

Micro-Cap 8.0

Electronic Circuit Analysis Program User's Guide

Copyright 1982-2005 by Spectrum Software

1021 South Wolfe Road

Sunnyvale, CA 94086

Phone: (408) 738-4387

FAX: (408) 738-4702

Internet: www.spectrum-soft.com

Support: support@spectrum-soft.com

Sales: sales@spectrum-soft.com

Eighth Edition
Second Printing
February 2005

Copyright

© Spectrum Software. 1982-2005. This manual and the software described in it are copyrighted, with all rights reserved. No part of this publication, or the software, may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language in any form without the written permission of Spectrum Software.

Software license notice

Your license agreement with Spectrum Software, which is included with the product, describes the uses of the software which are permitted and the uses which are prohibited. Any unauthorized duplication of Micro-Cap 8, in whole or in part, in print, or in any other retrieval or storage system is expressly forbidden.

Contents

Chapter 1 - Before You Begin	19
Check your package	20
Check your equipment	21
Register your product	21
Let us know	21
Install the program	22
Chapter 2 - Exploring the Basics	23
What's in this chapter	23
How to start MC8	23
A quick tour	24
Command lines and batch files	35
Terms and concepts	37
Function keys	40
The Undo and Redo functions	41
Chapter 3 - Creating and Editing Simple Circuits	43
What's in this chapter	43
Creating a simple circuit	44
Editing component parameters and text	52
Deleting objects	55
Undoing and redoing operations	55
The clipboard	55
Selection	57
Drag copying	58
Navigating large schematics	59
Creating and editing SPICE text files	62
Summary	64
Chapter 4 - Transient Analysis	65
What's in this chapter	65
The Transient Analysis Limits dialog box	66
Selecting curves to plot or print	71

The Plot Properties dialog box	75
State variables and initialization	78
The State Variables editor	80
Chapter 5 - AC Analysis	83
What's in this chapter	83
What happens in AC analysis	84
The AC Analysis Limits dialog box	86
Experimenting with options and limits	91
Numeric output	93
Noise plots	94
Nyquist plots	96
Smith charts	97
Polar plots	98
Chapter 6 - DC Analysis	99
What's in this chapter	99
What happens in DC analysis	100
The DC Analysis Limits dialog box	102
Chapter 7 - Using Scope	107
What's in this chapter	107
Experimenting with Scope	108
Magnifying	114
Panning	115
Cursor mode	116
Cursor mode panning and scaling	117
Cursor mode positioning	118
Adding text to plots	123
Adding tags to plots	124
Performance functions	126
Animating an analysis	130
Chapter 8 - Using Probe	131
What's in this chapter	131
How Probe works	132
Transient analysis variables	134
Probing node voltages	136

Probing lead-to-lead voltages	137
Navigating the schematic	138
An AC example	139
A DC example	141
Probing a SPICE file	143
Tips for easy probing:	144
Chapter 9 - Stepping Component Parameters	145
What's in this chapter	145
How parameter stepping works	146
Stepping in transient analysis	148
AC and DC examples	150
Text stepping	152
Stepping summary	154
Chapter 10 - Using Monte Carlo	157
What's in this chapter	157
How Monte Carlo works	158
Distributions	160
Performance functions	161
Options	162
An example	163
Statistical summary	170
Chapter 11 - Working with Macros	171
What's in this chapter	171
What is a macro?	171
Creating the macro circuit file	172
Selecting a suitable shape for the macro	173
Entering the macro into the Component library	174
Using the macro in a circuit	176
An easy way to create macros	178
Chapter 12 - Working with Subcircuits	181
What's in this chapter	181
What is a subcircuit?	181
Creating the subcircuit text file	182
Selecting a suitable shape for the subcircuit	183

Putting subcircuits into the Component library	184
Using the subcircuit as a component	186
Using the Add Part wizard to add subckt parts	187
Chapter 13 - Printing	189
What's in this chapter	189
Printing schematics	190
An example of schematic printing	192
Printing an analysis plot	194
Chapter 14 - Using Animation Mode	199
What's in this chapter	199
How Animation mode works	200
Animation components	201
Animate Options dialog box	203
A transient analysis example	204
A Dynamic DC example	206
Index	207

Spectrum Software and Micro-Cap

Spectrum Software was founded in February of 1980 by Andy Thompson to provide software for personal computers. Initially, the company concentrated on providing software for Apple II systems.

One of the earliest products was Logic Designer and Simulator. Released in June 1980, this product was the first integrated circuit editor and logic simulation system available for personal computers. In many ways it was the forerunner of the Micro-Cap products. Its primary goal was to provide a “circuit creation and simulation” environment for digital simulation.

In August of 1981, the analog equivalent of the first program, Circuit Designer and Simulator, was released. Its integrated text editor created circuit descriptions for a simple, linear, analog simulator.

September of 1982 saw the release of the first Micro-Cap package. The first program to offer a “sketch and simulate” motif, it provided a schematic editor which created netlists for the companion simulator to analyze. It also included an improved nonlinear simulator, based upon the modified nodal circuit formulation and backward Euler integration techniques. One of its key features was the ability to plot the results dynamically during the run. Because you could see what was happening, you could often terminate the run long before it was finished, saving valuable time. Both Apple II and IBM versions of the product were released.

In November of 1984, Spectrum released the Micro-Cap II package. It greatly improved upon the earlier schematic editor and included enhanced models. Versions 2.0, 3.0, and 4.0 were introduced in 1985, 1986, and 1987 respectively, and included support for higher resolution displays, coprocessors, and plotters. Both Macintosh and IBM versions of the product were released.

Micro-Cap III was released in December of 1988. Built around the use of the mouse and windows, it provided a very easy-to-learn environment. Its simulator, while not 100% SPICE compatible, was based very closely on the UC Berkeley SPICE 2G model. Standard device models included the Gummel-Poon bipolar transistor model, JFET model, and the Level 1 and Level 3 MOSFET models. Subsequent revisions, have converted it to the C language and added numerous improvements on and extensions to the SPICE 2G models. These include models for analog behavioral sources (Laplace and nonlinear function), nonlinear magnetic cores, GaAsFETS, and OPAMPs. It included many other advanced features such as Monte Carlo analysis and parameter stepping.

In February of 1992, Spectrum introduced Micro-Cap IV. In addition to a much improved schematic editor, Micro-Cap IV included a simulator that could read and analyze SPICE 2G text files directly. The schematic editor sported many new refinements, including a clipboard. The DOS version of the simulator was faster than any DOS-based simulator and the extended DOS version was even faster. The latter used up to 16 Megabytes of extended RAM and handled circuits as large as ten thousand transistors. The core simulator was identical to SPICE 2G, with enhancements that derived from SPICE 3 and Spectrum's own developments. For those who required device models not in the large part library, an optimizing model generator was included that made model creation from data sheets easy and accurate. Standard device models included all SPICE 2G models, analog behavioral sources (Laplace and nonlinear function), nonlinear magnetic cores, GaAsFETs and OPAMPs. The Probe feature let users probe the schematic with the mouse to display waveforms in a 'scope'. The use of mathematical expressions provided great flexibility in plotting and defining component models. Expressions included all of the usual arithmetic, transcendental, hyperbolic, Boolean, and relational operators as well as some special ones like forward and inverse Fourier transform, correlation, auto-correlation, coherence, numerical integration and differentiation.

In August of 1995, Spectrum introduced the first Windows version, Micro-Cap V. Micro-Cap V included all of the features of Micro-Cap IV and added an integrated PSpice® compatible 5-state event-driven digital logic simulator, time synchronized with the internal analog simulation engine. The user interface was similar to that of Micro-Cap IV, but updated to conform to the Windows interface standard. Additional component models were added, including lossy transmission lines, and voltage and current-controlled switches. Schematic structure was enhanced with the addition of a separate text area to hold the text needed in a simulation file. Schematics were generalized to multi-page documents using inter-page ties for connection. New editing commands included region mirroring, rotating, and flipping about the X and Y axes. Components acquired additional attributes, each with independently movable text. Lines became wires with the new property of connecting only at their endpoints, an essential feature for the dense interconnect of logic schematics. Analysis plots acquired graphical objects, text, and tags for labelling individual data points and differences between data points.

In June of 1997 Micro-Cap V 2.0 was released. It included full MOSFET BSIM models, 3D plotting, performance function plotting, multidimensional parameter stepping, symbolic stepping, and animated displays. Animated displays included blinking seven-segment displays, LEDs, node states, and switches.

New Features Introduced in Micro-Cap 6

In July of 1999 Micro-Cap 6 was released. Its principal new features were:

- Active and passive filter designer
- PCB netlist interface
- New BSIM3 3.2 MOSFET model
- Dynamic DC operating point analysis mode
- Sensitivity analysis
- Transfer function analysis
- Temperature list for all analyses
- List, log, and linear sweep options for DC analysis
- Automatic macros
- Network version
- Enhanced user waveforms
- Gmin stepping
- Slope, X_Low, X_High, Y_Low, and Y_High performance functions
- Probe macro pin selector
- New power variables
- Complex Bessel functions, series, and factorials
- Symbolic parameter tolerances
- A plot polygon region for defining desired design areas
- Extended DC analysis

New Features Introduced in Micro-Cap 7

In September of 2001, Micro-Cap 7 was released. Its major features were:

- Optimizer
- RF models and Smith charts
- Portable schematic file format.
- User-specified paths (folders) for all major file groups
- Characteristic curves in the Attribute dialog box
- Import and Add Part wizards in the Component editor
- Attribute dialog box editor
- Multistage undo and redo
- Improved component find command
- Text stepping
- Symbolic derivative finder
- Status bar monitor
- Waveform branch and data point labeling
- Smooth schematic panning
- Bill of materials
- Sanity checker
- SPICE file probing
- Live formulas
- New X and Y Scale format for constant grid values
- Even decimal values cursor positioning
- Thumbnail plot
- Incremental auto-ranging
- Simultaneous multi-row analysis limits edits
- Run-invariant expressions in model parameters
- Complex trigonometric functions
- Complex conditional functions
- Expressions for histograms and performance plots
- Attribute search and replace
- Multiple file opener
- PADS PCB output

What's New in Micro-Cap 8

In February of 2004, Micro-Cap 8 was released. Its major features were:

IBIS modeling tool

New IBIS modeling tools were added to translate IBIS files to standard Micro-Cap / SPICE input and output pin models. These models accurately reproduced the IBIS golden waveforms. The IBIS editor created both standard library SPICE models and verification models for testing and verifying golden waveforms.

EKV MOSFET model

The latest EKV MOSFET model from the Swiss Institute of Technology was added. It featured a dedicated charge-based dynamic model using only 18 intrinsic core parameters.

BSIM4 MOSFET model

The BSIM4 MOSFET model was added to address MOSFET physical effects into the sub-100nm realm.

HSPICE style binning

BSIM3 and BSIM4 devices were upgraded to include the HSPICE binning option, as well as the standard Berkeley binning option.

Short distance VT matching for BSIM3 and BSIM4 models

Both BSIM3 and BSIM4 were upgraded to include a short distance VT matching parameter, to better model short distance threshold variations.

N-Port device using Touchstone data format

A new N-Port device was added that used S, Y, Z, G, or H parameters and read standard Touchstone data files.

SPICE voltage and current source enhancements

New time-domain Noise and Gaussian waveforms and preset tabs for common waveforms were added to make complex waveform creation easy.

Timer device

A flexible new timer device was added to count events, and to measure both elapsed and last event time.

Enhanced S and W switches

A hysteresis option was added to the S and W switches.

Improved magnetics model

A new, more robust Jiles-Atherton magnetics model was added. Both MODEL and the mainline simulator could plot BH values in both CGS and MKS (SI) units.

Animation components

- SPST, SPDT, DPST analog switches for dynamically configuring a circuit.
- Analog bar with height proportional to the input DC voltage.
- Analog LED with user-defined color and on voltage.
- Rotating DC motor with DC voltage dependent RPM.
- Analog / digital DC voltage/current meter.
- Relay with programmable resistance, inductance, and on and off currents.
- Three color traffic light.

Password protection

A new protection feature added password protection to circuit and macro files.

Find in Files command

A new Find in Files command was added to search circuit or other files for text, components, or other content.

Info Page

An Info text page was added to show the location of macros/subcircuits used in the circuit, resolving uncertainty over which model file was used.

Component menu shape display

The Component menu could now show the component shape as you browse.

Local path commands

New local .PATH commands let a circuit specify DATA and LIBRARY paths, overriding the global paths.

Subcircuit maker

A new Make SUBCKT command translated a schematic to a SPICE subcircuit.

Improved Bill of Materials report

A new Bill of Materials report featured sorting on any field and a live report that responded to user format changes immediately.

Apply Display Properties command

A new command was added to copy a selected part's display properties (part color, attribute font, size, color, and print location) to other similarly named and rotated parts.

Change Attribute command

A new Change Attribute command was added to add, delete, edit, show, or hide attributes of any subset of the parts in a schematic, making mass changes easy.

New graphics formats

TIFF, PNG, JPG, and ICO graphics formats were added to the standard BMP, WMF, and EMF formats for saving schematic and plot images.

Automatic node highlight

Node highlighting was added.

Cleanup command

A new Cleanup command for deleting miscellaneous extraneous files was added.

New Attribute dialog box plot

Diodes acquired a new I_r vs. V_r plot to show reverse characteristics.

New color options

Parts and wires could now have individual colors.

New Dynamic AC analysis

A new analysis mode was added that featured dynamic on-schematic display of AC voltage, current, and power as frequency is stepped, with the display dynamically responding to schematic edits.

Enhanced Dynamic DC analysis

Stepping, optimization, and an Analysis Limits dialog box were added to Dynamic DC.

New Distortion analysis

A new analysis mode was added that employed the DSP routines to calculate and plot measures of distortion, including raw harmonics, as well as total and individual harmonic distortion.

Enhanced Fourier analysis

A new FFT window was added to automatically convert transient plots to Fourier plots. A new FS(expr,n1,n2) function was added that created a waveform from the n1 to n2 terms of the Fourier series of (expr). A new RES(expr,n1,n2) function was added to create a residual waveform by subtracting the n1 to n2 terms of the original waveform. FFT auto-scale commands were upgraded to allow ignoring the DC component and auto scaling over only selected harmonics.

Analysis plot improvements

- Scale factors (Meg, K,...p, f) for X/Y axes. e.g. "Time (nS)"
- Units (Amps, Volts, Watts,...) for plots. e.g. "V(26) (Volts)"
- Open and closed circles and squares data point markers
- Normal and popsicle data point styles
- X, Y, or both autoscaling options
- New option to force all X scales to be the same
- Retrace mode for rolling displays
- Right clicking on expressions to invoke FFT, performance, and 3D plots
- New minor log grid option to show and label 1-2-5-10 sequence

New expressions and functions

- Integration and differentiation operators for use in function sources
- Array variables for indexed selection of variable values from a predefined array of values
- Transmission line power and energy terms
- AC power and energy terms
- New random functions, RNDR, RNDC, and RNDI(*interval*)
- New DELAY(x,d) function returns expression x delayed by d seconds.
- S and W switch resistance, voltage, current, power, and energy terms
- SPICE3 analog Boolean operators, &, |, ~, and ^ with analog parameters
- LAST(expr,n) function
- MAXR and MINR function to track an expression's highest and lowest value during a run
- NORM, NORMMAX and NORMMIN functions to normalize at specified, maximum, and minimum points
- International engineering notation (e.g. 3R3 = 3.3)
- Global settings variables
- Model parameter use of variables like TEMP that are constant during a run
- Nested series expressions
- Product series function (e.g. PROD(I,1,3,I) = 1*2*3)

Analysis limits dialog box

The dialog box size and X and Y fields were made user-adjustable.

Optimizer enhancements

The new optimizer supported dynamic plots to show optimization progress and worked with both Dynamic DC and Dynamic AC.

MODEL import

MODEL was upgraded to allow simple one-step exporting of its parts directly to the Component library.

Numeric formats

A new decimal numeric format was added that used only numbers, commas, and decimal points. 1.234e6 could be formatted as 1,234,000.000. Support was added for international engineering notation (e.g. 3R3 = 3.3).

Filter Designer improvements

Step and Impulse response plots were added.

Monte Carlo improvements

A new feature let users reproduce statistical distributions by specifying the random number seed. New features included user-specified histogram range and intervals.

Expanded LAN operation

Several features were added to better support LAN installations. The model library index was now written locally, avoiding the need to write to possibly write-protected LAN disks. Component, Shape, and Package library files could also be located on write-protected LAN disks.

Miscellaneous

- New .WARNING message condition command to let users create custom warnings such as .WARNING "Capacitor voltage exceeded." V(C1) > 50
- SPACEBAR toggle between voltage and current in Probe.
- .DEFINE statement for use in SPICE files.
- All On and All Off buttons to enable/disable all stepped variables.
- Rename command to provide left to right and top to bottom options.
- Translator to convert file from S, Y, Z, G, or H format to any other format.
- New numeric output option to let the user decide what to include.

About the User's Guide

This guide helps you get started. It shows you how to install the product and describes the main features of the program. Tutorials are used throughout, first to teach basic concepts and later, to develop skill in using the features of the program. The chapters cover these topics:

- Chapter 1, "Before You Begin", shows you how to check your equipment, make backup copies of the disks, and install the product.
- Chapter 2, "Exploring the Basics", provides a quick tour of the product, followed by an introduction to the basic Micro-Cap user interface.
- Chapter 3, "Creating and Editing Simple Circuits", illustrates techniques for circuit creation and editing.
- Chapter 4, "Transient Analysis", describes the time domain simulator.
- Chapter 5, "AC Analysis", describes AC analysis.
- Chapter 6, "DC Analysis", describes the nonlinear DC simulator.
- Chapter 7, "Using the Scope", shows how to use Scope to analyze plots.
- Chapter 8, "Using Probe", illustrates Probe's 'point and shoot' tools for interactive display of simulation results.
- Chapter 9, "Stepping Component Parameters", shows how to use parameter stepping to see the effect of changing the numeric model parameters.
- Chapter 10, "Using Monte Carlo", describes how to use the Monte Carlo function to analyze the statistical performance of a circuit.
- Chapter 11, "Working with Macros", describes the use of macros.
- Chapter 12, "Working with Subcircuits", describes the use of subcircuits.
- Chapter 13, "Printing and Plotting", describes the options for printing and plotting the various text and graphic displays produced by Micro-Cap.
- Chapter 14, "Using Animation Mode", describes the animation methods.

Typographic conventions

Certain typographic conventions are employed to simplify reading and using the manuals. Here are the guidelines:

1. Named keys are denoted by the key name alone. For example:

Press HOME, then press ENTER.

2. Text that is to be typed by the user is denoted by the text enclosed in double quotes. For example:

Type in the name "TTLINV".

3. Combinations of two keys are shown with the key symbols separated by a plus sign. For example:

ALT + R

4. Option selection is shown hierarchically. For example this phrase:

Options / Preferences / Options / File Warning

means the File Warning item from the Options section of the Preferences dialog box, which is chosen from the Options menu.

5. User entries are shown in italics. For example:

emitter_lead

6. The [and] characters bracket optional user entries. For example:

[*low*]

7. The < and > characters bracket required user entries. For example:

<*emitter_lead*>

8. The OR symbol (|) designates mutually exclusive alternatives. For example, PUL | EXP | SIN means PUL or EXP or SIN.

Welcome to Micro-Cap 8, a mixed analog-digital circuit analysis program for personal computers. Micro-Cap 8 is the eighth generation of a program that began in 1982 with the release of Micro-Cap. Since then it has acquired many refinements, always adhering to its primary goal of providing an easy to use, sketch and simulate environment. Micro-Cap 8 continues that tradition by providing:

- A modern, Windows-based, easy to learn user interface
- A powerful schematic editor
- A fast PSpice™ compatible analog-digital simulator
- An optimizer for fine-tuning circuit values
- Schematic waveform probing
- Fast, accurate device model creation from data sheets
- A large device library with more than 16,000 analog and digital parts
- Analog and digital behavioral modeling features
- Eight MOSFET models, including BSIM3, BSIM4, and EKV
- IBIS modeling
- Powerful performance measurement and plotting functions
- A netlist interface to popular PCB packages
- Integrated active and passive filter design function
- Animated analysis display
- Dynamic on-schematic display of DC and AC voltage, current, and power.

Micro-Cap 8 lets you describe your circuits in the way most natural for you. You can sketch a schematic or type in a SPICE text description. Either way, Micro-Cap 8 quickly analyzes the circuit and plots the results dynamically during the run.

This chapter tells you what you need to do before running the program.

Check your package

Your Micro-Cap 8 package includes the following items:

- A single CD ROM
- A security key (if not upgrading from an earlier version)
- A password sheet (if upgrading from an earlier version)
- Two instruction manuals
- A registration card

The instruction manuals include a User's Guide and a Reference Manual.

- The User's Guide is designed to get you started quickly. It shows you how to install the program and how to begin using it. The manual provides a brief introduction to most of the product's features and a series of graduated tutorials that introduce you to the main features and teach you how to use them.
- The Reference Manual provides a detailed description of the capabilities of every feature of Micro-Cap 8. It describes and illustrates some of the more advanced uses of those features.

The security device plugs into any of the USB, or parallel printer ports on your computer. The program will check all ports for the existence of the device. The device need not be present to install the program, but it must be present whenever any of the Micro-Cap 8 programs are running. The device does not interfere with other printers or devices that may be attached to the port.

If you are upgrading from an earlier version of Micro-Cap, your package includes a sheet with a password generated from your Key ID. When you run Micro-Cap 8 for the first time, it will ask you for the password. Type it in and the program will permanently upgrade your key so that it can run Micro-Cap 5, 6, 7, or 8.

Check your equipment

You can use Micro-Cap 8 on most PC systems. The minimum hardware requirements are as follows:

- A Pentium II or later computer
- Any Windows operating system later than 3.1. This includes NT 4.0 / 2000 and Windows 95 / 98 / ME / XP.
- System memory of 64 MB
- An SVGA or better display adapter and monitor

Register your product

Be sure to fill out and return the enclosed registration materials. You may also register online at our web site:

www.spectrum-soft.com

Registration is important to you for two reasons.

First, it provides access to technical support for the product. Spectrum provides technical support only to registered users. Second, it helps us keep you informed about upgrades and new versions of the product.

Let us know

Let us know what you liked and what you didn't. We want to know your opinions, suggestions, and comments about how to improve the product. So call us, write, or email us. Either way, please tell us what you think about this product and what you'd like to see in future versions.

Install the program

The installation of Micro-Cap 8 is done with the SETUP program. Here are the specific steps:

- Insert the CD in the drive.
- Select the Run item from the Start menu.
- Browse to the CD ROM drive and run the SETUP program.
- Follow the instructions provided by the SETUP program.

During the installation you will be asked where you wish to install MC8. Do not install it in an old MC5, MC6, or MC7, or MC8 Demo folder. Use the default folder, or give it a new folder name. After the installation is complete, you can copy selected circuits, macros, and model files from older folders (MC5, MC6, or MC7) to the MC8 folder. This method is preferred to simply installing MC8 in the older version folder because this way the older version continues to function as before and its files are not overwritten by the MC8 installation.

SETUP creates a main Micro-Cap 8 directory called MC8, a data subdirectory called DATA, and a library subdirectory called LIBRARY. It then copies the programs into the MC8 folder, the model libraries into the LIBRARY folder, and the sample circuit files into the DATA folder.

What's in this chapter

This chapter covers the basics of operating the program. It presumes that you are familiar with the Windows user interface and shows you how Micro-Cap employs that interface to create and analyze electronic circuits. The chapter also reviews the key terms and concepts central to the operation of the program.

How to start MC8

Micro-Cap 8 is activated in the usual way if Windows is already active, by double-clicking on its icon.

Before you run Micro-Cap 8, insert the security key into one of the parallel or USB ports. The key must be connected to a port to start MC8 and must remain there while running the program. If you are using the LAN version, the red LAN key should be in the server parallel or USB port.

A quick tour

As a quick introduction to MC8, we'll take a brief tour of the program. Begin with a double click on the MC8 icon.

To begin the tour, we'll load one of the sample circuit files supplied with MC8. Choose **Open** from the **File** menu. When the file prompt comes up, type in the file name MIXED4. Click Open. The program loads the circuit and displays it.

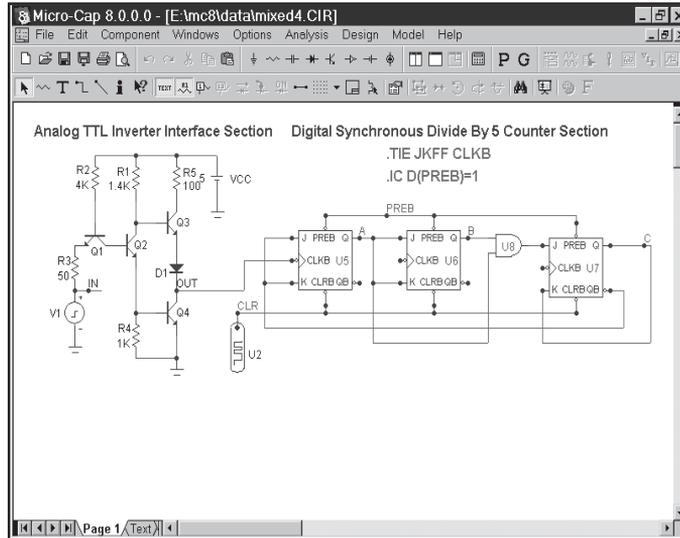


Figure 2-1 The MIXED4 sample circuit

The circuit uses an analog TTL inverter to drive a digital three-stage synchronous divide-by-five counter. The digital clock input to the first JK flip-flop is driven by the analog TTL output, while the CLR initializing pulse for the flip-flops comes from a digital source, U2. The CLK inputs are connected together with a .TIE command. The flip-flop preset nodes are connected together by a wire and the node is labeled PREB with grid text. This node is initialized to a '1' state with an .IC command. The digital source states are specified in a .DEFINE command statement located in the schematic text area. The text area is a private cache in each schematic for storing text. You can toggle the display between the drawing and text areas by pressing CTRL + G or by clicking on the Text tab in the lower page scroll bar. This combination of analog and digital circuitry is easily handled by the program, since it contains a native, event-driven digital simulation engine, internally synchronized with its SPICE-based analog simulation engine. Micro-Cap 8 is a true mixed-signal simulator.

Transient analysis display

Select **Transient** from the **Analysis** menu. MC8 presents the Analysis Limits dialog box where you specify the simulation time range, choose plot and simulation options, and select waveforms to be plotted during the run.

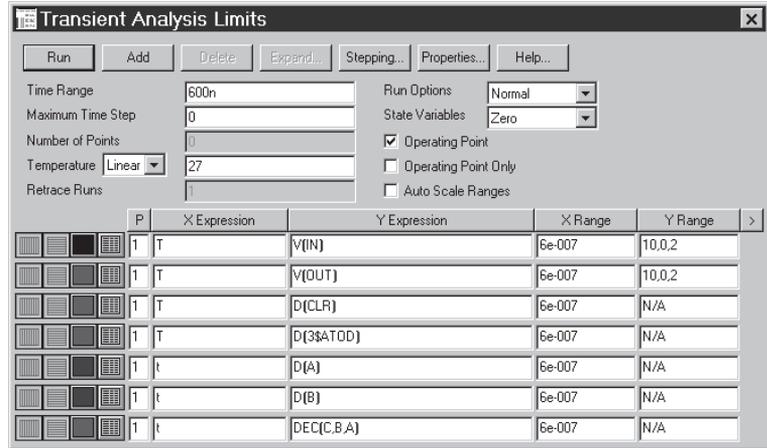


Figure 2-2 The Analysis Limits dialog box

Here we have chosen to plot the analog voltage waveforms at the input pulse source, the output of the analog section, and several of the digital waveforms. In general, MC8 can plot any expression using as variables any node voltage or state or any device terminal voltage or current. Here are a few other expressions that could have been plotted in this circuit.

d(CLR)	Digital state waveform on node CLR.
hex(CLR,PB,1,2)	Hex waveform at nodes CLR,PB,1,2.
vbe(q3)	Base-emitter voltage of transistor q3.
I(VCC)	Current through the VCC source.
PG(VCC)	Power generated by the VCC source.
qbe(q1)	Charge stored in the base-emitter cap of q1.
cbc(q4)	Base-collector capacitance of q4.

There are a great many other variables available for plotting. Click the *right* mouse button in the Y Expression field to see a list of variables available for plotting or printing. Click outside of the menu to remove it.

Now let's run the actual simulation. Click on the Run button  or press F2 to start the simulation. MC8 plots the results during the run. The run can be stopped at any time by clicking the Stop button , or by pressing ESC.

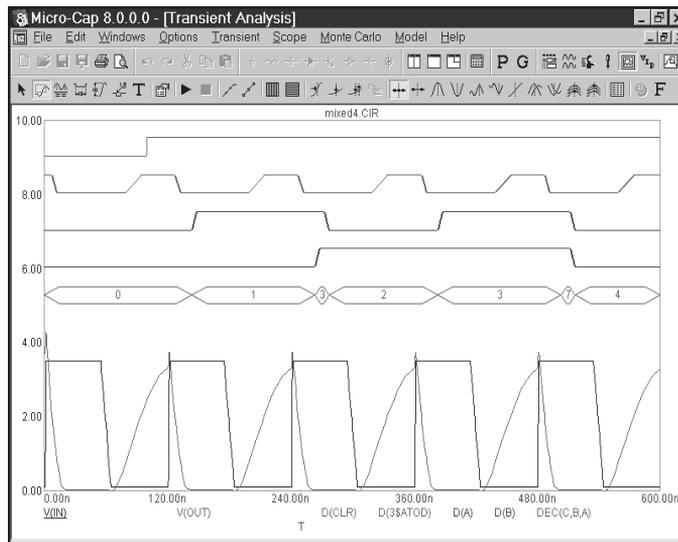


Figure 2-3 Transient analysis of the circuit

The final waveform plots look like Figure 2-3. Each waveform can have a unique token, width, pattern, and color. These options are set from **Options / Default Properties for New Circuits** for new circuits and from the Properties (F10) dialog box for the current circuit. Waveforms can be grouped into one or more plots. Grouping is controlled by the plot number assigned to each waveform. The plot number is the number in the P column of the Analysis Limits dialog box. You can assign all the waveforms to a single plot, or group them into as many plots as will fit on the screen. To assign several waveforms to the same plot, give them all the same plot number. To create several plots, use different plot numbers. To disable plotting a waveform, enter a blank for the plot number.

In this example, all waveforms in the plot group share a common vertical scale. To use different scales disable the **Plot Properties (F10) / Same Y Scales for Each Plot Group** option and enter the scales you want in the Y Range field. All waveforms in a plot group always share the same horizontal or X scale.

Digital and analog waveforms may be mixed or separated. Separated digital waveforms show the expression on the left, adjacent to the plot. Mixed digital waveforms display the expression at the bottom of the plot along with the analog expressions in the order of their occurrence in the Analysis Limits dialog box.

Zooming in on a waveform

Place the mouse near the top middle of the waveform plot group. Press the left mouse button and while holding it down, slide the mouse down and to the right to create an outline box as shown in the figure below. Release the mouse button and MC8 magnifies and redraws the box region.

You must be in Scale mode to zoom in with the mouse.

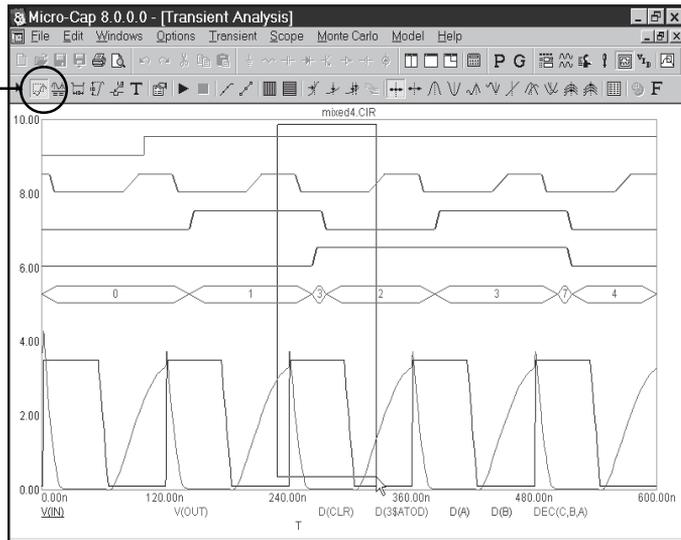


Figure 2-4 Using the mouse to magnify a waveform

If you make a mistake, press CTRL + HOME and try again.

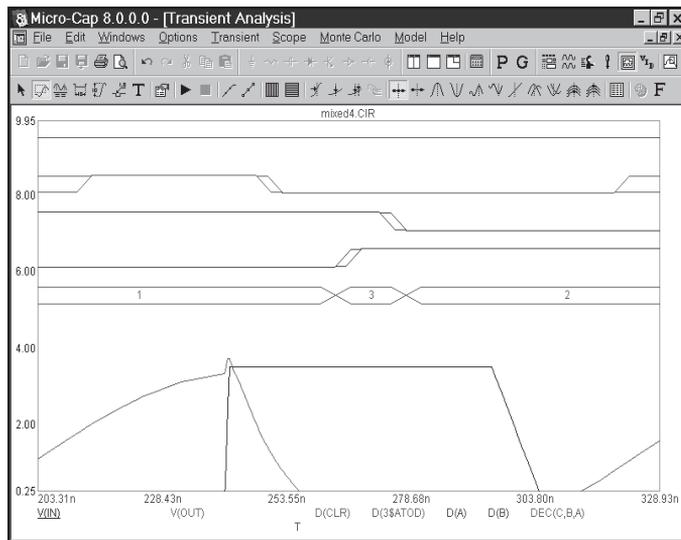


Figure 2-5 The magnified region

Click  to turn off currents and click  to display power terms like this:

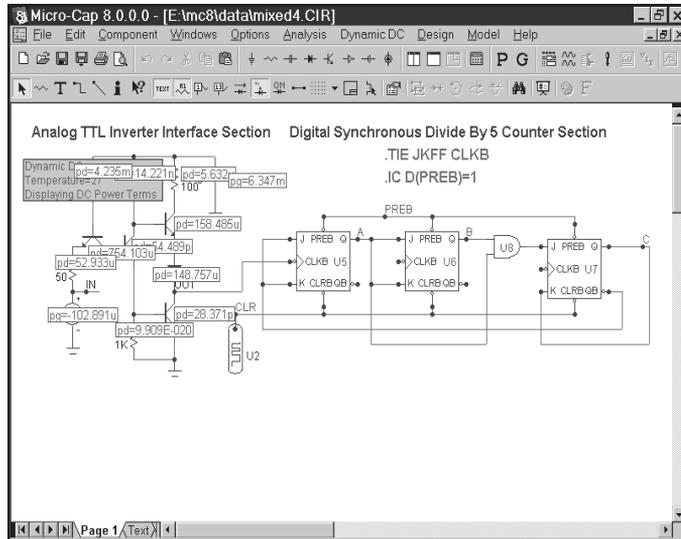


Figure 2-8 Operating point DC power terms

Click  to turn off power and click  to display conditions like this:

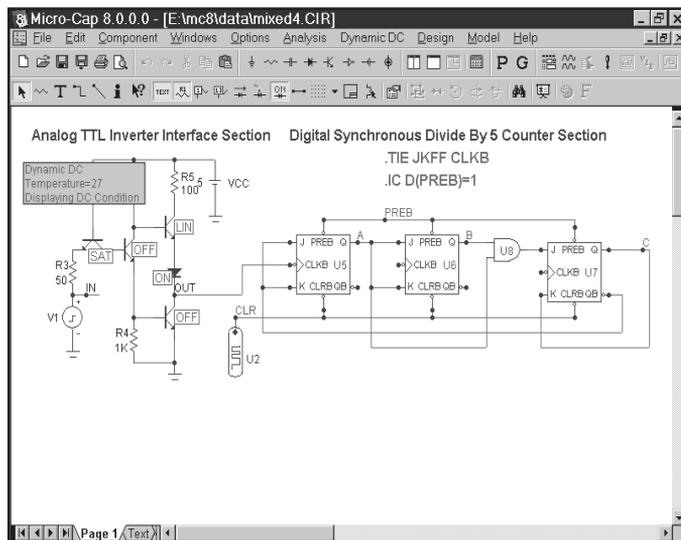


Figure 2-9 Operating point DC conditions

The voltage, current, power, and condition are the current time-domain values, which in this case is the result of a DC operating point calculation.

Turn off the **Dynamic DC** analysis mode from the **Analysis** menu. Remove the MIXED4 circuit by choosing **Close** from the **File** menu. Load a new file by choosing **Open** from the **File** menu. Type RCA3040 for the name. Click Open.

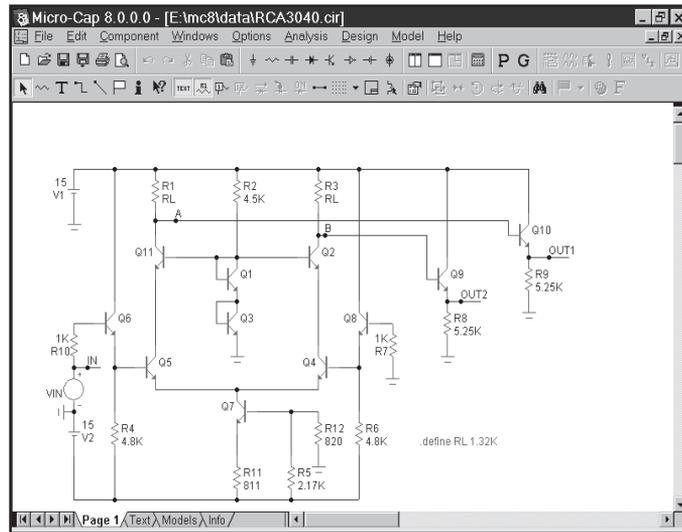


Figure 2-10 The RCA3040 circuit

Select **AC** from the **Analysis** menu. As before, an analysis limits dialog box appears. We have chosen to plot the voltage of the two nodes, Out1 and Out2, in dB. Click on the Run button, and the analysis plot looks like this:

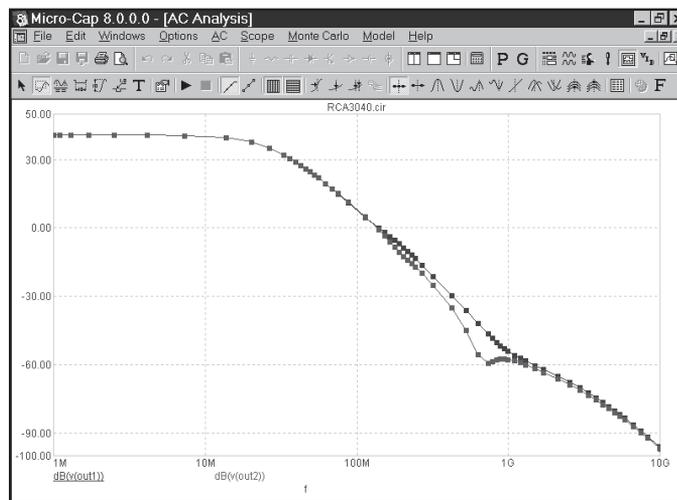


Figure 2-11 AC analysis plot

The same rich set of expressions available in transient analysis is also available in AC analysis. In addition, these operators are very useful:

db(c)	Finds the decibel value of the complex expression c.
re(c)	Finds the real value of the complex expression c.
im(c)	Finds the imaginary value of the complex expression c.
ph(c)	Finds the phase in degrees of the complex expression c.
gd(c)	Finds the group delay of the complex expression c.
inoise	Finds the input noise reflected to the input node.
onoise	Finds the output noise at the output node.

Scope in Cursor mode

Click on the Cursor mode button  in the Tool bar or press F8. In this mode, two cursors are placed on the graph and may be moved about. The table below the plots show the waveform values at each of the two cursors, the difference between them, and the slope. The display should look like this:

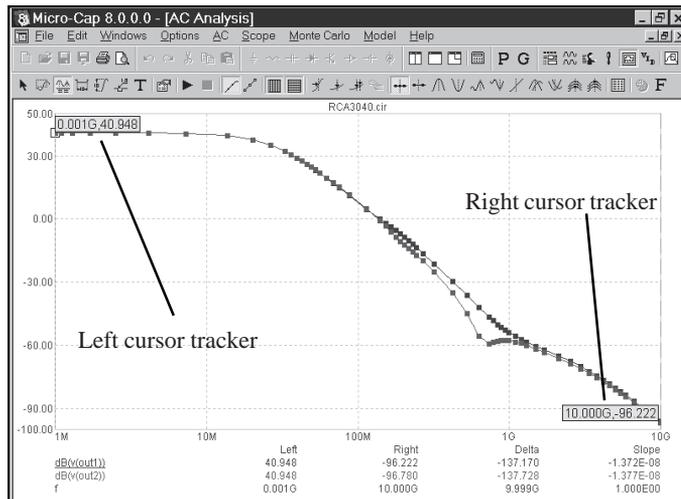


Figure 2-12 Cursor mode

The left mouse button controls the left cursor and the right button controls the right cursor. The cursor keys LEFT ARROW and RIGHT ARROW also control the cursors. As the cursors move around, the numeric display continuously updates to show the value of the waveforms, the change in waveforms between the two cursors, and the waveform slope between the two cursors. The optional cursor trackers also track the X and Y values as the cursors are moved.

Note from Fig 2-10 that the load resistors R1 and R3 are set to RL, a symbolic variable defined as 1.32K. We'll optimize the value of RL to maximize the gain at 1MHz. Press CTRL + F11 to bring up the Optimizer. It looks like this:

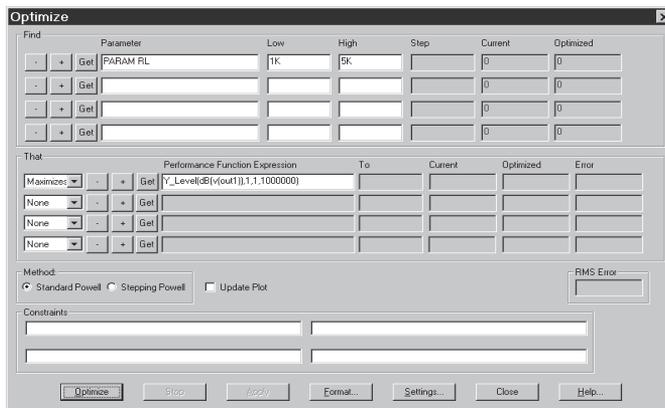


Figure 2-13 The Optimizer dialog box

These settings seek the value of RL that maximizes the performance function $Y_Level(DB(V(OUT1)),1,1,1E6)$. From Fig 2-12 the value of $DB(V(OUT1))$ at 1.0Mhz before optimization is 40.9dB. Click on the Optimize button. The optimizer whirs and chugs and eventually spits out an optimal value of 2.3K. Click on the Apply and Close buttons. Press F2 to rerun the analysis, then F8 and you'll see that with $RL=2.3K$, $DB(V(OUT1))$ at 1Mhz is now 45.4dB, a net gain of 4.5dB. Press F3 to exit the AC analysis routine. Choose **Close** from the **File** menu to unload the circuit.

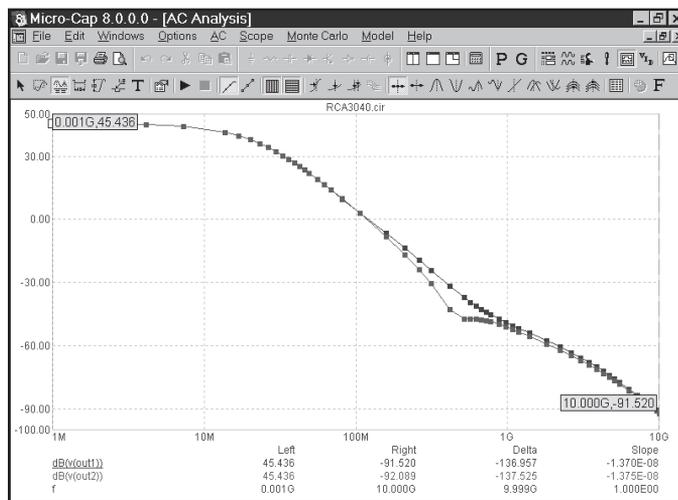


Figure 2-14 The optimized gain curve

DC analysis

Choose **Open** from the **File** menu to load a new circuit, then type "CURVES" and click the Open button.

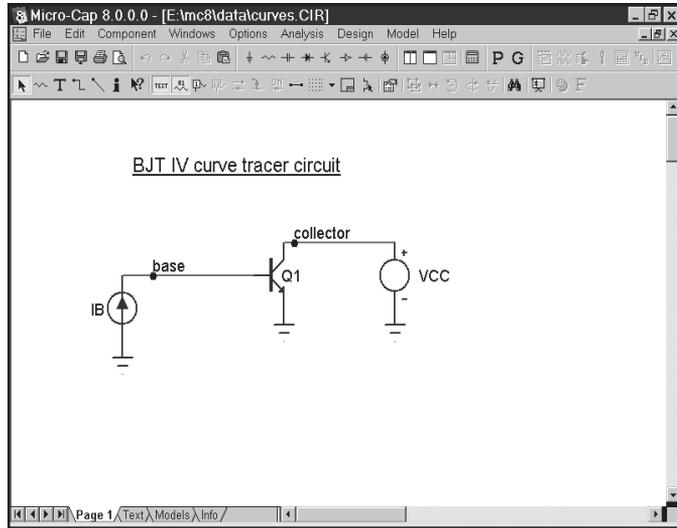


Figure 2-15 The IVBJT circuit

This creates a set of bipolar transistor IV curves. Choose **DC** from the **Analysis** menu. Click on the Run button. The plots look like this:

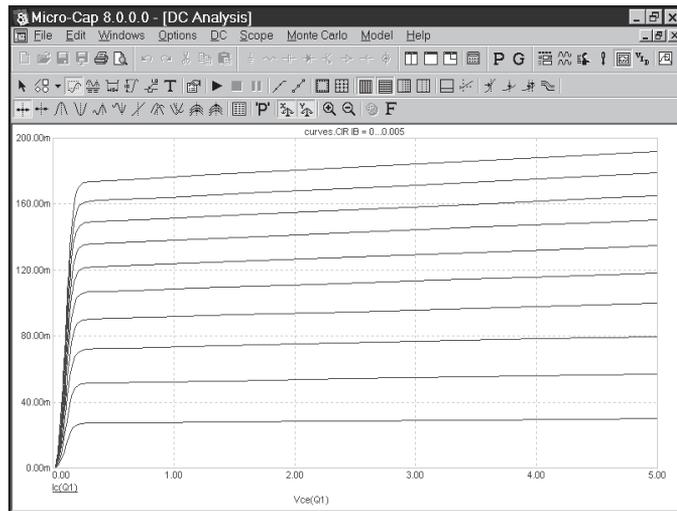


Figure 2-16 The IV curves of a bipolar transistor

Press F3 to exit the DC analysis, then unload the circuit with CTRL + F4.

