



## Contents

The following Help topics are available:

[About Message Viewer](#)

[Starting Message Viewer](#)

[Using Files](#)

[Setting Options](#)

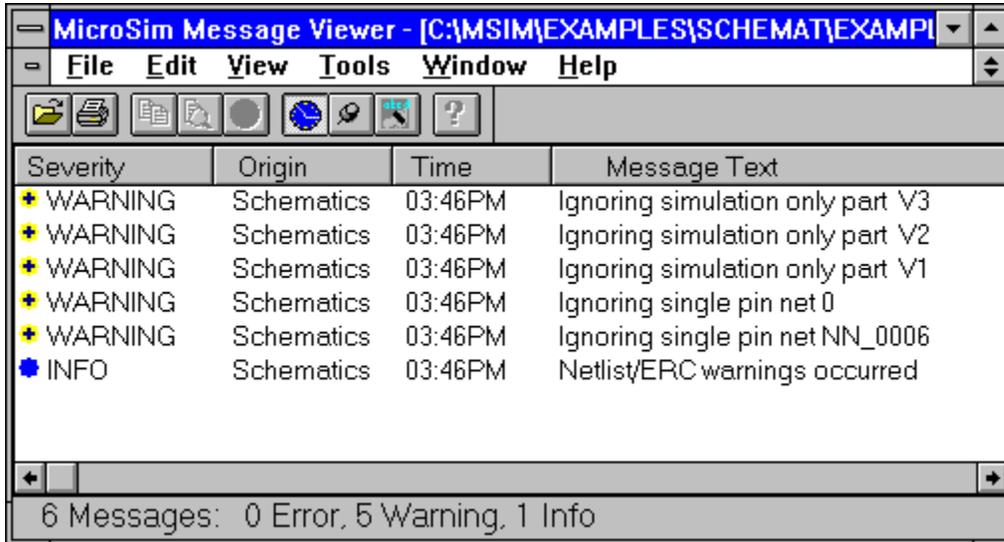
[Using Messages](#)

For help on help, press F1.

## About Message Viewer

The Message Viewer allows you to see and act on messages generated by the MicroSim applications.

Message Viewer shows the messages for each design on a separate window. When you close the design, the Message Viewer window for that design saves and closes as well.



## Starting Message Viewer

The Message Viewer appears automatically when an error or warning occurs that needs your attention.

You can also start the Message Viewer from the MicroSim applications. The Message Viewer automatically displays any messages in effect the last time you closed the design.

### To start the Message Viewer

∅ Do one of the following:

- From the File menu, choose View Messages.
- Press the F10 key.



#### See also

[About Message Viewer](#)

[Setting Options](#)

[Using Files](#)

## Using Files

Using files in Message Viewer is the same as in any Windows application.

[To open a file](#)

[To close a file](#)

[To print a file](#)



### See also

[Finding Message Information](#)

**To open a file**

Opens a message file for use.

- 1** Do one of the following:
  - Click the [Open icon](#).
  - From the File menu, choose Open.
- 2** Select the file from the list.
- 3** Click OK.

**To close a file**

Closes the active message display. The messages are saved in a binary file named *DESIGN.MVI*.

- ∅ From the File menu, choose Close.

The file closes.



The Message Viewer automatically closes the associated message file when you close a design.

**To print a file**

Prints the current file.

- ∅ Do one of the following:
  - Click the [Print icon](#).
  - From the File menu, choose Print.

## Setting Options

You can set options about how messages appear in Message Viewer.

### To set options

- 1 From the Tools menu, choose Options. The [Options dialog box](#) appears. All options are on by default.
- 2 Select the following:
  - [Show ToolBar](#)
  - [Show Status Bar](#)
  - To select the display font, click the Fonts button. In the [Fonts dialog box](#), select a font to use in the Message Viewer window when errors are displayed. When you are done, click OK.
- 3 Click OK.



### See also

[About Message Viewer](#)

[Starting Message Viewer](#)

[Using Files](#)

### **To set always on top**

Places the Message Viewer in the foreground, regardless of the position of other applications.

∅ Do one of the following:

- Click the [Always On Top icon](#).
- From the View menu, choose Always On Top.

## Using Messages

MicroSim Applications can place messages in the Message Viewer to inform you of status or to alert you to a problem.

