Personal Computer
Circuit Design
Tools

intusoft

GETTING STARTED

Tel. (310) 952-0657
Fax (310) 952-5468
web - www.intusoft.com
email - info@intusoft.com

intusoft provides this manual "as is" without warranty of any kind, either expressed or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

This publication may contain technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of this publication.

Printed in the U.S.A.

rev. 09/07 Build 3090

Table Of Contents

ICAP/4

New Features Overview

ICAP/4 delivers a host of new and enhanced features including new interactive simulation capabilities and improved integration between the schematic, simulation, and post processor.

New Interactive SPICE...

SPICE Like You've Never Seen Before

ICAP/4 contains IsSpice4, the first and only interactive version of SPICE

Real Time Waveform Display:
View circuit waveforms as the simulation runs. Add, delete, or re-scale waveforms at any time.

Full Native Mixed Mode Capability:
IsSpice4 combines the SPICE 3 core with a 12-state digital logic simulator to allow mixed analog and digital circuits to be simulated at the same time.

Use New C Code Models:
Access user/vendor developed models written in a C language- based HDL. Over 50 code models are already included.

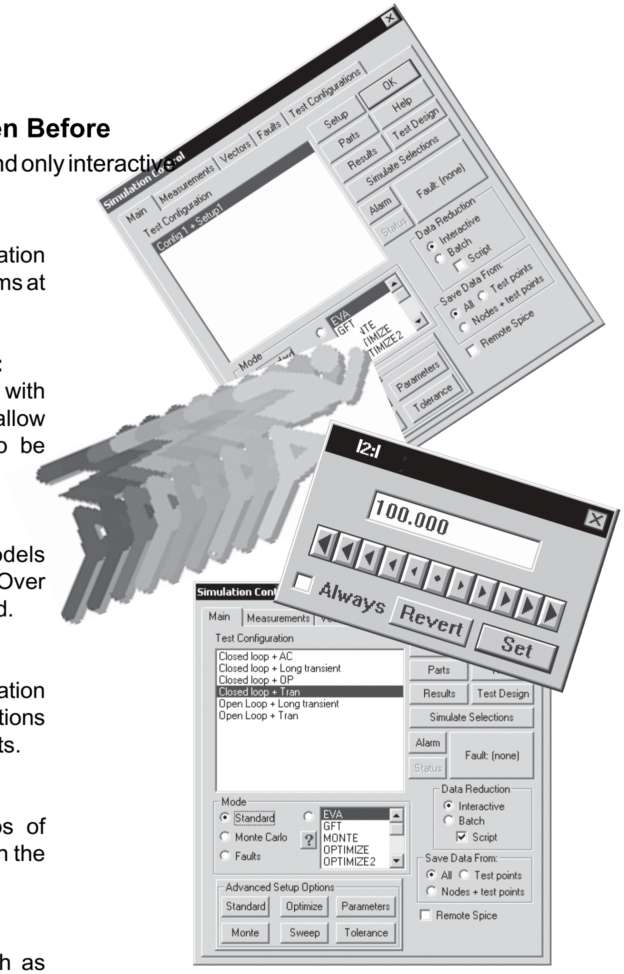
Control The Simulation:
Start, stop, pause, or resume the simulation at will. Interactively run different simulations and alter values using powerful scripts.

Interactive Stimulus:
Sweep single parameters or groups of interdependent parameters and watch the circuit performance changes.

Simulation Templates:
Advanced worst case analyses such as EVA, RSS, and Sensitivity.

Monte Carlo:

Perform statistical analysis early on, including generating histogram and probability plots and reports for a variety of electrical measurements.



I2:I

100.000

□ Always Revert Set

Simulation Cont

Main | Measurements

Test Configuration
- Closed loop + AC
- Closed loop + Long transient
- Closed loop + OP
- Closed loop + Tran
- Open Loop + Long transient
- Open Loop + Tran

Parts | Results | Test Design | Simulate Selections | Alarm | Status | Fault: (none)

Mode
- ● Standard
- ○ Monte Carlo
- ○ Faults
- ○ EVA / GFT / MONTE / OPTIMIZE / OPTIMIZE2

Data Reduction
- ● Interactive
- ○ Batch
- ☑ Script

Save Data From:
- ● All ○ Test points
- ○ Nodes + test points

□ Remote Spice

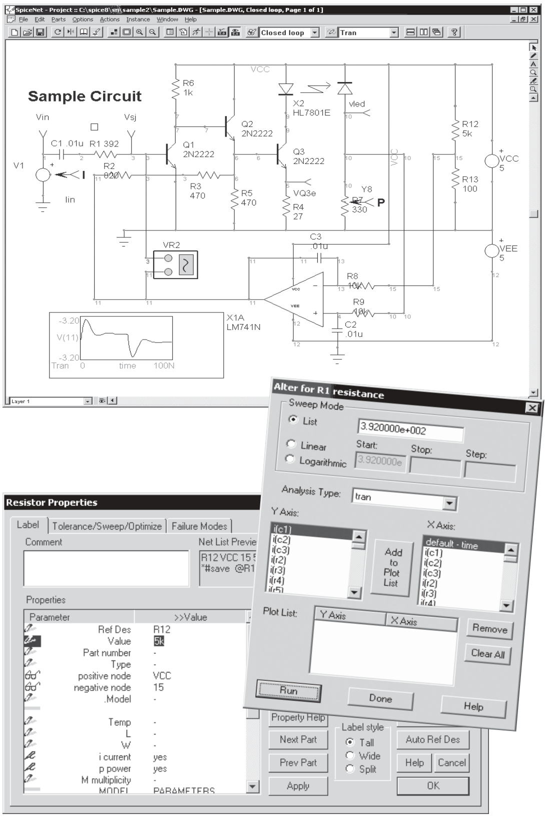
Advanced Setup Options
Standard | Optimize | Parameters | Monte | Sweep | Tolerance

Optimization:

Measure a design objective, with automated selection of component values, such as output voltage, unity gain and filter passband.

Sweep:

Set a start point, end point and step size for up to two parameters simultaneously and sweep the values.



Alter for R1 resistance

Sweep Mode: ● List 3.920000e+002 ○ Linear ○ Logarithmic Start: 3.920000e Stop: Step:

Analysis Type: tran

Y Axis: i(c1) i(c2) i(c3) i(r2) i(r3) i(r4) i(r5)

X Axis: default - time i(c1) i(c2) i(c3) i(r2) i(r3) i(r4)

Plot List: Y Axis X Axis

Resistor Properties

Comment

Net List Preview: R12 VCC 15 5 *#save @R1

Properties

Parameter	>>Value
Ref Des	R12
Value	5k
Part number	-
Type	-
positive node	VCC
negative node	15
.Model	-
Temp	-
L	-
W	-
i current	yes
p power	yes
M multiplicity	-
MODEL	PARAMETERS

Label style: ● Tall ○ Wide ○ Split

Unparalleled Power In The Schematic

Multiple Configurations:
Place/access design variations and production drawings in a single schematic.

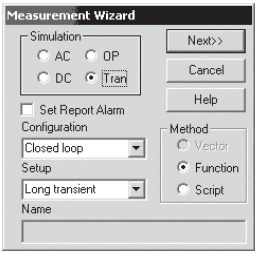
Multiple Simulation Setups:
Save several sets of analyses and combine them with various circuit configurations to make multiple test scenarios.

Interactive Cross-Probing:
Cross-probe directly into schematic to view waveforms in waveform viewer.

Easy Parametric Function:
Sweep a component's value and display the family of curves with one button click.

Analysis and Convergence Wizards:
Easily create your analysis setup and overcome tough convergence problems automatically.

Attribute, Model, Subcircuit Editing:
On-the-fly editing of all SPICE attributes alleviates the need to know SPICE syntax.


Measurement Wizard

Simulation: ○ AC ○ OP ○ DC ● Tran

□ Set Report Alarm

Configuration: Closed loop

Setup: Long transient

Name:

Method: ○ Vector ● Function ○ Script

Failure Analysis:
Quickly simulate failures, out-of-tolerance conditions, or other user-defined device behavior without changing the appearance of the schematic.

Automated Measurements:
Specify a variety of electrical measurements for numerical readout and pass/fail anywhere in design.

The Biggest and Best Model Libraries

23,500+ Models:
The main ICAP/4 libraries, along with the Power Device Library, Mechatronics Library and RF Device library, contain over 23,500 models.

Modeling Support:
ICAP/4 provides a wealth of model creation tools, including datasheet entry for 13 types of discrete devices, C-code, B-element and generic templates for active devices.

Introduction

Welcome to Intusoft's ICAP/4 circuit simulation environment for Windows 98SE/ME and NT/2000/XP. This manual provides complete installation instructions and a tutorial guide for ICAP/4.

We thank you for purchasing Intusoft's design simulation tools. As you know, design simulation and verification is a critical part of the overall product cycle. ICAP/4 will provide you with capabilities that shear time to market and production expense, by thoroughly testing designs for accuracy and production variations early on. Using computer-aided capability, the tools will also augment your insight into designs in ways not possible using conventional lab equipment and debug methodologies.

What's In the ICAP/4 Package

Your package should include:

ICAP/4 CD-ROM
Getting Started Manual
IsSpice4 User's Guide, Volumes 1 and 2
Working with Model Libraries
Vendor-Supplied IC/Op-Amp Models

The ICAP/4 Deluxe Options (RF or Power) also include RF or Power Device Model libraries respectively, SpiceMod5 for Windows, and the following materials:

Spice Applications Handbook (CD)
A Spice Cookbook (Manual and CD)

Hardware & Software Requirements

- Windows 98SE/ME or Windows NT4/2000/XP/Vista
- Any computer with Intel or AMD processor
- Hard disk (approximately 100 megabytes of free disk space)
- At least 128 megabytes of RAM, or more, recommended

Hardware Protection Key Installation

Included in your package is a copy protection key that MUST be connected to your computer before the software is installed.

To install the copy protection key;

- Connect the key to the computer's USB or parallel port.

The key has a 25-pin connector at each end and should be plugged directly into your parallel port. The key should be located as close to the port on the back of the computer as possible. If you already have an output device connected to the parallel port, install the key between the device and the port. The protection key is transparent to other users of the parallel port and does not interfere with data sent to or from the port.

DO NOT LOSE THIS KEY! The ICAP/4 software package can not be run without it.

Troubleshooting Hardware Protection Key Operation
Although every effort has been made to ensure the highest level of quality and compatibility, the protection key, like any other PC peripheral, might not run on certain PC configurations.

If you have a problem with the key, it can usually be overcome by checking the following:

1) Are the key and parallel printer card connected, and are their screws properly fastened? If they are not, re-seat them.

2) Is a printer connected to the same port as the key? If so, does it work properly? (Although this is a good sign, it does not necessarily mean that the port and the printer are fully operational). If there is a printing problem, try using another printer cable or another printer.

3) Make sure you're using the latest version of the HASP driver. Updates are available on the web www.intusoft.com/support.htm.

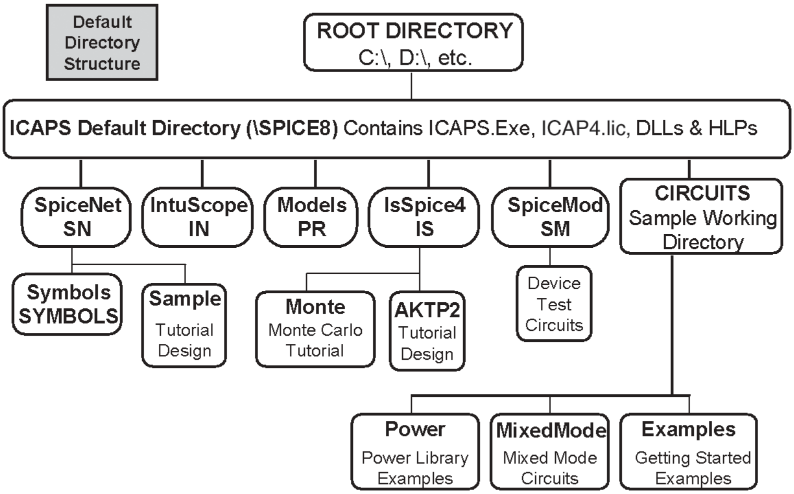
Installing ICAP/4Windows

CD Installation

- Insert the ICAP/4Windows CD-ROM into the CD drive. A Welcome program should start automatically.

  If it doesn't, navigate to the root of the CD-ROM drive and run the "Setup.Exe" program.

- Follow the instructions.


Default Directory Structure
ROOT DIRECTORY C:\, D:\, etc.
ICAPS Default Directory (\SPICE8) Contains ICAPS.Exe, ICAP4.lic, DLLs & HLPs
SpiceNet SN
IntuScope IN
Models PR
IsSpice4 IS
SpiceMod SM
CIRCUITS Sample Working Directory
Symbols SYMBOLS
Sample Tutorial Design
Monte Monte Carlo Tutorial
AKTP2 Tutorial Design
Device Test Circuits
Power Power Library Examples
MixedMode Mixed Mode Circuits
Examples Getting Started Examples

When complete, all software will be installed in the SPICE8 directory as shown above. The IS, PR, SN, IN, and (optional) SM directories under the SPICE8 directory must not have their names changed; the SPICE8 directory name may be customized at the time when the software is installed.

Installing Network Versions of ICAP/4

- Please refer to the ICAP/4 Network instructions, which are included within the server software setup.

Registration Information

Contained in this package is the Intusoft Software Registration card. In order to receive the Intusoft Newsletter, you must return this registration card. The Intusoft Newsletter is a valuable resource for modeling and simulating hints, and is your primary notification of product updates.

Please take this opportunity to fill out and mail or FAX your registration form to us.

Included with your software shipment are the following items:

Intusoft Newsletter, Newsletter Subscription Order Form, License Agreement and Software Registration Card.

Viewing Readme Files

The SPICE8 directory contains a Readme file that provides important information and last-minute documentation updates. Please read it before proceeding.

Where to Go for Information

SpiceNet, Schematic Entry - The SpiceNet schematic entry program is discussed in this Getting Started manual.

IsEd, Netlist Editing - The text editor is described in this Getting Started manual. It can be used to create or edit a circuit netlist that you wish to simulate, or to review the simulation's output text file, or to create or edit other text files, e.g., SPICE model library file.

IsSpice4, Simulation - Refer to the IsSpice4 User's Guide, or IsSpice4 tutorial in this manual, to enable you to efficiently simulate circuits, or optionally engage in the syntax associated with the IsSpice4 netlist description. An overview of the mixed-mode simulation features can also be found in this manual.

IntuScope - IntuScope's in-application help contains a complete description of each menu function in IntuScope. Also see the chapter dedicated to IntuScope in this manual.

Tutorials -- Several tutorials are provided to enable design simulation and verification in a minimum amount of time. In-depth tutorials on IsSpice4, SpiceNet and IntuScope provide a comprehensive rundown on the operation of these applications. The Design Validation and Automated Measurements tutorial (for ICAP/4 Windows, ICAP/4 Professional and Test Designer users) covers basic failure analysis using automatic measurements. The Test Designer tutorial (for Test Designer users) covers some of the basic functions of the Test Designer product. The Simulation Examples chapter provides more than 25 different sample simulations that address various key topics.

Models -- The Parts Browser dialog in the SpiceNet schematic entry program contains a complete listing of every model that is supported in your installed ICAP/4 version, including any optional libraries you may have purchased.

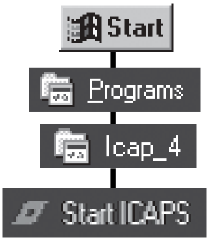
SpiceMod -- The Working with Models manual covers the basic modeling concepts and how to use SpiceMod. SpiceMod's in-application help lists all the datasheet parameters used for each model type, notes on how to obtain these values from the datasheet, and all important equations used.

Library Manager -- Library Manager is used to easily add and modify models for all non-ICAP/4 Rx packages. It also adds a test button to SpiceMod5 to automate model testing. It launches and simulates test drawings so you can compare at a glance simulated measurements with data entered into SpiceMod. Appendix D and E of the IsSpice4 User's Guide covers all aspects on how to use Library Manager to import a Spice model and export a schematic as a subcircuit netlist.

Code Models--A Code Model overview is furnished in the IsSpice4 User's Guide. Several examples are also in the Getting Started manual under both the Analog Simulation Examples chapter and Digital/Mixed Mode Examples chapter.

Tutorial #1 - Introduction to ICAP/4

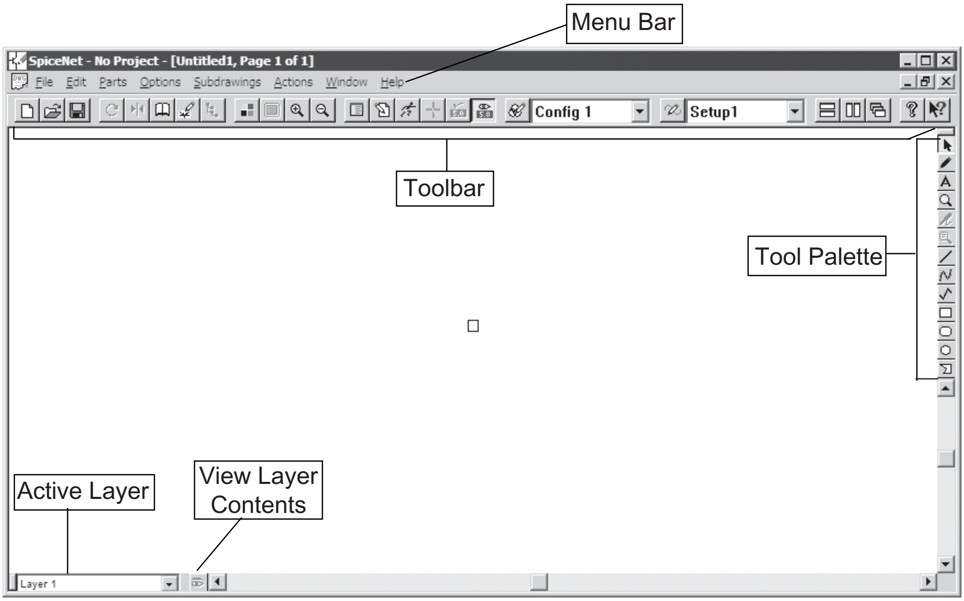
This chapter will familiarize you with ICAP/4's SpiceNet schematic capture program. SpiceNet is designed to dramatically ease the process of generating SPICE netlists. Netlists are necessary to run circuit simulations.



Now Launch SpiceNet to get started.