Press CTRL + O, then type "MIXED4". Click Open. Choose **Probe Transient** from the **Analysis** menu. MC8 presents the analysis limits. Click on the Close button. If needed, MC8 runs the analysis, saves it to disk, and presents the Probe display. You scroll the schematic or pan it with a click and drag of the *right* mouse button and probe any node or component with a click of the *left* mouse button. Move the mouse to the pulse source and click the *left* button on the dot near the IN node, and its voltage waveform is fetched from disk and plotted. Click the mouse on the OUT node, CLKB pin, J pin, and on the CLR node. This plots the analog input and output, and the clock, J, and clear digital waveforms. Analog and digital waveforms are placed in the same plot when the **Separate Analog and Digital** option on the **Probe** menu is not checked.

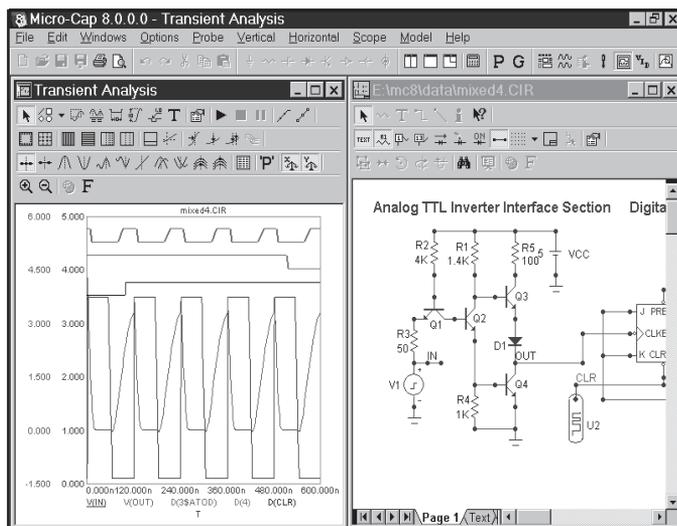


Figure 2-17 The Probe display

Probing on a node produces either analog node voltage waveforms or digital node state waveforms. Probing between the leads of an analog device produces pin to pin terminal voltage waveforms such as $V_{BE}(Q4)$. Probing on a digital device produces a list of the pin names. Selecting a pin name from this list plots the digital waveform on that pin. Current waveforms are also available, as are charge, capacitance, inductance, flux, and all the usual variables. In addition, you can enter expressions involving circuit variables.

Press F3 to exit the Probe routines and return to the Schematic editor. Remove the circuit with CTRL + F4. This completes the quick tour. For more extensive tours of MC8, see the **Demo** options in the **Help** menu.

Command line
format

Command lines and batch files

Although Micro-Cap 8 is designed to run primarily in an interactive mode, it can also be run as a batch process from the Program Manager *command line*. Either of two formats may be used.

```
MC8 [F1[.EXT]]...[FN[.EXT]]
```

```
MC8 [ [/S | /R] | [[/P] | [/PC | /PA ]] ] [@BATCH.BAT]
```

In the first format, F1, ...FN are the names of one or more circuit files to be loaded. The program loads the circuit files and awaits further commands.

Batch mode
format

In the second format, circuits can be simulated in batch mode by including the circuit name and the analysis type, /T(transient), /A(AC), or /D(DC) on a line in a text file. The syntax of the *circuit line* inside the batch file is:

```
CIRCUITNAME [ [/T /A /D] [/S | /R] | [[/P] | [/PC | /PA ]] ]
```

The definition of each command is as follows:

/T	Run transient analysis on the circuit.
/A	Run AC analysis on the circuit.
/D	Run DC analysis on the circuit.
/S	Save the analysis run to disk for later recall.
/R	Retrieve the analysis run from disk and plot the waveforms specified in the Analysis Limits dialog box.
/PC	Print the circuit diagram.
/PA	Print the circuit analysis plot.
/P	Print the circuit diagram and analysis plot.

The /T, /A, and /D commands are for the batch file lines only. The other commands may be applied globally for all circuits in the batch file by placing them on the Program Manager *command line*, or locally to a particular circuit file by placing them on a *circuit line* within the batch file.

In the second format, MC8 must be invoked with the name of the batch file preceded by the character '@'. For example, consider a text file called TEST.BAT containing these three lines:

```
PRLC /A /T  
SENSOR.CKT /A  
LOGIC /T
```

If MC8 is invoked with the command line, MC8 @TEST.BAT, it will:

1. Load the PRLC.CIR circuit and run an AC and transient analysis.
2. Load the SENSOR.CKT circuit and run an AC analysis.
3. Load the LOGIC.CIR circuit and run a transient analysis.

During these analyses, the expressions specified in the circuit file would be plotted during the run and be visible on the screen, but the results would not be saved to disk.

Note that the default circuit file name extension is .CIR.

By appending an '/S' to the command line, all of the analyses specified in the batch file will be run and the results saved to disk for later recall. By appending an '/R' to the command line the analyses are bypassed and the results merely recalled and plotted.

By appending a '/P' to the command line, all of the analyses specified in the batch file will be run and the circuit and analysis plot printed.

Terms and concepts

Here are some of the important terms and concepts used in the program.

Attribute. Components use attributes to identify part names, model names, and other defining characteristics. Attributes are created and edited in the Attribute dialog box which is accessed by double-clicking on the component.

Box. The box is a rectangular region of a circuit or an analysis plot. Created by a drag operation, it defines plot regions to be magnified or circuit regions to be stepped, deleted, rotated, moved, turned into a macro, or copied to the clipboard.

Click the mouse. Most mice have two or more buttons. 'Click the mouse' means to press and release the left or right button. The middle button is not used.

Clipboard. This is a temporary storage area for whole or partial circuits from a schematic, or text from any text entry field. Items are copied to the clipboard with the Copy command (CTRL + C) and pasted (CTRL + V) from the clipboard into the current schematic or data field with the Paste command. Anything that is cut is also copied to the clipboard. Anything that is cleared or deleted is not copied to the clipboard.

Component. A component is any electrical circuit object. This includes connectors, macros, and all electrical devices. This excludes grid text, wires, pictures, and graphical objects.

Copy. This command (CTRL + C) copies text from a text field or circuit objects from a schematic to the clipboard for use with the Paste command (CTRL + V).

Cursor. The cursor marks the insertion point. When an object is inserted into a circuit or text inserted into a text entry field, it is placed at the cursor site. The schematic cursor is a mouse arrow or the actual component shape, depending upon a user-selected option. The text cursor is a flashing vertical bar.

Disabled. This means temporarily unavailable. See Enabled.

Drag. A drag operation involves pressing the mouse button, and while holding it down, dragging the mouse to a new location. Dragging with the right button is used for panning shape displays, circuit schematics and analysis plots. Dragging with the left button is used for box region selection, drag copying, moving circuit objects, and magnifying an analysis plot region.

Enabled. A check box button feature is enabled by clicking it with the mouse. It shows an X or a check mark when enabled. A menu item feature is enabled by clicking it to show a check mark. A button feature is enabled by clicking it to depress the button. Button features are disabled by clicking them to release the button. Other features are disabled when they show neither an X nor a check mark.

ESC. The ESC key is used as a single key alternative to closing the Shape editor, Component editor, Stepping, Monte Carlo, and State Variables dialog boxes and to stop a simulation. It is equivalent to Cancel in a dialog box.

Grid. The grid is a rectangular array of equally spaced locations in the circuit that defines all possible locations for the origin of circuit objects. The grid can be seen by clicking on the Grid icon.

Grid text. This refers to any text placed in a schematic. It is called grid text to distinguish it from component attribute text. Grid text may be placed anywhere. Attribute text is always adjacent to and moves with the component. When grid text is placed on a node it gives the node a name which can be used in expressions. Grid text may be moved back and forth between a schematic page and the schematic text area by selecting the text and pressing CTRL + B.

Wire. Wires connect one or more nodes together. Crossing wires do not connect. Wires that terminate on other wires are connected together. A dot signifying a connection is placed where a wire connects to another wire or to a component pin. Two wires placed end-to-end are fused into a single wire.

Macro. A macro is a schematic, created and saved on disk to be used as a component in other schematics. It uses pin names to specify its connections to the calling circuit.

Node. A node is a set of points connected to one or more component pins. All points on a node have the same voltage or digital state. When analog and digital nodes are joined, an interface circuit is inserted and an additional node created. The process is described in detail in the Reference Manual.

Node name. A node name may be either a node number assigned by MC8 or a piece of grid text placed somewhere *on* the node. Grid text is *on* a node if its pin connection dot is on the node. The dot is in the lower left corner of the box that outlines the text when it is selected. To display pin connections, select the **Pin Connections** option from the **View** item on the main **Options** menu. *Nodes that have the same text are connected together.*

Node number. A node number is an ID number assigned by the system. You may refer to nodes either by number or name. Names are better, however, since node numbers can change when you change the schematic.

Object. An object is a general term for anything placed in a schematic. This includes components, wires, grid text, flags, picture files, and graphical objects.

Pan. Panning is the process of moving the view of a shape, schematic, or an analysis plot. It is like scrolling, except that it is omnidirectional. Panning is done with a drag operation using the right mouse button.

Paste. The Paste command (CTRL + V) copies the contents of the clipboard to the current cursor location. Pasting is available for schematics and text fields. To control the location of a paste operation, first click in the schematic or text field where you want the upper-left corner of the paste region to be, then paste.

Pin. A pin is a point on a component shape where electrical connections to other components are made. Pins are created and edited using the Component editor.

Select. An item is selected so that it may be edited, moved, rotated, deleted or otherwise changed. The general procedure is:

- 1) If the item is anything but text in a text field, enable the Select mode by clicking on the Select mode  button.
- 2) Select the item.
- 3) Change it.

Item means any dialog box feature, text fragment in a text field, or any object in the Shape editor, schematic window, or analysis plot. Items are selected with the mouse by moving the mouse arrow to the item and clicking the left button. Items are selected with the keyboard by pressing the TAB key until the desired item is selected. Keyboard selection is available only for text objects and dialog box features. Shading, outlining, or highlighting is used to designate selected items.

Shape. All components are represented by shapes. These are created, modified, and maintained by the Shape editor.

Function keys

F1 is used to invoke the Help system. It accesses an information database by contents and by alphabetical index.

F2 is used to start an analysis after selecting the type of analysis from the Analysis menu.

F3 quits the AC, DC, or transient analysis and returns to the Schematic editor. F3 also repeats the last search when in the Schematic editor.

F4 displays the analysis plot.

CTRL + F4 closes the active window.

F5 displays the Numeric Output window.

F6 scales and plots the selected analysis plot group.

CTRL + F6 cycles through the open windows.

F7 switches the analysis plot to Scale mode.

F8 switches the analysis plot to Cursor mode.

F9 displays the Analysis Limits dialog box for AC, DC, or transient analysis or their Probe equivalents.

CTRL + F9 clears the waveforms in Probe and invokes the Analysis Limits dialog box when in an analysis module.

F10 invokes the Properties dialog box for the front window. The type of dialog box depends upon the type of front window.

F11 invokes the Parameter Stepping dialog box.

CTRL + F11 invokes the Optimizer dialog box.

F12 invokes the State Variables editor.

The Undo and Redo functions

Micro-Cap 8 has a multistep undo and redo capability. All schematic edits can be undone back to the point where the circuit was loaded. In general, text field undo is limited to the last state. The one exception to this is the Component editor, where text field edits are multistep.

Undo is accomplished by clicking on the Undo button  or by pressing CTRL + Z. The Undo function is shown under the **Edit** menu.

Redo is accomplished by clicking on the Redo button  or by pressing CTRL + Y. The Redo function is shown under the **Edit** menu.

For instance, suppose you load a circuit, and add a diode to it, then change a resistor value from 1K to 2K, then delete a capacitor. Pressing Undo once restores the capacitor. Pressing Undo again restores the resistor value to 1K. Pressing Undo again removes the diode, restoring the schematic to its initial state. Pressing Redo once restores the diode. Pressing it again restores the resistor to 2K, and a third press deletes the capacitor. Only RAM memory limits the depth of the undo/redo for schematics.

Schematic edits may be undone even after running an analysis. You can delete a region of circuitry from a circuit, run an analysis, then return to the Schematic editor and use the Undo function to restore the circuit to its former condition.

Note that the **Revert** command on the **File** menu also functions like a large-scale undo command. It loads the existing version of the front window file from disk, discarding any changes to the front window that occurred since it was loaded.

What's in this chapter

This chapter shows you how to create and edit simple circuits. The goal of this chapter is to introduce the basic techniques and to refine them into working skills by practicing on sample circuits. The particular subjects covered are:

- Schematic creation
 - Adding components
 - Entering component parameters
 - Connecting components with wires
 - Using define and model statements
 - Adding node names
- Schematic editing
 - Text editing modes
 - Editing component parameters and text
 - Deleting objects
 - The clipboard
 - Selection
 - Viewing large schematics
 - Creating and editing SPICE text files

Creating a simple circuit

Begin by double-clicking the mouse on the MC8 icon. The program automatically opens a new circuit window and names it CIRCUI1. Using the mouse, select the **Component** menu, then **Analog Primitives**, then **Waveform Sources**, and finally, select **Voltage Source**.

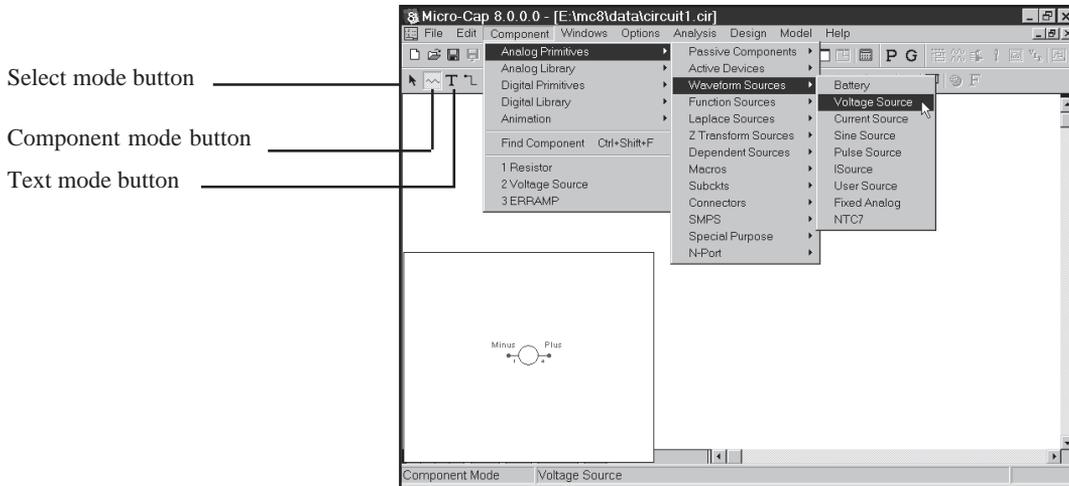


Figure 3-1 Selecting a part from the Component menu

Selecting a part from the menu changes the mode to Component, anticipating the placement of the component in the schematic.

The cursor changes to a voltage source shape. Click the left mouse button and hold it down. Press the *right* mouse button and the shape rotates. Eight clicks of the right mouse button produce the eight possible orientations. While still holding the mouse button down, slide the source to a position halfway down the circuit window and near the left window edge. Rotate it until the plus symbol is on top and the minus symbol is on the bottom. Release the mouse button. This activates the Attribute dialog box as shown in Figure 3-2.

For this type of source there are two important attributes, the part name and the value. MC8 creates a suitable part name, in this case V1, which you can accept or edit. In this type of source, the value attribute embodies all of the necessary parameters. You can enter a new value attribute, edit the existing one, or select a different source from a combination of tabs and buttons. Each of the tabs selects one of the eight basic source types; None, Pulse, Sin, Exp, PWL, SFFM, Noise, or Gaussian. Each of the buttons selects a common waveshape. Select the Pulse tab and the SAWTOOTH button. You can change the parameter values and name associated with any button by clicking on the Save Settings button.

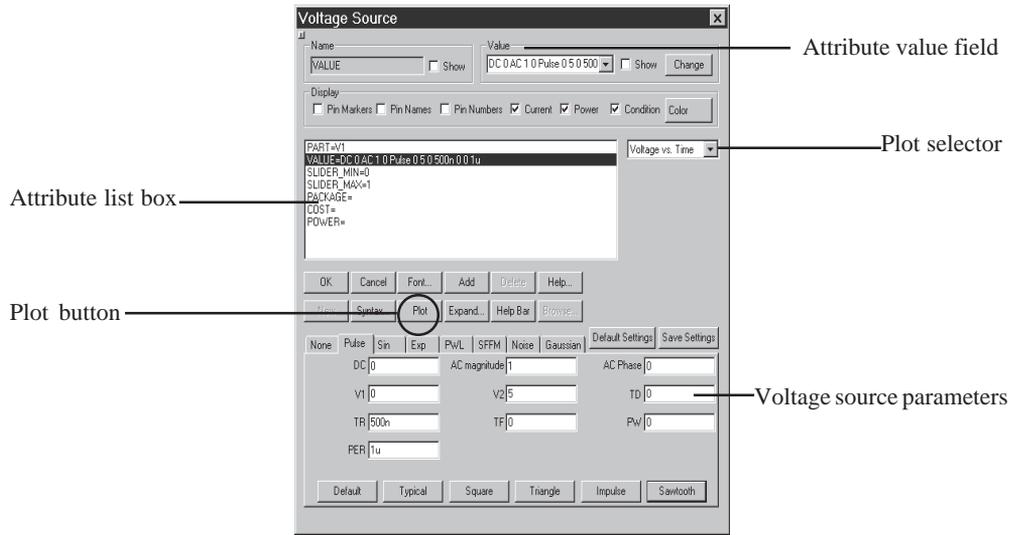


Figure 3-2 The Attribute dialog box

You can see the pulse source waveform by clicking on the Plot button. For this set of parameters the waveform looks like this:

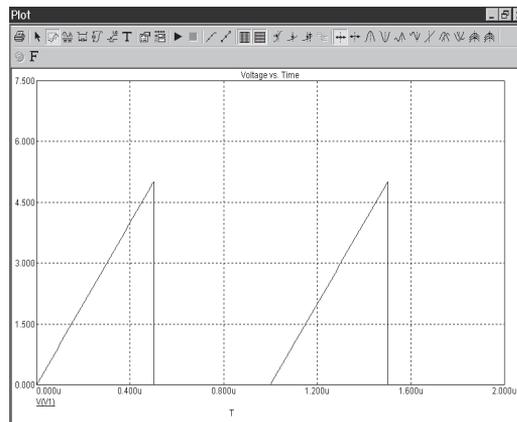


Figure 3-3 Characteristic plot of the source

The Plot button produces from zero to three plots for each basic part type. They are created on the fly from mini-simulations of test circuits designed to produce the plot in question. Some parts, like this source, have a simple simulation and a single plot. Others, like the NPN, have several plots that can be chosen from the plot selector. Some parts, like macros and subckts, have no plot at all since it is impossible to know what would be a relevant plot for a general part. Once the plot is up, it responds dynamically to parameter edits. To illustrate, change the value of TF to 500n and press the Tab key. The plot looks like Figure 3-4.

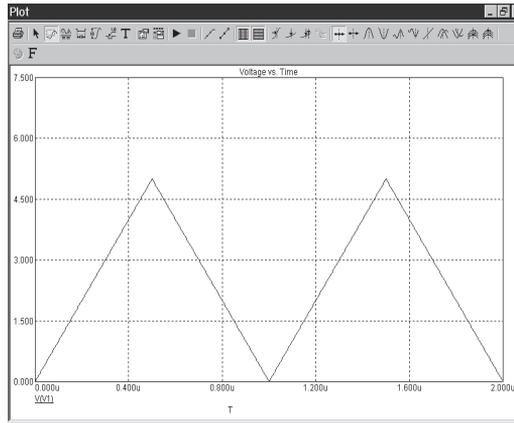


Figure 3-4 Characteristic plot of the source with TF=500n

While the Component menu hierarchical structure provides easy access to parts, it can take several steps to find the one you want. A fast way to select frequently used parts is the Component palette. These user-selected parts palettes can be toggled on and off with a single key. You click on the part you want, then leave the palette on the screen or make it disappear with the same key that invoked it. The palettes, numbered 1 to 9, are toggled on and off with CTRL + number. Membership in a palette is specified with the Component editor. We'll use a palette to add a resistor. Press CTRL + 1 to invoke palette 1. The screen should now look something like this:

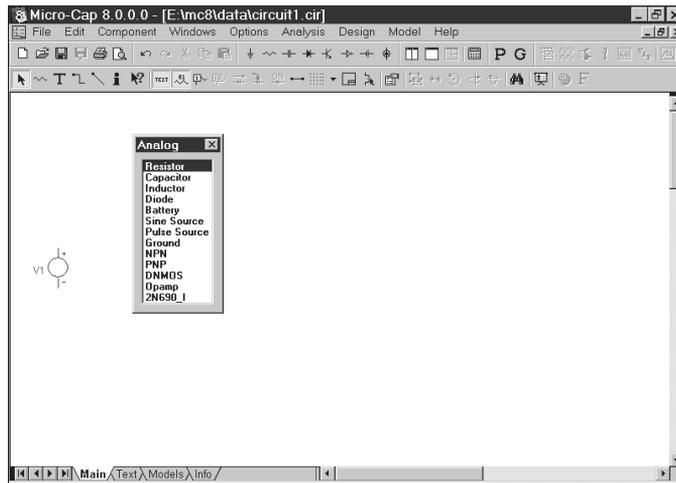


Figure 3-5 A user palette

Click on the Resistor name in the palette. Press CTRL + 1 to remove the palette. Drag the resistor to the source and rotate it until it is horizontal. When one of the

resistor leads is just touching the top of the source, release the mouse button. If the Node Snap option is on, the resistor lead only needs to be near the source. The resistor lead will snap to the nearest node. Release the mouse button, and the Attribute dialog box will appear. MC8 suggests the name R1 which we'll accept. Type "1K" for the VALUE attribute, and press ENTER.

SPACEBAR
toggles between
the current mode
and Select mode.

Let's move the resistor attribute text. To move anything we must be in Select mode. To activate Select mode, click on the Select mode button  in the upper left portion of the Tool bar, or simply press SPACEBAR. Drag the resistor name, R1, and its value 1K, until they are centered, respectively, above and below the resistor body. Attribute text can be moved relative to the part by simply dragging it to the desired position. The initial position of attribute text relative to the shape is determined in the Component library. After the part is placed, its text attributes (e.g. NAME and VALUE) can be moved as desired. Attribute visibility can also be toggled on and off from the Attribute dialog box.

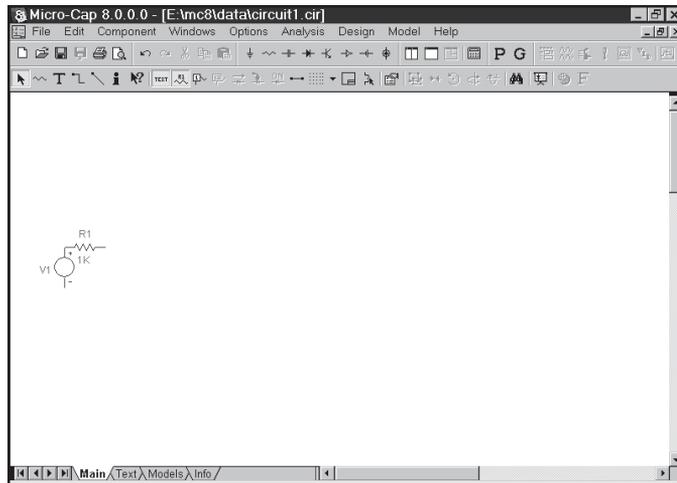


Figure 3-6 The circuit after moving the VALUE attribute

Select **Analog Library / BJT / General / 2N0000- / 2N3903- / 2N4013** from the **Component** menu. This selects the 2N4013, a pre-modeled bipolar NPN transistor. Drag the NPN so that its emitter points down and its base touches the right resistor lead. Click the mouse. The Attribute dialog box doesn't appear since all of the components in the Analog Library and Digital Library groups have their Assign Component Name to MODEL flag set. This sets the MODEL attribute to the component name. When you select the part name 2N4013, the model name is known to be the same, and its model parameters are readily available in the model libraries. Since the part name and model name are already known, the dialog box isn't needed and doesn't appear. It is easier to use a part from the Analog Library

and Digital Library sections of the Component menu than from the Analog Primitives and Digital Primitives sections. You simply pick a part from the library and drop it into the schematic. No modeling decisions are required. Parts chosen from the primitives portion of the library, like the Voltage Source, require a specific model name or parameter sets, which may be selected from the libraries or entered by editing if desired.

Select a new part from the Component menu or User palette only if the part is different from the last one chosen.

Invoke palette 1 again with CTRL + 1. Select Resistor from the palette. Click in the schematic and add a 10K resistor. Point it up from the collector lead. Select Battery from the palette and place a battery vertically so that its positive lead touches the top resistor lead. Set its value to 10. From the palette, select Ground. Place a ground vertically with its lead touching the negative battery lead. Add grounds to the emitter and to the pulse source minus lead. Toggle the palette off with CTRL + 1. The circuit should now look like Figure 3-7.

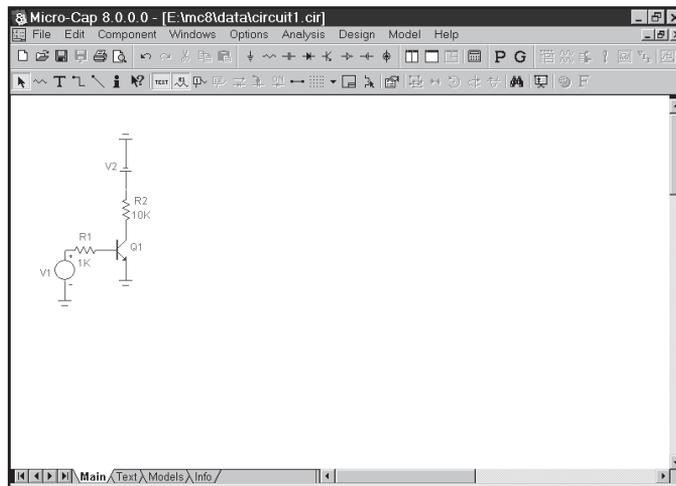


Figure 3-7 The circuit after adding the bipolar stage

To illustrate a third way to select parts, select the **Find Component** command from the **Component** menu or press SHIFT + CTRL + F. Enter OPO8_AD. Click on the Find and OK buttons. Place the opamp to the right of the NPN.

To illustrate a fourth method of part selection, click on the battery  button in the middle of the top row of the Tool bar. Click in the schematic and place a 4.0 volt battery vertically so that its positive lead touches the '99' or 'VCC' lead of the OPO8_AD. Click on the ground  button in the Tool bar. Place a ground at the negative lead of this battery and at the '50' or 'VEE' lead of the OPO8_AD.

For a wire to connect two leads together, the endpoints must terminate on a lead or a wire connected to a component lead. To see where the leads are, click on the Pin Connections  button. This draws a small dot at each component lead. You can also display pin names and connections individually for each component by enabling the Display Pin Names option from the Attribute dialog box.

Wire mode lets you connect components by dragging the mouse to create a wire.

Click on the Wire mode button  in the Tool bar or press CTRL + W. This changes the mode so that dragging the mouse draws a wire. Place the mouse at the '3' or 'plus input' lead. Click the mouse button, and while holding it down, drag the mouse horizontally to the left until the wire touches the junction between the collector lead and the resistor lead. Release the mouse button. This connects the two leads together.

Connect the '2' or 'minus input' lead to the '45' or 'output' lead by clicking near the output lead and dragging a wire straight down until it is below the opamp. While still holding the button down, drag the mouse to the left until it is just below the '2' lead. Release the mouse. This creates a right-angled wire extending from the output to an intermediate point below the input lead. Click again at this intermediate point to start a second wire. Drag it to the '2' lead, and release the mouse.

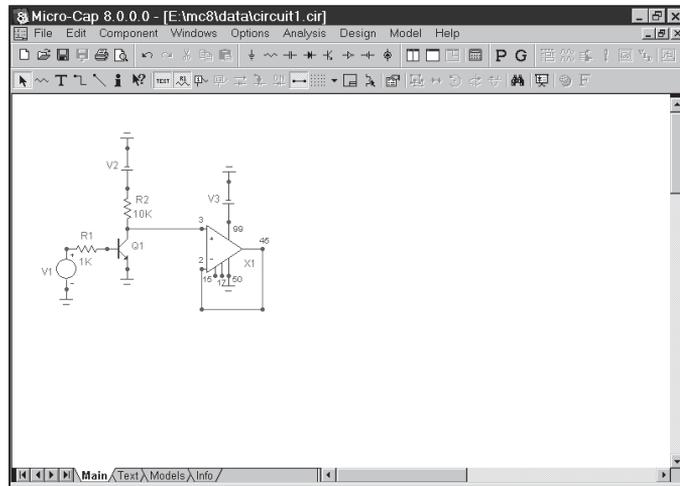


Figure 3-8 The circuit after wiring the OP08

From the **Component menu / Digital Library / 74xx00- / 04-**, select the 7404 buffer. Place the buffer horizontally so that its input lead is to the right of the OP08. Draw a wire from the OP08 output lead to the 7404 input lead. Draw another wire from the 7404 output lead horizontally to the right. Finally, draw the final wire from the pulse source plus lead horizontally to the left.

Text mode lets you place grid text anywhere in the schematic.

Grid text is mainly used to name nodes.

Text is not case sensitive. 1u is the same as 1U.

Click on the Text mode button **T** or press CTRL + T to enter Text mode. This mode lets us add text to the schematic. Text placed directly on nodes gives the node a name which can be used to refer to it when plotting its node voltage or digital state.

Click on the horizontal wire just added to the pulse source. When the dialog box comes up, type "IN" and click on the OK button. The lower edge of the selection box outlining the text "IN" should just touch the wire. If it doesn't touch, enable the Select mode and drag the text until it does touch. Add the text "A" to the wire connecting the NPN collector to the OP08 plus input lead. Add the text "A" to the wire connecting the NPN collector to the OP08 plus input lead. Add the text "C" to the wire connected to the 7404 output. The circuit should now look like Figure 3-9.

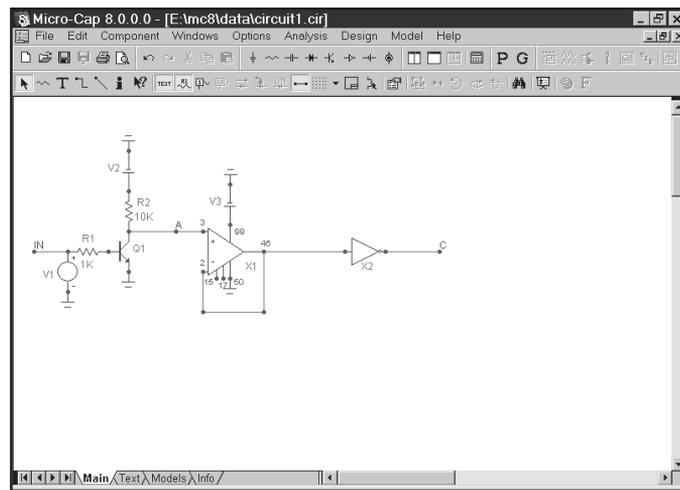


Figure 3-9 The completed circuit

MC8 assigns node numbers to every analog and digital node to identify them. To see the node numbers, click on the Node Numbers button **IN** in the Tool bar.

The circuit should look like Figure 3-10.

Node names and numbers

The program automatically assigns numbers to the circuit nodes when an analysis is requested or when the circuit is saved to disk. It displays them on the schematic only when the Node Numbers option is enabled. When you want to plot or print a node's voltage waveform you refer to the waveform as V(node name), where the node name may be either the node number assigned by the program or a text name assigned by you.

For the program to know about node names, they must be placed directly on the node. The lower left corner of the text outline box must be placed directly on the

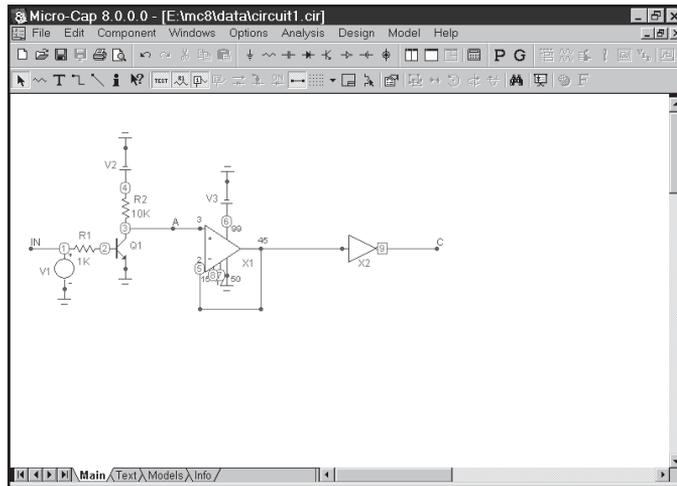


Figure 3-10 Node numbers

node. Node Snap, if enabled, makes this easier by moving the text to the nearest node within one grid. The Node Snap option is selected from **Options menu / Preferences / Options / Circuit**.

The system assigns and displays node numbers according to the following rules:

1. Any node connected to a ground symbol  is assigned the number 0, but its node number is never displayed.
2. The other nodes are numbered from 1 to the highest node number.
3. When an analog node and a digital node touch, or are connected by a wire, each node is assigned a unique number. MC8 automatically inserts an interface circuit between the two nodes. The interface circuit creates an interface node of the form <num>\$ATOD or <num>\$DTON depending upon whether the digital node is an input or an output, respectively. If the interface node is accessible (can be referenced in a plot expression), it will be printed on the schematic. In general, interface nodes between analog parts and digital primitives are accessible unless they occur at a subcircuit interface. Since the interface node in this circuit occurs at a subcircuit interface it isn't accessible.
4. Analog node numbers are displayed in a box with rounded corners while digital node numbers are displayed in a box with square corners.
5. Nodes with the same text node name are connected together. This is done to provide a convenient way of connecting large numbers of common nodes.

Editing component parameters and text

Schematics are composed of components, text, graphical objects, picture files, and wires. To edit any of these items, the item must be selected. To select an item, MC8 must be in the Select mode. To enable the Select mode, press CTRL + E, click the Select  button in the Tool bar, or press SPACEBAR.

Once in the Select mode, objects or regions can be selected for editing, moving, or deleting. To illustrate, load the sample circuit DIFFAMP and click on the Select button in the Tool bar. Move the mouse to the 6V battery in the upper right part of the schematic. Click the mouse in the center of the battery. This selects and redraws the battery. The display mode, Select or Block Select, can be changed through the Preferences dialog box. The Select or Block Select color can be changed in the Properties dialog box (F10).

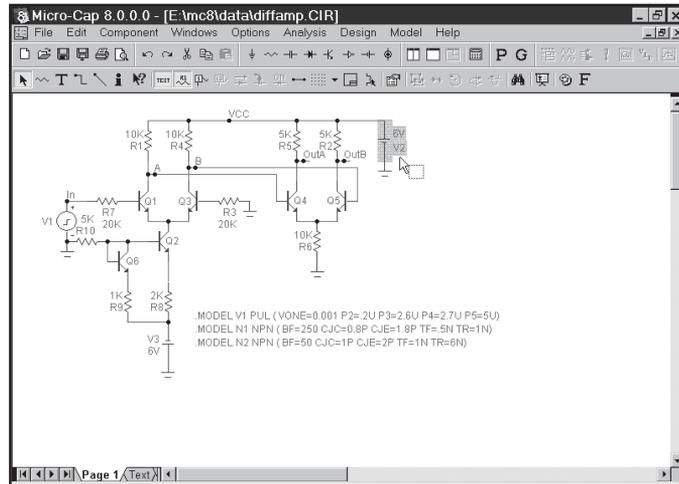


Figure 3-11 Selecting the battery for editing

Double-clicking on a selected component activates the Attribute dialog box which lets you edit its attributes. Double click on the battery. MC8 presents the dialog box showing the value attribute 6V and awaits your edits. Type "6.5". Click OK. That's how you change the parameter of this simple part.

To change the forward beta of a transistor simply double-click on the device and when the Attribute dialog box comes up, edit the BF parameter. Editing always localizes the parameters. If a transistor's model parameters are global, editing will make them local. Mechanically, localization is implemented by placing a model statement with the edited parameters in the text area of the circuit.

To illustrate, double-click on the Q5 transistor. This invokes the Attribute dialog box which should look like this:

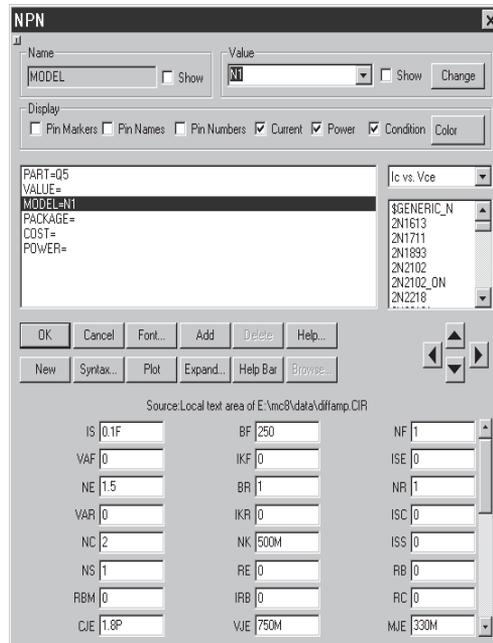


Figure 3-12 Editing model parameters in the Attribute dialog box

In this case, MC8 searched for Q5's model statement, N1, and displayed it in the edit area of the dialog box. When searching for the model statement, MC8 first looks in the schematic and text area. If it doesn't find it there, it looks in libraries referenced by a .LIB statement, including the default master index, NOM.LIB. If it still doesn't find the model statement, it makes one up from default parameter values and copies it to the text area.

After locating the model statement (or creating a suitable default), MC8 displays the parameter values in the edit fields. You can scroll through the parameters, editing them as you wish. When you click OK, MC8 updates your local copy of the model statement. The program never modifies any global libraries as a result of these edits. It only modifies model parameters local to the circuit.

You can click on any model name in the Model list box. Selecting one of these by clicking on it will change the part's model name attribute .

If you select a model name that is not in the Model library, MC8 loads a set of default parameters. If you pick a part from the Model list, MC8 reads the model

parameters from the library source file. Either way the parameters are displayed for review and editing. The model parameters shown for the N1 NPN model are local, as you can see by the line:

Source: Local text area of E:\MC8\DATA\DIFFAMP.CIR

This illustrates how MC8 accesses model data such as subcircuit descriptions and model parameters:

Model data is global until edited when it becomes local to the circuit.

Global means the information is stored in the MC8 library folders. Each time you run an analysis, MC8 uses the global libraries to build the simulation data base.

Local means the information is stored within the circuit. Each time you run an analysis, MC8 uses the local in-circuit copy of the models to build the simulation data base.

To localize or simply refresh model information, use the **Edit Menu / Refresh Models**  command. This command can either copy model data into the local circuit or refresh model data already there depending upon the options chosen from its dialog box.

Deleting objects

Clear saves clipboard contents. *Cut* replaces clipboard contents with deleted objects.

We can now edit text and component parameters. How about deleting objects? Like editing, deleting requires that the objects be selected. Once selected, they can be deleted with one of two commands; Cut and Clear. Clear, activated by the Del key, deletes without copying to the clipboard. Cut, activated by CTRL + X, deletes and copies to the clipboard. To illustrate, select the pulse source in the DIFFAMP circuit by clicking it. As usual the program colors it to show that it is selected. Press the Del key and the program removes the source.

Undoing and redoing operations

MC8 provides multistage undo and redo functions to reverse the effects of schematic edits and a simple one-stage undo function for most text fields. To illustrate, press CTRL + Z to undo the deletion. The source, which we just deleted, reappears. Press CTRL + Y to redo the change and the source disappears again.

The clipboard

MC8 maintains two clipboards for temporary storage. One is for schematics and the other is for text fields. There are two important clipboard operations.

To copy something to the clipboard, select it, then press CTRL + C.

To paste the contents of the clipboard to a schematic, click the mouse on the schematic at the desired insertion point and press CTRL + V. The clipboard contents are then copied to the schematic and selected. They can be moved by dragging on any selected object in the group. Clicking on any deselected object deselects all objects.

To illustrate, click on the Select button  in the Tool bar. Place the mouse at the upper left of the second differential stage of the DIFFAMP circuit. Drag the mouse to the lower right until the display looks like Figure 3-13. Upon release of the mouse button, objects originating inside the outline box are selected. Objects originating outside the box are not selected.

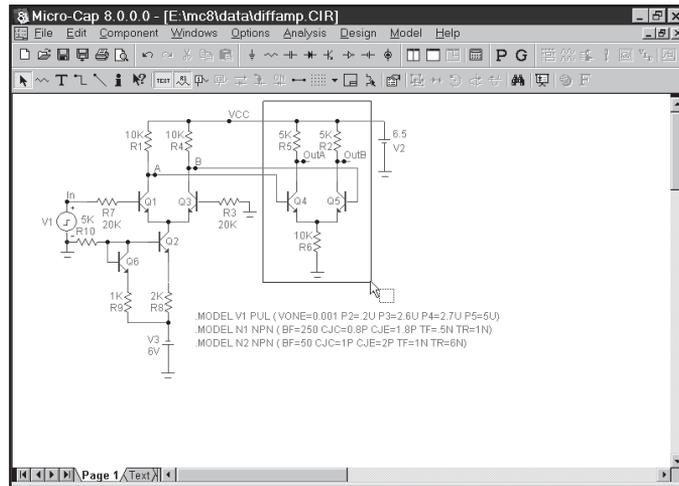


Figure 3-13 Selecting a region for copying to the clipboard

Now press CTRL + C to copy the selected region to the clipboard. Move the mouse to the right of the battery and click the left button. This moves the insertion point. Press CTRL + V to paste the contents of the clipboard to the insertion point. Your circuit should now look like this:

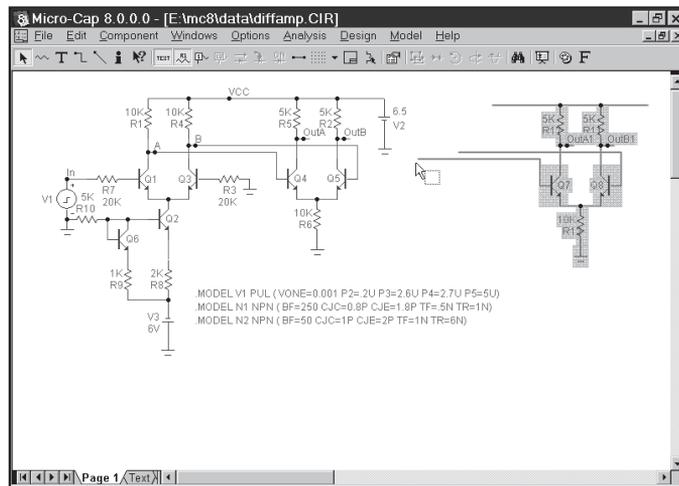


Figure 3-14 After pasting the clipboard contents

The saved region has been copied, or pasted, from the clipboard to the schematic.

Drag copying

Ordinarily, when you drag a selected object or group of objects, the entire group moves with the mouse. However, if you hold down the CTRL key while you drag, MC8 leaves the original selected objects in place, makes a copy, and drags the copy along with the mouse. It's like tearing off a sheet from a pad of printed copies. As with stepping and clipboard pasting, the part names are incremented. Grid text is incremented only if the **Options / Preferences / Options / Text Increment** option is enabled.

To illustrate with the DIFFAMP circuit, click on the Select button, and drag a region box around the V2 battery at the upper right. Release the button. Press CTRL. Click within the selected region and drag it to the right. Notice that a copy of the selected region is created as shown below.

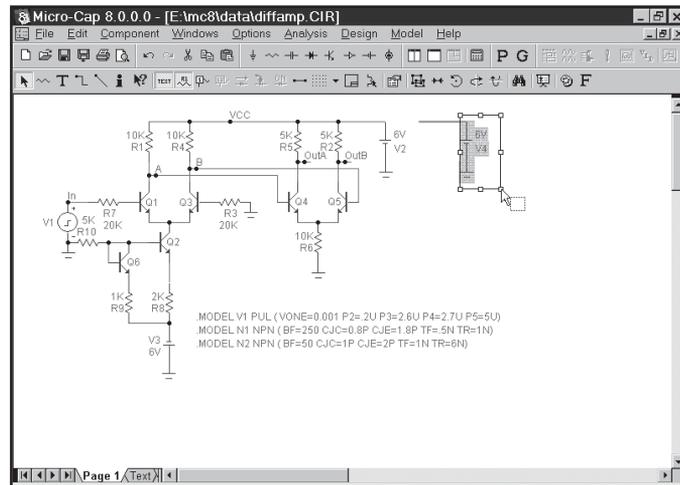


Figure 3-16 Drag copying

Drag copying is usually more convenient than clipboard cut and paste when you need to make only one copy. Its one step operation is easier and simpler to use than copying and pasting from the clipboard.

If you plan to paste more than one copy, then copying and pasting from the clipboard is usually faster.

Navigating large schematics

How do you navigate a large a schematic?

Pan means to move the window view.

1. Panning: This is by far the easiest method. To mouse pan, drag the *right* mouse button. It's like sliding a piece of paper across a desktop. Keyboard panning uses CTRL + <any cursor key> to move the view in the direction of the arrow.
2. Scroll bars: Use the schematic scroll bars. This is slow but sure.
3. Relocation: Use the SHIFT + click method to relocate and change views. While holding down the SHIFT button, click the right mouse button where you want the window centered. Clicking toggles the scale between high and low magnification and centers the schematic at the mouse position.
4. Use the Page scroll bar if the desired area is on another page or use CTRL + PAGE UP and CTRL + PAGE DOWN to navigate the pages.
5. Place flags where desired using the flag mode  button. Then select a flag from the list by clicking on the Go To Flag  button.
6. Use the Zoom-Out  or Zoom-In  buttons in the Tool bar.