Each message includes:

- Severity indication
- The name of the application that produced the message
- Time of the message
- Brief description of the condition or problem

With the exception of information messages, each message has a detailed explanation through on-line help. When applicable, messages also allow you to find the source of a problem within your design - whether the problem is on a schematic page or in a text file.

[To get help on an error message](#)

[To find in design](#)

[To find the source of a problem in your design](#)

[To see TimeStamps](#)

[To copy message information](#)

[To clear the message window](#)

[To set always on top](#)



### **See also**

[About Message Viewer](#)

[Setting Options](#)

[Expanded Text](#)

### **To copy message information**

Copies error text to the clipboard, where you can paste it into another application.

- 1** Click the error message.
- 2** Do one of the following:
  - Click the [Copy icon](#).
  - From the Edit menu, choose Copy.

### **To find in design**

Starts the appropriate tool, and positions the cursor on the source of the problem.

**1** Click the error message.

**2** Do one of the following:

- Click the [Find In icon](#).
- Right-click. From the menu that appears, choose Find In Design.
- From the Edit Menu, choose Find in Design.

The item or area that caused the error appears highlighted on the screen.

### To find the source of a problem in your design

If the selected message contains more information, you can see more information about the error.

- 1 Click the error message.
- 2 Do one of the following:
  - Click the [Additional Information icon](#).
  - Right-click. From the menu that appears, choose Additional Information.
  - From the Edit Menu, choose Additional Information.The Additional Information dialog box appears. [More information](#) about the error appears.
- 3 When you are done, click OK.

### To see TimeStamps

Shows or hides the Time column. The time values indicate when the messages were received by the Message Viewer.

∅ Do one of the following:

- Click the [TimeStamps icon](#).
- From the View menu, choose TimeStamps.

**To clear the message window**

Clears the currently-selected message area.

∅ Do one of the following:

- Click the [Clear icon](#).
- From the View menu, choose Clear.

### **To get help on an error message**

Help is available for error messages displayed in the message window.

**1** Click an error message.

**2** Do one of the following:

- On the icon bar, click the [Help On icon](#).
- Right-click. From the menu that appears, choose Help On.

The on-line Error Message help appears, showing you help on the error. The search window is open if you want to search another error.

**Menu bar**

Click a menu bar option to see a menu.

**Tool bar**

Hold the cursor over a button for a few moments to see a short description of the button.

**Message display**

Displays errors, warnings, and important information about the current process.

By selecting an option, you can:

- Get an explanation and solution using the on-line help
- Find the source of the condition or problem in the design
- Get additional supporting information specific to the message

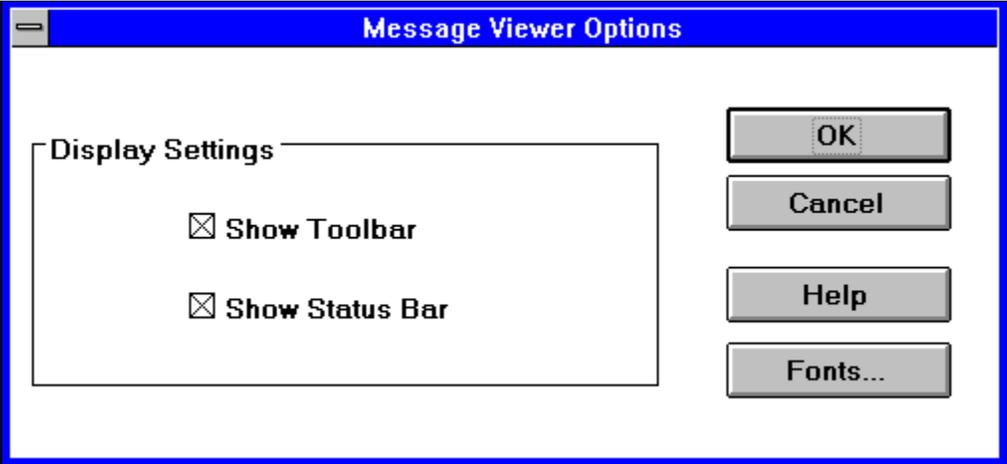
**Status bar**

Provides a brief description of the current activity or selected option.