- **Select the Windows Start (Programs) entry.**

- **Choose the (ICAP_4) program group, (Start ICAPS) program.**


Menu Bar
SpiceNet - No Project - [Untitled1, Page 1 of 1]
File Edit Parts Options Subdrawings Actions Window Help
Config 1
Setup1
Toolbar
Tool Palette
Active Layer
View Layer Contents
Layer 1

If you are running your computer at a screen resolution greater than 1024 by 768 you will need to re-size the SpiceNet window to fill the entire computer screen. Move your mouse cursor to the edge of the SpiceNet window. The cursor will change into a double arrow cursor and you will be able to stretch the window by holding down the left mouse button and moving the mouse. Drag the title bar to reposition the window.

Let’s set your preferred SpiceNet window size and position.

- **Use the mouse to re-size and reposition the SpiceNet window.**

- **From the Options Menu, select Save Preferences.**

Other SpiceNet settings you can save from the Options menu:
1) **Ruler on/off** (Drawing, Show Ruler)
2) **Grid on/off and grid size** (Drawing, Show Grid)
3) **Drawing Page Size** (Drawing, Drawing Page Size)
4) **Drawing Item Colors**
5) **Rubberbanding on/off**

IsSpice4 Requirements for Simulation

To perform an IsSpice4 simulation, the software produces a netlist (or you can manually), which has the following elements:

- **Title line and .END line**
- **Component connectivity and attributes**
- **Stimulus, power, and ground**
- **IsSpice4 control statements** (i.e., via the Simulation Setup dialog)
- **Output control statements** (i.e., from Test Point symbols)

Note: Each circuit element must have a path to ground.

SpiceNet will take care of the Title line, .END statement and the circuit connectivity, which consists of unique reference designations for each device and the numbering of each node. You must add the following:

- **Component attributes**
- **Stimulus, power, and ground**
- **IsSpice4 control statements**
- **Output control statements**

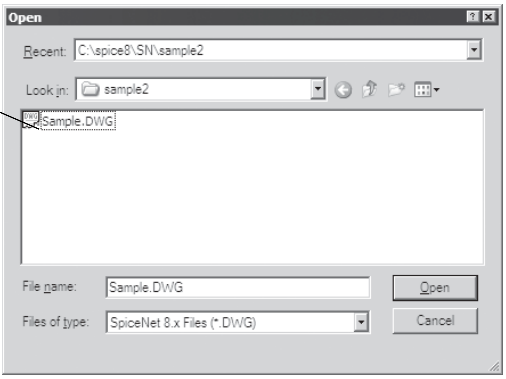
Remember: If even one of these four key pieces is missing, you will not be able to run a simulation.

Opening an Existing Project in SpiceNet

Let’s open a sample schematic (.DWG file) to work with:

- **From the File menu, choose Open, then navigate to the :\spice8\SN\sample2 directory.**

SpiceNet schematic files have a .DWG extension.



Open

Recent: C:\spice8\SN\sample2

Look in: sample2

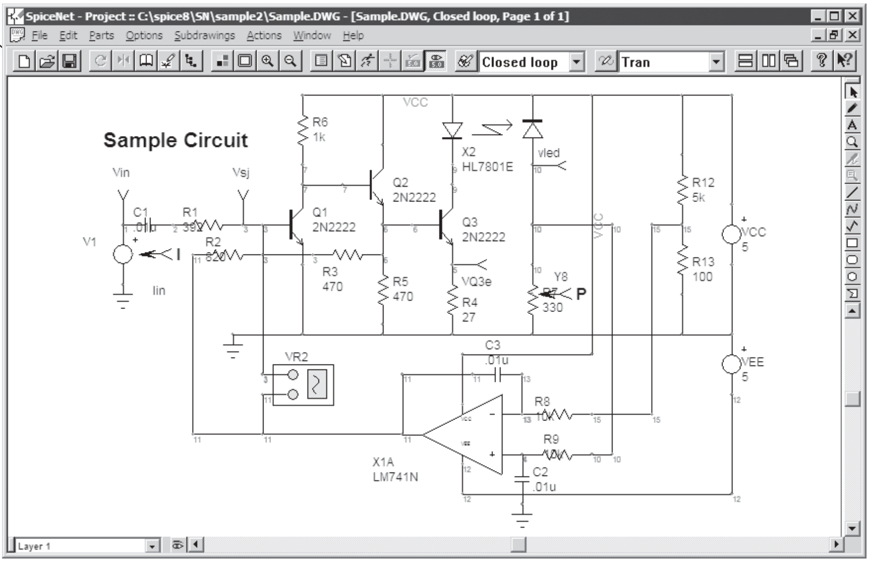
Sample.DWG

File name: Sample.DWG

Files of type: SpiceNet 8.x Files (*.DWG)

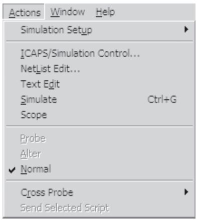
- **Left click on the Sample.DWG, then select Open**

SpiceNet Tool bar



SpiceNet - Project :: C:\spice8\SN\sample2\Sample.DWG - [Sample.DWG, Closed loop, Page 1 of 1]

Running a Simulation

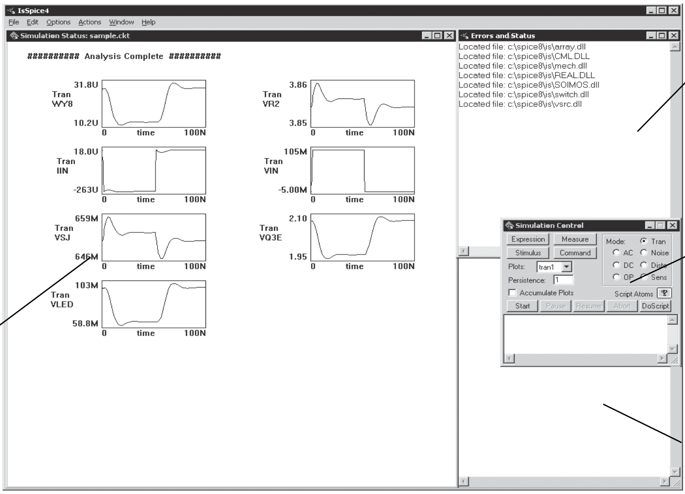


Let's run our first simulation on the Sample.DWG

- **From the Actions Menu, select Simulate, or click on the Run Simulation icon in the SpiceNet toolbar.**

This launches IsSpice4 and runs a simulation on the Sample circuit. The IsSpice4 Window consists of four separate windows, each with its own scroll bar:

Run Simulation icon.



########## Analysis Complete ##########

Located file: c:\spice8\is\array.dll
Located file: c:\spice8\is\CML.DLL
Located file: c:\spice8\is\mech.dll
Located file: c:\spice8\is\REAL.DLL
Located file: c:\spice8\is\SOIMOS.dll
Located file: c:\spice8\is\switch.dll
Located file: c:\spice8\is\vsrc.dll

Error and Status Window

Simulation Control Window

Output Window

IsSpice4 functions are discussed in detail in the IsSpice4 Tutorial.

Simulation Status Window

Setting Preferences in IsSpice

We can now set preferences for the placement of these windows that monitor the simulator's results. (note: the waveform window is not the full "IntuScope" waveform viewing window covered later)

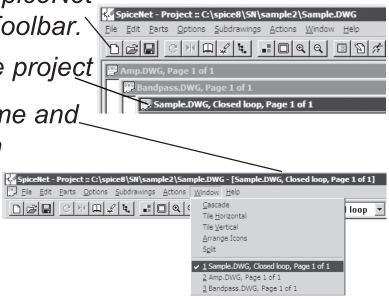
- **Stretch the IsSpice4 window to the desired size.**
- **From the Options Menu, select Auto Size Windows.**
- **Drag the Simulation Control window to allow you to see other windows.** (While holding down the left mouse key in the window banner, move the mouse to drag)

- **From the Options Menu, select Save Preferences.**

Next time a simulation is run, the IsSpice4 Windows will open in this new position.

Changing Projects

SpiceNet allows you to have multiple projects opened concurrently, Therefore, for instance, you can change simulation options on a desired project, then simulate without having to close other projects. Switch between different projects by clicking on the schematic.


SpiceNet - Project :: C:\spice8\SN\sample2\Sample.DWG
File Edit Parts Options Subdrawings Actions Window Help
Amp.DWG, Page 1 of 1
Bandpass.DWG, Page 1 of 1
Sample.DWG, Closed loop, Page 1 of 1
SpiceNet - Project :: C:\spice8\SN\sample2\Sample.DWG - [Sample.DWG, Closed loop, Page 1 of 1]
File Edit Parts Options Subdrawings Actions Window Help
loop
Cascade
Tile Horizontal
Tile Vertical
Arrange Icons
Split
✓ 1 Sample.DWG, Closed loop, Page 1 of 1
2 Amp.DWG, Page 1 of 1
3 Bandpass.DWG, Page 1 of 1

This is the SpiceNet Toolbar.

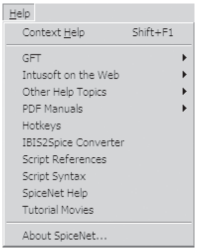
Active project

Project name and configuration

For example, if you are tiling or cascading schematics, simply click on the one you want to become your Active Project.

If you maximize a schematic window, the currently selected (active) schematic will show in the title bar in brackets. Several schematic manipulation functions are in the Windows drop down menu.

Using SpiceNet Help



If you get into trouble during SpiceNet, you can access the help files from the Help drop down menu. You can also quit SpiceNet without saving changes.

- **Select the Help menu, SpiceNet Help (<Ctrl>+H)**
- **Or, Select the File menu, Exit / Quit**

If this is your first time using SpiceNet, you can also watch the tutorial movies available from the SpiceNet Help menu.

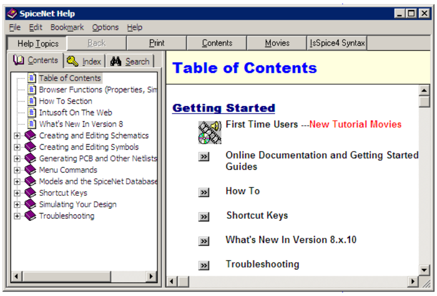


Table of Contents

Getting Started

First Time Users ---New Tutorial Movies

Online Documentation and Getting Started Guides

How To

Shortcut Keys

What's New In Version 8.x.10

Troubleshooting

SpiceNet Mouse and Keyboard Functions

The following two tables summarize the most common SpiceNet mouse functions and keyboard commands. Select Hotkeys from the Help Menu for a complete keyboard command list.

The following tutorials will familiarize you with the mouse and keyboard functions.

Mouse Usage

Desired Action	**Mouse Action**
Place a pointer box (determines where the next part will be placed)	Click the left mouse button.
Pan the schematic	Press and hold the left mouse button until the pan cursor appears, then drag the mouse.
Select an object	Hold the left mouse button over an object until the object is highlighted.
Create a selection region	Press and hold the left mouse button, dragging the mouse over the desired region.
Select/Deselect Objects	Shift-click on the object to select/deselect.
Move an object	Press and hold the left mouse button over an object until the object is highlighted. Move the mouse. Release the mouse button to place the object.
Add Part Values or edit part parameters	Double-click on a part, then type the value.
Change Node Numbers	Select a part that is connected to the node, then click the right mouse button, and select the “Node Names...” function.
Zoom In and Out	Click the right mouse button, then select the desired zoom level.

Keyboard Command Usage

Common Actions	**Key To Press**
Open Parts Browser	Press X, then select the part, or select Find
Place Passive Parts	Press the IsSpice4 keyletter (R, L, C, etc.)
Wiring Tool	Press W
Bus Tool	Press 2, then press W
Ground	Press O (the letter “o”, not the number zero)
Test Points (.PRINT)	Press Y (voltage), Press 2, then press Y (differential voltage), Press 3, then press Y (current), Press 4, then press Y (power) Press 6, then press Y (current subcircuit)
Rotate an object	Press + (plus key) after selecting the part(s)
Flip an object	Press - (minus key) after selecting the part(s)
Copy or Delete an Object(s)	Select the object(s), then press (<Ctrl>+C) or Del key
Fit View To Screen	Press F6 function key
Zoom In/Out	Press F7/F8 function key
Undo a task	Press Ctrl Z keys

Panning the Schematic

The Pan Cursor

There are two ways to move or pan the schematic.

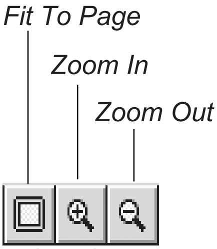
- **Point at an empty area of the schematic. Press and hold the left mouse button until the cursor changes to the pan cursor, which looks like a four-way arrow.**
- **Move the mouse while holding the left mouse button down.**
- **Release the button when you are finished panning.**
- **Or, press and hold the Shift key and press any arrow key (up, down, left or right on the keypad or regular numeric pad).**

When the mouse arrow changes to a 4-way arrow, you will be able to drag the schematic around the window. When the mouse arrow comes to the edge of the window, the schematic window will begin to continuously pan. Note: if the cursor is near a component when you press the button, you might select the component instead of the schematic.

- **Press F6 (Fit to Page) to view the entire schematic.**

The panning motion will not alter the symbol positions relative to the page borders or the rulers.

Zooming In, Out, and on a Portion of the Schematic



icon bar

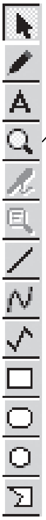
The F6, F7, and F8 function keys can also be used to change the zoom level.

SpiceNet enables you to view different sections of the schematic by changing the current zoom level. Try any of these methods:

- **Right click any place on the Sample.DWG schematic.**
- **From the Pop-up Menu, Choose "Zoom In (2 x)."**
- **From the Options Zoom menu, Choose "Zoom In (2x)."**

The schematic image is now shown at 2X resolution.

- **Use the F6 keyboard function key to "Fit to Page," F7 to "Zoom In," and F8 to "Zoom Out." Note: "Zoom In" and "Zoom Out" are done with respect to the blinking insertion point.**
- **Or, select the "Zoom Function" from the right side Tool Palette.**



Tool Palette

The mouse cursor arrow becomes a crosshair symbol.

- **Point the mouse, then click and hold the left mouse button and drag a bounding box around the desired zoom area.**
- **When the area has been selected, release the mouse button. Then, zoom in on the selected area. When you've finished zooming, press the Escape key to return to normal cursor mode.**
- **Close the drawing and do not save changes.**

Tutorial #2 - SpiceNet Basics

In this tutorial you will learn how to place, delete and select parts or symbols on a schematic.

Placing Parts

Now that you know how to manipulate a schematic, try placing some parts. This can be done in one of four ways: from the parts menu, using hotkeys, part browser, or Parts Bin Window.

Click in an empty area of the schematic. When a part is placed, SpiceNet will place the part where the square box is blinking. Then, the box will automatically move so it will be ready for the next part.

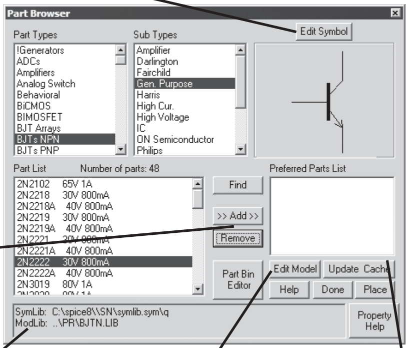
Place passive parts from the Parts menu or use the hotkeys. For active, behavioral and code models, use the part browser. Note that the Preferred submenu will expand to show a user-preferred part list, any recently placed parts, and any parts added to the Preferred Parts Bin. This Preferred submenu can also be accessed from the top of the right mouse-click pop up menu.

Use SpiceNet hot keys to place parts quickly.

Parts Browser icon

To edit an existing symbol click on the "Edit Symbol" button.

Parts are organized by Part Types, Sub Types, then Part List.


Part Browser

Part Types: !Generators, ADCs, Amplifiers, Analog Switch, Behavioral, BiCMOS, BIMOSFET, BJT Arrays, BJTs NPN, BJTs PNP

Sub Types: Amplifier, Darlington, Fairchild, Gen. Purpose, Harris, High Cur., High Voltage, IC, ON Semiconductor, Philips

Part List   Number of parts: 48

```
2N2102   65V 1A
2N2218   30V 800mA
2N2218A   40V 800mA
2N2219   30V 800mA
2N2219A   40V 800mA
2N2221   30V 800mA
2N2221A   40V 800mA
2N2222   30V 800mA
2N2222A   40V 800mA
2N3019   80V 1A
```

Preferred Parts List

SymLib: C:\spice8\\SN\symlib.sym\q
ModLib: ..\PR\BJTN.LIB

Use the "Add" and "Remove" buttons to generate your Preferred Parts List.

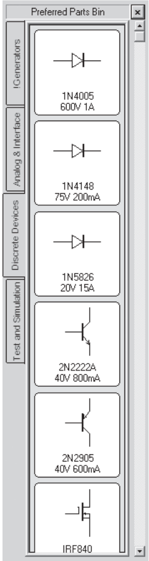
Where the symbol and part netlist is coming from.

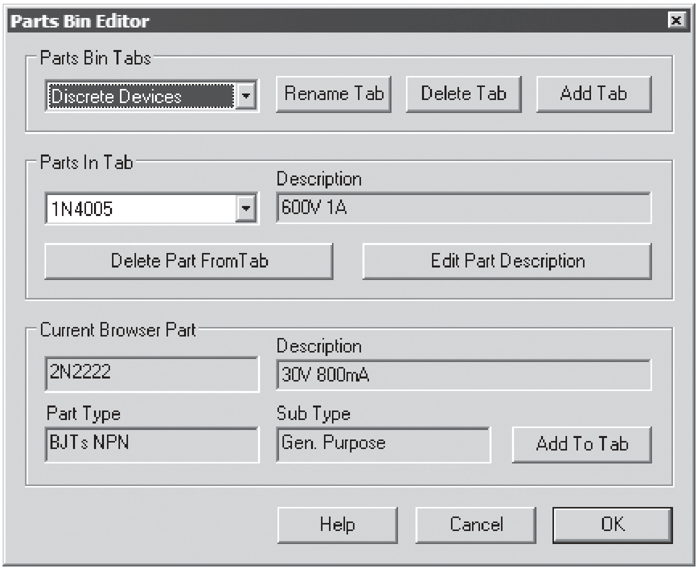
To view or edit the part's Spice model, simply click on the "Edit Model" button.

If you place a part and subsequently edit its library file or symbol, you must use the "Update Cache" button to use the new model definition and symbol.

