for the GI macro in the schematic is X3, and the PART attribute for the GV macro in the schematic is X4.

Because the loop has never been broken, the operating point calculations are exactly the same between the closed loop circuit in Figure 11 and the open loop circuits in Figure 14.

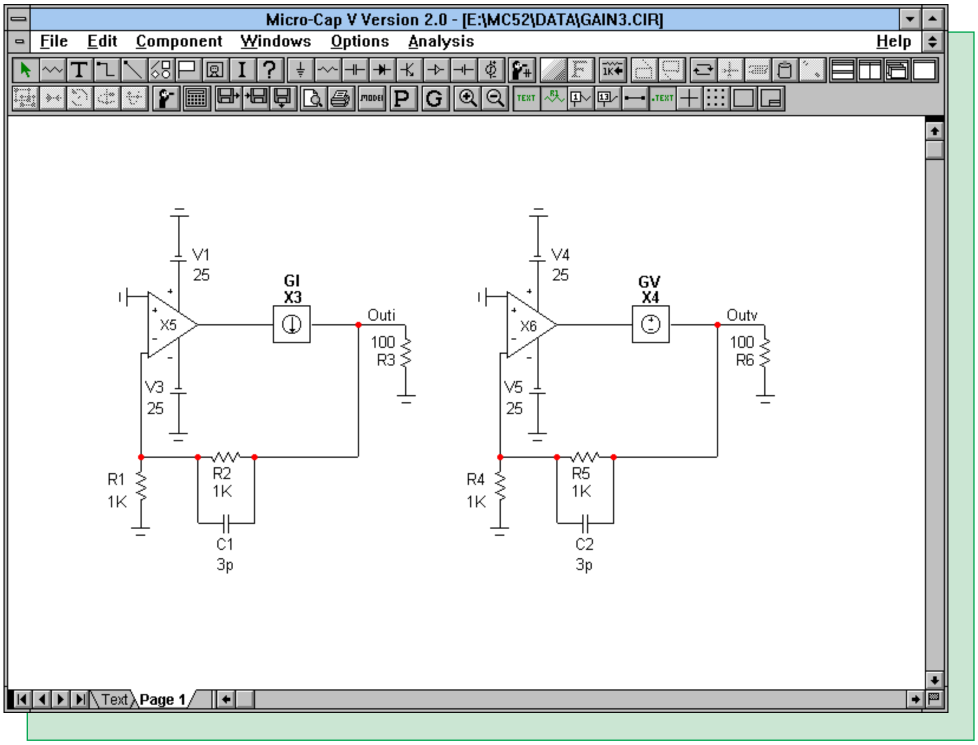



*Fig. 14 - Open loop gain circuit*

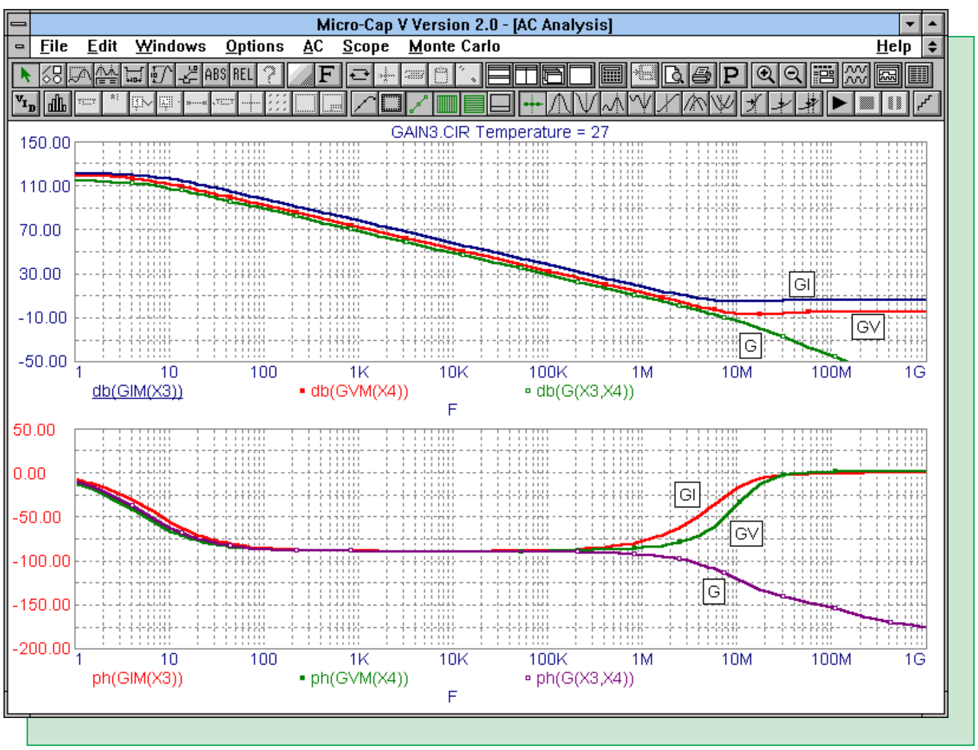



*Fig. 15 - Open loop gain AC analysis*

## Product Sheet

### Latest Version numbers

Micro-Cap V IBM/NEC .......................................... Version 2.0
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 IBM ................................................ $3495.00
- Micro-Cap V NEC ................................................. $3495.00
- Micro-Cap IV IBM ................................................ $2495.00
- Micro-Cap IV MAC ............................................. $2495.00
- Micro-Cap IV NEC ............................................... $2495.00

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