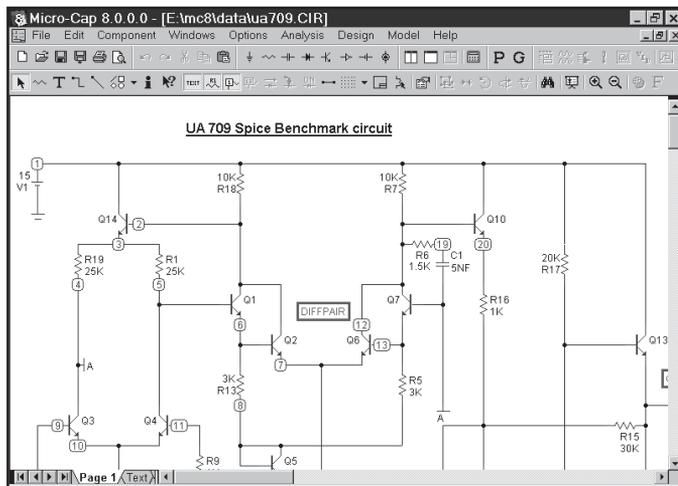


Figure 3-17 The UA709 circuit

To illustrate these methods, load the circuit UA709. It looks like Figure 3-17. At a resolution of 600 X 800, it's too large to see at the default scale, so shrink its scale by clicking twice on the Zoom-Out button. At this scale the circuit looks like this:

The Zoom-In button increases the image size



The Zoom-Out button decreases the image size

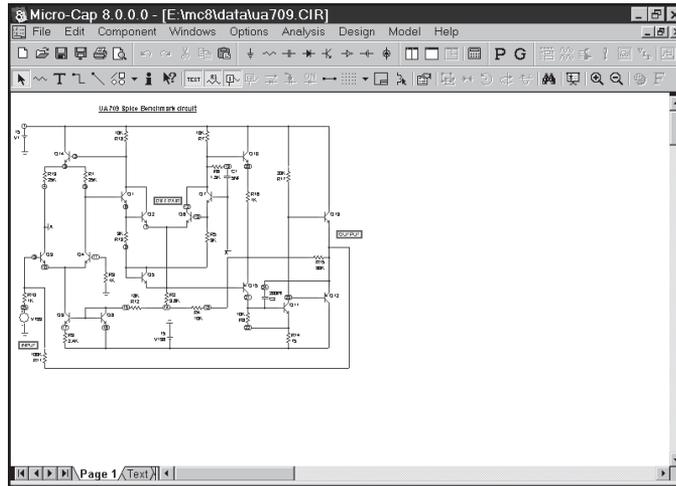


Figure 3-18 After clicking the Zoom-Out button

Simultaneously press the SHIFT key and click the right mouse button in the lower right, near the two transistors. This centers the schematic at the mouse position and redraws it at normal scale.

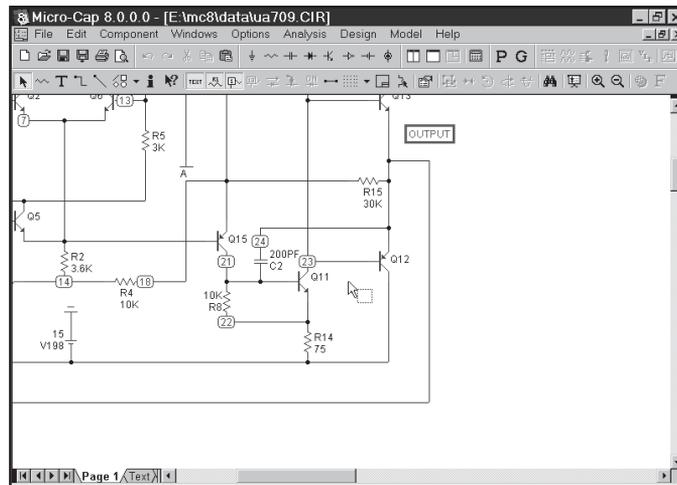


Figure 3-19 The UA709 after repositioning

Another SHIFT + click redraws the circuit at the low magnification scale. A final SHIFT + click near the upper left corner returns the schematic to 1:1 scale.

Panning uses the keyboard cursor keys or a right button mouse drag to move the schematic.

Panning is probably the easiest and most powerful method of moving around most schematics. To pan, drag anywhere on the schematic using the right mouse button. While holding the button down, drag the mouse and move the schematic. When the mouse is at or near a window edge, release the mouse button. Repeat the procedure until the schematic is in the desired position. Pan the UA709 circuit by clicking near the center and dragging left. This moves the schematic to the left, exposing more of the right-hand portion.

Finally, you can use flags to navigate the schematic. This method is most useful for very large schematics where panning might take many attempts to span the entire schematic. In this method you place flags in the schematic where you expect to want to visit in the future. Then simply picking the flag from the list centers the schematic view around the selected flag. To illustrate, click on the Go To Flag  button or select **Edit / Go To Flag**. Select the 'OUTPUT' flag from the list. This redraws the schematic, centered on the flag, as shown below.

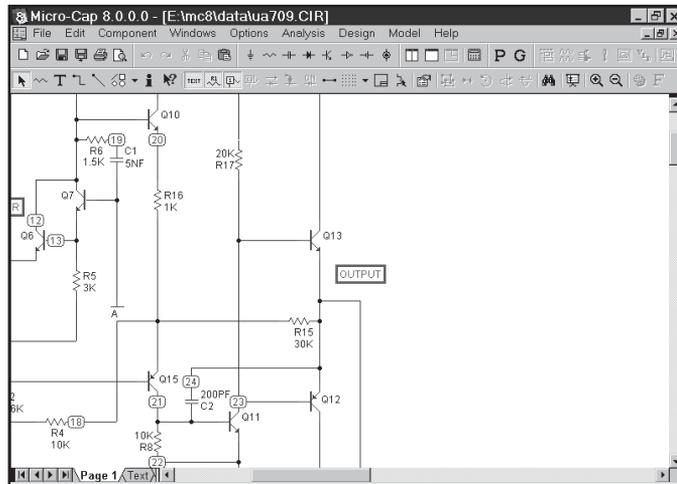


Figure 3-20 Navigating with flags

Close the UA709 file without saving it.

Creating and editing SPICE text files

MC8 can create and analyze SPICE text file circuit descriptions as well as schematics. SPICE text files may be created externally with a word processor or text editor, or internally using the MC8 text editor. To illustrate the procedure for creating a text file, select the **New** option from the **File** menu. This presents a dialog box where you specify the type of file to create. Click on the SPICE/Text option, then on the OK button. This opens a new text window and positions the text cursor at the upper left.

Type in the following:

```
CHOKE.CKT
V1 1 0 SIN (0 100 50)
V2 0 3 SIN (0 100 50)
D1 1 2 DIO
D2 3 2 DIO
L1 4 2 5
R1 0 2 10K
R2 0 4 10K
C1 0 4 2UF
.MODEL DIO D (IS=1E-14 CJO=10PF)
.TRAN .2m 0.05 0 .2m
.TEMP 27
.PLOT TRAN V(1) V(2) V(3) V(4) -150,200
.END
```

Save the circuit under the name CHOKE1.CKT using the **Save As** option from the **File** menu. Note that SPICE circuit file names do not require or assume an extension. There is no default extension. Any extension except "CIR" can be used. The "CIR" extension is reserved for schematic files. By convention, MC8 uses "CKT" for SPICE files, but any extension except "CIR" will do.

New SPICE/Text files are created in the same way as schematics, by using the **New** command from the **File** menu.

```

Micro-Cap 8.0.0.0 - [E:\mc8\data\choke.ckt]
File Edit Component Windows Options Analysis Design Model Help
CHOKE.CKT
V1 1 0 SIN (0 100 50)
V2 0 3 SIN (0 100 50)
D1 1 2 DIO
D2 3 2 DIO
D1 4 2 S
R1 0 2 10K
R2 0 4 10K
C1 0 4 2UF
.MODEL DIO D (IS=1E-14 CJO=10PF)
.TRAN .2m 0.05 0 .2m
.TEMP 27
.PLOT TRAN V(1) V(2) V(3) V(4) -150,200
.END

```

Figure 3-21 The sample SPICE circuit

Your circuit should look like Figure 3-21. Press ALT + 1 to run a transient analysis. When the Analysis Limits dialog box is displayed, press F2 to run the analysis. The results should look like Figure 3-22.

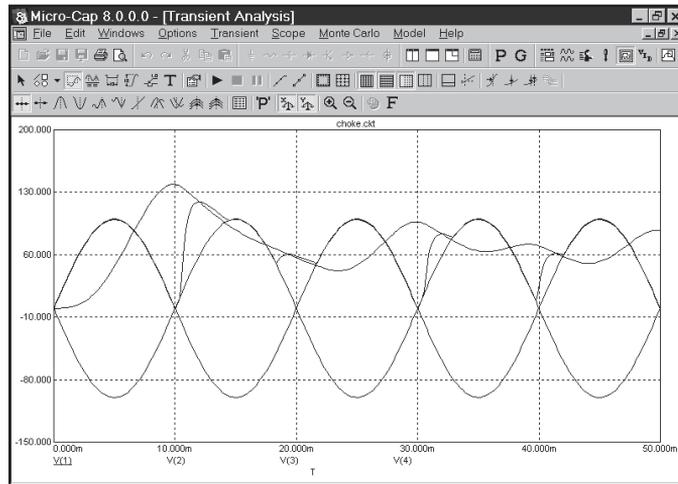


Figure 3-22 Transient analysis of the SPICE circuit

After the analysis the program automatically determines suitable scales if needed and plots the requested waveforms as specified in the .PLOT statement.

Exit the analysis by pressing F3, then close the file with CTRL + F4.

Summary

The most important ideas and methods covered in this chapter are:

- To open a new schematic use the **New** item on the **File** menu, or use the one automatically opened when the program first starts.
- Change the default component if it is different from the one you want to add. Default components are selected from the Component menu, a Component palette, or one of the part buttons in the Tool menu.
- To add a component, press CTRL + D or click the Component mode button. Select the component you want from the Component menu, User palette, or Tool bar button. Click the mouse on the schematic and drag it into place. If necessary, change its orientation with the right mouse button.
- Press CTRL + W or click the Wire item in the Tool bar to add wires.
- Components are connected with wires. To add a wire, click at one endpoint, drag the mouse to the other endpoint, then release it. Crossing wires do not connect.
- Press CTRL + T or click the Text button in the Tool bar to add text.
- Add text to nodes to name them for easy reference or use the node numbers. Nodes with the same name are connected.
- To edit, move, copy to the clipboard, or delete an object, it must first be selected. To select an object, press CTRL + E to enter the Select mode, then click the object. To select a region, drag the mouse over the region. Use SHIFT + click to deselect an object in a selected group.
- Selected objects are copied to the clipboard with CTRL + C and pasted to the cursor location with CTRL + V.
- CTRL + drag copies and renames selected objects.
- Navigating a schematic is done by scrolling, panning, changing the scale, positioning with SHIFT + click, using flags, and page scrolling.

What's in this chapter

Transient analysis is the topic of this chapter. Using the MIXED4 circuit, the capabilities of the time domain simulator are explored. The goal is to introduce and build a sound understanding of these basic subjects:

- Controlling the analysis through the Analysis Limits dialog box
- Selecting curves to plot or print
- State variables and initialization
- State Variables editor

The Transient Analysis Limits dialog box

Load the MIXED4 circuit file and select **Transient** from the **Analysis** menu. MC8 extracts the necessary circuit information directly from the schematic. More information is needed before the analysis can begin, and that information is supplied by the Analysis Limits dialog box.

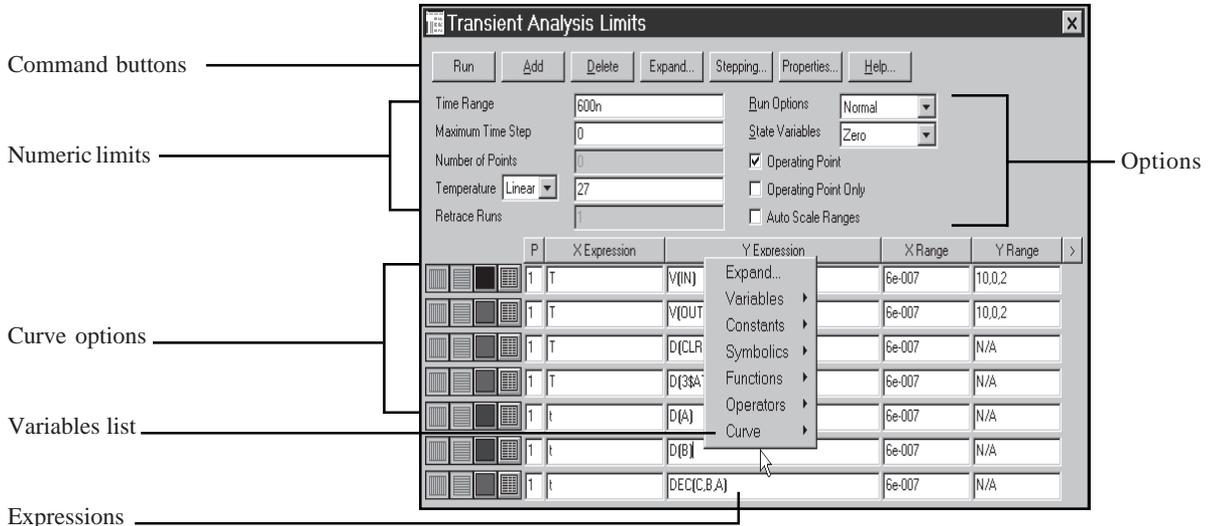


Figure 4-1 The Analysis Limits dialog box

The Analysis Limits dialog box has five main areas: Command buttons, Numeric limits, Curve options, Expression fields, and Options.

Command buttons

The Command buttons, located just above the Numeric limits field, contain seven commands.

Run: This command starts the analysis run. Clicking the Tool bar Run button  or pressing F2 will also start the run.

Add: This command adds another Curve options field and Expression field line after the line containing the cursor. The scroll bar to the right of the Expression field scrolls through the curves when there are more than can be displayed.

Delete: This command deletes the Curve option field and Expression field line where the text cursor is.

Expand: This command expands the text field where the text cursor is into a large dialog box for editing or viewing. To use the feature, click the mouse in the desired text field, and then click the Expand button.

Stepping: This command invokes the Stepping dialog box. Stepping is covered in a separate chapter.

Properties: This command invokes the Properties dialog box which lets you control the analysis plot window and the way curves are displayed.

Help: This command invokes the Help Screen. The Help system provides information by index and topic.

Numeric limits

The Numeric limits field provides control over the analysis time range, time step, number of printed points, and the temperature(s) to be used.

- **Time Range:** This field determines the start and stop time for the analysis. The format of the field is:

$\langle tmax \rangle [, \langle tmin \rangle]$

The run starts with time set equal to $\langle tmin \rangle$, which defaults to zero, and ends when time equals $\langle tmax \rangle$.

- **Maximum Time Step:** This field defines the maximum time step that the program is allowed to use. The default value, $(\langle tmax \rangle - \langle tmin \rangle) / 50$, is used when the entry is blank or 0.

- **Number of Points:** The contents of this field determine the number of printed values in the numeric output. The default value is 51. Note that this number is usually set to an odd value to produce an even print interval. The print interval is the time separation between successive printouts. The print interval used is $(\langle tmax \rangle - \langle tmin \rangle) / ([number\ of\ points] - 1)$.

- **Temperature:** This field specifies the global temperature(s) of the run(s) in degrees Celsius. This temperature is used for each device unless individual device temperatures are specified. If the Temperature list box shows Linear or Log the format is:

$\langle high \rangle [, \langle low \rangle [, \langle step \rangle]]$

The default value of $\langle low \rangle$ is $\langle high \rangle$, and the default value of $\langle step \rangle$ is

<high> - <low> (linear mode) or 1.0 (log mode). Temperature values start at <low> and are either incremented (linear mode) or multiplied (log mode) by <step> until <high> is reached.

If the Temperature list box shows List the format is:

<t1> [, <t2> [, <t3>] [,...]]

where t1, t2,.. are individual values of temperature.

One analysis is done at each specified temperature, producing one curve branch for each run.

Curve options

The Curve options field is located below the Numeric limits field and to the left of the Expressions field. Each curve option affects only the curve in its row. The options function as follows:

The first option toggles the X-axis between a linear  and a log  plot. Log plots require positive scale ranges.

The second option toggles the Y-axis between a linear  and a log  plot. Log plots require positive scale ranges.

The  option activates the color menu. There are 64 color choices for an individual curve. The button color is the curve color.

The  option prints a table of the curve's numeric values. The number of values printed is set by the Number of Points value. The table is printed to the Numeric Output window and saved in the file CIRCUITNAME.TNO.

A number from 1 to 9 in the (P) column places the curve into a numbered plot group. All curves with like numbers are placed in the same plot group. If the P column is blank, the curve is not plotted.

Expressions

The X Expression and Y Expression fields specify the horizontal (X) and vertical (Y) expressions. MC8 can evaluate and plot a wide variety of expressions for either scale. Usually these are single variables like T (time), V(10) (voltage at node 10), or D(OUT) (digital state of node OUT), but the expressions can be more elaborate like $V(2,3)*I(V1)*\sin(2*PI*1E6*T)$.

Variables list

Clicking the *right* mouse button in the Y expression field invokes the Variables list which lets you select variables, constants, functions, and operators, or expand the field to allow editing long expressions. Clicking the *right* mouse button in the other fields invokes a simpler menu showing suitable choices.

The X Range and Y Range fields specify the numeric scales to be used when plotting the X and Y expressions. The format is:

$\langle high \rangle$ [, $\langle low \rangle$] [, $\langle grid\ spacing \rangle$] [, $\langle bold\ grid\ spacing \rangle$]

$\langle low \rangle$ defaults to zero. [, $\langle grid\ spacing \rangle$] sets the spacing between grids. [, $\langle bold\ grid\ spacing \rangle$] sets the spacing between bold grids. Placing "AUTO" in the scale range calculates that individual range automatically. The Auto Scale Ranges option calculates scales for all ranges during the simulation run and updates the X and Y Range fields. The Auto Scale (F6) command immediately scales all curves, without changing the range values, letting you restore them with CTRL + HOME if desired. Note that $\langle grid\ spacing \rangle$ and $\langle bold\ grid\ spacing \rangle$ are used only on linear scales. Logarithmic scales use a natural grid spacing of 1/10 the major grid values and bold is not used. Auto Scale uses the number of grids specified in the **Properties dialog box (F10) / Scales and Formats / Auto/ Static Grids** field.

Transient options

These options include the following:

- **Run Options**

- **Normal:** This runs the simulation without saving it.
- **Save:** This runs the simulation and saves it to disk, using the same format as in Probe. The file name is CIRCUITNAME.TSA.
- **Retrieve:** This loads a previously saved simulation and plots and prints it as if it were a new run. The file name is CIRCUITNAME.TSA.

- **State Variables**

These options determine the state variables at the start of the next run.

- **Zero:** This sets the state variable initial values (node voltages, inductor currents, digital states) to zero or X.
- **Read:** This reads a previously saved set of state variables and uses

them as the initial values for the run.

- **Leave:** This leaves the current values of state variables alone. They retain their last values. If this is the first run, they are zero. If you have just run an analysis without returning to the Schematic editor, they are the values at the end of the run. If the run was an operating point only run, the values are the DC operating point.
- **Retrace:** This runs the analysis N times, where N is the number in the Retrace Runs field. For the first run, normal initialization is done and, if requested, the operating point is calculated. Initial conditions are retained for subsequent runs, whether invoked manually with F2 or automatically by using a number greater than 1 in the Retrace Runs field.
- **Operating Point:** This calculates a DC operating point. It uses the initial state variables as a starting point and calculates a new set that represents the DC steady state response of the circuit to the T=0 values of all sources.
- **Operating Point Only:** This calculates a DC operating point only. No transient run is made. The state variables are left with their final operating point values.
- **Auto Scale Ranges:** This sets the X and Y range to AUTO for each new analysis run. If it is not enabled, the existing scale values from the X and Y Range fields are used.

Resizing

- **Field size adjustment:** You can adjust the field sizes by dragging the lines that separate the column titles (e.g. X Expression, Y Expression, etc.).
- **Auto-size:** Clicking the Auto-size button automatically adjusts the field sizes to match existing expression lengths.
- **Grow:** Drag on the Grow icon to increase the size of the dialog box.

The Run, State Variables, and Analysis options affect the simulation results. To see the effect of changes of these options you must do a run by clicking on the Run command button or pressing F2.

Selecting curves to plot or print

Note: MC8 initially selects the voltage curves of the first two named nodes for plotting.

MC8 automatically creates a set of analysis limits for newly created circuits. It tries to guess what curves you may want to look at and what plot and analysis options you may want. Usually, it is necessary to change these initial guesses. Changes you make to any entry in the Analysis Limits dialog box are saved with the circuit file, so you don't need to enter these each time you run.

Click on the Run button to do the transient analysis of the MIXED4 circuit.

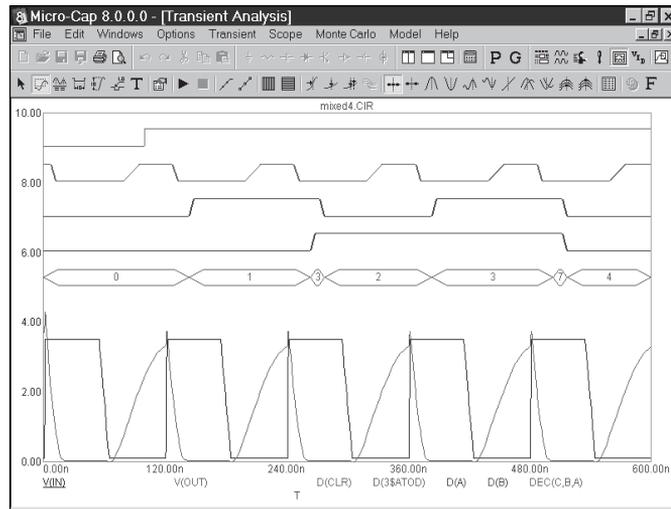


Figure 4-2 Transient analysis of MIXED4

The top five curves are digital and show the CLR input, clock input, the state outputs of the first two flip-flops, and the decimal value of the three flip-flop outputs. The next two curves are plots of the voltage at the input pulse source and the output of the analog section. In general, V(A) plots the voltage on an analog node A. D(B) plots the state of a digital node B.

Plotting other curves

You can plot expressions using a variety of variables such as node voltage or device terminal voltage or current. To plot different curves, you can add new curve fields or you can change existing ones. To illustrate, press F9 to display the Analysis Limits dialog box. Click in any column of the last row. Click the Add button in the upper part of the dialog box. This adds a new curve row and moves the text cursor to the P column of the new row. It also duplicates the contents of all fields of the row above. Click the Delete button. This deletes the curve just added. Click the Add button to add the curve row back. Since we want the new

curves to be placed on a separate plot group, change the '1' in the P column to '2'. Press the Tab key twice or click the mouse in the Y Expression field. Type in the following expression:

$$\text{"IC(Q4)*VCE(Q4)"}$$

Press the Tab key twice to move the cursor to the Y Range field and type "Auto". This tells the program to determine the Y scale range automatically, after the run. During the run, MC8 uses a default scale to start the plot. Several times during the run it will revise the scales based upon the curves. When the run is complete, MC8 determines the scale necessary to show the complete curves, plots it, and enters the new scales into the X and Y range fields.

Use the Variables list to select hard-to-remember function and variable names.

If you can't remember the variable or function name you want, use the Variables list as a memory aid. To illustrate, click the Add button again. Click in the new Y Expression field with the *left* mouse button. Type CTRL + A to select the entire Y expression. Click on any part of the selected Y expression with the *right* mouse button. This invokes the Variables list, a small pop-up menu that lets you select variables, functions, and operators. It also lets you expand the limited text editing area to a full size text edit dialog box. This is handy for complex expressions with many parentheses. Click on **Variables** and then on **Device Currents**. This displays a dialog box with a list of the available device currents in the circuit. Click on I(VCC), then on OK. The text "I(VCC)" is entered in the Y Expression field and the cursor is placed to the right of the text. The Y expression will now look like this:

$$\text{I(VCC)}$$

Type "*". Click the *right* mouse button and select **Variables / Device Voltages / V(VCC)**. Click OK. The Y expression should now look like this:

$$\text{I(VCC)*V(VCC)}$$

Press the Home key to move the cursor to the start of the field. Click the *right* mouse button and click **Functions / Calculus / SUM**. SUM is the integration operator. It integrates the expression to the right of the text cursor, with respect to the X expression variable, T. Press the End key to put the cursor at the end of the field and type "/T". This produces an expression for the average power supplied by the VCC battery. The Y expression should now look like this:

$$\text{SUM(I(VCC)*V(VCC),T)/T}$$

Since the new curve was created with Auto MC8 will determine the actual range after the run and calculate a suitable Y scale. Because there are two curves sharing the same plot group, it can do this two ways; 1) by finding a common scale large enough to show all of both graphs, or 2) by finding an individual scale for each curve and printing two Y scales. The choice is determined by the **Same Y Scales For Each Plot Group** option on the **Scope** menu. If this option is checked, then all curves in each plot group will share a common scale large enough to plot all parts of each curve. This is true regardless of whether auto or user-specified scales are used. If you select this option, MC8 draws a single scale using the largest of the specified or calculated scales. Click the Run button. Here is how the new run looks when the option is disabled.

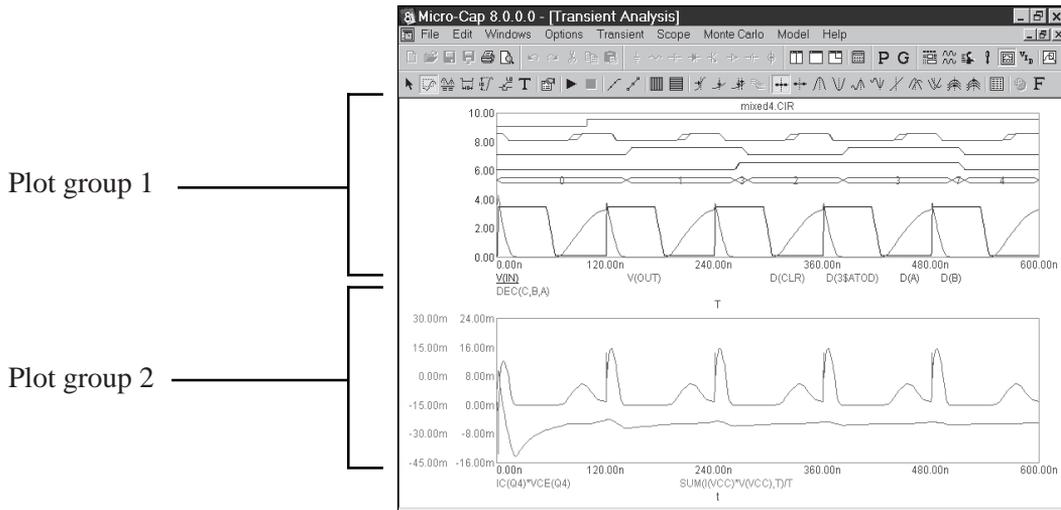


Figure 4-3 Transient analysis with two plot groups

Group 1 contains the original curves. Group 2 contains the new curves: the instantaneous collector power of the output transistor Q4, and the average power supplied by the VCC battery. Plot grouping is specified by the P column number. All curves with the same number are plotted in the same group.

Analog and digital curves may be grouped together or separately. When grouped together, digital curves are plotted in the same order they occur in the dialog box, starting at the top of the graph. Analog curves are plotted using the range scales, which only apply to analog curves. This can cause the curves to overlap. Overlap can be minimized by compressing the analog curves with a large Y Range value. Separating analog and digital curves into groups avoids overlap but is less convenient when comparing the two sets of curves at the same time value.

Same Y Scales

To see how the plot would look with common Y scales, enable the **Same Y Scales For Each Plot Group** option on the **Scope** menu. The plot looks like this:



Figure 4-4 The plot with the Same Scales option enabled

The two scales have been collapsed into a common scale, containing the largest and smallest scale values from each.

Tokens Option

While color identifies individual curves, black and white printers create plots where it is hard to tell the curves apart. Tokens can help identify the different curves. To illustrate, click on the Tokens button .

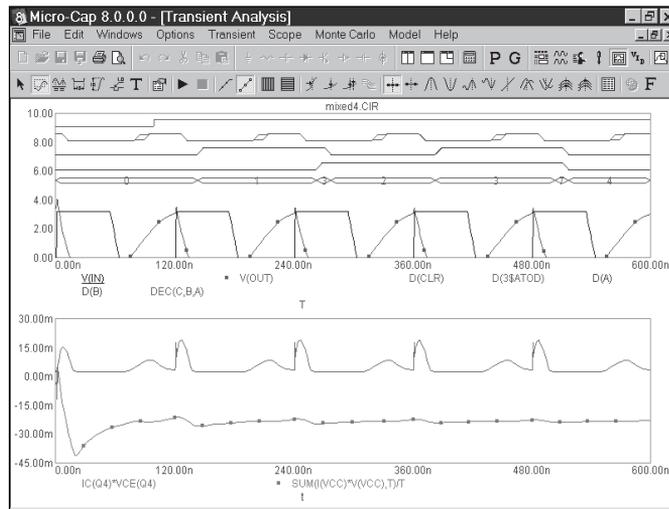


Figure 4-5 The plot with tokens

The Plot Properties dialog box

Curve display and format can be changed with the Plot Properties dialog box. Press F10 to invoke the dialog box for the plot. It looks like this:

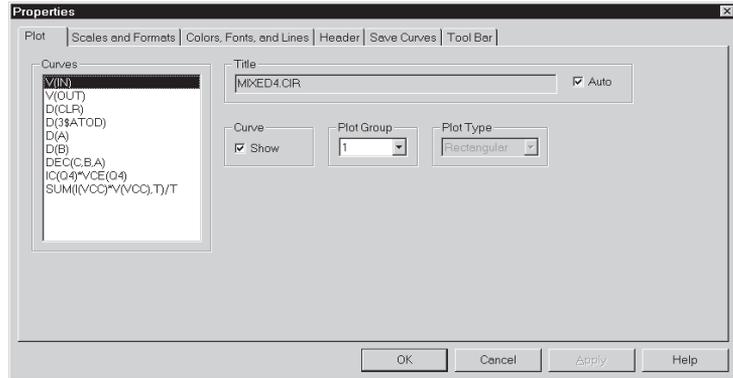


Figure 4-6 The Properties dialog box

From the **Curve Line** section of the **Colors, Fonts, and Lines** panel, select V(OUT) and a width of 3. Click on V(IN) and select the third pattern in the list. Click the OK button. Disable tokens by clicking the  Tokens button in the Tool Bar. The screen should look like this:

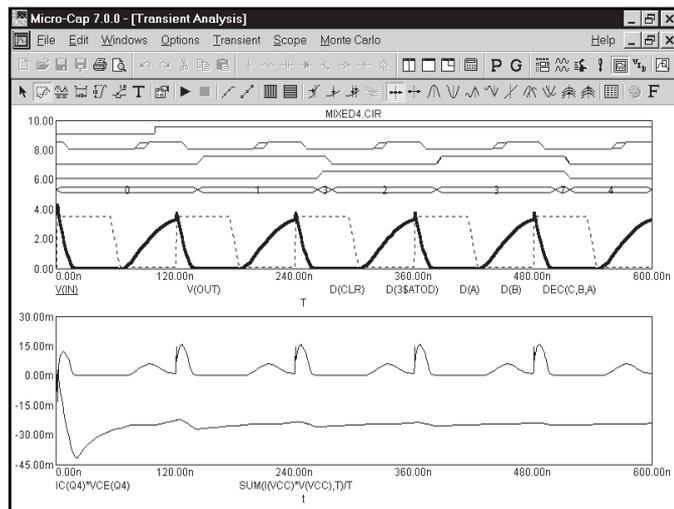


Figure 4-7 Using line width and pattern

Let's experiment with some of the other options. Click on the Data Points button . This toggles the display of marker icons for the calculated data points. Since the option was initially disabled, clicking the button enables it, and the data point marker icons are added producing a plot that looks like this:

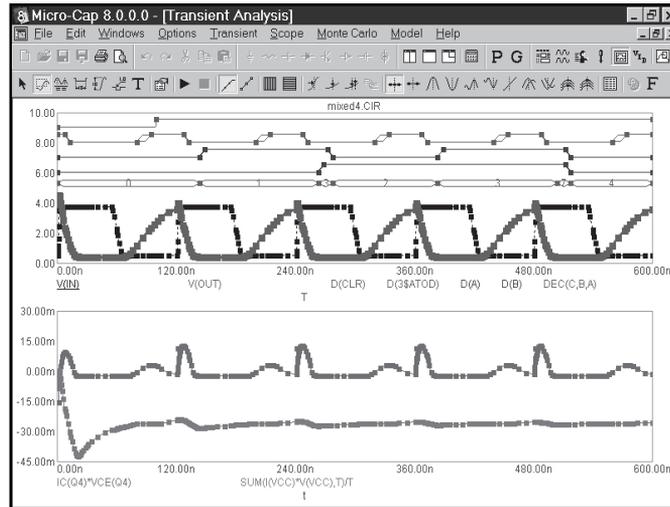


Figure 4-8 The plot with the Data Points option enabled

Since the simulator uses a variable time step algorithm to minimize errors and accelerate the run, the actual time points will not be regularly spaced. The data point markers are placed at the actual computed time points. There are at least two situations where you might want to see these:

Inadequate curve sampling: This can happen in transient analysis when the simulator flies past a switch threshold. It may also happen that you want more data points to "fill in" source curves that look choppy. Even though the charge error is strictly controlled by the internal truncation error routines, the data sampling may seem inadequate. Data point icons show where the data points are and whether more will help. In AC analysis, your plot may miss a steep spike on a narrow pass or reject band. Data point icons show whether the simulator actually made a calculation within the band.

Identifying real curve spikes: Sometimes you see spikes on curves. If the curve changes direction from one data point to the next, it's likely the spikes are simply artifacts of the integration routines. The only way to tell is to mark the data points. If they are integration artifacts, you can reduce them by decreasing RELTOL or the minimum time step or both.

Experimenting
with the options

To illustrate some other options, click on the horizontal axis grid  and vertical axis grid  buttons. The plot should now look like this:

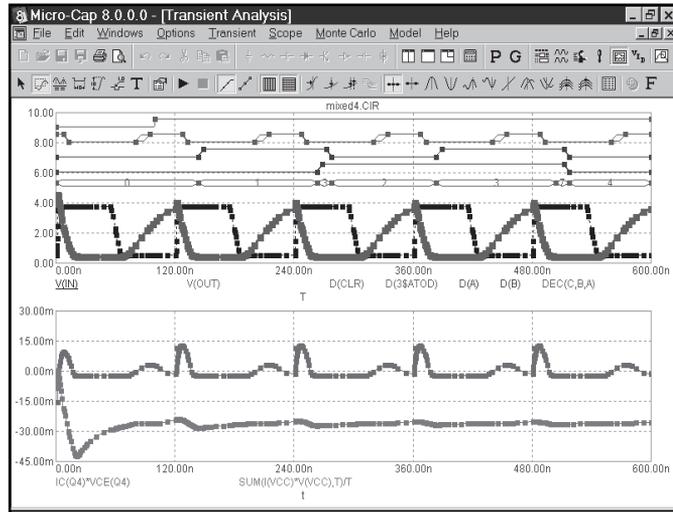


Figure 4-9 Adding grids to the plot

Press F9, then click on the Operating Point and Operating Point Only buttons. This tells the program to do an operating point calculation, and then stop. An operating point calculation sets all sources to their defined initial values, removes all capacitors, and replaces all inductors with short circuits. It then calculates the steady-state DC values of the state variables (node voltages, inductor currents and digital states) at the initial time point. This set of values is referred to as the DC operating point. Press F2 to do the operating point only run. Since we're doing only an operating point, no plot will appear. Exit the analysis by pressing F3. Click on the Node Voltage  button and MC8 displays the last calculated states and voltages. In this case, since we just completed an operating point only calculation, these are the operating point values.

State variables and initialization

State variables define the state or condition of the mathematical system that represents the circuit at any instant. These variables must be initialized to some value prior to starting the analysis run. Here is how MC8 does the initialization:

Setup initialization:

When you first select a transient, AC, or DC analysis, all state variables are set to zero and all digital levels to X. This is called the *setup initialization*.

Run initialization:

Each new run evokes the *run initialization* based upon the State Variables option from the Analysis Limits dialog box. This includes every run, whether initiated by pressing F2, clicking on the Run button, stepping parameters, using Monte Carlo, or stepping temperature. There are three choices:

Zero: The analog state variables, node voltages, and inductor currents are set to 0. Digital levels are set to X, or in the case of flip-flop Q and QB outputs, set to 0, 1, or X depending upon the value of DIGINITSTATE. This value is defined in the Global Settings dialog box. This is the only option in DC analysis.

Read: MC8 reads the variables from the file CIRCUITNAME.TOP. The file itself is created by the State Variables editor Write command.

Leave: MC8 does nothing to the initial state variables. It simply leaves them alone. There are three possibilities:

First run: If the variables have not been edited with the State Variables editor, they still retain the setup initialization values.

Later run: If the variables have not been edited with the State Variables editor, they retain the ending values from the last run.

Edited: If the variables have been edited with the State Variables editor, they are the values shown in the editor.

Retrace: MC8 does nothing to the initial state variables. It simply leaves them alone.

Device initialization:

Device initialization is the third step. It is always done for the first run and for subsequent runs if Retrace is disabled.

After the State Variables option has been processed, .IC statements are processed. Device IC statements, such as those for inductor current and capacitor voltage, override .IC statements if they are in conflict.

Note that .IC statements specify values that persist throughout the initial bias point calculation.

They are more resilient than simple initial values which can (and usually do) change after the first iteration of the bias point. This may be good or bad depending upon what you are trying to achieve.

Using these initial values, an optional operating point calculation may be done and the state variables may change. If no operating point is done, the state variables are left unchanged from the initialization procedure. If an operating point only is done, the ending state variable values are equal to the operating point values.

The transient analysis begins after the setup, run, and device initializations are complete.

The State Variables editor

To illustrate the State Variables editor load the circuit, MIXED1. It looks like this:

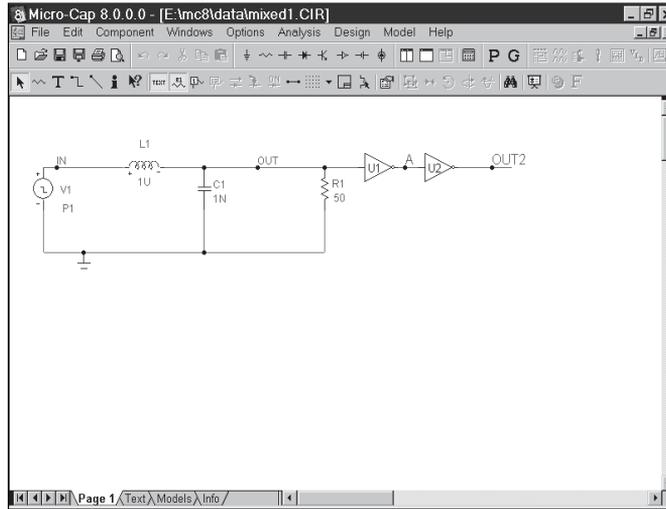


Figure 4-10 The MIXED1 circuit

This circuit contains an analog section driving a digital section. Select **Transient** from the **Analysis** menu. The analysis limits look like this:

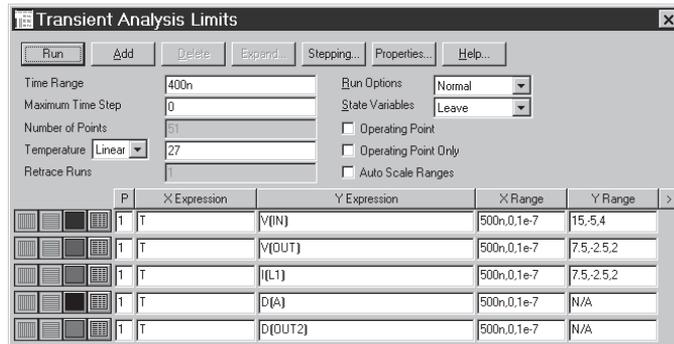


Figure 4-11 MIXED1 analysis limits

Press F2 to start the run. The results look like this:

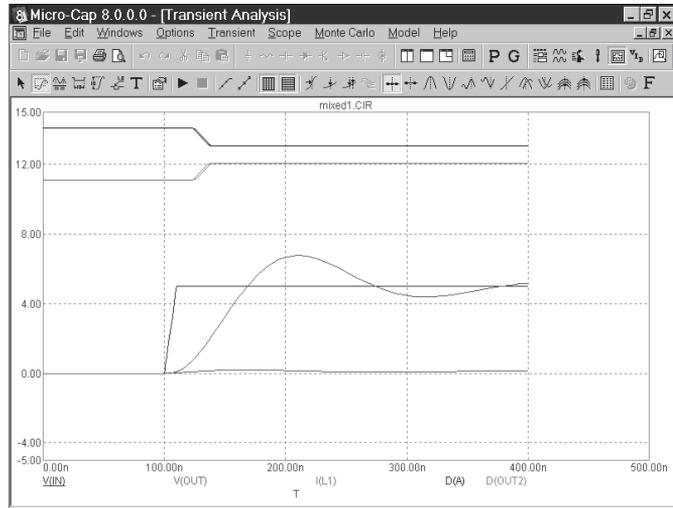


Figure 4-12 The transient analysis run

Select the **State Variables Editor** from the **Transient** menu. It looks like this:

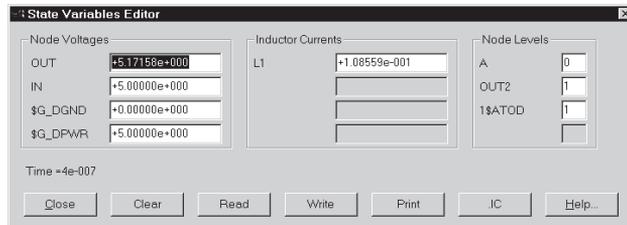


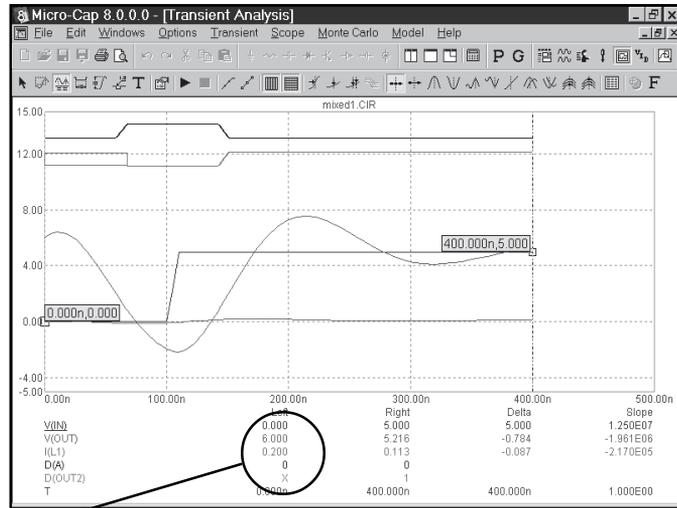
Figure 4-13 The State Variables editor

This editor lets you view and edit state variable values. State variables consist of node voltages, shown in the left column, inductor currents, shown in the middle column, and digital node levels, shown in the right column. Node voltages and digital node levels are identified by the node name, if one exists, or the node number otherwise. Inductors are identified by the part name.

The display shows the current values of the state variables. Since we've just finished a transient analysis these values are from the last data point of the run.

Let's experiment with these values. Double-click the mouse in the L1 inductor field and type ".2". This sets the inductor current to 0.2 amps. Double-click in the OUT voltage field and type "6". This sets the OUT node to 6.0 volts. Double-click

in the OUT2 digital node level field and type "X". This sets the OUT2 node level to X. Click on the Close button. The Run Options from the Analysis Limits dialog box are set to skip the operating point and to leave the state variables when we start the run. These choices tell the program to use the current state variables values, which we have just edited, for the initial values. Since we have requested no operating point, the initial point on the plot should reflect the values in the State Variables editor. Press F2 to start the run. When the run is complete, press F8 to place the display in the Cursor mode. The results look like this:



Initial values

Figure 4-14 The new run in Cursor mode

Notice that the initial value of V(OUT) (shown under the Left cursor column), is the 6.0 volts we specified in the State Variables editor. Similarly, the initial value of the inductor current is 0.2 and the initial value of the digital level of node OUT2 is X.

Exit the analysis with F3 and close the circuit file with CTRL + F4.

What's in this chapter

AC analysis is the topic for this tutorial. In particular, these subjects are covered:

- AC analysis
- The AC Analysis Limits dialog box
- Experimenting with the options and limits
- Selecting curves to plot or print
- Picking plot options
- Numeric output
- Input and output noise plots
- Nyquist plots
- Smith charts
- Polar plots

What happens in AC analysis

What happens during an AC analysis? First, an optional DC operating point is calculated. Digital nodes stabilize at an operating point state and potentially affect analog nodes to which they may be attached. Linearized, small signal AC models are then created for each component and integrated into a set of linear network equations. These are repeatedly solved at many frequency points to obtain the small signal response of the circuit to whatever AC excitation is present in the circuit. The linearized AC model of a digital component is an open circuit. This means that digital parts are ignored during small signal analysis.

The excitation is supplied by one or more independent curve sources in the circuit. Pulse and Sine sources provide a fixed, real, 1.0 volt AC signal. User sources provide a signal comprised of the real and imaginary parts specified in their files. The SPICE independent sources, V and I, provide user-specified real AC signal amplitudes. Function sources can create an AC signal only if they have a `FREQ` expression. These are the only sources that generate AC excitation.

AC analysis represents the excitation and system variables (node voltages and various branch currents) as complex quantities. Operators, such as `RE` (real), `IM` (imaginary), `dB` ($20 * \text{Log}()$), `MAG` (magnitude), `PH` (phase), and `GD` (group delay) are provided to print and plot these complex quantities.