- **From the Parts menu, select Parts Browser or type "X".**
- **Or, click on the Parts Browser icon.**
- **Use Find to do a search of all text fields for specific text.**
- **Choose a part. Click the Place button to place selected part on the schematic.**
- **Parts placed from the Parts Browser will be added to the clipboard. Press (<Ctrl>+V) to place the same part again.**

In the Part Browser dialog, click on the Parts Bin Editor button. This brings up the editor with the currently selected browser part available for adding. At the top dropdown, select your Parts Bin Tab, and in the middle dropdown you will see all the parts in that selected tab (Parts In Tab). If you want to add the current browser part into the selected tab, then click on the "Add To Tab" button. The current browser part will be added to the Parts in Tab dropdown with the Part Browser Description. If you want to modify the description shown in the Parts Bin Window, then click on the "Edit Part Description" button. Finally, click the "Ok" button to keep all changes. This entire dialog just affects the tabbed side panel that you can bring up for quick direct part placement. You will see a graphical representation of the part, part name, and description of the part you just added.





Parts Bin Tabs: Discrete Devices

Parts In Tab: 1N4005
Description: 600V 1A

Current Browser Part: 2N2222
Description: 30V 800mA
Part Type: BJTs NPN
Sub Type: Gen. Purpose

Deleting Parts

You can delete a part from the schematic.

- **Highlight a part and use the delete key, (<Ctrl>+X) or backspace to delete.**
- **Use (<Ctrl>+Z) to undo.**

ICAP/4 can perform unlimited undo (<Ctrl>+Z), and redo (<Shift>+<Ctrl>+Z) to operations.

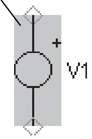
- **On the keyboard, hold down (<Ctrl>), press (Z) to delete the most recently placed part, or hold the (Z) down to delete all parts.**

- **(<Shift>+<Ctrl>+Z) will redo the string of removed parts.**

- **Or, from the Edit Menu, choose Redo or Undo.**

Selecting Parts or Symbols

Highlighted Bounding Box

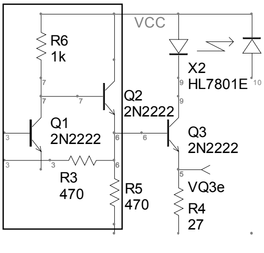


In order to copy, delete, rotate, flip, label or move a symbol, let's review how to select a part or symbol.

- **To select a single part, point at the center of the symbol, then press the left mouse button and briefly hold it until the part is highlighted.**

- **To select more than one part symbol, press and hold the Shift key. Point at a symbol and click the left mouse button. Then point at a different symbol and click the mouse button.**

Dragging A Bounding Box

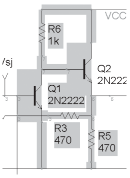


- **To select a group of symbols, point the mouse arrow outside the group of symbols. In one motion, press the left mouse button and move the mouse.**

- **As the mouse is moved, a bounding box is created. Release the mouse button to complete the selection. All chosen components will be highlighted.**

- **To deselect the symbols, click the left mouse button in a vacant area.**

Only Parts Completely Inside Bounding Box Will Be Selected

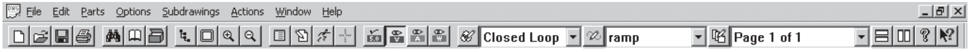


Review: To select a pan cursor (four arrows), hold down the left mouse button and try moving the mouse slowly. Also, you can use the F6 function key to view the whole schematic on the screen.

Customizing the SpiceNet Toolbar

Many of the functions used for editing a schematic can be placed on the SpiceNet toolbar to speed up things like deleting, copying, pasting, moving, rotating or flipping.

The SpiceNet toolbar shipped with the software should look like this:



Customize the toolbar to fit your preferences.

- **From the Options menu, select Customize Toolbar. As long as this Customize dialog is up, you can edit the toolbar. The icons are organized into Categories. Click on a button to see its description at the bottom of the dialog.**
- **Drag the desired icon from the Customize dialog to the toolbar, or drag an icon off the toolbar.**
- **Close the Customize dialog when done.**
- **From the Options menu, choose Save Preferences.**

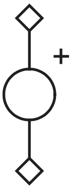
If you forget to save the changes, the program will remind you when you close SpiceNet.

Editing a Schematic

A symbol or a group of symbols must be selected for the copy, paste, delete, move, rotate and flip functions to work.

- **From the Sample.DWG schematic, select the Voltage Source symbol (extreme left) by pointing at the middle of the symbol and holding down the left mouse button.**
- **From the Edit Menu select Copy, then click in a vacant spot on the schematic.**

Voltage Source symbol



- **From the Edit Menu, select Paste.**

- **Or, from the keyboard, press (<Ctrl>+C) to copy, and (<Ctrl>+V) to paste in a vacant spot on the schematic.**

Both menu and keyboard commands for copy (<Ctrl>+C) and cut (<Ctrl>+X) store the selected part on the clipboard for later insertion. However, the menu clear command and the delete key on the keyboard does not store the selected object.

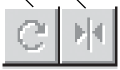
To Move a Component Symbol:

- **Highlight a desired symbol, click and drag the symbol to a preferred area on the drawing, then release the mouse button to place the symbol.**

To move a group of symbols:

- **Click on several symbols while pressing the Shift key. Then, release the Shift key.**

Rotate icon
Flip icon



- **Place the mouse arrow within one of the symbols' bounding boxes, then hold down the left mouse button.**

- **Drag the selected group. Release the mouse button to place the symbols.**

To rotate or flip a symbol or group of symbols:

Rotate or flip multiple selected parts as a group about their collective center.

- **Select a symbol or group of symbols. Press the rotate and flip buttons on the menu bar. Alternatively you can press the + key or the - key depending on the desired action.**

The + key rotates a symbol or group of symbols 90 degrees clockwise; the - key flips the symbol or symbols.

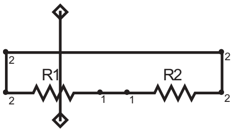
- **Close the schematic without saving.**

Wiring and Connecting Parts

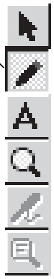
Right mouse click without moving to end wire mode.

Right mouse click to draw wire at any angle.

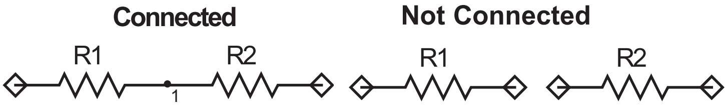
Press <Alt> key to toggle rubber banding. Rubberbanding is the ability to move parts while having the attached wires stretch and remain connected.



Wire button



Parts are connected by joining two pins together, or by connecting two pins with a wire. If the diamonds do not disappear, the parts are NOT connected.



When two parts are connected, the pins disappear and are replaced with a node number, as shown above.

To connect two parts by joining the pins:

- **Place two resistors on the schematic. Press “R,” then click somewhere else on the schematic and press “R” again.**

- **Left mouse click on the middle of one resistor. Hold down the mouse and drag the resistor by moving it close to the other until the diamonds overlap and become a dot.**

To connect two parts with a wire:

- **From the previous step, highlight one resistor, then drag it off the other resistor.**

- **Invoke the wire tool by pressing the letter “W,” or choose the wire button in the tool palette in the upper right side of the scroll bar.**

- **Point the wire pencil at the left pin on the resistor. Press and hold the left mouse button to define the start of the wire, then move the pencil toward a pin of the other resistor.**

- **Release the mouse button to end the wiring process.**

Note: A wire can be drawn at any angle by holding down the <Ctrl> key, holding down the left mouse button and moving the wire pencil.

Connecting with Buses

Buses are wires used to connect groups of wires or pins, to other groups of wires or pins. They appear as thick wires on the schematic.

To draw a bus at any angle, hold down the <Ctrl> key while drawing the bus.

Note: Labels for non-orthogonal buses (or wires) can not be displayed on the schematic.

To draw a bus:

- **From the Parts menu select Bus, or press the 2 and W keys.**
- **Point to the desired starting point, press and hold down the left mouse button, and draw a vertical bus that is about 1” long.**
- **Release the mouse button to end the segment.**

Bus Rules

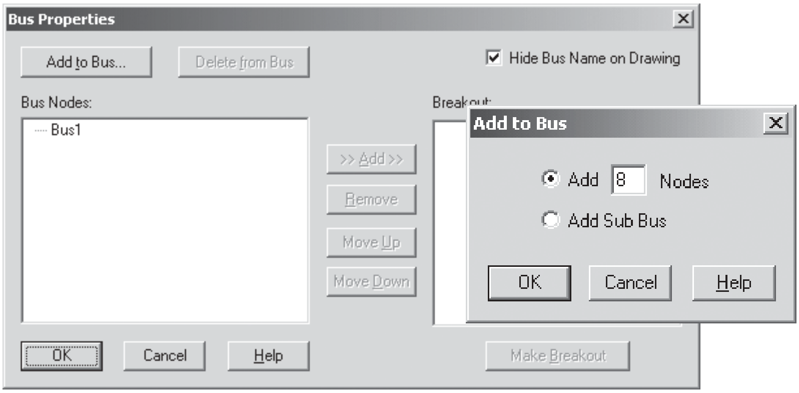
Buses may pass through wires, parts, pins or breakouts from other buses without making a connection. Buses may pass through other buses without making a connection when the end of each bus does not touch. If the ends touch, a connection is made as long as the buses are compatible.

Buses may be drawn orthogonally or at any angle. Although buses may not be stretched or re-sized, they may be placed end-to-end in order to create a continuous bus. In such a case, the two buses join to become one continuous bus, and both will have the same name.

Non-orthogonal buses, which are placed end-to-end, will join to form a continuous bus. This is also the case when an orthogonal bus is placed end-to-end with a non-orthogonal bus.

- **Double left-click on the Bus you’ve just drawn.**

The bus properties dialog opens. This enables the user to add nodes (breakouts), or sub-buses to the bus. Let’s add an 8-pin breakout to the bus.



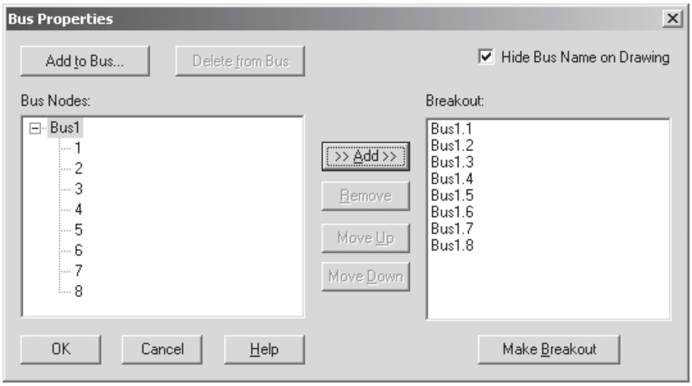
☑ Hide Bus Name on Drawing

Bus Nodes: Bus1

● Add 8 Nodes

○ Add Sub Bus

- **Click on the “Add to Bus...” button and enter a value of “8” in the Add field, then click the OK button.**



☑ Hide Bus Name on Drawing

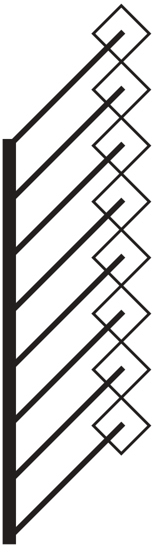
Bus Nodes:

Bus1
1
2
3
4
5
6
7
8

Breakout:

Bus1.1
Bus1.2
Bus1.3
Bus1.4
Bus1.5
Bus1.6
Bus1.7
Bus1.8

Bus with an 8-pin breakout.



- **Click on the “Bus1” name in the Bus Properties dialog.**

- **Click on the “>>Add>>” button, then click on the “Make Breakout” button.**

The 8-pin breakout has been added to the bus.

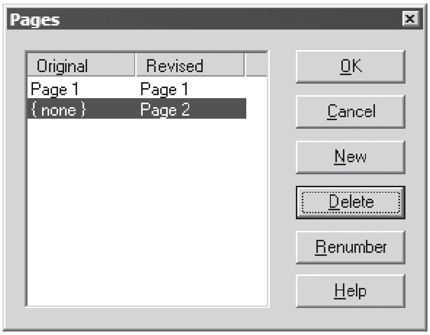
- **The Bus window then closes.**

- **Press the Delete key to remove the highlighted buses.**

Adding and Deleting Multiple Pages of Schematics

SpiceNet can edit multiple schematics and multiple pages of a single schematic at the same time. You can cut and paste between different schematics, and use previous designs in your current drawing. Dividing one schematic into multiple pages is not recommended, unless the schematic is very large.

To create one or more additional pages for a design:

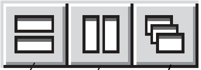


Original	Revised
Page 1	Page 1
{ none }	Page 2

- **From the Options menu choose Pages/Edit. A dialog shown on the left will pop up.**
- **Click on the New button, then OK to add a new page**
- **Or, you could have clicked the Delete button to delete a page.**

SpiceNet will automatically assign a name, e.g., “Page 2.” If you later decide to switch page numbers around, then click on the Renumber button. To activate a desired page in SpiceNet, the page must be highlighted.

Schematics can be tiled in three ways:



horizontally

vertically

or with cascading windows.

To copy and paste or navigate between multiple schematics:

- **From the File menu, choose Open, then navigate to a new schematic.**
- **From the Windows menu, choose Cascade, Tile Horizontal, or Tile Vertical.**
- **Or, select the Tiling option you want from the icons on the Toolbar** (horizontal or vertical rectangles)**.**

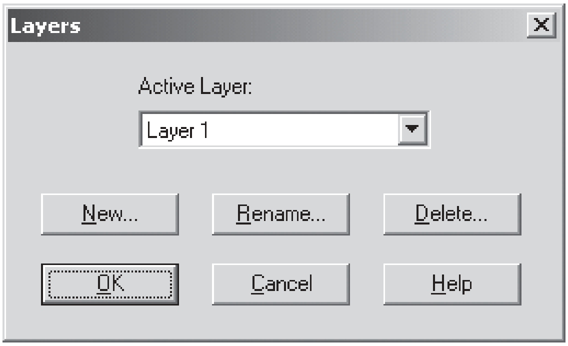
Working with Multiple Layers

SpiceNet is a layered (US Gov't. Patent No. US 6,230,305 B1) hierarchical schematic system, enabling different parts of the circuit to be placed on different drawing layers. Artwork and notes can be added to any layer. All layers are easily wired together since all layers are visible at the top level. The simulation is run on the culminated netlist.

:\spice8\SN\sample2\Sample.DWG is an example of a schematic using multiple layers flattened together into different configurations. For example, one can measure the Open and Closed Loop characteristics of a part by choosing which test circuitry (configuration) respectively employs that part.

To add, rename or delete a layer:

- **From the Options menu, choose Layers. To activate a layer, open the active layer pulldown menu.**



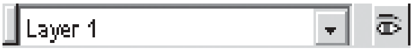
Layers

Active Layer: Layer 1

- **To add a new layer, select New. A name is automatically generated.**
- **To rename a layer, type the new name in the box, then click OK.**
- **To delete a layer, click the Delete button and OK.**

Note: You will be reminded that this deletion of layers is permanent and can't be reversed by the undo command.

To see one of the two layers in the current Sample.DWG, below is a closed loop configuration.


Layer 1

- **Hold the left mouse button over the eye in the lower left hand corner of the schematic. The circuitry placed in layer 1 is now highlighted.**

- **Open the drop down list to the left of the eye to see all layers in this schematic. Scroll to layer 4.**

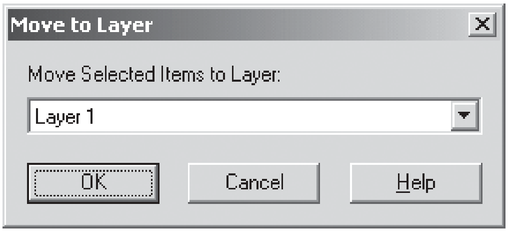
- **Press and hold the eye in the lower left hand corner of the schematic. The section of the total design placed in layer 4 is now highlighted.**

Select all the parts you want to move to another layer, right mouse click, select Move Item to Layer and choose a specific layer.

To move parts from one layer to another:

- **Select the parts individually or in a group by holding down the Shift key while selecting them.**

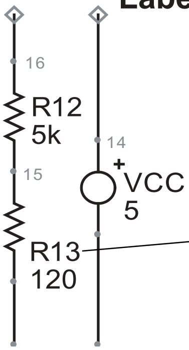
- **From the Edit menu, select Move to Layer**


Move to Layer

Move Selected Items to Layer:

Layer 1

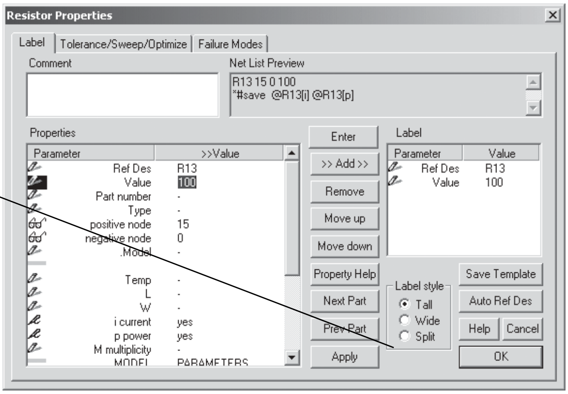
Labeling Parts and Changing Values



Attributes such as value, node number, passed parameters and reference designations for each symbol are accessed through the Part Properties dialog, or the NetList editor.

- **Double-click on the resistor R13 in Sample.DWG, type “120” into the Value field. Scroll down the parameter list to see all of the available model parameters.**

- **Click on the Next Part or Prev Part button.**



The Split Label Style allows you to position each part of the label separately.

Resistor Properties

Label | Tolerance/Sweep/Optimize | Failure Modes

Comment

Net List Preview

```
R13 15 0 100
*#save  @R13[i] @R13[p]
```

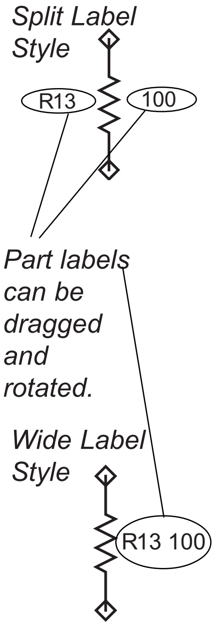
Properties

Parameter	>>Value
Ref Des	R13
Value	100
Part number	-
Type	-
positive node	15
negative node	0
.Model	-
Temp	-
L	-
W	-
i current	yes
p power	yes
M multiplicity	-
MODEL	PARAMETERS

Label

Parameter	Value
Ref Des	R13
Value	100

Label style: ● Tall ○ Wide ○ Split



Split Label Style

Part labels can be dragged and rotated.