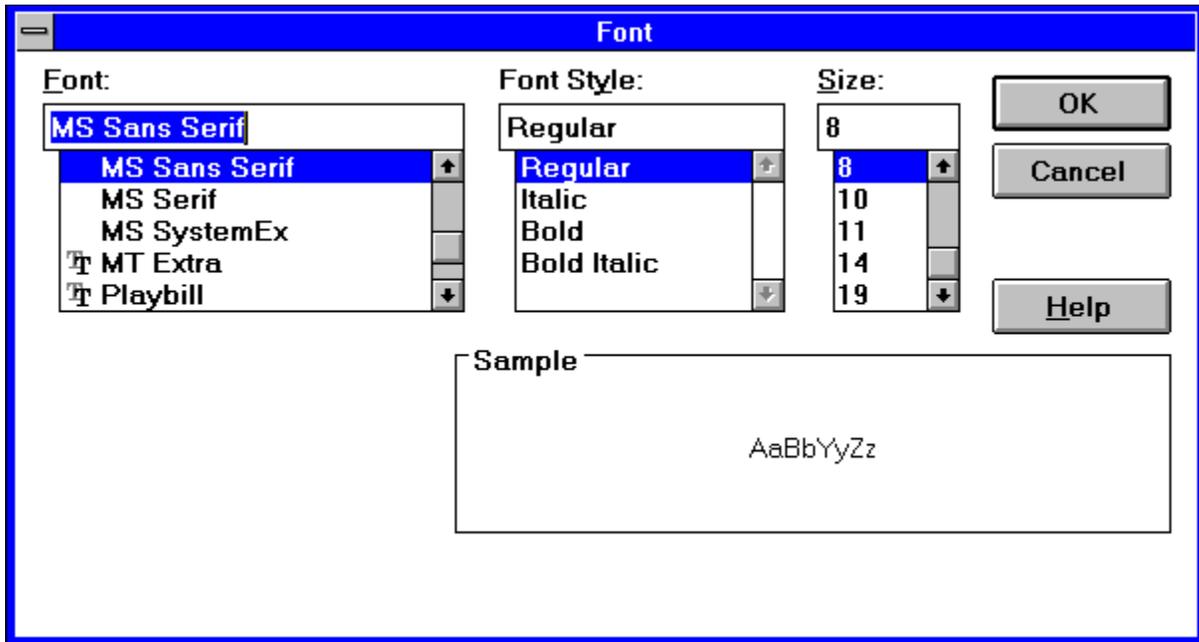
**Scroll bar**

Click to see more of the messages listed in the Message display area.

Options dialog box



Fonts dialog box



**Show ToolBar**

Click to display the tool bar.

**Show Status Bar**

Click to display the Status Bar.

**OK**

Click to accept your changes and close the window.

**Cancel**

Click to abandon your changes and close the window.

**Help**

Click to see the on-line help system.

**Fonts button**

Click to go to the Fonts dialog box to change the display fonts.

**Title bar**

Displays the name of the dialog box.

**Close**

Closes the dialog box.

**Font**

Lists the available fonts.

**Font Style**

Lists the available font styles for the selected font.

**Font Size**

Lists the available font sizes for the selected font.

**Sample**

Shows how the selected font will appear with the selected font settings.

**Script**

Lists the available language scripts for the selected font.

**Help**

Click to see the on-line help system.

### more information

Each message is preceded by a dot graphic. The color of the marker indicates the severity of the message.