To explore AC analysis, load the file `DIFFAMP`. It looks like this:

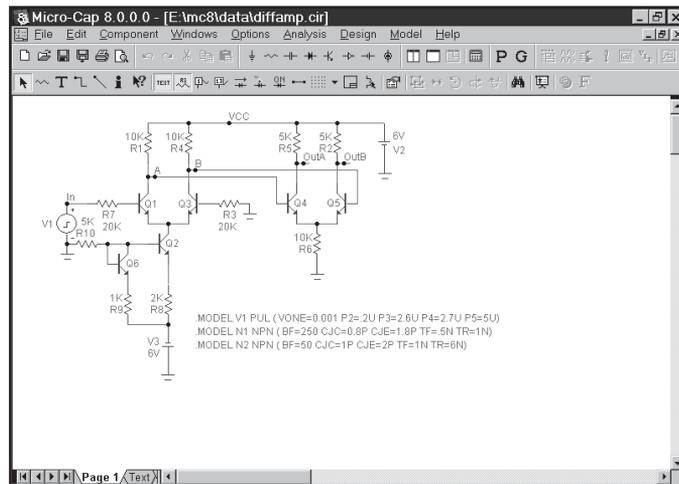


Figure 5-1 The DIFFAMP circuit

Choose **AC** from the **Analysis** menu. When the AC Analysis Limits dialog box comes up, the fields should look like this:

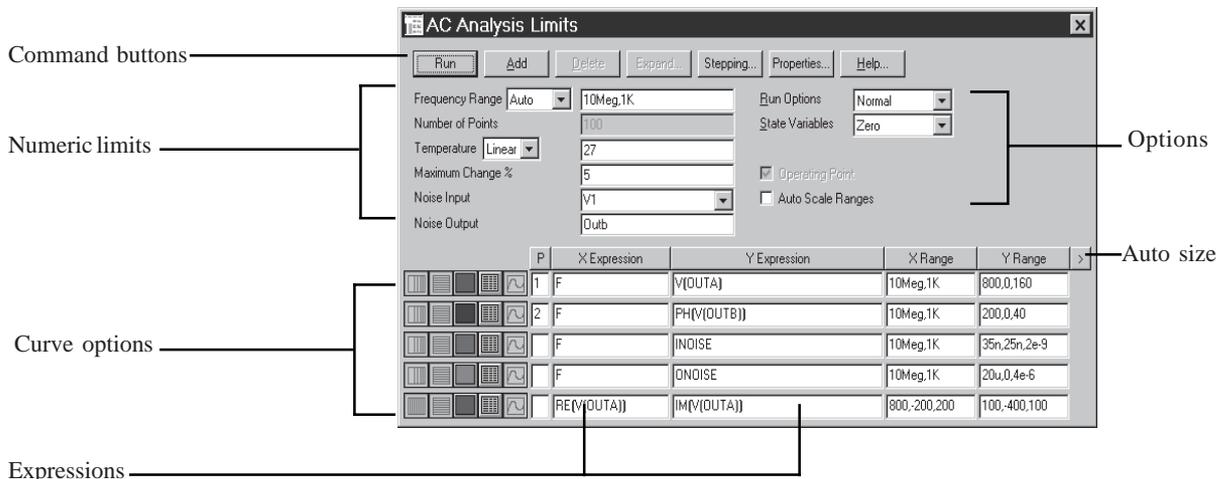


Figure 5-2 The AC Analysis Limits dialog box

In this circuit, we have a single source providing a 1.0 volt AC signal. The dialog box instructs MC8 to sweep the frequency from 1K to 10Meg and produce plots of the magnitude of the voltage at the node OUTA and the phase angle of the voltage at node OUTB. Press F2 and the result looks like this:

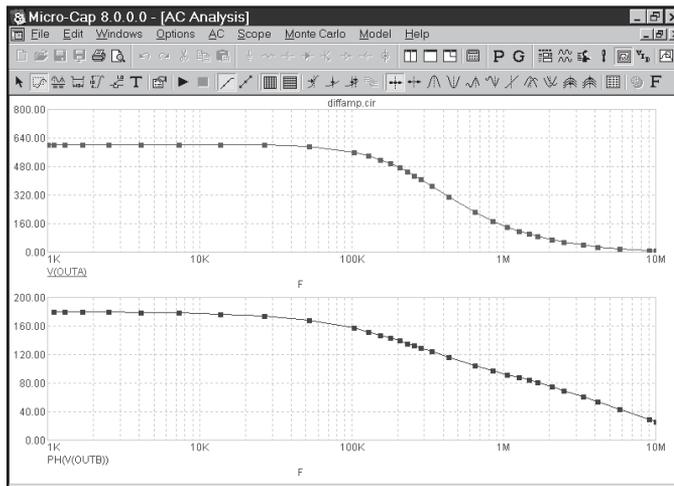


Figure 5-3 The AC analysis plot

The AC Analysis Limits dialog box

The Analysis Limits dialog box is divided into five principal areas: the Command buttons, Numeric limits, Curve options, Expressions, and Options.

The command buttons are located just above the Numeric limits.

Run: This command starts the analysis run. Clicking the Tool bar Run  button or pressing F2 will also start the run.

Add: This command adds another Curve options field and Expression field line after the line containing the cursor. The scroll bar to the right of the Expression field scrolls the curve rows when needed.

Delete: This command deletes the Curve option field and Expression field line where the text cursor is.

Expand: This command expands the working area for the text field where the text cursor currently is. A dialog box is provided for editing or viewing. To use the feature, click in an expression field, then click the Expand button.

Stepping: This command calls up the Stepping dialog box. Stepping is reviewed in a separate chapter.

Properties: This command invokes the Properties dialog box which lets you control the analysis plot window and the way curves are displayed. See the Plot Properties dialog box article in Chapter 4, "Transient Analysis".

Help: This command calls up the Help Screen. The Help System provides information by index and topic.

The definition of each item in the Numeric limits field is as follows:

- **Frequency step list box**

- **Auto:** This method uses the first plot of the first group as a pilot plot. If, from one frequency point to another, the plot has a vertical change of greater than [maximum change %] of full scale, the frequency step is reduced, otherwise it is increased. [maximum change %] is the value from the fourth numeric field of the AC Analysis Limits dialog box. Auto is the standard

method because it uses the fewest number of data points to produce the smoothest plot.

- **Linear:** This method uses a frequency step that produces horizontally equidistant data points on a linear scale. The number of data points is controlled by the Number of Points field.
- **Log:** This method uses a frequency step that produces horizontally equidistant data points on a log scale. The number of data points is controlled by the Number of Points field.
- **List:** This method uses a comma-delimited list of frequency points from the Frequency Range, as in 1E8, 1E7, 5E6.
- **Frequency Range:** This field controls the frequency range for the analysis. The syntax is *<Highest Frequency>* [, *<Lowest Frequency>*]. If *<Lowest Frequency>* is unspecified, the program calculates a single data point at *<Highest Frequency>*.

• **Number of Points:** This determines the number of data points printed in the Numeric Output window. It also determines the number of data points actually calculated if Linear or Log stepping is used. If the Auto Step method is selected, the number of points actually calculated is controlled by the *<Maximum change %>* value. If Auto is selected, interpolation is used to produce the specified number of points. The default value is 51. This number is usually set to an odd value to produce an even print interval.

For the Linear method, the frequency step and the print interval are:

$$(\textit{<Highest Frequency>} - \textit{<Lowest Frequency>}) / (\textit{<Number of points>} - 1)$$

For the Log method, the frequency step and the print interval are:

$$(\textit{<Highest Frequency>} / \textit{<Lowest Frequency>})^{1/(\textit{<Number of points>} - 1)}$$

• **Temperature:** This field specifies the global temperature(s) of the run(s) in degrees Celsius. This temperature is used for each device unless individual device temperatures are specified. If the Temperature list box shows Linear or Log the format is:

$$\textit{<high>} [, \textit{<low>} [, \textit{<step>}]]$$

The default value of $\langle low \rangle$ is $\langle high \rangle$, and the default value of $\langle step \rangle$ is $\langle high \rangle - \langle low \rangle$ (linear mode) or 1.0 (log mode). Temperature values start at $\langle low \rangle$ and are either incremented (linear mode) or multiplied (log mode) by $\langle step \rangle$ until $\langle high \rangle$ is reached.

If the Temperature list box shows List the format is:

$\langle t1 \rangle [, \langle t2 \rangle [, \langle t3 \rangle] [, \dots]]$

where $t1, t2, \dots$ are individual values of temperature.

- **Maximum Change %:** This value controls the frequency step used when Auto is selected for the Frequency Step method.
- **Noise Input:** This is the name of the input source to be used for noise calculations. If the INOISE and ONOISE variables are not used in the expression fields, this field is ignored.
- **Noise Output:** This field holds the name(s) or number(s) of the output node(s) to be used for noise calculations. If the INOISE and ONOISE variables are not used in the expression fields, this field is ignored.

The Curve options are located below the Numeric limits and to the left of the Expressions. Each curve option affects only the curve in its row. The options function as follows:

The first option toggles the X-axis between a linear  and a log  plot.

The second option toggles the Y-axis between a linear  and a log  plot. The  option activates the color menu. There are 64 color choices for an individual curve. The button color is the curve color.

The  option prints a table showing the numeric value of the curve. The values printed are those actually calculated. The table is printed to the Numeric Output window and saved in the file CIRCUITNAME.ANO.

The  option selects the basic plot type. In AC analysis there are three types available,  rectangular,  polar, and  Smith.

A number from 1 to 9 in the P column is used to group the curves into different plot groups. All curves with like numbers are placed in the same plot group. If the P column is blank, the curve is not plotted.

The Expressions field specifies the horizontal (X) and vertical (Y) scale ranges and expressions. The expressions are treated as complex quantities. Some common expressions are F (frequency), db(v(1)) (voltage in decibels at node 1), and re(v(1)) (real voltage at node 1). *Note that while the expressions are evaluated as complex quantities, only the magnitude of the Y expression vs. the magnitude of the X expression is plotted.* If you plot the expression V(3)/V(2), MC8 evaluates the expression as a complex quantity, then plots the magnitude of the final result. It is not possible to plot a complex quantity directly vs. frequency. You can plot the imaginary part of an expression vs. its real part (a Nyquist plot), or you can plot the magnitude, real, or imaginary parts vs. frequency (a Bode plot).

The scale ranges specify the scales to use when plotting the X and Y expressions. The range format is:

<high> [,*<low>*] [,*<grid spacing>*] [,*<bold grid spacing>*]

<low> defaults to zero. [,*<grid spacing>*] sets the spacing between grids. [,*<bold grid spacing>*] sets the spacing between bold grids. Placing "AUTO" in the X or Y scale range calculates its range automatically. The Auto Scale Ranges option calculates scales for all ranges during the simulation run and updates the X and Y Range fields. The Auto Scale (F6) command immediately scales all curves, without changing the range values, letting you restore them with CTRL + HOME if desired. Note that *<grid spacing>* and *<bold grid spacing>* are used only on linear scales. Logarithmic scales use a natural grid spacing of 1/10 the major grid values and bold is not used. Auto Scale uses the number of grids specified in the **Properties dialog box (F10) / Scales and Formats / Auto/Static Grids** field.

Clicking the *right* mouse button in the Y expression field invokes the Variables list. It lets you select variables, constants, functions, and operators, or expand the field to allow editing long expressions. Clicking the *right* mouse button in the other fields invokes a simpler menu showing suitable choices.

• Run Options

- **Normal:** This runs the simulation without saving it to disk.

- **Save:** This runs the simulation and saves it to disk in the Probe format.
- **Retrieve:** This loads a previously saved simulation and plots and prints it as if it were a new run.
- **State Variables:** These options determine what happens to the time domain state variables (DC voltages, currents, and digital states) prior to the optional operating point.
 - **Zero:** This sets the state variable initial values (node voltages, inductor currents, digital states) to zero or X. This option forces an operating point.
 - **Read:** This reads a previously saved set of state variables and uses them as the initial values for the run. Normally the values read in are from a prior operating point run in AC or transient analysis, and are the desired bias conditions about which to linearize and do the AC analysis. If so, you would want to disable the Operating Point option when you use this Read feature.
 - **Leave:** This leaves the current values of state variables alone. They retain their last values. If this is the first run, they are zero. If you have just run an analysis without returning to the Schematic editor, they are the values from that run. This option does not force an operating point.
- **Operating Point:** This calculates a DC operating point, changing the state variables as a result. If an operating point is not done, the time domain variables are those resulting from the initialization step (zero, leave, or read). The linearization for AC analysis is done using the state variables after this optional operating point. In a nonlinear circuit where no operating point is done, the validity of the small-signal analysis depends upon the accuracy of the state variables read from disk, edited manually, or enforced by device initial values or .IC statements. Calculating an operating point may (and usually does) change the initial state variables.
- **Auto Scale Ranges:** This sets all ranges to Auto every time a simulation is run. If disabled, the values from the range fields are used.

All options affect simulation results. To see their effect, you must do a run by clicking on the Run command button.

Experimenting with options and limits

The small squares on the plots in Figure 5-3 mark the actual points calculated by MC8. We can produce a smoother plot at the expense of more data points by reducing the *Maximum change %* value from 5% to 1%. Press F9 to display the AC Analysis Limits dialog box. Double-click the mouse in the Maximum Change % field and type "1". Press F2 to run the analysis. It looks like this:

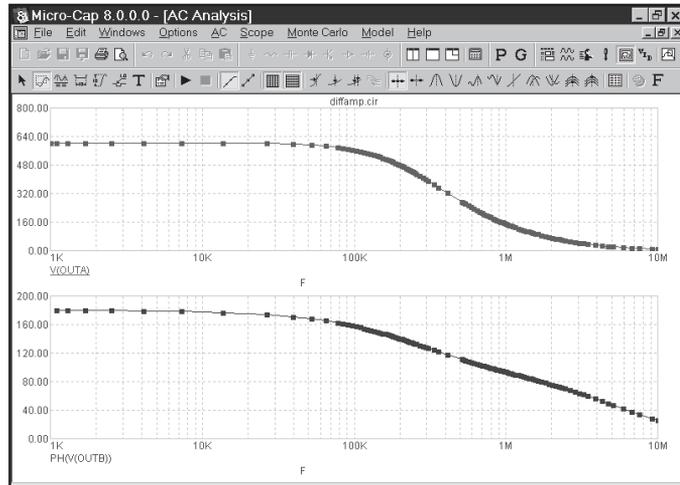


Figure 5-4 The plot with *Maximum change %* set to 1%

The lower value of 1% produces a smoother plot by calculating more data points. The current Frequency Step method is Auto. In this mode, the frequency step is automatically adjusted to restrain the point-to-point change of both the vertical value and the angle of the plot itself to less than the specified maximum change. The Auto method produces the smoothest plot at the lowest computation price for most curves.

This analysis uses 1% and generates about three times the number of data points as the 5% analysis. Since the analysis time increases in direct proportion to the number of data points, it is wise to keep the *Maximum change %* as large as possible. A value of 1 to 5 is usually suitable. If the first curve is very flat, you may want to try a fixed frequency step method.

Let's try one of these methods. Press F9 to display the Analysis Limits dialog box. Click in the Frequency Range list box. When the list drops down, click on the Log option. Press F2 to start the run. The result looks like Figure 5-5.

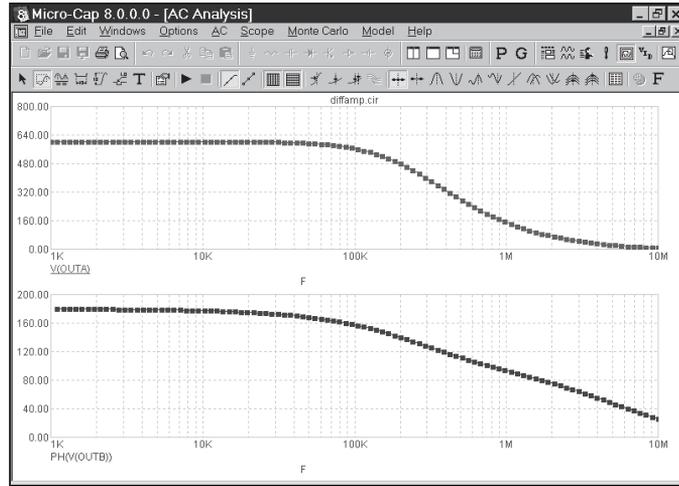


Figure 5-5 The plot using the log option

The constant horizontal distance between data points is achieved by making each new frequency a product of the prior frequency and a fixed multiplier. Since the lowest frequency forms the base that is multiplied, it must not be zero.

The Linear option is only used when the X axis is linear. To illustrate, press F9 and edit the Frequency Range field to 1E7,0. Change the Frequency Range method to Linear and change the X Log/Linear button states of the first two curves to linear. Press F2 to run. The result looks like Figure 5-6.

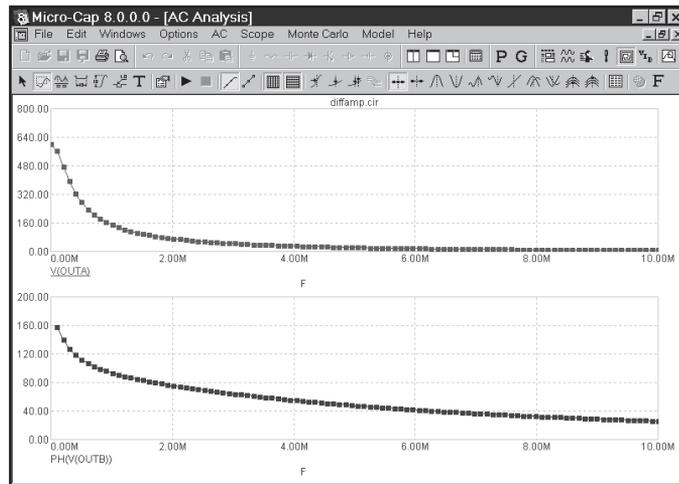
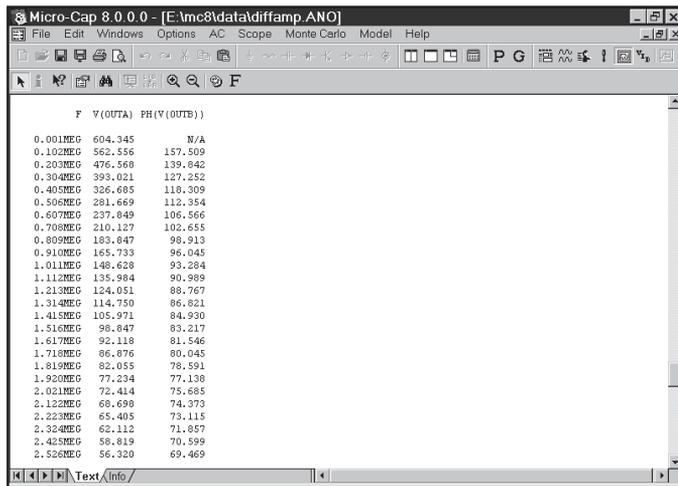


Figure 5-6 The plot using the linear option

Numeric output

To obtain a tabulation of the numerical values simply click the Numeric Output button  for each curve for which numeric output is desired. Do that now for the first two curves. Press F9, then click the buttons. Press F2 to run the analysis. All text or numeric output is directed to the Numeric Output window and to a disk file with the name DIFFAMP.ANO. Press F5 to see the window. The first part shows the results of the DC operating point. Scroll down to see the numeric values for the curves. The display should look like this:



F	V(OUTA)	PH(V(OUTS))
0.001MEG	604.345	N/A
0.102MEG	562.556	157.509
0.203MEG	476.568	139.842
0.304MEG	393.021	127.252
0.405MEG	326.685	118.309
0.506MEG	281.669	112.354
0.607MEG	237.849	106.566
0.708MEG	210.127	102.555
0.809MEG	183.847	98.913
0.910MEG	165.733	96.045
1.011MEG	148.628	93.284
1.112MEG	135.984	90.989
1.213MEG	124.051	88.767
1.314MEG	114.750	86.821
1.415MEG	105.971	84.930
1.516MEG	98.847	83.217
1.617MEG	92.119	81.546
1.718MEG	86.876	80.045
1.819MEG	82.055	78.591
1.920MEG	77.234	77.138
2.021MEG	72.414	75.685
2.122MEG	68.698	74.373
2.223MEG	65.405	73.115
2.324MEG	62.112	71.857
2.425MEG	58.819	70.599
2.526MEG	56.320	69.469

Figure 5-7 Numeric output

A printout of the data in the Numeric Output window may be obtained from the **Print** option of the **File** menu.

To display the Analysis Plot window, press F4, or click on the Analysis Plot button .

To display the Output window, press F5, or click on the Numeric Output button .

Noise plots

MC8 models three types of noise; thermal, shot, and flicker noise.

Thermal noise is produced by the random thermal motion of electrons and is always associated with resistance. Discrete resistors and the parasitic resistance of active devices contribute thermal noise.

Shot noise is caused by a random variation in the production of current from a variety of sources including recombination and injection. It is usually associated with dependent current sources. All semiconductor devices generate shot noise.

Flicker noise comes from a variety of sources. In BJTs, the source is contamination traps and other crystal defects that randomly release their captured carriers.

Noise analysis computes the contribution from all these sources as seen at the input and output. Output noise is calculated at the output node specified in the Noise Output field. To plot it, specify `ONoise` as the Y expression. Input noise is calculated at the same output node(s) specified in the Noise Output field, but is divided by the gain from the node(s) specified in the Noise Input field to the output node. To plot or print input noise, specify `INoise` as the Y expression.

Press F9 and edit the analysis limits to match Figure 5-8. Be sure to disable numeric output for the first two curves and to select the Log option.

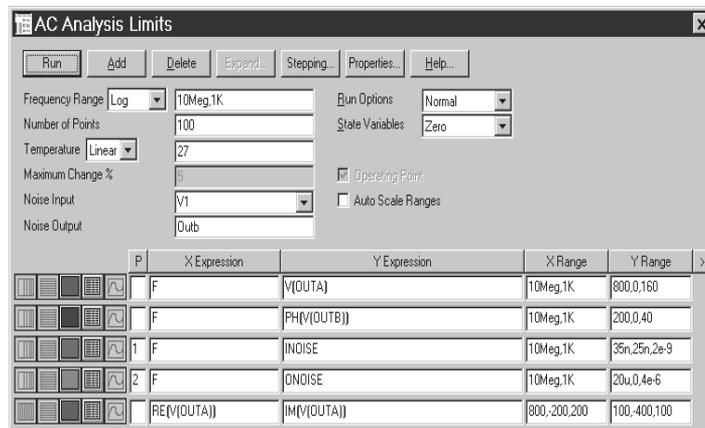


Figure 5-8 The Analysis Limits for Noise plots

Press F2 to run the analysis. The results look like this:

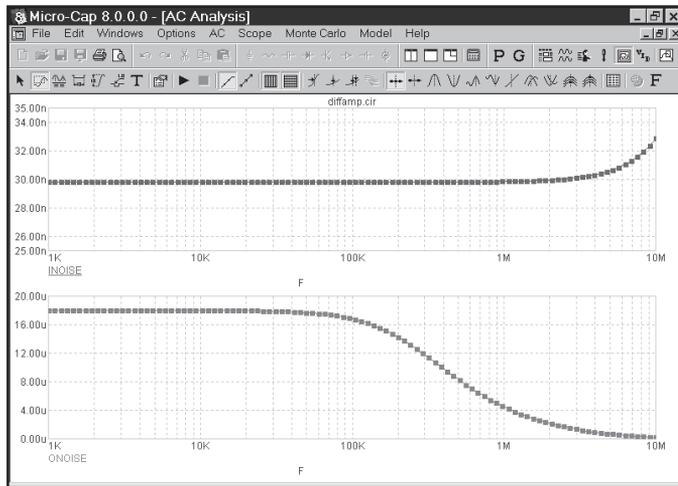


Figure 5-9 Plots of input and output noise

In this circuit, thermal noise is generated by the resistors. Shot and flicker noise are generated by the transistors.

Because the equations are structured differently for noise, it is not possible to simultaneously plot noise variables and other types of variables such as voltage and current. That is why we disabled the other curves by placing a blank in their P fields. If you attempt to plot both noise and other variables, the program will issue an error message.

Noise is an essentially random process and as a result carries no phase information. For this reason, noise is defined as an RMS quantity. Thus the phase (PH) and group delay (GD) operators will not produce meaningful results when plotting noise. With noise, only magnitudes are meaningful.

The units of noise are $V / \sqrt{\text{Hz}}$.

Nyquist plots

A Nyquist plot is easy to generate. Edit the analysis limits to look like these. Here we have enabled the last curve and disabled the other curves and selected log stepping.

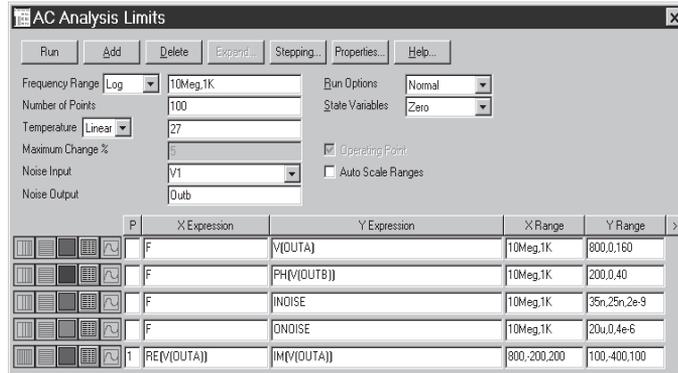


Figure 5-10 Analysis limits for the Nyquist plot

Press F2 to run the analysis. The result looks like this:

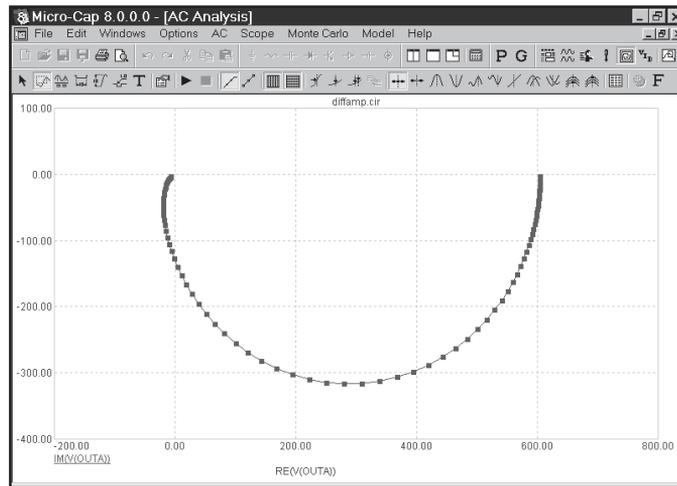


Figure 5-11 The Nyquist plot

Note that the X expression in this case was $RE(V(OUTA))$, not F as it usually is. In general, it can be any expression involving any AC variable.

Smith charts

To generate a Smith chart plot, use the  option from the chart type selector, adjacent to the P column. Here is how the Analysis Limits dialog box looks when set up to create a Smith chart plot for S22, using the SMITH.CIR file.

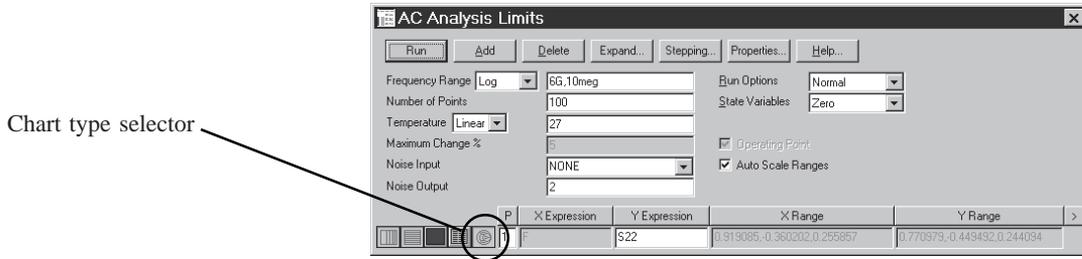


Figure 5-12 Analysis limits for the Smith chart

Note that when the Smith chart option is selected, the X expression must be F, so this field is not available for editing. Here is the resulting Smith chart plot.

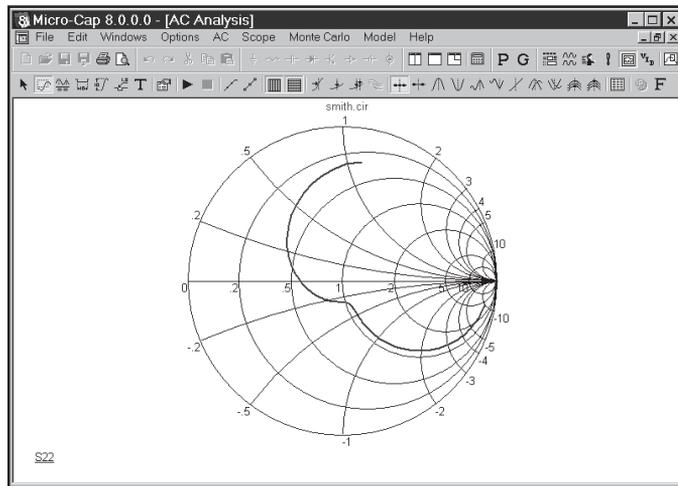


Figure 5-13 Smith chart plot of S22

Polar plots

To create a polar plot, use the polar plot option, . Here is how the dialog box appears when set up to create a polar plot for S12, using the SMITH.CIR file.

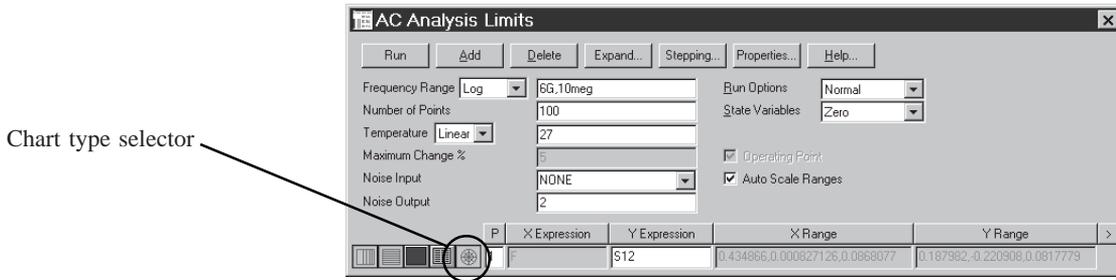


Figure 5-14 Analysis limits for a polar plot

When the polar plot option is selected, the X expression must be F, so this field is not available for editing. Here is the resulting polar plot.

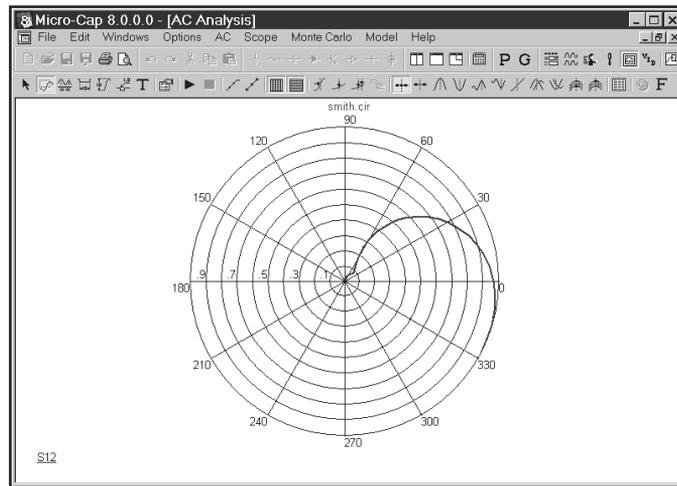


Figure 5-15 Polar plot of S12

What's in this chapter

DC analysis is the topic for this chapter. It is a short tutorial covering these subjects:

- What DC analysis does
- What the DC analysis limits do
- A simple circuit for generating IV-curves

What happens in DC analysis

In DC analysis, the program treats capacitors as open circuits and inductors as short circuits. It sweeps the specified Variable 1 and solves for the steady-state DC node voltages and branch currents. Two stepping loops are provided for easy creation of parametric plots such as device IV curves. The variable for either loop can be a voltage or current source value, temperature, a model parameter, or a symbolic variable. As in the other analysis routines, the complete set of variables are available for printing and plotting, with the obvious exceptions of charge, flux, capacitance, inductance, B field, and H field. These variables are not provided since flux and charge are ignored in DC analysis. To explore DC analysis, load the file CURVES. It looks like this:

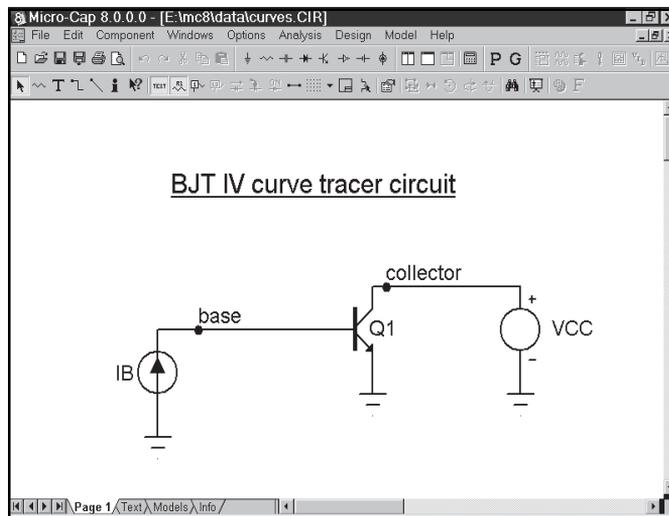


Figure 6-1 The CURVES circuit

This circuit is used to plot the I-V characteristics of a bipolar transistor. It can be easily adapted to other device types, such as GaAsFets, MOSFETs, and JFETs. There is a voltage source for the collector voltage sweep and a current source for the base current sweep.

After loading the circuit, select **DC** from the **Analysis** menu. When the DC Analysis Limits dialog box comes up, it looks like Figure 6-2.

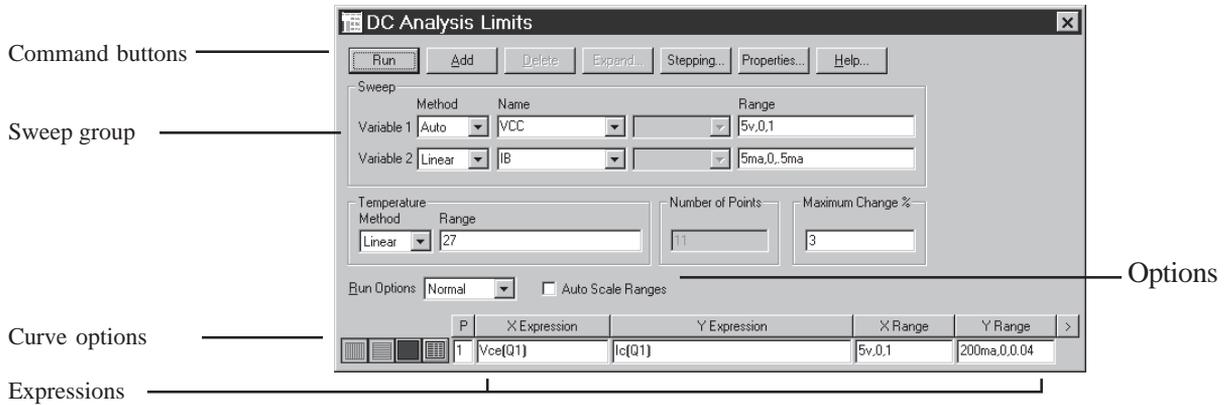


Figure 6-2 The DC Analysis Limits dialog box

The dialog box instructs the program to step the value of the current source IB from 0 to 5 ma, using a maximum step size of 0.5 ma. For each value of the current source, IB, the voltage source, VCC, is to be swept from 0 to 5 volts, using a maximum step size of 10. The step size may be varied during the run to optimize convergence and to conform to the *<Maximum change %>* constraint, but it is never to be larger than 1. Click on the Run command button and the results look like this:

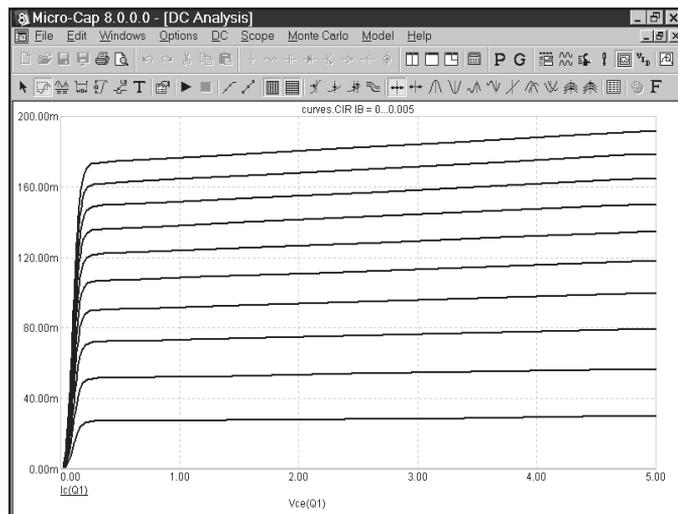


Figure 6-3 DC analysis plot of IV characteristics

The DC Analysis Limits dialog box

The Analysis Limits dialog box is divided into five major areas: the Command buttons, Sweep group, Curve options, Expression fields, and Options.

Command buttons provide these commands.

Run: This command starts the analysis run. Clicking the Tool bar Run  button or pressing F2 will also start the run.

Add: This command adds another Curve options field and Expressions field line after the line containing the cursor. The scroll bar to the right of the Expressions field scrolls through the curves when needed.

Delete: This command deletes the Curve option field and Expressions field line where the text cursor is.

Expand: This command expands the working area for the text field where the text cursor currently is. A dialog box is provided for editing or viewing. To use the feature, click in an expression field, then click the Expand button.

Stepping: This command calls up the Stepping dialog box. Stepping is reviewed in a separate chapter.

Properties: This command invokes the Properties dialog box which lets you control the analysis plot window and the way curves are displayed. See the Plot Properties dialog box article in Chapter 4, "Transient Analysis".

Help: This command calls up Help information for DC analysis.

The definition of each field in this group is as follows:

- **Variable 1:** This row specifies the Method, Name, and Range fields for Variable 1. Its value is usually plotted along the X axis. The column fields for this variable are:
 - **Method:** This field specifies one of four methods for sweeping or stepping the variable: Auto, Linear, Log, or List.
 - **Auto:** In this mode the rate of step size is adjusted to keep the point-to-point change less than Maximum Change % value.

- **Linear:** This mode uses the following syntax from the Range column for this row:

`<end> [,<start> [,<step>]]`

Start defaults to 0.0. *Step* defaults to $(start - end)/50$.

Variable 1 starts at *start*. Subsequent values are computed by adding *step* until *end* is reached.

- **Log:** This mode uses the following syntax from the Range column for this row:

`<end> [,<start> [,<step>]]`

Start defaults to $end/10$. *Step* defaults to $exp(\ln(end/start)/10)$.

Variable 1 starts at *start*. Subsequent values are computed by multiplying by *step* until *end* is reached.

- **List:** This mode uses the following syntax from the Range column for this row:

`<v1> [,<v2> [,<v3>] ...[,<vn>]]`

The variable is simply set to each of the values *v1*, *v2*, .. *vn*.

- **Name:** This field specifies the name of Variable 1. The variable may be a source current or voltage, temperature, a model parameter, or a symbolic parameter (one created with a .DEFINE statement). Model parameter stepping requires a model name and model parameter name.
- **Range:** This field specifies the numeric range for the variable. The range syntax depends upon the Method field described above.
- **Variable 2:** This row specifies the Method, Name, and Range fields for Variable 2. The syntax is the same as for Variable 1, except that the stepping method options include None and exclude Auto. Each discrete value of Variable 2 from the range field produces one complete branch.
- **Temperature:** This field specifies the global temperature(s) of the run(s) in degrees Celsius. This temperature is used for each device unless individual device temperatures are specified.

If the Temperature Method list box shows Linear or Log the format is:

$\langle high \rangle [, \langle low \rangle [, \langle step \rangle]]$

The default value of $\langle low \rangle$ is $\langle high \rangle$, and the default value of $\langle step \rangle$ is $\langle high \rangle - \langle low \rangle$ (linear mode) or 1.0 (log mode). Temperature values start at $\langle low \rangle$ and are either incremented (linear mode) or multiplied (log mode) by $\langle step \rangle$ until $\langle high \rangle$ is reached.

If the Temperature Method list box shows List the format is:

$\langle t1 \rangle [, \langle t2 \rangle [, \langle t3 \rangle] [, \dots]]$

where $t1, t2, \dots$ are individual values of temperature.

One run is produced for each temperature. *Note that when temperature is selected as one of the stepped variables (Variable 1 or Variable 2) this field is not available.*

• **Number of Points:** This is the number of data points to be interpolated and printed if numeric output is requested. This number defaults to 51 and is usually set to an odd value to produce an even print interval. Its value is:

$(\langle final1 \rangle - \langle initial1 \rangle) / (\langle Number \text{ of points} \rangle - 1)$

$\langle Number \text{ of points} \rangle$ values are printed for each value of variable 2.

• **Maximum Change:** If the sweep method of Variable 1 is Auto, its rate of change is constrained to keep the point-to-point change in the first Y expression less than Maximum Change % value.

The Curve options are located below the Numeric limits and to the left of the Expressions. Curve options affect the curve in the same row. The definition of each option is as follows:

The first option toggles the X-axis between a linear  and a log  plot. Log plots require positive scale ranges.

The second option toggles the Y-axis between a linear  and a log  plot. Log plots require positive scale ranges.

The  option activates the color menu. There are 64 color choices for an individual curve. The button color is the curve color.

The  option prints a table showing the numeric value of the curve. The number of values printed is controlled by the Number of Points value. The table is printed to the Numeric Output window and saved in the file CIRCUITNAME.DNO.

A number from 1 to 9 is used in the Plot (P) column to group the curves into different plots. All curves with like numbers are placed in the same plot. If the P column is blank, the curve is not plotted.

The Expressions field is used to specify the horizontal (X) and vertical (Y) scale ranges and expressions. Some common expressions used are VCE(Q1) (collector to emitter voltage of transistor Q1) or IB(Q1) (base current of transistor Q1).

The X Range and Y Range fields specify the scales to be used when plotting the X and Y expressions. The range format is:

<high> [,<low>] [,<grid spacing>] [,<bold grid spacing>]

<low> defaults to zero. [*<grid spacing>*] sets the spacing between grids. [*<bold grid spacing>*] sets the spacing between bold grids. Placing "AUTO" in the scale range calculates that individual range automatically. The Auto Scale Ranges option calculates scales for all ranges during the simulation run and updates the X and Y Range fields. The Auto Scale (F6) command immediately scales all curves, without changing the range values, letting you restore them with CTRL + HOME if desired. Note that *<grid spacing>* and *<bold grid spacing>* are used only on linear scales. Logarithmic scales use a natural grid spacing of 1/10 the major grid values and bold is not used. Auto Scale uses the number of grids specified in the **Properties dialog box (F10) / Scales and Formats / Auto/ Static Grids** field.

Clicking the *right* mouse button in the Y expression field invokes the Variables list which lets you select variables, constants, functions, and operators, or expand the field to allow editing long expressions. Clicking the *right* mouse button in the other fields invokes a simpler menu showing suitable choices.

The Options are below the Numeric limits. The Auto Scale Ranges option has a check box. Clicking the mouse in the box will toggle the options on or off with an X in the box showing that the option is enabled.

The options available from here are:

- **Run Options**

- **Normal:** This runs the simulation without saving it to disk.
- **Save:** This runs the simulation and saves it to disk.
- **Retrieve:** This loads a previously saved simulation and plots and prints it as if it were a new run.
- **Auto Scale Ranges:** This sets the X and Y range to auto every time a simulation is run. If it is disabled, the values from the range fields will be used.

Press F3 and unload the CURVES circuit without saving it.

What's in this chapter

Every analysis routine generates curves as a part of the run, based upon the instructions in the Analysis Limits dialog box. After the run is complete, you can inspect, manipulate, magnify, and annotate the plot, as well as search for and display specific numeric positions or values for each of the curves. The collection of tools that helps you do this is called Scope.

This chapter shows how to use these features to get the maximum information from an analysis run. The topics include:

- Experimenting with Scope
- Magnifying
- Panning
- Cursor mode
- Cursor mode panning and scaling
- Cursor positioning
- Adding text to plots
- Adding tags to plots
- Performance functions
- Animating an analysis

Experimenting with Scope

Load the file CHOKE. It looks like this:

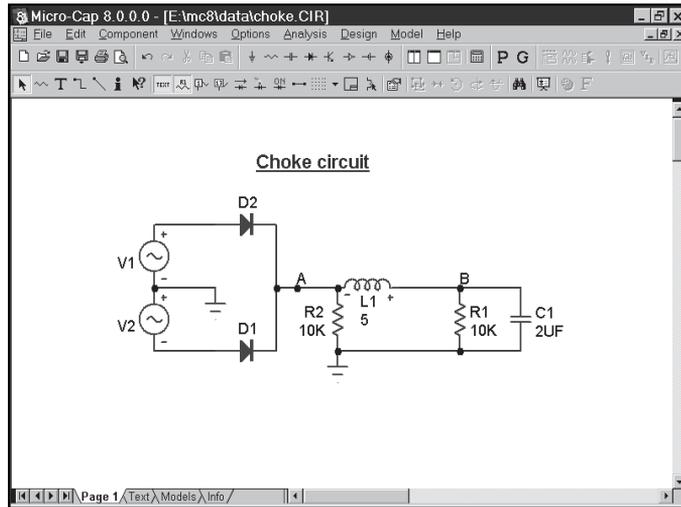


Figure 7-1 The CHOKE circuit

After loading the circuit, select transient analysis by pressing ALT + 1. Press F2 to start the run. When finished, the plot looks like Figure 7-2.

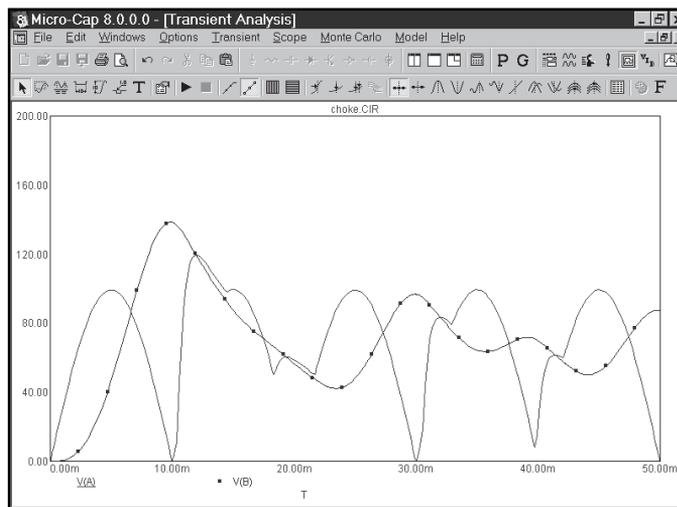


Figure 7-2 Transient analysis of the CHOKE circuit

When the analysis is over, click on the **Scope** menu. It provides these options:

- **Delete All Objects:** This command removes all objects (shapes, tags, or text) from the analysis plot. To delete an object, select it by clicking on it, then press CTRL + X or the DELETE key. To delete all objects use CTRL + A to select all objects, then press CTRL + X or the DELETE key.
- **Auto Scale:** This command scales the plot group containing the selected curve. F6 may also be used. The selected curve is the one whose Y expression is underlined.
- **Restore Limit Scales:** This command restores the range scales to the values in the Analysis Limits dialog box. CTRL + HOME may also be used.
- **View:** The view options only affect the display of simulation results, so you may change these after a run and the screen is redrawn accordingly. These options may also be accessed through Tool bar icons shown below.