Wide Label Style

Each part symbol in the schematic is highlighted as the button is pressed. This feature allows you to enter all of the part values and place all of the labels without leaving the Properties dialog. You may have to move the Properties dialog, depending on your screen size, in order to see the highlighted part.

- **Return to R13 in the box toward the right side of the dialog.**

- **Click on Split label style button just below.**

- **Or, Click on the “Wide” Label style button to display the label in “Wide” mode.**

- **Click on the OK button to close the properties dialog for the resistor. Notice that the value is now 120.**

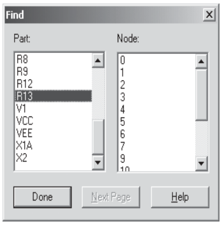
- **Click on the R13 label on the schematic and use the “+” key to rotate the label 90 degrees clockwise.**

You can move a label independently of its associated part, but if you move the part then all labels will move with it. If you lose track of which label belongs to its part, just double-click on the part and the labels belonging to that part will be highlighted. Alternatively, you can double-click on a label to highlight which part it belongs to.

Finding Parts and Editing Values Using NetList Editor

To quickly locate parts and nodes in your schematic, use the find feature.

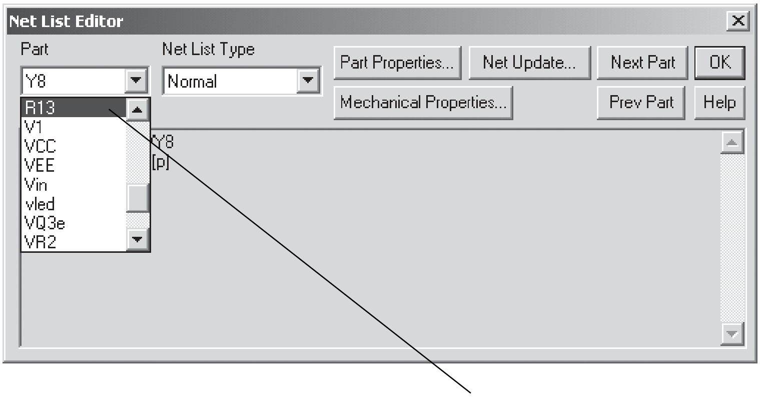
- **Press (<Ctrl> + F) to bring a Find dialog to view. This contains all the parts and nodes of the design. The node or part will be highlighted in the schematic while it is selected in the Find dialog.**



<Ctrl>+F for find parts dialog.

- **Double click on the highlighted part on the schematic after the Find dialog is closed. The Part Properties dialog will come to view.**

- **To find the part using the NetList Editor, select the Actions Menu, then NetList Edit function.**



- **Select the desired part from the Part list in the upper left of the dialog, or click the Next Part or Prev Part button to navigate to the desired part. The part will be automatically highlighted.**

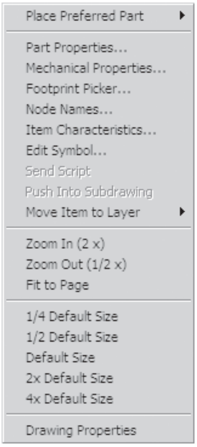
- **Click on the "Part Properties..." button to bring up the Properties dialog for the part that you wish to edit.**

Note: When the Properties dialog is dismissed, the NetList Editor dialog remains open to permit subsequent operations.

- **Click on the OK button to close the NetList Editor dialog.**

Renaming Nodes

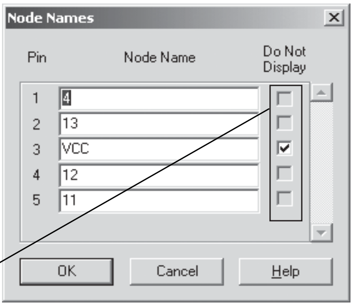
Pop-up menu



Node Names are automatically assigned by SpiceNet as they are created. These names may be changed at any time.

Rename any nodes as follows:

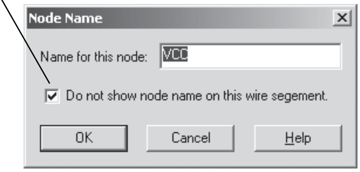
- **Highlight the desired parts and right click.**
- **Select Node Names in the pop-up menu.**



Pin	Node Name	Do Not Display
1	4	□
2	13	□
3	VCC	☑
4	12	□
5	11	□

- **Select a node name and change it.**
- **Then click on the OK button.**

Note: The display check boxes are only enabled for node names that are alphanumeric strings or integers of more than 3 digits.



Name for this node: VCC

☑ Do not show node name on this wire segement.

- **Or, double-click on a node and enter the name in the Node Name dialog.**
- **Then click on the OK button.**

Node names that are integers of 3 digits or less, e.g., 1, 22 and 333. They are not movable schematic objects and are always displayed.

Alternatively, node names changed to alpha or alphanumeric strings, or integers with more than 3 digits (examples: P1, V123, VCC, VOUT and 4444) will be displayed and are movable on the schematic. A check box allows for non display of these names.

IsSpice4 Control Statements

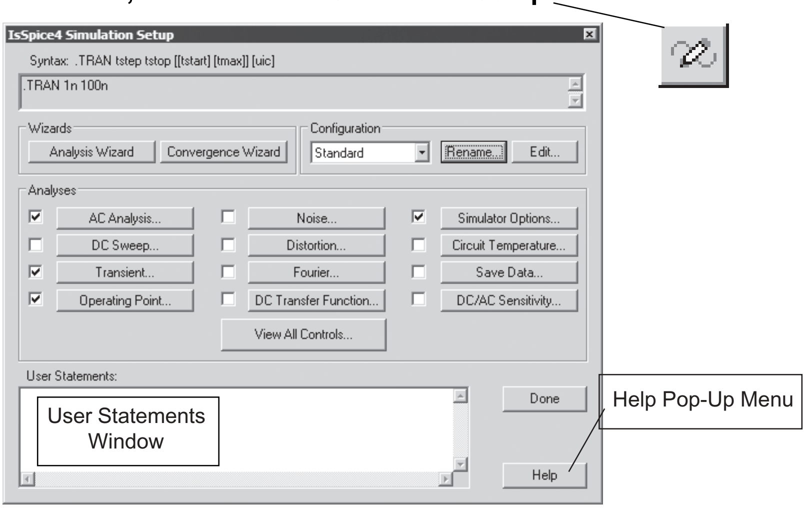
This is the preferred method, as opposed to direct netlist editing, for adding statements to your netlist.

The control statement for SPICE simulation setup is now examined. Importantly, the Analysis Wizard in the Simulation Setup dialog will walk you through all available analyses and their required parameters. You may also add or modify a control statement by clicking on the button with the analysis name (see below).

The User Statements window within the IsSpice4 Simulation Setup dialog is used to **optionally** add IsSpice4 control statements and ICL commands. Any line that is entered into this area of the dialog will be inserted into the input Netlist.

Click on the Help button for IsSpice4 syntax and analysis control statement information.

- **Select Simulation Setup/"Edit..." from the Actions menu, or click on the Simulation Setup icon.**



IsSpice4 Simulation Setup

Syntax: .TRAN tstep tstop [[tstart] [tmax]] [uic]

.TRAN 1n 100n

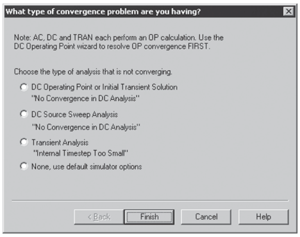
Configuration: Standard

Analyses:

☑ AC Analysis...
☐ Noise...
☑ Simulator Options...
☐ DC Sweep...
☐ Distortion...
☐ Circuit Temperature...
☑ Transient...
☐ Fourier...
☐ Save Data...
☑ Operating Point...
☐ DC Transfer Function...
☐ DC/AC Sensitivity...

The Convergence Wizard helps you resolve simulation convergence problems. See the on-line Help for more information.

There is a Convergence Wizard that will help you to resolve simulation convergence problems, if they occur.



What type of convergence problem are you having?

Note: AC, DC and TRAN each perform an OP calculation. Use the DC Operating Point wizard to resolve OP convergence FIRST.

Choose the type of analysis that is not converging.

○ DC Operating Point or Initial Transient Solution
"No Convergence in DC Analysis"
○ DC Source Sweep Analysis
"No Convergence in DC Analysis"
○ Transient Analysis
"Internal Timestep Too Small"
○ None, use default simulator options

- **Click on the Convergence Wizard button to view the dialog.**
- **Then, click on the Finish button.**

Entering Multiple Sets of IsSpice4 Control Statements

SpiceNet has the ability to save multiple sets of IsSpice4 control statements in different simulation setups (*Configurations*).

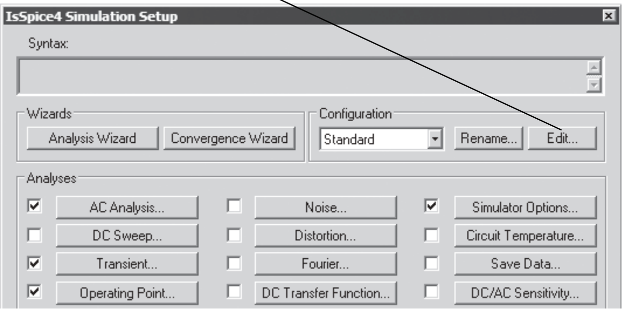
The default simulation setup configuration is “Setup1.”

A simulation setup configuration consists of a specific set of IsSpice4 control statements. Any combination of IsSpice4 control statements can be included into each configuration.

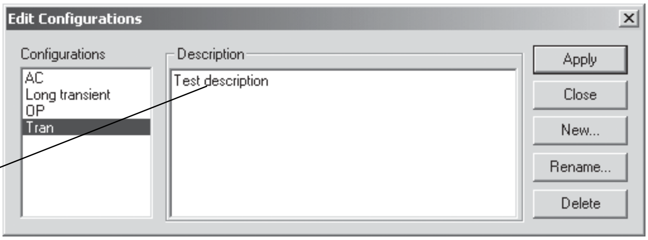
Different simulation setup configurations can be used to drive different simulation scenarios. For example, you might have one configuration for a “Monte Carlo” analysis, and another for a Transient analysis.

The configuration methodology alleviates the need for you to edit IsSpice4 control statements for different analyses or test cases.

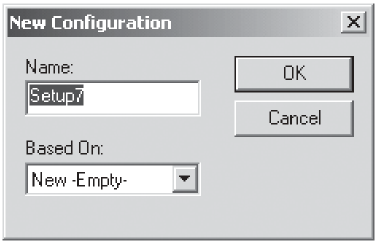
- **Click on the Edit button beside the Configuration drop down list box.**



- **Then click on the New button.**



You can type a test description here.



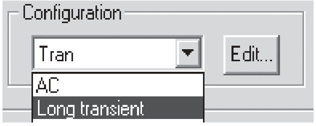
You may enter a name or simply accept the suggested name.

- **Click on the OK button to accept the suggested name.**

- **Click on the Apply button to close the Edit Configurations dialog.**

The IsSpice4 Simulation Setup dialog for the Setup4 configuration is now displayed. It has not yet been configured with a .AC or .TRAN statement. We won't configure it for the purpose of our exercise.

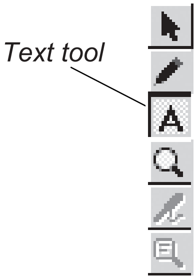
Once multiple simulation configurations have been defined, you may select the desired one from the drop down list box in the IsSpice4 Simulation Setup dialog, atop in SpiceNet's toolbar.



The selected simulation setup configuration will become active and will be used when the next simulation is performed.

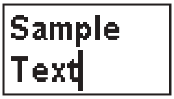
- **Click on the Done button to close the IsSpice4 Simulation Setup dialog.**

Inserting Text in the Schematic



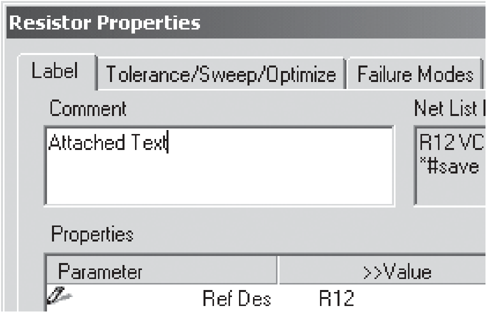
Text objects may be inserted directly on the schematic either from the Text Object in the Parts menu, or using the Text tool in the Tool Palette. Bring a schematic to view.

- **Left click on the Text tool in the right SpiceNet toolbar.**
- **Place the mouse cursor in a free space on the schematic and click. A text box will appear.**
- **Type the words "Sample Text" in the box.**
- **Highlight the text.**
- **Press the Delete key to delete the text.**



Sample Text

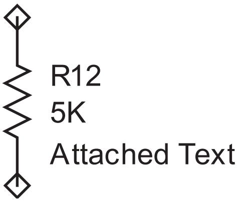
Note: By entering text into the Comment box in the Resistor Properties dialog, text is attached to the symbol and moved with the symbol.



Comment: Attached Text

Net List I: R12 VC
*#save

Parameter	>>Value
Ref Des	R12

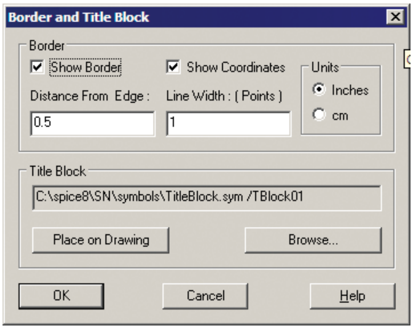


- **Press the Esc key to change the cursor back to normal mode or click any place in the schematic.**
- **OR Select ↖ in the Tool Palette to return to the normal mode.**
- **Press (<Ctrl >+Z) to undo the previous actions.**

Displaying a Border on a Schematic

A page border and coordinate can be displayed on a schematic to easily identify the position of each component on the schematic.

- **From the Options menu, select Drawing, Border and Title Block. The Border and Title Block dialog will pop up.**
- **Check the Show Border and Show Coordinates boxes, enter the desired Distance From Edge and Line Width values.**



Border and Title Block

Border

☑ Show Border ☑ Show Coordinates

Distance From Edge : 0.5

Line Width : (Points) 1

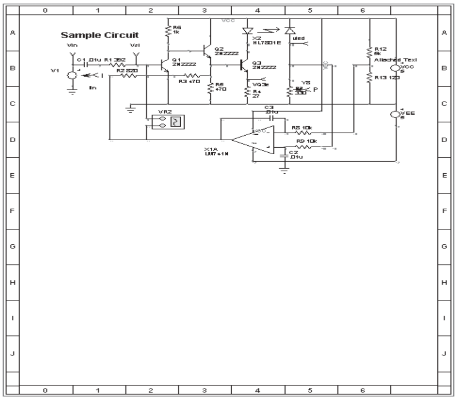
Units: ● Inches ○ cm

Title Block

C:\spice8\SN\symbols\TitleBlock.sym /TBlock01

- **Click on the OK button to dismiss the dialog.**

A page border with coordinates now appears on the schematic.

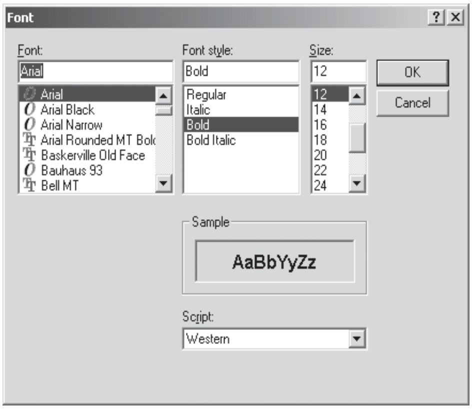


- **Press the F8 function key to zoom out and view the entire page.**

Adjusting Text Font, Alignment and Schematic’s Color

To modify text fonts:

- **Select the “Sample Text” string on the schematic.**
- **From the Options menu, select the font.**
- **Select a desired font style and click on the OK button.**



Font

Font: Arial

Font style: Bold

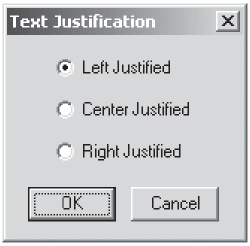
Size: 12

Sample: AaBbYyZz

Script: Western

To modify text alignment:

- **From the Options menu, select Artwork, Text Justification, and make a selection.**



Text Justification

● Left Justified

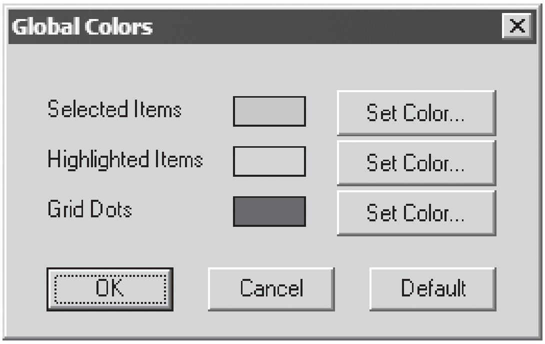
○ Center Justified

○ Right Justified

Changing Color Settings on the Schematic

To change the color of grid dots or highlighted items to be selected on the schematic:

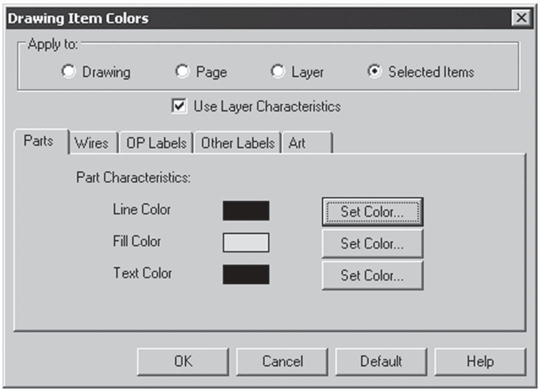
- **Choose Options menu, Global Colors.**



To change the color of parts, wires, labels and art:

- **Choose Options menu, Drawing Item Colors**.

For information about the Global Colors or Drawing Item Colors dialog, press the Help button in the lower right-hand corner.