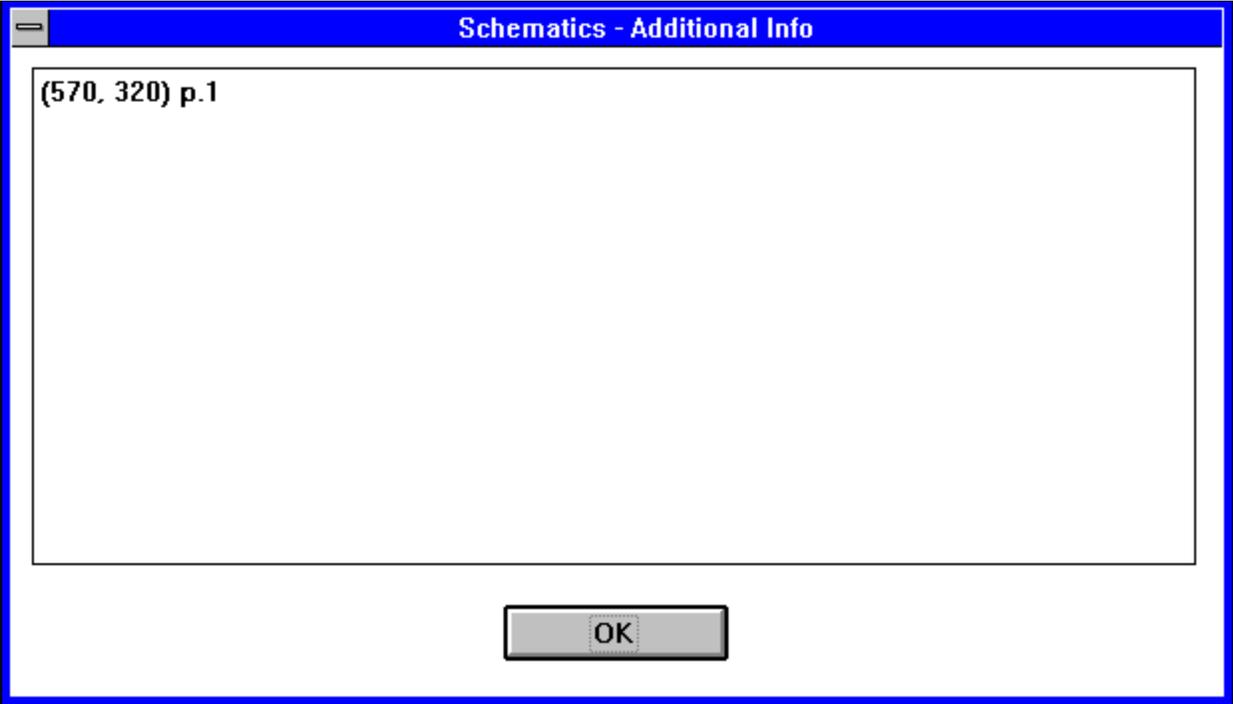
Color	Indication
Blue	Informational message. No action is required.
Yellow	Warning message. Might require some action.
Red	Error message. Syntax errors and ERC errors that prevent the program from producing the intended results.
Black	Fatal error message. A non-recoverable error condition.

### Expanded Text

Some messages contain [expanded text](#). Expanded text lines are preceded by a plus sign and the More Information icon is active on the tool bar.

- 1 Do one of the following:
  - Click the plus sign in the line of text
  - Click the Expand icon.The [Additional Information dialog box](#) appears.
- 2 Click OK when you are done.

Additional Information dialog box



Help icon



**File Open icon**



**Print icon**



**Always on Top icon**



Copy icon



Find icon



**Additional Information icon**



**TimeStamps icon**



Clear icon



**Additional Information**

More information about the condition or error is shown here.

**Severity**

The level of severity of the message is shown.

### **Message Display**

Click an individual message to:

- Get a more complete explanation and solution using the on-line help.
- Find the source of the problem in the design.
- See additional supporting information specific to the message.

**Time**

The time the message was generated.

**Message Text**

A brief description of the condition is shown.

### Severity

The severity of the message is shown. A plus sign in the marker indicates that more information is available by clicking the .

#### Information

No action is needed.

#### Warning

May require some action.

#### Error

Must correct before continuing.

#### Fatal

A non-recoverable condition has occurred/

### Origin

The MicroSim application that produced the error.

**Time**

The time the message was generated.

**Message Text**

A textual description explaining the condition or problem.

## Glossary

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

### A

[AC sweep](#)

[annotation](#)

### B

[back annotate](#)

### C

[CSDF files](#)

### D

[DC sweep](#)

[display](#)

## **E**

[expressions](#)

## **F**

[Fourier Transform](#)

## **G**

[global file](#)

[goal function name](#)

## **H - I**

[histograms](#)

## **J - K**

[keyboard accelerator](#)

## **L**

[label](#)

[local file](#)

## **M**

[macros](#)

[Monte Carlo analysis](#)

## **N - O**

[net](#)

[netlist](#)

## **P - Q**

[Performance Analysis](#)

## **R - S**

[schematic](#)

[simple Performance Analysis](#)

[system-level DSL file](#)

## **T - Z**

[text files](#)

[trace](#)

[transient](#)

[type](#)

**alias**

Relates the schematic names for parts and signals to netlist names. An alias is an exact electrical equivalent that can be used to reference a symbol.

**annotation**

The process by which parts are labeled when they are placed on a schematic. MicroSim Schematics allows the labels to be placed automatically or manually.

**annotation symbol**

Has no electrical equivalent. It is used to clarify, or define, items on the schematic.

arrays

## **attributes**

Special characteristics such as a name and an associated value contained in a part instance or definition. Attributes can be changed in the Schematic Editor or the Symbol Editor.