- **Data Points:** This marks the actual points calculated by the program on the curve plot. All other values are linearly interpolated.



- **Tokens:** This adds tokens to each curve plot. Tokens are small graphic symbols that help identify the curves.



- **Ruler:** This substitutes ruler tick marks for the normal full screen X and Y axis grid lines.



- **Plus Mark:** This replaces continuous grids with "+" marks at the intersection of the X and Y grids.



- **Horizontal Axis Grids:** This adds grids to the horizontal axis.



- **Vertical Axis Grids:** This adds grids to the vertical axis.



- **Minor Log Grids:** This adds but does not label minor log grids between major grids at the 2, 3, 4...9 positions.



- **Minor Log Grids 2 5:** This adds and labels minor log grids between major grids at the 2 and 5 positions.



- **Baseline:** This adds a plot of the value 0.0 for use as a reference.



- **Horizontal Cursor:** This adds a horizontal cursor intersecting each of the two vertical numeric cursors at their respective data point locations.

- **Trackers:** These options control the display of the cursor, intercept, and mouse trackers, which are little boxes containing the numeric values at the cursor data point, its X and Y intercepts, or at the current mouse position.

- **Cursor Functions:** These control the cursor positioning in Cursor mode. Cursor positioning is described in more detail in a later section.

- **Label Branches:** This accesses a dialog box which lets you select from the automatic and user-specified X location method for labeling multiple branches of a curve. This option appears when multiple branches are created by Monte Carlo, parameter stepping, or temperature stepping. To remove labels, select them with the mouse, or select all objects with CTRL + A, then press the delete key.

- **Label Data Points:** This accesses a dialog box which lets you specify a set of time, frequency, or input sweep data points, in transient, AC, or DC analysis, respectively, which are to be labelled. This command is mostly used to label frequency data points on polar and Smith charts. To remove labeled points delete them from the dialog box.



- **Animate Options:** This option displays the node voltages/states, and optionally, device currents, power, and operating conditions (ON, OFF, etc.), on the schematic as they change from one data point to the next. To use this option, click on the  button, select one of the wait modes, then click on the Tile Horizontal  button, or Tile Vertical  button, then start the run. If the schematic is visible, MC8 prints the node voltages on analog nodes and the node states on digital nodes. To print device currents click on . To print device power click on . To print the device conditions click on . It lets you see the evolution of states as the analysis proceeds. To restore a full screen plot, click on the  button.

- **Normalize at Cursor: (CTRL + N)** This option normalizes the selected (underlined) curve at the current active cursor position. The active cursor is the last cursor moved, or if neither has been moved, then the left cursor. Normalization divides each of the curve's Y values by the Y value of the curve at the current cursor position, producing a normalized value of 1.0 there. If the Y expression contains the dB operator and the X expression is F

(Frequency), then the normalization subtracts the value of the Y expression at the current data point from each of the curve's data points, producing a value of 0.0 at the current cursor position.

- **Normalize at Minimum:** This option normalizes the selected (underlined) curve at its minimum value.

- **Normalize at Maximum:** This option normalizes the selected (underlined) curve at its maximum value.



- **Go to X: (SHIFT + CTRL + X)** This command lets you move the left or right cursor to the next instance of a specific value of the X expression of the selected curve. It then reports the Y expression value at that data point.



- **Go to Y: (SHIFT + CTRL + Y)** This command lets you move the left or right cursor to the next instance of a specific value of the Y expression of the selected curve. It then reports the X expression value at that data point.



- **Go to Performance:** This command calculates performance function values on the Y expression of the selected curve. It also moves the cursors to the measurement points. For example, you can measure pulse widths and bandwidths, rise and fall times, delays, periods, and maxima and minima.



- **Go to Branch:** This command lets you select which branch of a curve to place the left and right cursors on. The left cursor branch is colored in the primary select color and the right cursor branch in the secondary select color.

- **Tag Left Cursor: (CTRL + L)** This command attaches a tag to the left cursor data point on the selected curve.

- **Tag Right Cursor: (CTRL + R)** This command attaches a tag to the right cursor data point on the selected curve.



- **Tag Horizontal: (SHIFT + CTRL + H)** This command places a horizontal tag between the left and the right cursor, showing the difference between the two X expression values.



- **Tag Vertical: (SHIFT + CTRL + V)** This command places a vertical tag between the left and the right cursor, showing the difference between the two Y expression values.

- **Align Cursors:** This option, available only in Cursor mode, forces the numeric cursors of different plot groups to stay on the same data point.
- **Keep Cursors on Same Branch:** When stepping produces multiple runs, one line is plotted for each run for each plotted expression. These lines are called branches of the curve or expression. This option forces the left and right cursors to stay on the same branch of the selected curve when the UP ARROW and DOWN ARROW cursor keys are used to move the numeric cursors among the several branches. If this option is disabled the cursor keys will move only the left or right numeric cursors, not both, allowing them to occupy positions on different branches of the curve.
- **Same Y Scales For Each Plot Group:** This option forces all curves in a plot group to use the same Y scale. If the curves have different Y Range values, one or more Y scales will be drawn, which may lead to crowding of the plot with scales.



- **Thumb Nail Plot:** This option draws a small guide plot which gives a global view of where the current plot(s) are on the whole curve. You select different views by dragging a box over the curve with the left mouse button. Dragging the box with the right mouse button pans the main plot.

- **Enable X Scaling:** This option enables horizontal axis scaling for the F6 (Auto Scaling), Zoom (+,-), and panning and drag scaling commands. If this option is enabled and you press F6, the curves in the selected plot group will be auto-scaled in the horizontal direction. Vertical axes are not affected by this option. The Auto Scale option on the Analysis Limits dialog box overrides this option and auto-scales both the X and Y axes.

- **Enable Y Scaling:** This option enables vertical axis scaling for the F6 (Auto Scaling), Zoom (+,-), and panning and drag scaling commands. If this option is enabled and you press F6, the curves in the selected plot group will be auto-scaled in the vertical direction. Horizontal axes are not affected by this option. The Auto Scale option on the Analysis Limits dialog box overrides this option and auto-scales both the X and Y axes.

- **Keep X Scales the Same:** This option forces all horizontal scales using same X expression to be the same. In other words, drag scaling or panning in plot group 1 will change the horizontal scale in all all plot groups that have the same X expression. This option is convenient when you want separate plot groups but want the scales to remain the same.

Select **Mode** from the **Options** menu. The five modes beginning with **Scale** deal with analysis plots and are usually accessed by clicking on their Tool bar buttons shown below, but they may also be activated from this menu.



- **Scale:** In this mode, the left mouse button drags an outline box to magnify a portion of the plot. F7 also accesses this mode.



- **Cursor:** This mode shows the numeric value of each curve at each of two plot cursors. The left mouse button controls the left plot cursor and the right button controls the right plot cursor. The left cursor is initially placed on the first data point. The right cursor is initially placed on the last data point. The LEFT ARROW or RIGHT ARROW keys move the left or right cursor to a point on the selected curve. Depending on the positioning mode, the point may be the next local peak or valley, global high or low, inflection point, or simply the next data point. The direction is left or right depending on whether a LEFT ARROW or RIGHT ARROW key is pressed. If the SHIFT key is used with the cursor key, the right cursor is moved, otherwise, the left cursor is moved. F8 may also be used to access this mode.



- **Point Tag:** In this mode, the left mouse button is used to tag a data point with its numeric (X,Y) value. The tag snaps to the nearest data point.



- **Horizontal Tag:** In this mode, the left mouse button can be used to drag between two data points to measure the horizontal difference.



- **Vertical Tag:** In this mode, the left mouse button can be used to drag between two data points to measure the vertical difference.

Note 1: In all modes except Cursor, a drag of the right mouse button pans the plot. In Cursor mode, CTRL + right mouse drag pans the plot.

Note 2: Press the SPACEBAR key to toggle between the current mode and Select mode.

Magnifying

Press ESC to remove the menus. Click the left mouse button just above and to the left of the large peak of the V(B) curve. While holding the left button down, drag the mouse down and to the right to create a magnification box like the one shown in Figure 7-3.

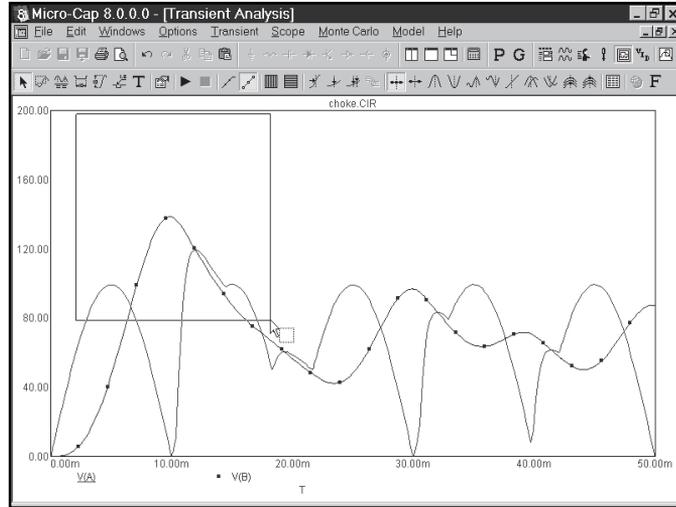


Figure 7-3 Magnifying a plot region

Release the button and MC8 scales the plot to match the outline of the box.

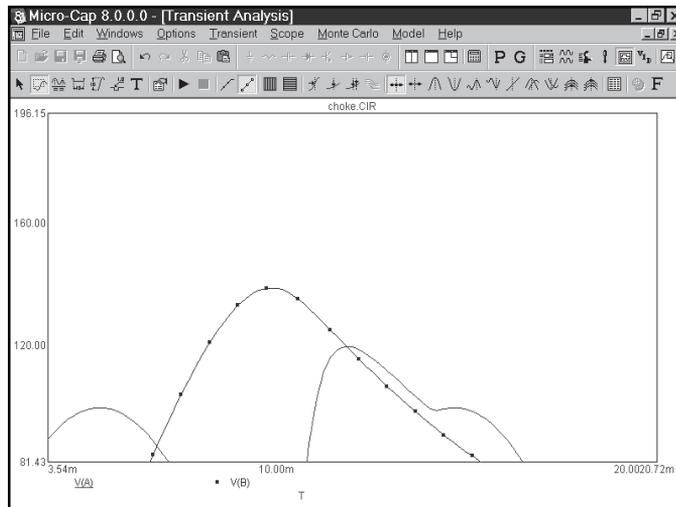


Figure 7-4 The magnified region

Panning

Panning is like scrolling in a text field, except that it is omnidirectional. To illustrate, click the right mouse button in the middle of the plot. The mouse arrow cursor changes to a hand cursor. While holding the right button down, drag the mouse slowly up and to the left until it nearly touches the upper left corner of the plot window. The effect is like sliding a paper across a desk. Dragging the mouse dynamically updates the display to reflect the change in view. Release the button and the plot looks like this:

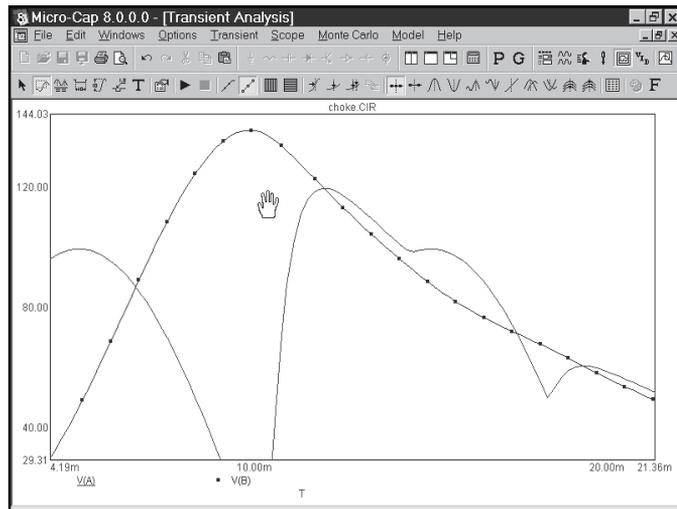


Figure 7-5 Panning the plot

As with panning in the Schematic editor, the keyboard may also be used. The CTRL key plus the arrow keys move the view in the direction of the arrow.

The combination of panning and magnifying allows close inspection of any portion of the curves.

Note that you can also scale (magnify) and pan while in Cursor mode:

- Scaling: CTRL + left mouse drag
- Panning: CTRL + right mouse drag

Press CTRL + HOME or choose **Restore Limit Scales** from the **Scope** menu to return the analysis to its original offset and scale.

Cursor mode

Click on the Cursor mode button  or Press F8 to place Scope in the Cursor mode. In this mode the display looks like this:

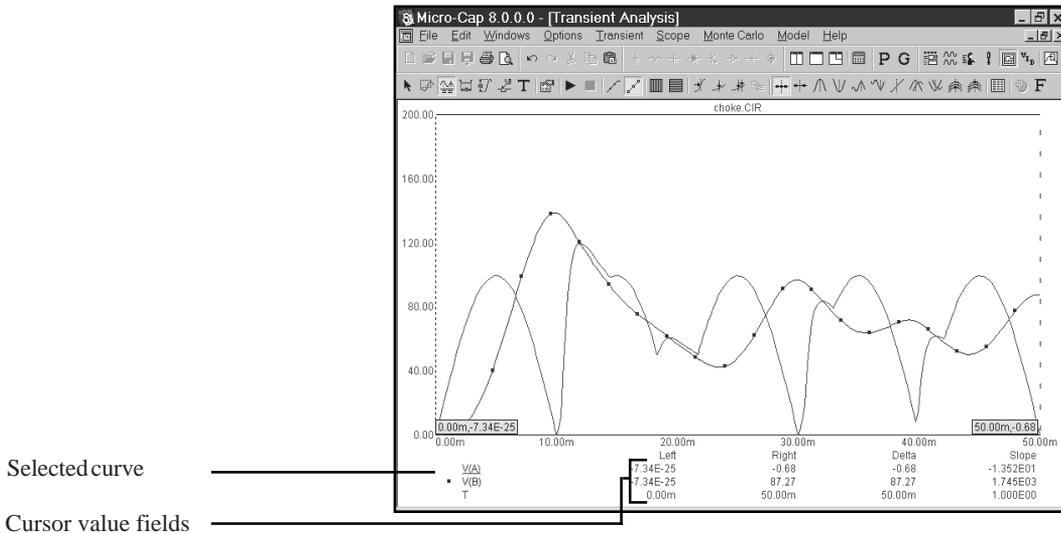


Figure 7-6 The Cursor mode

The Cursor value fields, located below the plots, show the numeric value of each curve or expression at both cursor positions. The X expression value is shown below the Y expression values. The purpose of each field is as follows:

- **Left:** This shows the value of the row's Y expression at the left cursor.
- **Right:** This shows the value of the row's Y expression at the right cursor.
- **Delta:** This shows the value of the row's Y expression at the right cursor minus the value of the Y expression at the left cursor.
- **Slope:** This shows the row's Y expression delta value divided by the delta value of the X expression. If the X expression is T (time), then this field approximates the time derivative of the curve between the two cursors. The slope calculation can be changed to dB/Octave or dB/Decade from **Properties / Scales and Formats**, if dB is used in the Y expression and F (frequency) is the X expression.

Cursor mode panning and scaling

Mouse

Normal mouse-controlled scaling and panning are not available in Cursor mode, since the mouse movements in this mode are used to control the left and right curve cursors. However, by using the CTRL key together with the mouse, you can pan and scale within Cursor mode as follows:

Scaling: CTRL + left mouse drag
Panning: CTRL + right mouse drag

To scale (magnify) the curves, press the CTRL key down and drag with the left mouse button.

To pan the curves, press the CTRL key down and drag with the right mouse button.

Keyboard

To pan the curves with the keyboard, while holding down the CTRL, press the LEFT ARROW, RIGHT ARROW, UP ARROW, or DOWN ARROW keys to pan the view.

There is no keyboard-based scaling procedure.

Cursor mode positioning

Drag with the left mouse button to control the left cursor, and the right button to control the right cursor. The LEFT ARROW and RIGHT ARROW cursor keys move the *left* cursor and SHIFT + cursor keys move the *right* cursor. The mouse places the cursor anywhere, even between simulation data points. The keyboard moves the cursor to one of the simulation data points, depending on the positioning mode. The positioning mode, chosen with a Tool bar button, affects where the cursor is placed the next time a cursor key is pressed. Cursor positioning picks locations on the selected, or underlined, curve. Curves are selected with the Tab key, or by clicking the Y expression with the mouse.



- **Next Simulation Data Point:** In this mode, pressing the cursor keys finds the next actual data point in the direction of the cursor arrow from the run.



- **Next Interpolated Data Point:** In this mode, pressing the cursor keys finds the next rounded interpolated data point in the direction of the cursor arrow.



- **Peak:** In this mode, pressing the cursor keys finds the next local peak on the selected curve.



- **Valley:** In this mode, pressing the cursor keys finds the next local valley on the selected curve.



- **High:** Clicking this button finds the data point with the largest algebraic value on the current branch of the selected curve.



- **Low:** Clicking this button finds the data point with the smallest algebraic value on the current branch of the selected curve.



- **Inflection:** In this mode, pressing the cursor keys finds the next data point with the largest magnitude of slope, or first derivative, on the selected curve.



- **Top:** Clicking this button finds the data point with the greatest Y value of all branches of the selected curve *at the current X cursor position*.



- **Bottom:** Clicking this button finds the data point with the smallest Y value of all branches of the selected curve *at the current X cursor position*.



• **Global High:** Clicking this button finds the data point with the largest algebraic Y value of all data points on all branches of the selected curve.



• **Global Low:** Clicking this button finds the data point with the smallest algebraic Y value of all data points on all branches of the selected curve.

When Cursor mode is selected, a few initialization events occur. The selected curve is set to the first, or top curve. The Next mode is enabled. Clicking on a mode button not only changes the mode, it also moves the last cursor in the last direction. The last cursor is set to the left cursor and the last direction is set to the right. The left cursor is placed at the left on the first data point. The right cursor is placed at the right on the last data point.

Click on the Peak  button. The left cursor moves to the first peak to the right of its current position on the selected curve, V(A). Press SHIFT + LEFT ARROW key. The right cursor moves to the first peak on its left.

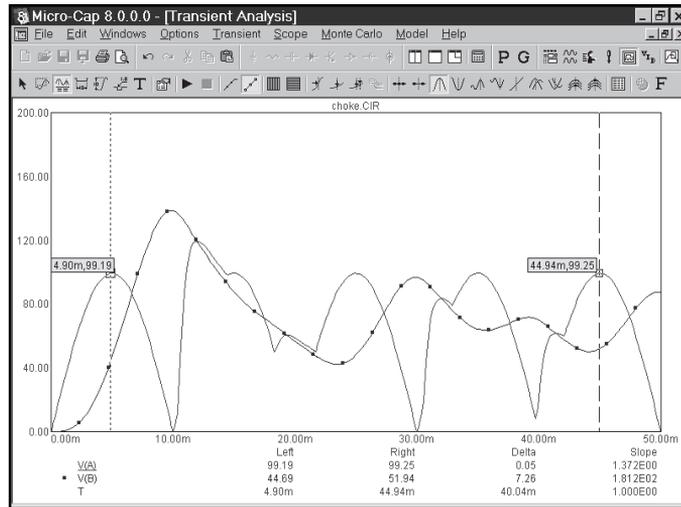


Figure 7-7 Using the Peak mode

Click on the Valley  button to select the Valley mode. Press the RIGHT ARROW key. The cursors are placed on the next valley in the right and left directions.

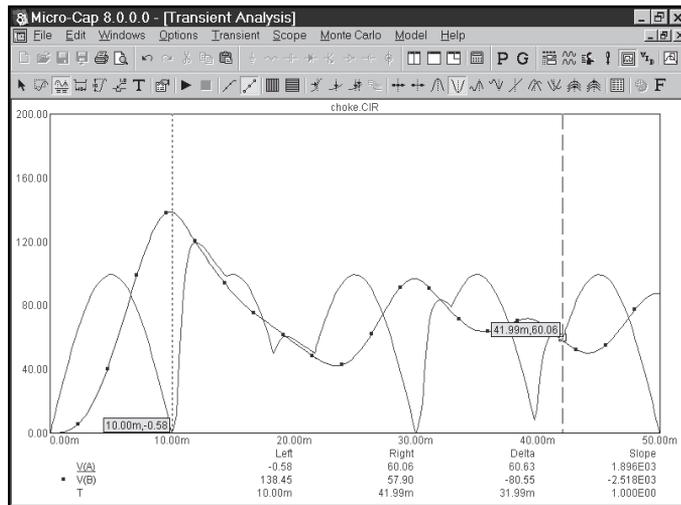


Figure 7-8 Using the Valley mode

Click on the High  button. The left cursor moves to the global high on the selected curve.

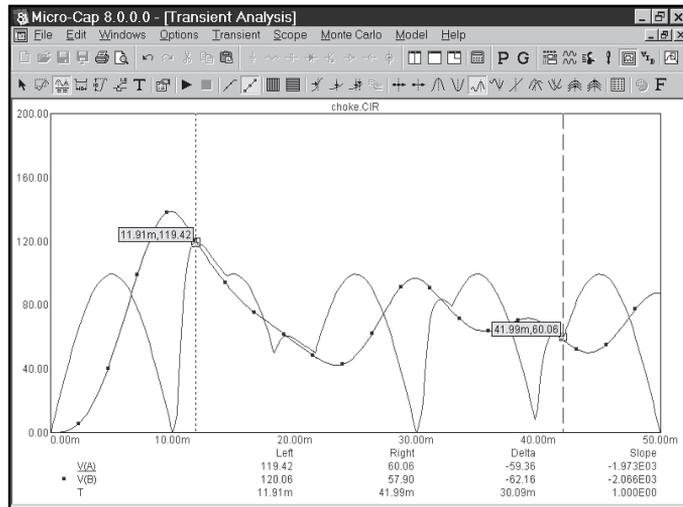


Figure 7-9 Using the High mode

Click on the Low  button to select the Low mode. The left cursor is positioned at the right hand edge of the plot, the global low for the curve.

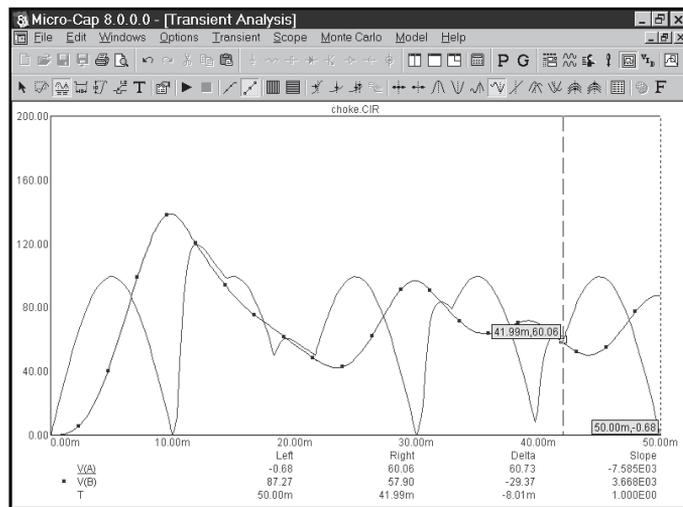


Figure 7-10 Using the Low mode

Click on the Inflection  button. The left cursor moves to the next inflection point on the selected curve. Press SHIFT + RIGHT ARROW key. The right cursor moves to the next inflection point on the selected curve moving left to right.

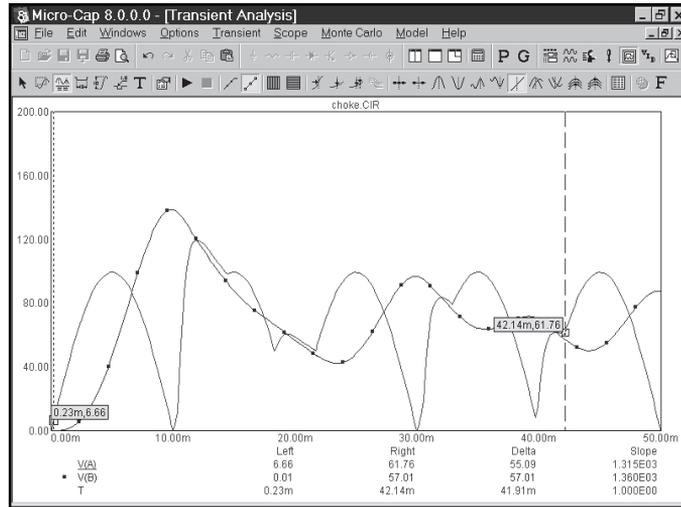


Figure 7-11 Using the Inflection mode

Click on the Go to Y  button. Type "50" in the dialog box that comes up and click on the 'Left' button in the dialog box. The left cursor is positioned at the first incidence of 50.0 for the selected curve's Y expression value.

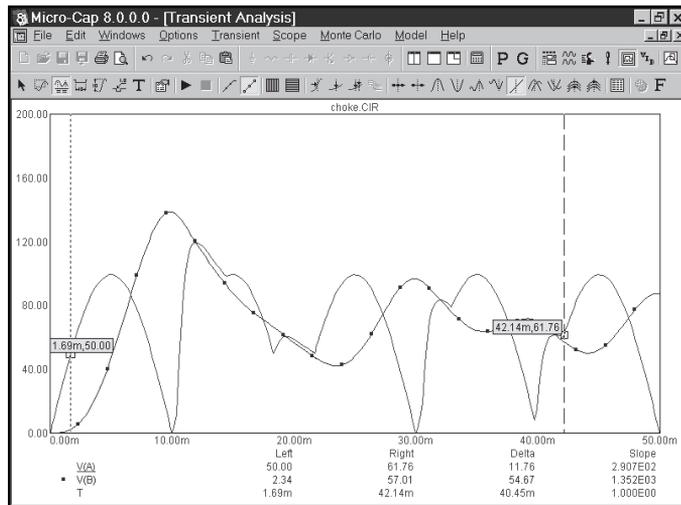


Figure 7-12 Using the Go to Y mode

Adding text to plots

Adding text to an analysis plot is easy. Click on the Close button to exit the Go to Y mode. Click on the **T** button to select the Text mode. Click the mouse near the large peak of the V(B) curve and type "V(B) peaks at 138 volts.". Click OK. The screen looks like this:

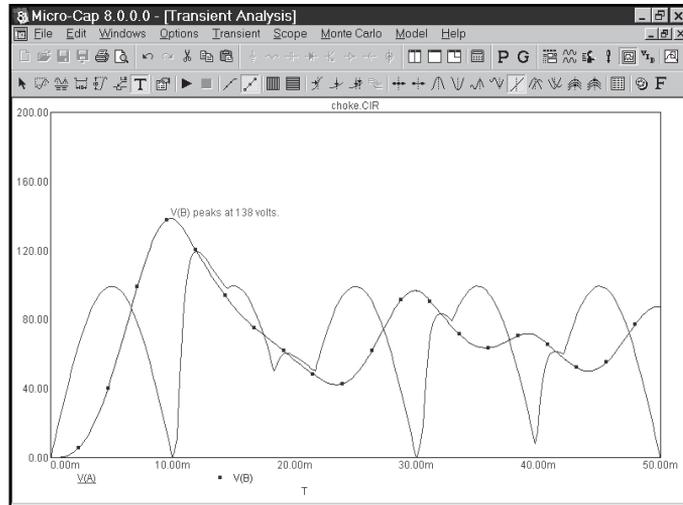


Figure 7-13 Adding text to a plot

Would the text look better if it were centered over the peak? To see if it would, first enter Select mode by pressing the SPACEBAR or by clicking on the Select button . Next, click on the text with the left button and drag it to the left until it is centered. Release the button and the text is redrawn at the mouse position.

Perhaps the text would look better if it were bold. Double click on the text and when the text dialog box is presented, click on the Font tab in the dialog box. This part of the dialog box lets you change the font and several font attributes. In this case, we need to change the Style attribute. From the Font Style group, select the Bold option. Click on the OK button. The text should now appear as in Figure 7-14. All font attributes for each instance of text are independently changeable.

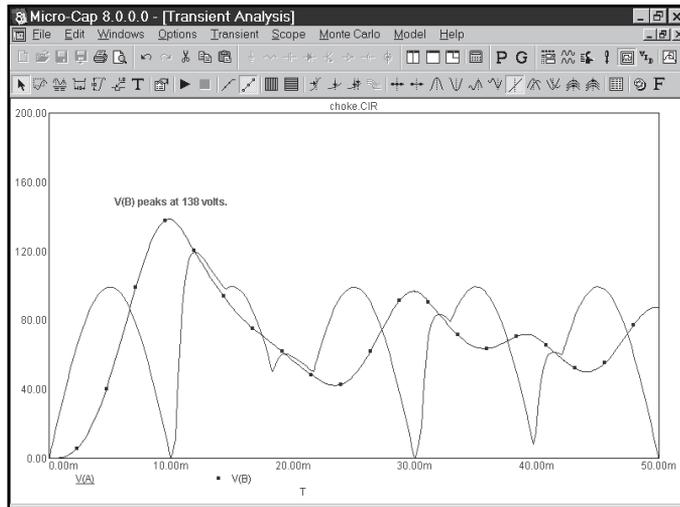


Figure 7-14 Changing the font style of the text

Adding tags to plots

Click on the  button to activate the Vertical Tag mode. Click the left mouse button near the highest peak of V(A). Drag the mouse to the highest peak of V(B) and release the button. The tag shows the vertical delta between the two data points.

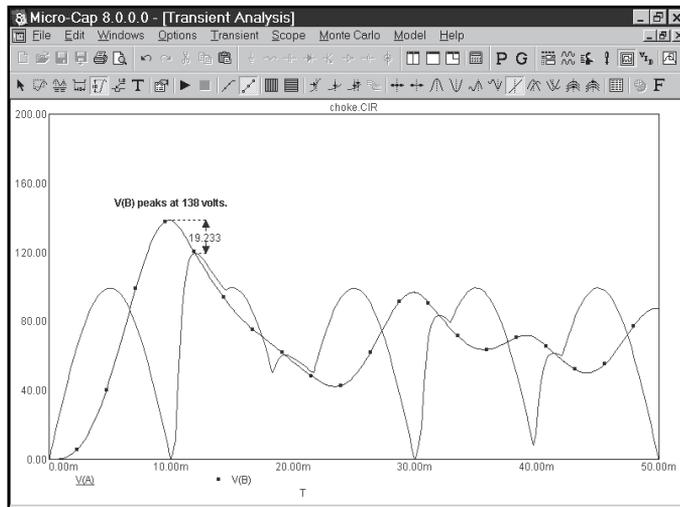


Figure 7-15 Using the Vertical tag mode

In a similar manner, the horizontal tag, enabled by the  button, measures the horizontal delta between the two data points of the tag. It is most useful for measuring the time difference between two digital events, such as the width of a pulse or the time delay between two events. Choose **Delete All Objects** from the **Scope** menu to delete the text and tag.

Click on the  button to activate the Point Tag mode. Click the left mouse button at the second peak of $V(B)$ and drag the mouse. The point tag labels the actual analysis data point nearest the initial mouse click. A shadow of the tag will track the mouse movements. Release the mouse button when the shadow is at the desired spot. The screen should look like Figure 7-16.

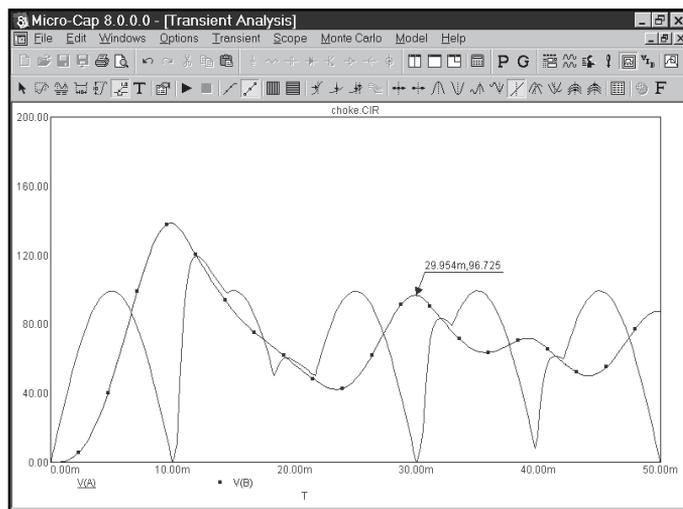


Figure 7-16 Using the Point tag

When in Select mode, double clicking on a tag element invokes the Tag dialog box. This dialog box lets you edit the arrow style, weight, color, and the font attributes of the number. All of the tag options are available for each instance of a tag element.

Both the text body and arrow tip of a tag may be repositioned after placement. Press SPACEBAR to toggle into Select mode. Double click on the tag and then deselect the Lock item in the dialog box. Click OK. Drag either the body handle or the arrow tip handle. The body handle is the small black rectangle near the text. The arrow tip handle is the small black rectangle at the tip of the tag arrow.

Press F3 to exit the analysis, then CTRL + F4 to close the file.

Performance functions

MC8 provides a group of functions for measuring performance-related curve characteristics. They can be used for performance plots, Monte Carlo analysis, 3D plotting and may also be applied directly to single curves. To illustrate, load the PERF2 circuit and run transient analysis. When the run is done, click on the Go To Performance  button, and drag the dialog box to the middle of the screen. The dialog box should look like this:

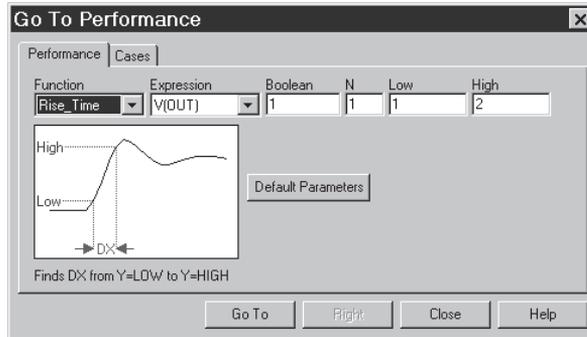


Figure 7-17 Go To Performance dialog box

Select Rise_Time from the list. Enter "3" in the High field, ".5" in the Low field, and "1" in the N field. Click on the Go To button. The screen looks like this:

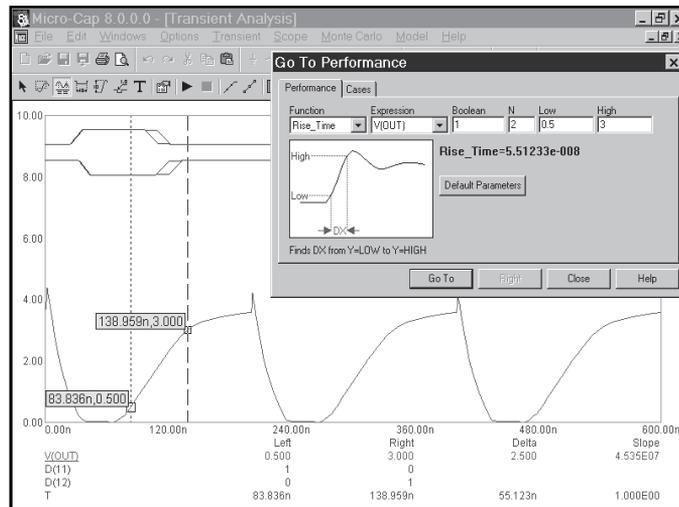


Figure 7-18 Measuring rise time

N is incremented for each click of the Go To button, so successive clicks measure the rise time of successive pulses. Select Fall_Time. Enter "1" in the N field, "3" in the High field, and ".5" in the Low field. Click the Go To button twice. The screen should look like Figure 7-19.

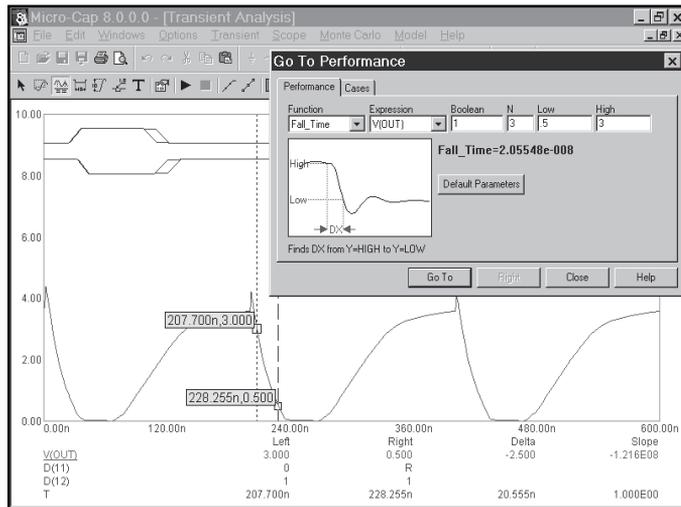


Figure 7-19 Measuring fall time

Select the Period function. Click on the Go To button. The screen looks like this:

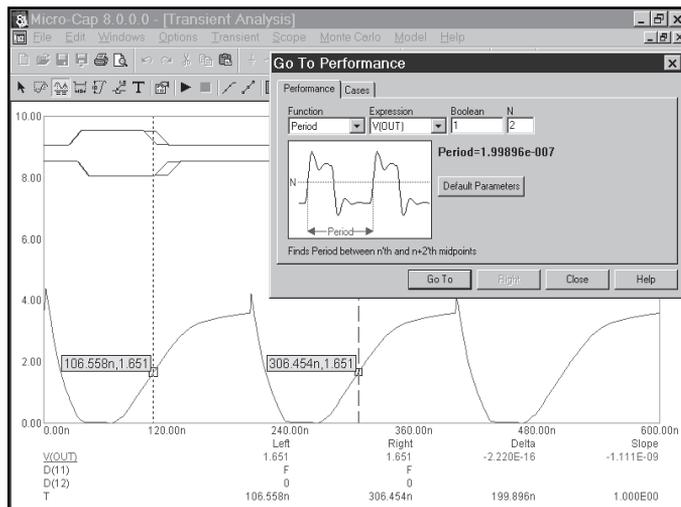


Figure 7-20 Measuring the pulse period

Select the Width function and click on the Go To button. Width is computed by finding the N'th rising instance of the value Level, then the N'th falling instance of the value Level. The function returns the difference between the X expression values, and places cursors at the two locations. The screen should look like this:

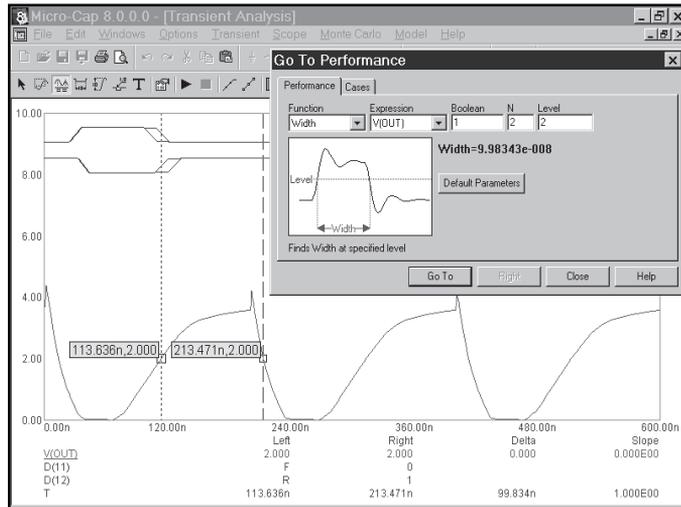


Figure 7-21 Measuring the pulse width

Select the X_Delta function. Click on Go To. The screen looks like this:

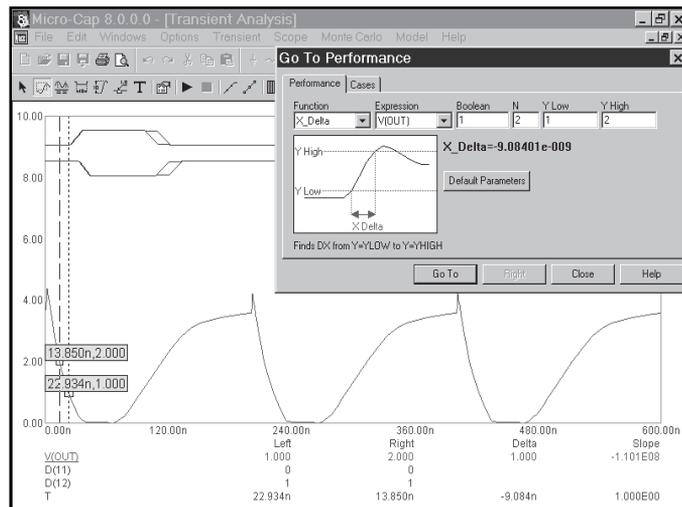


Figure 7-22 The X_Delta function

X_Delta locates the N'th instance of the YLow and YHigh values, places cursors at the data points and returns the difference between the X expression values.

The Y_Delta function works similarly. Select it and set XLow = 100n and XHigh = 200n, and click the Go To button. The result looks like this:

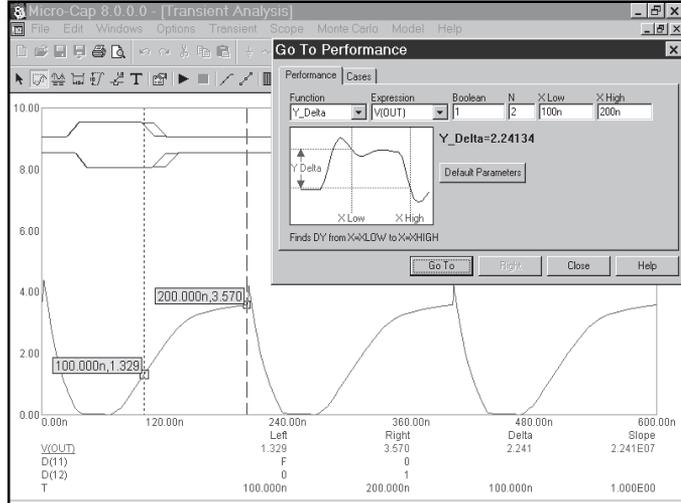


Figure 7-23 The Y_Delta function

Performance functions also work on digital curves. Select D(11) from the Expression list and Rise_Time from the Function list. Set N=2 and click the Go To button. The result looks like Figure 7-24.

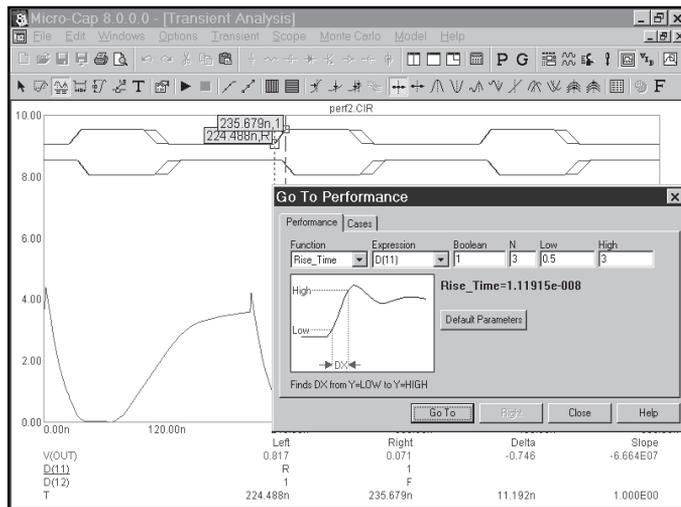


Figure 7-24 Measuring digital rise times

Animating an analysis

It is often useful to step through an analysis one time point at a time. Animation mode lets you do this and displays the node voltages and digital states on the schematic at each time point. To see how the mode works, load the file MIXED4. Select **Transient** from the **Analysis** menu. Press ALT + F4 to remove the Analysis Limits dialog box. Click on the Tile Vertical  button. Click on the Animate  button, select the Wait for Key Press option, and hit OK. Click on the Node Voltages . Press F2 to start the run. Press and hold the CTRL and SPACEBARS keys down to observe each new data point being computed and printed on the schematic.

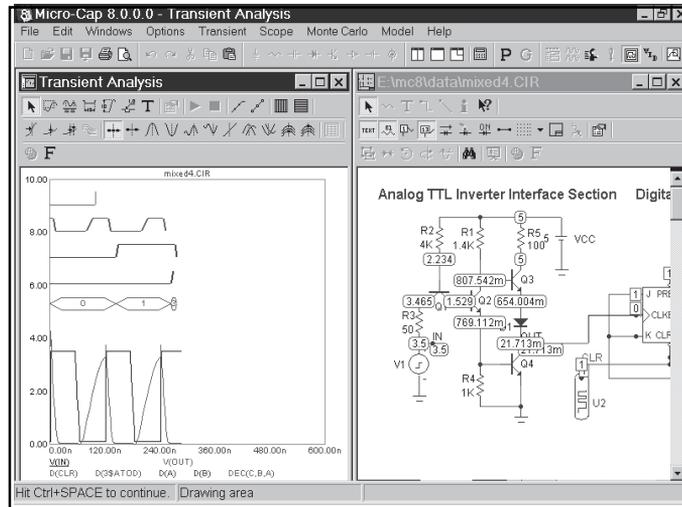


Figure 7-25 Animating an analysis

The figure above shows what the screen should look like midway through the run. In this mode, MC8 computes a single data point, optionally prints the voltages/ states, currents, power, and conditions on the schematic, and then awaits the CTRL + SPACEBAR key or time delay. It then repeats the cycle of compute, print, and pause until the run is complete. When Animate mode is canceled by clicking on the Don't Wait button in the Animate dialog box, the run continues in normal mode. To see both the schematic and analysis plot simultaneously, you must select either Tile Horizontal or Tile Vertical. This option is available for AC, DC, and transient analysis, although it is most useful in the transient analysis of digital and mixed-mode circuits. Press ESC to stop the run, F3 to exit the analysis, and CTRL + F4 to close the file.