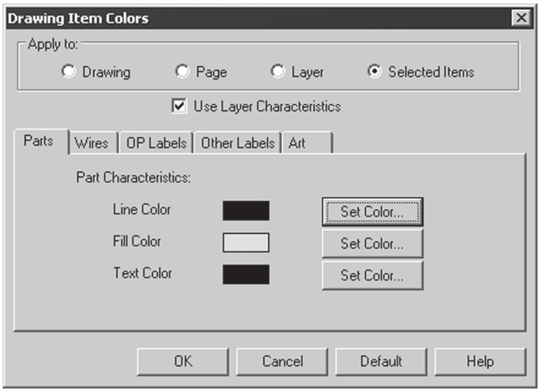
○ Drawing ○ Page ○ Layer ● Selected Items

☑ Use Layer Characteristics

Working With Parent Colors

Objects on a schematic inherit their colors in the form of a hierarchy: Drawing, Page, Layer and Item.

- **From the Options menu, choose Drawing Item Colors and a dialog will pop up as shown.**



Drawing Item Colors

Apply to: ○ Drawing ○ Page ○ Layer ● Selected Items

☑ Use Layer Characteristics

The “Use Parent Characteristics” check box causes the object characteristics to reset to those of the parent.

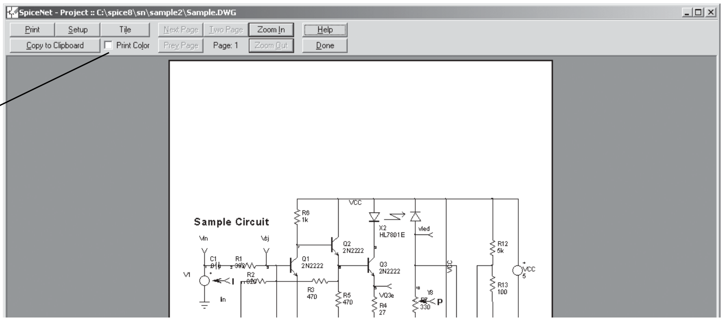
If the “Apply to Layer” button atop is selected and the “Use Page Characteristics” box is checked, then the items that are selected by the current tab on the current layer will have their characteristics set to those defined for the entire page. If “Apply to Page” is selected and the “Use Drawing Characteristics” box is checked, then the items that are selected by the current tab on the current page will have their characteristics set to those defined for the entire drawing.

Checking the “Use Parent Characteristics” check box will not directly affect changes that may have been made previously at a lower level in the schematic hierarchy. For example, specifying the layer to use the page’s characteristics will not override characteristics that have been modified for a specific part or wire on that layer. However, when the OK button is pressed, and if changes have been made at a lower level, a dialog will query the user if those changes should also be discarded.

Printing Your Schematic

The Print function under the File menu invokes the schematic preview mode. The options at the top of the screen can be used to adjust the size of the schematic on the preview page.

- **From the File menu, select the Print function.**



Note: Check the Print Color box when you print filled objects. This box affects printing, but not viewing.

If the page you are printing to is smaller than the drawing page size, then multiple pages will be printed.

- **Click on the Copy to Clipboard button to place an image on the clipboard for pasting into another application.**
- **Hold the SHIFT key and click on Copy to Clipboard to choose the type of MetaFile to place on the clipboard or save the image as an EMF (Enhanced MetaFile).**

The Setup button will change the paper size, source, orientation and margins. Click either print or setup to change the printer.

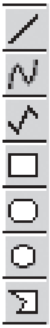
Note: If you plan to reduce the schematic image for placement in a publication, then you will want to choose Minimum Printing Line Width from the Options > Drawing Menu. Enter a larger number to increase the line width. In this way, no lines will disappear when the image is shrunk. This setting will only affect the printed image and the image copied to the clipboard from print preview. You can copy and paste a small section of your circuitry directly from SpiceNet to another application, such as Microsoft Word, but Minimum Printing Line Width does not apply.

Drawing Artwork Shapes on the Schematic

SpiceNet provides drawing tools to construct common artwork shapes on the schematic. Shapes include line, bezier, polyline, rectangle, round rectangle, ellipse, and polygon. They are located directly above the vertical scrolling bar.

Holding down the <Shift> key to constrain artwork shapes has the following affect:

- Lines are constrained to angles of 0, 45, 90, 135, 180 and 225 degrees. This is applicable for lines, poly-lines, polygons, and base lines for Bezier curves.
- Rectangles, rounded rectangles, and ellipses are initially constrained to squares, rounded squares, and circles respectively, in both the main window and the Symbol Editor. Re-sizing after an initial draw constrains the artwork to the last drawn ratio. For instance, if a rectangle is drawn with length twice the height while re-sizing by holding down shift key, the rectangle will keep this ratio and not become a square.



line

bezier

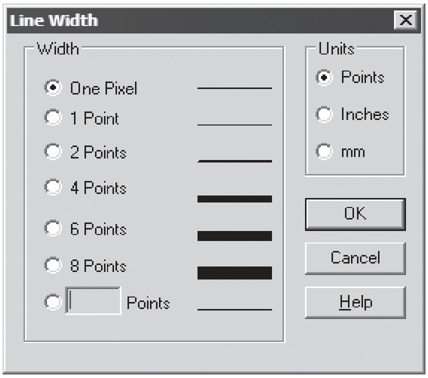
polyline

rectangle

round rect

ellipse

polygon



Line Width

Width

● One Pixel

○ 1 Point

○ 2 Points

○ 4 Points

○ 6 Points

○ 8 Points

○ Points

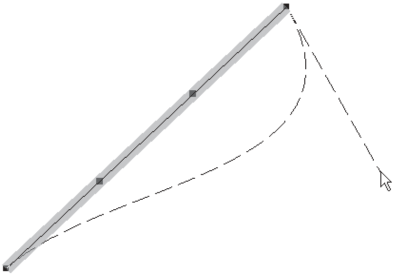
Units

● Points

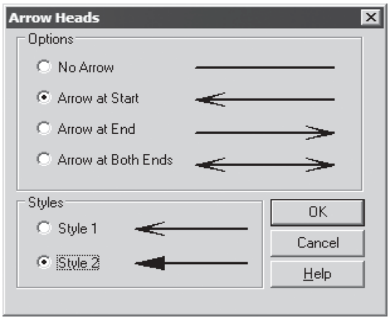
○ Inches

○ mm

Control the line width of your artwork shape by selecting Line Width... from the Options, Artwork menu. You can increase the width up till 19 points. If no artwork is selected, this becomes the default width. If an artwork is selected, then only that artwork's line width will be changed.



Select the "Bezier line" icon. Then, hold the left mouse button on the schematic to drag a line, then release the mouse. Left click the mouse once, away from the shape. Next, hold down the left mouse button on top of the Bezier line to highlight it, then release the left mouse button. Finally, hold the left button and drag any of the points on the Bezier line to form a curve.



Arrow Heads

Options

○ No Arrow

● Arrow at Start

○ Arrow at End

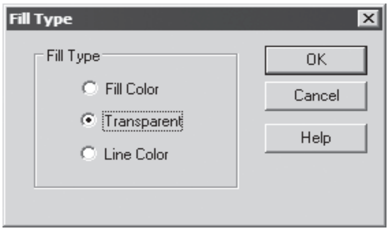
○ Arrow at Both Ends

Styles

○ Style 1

● Style 2

Select Arrow Heads... from the Options, Artwork menu to add an arrow head(s) to your lines. Select an existing line to give it an arrow head(s), or none to set the default. This only applies to line artwork, not poly-lines or bezier curves.



Fill Type

Fill Type

○ Fill Color

● Transparent

○ Line Color

Select Fill Type... from the Options, Artwork menu to select how to fill in your artwork shape. Colors are chosen in the Drawing Item Colors... dialog's artwork tab.

If you place multiple artwork on top of each other, you can use "Move to Front" or "Move to Back" in the Edit menu to arrange the artwork. This only affects the order of artwork on that layer. If you have multiple layers and you place the artwork on the first layer, then any circuitry drawn on latter layers will appear on top of the artwork. Artwork can be moved to any layer desired depending on the effect you want. You can also rotate and flip artwork like any other object.

You can group selected artwork and free text labels into a single symbols. Select the artwork and text labels you want to combine, then choose Selected Artwork to Symbol... from the Edit menu.

Quitting the Tutorial

- **From the File menu, select Exit/Quit to end the tutorial.**

- **Do not save changes.**

If you want to save Sample.DWG, from the File menu select "Save As" and use a different file name. This step is recommended because you will use the original Sample.DWG in the later sections of this manual.

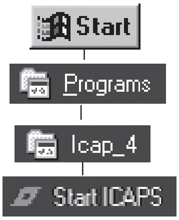
Tutorial #3 - Capturing a Schematic for Simulation

In this tutorial, we'll capture a schematic, perform a simulation and examine some waveforms at desired points in the design.

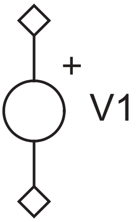
Placing Components

Let's launch SpiceNet with a new blank schematic:

- **Open a new schematic from the <File, New> menu on SpiceNet's top toolbar.**



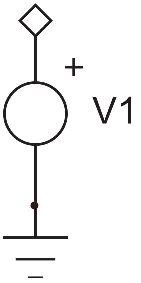
- **Click on the schematic where you want to place a component. An insertion box will appear.**
- **Type the letter V (it is not case sensitive) to draw a voltage source symbol, or select the Voltage Source from the Parts pulldown menu.**



V1 is the Spice Reference Description for the voltage source. This can be changed at any time.

- **Type the letter O (not the number zero) to draw a ground symbol.**

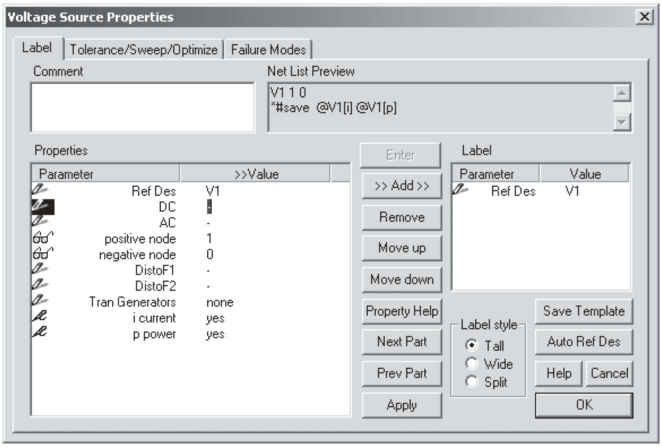
Note: Resistors, capacitors, inductors, etc. may be placed using hot keys, e.g. R, C, and L, or by using the Parts pulldown menus.



Adding Component Attributes and Electrical Stimulus to Design

Every component has its own set of values loaded into its Part Properties dialog.

- ***Double-click on the voltage source to bring up its properties dialog.***



```
V1 1 0
*#save  @V1[i] @V1[p]
```

Parameter	>>Value
Ref Des	V1
DC	
AC	-
positive node	1
negative node	0
DistoF1	-
DistoF2	-
Tran Generators	none
i current	yes
p power	yes

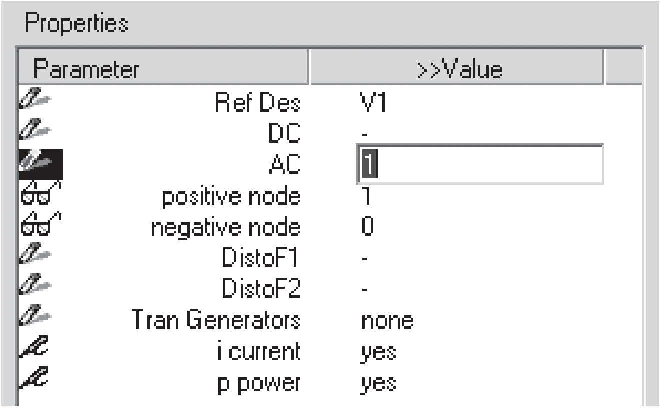
Parameter	Value
Ref Des	V1

Label style: ● Tall

To set up an AC analysis:

- **Click on the AC value field and enter 1.**

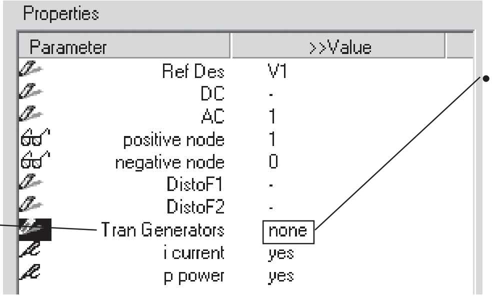
Note: *AC=1 means that AC has a magnitude of 1 and phase=0.*



Parameter	>>Value
Ref Des	V1
DC	-
AC	1
positive node	1
negative node	0
DistoF1	-
DistoF2	-
Tran Generators	none
i current	yes
p power	yes

To set up a transient (time-based) analysis:

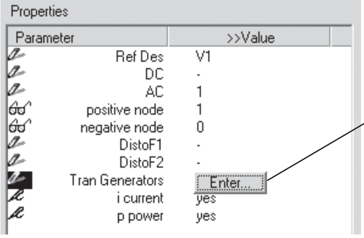
Tran Generators is used to configure Time Domain Stimuli.



Properties

Parameter	>>Value
Ref Des	V1
DC	-
AC	1
positive node	1
negative node	0
DistoF1	-
DistoF2	-
Tran Generators	none
i current	yes
p power	yes

- **Click on the Tran Generators value field.**

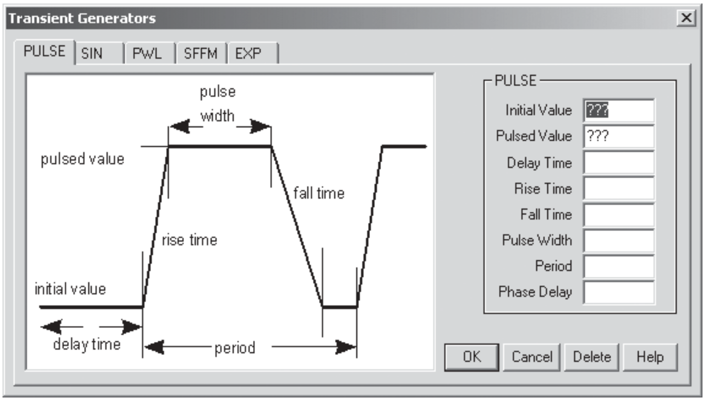


Properties

Parameter	>>Value
Ref Des	V1
DC	-
AC	1
positive node	1
negative node	0
DistoF1	-
DistoF2	-
Tran Generators	Enter...
i current	yes
p power	yes

- **Click the “Enter” button to bring up Transient Generators dialog.**

“ Pulse 0 1” generates a pulse stimulus in the Transient analysis.



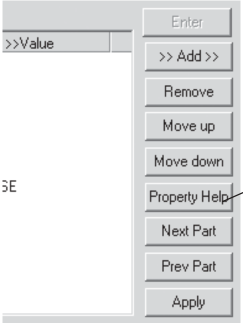
PULSE

Initial Value	???
Pulsed Value	???
Delay Time	
Rise Time	
Fall Time	
Pulse Width	
Period	
Phase Delay	

- **Select Pulse tab in the Transient Generators dialog.**
- **Enter “0” in the Initial Value field.**
- **Enter “1” in the Pulsed Value field.**
- **Click OK to close the window.**

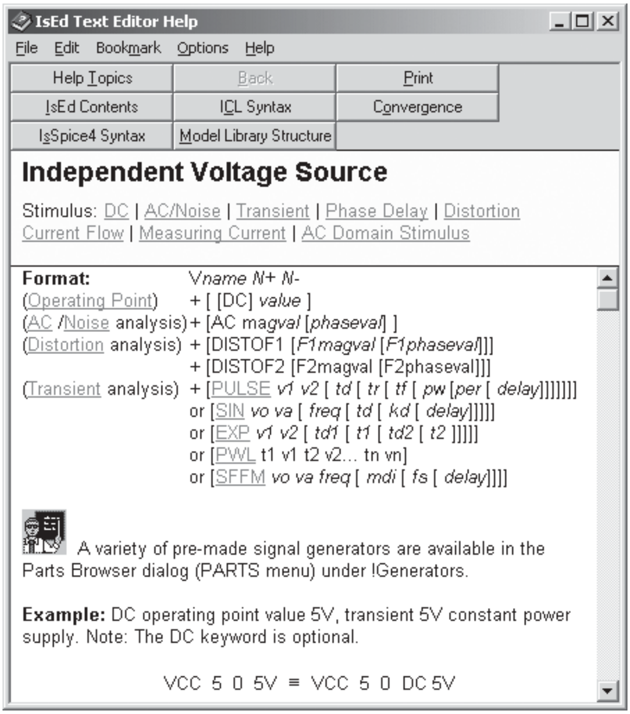
Note: The values we entered will appear in the NetList Preview field in the Properties dialog once the Transient Generator window is closed.

If you have any trouble with setting up voltage source properties, help files can always be launched. There are two ways to get help:



- **Click on the Property Help button in the Voltage Source Properties dialog.**

The help window is shown below. Different types of analyses are listed in the window. Click on any phrase for a full explanation of that particular topic.



Independent Voltage Source

Stimulus: DC | AC/Noise | Transient | Phase Delay | Distortion
Current Flow | Measuring Current | AC Domain Stimulus

```
Format:                  Vname N+ N-
(Operating Point)        + [ [DC] value ]
(AC /Noise analysis)     + [AC magval [phaseval] ]
(Distortion analysis)    + [DISTOF1 [F1magval [F1phaseval]]]
                         + [DISTOF2 [F2magval [F2phaseval]]]
(Transient analysis)     + [PULSE v1 v2 [ td [ tr [ tf [ pw [per [ delay]]]]]]]
                         or [SIN vo va [ freq [ td [ kd [ delay]]]]]
                         or [EXP v1 v2 [ td1 [ t1 [ td2 [ t2 ]]]]]
                         or [PWL t1 v1 t2 v2... tn vn]
                         or [SFFM vo va freq [ mdi [ fs [ delay]]]]
```

A variety of pre-made signal generators are available in the Parts Browser dialog (PARTS menu) under !Generators.

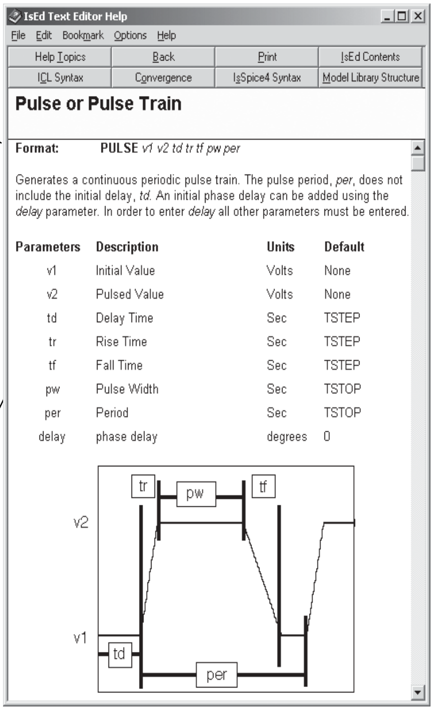
Example: DC operating point value 5V, transient 5V constant power supply. Note: The DC keyword is optional.