Parts have attributes such as resistor values, simulation models and package types. Wires can have attributes such as labels and trace widths. Attribute values may or may not be displayed on the schematic.

**back annotate**

Apply modifications to part attributes in a schematic. Items modified include part references, pin numbers, swapping gates, and swapping pin numbers.

**bookmark**

A marker you can use to navigate through a schematic.

**child schematic**

A schematic represented on parent schematic by a hierarchical block. The child schematic is a lower level than the parent. A child schematic contains circuitry referenced by the parent schematic. A child schematic can have children schematics of its own.

## **DSL**

**Design Synthesis Language** is a high-level behavioral language used to describe programmable logic. DSL provides constructs for state machines, truth tables, and boolean equations, as well as programming language constructs such as procedures and functions, IF and CASE statements, and macro capabilities.

DSL source files have the file extension `.DSL`.

## **DSL blocks**

A hierarchical block in a schematic which contains a DSL definition. A DSL block corresponds to a PROCEDURE definition in a DSL source file. The procedure name is defined by the PLMODEL attribute on the block. The DSL source file is associated with the block when the block is created.

When you use DSL blocks in a schematic, PLSyn automatically compiles the blocks' DSL source files when you create a netlist.

**expressions**

A combination of one or more identifiers, signals, and/or constants that are related by operators.

**flip-flops**

**footprints**

**functions**

**global library**

A library is a collection of symbols. A global library allows any schematic, new or existing, to use the symbols in it. When looking for a symbol, MicroSim Schematics always searches the local libraries before searching the global libraries.

**hierarchical block**

A symbol on a schematic that represents or refers to a child schematic. The connection point is a hierarchical port. You place hierarchical ports on a schematic and specify the child schematic.

**hierarchical design**

A design structure that involves a parent schematic and child schematics. The parent schematic has symbols specifying the child schematic.

**hierarchical port**

A symbol on a schematic page that specifies a signal on one schematic page connects to another signal on another schematic page. The symbol usually includes text specifying the name and type of signal. A hierarchical port can be part of a hierarchical block.

**junction dot**

Indicate connectivity. Junction dots appear wherever three or more pins, wire or bus segments meet. Junction dots are created and removed automatically by MicroSim Schematics.

**library**

A library is a collection of symbols. You can use the libraries that are provided with MicroSim Schematics or you can create your own.

**local library**

A library is a collection of symbols. A local library is only available to a schematic you specify.

When looking for a symbol, MicroSim Schematics always searches the local libraries before searching the global libraries.

**# name**

Specifies the attribute name. If the name of the attribute is the same as that of another part in the list, the part in the list is modified.

**netlist**

An ASCII file that lists the connections of a schematic by naming the connected signals, parts, and pins. MicroSim Schematics automatically creates the netlist.

node

**nonprimitive**

A part on a schematic representing a child schematic.

**off-page connector**

An object on a schematic page that send signals from one page to another page.

**operators**

**orthogonal**

Perpendicular to each other.

**package**

A physical part that contains more than one logical part. Each part in a package has a unique part number.

## **packaging**

Packaging is the process of collecting individual gates into physical packages and reassigning reference designators and gate names to replace how they are package.

Package definitions contain information about the number of gates per package, gate names and pin numbers for the different gates. The packager uses package definitions contained in the package libraries (.PLB).

**pan**

To change the part of the schematic that is in view on the screen as you drag a part across the screen.

## **parent**

A hierarchical schematic that contains child schematic. The parent contains the main design. A child can be a parent to another schematic, resulting in several layers of hierarchy.

**part alias**

A copy of a part using a different name in the library. A part alias uses identical graphics, attached schematics and properties as the original, but is referenced by a different part value.

**# part name**

The most commonly used full name of the part. This name, or an alias, is the name that is used to create netlists and to bring an instance of the part into the schematic.

### physical node

A node which, when fit, occupies a physical location in a device, such as an I/O pin or macrocell. The physical designation is the opposite of virtual, in which case the node is collapsed out by the Optimizer.

You can designate a node as physical in one of three ways:

- 1 Use the PHYSICAL modifier on a NODE definition in a DSL source file.
- 2 Attach the PHYNODE/PL symbol found in DIG\_PRIM.SLB to a wire on the schematic.
- 3 Include the node's name in one of the .PI file constructs.