What's in this chapter

Probe is an alternative tool for viewing the results of an analysis. It performs the selected analysis, saves the entire analysis in a disk file, then lets the user probe the schematic. Unlike a normal analysis, where only the curves specified in the Analysis Limits dialog box are saved, Probe saves everything, so you can decide what to plot after the analysis run. It is very much like probing a real circuit with an oscilloscope, spectrum analyzer, or curve tracer. In this chapter we demonstrate Probe's capabilities with several examples. The topics include:

- How Probe works
- Transient analysis variables
- Experimenting with Probe
- Navigating the schematic
- An AC example
- A DC example
- Probing SPICE files

How Probe works

Probe is another way to view simulation results. It lets you point to a location in a schematic and see the curves. It functions exactly like a normal simulation, saving all of the variables at each solution point in a disk file. When you click the mouse in the schematic, Probe determines where the mouse pointer is, extracts from the file the appropriate values for both the vertical and horizontal axes, and plots the resulting curve. You can also plot expressions using circuit variables as you can in a standard analysis.

To illustrate Probe, load the file ECLGATE. Select **Probe Transient** from the **Analysis** menu. Click on the Analysis Limits dialog box Close button. If a current analysis results file is not already available, the program runs a transient analysis on the circuit and saves the results in a file called ECLGATE.TSA. It then presents the Probe display. Click on the **Probe** menu. It provides these options:

- **New Run: (F2)** This option forces a new run. Probe automatically does a new run when the time of the last saved run is earlier than the time of the last edit to the schematic. However, if you have changed RELTOL or some other Global Settings value or option that can affect a simulation run, you may want to force a new run using the new value.
- **Limits: (F9)** This accesses the Analysis Limits dialog box for the particular Probe analysis type selected. From here you can choose a subset of the usual non-probe analysis options.
- **Add Curve:** This option lets you add a plot defined by a literal expression using any circuit variable. For example you might enter $VCE(Q1)*IC(Q1)$ to plot a transistor's collector power.
- **Delete Curves:** This option lets you selectively remove curves.
- **Delete All Curves: (CTRL + F9)** This removes all curves.
- **Separate Analog and Digital:** This mode places the analog and digital curves in separate plot groups.
- **One Curve:** In this mode, only one curve is plotted. Each time the schematic is probed, the old curve is replaced with the new one.

You can add more than one trace in One Curve mode by holding the CTRL key down while clicking on an object. This adds the curve if not already plotted or deletes it if it is already plotted.

- **Many Curves:** In this mode, new curves do not replace old ones, so many curves are plotted together, with common or individual vertical scales. The choice of common or individual scales is made by enabling or disabling the **Same Y Scales For Each Plot Group** option from the **Scope** menu.
- **Save All:** This option forces Probe to save all variables. Use it only if you need to display power, charge, flux, resistance, capacitance, inductance, B field, or H field.
- **Save V and I Only:** During a run, MC8 generates many variables. Saving them all to disk for display by Probe can create huge files. This option saves file space and lowers access time by saving only time, digital states, and voltage and current variables. It discards the remainder, including power, charge, flux, capacitance, inductance, resistance, and magnetic field values.
- **Plot Group:** This option lets you pick the plot group to place the next curve in. The next time you add an expression or click in the schematic to plot a curve it will be placed in the specified plot group.
- **Exit Probe (F3) :** This exits Probe.

Transient analysis variables

The **Vertical** menu selects the vertical variable and the **Horizontal** menu selects the horizontal variable. Where the arrow tip of the mouse cursor touches the schematic, Probe determines whether the object is a node or a component and whether it is analog or digital.

If the object is a digital node, Probe plots the digital waveform for the node.

If the object is analog, Probe extracts the vertical and horizontal variables specified by these menus and uses them to plot the analog curve. The analog variables available in transient analysis include:

- **Voltage:** If the mouse probes on a node, a node voltage is selected. If the mouse probes on the shape of a two-lead component, the voltage across the component is selected. If the mouse probes between two leads of a three or four lead active device, the lead-to-lead difference voltage is selected.
- **Current:** If the mouse probes on the shape of a two-lead component, the current through the component is selected. If the mouse probes on a lead of a three or four lead active device, the current into the lead is selected. Probe cannot directly plot current into a macro or subcircuit pin.
- **Energy:** If the mouse probes on a component, it plots the energy dissipated (ED), generated (EG), or stored (ES) in that component. If the component has more than one of these, a list appears allowing selection.
- **Power:** If the mouse probes on a component, it plots the power dissipated (PD), generated (PG), or stored (PS) in that component. If the component has more than one of these, a list appears allowing selection.
- **Resistance:** If the mouse clicks on a resistor, this selects its resistance.
- **Charge:** If the mouse clicks on a capacitor, the capacitor's charge is selected. If the probe occurs between the leads of a semiconductor device, the charge of the internal capacitor between the two leads is selected. For example, a click between an NPN base and emitter selects the CBE charge.
- **Capacitance:** This is the capacitance associated with the charge.
- **Flux:** If the mouse clicks on the body of an inductor, this selects its flux.

- **Inductance:** This is the inductance associated with the flux.
- **B Field:** If the mouse clicks on a core, this selects the B field of the core.
- **H Field:** If the mouse clicks on a core, this selects the H field of the core.
- **Time:** This selects the transient analysis simulation time variable.
- **Linear:** This selects a linear plot.
- **Log:** This selects a log plot.

Probing node voltages

Click the **Vertical** menu and select **Voltage**. Select **Time** on the **Horizontal** menu. Click the left mouse button on the dot above the Q1 collector lead. Probe scans the schematic, determines that the mouse is pointing to node 1, and plots $V(1)$ vs. time. Probe on Q1's base lead dot, then probe on its emitter lead dot. The display should now look like this:

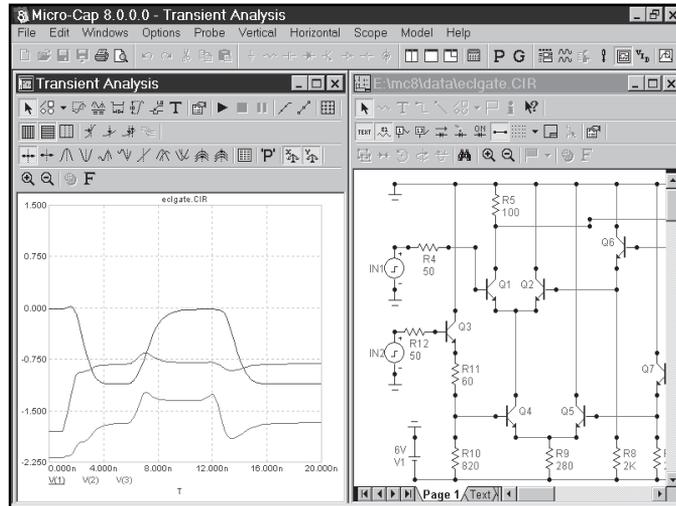


Figure 8-1 Probing node voltages

Probe first checks if the location is within the shape outline of a component. If so and if the component is a digital part, a subckt, or a macro, Probe presents a list of pin names and lets you select one. If the part is anything else (e.g. a standard analog part), Probe tries to match the selected vertical and horizontal variables, like voltage or current, with the variables available from the component. If there is a match, the variables are assigned and retrieved from the file. If there is no match, Probe checks to see if one of the selected variables is voltage and the location is on a non-ground node. If it is, it assigns the node's digital state or analog voltage to the variable and retrieves the values from the file.

To illustrate plotting lead-to-lead voltages, press CTRL + F9 to remove all curves. Click the mouse midway between the Q1 base lead dot and the Q1 emitter lead. This plots one of the curves, $V_{BE}(Q1)$ or $V_{EB}(Q1)$, depending upon which lead the probe was closest to. It may take some practice to get the feel for where to probe. Add the $V_{BC}(Q1)$ curve by clicking midway between the base and collector leads.

Probing lead-to-lead voltages

The lead-to-lead voltage curves look like this:

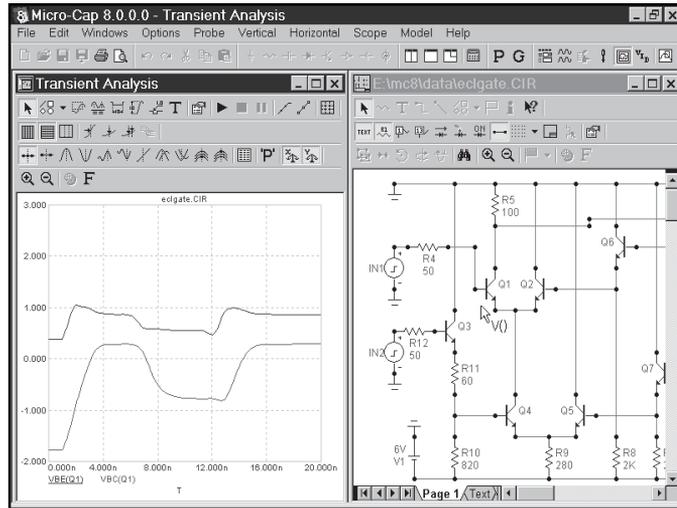


Figure 8-2 Probing lead-to-lead voltages

Lead currents can be plotted just as easily. Press SPACEBAR to switch from voltage to current. Press CTRL + F9. Click on the base and emitter leads of Q1. The resulting curves show the current into each of these leads:

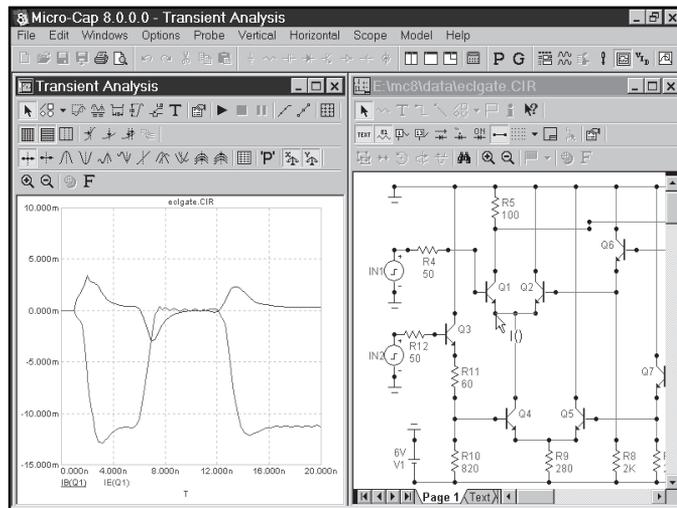


Figure 8-3 Probing lead currents

Navigating the schematic

Suppose you want to probe in an area of the schematic that is not visible? You can use any of the usual methods: scroll bars, mouse or keyboard panning, changing the scale, **SHIFT** + click relocation, or flag relocation.

To move around quickly use the **SHIFT** + click method. Press the **SHIFT** key and click the right mouse button anywhere in the circuit. The circuit shrinks. Move the mouse to another part of the circuit and press **SHIFT** + click. The scale reverts to normal and the view is centered at the mouse position.

Keyboard panning uses **CTRL** + any arrow key to move the view in the direction of the arrow. **CTRL** + **PGUP** and **CTRL** + **PGDN** move the view up or down one page. The circuit window must be selected to pan the circuit view.

Mouse panning is the best way of navigating a localized area of the schematic. To illustrate, click in the Probe display and press **CTRL** + **F9**. Press **SPACEBAR** to toggle back to voltage mode. Click the left mouse button on Q1's collector dot to plot **V(1)** vs. time. Point the mouse near the middle of the circuit window, click the right mouse button, and hold it down. Drag the mouse to the left until the mouse is near the window border. Release the button and repeat the procedure until the Q8 transistor is in view. Click on the dot by the node labeled Out. The screen should look like this:

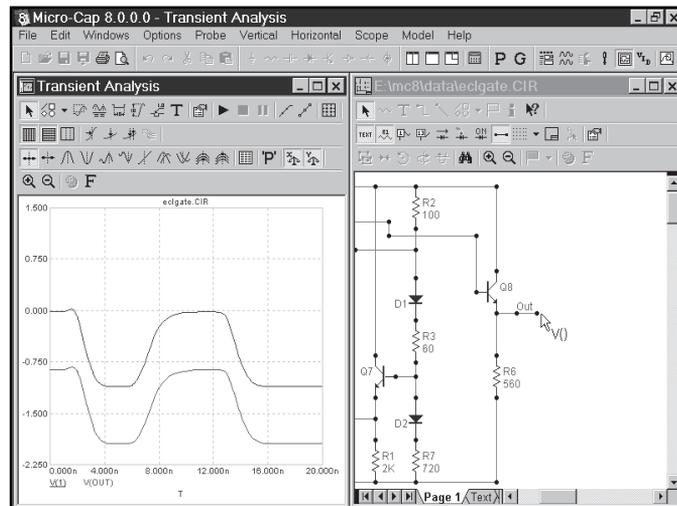


Figure 8-4 Curves from widely separated areas

An AC example

How about AC? The basic operation is the same, but there are differences. To illustrate the differences, press F3 to exit Probe and unload the ECLGATE circuit. Load the file DIFFAMP. Select **Probe AC** from the **Analysis** menu. Click on the **Vertical** menu. It looks like this:

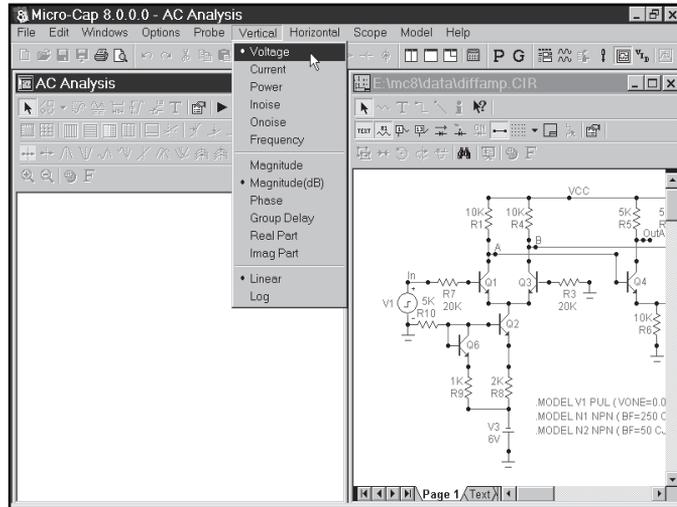


Figure 8-5 AC Probe variables and operators

The list of variables includes the usual voltage and current variables, but excludes charge, capacitance, flux, inductance, B, H, and time. There are several new variables; INOISE, ONOISE, and frequency. Additionally, the complex math operators, Magnitude, Magnitude (dB), Phase (PH), Group Delay (GD), Real Part (RE), and Imag Part (IM) are available. The short form is shown in parentheses. One of these operators is always selected. When you choose a circuit variable by clicking in the schematic, the selected operator is applied to the circuit variable and the result plotted. Magnitude is the default operator. For example, V(10) is the magnitude of V(10). PH(V(10)) is the phase of V(10). As in normal AC analysis, the noise variables can not be mixed with other variables, such as voltage or current.

Click on the dot near the node labeled A. Using the right mouse button, pan the schematic to the left by dragging in the schematic. Pan until the right portion of the diffamp is visible. Click on the dot near the node labeled OutA. Click on the Cursor mode button  to enable the Cursor mode. The results should look like Figure 8-6.

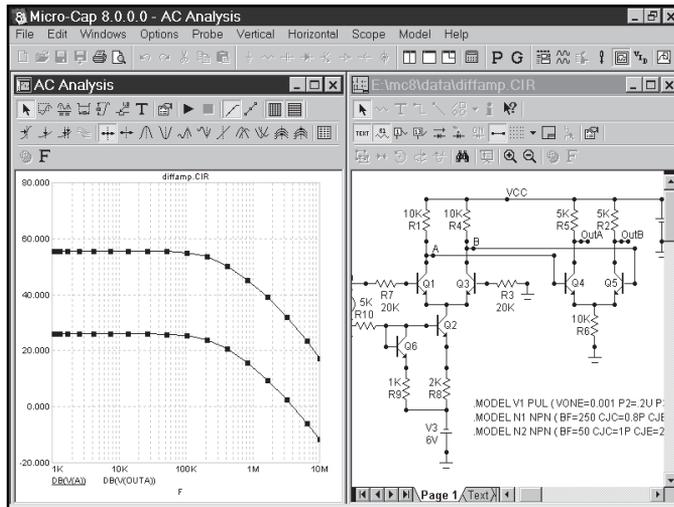


Figure 8-6 Reading AC gains using the Cursor mode

The left plot cursor is initially placed at the first data point, 1KHz. The low frequency gain at the two nodes A and OutA can be read out directly under the Left column. Use the left mouse button to move the left cursor to obtain additional readings. The right plot cursor is controlled by the right mouse button.

Press CTRL + F9. Select Magnitude, Log, and ONOISE from the **Vertical** menu and click anywhere in the schematic. Select INOISE from the **Vertical** menu and click anywhere in the schematic. This plots input and output noise:

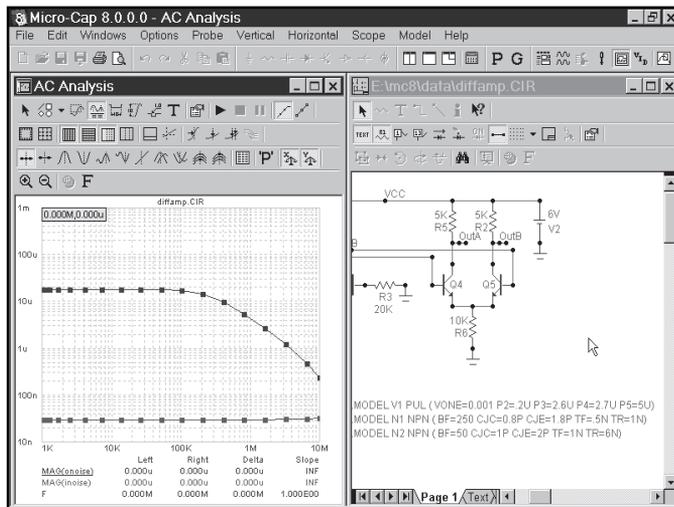


Figure 8-7 AC noise plots

A DC example

We'll use the file DIFFAMP to illustrate the use of Probe in DC. Exit the AC Probe routine by pressing F3. Select **Probe DC** from the **Analysis** menu. Click on the **Vertical** menu. It looks like this:

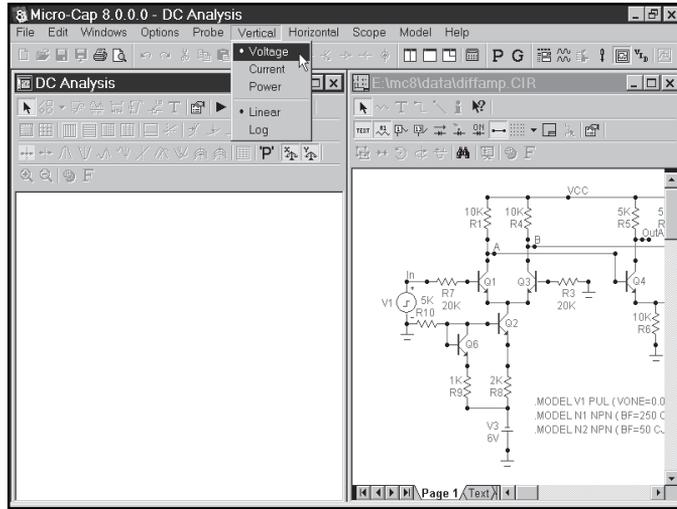


Figure 8-8 DC Probe variables

The list of available vertical variables includes voltage, current, and power. The horizontal variable is initially the voltage across the Input1 source when you start or press CTRL + F9 to clear the curves. The horizontal variable can be changed after the first curve has been placed by first selecting the plot window, and then pressing F10 to invoke the Properties dialog box. Simply type in a new expression into the X Expression field.

Press ESC to remove the **Vertical** menu.

Click on the dot near the node labeled OutA. Using the right mouse button, pan the schematic to the left by dragging in the schematic. Pan until the right portion of the diffamp is visible. Click on the dot near the node labeled 'OutB'. Click on the Cursor button  to enable Cursor mode. The results should look like Figure 8-9.

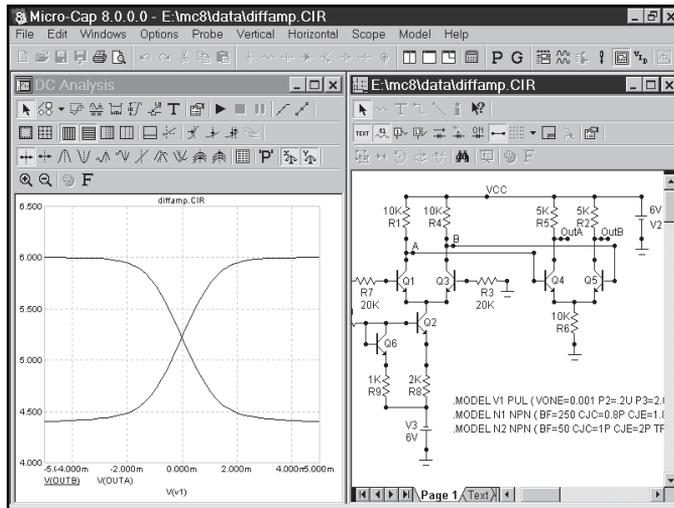


Figure 8-9 DC transfer function plot

Click on the Inflection  button. This moves the left plot cursor to the next inflection point to the right of its current position (first data point). Click the *right* mouse button in the plot area, then press SHIFT + LEFT ARROW key. This moves the right plot cursor to the next inflection point to the left of its current position (last data point). The peak gain 603, which occurs near the inflection point, can be read out directly in the Slope column. The inflection mode is useful for determining the peak DC gain of an amplifier.

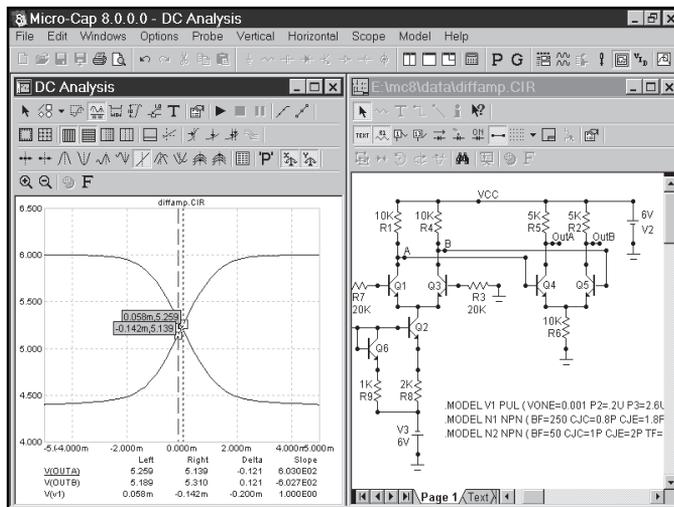


Figure 8-10 Using the cursors to find the peak gain

Probing a SPICE file

MC8 can also probe SPICE files. Exit DC Probe with F3 and load CHOKE.CKT. Select **Analysis / Probe Transient**. MC8 presents the Analysis Limits dialog box to let you change the run conditions if you want. Press ESC to remove the dialog box and click on the text V1 and again on C1. MC8 plots the voltage across the V1 source and the C1 capacitor.

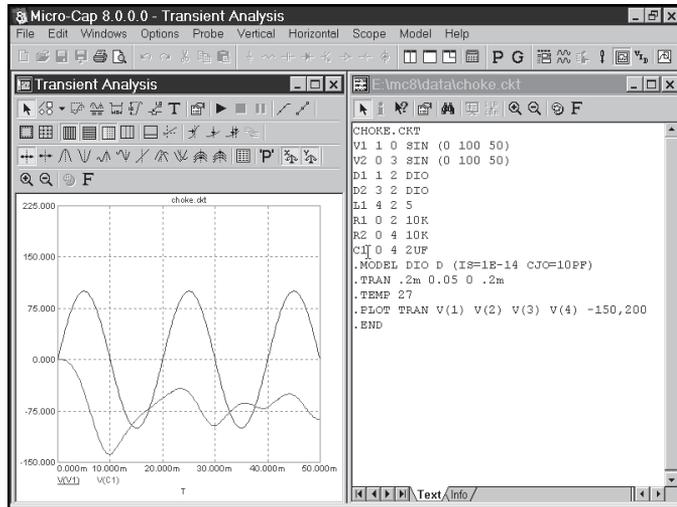


Figure 8-11 Probing SPICE circuits

Clicking on two-terminal devices like sources, resistors, diodes, and the like plots the voltage across or current through the device, depending upon whether the Vertical option is set to Voltage or Current.

Clicking on three-terminal devices, presents a dialog box where you can select the desired voltage or current variable.

Clicking on a node number plots the node voltage.

Tips for easy probing:

Here are three things to remember to make probing easier:

- Press the SPACEBAR key to quickly toggle between voltage probe mode and current probe mode.
- In voltage probe mode, press the SHIFT key and click on two nodes to get the differential voltage across the two nodes.
- In one curve mode, press CTRL while probing to add a new curve or delete an existing curve.

What's in this chapter

Stepping is the process of systematically changing one or more numeric parameters to see what effect the parameter(s) may have on circuit behavior. You can step simple parameters, like a resistor's value, or a model statement parameter like the forward beta of a transistor, or a symbolic parameter.

In this chapter, we demonstrate how to use stepping with several tutorials. The topics include:

- How parameter stepping works
- Stepping in transient analysis
- AC and DC examples
- Stepping summary

How parameter stepping works

Stepping and Monte Carlo are mutually exclusive. Only one may be active at a time.

Stepping systematically alters the value of one or more parameters of one or more components and then runs the analysis, drawing multiple branches for each curve. Most parameter types can be stepped, including component parameters like the resistance of a resistor, model parameters like a transistor beta, and symbolic parameters created with a .define or .param command. If the parameter changes the equation matrix, MC8 simply recreates the equations. For each parameter set, a run is made and the specified curves plotted.

To illustrate stepping, load the circuit DIFFAMP. Select **Transient** from the **Analysis** menu. Press F11, click on the Stepping button in the Analysis Limits dialog box, or click on the Stepping  button to display the Stepping dialog box. It looks like this:

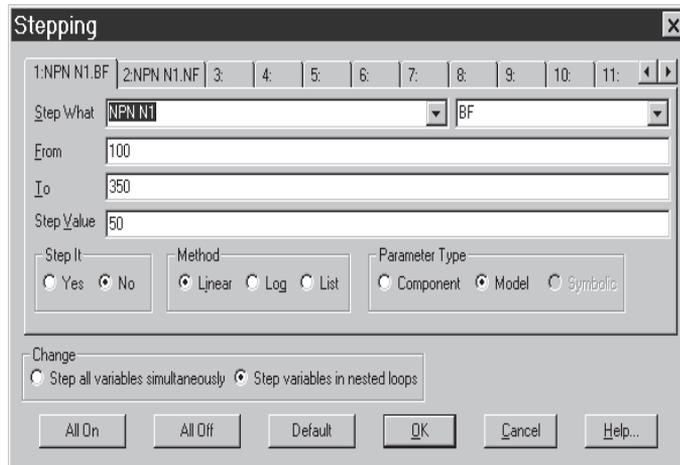


Figure 9-1 The Stepping dialog box

The dialog box provides tabs for as many as twenty parameters, although two or three is a practical maximum. Each tab accesses a panel which controls a single parameter. Stepping for each parameter is enabled when its Step It option is set to Yes.

The left Step What list box specifies the name of the model, component or symbol whose parameter or value is to be stepped. Since dissimilar parts can use the same model name, the electrical definition is shown along with the name. This can be seen in Figure 9-1 where the first item is NPN N1. This specifies that we want to step one of the parameters of all bipolar NPN transistors using the

model name N1. Clicking on the list box arrow displays a list of the models available for stepping. To select a model, click on it.

The right Step What list box specifies the name of the model parameter to be stepped. Clicking on the arrow displays a list of parameters available for stepping. To select a particular parameter, click on it. In this case we want to step the BF (forward beta) parameter, so leave it as it is.

The From field in each panel specifies the starting value of the parameter for the first run. The To field specifies the ending value of the parameter for the last run. The Step Value field specifies how much the parameter changes between steps.

The Method option in each panel controls how the Step Value affects the parameter value. Linear stepping adds the Step Value to the parameter value. Log stepping multiplies the parameter by the Step Value. In list stepping you enter a comma-delimited set of specific values in the From field.

The Parameter Type option in each panel specifies whether the Step What field refers to a model name, component name, or symbolic name. You can step an individual instance of a model parameter or all instances of the model parameter. If the Type option is set to Model, all instances are simultaneously stepped. If this option is set to Component and the PRIVATEANALOG / PRIVATEDIGITAL (**Options / Global Settings**) flags are enabled, then only the instance whose component name matches the Step What field will have its parameter stepped.

The Change option only comes into play when stepping multiple parameters. It controls whether the parameter changes are to be nested or simultaneous. For example, suppose you step R1 through the values 10 and 20. Suppose also that you step C2 through the values 1n and 5n. Simultaneous stepping will generate two runs and nested stepping will produce four runs as shown below.

<u>Nested</u>	<u>Simultaneous</u>
R1=10C2=1n	R1=10C2=1n
R1=10C2=5n	R1=20C2=5n
R1=20C2=1n	
R1=20C2=5n	

Simultaneous stepping requires an equal number of steps for each parameter. Use nested stepping when you want all combinations of parameter variation. Use simultaneous stepping when you want only specific combinations.

Stepping in transient analysis

To see how stepping works in transient analysis, click the Yes button in the Step section of Parameter panel 1, click OK, and press F2 to start the run. It looks like this:

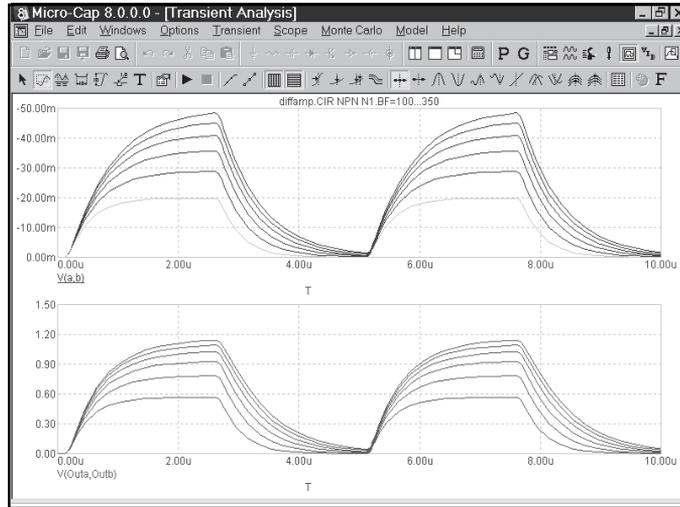


Figure 9-2 Stepping BF in four transistors simultaneously

In this run, each device which uses the N1 model, which includes the transistors Q1, Q3, Q4, and Q5, has its BF (forward beta) stepped through the values 100, 150, ... 350. To see which devices use N1, double-click on each device in the schematic to see its Attribute dialog box. From the results, it seems the offset voltage of V(OUTA,OUTB) is insensitive to large changes in BF, if all four devices have the same value.

Now we'll step the BF of just Q1 to see how the circuit tolerates small relative differences in beta. Press F11 to access the Stepping dialog box. Click on the Component item in the Parameter Type group. From the Step What list, click on Q1 and from the parameter list select BF. Type "250" in the From field, "255" in the To field, and "1" in the Step Value field. Click on the OK button and press F2 to start the run. The plot, shown in Figure 9-3, demonstrates the circuit sensitivity to differential beta. When Q1's BF changes from 250 to 255, while all other transistors have constant BF, large shifts occur in the output offset of both curves V(A,B) and V(OUTA,OUTB).

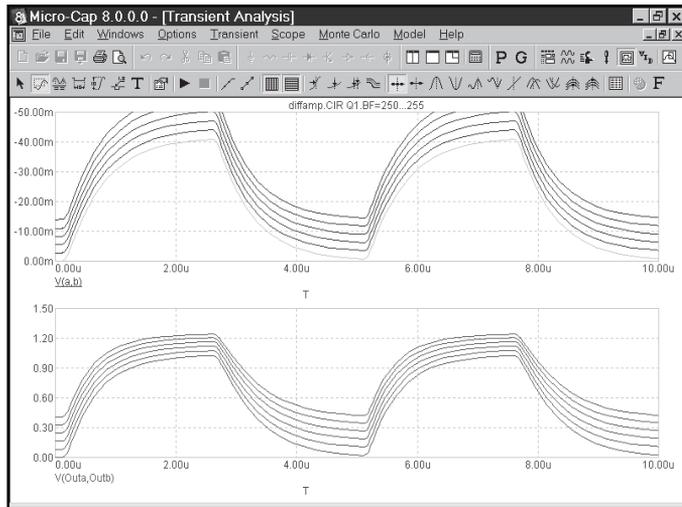


Figure 9-3 Stepping BF in a single transistor

Now we'll use CJE to illustrate log stepping. Press F11 to access the Stepping dialog box. Click on the Model button and the Log button. From the left list box select NPN N1, and from the right list box, select CJE. Type "2P" into the From field, "16P" into the To field, and "2" into the Step Value field. Click OK. Press F2 to start the run and it looks like this:

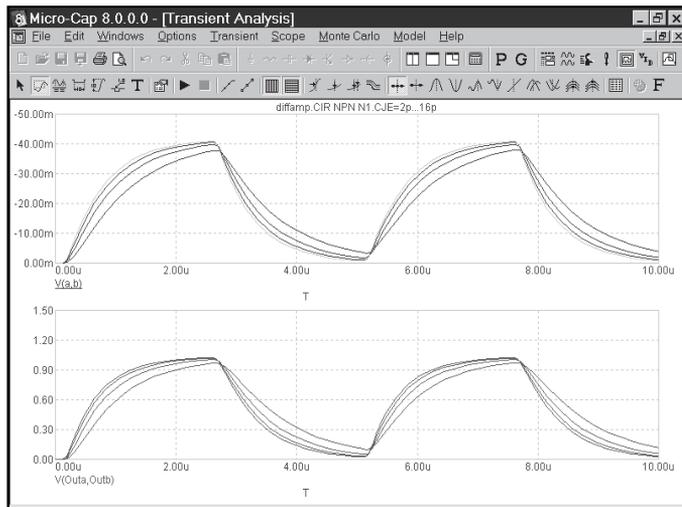


Figure 9-4 Log stepping CJE

AC and DC examples

Press F3. Select **Revert** from the **File** menu. Press ALT + 2 to select AC. Press F11, click the Yes button in the Step It group. Press ENTER. Press F2.

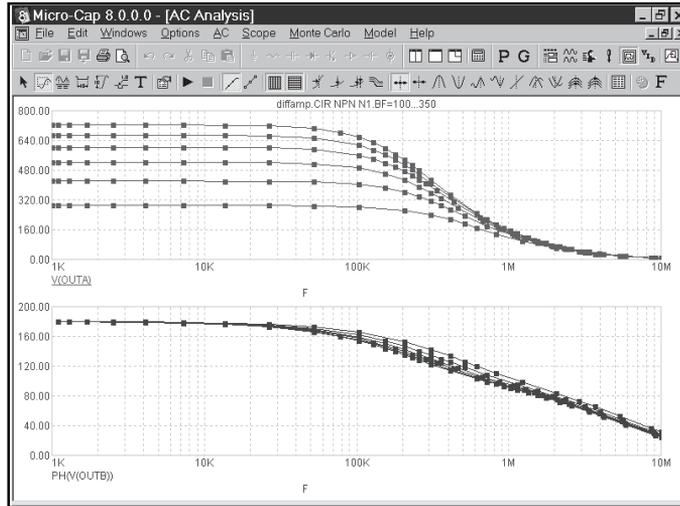


Figure 9-5 AC analysis linear stepping

To illustrate list stepping, press F11. Click the List option in the Method group. Type "100,110,150,160,200,210" in the From field. Press ENTER, then F2. The run looks like this:

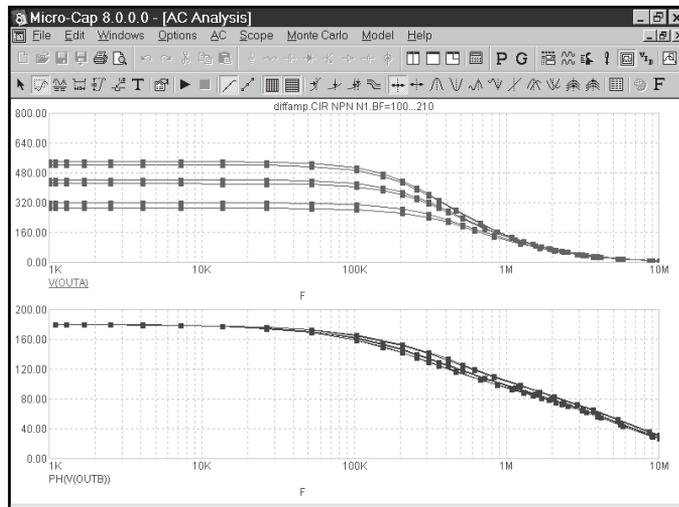


Figure 9-6 AC analysis list stepping

To illustrate stepping in DC, press F3 to exit the analysis. Press ALT + 3 for DC analysis. Press F11, click on the Linear option, Click on the Yes button, click OK, then press F2 to start the DC run. It looks like this:

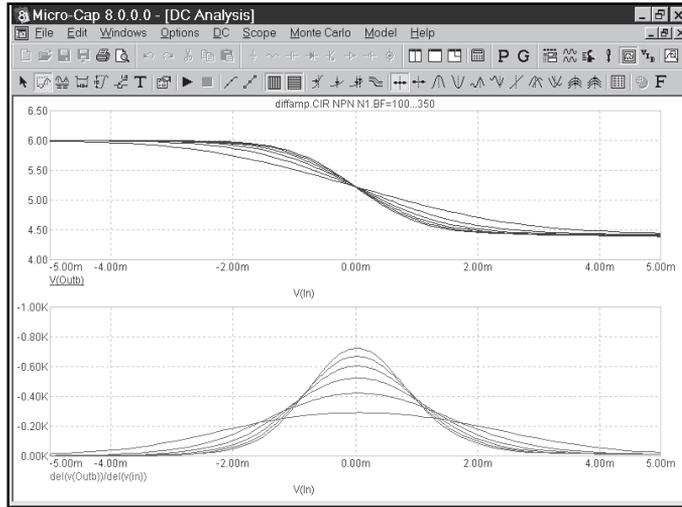


Figure 9-7 DC analysis linear stepping

To illustrate multiple parameter stepping, press F11, click in the Yes option in the Step It field of parameter 2. Press ENTER, then F2. The run looks like this:

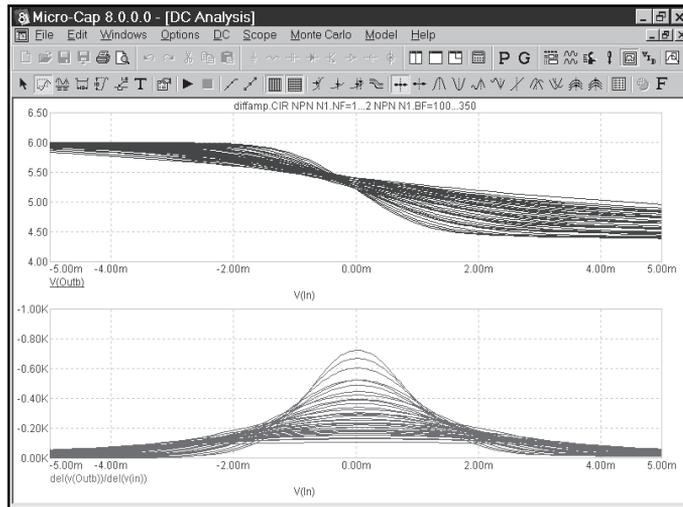


Figure 9-8 DC analysis stepping

Here we've simultaneously stepped BF and NF of all transistors using the N1 NPN model. Nested stepping was selected, and each parameter has six values, so $6 \cdot 6 = 36$ runs were done.

Text stepping

Text can also be stepped. Consider the circuit below:

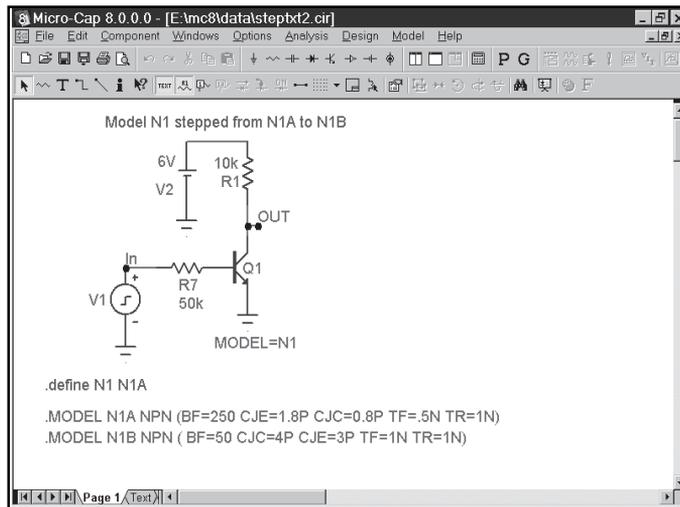


Figure 9-9 Stepping the N1 variable

Here Q1's model is set to N1, which is defined as N1A via the statement,

```
.DEFINE N1 N1A
```

If we step N1 through a list that includes N1A and N1B, we get two simulations. One is for the case when N1 is N1A and uses the N1A model parameter set. The second is for the case when N1 is N1B and uses the N1B model parameter set. The Stepping dialog box for this case is shown below.

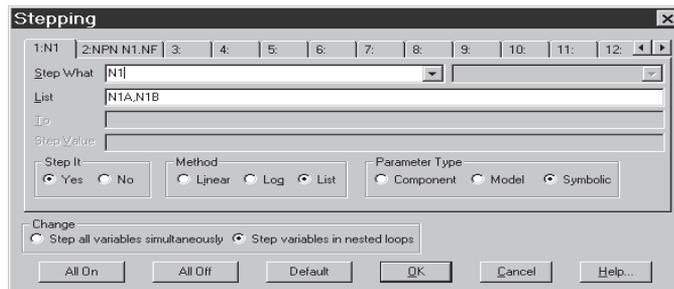


Figure 9-10 The Stepping dialog box for text stepping the N1 variable

Notice two things about text stepping:

- 1) Text stepping is done with a symbolic variable that is created and defined with a `.DEFINE` statement.
- 2) Text stepping is done with the List method.

Here is what the simulation looks like for the specified stepping.

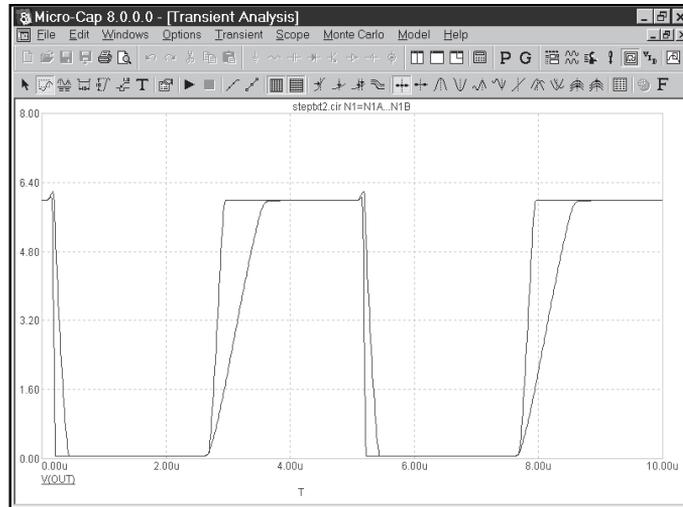


Figure 9-11 The simulation run

The two runs use two different model parameter sets and produce two different output curves.

Stepping summary

The most important things to remember when using stepping include:

1. The parameters of these components may not be stepped:
 - Transformer
 - User sources
 - Laplace sources
 - Function sources
 - SPICE dependent sources (E,F,G,H sources)
 - MC4 switch, which has been largely replaced by the S and W switches. Their parameters can be stepped.

The behavior of User, Laplace, Function, and SPICE sources is embodied in algebraic formulas and numeric tables, and thus can't be stepped. You can, however, step symbolic parameters which in turn are used in Laplace, Function, and SPICE source functions. For example, you could use,

```
.DEFINE TAU 5
```

A Laplace function source like,

```
1/(1+TAU*S)
```

could then be changed by stepping TAU.

Similarly in a Laplace table source like,

```
.DEFINE RVAL 2.0
```

```
.DEFINE TAB (1k,0,RVAL)
```

you could change the behavior of the source by stepping RVAL.

The User source gets its data from an external data file and has no parameter that can be stepped.

2. In Component mode, stepping affects one parameter of one device only if the PRIVATEANALOG and PRIVATEDIGITAL options (Global Settings) are enabled. In Model mode, stepping affects one parameter of all devices

that use that model name. Thus you are potentially affecting many devices. This is true regardless of the state of the PRIVATEANALOG or PRIVATEDIGITAL flags.

3. The MOSFET Level model parameter may be stepped, but an error will occur if the model statement has been created with parameters for level 1, 2, or 3 and the level changes to a short channel BSIM or EKV model (4, 5, 8, 14, or 49). These models use parameter names that are so different from the level 1, 2, and 3 models that the parameter names would be meaningless.

4. Linear stepping starts with the From value and adds the Step Value until it reaches the To value. Log stepping starts with the From value and then multiplies by the Step Value until it reaches the To value. A Step Value of 2 is often convenient and is called octave stepping. A Step Value of 10 is sometimes referred to as decade stepping. List stepping simply uses the values specified in the From field.

5. If multiple parameters are to be *simultaneously* stepped, they must each specify the same number of steps. If there is a mismatch, an error message is generated.

6. At least two parameters must be varied to create 3D performance plots.

7. Stepping a resistor, capacitor, or inductor that uses an expression for its value, replaces the value calculated from the expression with the step value. In other words, the step value takes precedence over the calculated value.

8. Text stepping requires a variable created with a .DEFINE statement and the List method must be used.

What's in this chapter

Monte Carlo analysis statistically tests circuit performance. It creates a batch of circuits populated with toleranced components. After the analysis of each circuit, it employs user-selected functions to extract performance data from curves plotted or printed during the run. The results are statistically analyzed and plotted in the form of histograms. Careful review of the data can yield valuable insights into the reliability, cost, and the ability to manufacture the circuit.

Monte Carlo statistical analysis is available in transient, AC, and DC analysis.