VCC 5 0 5V ≡ VCC 5 0 DC 5V

While editing the properties in the Transient Generator dialog, you may also get help without switching windows.

This field shows the IsSpice4 syntax for the PULSE statement.

All of the PULSE parameters are defined here.



Pulse or Pulse Train

Format: **PULSE** *v1 v2 td tr tf pw per*

Generates a continuous periodic pulse train. The pulse period, *per*, does not include the initial delay, *td*. An initial phase delay can be added using the *delay* parameter. In order to enter *delay* all other parameters must be entered.

Parameters	Description	Units	Default
v1	Initial Value	Volts	None
v2	Pulsed Value	Volts	None
td	Delay Time	Sec	TSTEP
tr	Rise Time	Sec	TSTEP
tf	Fall Time	Sec	TSTEP
pw	Pulse Width	Sec	TSTOP
per	Period	Sec	TSTOP
delay	phase delay	degrees	0

- **Click on the "Help" button on the bottom of the Transient Generators dialog to open the Help window seen on the left.**

The help information will be on the topic that displays on the tab selected (in this case, it is Pulse) in the Transient Generators dialog.

- **Click ✕ on the up-right corner to close the help window.**

- **Click OK button to exit the Voltage Source Properties dialog.**

Now that we've placed and defined the voltage source, let's capture the rest of a simple RLC circuit.

- **Point to the positive pin of the voltage source and click the left mouse button to place the insertion box.**

Use the "+" key to rotate parts 90 degrees clockwise.

- **Type "RLC+O" (the letters R, then L, then C, then the + key, then the letter O) to place a resistor, inductor, capacitor and ground symbol.**

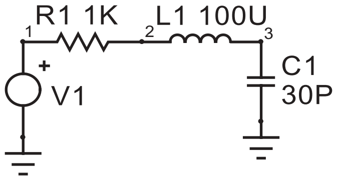
- **Double-click on the resistor and enter a value of "1K" in its Value field, then click on the Next Part button.**

- **Enter a value of "100U" in L1's Value field, then click on the Next Part button.**

- **Enter a value of "30P" in C1's Value field, then click on the OK button to dismiss the Capacitor Properties dialog.**

Node 0 is defined as system ground in all IsSpice4 netlists.

The schematic will look like this:



Save icon

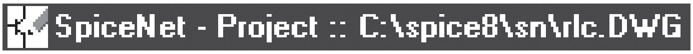
The Save icon is enabled when a change is made to the schematic.

To save the work you've done so far:

- **Select the Save function from the File menu, or click on the Save icon, then enter "RLC" in the "File name" field.**

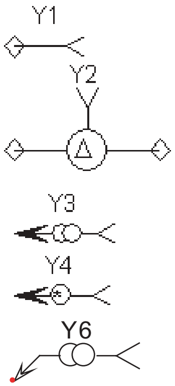
- **Click on the Save button.**

Notice that SpiceNet's title bar has changed:


SpiceNet - Project :: C:\spice8\sn\rlc.DWG

Test Point Symbols

Test Point Symbols



Test points are used to generate real-time waveforms (via VIEW and ALIAS statements). These statements cause the IsSpice4 simulator to calculate waveform data at the test point locations, and display it in the real-time waveform display. This will be covered a little later in this chapter.

There are five basic test point symbols available:
Y: node voltage (referenced to ground)
2Y: differential voltage
3Y: device current (placed within the body of a part)
4Y: device power (placed within the body of a part)
6Y: current subcircuit (placed on subcircuit pin)

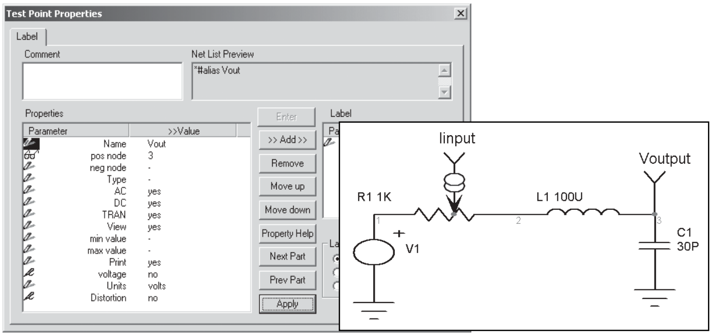
Note: When placing a test point on a part, a red dot indicates connectivity.

***Notes:** Power test points are not supported in AC analysis.*

Current and Power test points do not support subcircuit elements (reference designators that begin with an "X"). Use current subcircuit (6Y) to measure current going into subcircuit pin.

- **Click on node 3 (the output) and type the letter Y to place a voltage test point on that node.**
- **Double-click on the test point and enter "Voutput" in its Name field, then click on the OK button.**
- **Click on the body of R1 and input "3y+++" (the number 3, lower-case y, and 3"+"s) to place a current test point on R1.**
- **Double-click on the current test point and enter "Iinput" in its Name field, then click on the OK button.**

When the simulation is run, the waveform data for the test points will be calculated and available for viewing in IntuScope.


Test Point Properties

Comment:

Net List Preview: *#alias Vout

Properties

Parameter	>>Value
Name	Vout
pos node	3
neg node	-
Type	-
AC	yes
DC	yes
TRAN	yes
View	yes
min value	-
max value	-
Print	yes
voltage	no
Units	volts
Distortion	no

Automatic Test Point Ref Des Assignment

Test Point Reference Designation (Ref Des) is automatically assigned on initial placement based on the following rules:

a) When adding a voltage test point to a named node (i.e., a node with a name rather than a number), the test point's Ref Des is set to the name of the node, if the name starts with the letter "V." If not, it is set to the name, but with a "V" prefix.

b) If the node is unnamed, the Ref Des is set to "V<n>", where <n> is the node number.

c) When adding a differential voltage test point, its Ref Des is set to the names of its two nodes as defined in a) and b) above, and separated by an underscore character. If the second pin of the test point is not yet connected when the test point is placed, the underscore character is still appended to the name of the first node.

d) When adding a current or power test point to a part, the test point's Ref Des is set to a name that reflects the part for which it is connected (i.e., "WR1" for a power test point connected to resistor "R1," "IR1" for a current test point connected to R1, and "IX1_1" for a subcircuit current test point connected to pin 1 of X1)

e) If a test point is placed where it is not connected to anything, it is given a Ref Des consisting of the letter "Y," followed by a sequentially assigned number.

At any time, you can press the "Auto Ref Des" button in Test Point Properties dialog to rename any test point according to the above rules, based on its type and current connection(s).

Ref Des is retained

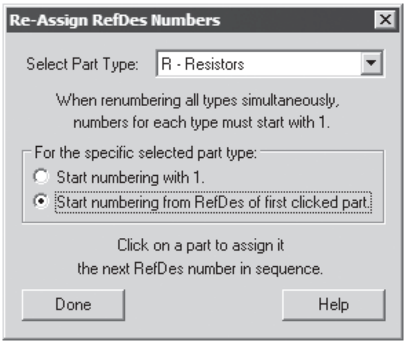
A part and test point’s Ref Des is retained when copying and pasting, as long as the prior Ref Des does not already exist on the target drawing. A Ref Des that has been modified to a non-standard form is also retained with one or more 'x' characters appended to its copied one, to distinguish it from the original. This allows you to copy and paste entire sections of circuitry onto another schematic drawing without losing their assigned part name and test point name.

Reassign Ref Des

At any time you can systematically renumber any or all of a schematic's part reference designations by selecting "Reassign RefDes..." from the Edit menu.

- If “Automatic” is selected then all Ref Des will be automatically renumbered.
- If “Manual” is selected then Re-Assign RefDes Numbers dialog will appear.
- If "All Types" is selected, you can begin sequentially renumbering each type of part starting from 1 (e.g., R1, C1, X1, etc.)
- If a specific part type is selected, you have the ability to start renumbering, based on the first part of that type that is clicked. For instance, if you clicked on R7, then the next resistor you selected will be renumbered as R8.

When a part is renumbered, it is highlighted so the user can remember which parts have been renumbered, and which ones haven’t. If an error is made, you can undo the renumbering by pressing (<Ctrl>+Z) or selecting undo from the Edit menu.



Re-Assign RefDes Numbers

Select Part Type: R - Resistors

When renumbering all types simultaneously, numbers for each type must start with 1.

For the specific selected part type:

○ Start numbering with 1.

● Start numbering from RefDes of first clicked part.

Click on a part to assign it the next RefDes number in sequence.

If you modify a part with a test point attached to it, remember to double click on the test point. Then select the “Auto Ref Des” button to update the test point Ref Des, based on the new part name.

This example RLC circuit shows you the ease of schematic capture. Your parts and test points are automatically assigned intelligent names. Part names are incremented after each placement, and test points are named based on their type and connection(s). You can re-use circuitry from other schematics, or copy, paste and flip mirror-image circuitry. If such copying and pasting of existing circuitry causes your circuitry names to appear messy, you can easily and systematically reassign them.

Entering IsSpice4 Control Statements

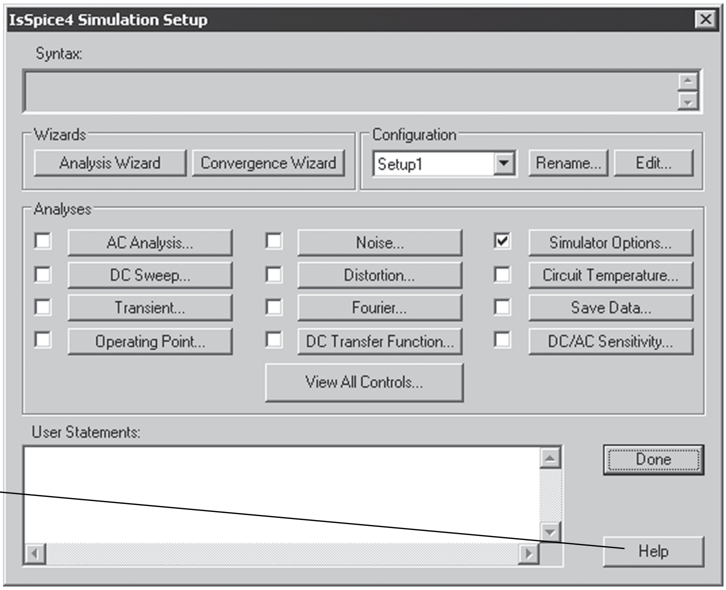
So far, we have taken care of all but one of the items necessary for generating an IsSpice4 simulation. We need to now tell Ispice4 what analyses to perform.

Simulation Setup icon

- **From the Actions Menu, select Simulation Setup, Edit, or click on the Simulation Setup icon.**

This dialog contains simulation control buttons.

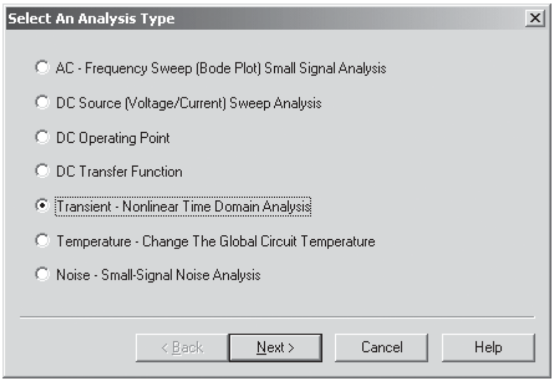
Click on the Help button for information on the analysis setup.



This dialog is used to generate the analysis statements, which will be automatically included in the IsSpice4 netlist and used to drive the simulator.

In addition to the in-application Help available in this dialog, there are also two powerful wizards that will help you generate simulation control information quickly and accurately. We introduced you to the Convergence Wizard in the SpiceNet Basics Tutorial. Let's take a look at the Analysis Wizard.

- **Click on the Analysis Wizard button.**



Select An Analysis Type

○ AC - Frequency Sweep (Bode Plot) Small Signal Analysis
○ DC Source (Voltage/Current) Sweep Analysis
○ DC Operating Point
○ DC Transfer Function
● Transient - Nonlinear Time Domain Analysis
○ Temperature - Change The Global Circuit Temperature
○ Noise - Small-Signal Noise Analysis

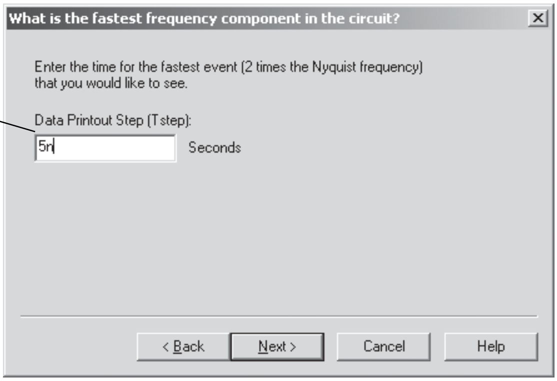
The Analysis Wizard automatically creates the simulation control information.

Now let's create the control statement automatically for a Transient (time domain) simulation for the RLC circuit.

- **Click on the "Transient - Nonlinear Time Domain Analysis" radio button and then click on the Next button.**

- **Enter a value of "5n" for the Data Printout Step (Tstep).**

Tstep must be small enough to resolve the highest frequencies in the circuit.



What is the fastest frequency component in the circuit?

Enter the time for the fastest event (2 times the Nyquist frequency) that you would like to see.

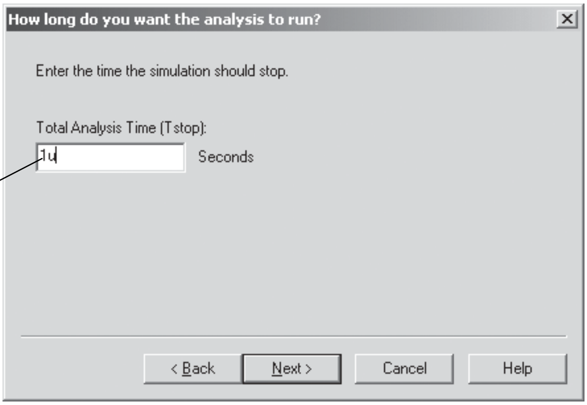
Data Printout Step (Tstep): 5n Seconds

Tstep is the time interval (sampling rate) for which interpolated data will be recorded and placed in the output file.

- **Then click on the Next button.**

- **Enter a value of “1u” for the Total Analysis Time (Tstop).**

- **Then click on the Next button.**



How long do you want the analysis to run?

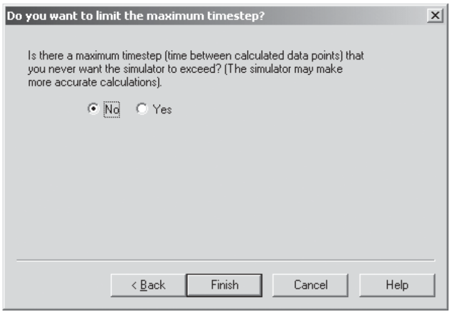
Enter the time the simulation should stop.

Total Analysis Time (Tstop): 1u Seconds

Tstop specifies the time length of the simulation.

- **Leave the “No” radio button checked. This is the default mode.**

- **Click on the “Finish” button to complete the operation and exit the Analysis Wizard.**



Do you want to limit the maximum timestep?

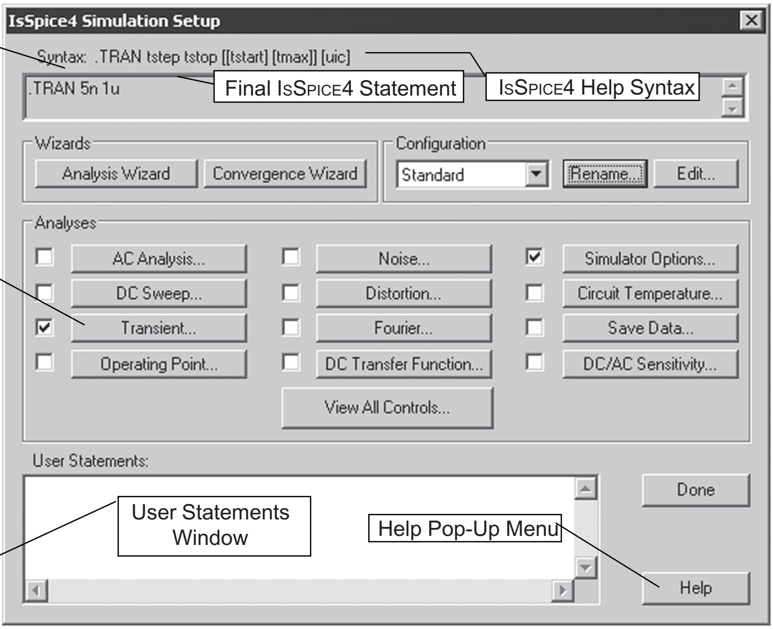
Is there a maximum timestep (time between calculated data points) that you never want the simulator to exceed? (The simulator may make more accurate calculations).

● No ○ Yes

The Analysis Wizard has just created the IsSpice4 Transient Analysis statement for you.

You can directly edit the Transient Analysis statement by clicking here.

You may enter any IsSPICE4 statements in this window.



IsSpice4 Simulation Setup

Syntax: .TRAN tstep tstop [[tstart] [tmax]] [uic]

.TRAN 5n 1u

Configuration: Standard

Analyses:

□ AC Analysis... □ Noise... ☑ Simulator Options...
□ DC Sweep... □ Distortion... □ Circuit Temperature...
☑ Transient... □ Fourier... □ Save Data...
□ Operating Point... □ DC Transfer Function... □ DC/AC Sensitivity...

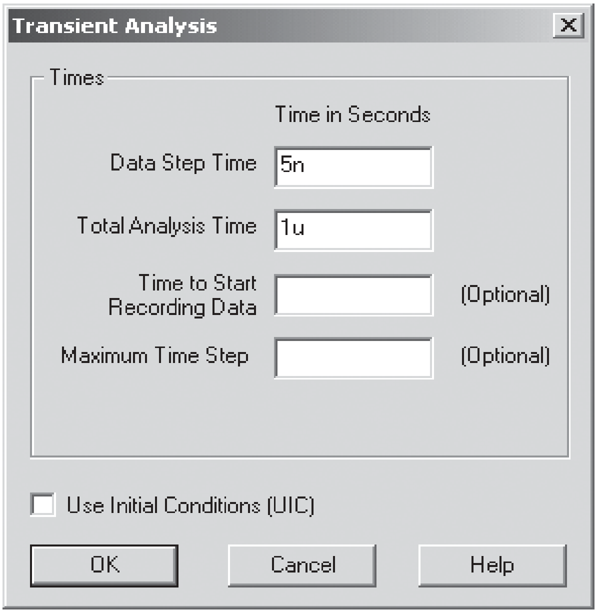
The ".TRAN 5n 1u" statement tells IsSpice4 to perform a time domain analysis when the simulation is launched. The simulation will begin at 0 seconds and end at 1 microsecond. Calculated data points will be written to the output file at 5 nanosecond intervals.