The fitter requires control signals (clocks, resets, presets, enables) to be specified as PHYSICAL when the control has a dedicated pin on the device. Therefore, the Optimizer normally forces all control signals to be physical. Overridethis behavior by selecting Collapse Control Signals.

procedures

### **rubberbanding**

Rubberbanding stretches wires when you move the objects the wires are attached to. Rubberbanding allows you to maintain connectivity. The default is to have rubberbanding turned off.

- To turn rubberbanding on, press F9.
- To turn rubberbanding off, press SHIFT+F9.

**schematic**

A representation of a circuit using a standard set of electronic parts. MicroSim Schematics is a tool to graphically draw and edit a schematic.

**schematic page**

The sheets of paper on which a schematic is drawn. In MicroSim Schematics, pages are shown on the screen in the Schematics Editor.

**state machines**

### **system-level DSL file**

A DSL source file which contains system-level signals and statements.

In schematic-based designs, system-level inputs and outputs are defined in the top-most schematic. Therefore, any system-level constructs in a DSL block source file is ignored.



You may use PLSyn to process system-level DSL files. Currently, only schematic-based designs can simulated.

### **title block**

The title block is automatically placed in the lower right corner of a schematic. It contains such information as author name, schematic name or title, page number, total number of pages, revision date, and name of the file.

**truth tables**

### virtual node

A node which is collapsed out by the Optimizer. The Optimizer collapses out a node by passing its equation to all equations that reference the node. The virtual designation is the opposite of physical, in which case the node is forced to occupy a physical location in a device.

Designate a node as virtual in one of two ways:

- 1 Use the VIRTUAL modifier on a NODE definition in a DSL source file.
- 2 Include the node's name in the VIRTUAL .pi file construct.

The fitter requires control signals (clocks, resets, presets, enables) to be specified as PHYSICAL when the control has a dedicated pin on the device. The Optimizer normally forces all control signals to be physical. This behavior can be overridden by enabling Collapse Control Signals.

**net**

All of the wires, parts, buses, and symbols on a schematic that are connected using net names, ports, aliases, and off-page ports.

**size, orient., hjust, vjust**

Sets the size, orientation, and justification for pins.

**Description**

The textual description of the symbol.

**Part Name**

The most commonly used full name of the part. This name, or an alias, is the name that will be used to create netlists and to bring an instance of the part into the schematic.

**Alias List**

The exact electrical equivalents of the part that can be used to reference the symbol.

### **A Kind Of (AKO) name**

A symbol that uses the graphics, pins, and attributes of the named part. Attributes on the original symbol is overridden by those from the new symbol, and new attributes may be added.

If graphics and/or pins already exist for the symbol, you are prompted to verify whether you want them replaced with those of the symbol referred to in the AKO. For symbols specifying an AKO, the following rules apply:

- The symbol it refers to must reside in the same library.
- You can only edit the attributes and the definition.

**Type**

Symbol types include: primitive, port, annotation, border, or title block. Only primitive and port symbols contain electrical properties. Annotation, border, and title block symbols have no electrical significance.

### **System attributes**

System attributes are attributes required for netlisting to PSpice/PLogic, are used in the title block, or are used in the Special Symbols.

**Name**

Specifies the attribute name. If the name of the attribute is the same as that of another part in the list, the part in the list is modified.

**Value**

Specifies the attribute value.

**Changeable in Schematic**

Determines whether the attribute can be edited in the schematic editor.

**Keep Relative Orientation**

Specifies whether the attribute text is rotated when the symbol is rotated in the Schematic Editor.

**Display Value**

Specifies whether the attribute value is shown on the schematic.

**Display Name**

Specifies whether the name is shown on the schematic.

**Layer**

Specifies the display layer for the attribute text.

**Save Attr**

Temporarily incorporates your changes to the list of attributes for a particular part.

**Del Attr**

Deletes an attribute from the list. Either select the attribute from the list or type in the name of the attribute you want to delete and click this button.

**Select All Parts**

Prints all parts in the current library.

**Clear Selections**

Clears any parts you have selected from the list.

**Symbol Image**

Prints the symbol graphics.

## Symbol Data

Prints the symbol definition.

## **Attributes**

Prints the symbol attributes.

**Pin Data**

Prints the pin information.

**Auto-Fit**

Fits one symbol on the printer page.