Stepping and Monte Carlo are mutually exclusive. Either one, but not both, may be run simultaneously.

The chapter is organized as follows:

- How Monte Carlo works
- Distributions
- Performance functions
- Options
- An example
- Statistical summary

How Monte Carlo works

Monte Carlo works by analyzing many circuits. Each circuit is constructed of components randomly selected from populations matching the user-specified tolerances and distribution type.

Tolerances are applied to a component's numeric model parameters. *Only model parameters and symbolic parameters can be tolerated.* Tolerances are specified as an actual value or as a percentage of the nominal parameter value.

Both absolute (LOT) and relative (DEV) tolerances can be specified. A LOT tolerance is applied absolutely to each device. A DEV tolerance is then applied to the first through last device relative to the LOT tolerated value originally chosen for the first device. In other words, the first device in the schematic list receives a LOT tolerance, if any. All devices, including the first, then receive the first device's value plus or minus the DEV tolerance. DEV tolerances provide a means for having some devices track in their critical parameter values.

Both tolerances are specified by including the key words LOT or DEV after the model parameter:

```
[LOT[t&d]=<value>[%]] [DEV[t&d]=<value>[%]]
```

For example, this model statement specifies a 10% absolute tolerance to the forward beta of the transistor N1:

```
.MODEL N1 NPN (BF=300 LOT=10%)
```

In this example, for a worst case distribution, each transistor using the N1 model statement has a forward beta of either 270 or 330. If a Gaussian distribution were used, a random value would be selected from a Gaussian distribution having a standard deviation of 30. If a uniform distribution were used, a random value would be selected from a uniform distribution having a half-width of 30.

This example specifies a 1% relative tolerance to the BF of the N1 model:

```
.MODEL N1 NPN (BF=300 DEV=1%)
```

The DEV value specifies the relative percentage variation of a parameter. A relative tolerance of 0% implies perfect tracking. A 1.0% DEV tolerance implies that the BF of each N1 device is the same to within 1.0%. Model statements that

use DEV tolerances force the use of private libraries, regardless of the state of the PRIVATEANALOG or PRIVATEDIGITAL flags.

This sample specifies a 10% absolute and 1% relative BF tolerance:

```
.MODEL N1 NPN (BF=300 LOT=10% DEV=1%)
```

In this example, assuming a worst case distribution, the first N1 model is randomly assigned one of the two values 270 or 330. These two values are calculated from the mean value of 300 and 10% LOT tolerance as follows:

$$BF = 270 = 300 - .1 \cdot (300)$$

$$BF = 330 = 300 + .1 \cdot (300)$$

Suppose that the LOT tolerated BF value was randomly chosen to be 330. Then all N1 transistors, including the first, are randomly given one of these values, based upon the 1% DEV tolerance:

$$327 = 330 - .01 \cdot 300$$

$$333 = 330 + .01 \cdot 300$$

If the LOT tolerated BF value had been randomly chosen to be 270, then all N1 transistors, including the first, would be randomly given one of these values, based upon the 1% DEV tolerance:

$$267 = 270 - .01 \cdot 300$$

$$273 = 270 + .01 \cdot 300$$

Assuming a worst case distribution, all BF values in any particular run would be chosen from the set {267, 273} or the set {327, 333}.

Resistors, capacitors, and inductors can only be tolerated through their multiplier model parameter. The example below provides a 5% LOT tolerance and a .2% DEV tolerance for a capacitor.

```
.MODEL CMOD CAP(C=1 LOT=5% DEV=.2%)
```

Any capacitor that uses the CMOD model will be tolerated, since the tolerated C multiplier will multiply its capacitance value.

See the "Monte Carlo Analysis" chapter in the Reference manual for further information on using tolerances.

Distributions

The actual values assigned to a tolerated parameter depend not only on the tolerance, but on the distribution as well.

A worst case distribution places all values at the extremes of the tolerance band. There are only two values:

$$\text{Min} = \text{Mean} - \text{Tolerance}$$

$$\text{Max} = \text{Mean} + \text{Tolerance}$$

The mean value is the model parameter value.

A uniform distribution places values equally over the tolerance band. Values are generated with equal probability over the range:

From *Mean - Tolerance* to *Mean + Tolerance*

A Gaussian distribution produces a smooth variation of parameters around the mean value. Values closer to the mean are more likely than values further away. The standard deviation is obtained from the tolerance by this formula:

$$\text{Standard Deviation} = \text{Sigma} = (\text{Tolerance}/100) \cdot \text{Mean}/\text{SD}$$

SD (from the Global Settings dialog box) is the number of standard deviations in the tolerance band. Thus, the standard deviation is calculated directly from the tolerance value. The value chosen depends upon how much of a normal population is to be included in the tolerance band. Here are some typical values:

Standard deviations	Percent of population
1.00	68.0
1.96	95.0
2.00	95.5
2.58	99.0
3.00	99.7
3.29	99.9

If you're certain that 99.9% of all 10% resistors are within the 10% tolerance, you would use the value 3.29. Using a Gaussian distribution, a 1K 10% resistor may have a value below 900 ohms or above 1100 ohms. The probability would be less than 0.1% for an SD of 3.29, but it could happen.

Performance functions

MC8 saves all X and Y expression values of each plotted or printed expression at each data point for each run, so you can create histograms after the runs are done by picking a performance function and an expression. The performance function calculates a single number from the selected expression for each analysis run. All the numbers are combined to form a population which is statistically analyzed and its histogram is displayed. The performance functions supplied with MC8 include:

- Rise_Time
- Fall_Time
- Peak_X
- Peak_Y
- Valley_X
- Valley_Y
- Peak_Valley
- Period
- Frequency
- Width
- High_X
- High_Y
- Low_X
- Low_Y
- X_Level
- Y_Level
- X_Delta
- Y_Delta
- X_Range
- Y_Range
- Slope
- Phase_Margin

For a more detailed description of how the performance functions work see the "Performance Functions" chapter in the Reference Manual.

Options

The Monte Carlo options dialog box provides these choices:

- **Distribution to Use:** This choice controls the type of distribution used to generate the individual populations of component parameters.

- *Gaussian distributions* are governed by the standard equation

$$f(x) = e^{-.5*s*s/\sigma(2*\pi)^{.5}}$$

Where $s = (x - \mu) / \sigma$ and μ is the nominal parameter value, σ is the standard deviation, and x is the independent variable.

- *Uniform distributions* have equal probability within the tolerance limits. Each value from minimum to maximum is equally likely.
- *Worst case distributions* have a 50% probability of producing the minimum and a 50% probability of producing the maximum.
- **Status:** Monte Carlo analysis is enabled by selecting the On button. To disable it, click the Off button.
- **Number of Runs:** The number of runs determines the confidence in the statistics produced. More runs produce a higher confidence that the mean and standard deviation accurately reflect the true distribution. Generally, from 30 to 300 runs are needed for a high confidence. The maximum is 30000 runs.
- **Show Zero Tolerance Curve:** If this option is enabled, the first run tolerances are set to zero to provide a kind of baseline or reference curve.
- **Report When:** This field specifies when to report a failure. The routine generates a failure report in the numeric output file when the Boolean expression in this field is true.
- **Seed:** The random number seed directly controls the sequence of random numbers generated by the program. By specifying an integer seed number greater than or equal to 1, you can identify and recall in a later simulation the same random numbers used to produce tolerance values and the corresponding histograms.

An example

To illustrate the Monte Carlo features, load the file CARLO4.CIR. This file contains a simple pulse source driving an RLC network. The circuit looks like this:

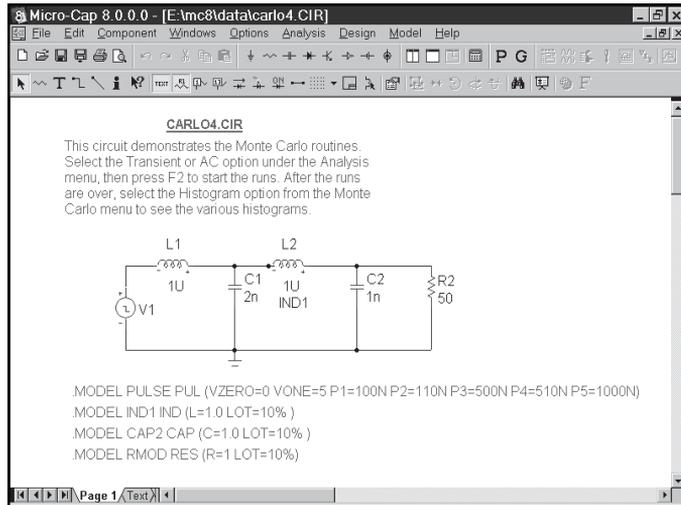


Figure 10-1 The CARLO4 circuit

Select transient analysis and then using the mouse, click the **Options** item from the **Monte Carlo** menu. It should look like this:

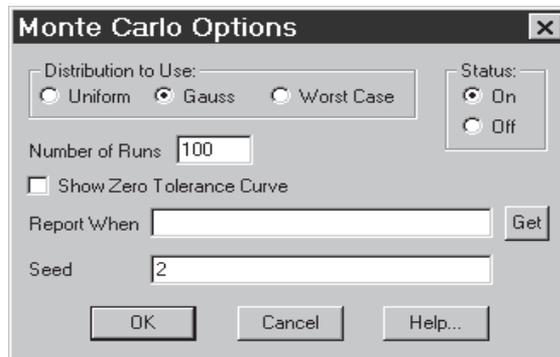


Figure 10-2 Monte Carlo options

The Monte Carlo options are set to run 100 runs using a Gaussian distribution. A seed of 2 is used. Click on the OK button and press F2 to start the runs. The curves accumulate on the screen and the result looks like this:

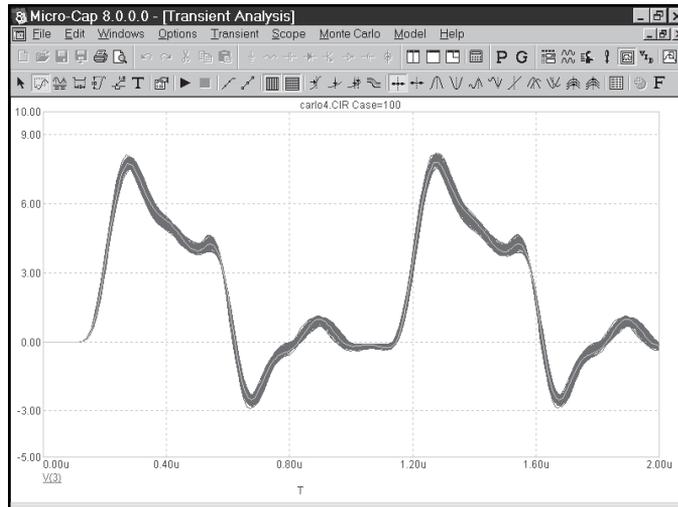


Figure 10-3 The analysis run

Select **Monte Carlo - Histograms - Show Rise_Time(V(3),1,1,2)**. This selects the existing Rise_Time histogram which looks like this:

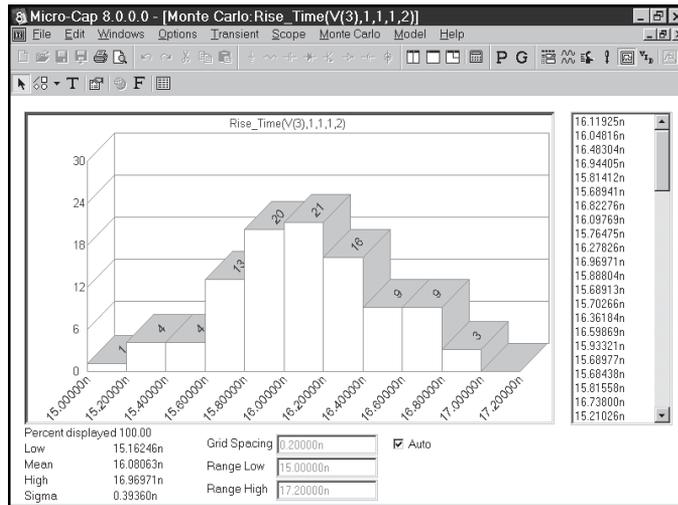


Figure 10-4 The Rise_Time histogram

This figure shows the frequency distribution for the Rise_Time performance function. The function is specified for the curve V(3), with a Boolean of 1, which selects all data points, an N value of 1 which searches for the first rise time in the V(3) curve, a Low value of 1.0, which defines the first level from which the rise time is to be measured, and a High value of 2.0, which specifies the second level from which the rise time is to be measured. The Rise_Time function measured one value from each of the 100 curves and the results are shown here in histogram form. The statistical summary at the lower left shows the lowest value, highest value, standard deviation, and mean, or average value, of the 100 numbers. The list to the right lets you scroll through individual rise time values. Grid Spacing sets the histogram bar width and indirectly determines the number of bars. Range High and Range Low can be changed to show the percentage of the total distribution included within the limits, predicting how these limits would affect production yields.

Now we'll add a new histogram to measure the width of the pulse. Select **Monte Carlo - Histograms - Add Histogram**. This presents the Properties dialog box, which you use to select a performance function. Click on the Get button and select the Width performance function. Click OK. Click OK again in the Properties dialog box and a new histogram is presented.

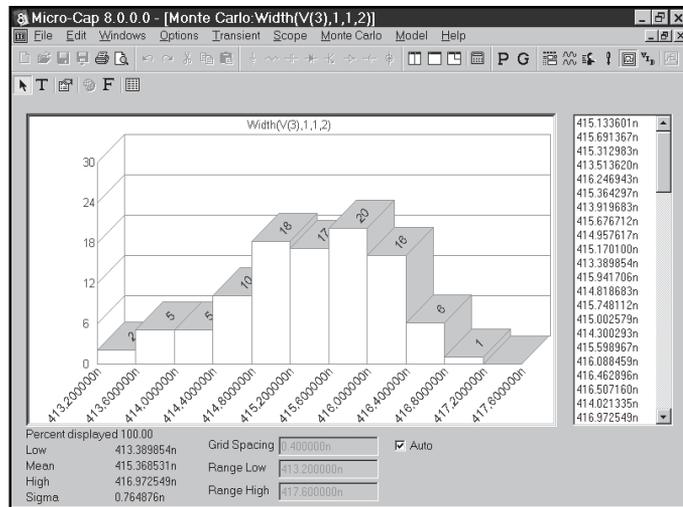


Figure 10-5 The Width histogram

The Width function is specified for the curve V(3), with a Boolean of 1, which selects all data points, an N value of 1 which searches for the first pulse in the V(3) curve, a Level of 2.0, which defines the level at which the width of the pulse is to be measured.

Press F3 to exit. Select **AC** from the **Analysis** menu. Press F2 and the screen should look like this:

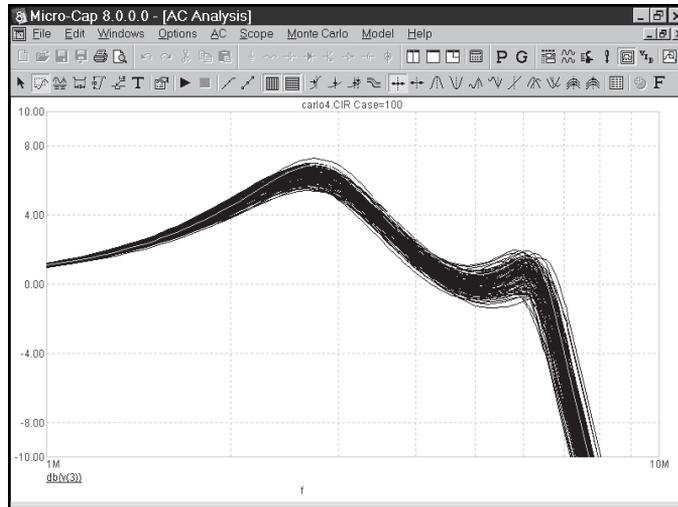


Figure 10-6 100 plots of DB(V(3))

Select **Add Histogram** from the **Histograms** item on the **Monte Carlo** menu. Use the Get button to select the **X_Level** function for the **What To Plot** field. Click in the **Y Level** field and type -3. Click OK. Click OK again in the Properties dialog box and a new histogram is presented.

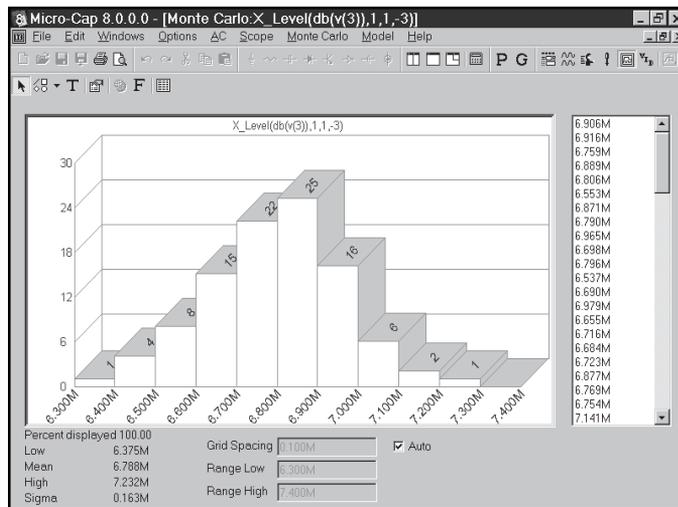


Figure 10-7 AC bandwidth histogram using the X_Level function

The function is specified for the curve DB(V(3)), with a Boolean of 1, which selects all data points, an N value of 1 which finds the first X value instance in the curve, a Y Level of -3.0, which defines the level at which the X value of the curve is to be measured. Since the curve is a plot of the gain of the circuit, which is a simple low pass filter, and since we chose -3 for the level, the performance function measures the 3dB bandwidth of the filter.

Press F10 to show the Properties dialog box. It includes items similar to the Plot Properties dialog boxes for the analysis plots.

Press Cancel to close the dialog box.

Press F3. Select **DC** from the **Analysis** menu. The Analysis Limits look like this:

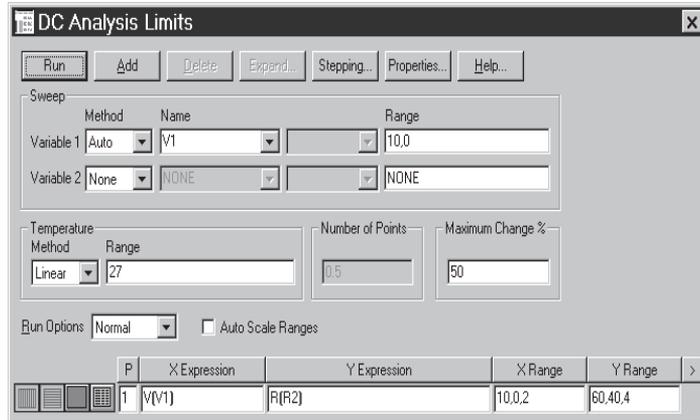


Figure 10-8 The DC analysis limits

These limits specify that we plot R(R2) while sweeping V1. V1 has no effect on R(R2). As a result, the plots are a series of straight lines. Press F2 to start the

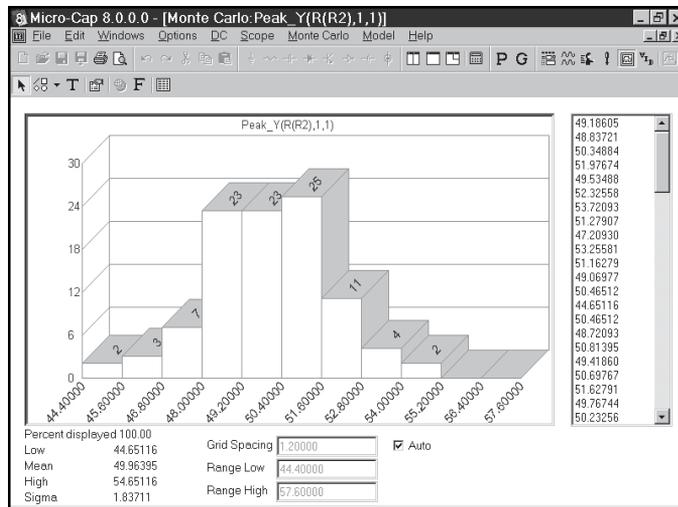


Figure 10-9 A Gaussian distribution

run. Select the **Show Peak_Y** item from **Monte Carlo - Histograms**. The Peak_Y function measures the largest value of R(R2), which is simply the tolerated resistor value. The distribution approximates a Gaussian distribution.

The Uniform distribution option produces a histogram like this:

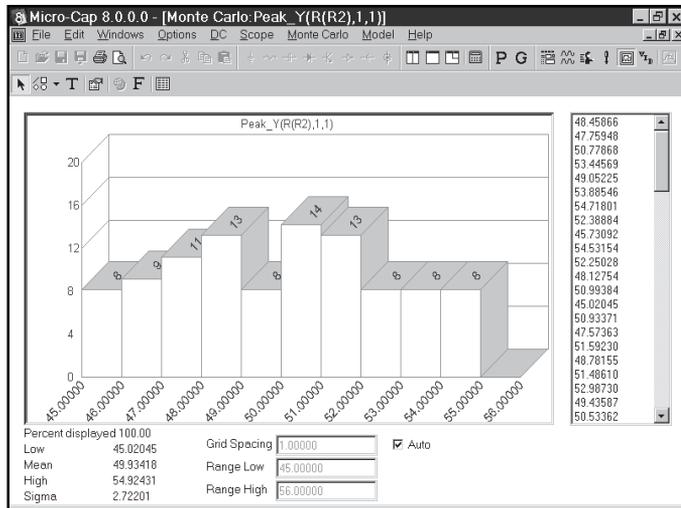


Figure 10-10 A uniform distribution

The Worst Case distribution option produces a histogram like this:

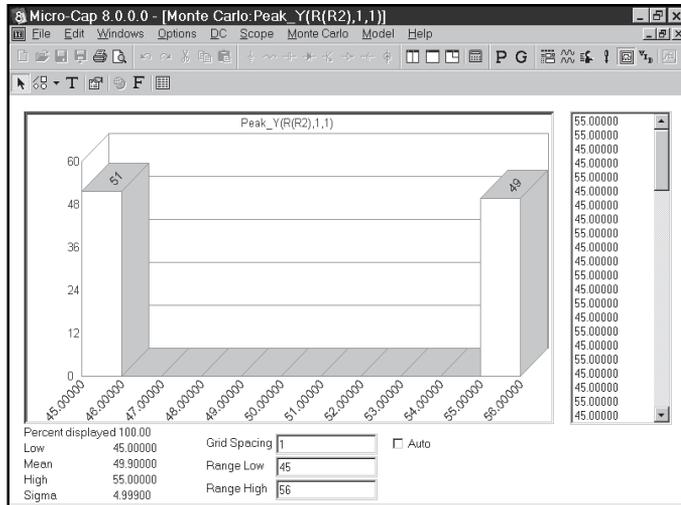


Figure 10-11 A worst case distribution

In each of the three cases, we plotted the resistance value directly, so each output histogram directly mirrored the distribution function used to determine its value.

The Gaussian distribution for the resistor tolerance resulted in a similar, Gaussian output distribution. For a large number of cases, the histogram closely approaches the bell shape of the standard Gaussian equation.

The uniform distribution for the resistor produced a similar uniform output distribution, although with a few bumps. If we had run a very large number of cases, we would expect a perfectly flat histogram, reflecting an equal probability of all outputs in the range.

The worst case distribution generated an output distribution that contained the two extremes. The proportion that fall into the low or high category is due to chance. The larger the number of runs, the greater the chance the proportions will be equal.

In more complex designs having many components with tolerances, the output distributions approach a standard Gaussian shape, even if the input distribution is chosen to be uniform or worst case, due to the Central Limit Theorem of statistics. It simply takes more cases to produce a normal-looking output distribution.

Press F3 to exit the analysis, and close the file with CTRL + F4.

Statistical summary

The statistical results are printed to a disk file. The file names used are:

Transient analysis	CIRCUITNAME.TMC
AC analysis	CIRCUITNAME.AMC
DC analysis	CIRCUITNAME.DMC

The content of the file may be viewed by selecting **Statistics** from the **Monte Carlo** menu when a histogram is the active window.

What's in this chapter

This chapter describes the use of macros. The term macro is defined and the four steps of macro creation and usage are described. They are:

- Creating the macro circuit file
- Selecting or creating a suitable shape for the macro
- Entering the macro into the Component library
- Using the macro in a circuit

What is a macro?

Macros are circuits, created and saved to disk to be used as building blocks by other circuits. The main advantage of macros is that the behavior of a complex circuit block can be incorporated into and represented by a single component. The complexity and detail of the macro is hidden from view in the calling circuit.

When an analysis is run, the macro file is loaded from disk and substituted for the macro. Macro circuits are constructed in the usual way, with two exceptions.

- 1) Macros use command text of the form,

`.PARAMETERS(par1,par2,...),`

to define parameters to be passed to the macro by the calling circuit.

- 2) Macros use grid text on circuit nodes to define the nodes (pin names) that are to be connected to the using circuit.

Creating the macro circuit file

The first step in using macros is to create the macro circuit itself. We won't do that here. Instead we'll use a macro circuit which is supplied with MC8. Load the macro circuit INT. It is an integrator circuit and looks like this:

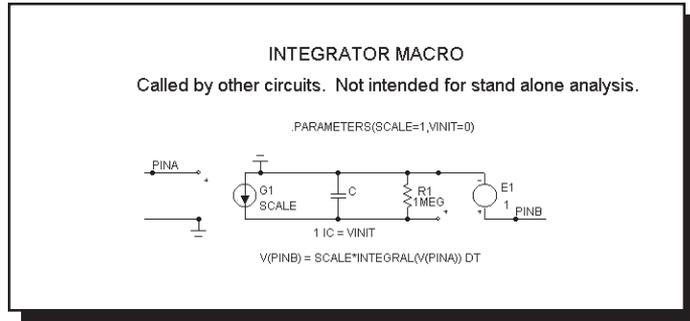


Figure 11-1 The INT macro

The first stage of the macro circuit is a voltage-controlled current source (IOFV). Its transconductance value, **SCALE**, is passed as a numeric parameter by the calling circuit. The circuit which uses the macro is referred to as the calling circuit. The first stage multiplies the input signal by the value **SCALE** and converts it to a current. The current flows directly into a capacitor, creating a voltage that is the integral of the scaled input signal. A final unity-gain stage buffers the capacitor voltage from the load in the calling circuit. The high-valued resistor avoids infinite voltages on the capacitor when DC input voltages occur. It also limits the useful range of the macro to frequencies above $1\text{E}-6$, which would be acceptable for virtually all applications.

This circuit illustrates the two features which distinguish macros from ordinary circuits. First, the external pin connections are defined by placing the pin name in the form of grid text on the node where connection is desired. In the INT circuit, the text **PINA**, connects the positive input of the voltage-controlled current source to **PINA** of the macro shape. The text **PINB**, connects the output of the buffer stage to **PINB** of the macro shape.

The second important feature in this macro is that it receives parameters from the calling circuit. A macro need not have passed parameters, but most do. It can greatly enhance a macro's usefulness. The first step in passing parameters to a macro is to include the **.PARAMETERS** control statement in the macro circuit. This is done by including a piece of text in the macro circuit using this format:

Entering the macro into the Component library

The third step is to enter the macro in the Component library. To do this, select the **Component Editor** from the **Windows** menu. From the Component Selector, double-click on the Macros name in the Analog Primitives group. In this hierarchical selector, double-clicking opens a closed group or closes an open group. Click on the INT name. This selects and displays the INT component. If you were adding INT as a new macro you would have clicked first on the Macro group to select it, then clicked on the Add Part  button to add INT to the macro group. The INT component display looks like this:

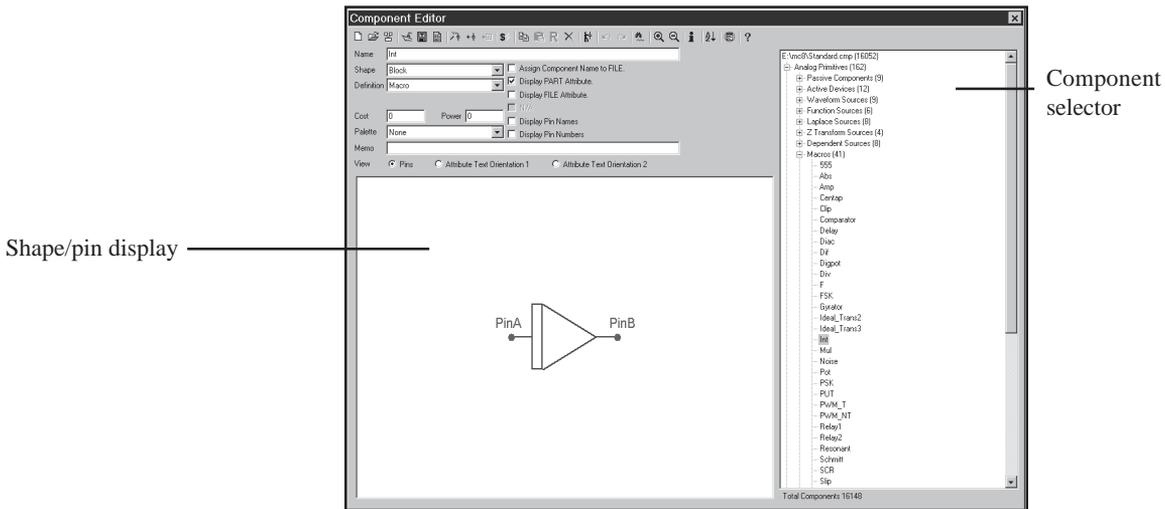


Figure 11-3 Component library entry for the INT macro

The Name data field is for the component name. This is the same as the macro file name. In this case we used the name INT.

The Shape data field is for the name of the shape to be used to represent the macro. The shape names are from the Shape library, which is created and maintained by the Shape editor. To select a shape, click in the Shape field and press the first letter of the shape name until the one you want appears, or use the arrow adjacent to the list to browse. Here we have chosen the Block shape.

The Definition field specifies the electrical definition. To select a definition, click in the field and press the first letter of the definition name until the one you want appears, or use the arrow to browse. In this case, we must specify Macro to indicate that the electrical definition comes from a circuit file.

The Memo field is for descriptions or comments. It has no effect on the electrical behavior of the part.

The Attribute text orientation options show the location of attribute text relative to the shape. Each shape has potentially eight combinations of rotation and reflection, called orientations. Orientation 1 is for the four horizontal orientations. Orientation 2 is for the four vertical orientations. Attribute text for a macro component is comprised of the part name, macro name, and optional parameters. These are entered via the Attribute dialog box when the component is added to a schematic. You can drag the text with the left mouse button to indicate where the first (XX) and second (YY) attribute text will be initially placed. After the component is initially placed in a circuit, the attribute text can be relocated by dragging the text attribute with the mouse.

An important aspect of entering a macro into the library is defining where the pins are. For a component like a diode, the pin names are predefined as Anode and Cathode.

For a macro, pin names are defined by the user. They must match the actual node names as defined by grid text in the macro circuit.

The names are defined by clicking in the Shape / pin display. This activates a Pin dialog box, allowing you to enter a new pin name and to specify if it is digital or analog. Double-clicking on an existing pin lets you edit it. Both the pin marker dot and the pin name can be independently moved with the mouse to new locations.

Using the macro in a circuit

The final step is using the macro. This is the easiest part. Once the macro has been entered into the Component library, it is available for use in circuits. It is accessed from one of the groups on the **Component** menu.

To illustrate using a macro, we'll create a simple circuit that contains a voltage source with a triangular waveform driving an INT macro. The output of the INT macro should be the integral of the input waveform.

Close the Component Editor without saving any changes. Close the INT circuit. Select the **New** option from the **File** menu, then Schematic from the New dialog box. From **Component Menu / Analog Primitives / Waveform Sources** select the **Voltage Source**. Place it in the schematic with the plus sign pointing up. From the Attribute dialog box, select the Pulse panel, then select the TRIANGLE button. Click on the OK button. Click the Wire mode button. Draw a wire from the top of the pulse source to the right.

From the **Macros** group of the **Analog Primitives** section of the **Component** menu, select INT. Place the INT macro input at the end of the line just drawn. Click on the SCALE attribute and type "1E7" for its parameter. Click on the VINIT attribute and type "1" for its parameter. This passes a scale value of 1E7 and an initial value of 1 to the macro. Click OK. Finally, click on the Ground button in the upper tool bar, rotate it as needed with the *right* mouse button and place it at the minus lead of the Pulse source.

The key points about placing a macro in a circuit are:

1. Remember where you put it in the Component hierarchy. Macros are usually placed in and selected from the **Analog Primitives / Macros** section of the **Component** menu, but their location is up to you.
2. The FILE attribute contains the name of the macro circuit file without its extension. In this case it is simply INT. This name must be the exact same name that the macro circuit was saved, without the .CIR or .MAC extension.
3. Each parameter is entered as a separate attribute.

The circuit and its transient analysis are shown in the figures below. To produce this analysis plot you must 1) disable the operating point, 2) enable the Auto Scale option and 3) enter 1 and 2, respectively, in the V(1) and V(2) P columns.

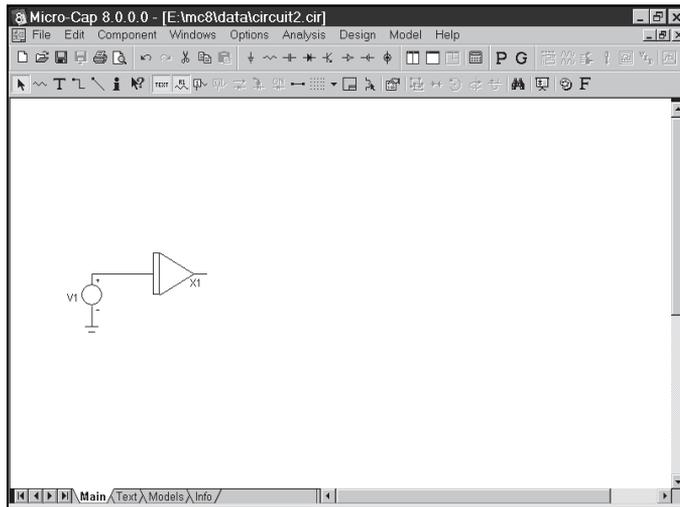


Figure 11-4 The INT macro used in a circuit

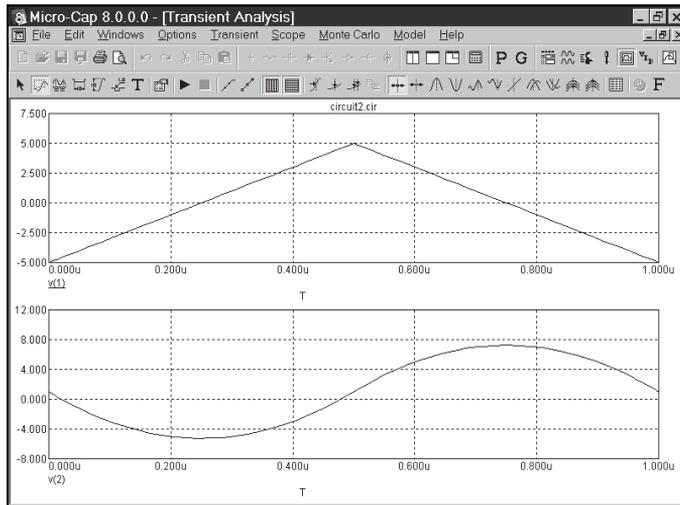


Figure 11-5 Transient analysis of the circuit

Press F3, and close the circuit.

An easy way to create macros

Another way to make macros from existing circuits is to use the macro creation command. To illustrate, load the file ECLGATE. Enter Select mode, and drag a box region across the circuit like this:

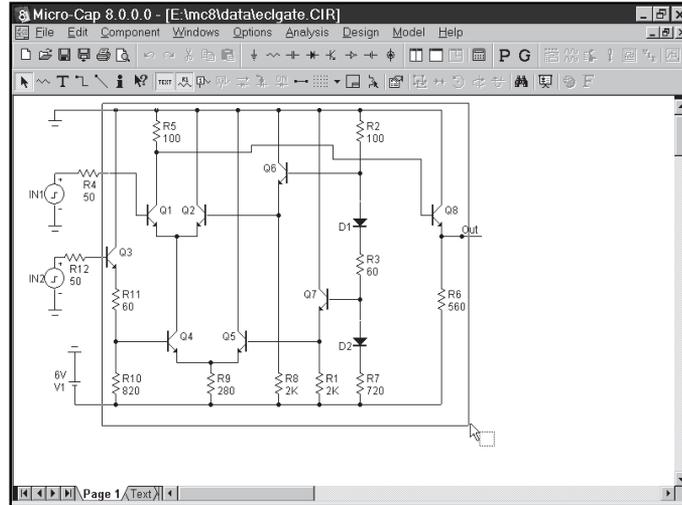


Figure 11-6 The ECLGATE macro box region

Be sure to draw the box region exactly as shown above. Now press CTRL + M. Click OK in the dialog box. The screen should look like this:

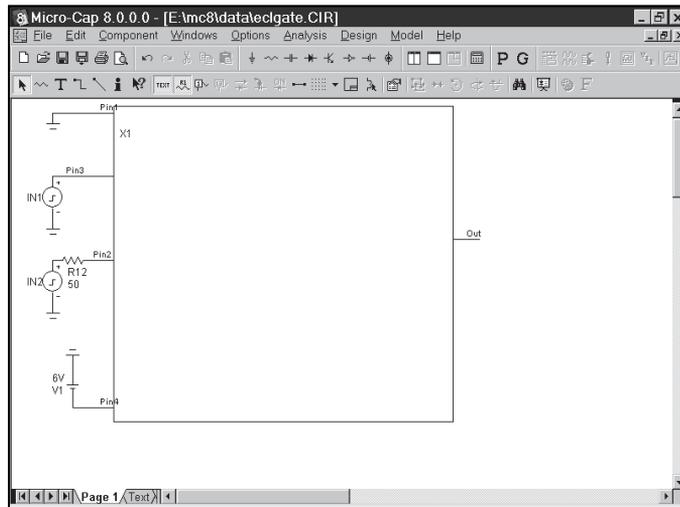


Figure 11-7 The ECLGATE circuit using a macro

The program has converted the circuitry in the box region into a macro circuit. To do this it did the following:

1. Placed the circuitry within the box region in a file called ECLGATE.MAC.
2. Placed an ECLGATE macro into the current component library.
3. Modified the ECLGATE circuit on the screen by replacing the contents of the region box with a macro call ECLGATE.

As a result of this action the ECLGATE circuit appears simpler. Its complexity is hidden and encapsulated in the macro block called ECLGATE. In addition the macro has been entered into the component library and is available for use by other circuits.

You can see the macro circuit by clicking on the Info  button and then clicking on the ECLGATE macro block. It looks like this:

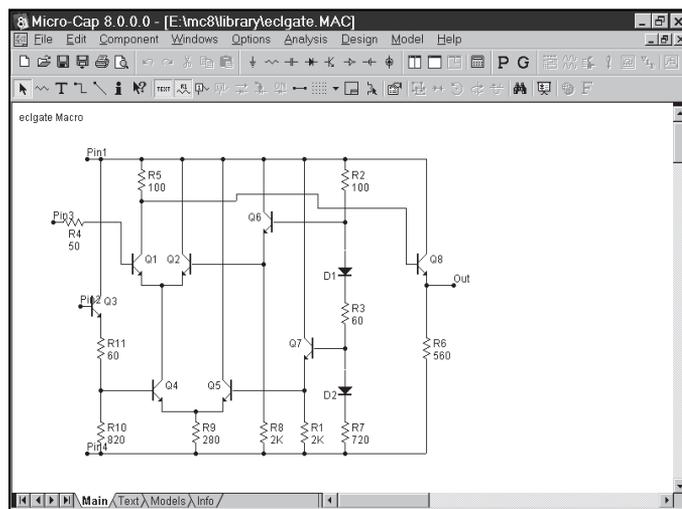


Figure 11-8 The ECLGATE macro circuit

The program has determined that five pins are needed and has located the pin text as required. This form of macro creation always uses an adjustable macro box. Its sides and pins can be moved about as desired. It always retains its rectangular shape, however.

To change the location of the pins, select the macro by clicking on it. Drag the dot at the end of the pin to the new location and release it.

To change the shape of the macro box, select the macro by clicking on it. Drag a corner dot to change both length and width. Drag a side dot to change length or width. For example, the figure below shows an example of how the shape can be modified. Compare this with Figure 11-7.

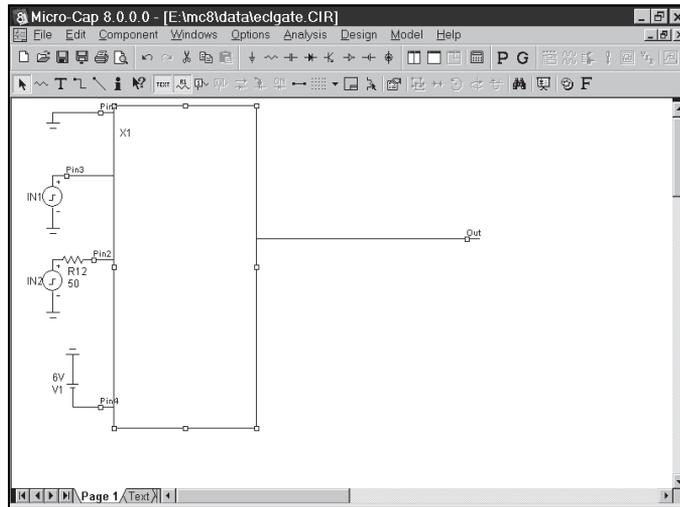


Figure 11-9 The altered macro box shape

What's in this chapter

This chapter describes the use of SPICE subcircuits. The four steps of subckt creation and usage are described. They are:

- Creating the subckt circuit file
- Selecting or creating a suitable shape for the subckt
- Entering the subckt into the Component library
- Using the subckt in a circuit
- Using the Add Part wizard to add subckt parts

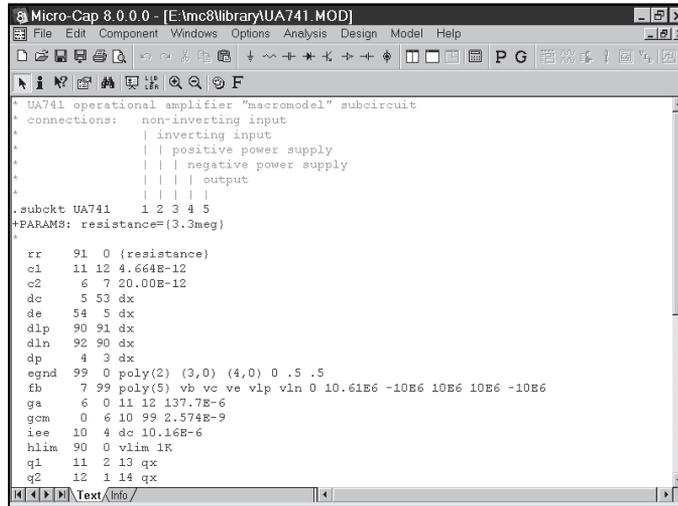
What is a subcircuit?

Subcircuits are complete SPICE text file circuits, created and saved on disk to be used by other circuits. The great virtue of subcircuits is the same as for macros; the behavior of a complex circuit block can be incorporated into and represented by a single component. The complexity and detail is hidden from view in the circuit that uses the macro.

Another major advantage is that most parts vendors use the SPICE subcircuit as the principal means for modeling their parts. Access to these models is the key reason for using subcircuits.

Creating the subcircuit text file

The first step is to create the subcircuit circuit itself. We won't do that here. Instead we'll use an existing subcircuit, supplied with MC8. To load the SPICE file, select the **Open** item from the **File** menu. Type "..\LIBRARY\UA741.MOD". The file looks like this:



```
Micro-Cap 8.0.0.0 - [E:\mc8\library\UA741.MOD]
File Edit Component Windows Options Analysis Design Model Help
+ UA741 operational amplifier "macromodel" subcircuit
+ connections: non-inverting input
+               | inverting input
+               | | positive power supply
+               | | | negative power supply
+               | | | | output
+               | | | | |
+
+ .subckt UA741 1 2 3 4 5
+PARAMS: resistance=(3.3meg)
+
rr 91 0 {resistance}
c1 11 12 4.664E-12
c2 6 7 20.00E-12
dc 5 53 dx
de 54 5 dx
dip 90 91 dx
dln 92 90 dx
dp 4 3 dx
egnd 99 0 poly(2) (3,0) (4,0) 0 .5 .5
fb 7 99 poly(5) vb vc ve vlp vln 0 10.61E6 -10E6 10E6 10E6 -10E6
ga 6 0 11 12 137.7E-6
gcm 0 6 10 99 2.574E-9
iee 10 4 dc 10.16E-6
hlim 90 0 vlim 1K
q1 11 2 13 qx
q2 12 1 14 qx
```

Figure 12-1 The UA741.MOD subcircuit

This subcircuit is a standard Boyle-type model for the classic UA741. The circuit file can be created with the MC8 SPICE text editor or by using an external word processor. Frequently, it can be obtained as a text file from a manufacturer of the part. It must be placed in the library directory and the file name entered into the "NOM.LIB" file to be accessible to MC8. The file name is entered by adding a line such as,

```
.LIB "UA741.MOD"
```

to the NOM.LIB file. The NOM.LIB file is located in the library folder.