The Analysis Wizard has created the Transient statement for you, and has also checked the box beside the Transient button so that it will be performed when the simulation is launched.

As you become more familiar with the different types of simulation analyses, you may select the desired analysis function and enter the simulation parameters yourself.

- **Click on the "Transient..." button in the IsSpice4 Simulation Setup dialog.**

Click on the Help button for more information concerning the Transient analysis.



Data Step Time: 5n

Total Analysis Time: 1u

Time to Start Recording Data: (Optional)

Maximum Time Step: (Optional)

□ Use Initial Conditions (UIC)

Notice that the "Data Step Time" and "Total Analysis Time" fields already contain data (i.e., 5n and 1u respectively). The Analysis Wizard enters the information for you. If you want to change the simulation parameters for this analysis, you may change them here or revisit the Analysis Wizard.

Next, let's specify the simulation control information, which will be used to perform our AC analysis of the RLC circuit.

- **Click on the OK button to dismiss the dialog.**

We could use the Analysis Wizard to create the control information for the AC analysis, just as we did for the Transient analysis.

The AC analysis is a linear analysis that sweeps the circuit response through a range of frequencies.

See the on-line Help for more information concerning AC analysis.

Alternatively, let's enter the data directly in the AC analysis dialog within the IsSpice4 Simulation Setup dialog.

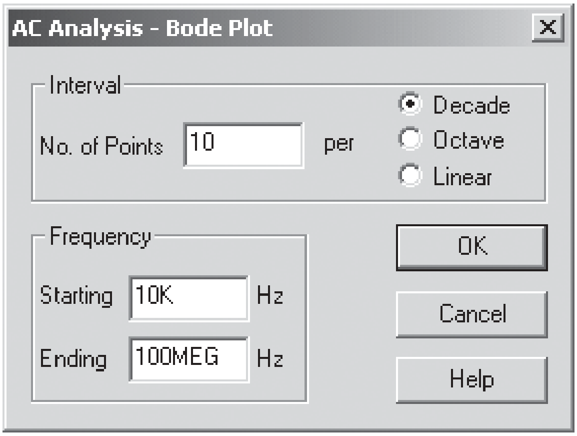
- **Click on the "AC Analysis..." button in the IsSpice4 Simulation Setup dialog.**

- **Enter "10" for the No. of Points.**

- **Click on the "Decade" button.**

- **Enter "10K" for the Starting Frequency value.**

- **Enter "100MEG" for the Ending Frequency value.**



AC Analysis - Bode Plot

Interval

No. of Points: 10 per ● Decade ○ Octave ○ Linear

Frequency

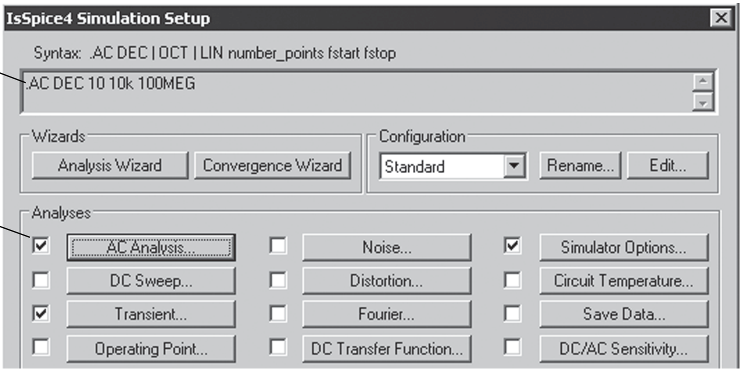
Starting: 10K Hz

Ending: 100MEG Hz

This is the IsSpice4 AC Analysis statement we just created.

The AC Analysis box has been checked (activated) automatically.

- **Click on the OK button to close the AC Analysis dialog.**



IsSpice4 Simulation Setup

Syntax: .AC DEC | OCT | LIN number_points fstart fstop

.AC DEC 10 10k 100MEG

Configuration: Standard

Analyses

☑ AC Analysis... □ Noise... ☑ Simulator Options...

□ DC Sweep... □ Distortion... □ Circuit Temperature...

☑ Transient... □ Fourier... □ Save Data...

□ Operating Point... □ DC Transfer Function... □ DC/AC Sensitivity...

The “AC DEC 10 10K 100MEG” statement defines an AC frequency sweep that collects 10 data points per decade. The starting frequency is 10KHz, and the ending frequency is 100MHz.

We’ve just finished specifying the second of two control statements that will be used in the simulation. We’re now ready to run both a Transient analysis and an AC analysis.

Notice that there are “check marks” in the check boxes beside the Transient and AC analysis buttons. This indicates that those analyses have been specified and activated.

You can activate and deactivate simulation options by checking or un-checking their boxes.

- **Click on the Done button to close the IsSpice4 Simulation Setup dialog.**

Running a Simulation

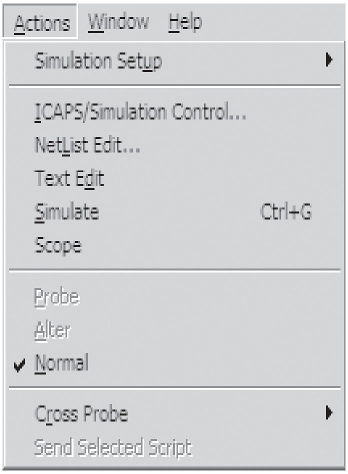
At this point, we have captured the RLC schematic and entered all of the required information.

When the simulation is performed, all applicable node voltage, device current and power dissipation data is saved by default. Device current and power measurements are accessed through the Device Properties dialog. Default power measurement is turned off for capacitors and inductors.

Now let's run the simulation.

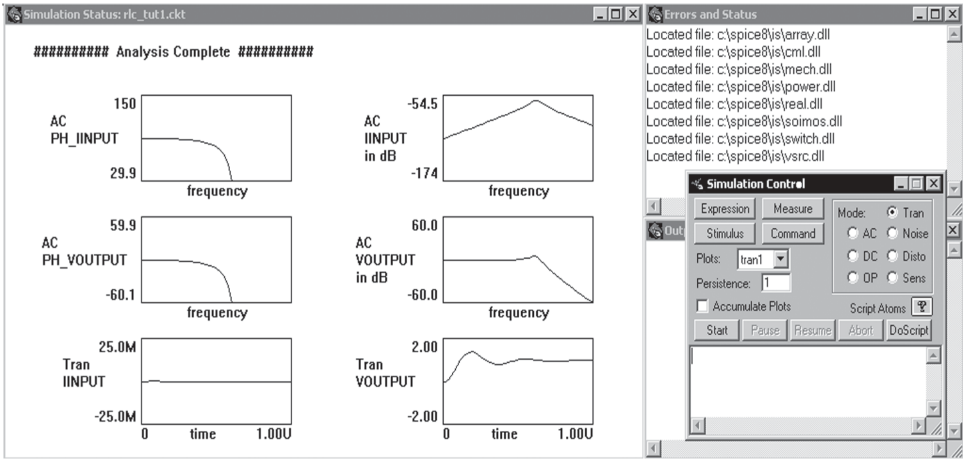
- **Select Simulate (<Ctrl>+G) from the Actions menu, or press the Simulate** (running man) **icon in the top toolbar.**

You can also click on the Run Simulation icon in the toolbar to launch the simulation.



The IsSpice4 simulation screen appears and the simulation is performed. The RLC circuit waveforms at the test points, i.e., Voutput and Iinput, are drawn in the real-time waveform display as they're calculated. You can see the waveforms for both the AC and Transient analyses.

Note: The real time waveform viewer provides a thumbnail sketch. Detailed waveform views are provided in IntuScope (covered later).

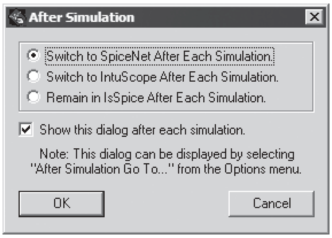


```
########## Analysis Complete ##########
```

```
Located file: c:\spice8\is\array.dll
Located file: c:\spice8\is\cml.dll
Located file: c:\spice8\is\mech.dll
Located file: c:\spice8\is\power.dll
Located file: c:\spice8\is\real.dll
Located file: c:\spice8\is\soimos.dll
Located file: c:\spice8\is\switch.dll
Located file: c:\spice8\is\vsrc.dll
```

The IsSpice4 simulator, and how to use it interactively, are covered in the next tutorial.

Important Note: Do NOT close the IsSpice4 simulation (e.g., don’t close the window out). It must remain open while we cross probe waveforms in the next section.



● Switch to SpiceNet After Each Simulation.
○ Switch to IntuScope After Each Simulation.
○ Remain in IsSpice After Each Simulation.

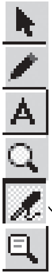
☑ Show this dialog after each simulation.

Note: This dialog can be displayed by selecting "After Simulation Go To..." from the Options menu.

In IsSpice4, select “After Simulation Go To...” in the Options menu. If desired, select another choice for the active window. Save preferences only saves window position and size.

An After Simulation dialog will be displayed after a simulation is run. It has three radio buttons providing the choices: "Switch to SpiceNet After Each Simulation," "Switch to IntuScope After Each Simulation" or "Remain in IsSpice After Each Simulation." There is also a check box for "Show This Dialog After Each Simulation." This allows you to specify which program immediately becomes active following a simulation, based on what you are trying to accomplish. If you do a lot of cross-probing, create simple curve families (e.g., alter function), or run scripts placed on your schematic, then select “Switch to SpiceNet After Each Simulation.” If you do a lot of waveform measurement, then select “Switch to IntuScope After Each Simulation.” If you want to quickly tweak any circuit parameter and see your real-time test point waveforms be instantly updated, then select “Remain in IsSpice After Each Simulation.”.

Cross Probing on the Schematic



ICAP/4 has the ability to interactively display node voltages, device currents and power dissipation directly on the schematic, and in the IntuScope waveform viewer. To display a waveform on the schematic, simply set the desired mode in the Actions/ Cross Probe menu, hold the SHIFT key and then click on the appropriate schematic node or device. Alternatively, right mouse click on the appropriate schematic node or device, hold the SHIFT key and then select the cross probe mode.

- **From the IsSpice4 window, select Actions menu, Schematic to switch back to the schematic so we can perform the cross probing.**

- **Select Probe in the Actions menu, or click on the Probe icon to activate the cross probing tool.**

The analysis type and voltage/ current or power setting are changed via Cross Probe in the Actions menu or the Cross Probe right mouse button menu.

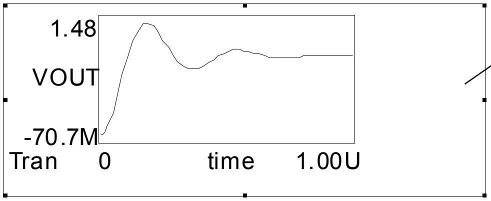
Cross Probing a Voltage

To view a voltage, simply hold down the SHIFT key and click on the desired node or test point.

Let’s display the voltage at the output of the circuit. *Notice that the mouse arrow has changed to a probe symbol.*

- **Click on the test point, “Voutput” with the probe tool while holding down the SHIFT key.**

The voltage waveform appears on the schematic.



Voutput test point voltage

The normal cursor allows you to move, re-size, and delete schematic waveforms. The Cross Probe cursor changes to the normal cursor when it is over a schematic waveform.

- **Move the Cross Probe cursor over the Vout schematic waveform. It will automatically change to the normal cursor. Drag and drop the Vout waveform out of the way to cross probe some more waveforms.**

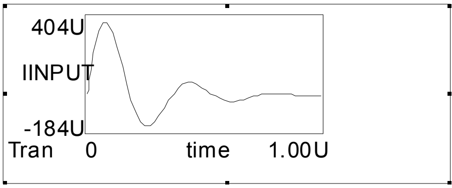
Cross Probing a Device Current

To view a device current, simply click on the device symbol while holding down the SHIFT key while in cross probe mode.

Let's display the current through resistor R1.

- **Click on the resistor, R1, with the cross probe tool while holding down the SHIFT key**

The current waveform appears on the schematic.



Current through R1

NOTE: It will stay in AC Voltage/Current Cross Probe mode until you change it.

Similarly, you may move the waveform out of the way and then cross-probe the current through L1 and C1.

As you recall, we ran both a Transient analysis and an AC analysis. The active analysis type, Tran, is shown in the Cross Probe menu item under the Actions menu.

To change Cross Probe mode and plot the schematic waveform at the same time, you can right mouse click on node 3, hold the SHIFT key down, and select AC Voltage/ Current.

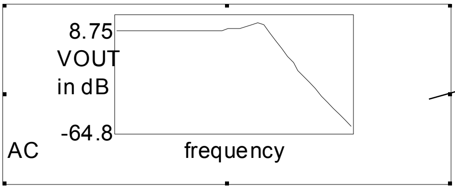
Cross Probing an AC node voltage

To view an AC voltage, simply set "Actions, Cross Probe" to AC mode, then click on the node with the probe tool while holding down the SHIFT key.

Let's change the probe analysis type to AC and take a look at the AC magnitude response at the output of the RLC circuit.

- **Select the AC mode in the Cross Probe function under the Actions menu.**
- **Move the R1 Current waveform out of the way, if necessary. Then click on node 3 with the cross probe tool while holding down the SHIFT key.**

The AC magnitude waveform for node 3 is displayed, as shown on the next page.



AC voltage at node 3, Voutput

__NOTE:__ Cross-probing does not support subcircuit elements (reference designators that begin with an "X"). To make subcircuit instance parameters available in IntuScope's Add Waveform dialog , go to IsSpice4's Simulation Setup dialog and check Save Data, Save all node voltages, Save all device currents and Save all power dissipations.

Cross Probing Device Power Dissipation

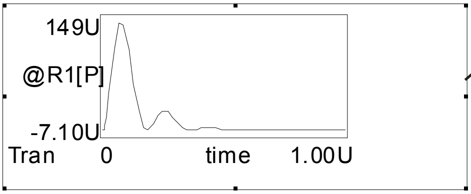
To view the power dissipation through a device, simply set the cross probe mode to Tran, and Power, in the "Actions, Cross Probe" menu.

Now let's display the power dissipation through R1.

- **In the Actions menu, select Cross Probe and the Tran and Power options.**

- **If necessary, move the V(3) waveform out of the way. Then click on the resistor, R1, with the probe tool while holding down the SHIFT key.**

The power through R1 is displayed.



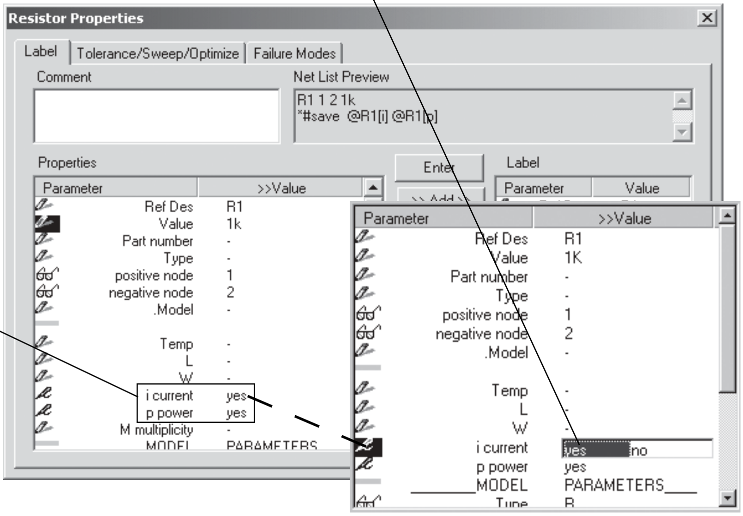
Power through R1

- **Press the Esc key or right mouse click and choose Normal Cursor to end Cross Probing.**

Power waveforms can only be viewed in the Transient analysis.

Resistor power waveforms are saved by default. To see if a part's current or power waveform will be saved, look in the parameter list in its Properties dialog, e.g., "i current" and "p power." A "yes" entry indicates that it will be saved, so that it is available for viewing. If a device's current or power option is turned off (e.g., the entry is "no") you may turn it on by clicking "no" to open the data field, then selecting the "yes" option.

Note: *Resistor power waveforms are saved by default. Look in the Part Properties dialog to see the saved device waveforms.*



```
R1 1 2 1k
*#save @R1[i] @R1[p]
```

Parameter	>>Value
Ref Des	R1
Value	1k
Part number	-
Type	-
positive node	1
negative node	2
.Model	-
Temp	-
L	-
W	-
i current	yes
p power	yes
M multiplicity	-
MODEL	PARAMETERS

Parameter	>>Value
Ref Des	R1
Value	1K
Part number	-
Type	-
positive node	1
negative node	2
.Model	-
Temp	-
L	-
W	-
i current	yes no
p power	yes
MODEL	PARAMETERS
Type	R

Cross Probing to IntuScope

IntuScope functions are discussed in Tutorial 4.

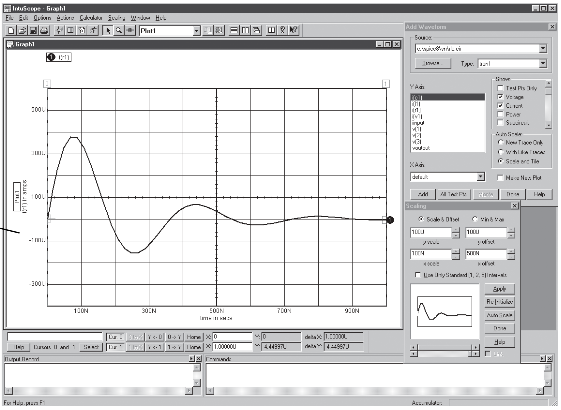
Just as we cross probed to display node voltage, device current and power dissipation waveforms on the schematic, we may also cross probe waveforms to an IntuScope graph. Cross probing to IntuScope is the same as cross probing to the schematic except you no longer hold down the SHIFT key while cross probing.

If IntuScope is not currently running it will automatically launch in the background when you try to cross probe to it.