### **User-Definable Zoom Factor**

At the default of 200% setting, symbols are always printed in constant relation to each other. Leave this set to the default, unless you want to print an enlarged or smaller image of a symbol.

reference designator

**output variables**

**keyboard alternatives**

A combination of keys that are equivalent to selecting a menu item or dialog box item. They are often shortcuts to doing something in a Windows application.

For example, Control+C copies the selected item.

### **.PRB file**

A .PRB file is an ASCII file containing three sections, one for macros, one for goal functions, and one for displays. Each section begins with a header. The name of the header is enclosed in brackets.

Two .PRB files are automatically loaded when a data (.DAT) file is opened: a global file and a local file. The global .PRB file is specified in the [PROBE] section of MSIM.INI by the parameter PRBFILE=<filename>. The local .PRB is located in the same directory as the data file and has the same name as the data file, but with a .PRB extension.

By default, Probe loads the global file first, then each local file, each time a new data file is opened. If a .PRB file contains any macros that have the same name as a macro that is already loaded, then the new macro replaces the existing macro.

Each plot window can have its own local .PRB file. Each plot window can use one or more . .PRB files at the same time.

## macros

A macro can be directly substituted into an expression.

A macro can have a maximum of 80 characters, including macro name and arguments. A line beginning with an asterisk \* is a comment line. In-line comments are marked with a semicolon after the macro definition. Blank lines are ignored.

The macro format is:

`<macro name>[(arg[,arg]*)] = <definition>`

Some macro examples:

`ADD(A,B) = A+B`

`10X(A) = 10*A`

`100X(A) = 10*10X(A)`

`PI = 3.14159`

## histograms

Histograms are vertical bar charts that show the number of occurrences of an event during each of several divisions of the Y variable. Thus, the X axis becomes divisions of the Y variable, and the Y axis become the number of occurrences represented by percent of samples.

In Performance Analysis, the goal function value calculated from each Monte Carlo run is an occurrence.

### **Probe Binary Files**

Binary files are compact and provide the quickest access to the waveform data. However, binary files can present problems if you want to manipulate the data, or when you want to transfer the data file to a different type of computer. There are several problems that you can encounter. These include:

- the binary formats for real numbers differ between platforms
- the link between computers can not handle binary data adequately
- the binary data format is difficult to access or process

For more information about how to transfer files between computers, see your network software manual.

### **Text or CSDF Files**

The Common Simulation Data File (CSDF) format has been proposed as a standard format for results from simulation programs. When you specify the CSDF option, the simulation results are generated in a generic manner so that any computer link can handle the file transfer and any computer platform can process the data.

When Probe accesses a CSDF file, it automatically recreates the compact binary file, maintaining the performance benefits.

**global file**

A file specified in the MSIM.INI. It is automatically read when a data file is opened.

**local file**

A file with the same name and a different extension as the data file. It is automatically opened when the data file is opened.

**Monte Carlo analysis**

A mathematical model constructed from randomly selected components taken from a statistically representative population.

**trace**

A trace can be either a digital signal or a bus made up of a maximum of 32 digital signals.

### **simple Performance Analysis**

A simple Performance Analysis trace uses a single Goal Function.

## **goal function**

A goal function is a set of instructions used in MicroSim Probe. Goal functions are used to evaluate characteristics of a waveform according to a set of criteria. Goal function definitions specify the procedure for measuring the waveform characteristic, and assign the function a name.

Goal Functions contain definitions for specifying search commands, marking points on a curve, and evaluating expressions of marked points.

A Goal Function calculates a single value, such as a maximum point on a curve, or the center frequency of a signal. Goal Functions are used in Performance Analysis for simulations with multiple runs to show how a derived value changes.

**transient**

A PSpice analysis type that steps through time.

**AC sweep**

A PSpice analysis type that sweeps through an average range of frequencies.

**DC sweep**

A PSpice analysis type that sweeps through an average range of DC values.

## **display**

A display includes labels, a data range and the plot specifications. You can create a display showing traces with labels and data range specifications.

You can save the display to use again to view the results of a different simulation with identically named variables as saved in the display.

## **label**

A label annotates your plot with either text or graphical items. All labels become part of the plot, so they appear on hard copies and are saved with display control.

## **Fourier Transform**

Use a Fourier analysis to evaluate DC, fundamental, and harmonic components of a complex wave.

## **Performance Analysis**

Using a Performance Analysis allows you to view how some characteristic of a waveform changes or varies when a variable is changed. between several simulation runs.

You must have multiple simulation runs to do a Performance Analysis.