Putting subcircuits into the Component library

The third step is to enter the subcircuit in the Component library. To do this, select **Component Editor** from the **Windows** menu. Double-click on the Subckts group of the Analog Primitives group. This presents a list of the existing subcircuits. Click on the Opamp_subckt_5. This shows the existing Opamp_subckt_5 component. If you were adding this component as a new subcircuit you would have clicked first on the Subckt group to select it, then clicked on the Add Part button to add the Opamp_subckt_5 component to the macro group and edited the data fields to look like these:

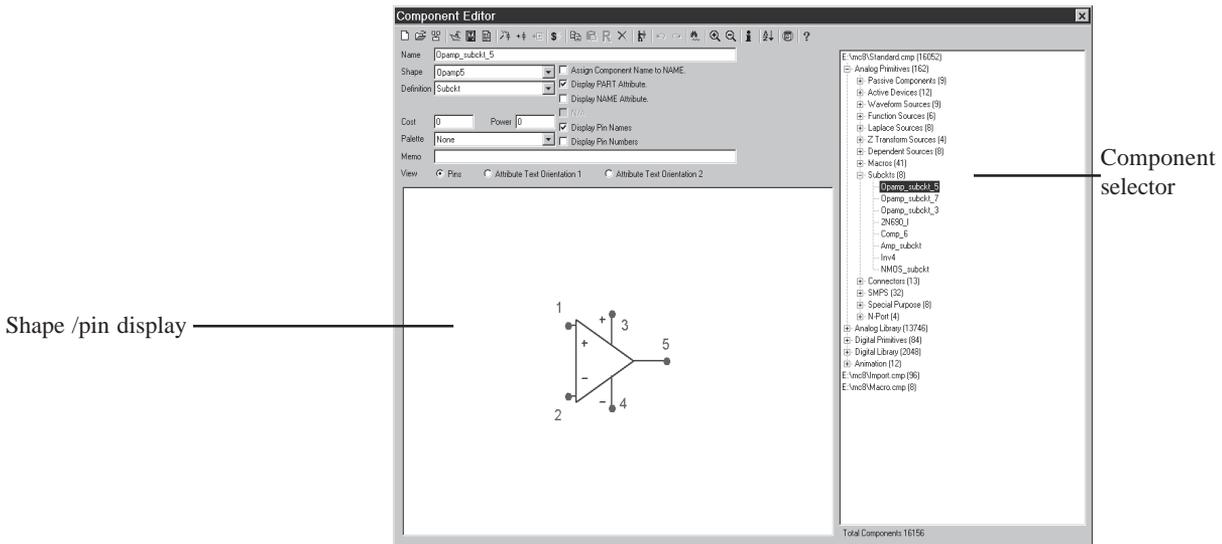


Figure 12-3 The subckt Component library entry

The first data field is the Component name. This can be any alphanumeric name. It is usually the name of the subckt, though it does not need to be . In this case we used the general name, Opamp_subckt_5, but we could also have chosen the name UA741.

The Shape entry is the name of the shape to be used to represent the macro. Here we have selected the Opamp5 shape. Click on the down arrow to see a list of the available shapes from the Shape library.

The Definition field is where you specify the electrical definition. In this case, we specify Subckt. Click on the down arrow to see a list of the available definitions.

The Attribute text orientation options show the location of attribute text relative to the shape. Each shape has potentially eight combinations of rotation and reflection, called orientations. Orientation 1 is for the four horizontal orientations. Orientation 2 is for the four vertical orientations. Subckt attribute text is comprised of the PART attribute which provides the part name, the NAME attribute which specifies the name of the subcircuit, the optional FILE attribute which specifies the name of the file containing the .SUBCKT statement, the PARAMS attribute which provides any passed numeric parameters, and the TEXT attribute which provides any passed text parameters. These attributes are specified when the component is placed in a schematic. The Attribute placement fields let you drag the text with the left mouse button to define where the first (XX) and second (YY) attribute text will be initially placed. After the component is placed, attribute text can be relocated by dragging with the mouse.

An important aspect of entering a subckt into the library is defining where the pins are. For a component like a BJT, the pin names are predefined as Emitter, Base, and Collector. For a subckt, the pin names and locations must be defined by the user. This is done by clicking in the Shape / pin display. This activates a Pin dialog box, allowing you to enter a new pin name and to specify if it is digital or analog. Double-clicking on an existing pin lets you edit it. Both the pin marker dot and the pin name can be independently dragged with the mouse to the desired locations.

The pin names used in the Component library must match those in the .SUBCKT statement in the file that contains the subckt itself. For example, the UA741 uses "1" as the name of the positive input pin, so we need a pin named "1" at the position on the shape we want to identify as the positive input pin. Similarly, "5" is used as the name of the output pin so we want to have a pin named "5" placed on the shape where we expect to find the opamp output.

UA741 Subckt Pin Names	Description
1	Positive input pin
2	Negative input pin
3	Positive power pin
4	Negative power pin
5	Output

Notice that you could put the pins anywhere. There is no magic in placing the positive input pin near the + symbol. It is simply easier to remember where the pins are when you wire them into a circuit.

Close the Component editor.

Using the subcircuit as a component

The final step is using the subckt. Once it has been entered into the Component library, it is available for use in any circuit and can be accessed from the Component menu group where it was added. Unload the UA741.MOD SPICE file and load the SUBCKT1 circuit. It looks like this:

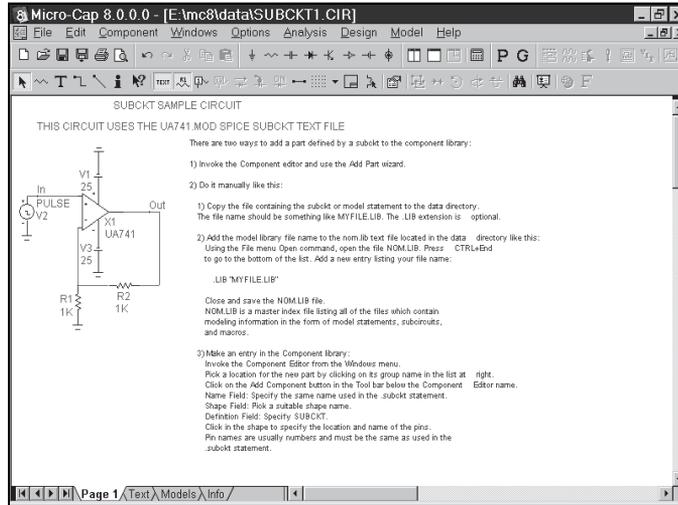


Figure 12-4 Using the subckt component

The key points about using a subckt are:

1. When the part is placed in a circuit, its NAME attribute must be the same as its .SUBCKT name.
2. MC8 searches for the subckt in the following places:
 - The text area of the schematic. You can paste the subckt to the text area and avoid the need to copy the subckt file to the library directory.
 - The file name listed in the File attribute.
 - The files in any .LIB statements, including the default .LIB NOM.LIB. *This is the preferred and most common access method.*
3. Many different subcircuits can be covered with a single, generic subckt entry in the Component library, if they have the same pins and node order.

Using the Add Part wizard to add subckt parts

The procedure described in the previous pages describes how to add a subckt part from scratch. Frequently, you may wish to add a part model supplied by a semiconductor parts vendor. The easiest way is to use the Add Part wizard. It combines the three basic steps into one operation.

- 1) Copy the file containing the subckt or model statement to the library folder.
- 2) Add the model file name to the nom.lib text file located in the library folder.
- 3) Make an entry in the Component library.

Suppose you have downloaded a file called AD.LIB from a vendor to the location C:\. In the file is a subckt statement for the power NMOS device IRF5101A. You want to add it to the library. Here is what you do.

- 1) Select **Component Editor** from the **Windows** menu.
- 2) Select the group where you want the part name to appear in the Component menu. For this illustration simply select Analog Library
- 3) Click on the Add Part wizard  button.
- 4) The first prompt is for the part name. Type in "IRF5101A". Click Next.
- 5) The next prompt is for the electrical definition. Select Subckt. Click Next.
- 6) The next prompt asks for the name of the file containing the subckt. Type C:\AD.LIB (or wherever the file currently resides). Click Next.
- 7) The next prompt is the tricky one. The wizard has scanned the Component library and compiled a list of all subckt based parts that have pin names matching those it found for the IRF5101A subckt. Actually it tries to present only a representative list, as there may be thousands of possible parts. Select one from the list. In this case we will select the DN MOS as we know this to be a discrete NMOS device. Click Next.
- 8) The next prompt asks for an optional memo field. Click Next.
- 9) The next prompt asks for an optional palette assignment. Click Next.

10) The next panel lets you set the visibility of the various part attributes. Click Next.

11) The next panel lets you make the assignment of the component name to the NAME attribute. This assignment avoids the need to invoke the Attribute dialog box when one of these parts are added to a schematic. Click to enable the option. Click Next.

12) The final panel advises you to examine the entry to be sure all elements have been selected properly, particularly the choice of shape and where the pin names have been placed on the shape.

For our example, the new part should look like this.

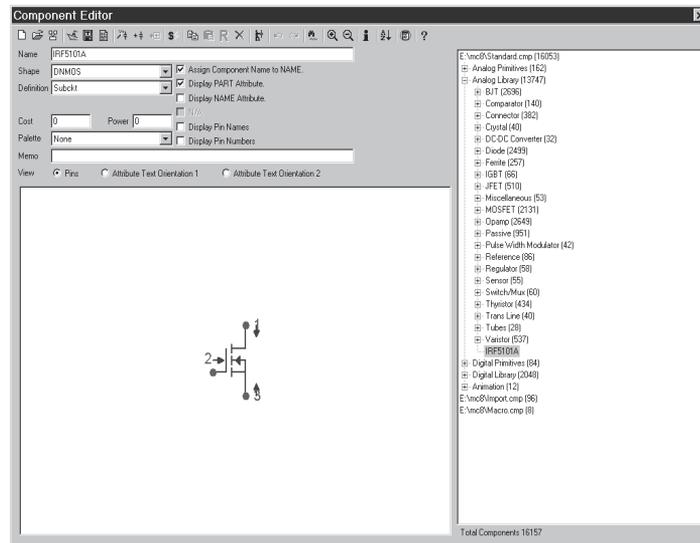


Figure 12-5 The subckt Component library entry

It is also possible to import many parts simultaneously from one file. That is done with the Import wizard. Its operation is described in the Reference manual.

What's in this chapter

This chapter describes how to get copies of your work on printers. MC8 provides output in two basic ways.

- Graphics
 - Schematics
 - Analysis plots
 - Performance plots
 - 3D plots
 - Monte Carlo histograms

- Text
 - Schematic text area
 - SPICE circuit descriptions
 - Document files
 - AC, DC, and transient analysis numeric output
 - Transient analysis state variables
 - Monte Carlo statistics

All text output is available in disk file or hard copy form. Graphics output is available on any printer that your system supports.

Printing schematics

Schematics may contain one or more pages which may be printed. Depending upon the scale chosen, each page is partitioned into one or more sheets for printing or plotting. The Auto option scales the largest page to fit on just one sheet. The User option lets the user pick a scale and thus adjust the physical size of the schematic. Larger image sizes produce more sheets of paper per page.

To setup or preview the printed output, use these **File** menu options:

- **Print Setup:** The contents of this dialog box depend upon the type of printer chosen, but it usually lets you choose the printer, orientation, paper size, and other options. The choice of orientation affects the partitioning.
- **Print Preview:** This option lets you preview how the schematic will be partitioned for printing into one or more sheets.
 - **OK:** This accepts any changes and exits the dialog box.
 - **Cancel:** This ignores any changes and exits the dialog box.
 - **Schematic Page**
 - Next: This displays the next page of the schematic. Note that the scale, whether Auto or User is chosen, applies to all pages.
 - Prior: This displays the prior page of the schematic.
 - **Page List:** This list box shows the name of the page being displayed. You can display any page by selecting its name from the list. Select pages to be printed by adding a check mark adjacent to the page name.
 - **Text Sheets:**
 - Next: If a text page is selected from the Page List, this shows the next sheet of the text page as it will appear when printed.
 - Prior: If a text page is selected from the Page List, this shows the prior sheet of the text page as it will appear when printed.
- **Print:** This invokes the Print dialog box.
- **Setup:** This accesses the Print Setup dialog box.

- **Properties:** This accesses the standard schematic Properties dialog box, which lets you make last minute changes to colors, fonts, and the title block.
- **Help:** This accesses the Help System.
- **Auto *Auto_Scale*:** This option adjusts the scale so that the largest page in the schematic will just fit on the chosen paper size. All other pages use this scale. If Auto is clicked, *Auto_Scale* shows the scale value used. The same result is achieved by clicking on the User button and typing the same number in the *User_Scale* data field.
- **User *User_Scale*:** This button enables the *User_Scale* value. A scale of 100 is normal size. The *User_Scale* data field holds the scale. A change in scale takes effect only when the Apply button is pressed.
- **Apply:** This updates the display to reflect edits to *User_Scale* field.

The Title Block item in the Properties dialog box lets you define the content of the five fields of the title block. Field 1 is printed in large text, and the remainder are printed in smaller text. There are several options to facilitate printing the file name, date, and page numbers. These options are enabled by placing the following key words in the desired text field.

- | | |
|---------------------|---|
| • \$NAME | This prints the circuit name. |
| • \$NAMEEXT | This prints the circuit name with an extension. |
| • \$DATE | This prints the current date. |
| • \$TIME | This prints the current time. |
| • \$USER | This prints the user name. |
| • \$COMPANY | This prints the company name. |
| • \$PAGE | This prints the page number. |
| • \$PAGENAME | This prints the schematic page name. |
| • \$MAXPAGE | This prints the largest page number. |
| • \$MC | This prints the Micro-Cap version number. |
| • \$SHEET | This prints the sheet number. |
| • \$MAXSHEET | This prints the largest sheet number. |

The **One Title** option box produces one title block per schematic if checked. If unchecked one title block per sheet is printed

An example of schematic printing

Load the file 'PRINT'. Select **Print Preview** from the **File** menu.

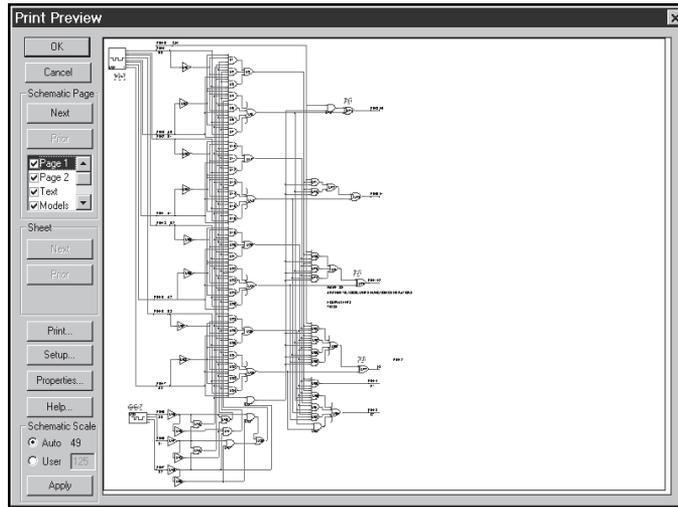


Figure 13-1 Print Preview of page one of the PRINT circuit

The Scale mode is set to Auto so MC8 prints one sheet for each of the two pages contained in this schematic. Figure 13-1 shows how page one will fit in landscape mode. Click on the Schematic / Next button to display page two. It looks like this:

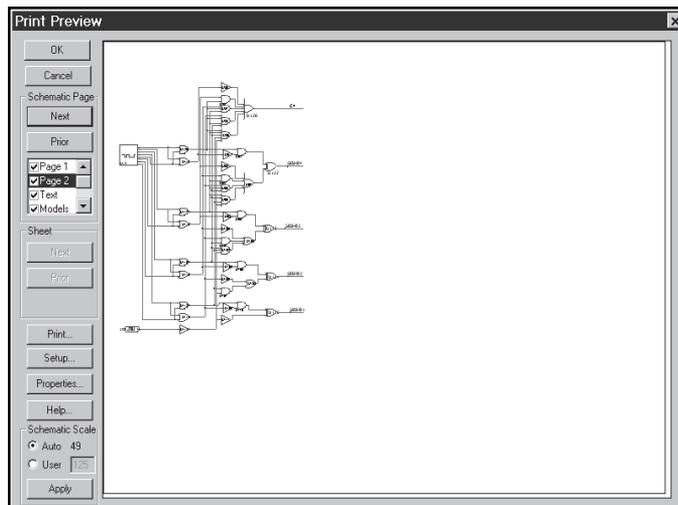


Figure 13-2 Print Preview of page two of the PRINT circuit

Now click on the User button. This applies the current user scale of 125% and partitions the schematic for printing. The result, shown in Figure 13-3, shows how page two will be partitioned. At this scale, two sheets of paper are required to print page two of the schematic.

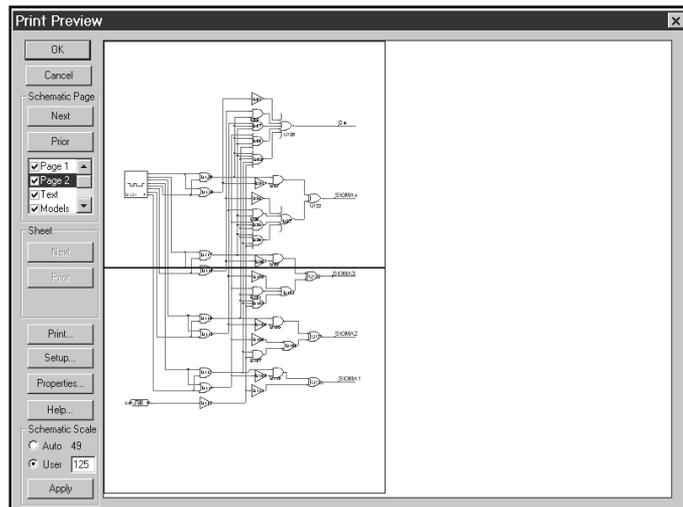


Figure 13-3 The Print Preview of page two with a scale of 125

Note that clicking on the Auto or User buttons automatically update the screen. Simply editing the User scale field will not, by itself, update the screen. Clicking the mouse on the Auto, User, or Apply buttons will cause a screen update.

Once a suitable scale is chosen and the desired page ranges are selected, click on the Print button to invoke the Print dialog box to initiate printing. Of course, you need not use the Print Preview each time you want to print. If you've previously checked the Print Preview and are happy with it, simply press CTRL + P to invoke the Print dialog box.

Click on the OK button and close the file.

Printing an analysis plot

To produce a plot, first run the analysis. When the analysis is complete select the **Print Setup** option from the **File** menu. From this dialog box, select the orientation and other options.

To illustrate, load the file ECLGATE and run transient analysis. Select **Print Preview** from the **File** menu. It should look like this:

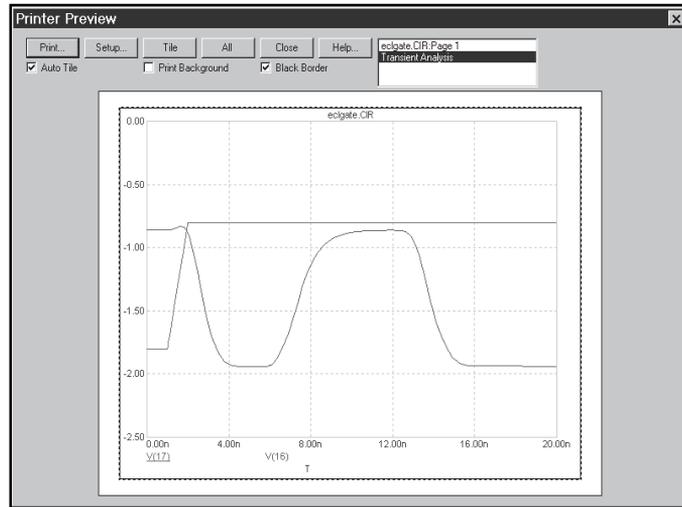


Figure 13-4 The Print Preview of an analysis plot

The Print Preview dialog box lets you include, and set the size and location on the paper of the analysis plots, 3D plots, and Monte Carlo plots. You can even include the circuit schematic. The dialog box provides these buttons:

Print: This sends the image to the printer.

Setup: This invokes the Print Setup dialog box.

Tile: This arranges the selected plots in contiguous non-overlapping areas of the paper.

All: This selects all available plots for printing.

Close: This closes the Print Preview dialog box.

Help: This accesses the Help system.

Plot List: This is a list of available plots. Click on a plot to toggle its selection state.

Auto Tile: This automatically tiles selected plots in the layout. If Auto Tile is off, each plot acquires a set of handles and may be moved and re-sized.

Print Background: This prints the selected background color. It's generally a bad idea on ink jet printers because the large quantity of ink needed for the background saturates and warps the paper surface.

Black Border: This adds a black border around each plot.

To illustrate these features, click on the All button. This adds the schematic to the analysis plot producing a display like this.

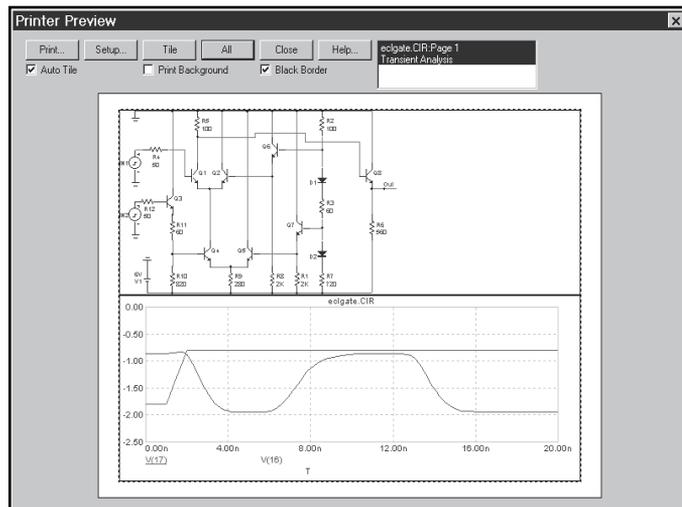


Figure 13-5 Adding the schematic to the analysis plot

Because Auto Tile is enabled the plots are tiled to fill the paper. Click on the Auto Tile button to disable it. The plots are redrawn with little black rectangles that serve as handles. You can now move individual plots by dragging on the interior part of the plot and you can re-size individual plots by dragging on one of the handles.

To illustrate modifying the layout, grab the lower-right handle of the schematic and drag it up and to the left. Release the handle. Drag the upper-right handle of

the analysis plot down and to the left. Release the handle. Drag the analysis plot from the interior so that it moves. Drag it up and drop it on the schematic so as to overlap the two. The display should now look like this:

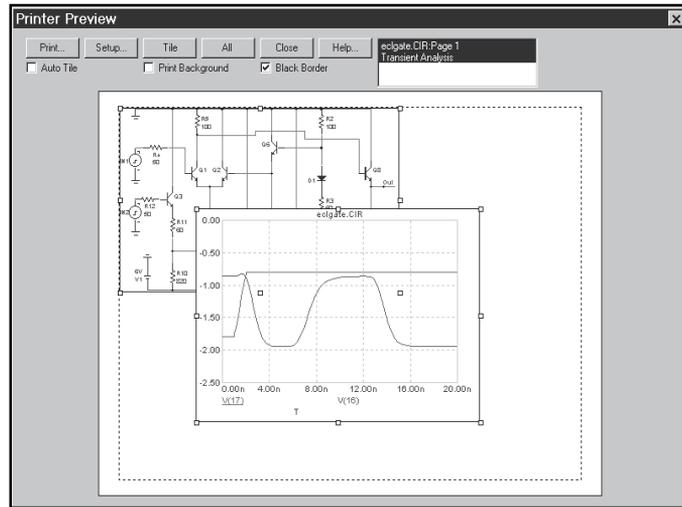


Figure 13-6 Modifying the preview layout

In addition to printing in the usual way, you can create picture files of the various schematics and plots created by MC8. These files can be imported into other programs such as Word, PageMaker, and Excel. They can also be placed in MC8 schematics.

To illustrate, exit the print routines by clicking on the Close button. From the **Edit** menu select **Copy the Entire Window to Picture File**. When the dialog box comes up, select the JPEG format from the Save as Type list box, then click on the Save button. This saves the analysis plot in JPEG format using the file name ECLGATE.JPG.

Quit transient analysis with F3. Select **Options / Mode / Picture**, or click on the Graphics button  and select Picture from its menu. This enters the picture file placement mode. Click in the lower right of the schematic. In the dialog box, click on the Browse button and select the ECLGATE.JPG file.

The screen should now look like Figure 13-7.

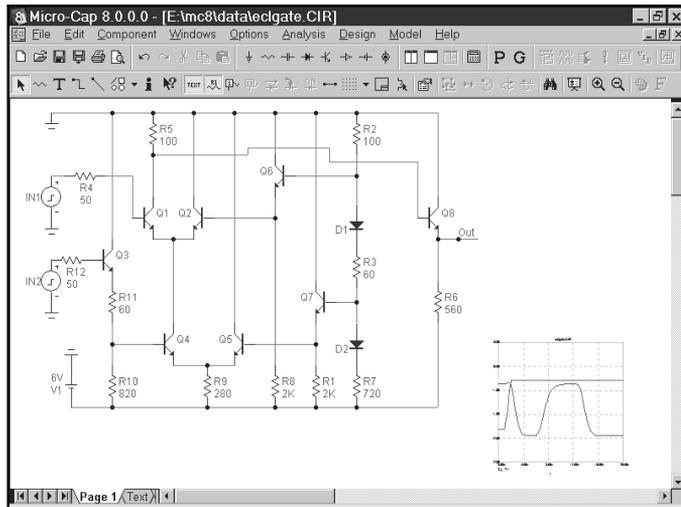


Figure 13-7 Adding a WMF of the analysis plot

This same picture file can be imported into Excel. From Excel, select **Insert / Picture / From File**. Enter the path and file name, or browse until you find it. The path and name will be something like:

C:\MC8\DATA\ECLGATE.JPG.

The picture should appear in Excel with handles for changing its size and shape.

What's in this chapter

Animation mode is an alternative way of visualizing analysis results. It performs the selected analysis and plots any specified waveform. While it is plotting, it is also updating any animation devices on the schematic along with any on-schematic display values. In this chapter, we demonstrate Animation mode's capability with an example. The topics include:

- How Animation mode works
- Animation components
- Animate Options dialog box
- A Transient example

How Animation mode works

Animation mode is another way to view simulation results. In this mode, a single analysis data point is taken and the on-schematic display values and animation components are updated on the schematic. Micro-Cap then waits for either a key press or a specified time delay before calculating the next data point. The purpose is to slow the simulation down to be able to monitor the updated changes on the schematic. The on-schematic display values let the node voltages, branch currents, power dissipation, and device conditions of the latest data point be viewed on the schematic. The animation components include analog bar, analog LED, DC motor, DPST, SPDT, SPST switches, analog/digital voltage/current meter, relay, three-color traffic light, (digital) LED, and seven segment display.

In transient analysis, a single time step is calculated, and any on-schematic display values or animation components are updated on the schematic. When a key is pressed or after the specified delay time has elapsed, the solution at a new time point is calculated and the schematic updated accordingly. The schematic always displays the values from the last time step calculated.

In AC analysis, the schematic displays the on-schematic display values and states of the animation components from the operating point calculation. These values are fixed during the entire simulation. Even though the schematic displays are fixed at the operating point, the animation mode will calculate a single frequency data point when a key has been pressed or the specified time delay has elapsed.

In DC analysis, a single DC sweep data point is calculated, and any on-schematic display values or animation components are updated on the schematic. When a key is pressed or after the specified delay time has elapsed, the next DC sweep data point is calculated and the schematic updated accordingly. The schematic always displays the values from the last DC sweep data point calculated.

Animation components

Animation components are designed to move, rotate, and change color, size or shape in response to user clicks and the electrical signals driving them. In most cases they simulate what a physical device might do.

Animated analog bar: This device produces a colored bar whose height is proportional to its input voltage.

Animated analog LED: This is a light emitting diode whose color appears when the voltage across its two terminals equals or exceeds the specified on voltage.

Animated DC motor: This device is a DC motor that rotates at an angular velocity which is controlled by the instantaneous input voltage. The rotation rate is user specified in revolutions per second per volt.

Animated DPST, SPDT, and SPST switches: These animated switches open or close with a double click.

Animated meter: This device is a voltage or current meter whose display can be either analog or digital.

Animated relay: This device is a relay that responds dynamically to changes in input current.

Animated traffic light : This device is designed to simulate a typical traffic light. There are three lights, red, yellow, and green. Each of the three lights is lit when its input control pin voltage exceeds the specified on voltage.

Animated meter: This device is a voltage or current meter whose display can be either analog or digital.

Animated digital switch: The switch is designed to produce either a digital zero or one state. It has a single output pin at which the selected digital state will appear. During a simulation, the switch may be clicked on to toggle it between its zero and one output. The appearance of the switch's arm will change to indicate which state the switch is currently connected to.

Animated digital LED: The LED component is designed to represent the display of a light emitting diode. It has a single input pin. Depending on the digital state or the analog voltage at the input pin, the LED will be lit with a different

color on the schematic. The colors the LED uses are defined in the Color/Font page of the circuit's Properties dialog box. In this page, there is a list of digital states that have a corresponding color.

Animated Seven Segment Display: The seven segment component is designed to represent the display of a seven segment display. It has an input pin for each segment of the display. The active state of the pins can be set to low or high through the ON STATE attribute. It is intended to work with basic seven segment decoders/drivers although any input will control the display of the corresponding segment.

Animated digital parts

The switch, LED, and seven segment display, are all digital in nature. The LED and seven segment display components do not model the electrical characteristics of the parts they represent. They are only available for display purposes. The digital switch is the only one of the animated digital parts that can affect a simulation. Each of these components must have an I/O model specified for them in order to work with analog components.

Animate Options dialog box

This dialog box sets the interval between data point calculations in an animation analysis. The dialog box is invoked by selecting **Animate Options** under the **Scope** menu and appears in Figure 14-1. It provides these options:

Wait

Don't Wait: This option turns off the animation which lets the analysis run at optimum speed.

Wait for Key Press: This option will produce one data point with each key press of CTRL + SPACEBAR.

Wait for Time Delay: This option will produce one data point per specified time delay.

Time Delay: This text field sets the specified time delay in seconds that will be between each data point calculation. This text field is only active if the Wait for Time Delay option is selected.

OK: This closes the dialog box and saves any changes that were made.

Cancel: This closes the dialog box and ignores any changes that were made.

Help: This accesses the help topic for this dialog box.

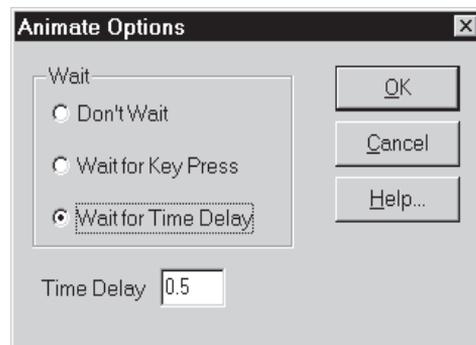


Figure 14-1 Animate Options dialog box

A transient analysis example

To see how the animation mode works in transient analysis, click on **Open** under the **File** menu and load the circuit ANIM3. The ANIM3 circuit uses the seven segment components to display the outputs of three 7448 seven segment decoders. The circuit should appear like this:

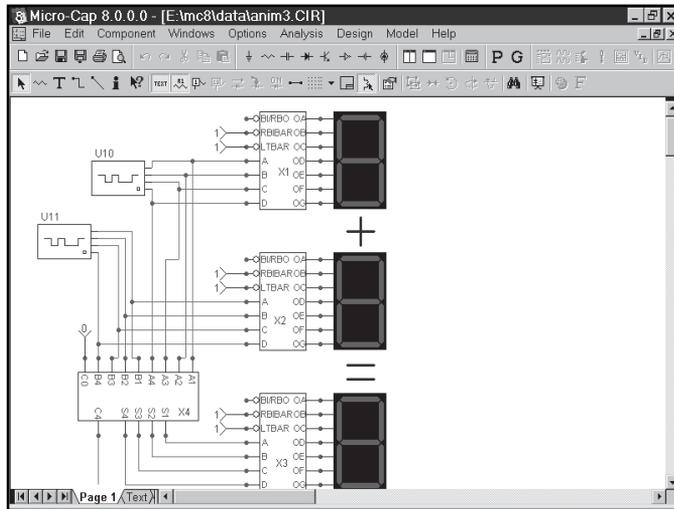


Figure 14-2 The ANIM3 circuit

Select **Transient** from the **Analysis** menu. Close the Analysis Limits dialog box. Click on the **Scope** menu and select **Animate Options**. The Wait setting for the ANIM3 circuit is currently set to Wait for Time Delay and the Time Delay has been defined as 0.2. This means that a single data point will be calculated every 0.2 seconds. Click OK. Hit the F2 key to run the simulation. The resulting values will be displayed in the plot and on the schematic. Notice that both the digital states on the schematic and the seven segment displays are updated with each data point. Figures 14-3 and 14-4 display the simulation results at two separate times in the circuit.

For animate mode, a split screen should be used to be able to view both the waveform plot and the schematic at the same time. This can be done by selecting **Tile Vertical** or **Tile Horizontal** under the **Windows** menu. The schematic can be moved within the window by dragging with the right mouse button.

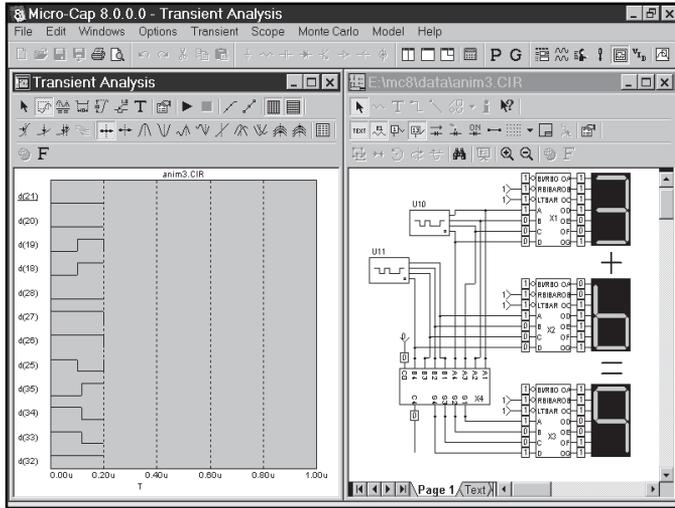


Figure 14-3 Animation mode results at 200ns

The waveforms in the plot also show the output values of the Ina stimulus (nodes 21-18), the Inb stimulus (nodes 28-25), and the resulting output (nodes 35-32) of the adder. As can be seen in the figures, it is much easier to view the basic output through the animation components than through the actual waveforms.

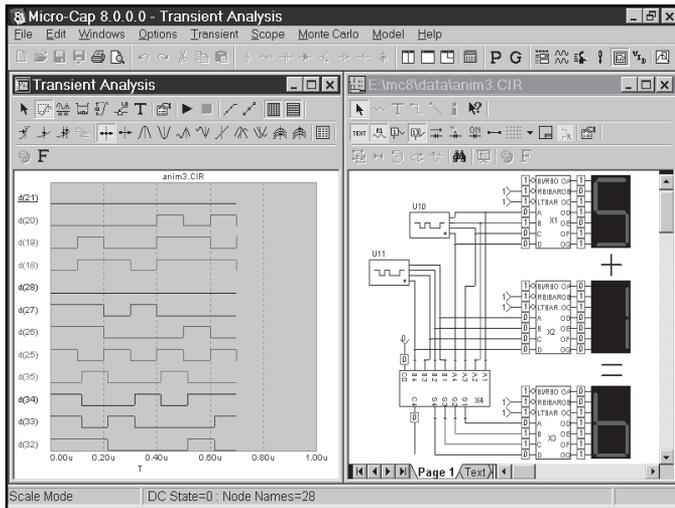


Figure 14-4 Animation mode results at 700ns

A Dynamic DC example

To see how the animation mode works in Dynamic DC analysis, click on **Open** under the **File** menu and load the circuit ANIM5. The ANIM5 circuit shows how to use the voltage and current meters, analog Bar, DPST switch, and motor. Run Dynamic DC. The circuit should appear like this:

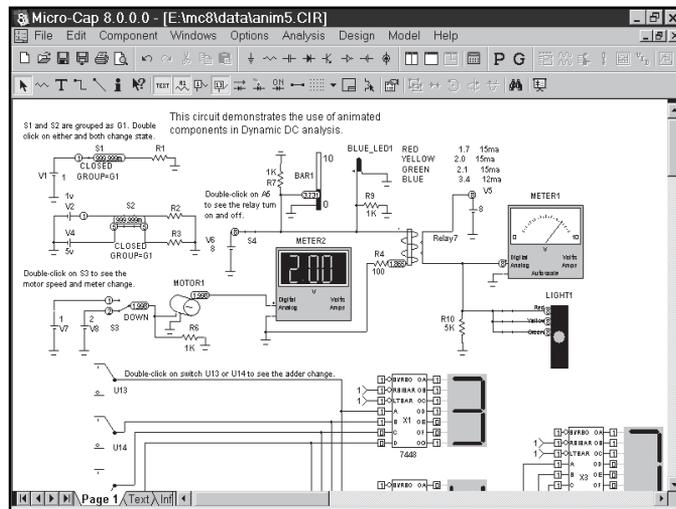


Figure 14-5 The ANIM5 circuit

In Dynamic DC MC8 does a DC operating point calculation every time a change is made to the circuit. It continually updates the display. In this circuit we have a 20 volt and a 2 volt battery connected through a DPST switch to an analog and a digital voltmeter. There is also an analog bar, a motor and a digital ammeter.

The motor rotates at $1.998\text{volts} * 2\text{RPS/volt} = 1.998\text{ RPS}$ (revolutions per second). The analog bar is at 2/10 of its full height.

Double click on the DPST switch. The switch opens and the meters now read picovolts and picoamps. Because the DC voltage across the motor is now zero, it ceases to rotate.

Symbols

\$COMPANY 191
 \$DATE 191
 \$MAXPAGE 191
 \$MAXSHEET 191
 \$NAME 191
 \$PAGE 191
 \$PAGENAME 191
 \$TIME 191
 \$USER 191
 .SUBCKT 185

A

AC analysis 83
 Accessing 85
 Excitation 84
 System variables 84
 Adding components 44, 64
 Adding tags to plots 124
 Adding text to plots 123
 Align cursor 112
 Analog primitives 48
 Analog waveforms 26, 34
 Analysis Limits dialog
 box 26, 66, 86, 101, 102
 Command buttons
 Add 66, 86, 102
 Delete 66, 86, 102
 Expand 67, 86, 102
 Help 67, 86, 102
 Properties 67
 Run 66, 86, 102
 Stepping 67, 86, 102
 Numeric limits field
 Frequency range 87
 Maximum change % 87, 88
 Maximum time step 67
 Noise input 88
 Noise output 88
 Number of points 67, 87, 104
 Temperature 67, 87, 103
 Time range 67
 Options
 Auto scale ranges 70, 90, 106

Frequency step 86
 Normal run 69, 89, 106
 Operating point 70, 90
 Operating point only 70
 Retrieve run 69, 90, 106
 Save run 69, 90, 106
 State variables options
 Leave 70, 90
 Read 69, 90
 Zero 69, 90
 Animated analog bar 201
 Animated analog LED 201
 Animated DC motor 201
 Animated digital switch 201
 Animated DPST switch 201
 Animated SPDT switch 201
 Animated meter 201
 Animated relay 201
 Animated Seven Segment Display 202
 Animated SPST switch 201
 Animated traffic light 201
 Animation 12, 19, 130, 199
 Components 201
 Example 204, 206
 Mode 199
 Options 203
 Attributes 37, 44
 Dialog box 44, 52, 53
 Text 38
 Auto scale option 69, 89, 105

B

B field plots 133
 Batch mode 35
 Behavioral modeling 19
 Bill of materials 10
 Bottom cursor position mode 118
 BSIM4 11

C

Capacitance plots 133
 Characteristic curves 10
 Characteristic plot command 45
 Charge plots 133

- Cleanup command 13
- Clear command 55
- Click the mouse 37
- Clipboard 37
- Color menu 68, 88, 105
- Command line 35
- Command line operation 35
- Component 37
- Component Editor
 - Creating pins 39
 - User palette membership 46
- Component library 19
- Copy command 37, 56, 64
- Current 25
- Cursor 37
- Cursor mode 31, 113, 116, 117
- Cursor value fields 116
- Cut command 55, 64

D

- Data directory 22
- Decimal numeric format 15
- Default component 64
- Define command 24
- Deleting objects 55
- Deselecting objects 57
- DEV tolerances 158
- Digital
 - Hex operator 25
 - State 25
- Digital waveforms 26, 34
- Distortion analysis 13
- Drag 37
- Drag copying 58, 64
- Dynamic AC analysis 13
- Dynamic DC analysis 13, 30

E

- Editing parameters and text 52
- EKV 11
- Equipment required 21
- Esc key 38
- Expressions 26, 31, 71

F

- FFT window 13
- Filter design 19
- Find in Files command 12
- Function keys 40
 - CTRL + F11 Optimizer 40
 - CTRL + F4 Close window 40
 - CTRL + F6 Change window 40
 - CTRL + F9 Delete all plots 40
- F1 Help 40
- F10 Properties dialog box 40
- F11 Stepping dialog box 40
- F12 State Variables editor 40
- F2 Run analysis 40
- F3 Exit analysis 40
- F4 Display analysis plot 40
- F5 Display numeric output 40
- F6 Auto scale 40
- F7 Scale mode 40
- F8 Cursor mode 40
- F9 Analysis Limits dialog box 40

G

- Gaussian distribution 160
- Global settings 147
- Go to performance command 111
- Go to X command 111
- Go to Y command 111
- Grid 38
- Grid text 38

H

- H field plots 133
- Hardware requirements 21
- Horizontal cursor 110
- Horizontal tag mode 113
- HSPICE style binning 11

I

- IBIS 11, 19
- Inductance plots 133
- Info Page 12
- Initialization

- How it works 78
- Run 78
- Setup 78
- INOISE 94
- Installation 22

L

- Linear DC stepping 103
- Linear plot option 68
- List DC stepping 103
- Live formulas 10
- Local Path commands 12
- Log DC stepping 103
- Log plot option 68
- LOT 158
- LOT tolerances 158

M

- Macro
 - Definition 38, 171
 - Parameters command 172
- Maximum time step 67
- Monte Carlo
 - Distributions
 - Gaussian 160
 - Uniform 160
 - Worst case 160
 - Example 163
 - How it works 158
 - Options
 - Distribution to Use 162
 - Number of Runs 162
 - Report When 162
 - Seed 162
 - Status 162
 - Standard deviation 160
- Multi-stage undo and redo 10

N

- N-Port device 11
- Navigating 59, 64, 138
 - Paging 59
 - Panning 59
 - Relocation 59

- New command 64
- Node
 - Definition 38
 - Names 38, 50, 64
 - Numbers 39, 50
- Node number assignment 50
- Node snap 51
- Noise
 - Analysis 94
 - Flicker 94
 - Shot 94
 - Thermal 94
- NOM.LIB 182, 186
- Normalize command 110, 111
- Number of points 67, 87, 104
- Numeric output
 - AC 93
 - DC 104
 - Transient 67
- Nyquist plots 96

O

- Object 39
- ONOISE 94
- Optimizer 32

P

- P column 26
- Panning 39, 61, 115, 138
- Password 12
- Paste command 39, 55, 56, 64
- PCB interface 19
- Performance functions 19, 126, 161
- Picture files 52
- Pin connections 49
- Pin definition 39
- Plot group number 68
- Plotting waveforms 25, 71
- Point tags 113
- Polar plots 88, 98
- Portable schematic files 10
- Power 25
- Print
 - An example 192

- Print preview
 - Apply command 191
 - Auto command 191
 - Help command 191
 - User command 191
- Title block 191
- Printing 189
 - Print preview 190
 - Schematics 190
 - Setup 190
- PRIVATEANALOG 147, 154, 159
- PRIVATEDIGITAL 147, 154, 159
- Probe 131
 - AC analysis 139
 - DC analysis 141
 - Linear plot 135
 - Log plot 135
 - Many traces command 133
 - Navigating the schematic 138
 - New run command 132
 - One trace command 132
 - Probing lead-to-lead voltages 137
 - Probing node voltages 136
 - Remove all command 132
 - Save all command 133
 - Save V & I only command 133
 - Separate analog and digital command 132
 - Transient analysis variables 134, 144
 - B field 135
 - Capacitance 134
 - Charge 134
 - Flux 134
 - H field 135
 - Inductance 135
 - Resistance 134, 144
 - Time 135
- Probing 19
- Properties dialog box 26
- PSPice 19

Q

- Quick tour 24

R

- Redo command 55
- Reference manual 20
- Refresh command 54
- Registering 21
- Revert command 41, 57
- RF models 10
- Rotating components 44

S

- Same Y scales command 112
- Sanity checker 10
- Scale factors 14
- Scale mode 113
- Scales 73
- Scope 107
 - Commands
 - Align cursor 112
 - Animate options 110
 - Auto scale 109
 - Cursor 110
 - Cursor functions 110
 - Go To Performance 111
 - Horizontal cursor 110
 - Keep cursor on same branch 112
 - Normalize 110, 111
 - Remove all objects 109
 - Restore limit scales 109
 - Same Y scales 112
 - Tag horizontal 111
 - Tag left cursor 111
 - Tag right cursor 111
 - Tag vertical 111
 - View baseline 109
 - View data points 109
 - View horizontal grids 109
 - View minor log grids 109
 - View ruler 109
 - View tokens 109
 - View vertical grids 109
- Cursor mode 116, 117
- Cursor positioning modes 118
 - Bottom 118
 - Go To X 111

- Go To Y 111
- High 118, 119
- Next 118
- Peak 118
- Top 118
- Valley 118
- Magnifying waveforms 114
- Modes
 - Cursor 113
 - Point tag 113
 - Scale 113
 - Vertical tag 113
- Panning waveforms 115
- Security key 20, 23
- Seed 162
- Select mode 47, 52
 - Definition 39
- Selection 57
- SETUP program 22
- Shape 39
- Short distance VT matching 11
- Shrink command 61
- Smith charts 10, 88, 97
- SPICE 24, 62
- SPICE file probing 10
- Standard deviation 160
- State variables
 - Defined 78
 - Editor 80
- Stepping
 - AC example 150
 - DC example 150
 - Decade 155
 - Dialog box 146
 - How it works 145, 146, 148
 - Octave 155
- Subcircuit maker 12
- Subckt
 - Creating 182
 - Definition 181
- Symbolic derivative 10

T

- Tag horizontal command 111
- Tag left cursor command 111
- Tag right cursor command 111
- Tag vertical command 111
- Temperature 67, 87, 103
- Terms and concepts 37
- Text
 - Adding 64
 - Mode 50
 - Stepping 152
- Text stepping 10, 152
- Timer 11
- Title block 191
- Tmax 67
- Tmin 67
- Tokens command 109
- Tolerances 158
- Top cursor position mode 118
- Trackers 110
- Transient analysis 65
- Transient Analysis Limits 25

U

- Undo command 41, 55
- Uniform distribution 160
- Units 14
- USB 20
- User-specified paths 10
- User's Guide 20

V

- Variable 1 (DC) 102
- Variable 2 (DC) 103
- Variables list 66, 72
- Vertical tag mode 113

W

- Waveform branch 68
- Waveforms
 - Analog 73
 - Digital 73
- Windows operating systems 21
- Wire
 - Definition 38
 - Mode 49, 64
- Wizards

Import 188
Working with macros 171
Working with subepts 181
Worst case 160

X

X expressions 89

Y

Y expressions 89

Z

Zoom-In command 60
Zoom-Out command 60