- **Move the R1 power waveform out of the way, if necessary. Then click on R1 with the probe tool.**

This will send the R1 power waveform directly to IntuScope instead of plotting it on the schematic. If you right mouse click on R1 while in cross-probe mode, you can choose to send either the current or power waveform to IntuScope. Note that once you change to cross probe mode it will stay in that mode until you change it again.

- **Select "Scope" from the Actions menu or toolbar icons to view the waveform in IntuScope.**



Source: c:\spice8\sn\rlc.cir
Type: tran1
Show: ☐ Test Pts Only ☑ Voltage ☑ Current ☐ Power ☐ Subcircuit
Auto Scale: ○ New Trace Only ○ With Like Traces ● Scale and Tile
X Axis: default
☐ Make New Plot
● Scale & Offset ○ Min & Max
y scale: 100U
y offset: 100U
x scale: 100N
x offset: 500N
☐ Use Only Standard (1, 2, 5) Intervals
Cur. 0 X: 0 Y: 0 delta X: 1.00000U
Cur. 1 X: 1.00000U Y: -4.44997U delta Y: -4.44997U

This is the IntuScope graph window, which now displays the power through R1.

Next, let's display the AC waveform at the output, node 3. Since the Transient waveform has a linear time scale, and the AC waveform(s) will have a semi-log scale, we'll open a new waveform graph in IntuScope.

- **Select New Graph from the File Menu in IntuScope.**

- **Select Schematic from the Actions menu to go back to the schematic, or press its box at the bottom desktop.**

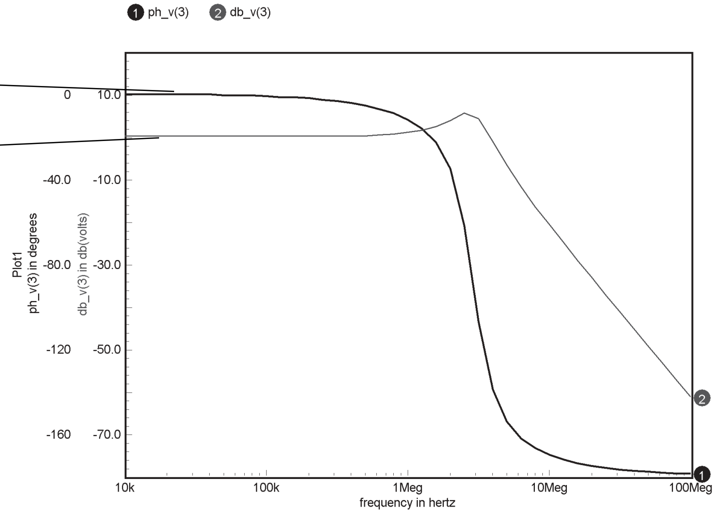
Do this in one step by right mouse click on node 3 with the cross probe tool and select AC Voltage/Current from the menu.

- **In the SpiceNet Actions/Cross Probe menu, select the AC and the Voltage/Current mode.**

- **Click on node 3 with the probe tool.**

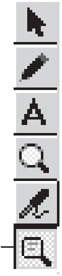
- **Select the "Scope" option from the Actions menu to view the waveform in IntuScope.**

AC phase and magnitude response of the RLC circuit.



Notice that the AC magnitude AND phase waveforms are displayed on the graph when you cross-probe a waveform during the AC analysis.

Running a Parametric Sweep



The Alter function allows you to create curve families quickly and easily.

The Alter function may also be used to change component values within hierarchical drawings.

Key circuit parameters can be varied directly from the schematic environment, without having to restart the IsSpice4 simulator.

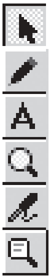
Once an initial simulation run has been performed, the Alter function enables you to automatically step a component value or its temperature, or design temperature, through a prescribed range, and automatically run a series of simulations. You can choose a linear or logarithmic sweep of the variation. You can select one or more design nodes or components to be plotted in IntuScope. The Alter function creates curve families quickly and easily.

In anticipation of the transient curve family that we'll generate as we sweep the value of R1, let's create a new waveform graph in IntuScope so the curve family will be easier to view.

- **Select New Graph from the File Menu in IntuScope.**

- **Select Schematic from IntuScope's Actions menu to go back to SpiceNet.**

Now we're ready to open the Alter dialog and set up the sweep. We'll sweep the value of resistor R1 from 1kW to 10kW, in 1kW increments, and display the voltage at node 3 (the output node) each time we change the value.



Normal cursor mode. You may also use this button to go back to normal mode.

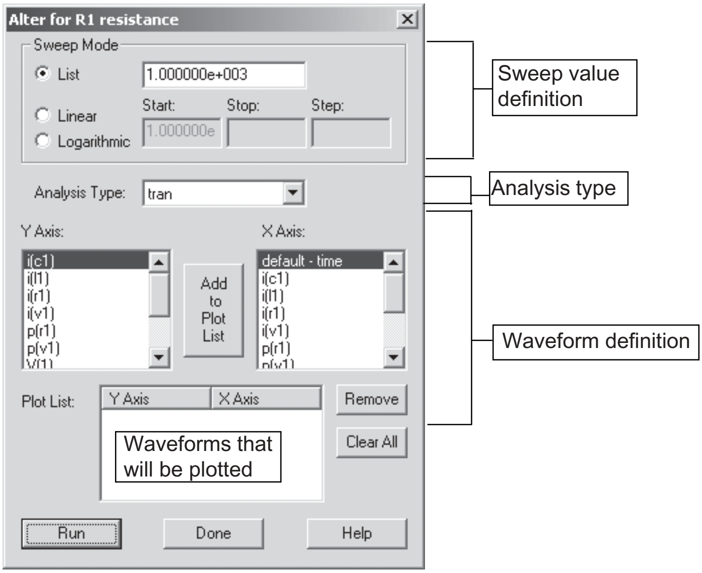
- **Press the Esc key or select the Normal icon from the right toolbar to exit cross probe mode. Now go back to normal cursor mode.**

- **Select Alter from the Actions menu, or click on the Alter icon on the right toolbar.**

Notice that the mouse arrow cursor has changed to Alter's magnifying glass symbol.

- **Click on resistor R1 with the Alter tool.**

The Alter dialog allows sweeps of component parameters.



Alter for R1 resistance

Sweep Mode: ● List: 1.000000e+003

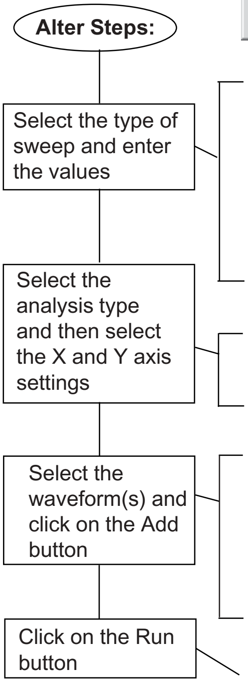
○ Linear ○ Logarithmic — Start: 1.000000e Stop: Step:

Analysis Type: tran

Y Axis: i(c1), i(l1), i(r1), i(v1), p(r1), p(v1), V(1)

X Axis: default - time, i(c1), i(l1), i(r1), i(v1), p(r1), p(v1)

Plot List: Y Axis | X Axis



Alter Steps:
Select the type of sweep and enter the values
Select the analysis type and then select the X and Y axis settings
Select the waveform(s) and click on the Add button
Click on the Run button

- **Click on the Linear button in the “Alter for R1 resistance” dialog to setup a linear sweep. (Warning: Never enter a value of zero, notably if the log button is selected)**

- **Enter a Start value of “1k,” a Stop value of “10k,” then a Step value of “1k.”**

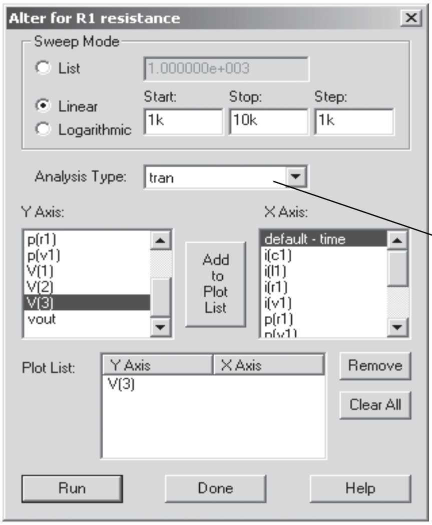
- **Make sure that the “default: time” option is selected under the X Axis field.**

- **Scroll down the list below the Y Axis field and select the “V(3)” entry in order to plot the RLC output at node 3.**

- **Click on the “Add to Plot List” button to add the “V(3)” entry to the Plot List at the bottom of the dialog.**

- **Click on the Run button to execute the simulation sweep. Dismiss the Alter dialog using the Done button.**

This will sweep the value of R1 from 1KΩ to 10KΩ in 1KΩ increments, and display a curve family in IntuScope.



Alter for R1 resistance

Sweep Mode

○ List 1.000000e+003

● Linear

○ Logarithmic

Start: 1k Stop: 10k Step: 1k

Analysis Type: tran

Y Axis: p(r1), p(v1), V(1), V(2), V(3), vout

X Axis: default - time, i(c1), i(l1), i(r1), i(v1), p(r1), p(v1)

Plot List:

Y Axis	X Axis
V(3)	

Other more complex sweeps can be performed directly in IsSpice4

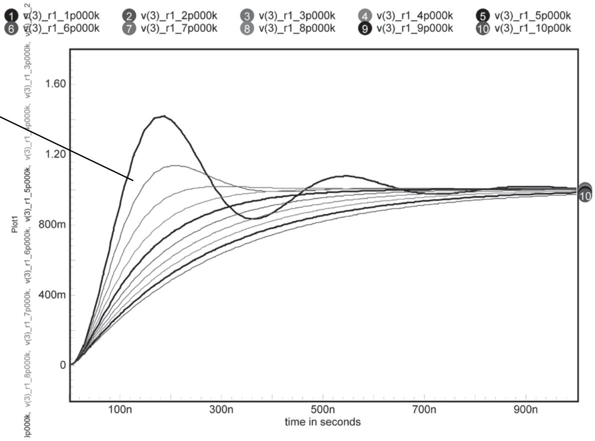
Note: When you're using the Alter function, you may only select an analysis that you've performed.

- **Select "Scope" from SpiceNet's Actions menu. Notice the curve family in IntuScope as shown below.**

This curve family was generated automatically as the value of R1 was swept.

All Y Axes can be scaled and linked.

IntuScope functions are discussed in the IntuScope tutorial.



- **Bring the schematic back to view, then press the Esc key to go back to normal cursor mode.**
- **Do "New Graph" in IntuScope before running temperature sweep example.**

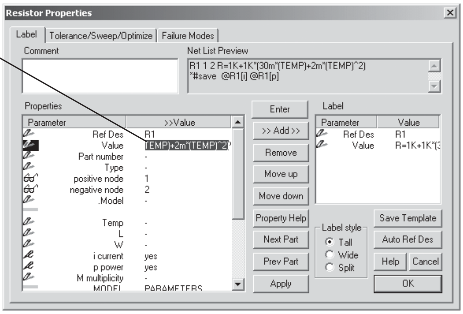
Running a Temperature Sweep

Next, we'll perform a temperature sweep for this circuit and look at the curve family output waveforms. In short, to perform a sweep through a range of circuit temperatures, simply click on an empty area of the schematic with the Alter tool. Then, proceed with the usual steps to sweep the design's temperature.

However, since there are no active parts (i.e., transistor, MOSFET) in our schematic example, and therefore the circuit is not very temperature sensitive, let's make the results more interesting. Let's change the value of R1 so that it's a function of temperature.

- **Double-click on Resistor R1.**
- **Then type the following equation into its value field: R=1K+1K*(1+30m*(TEMP)+2m*(TEMP)^2)**
- **Click on the OK button to close the Properties dialog.**
- **Run a simulation.**

You can enter an equation in the Value field for any resistor, inductor, or capacitor.



Resistor Properties

Net List Preview

```
R1 1 2 R=1K+1K*(30m*(TEMP)+2m*(TEMP)^2)
*#save  @R1[i] @R1[p]
```

Properties

Parameter	>>Value
Ref Des	R1
Value	TEMP)+2m*(TEMP)^2)'
Part number	-
Type	-
positive node	1
negative node	2
.Model	-
Temp	-
L	-
W	-
i current	yes
p power	yes
M multiplicity	-
MODEL	PARAMETERS

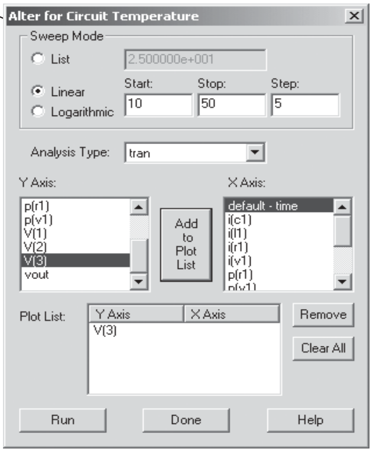
Label

Parameter	Value
Ref Des	R1
Value	R=1K+1K*(3

Label style: ● Tall ○ Wide ○ Split

- **Activate the Alter tool. Then click on an empty area of the schematic.**
- **Change the Sweep Mode to Linear. Enter the values 10 for start, 50 for stop and 5 for step, as shown.**
- **Add to Plot List the node voltage v(3) for the Y axis.**

Notice the title "Alter for Circuit Temperature".



Alter for Circuit Temperature

Sweep Mode

○ List 2.500000e+001

● Linear Start: 10 Stop: 50 Step: 5

○ Logarithmic

Analysis Type: tran

Y Axis: p(r1), p(v1), V(1), V(2), V(3), vout

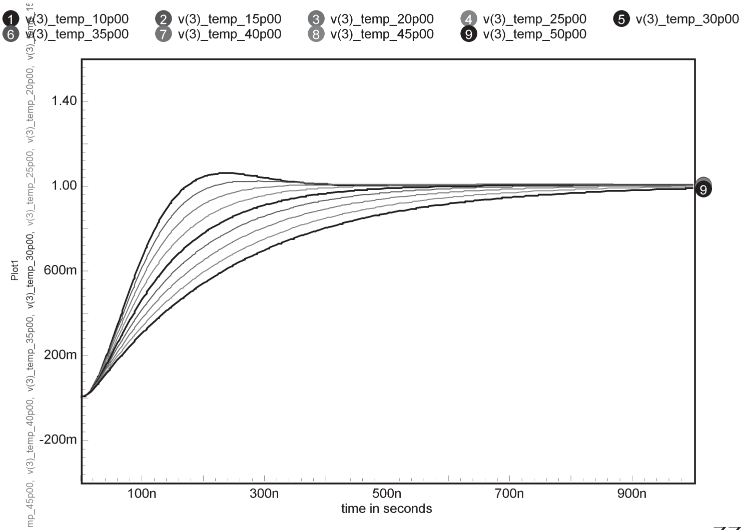
X Axis: default - time, i(c1), i(l1), i(r1), i(v1), p(r1), p(v1)

Plot List: Y Axis: V(3) X Axis:

This will sweep the circuit temperature from 10 degrees Celsius to 50 degrees Celsius, in increments of 5 degrees Celsius, and display a curve family in IntuScope.

- **Then click on the Run button in order to execute the simulation sweep.**

The resulting curve family is displayed.



Schematic Layers

Let's enhance our RLC circuit to create a more practical example.

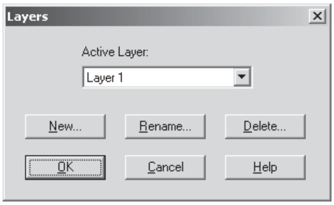
You may create a schematic that has multiple layers. All layers are superimposed to create the complete design.

The parts we've placed so far have been on Layer 1, which is the default layer. We'll create a new layer, Layer 2, and place two active devices (e.g. BJTs), a subcircuit device (e.g., WIRE) and some additional parts on that layer. Once we have added the new parts to the new layer of the schematic, we'll briefly discuss the topic of "configurations."

To create a new layer;

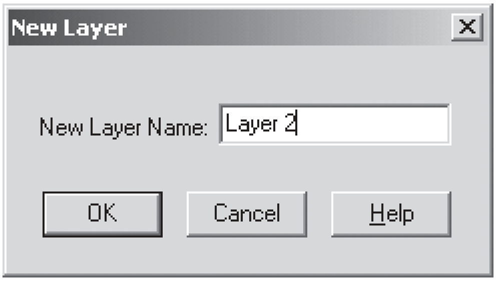
- **Select Layers from the Options menu in SpiceNet.**

See the "Working with Schematic Layers" section in the on-line help for more information about schematic layers.



Active Layer: Layer 1

- **Click on the New button.**
- **Type "Layer 2" into the "New Layer Name:" field.**



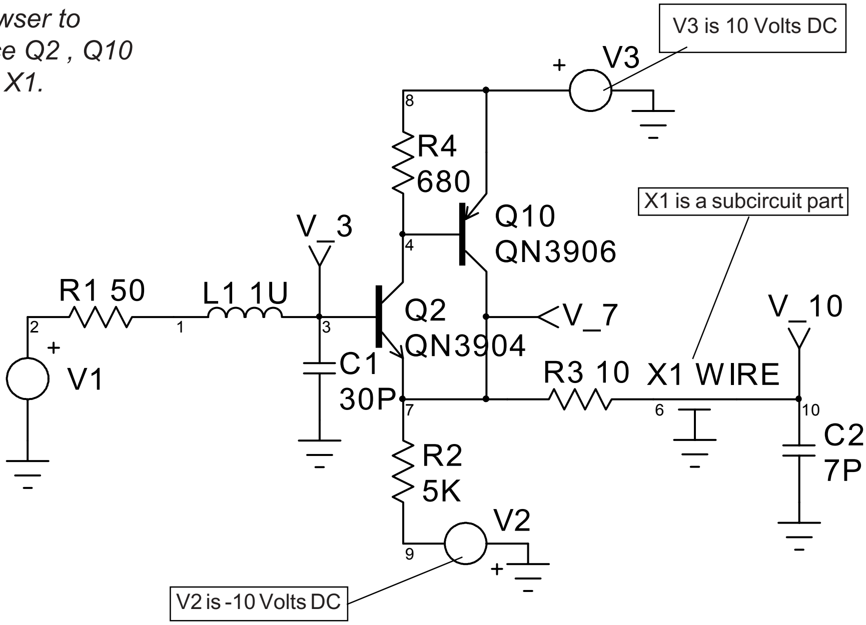
New Layer Name: Layer 2

- **Click on the OK button to close the New Layer dialog.**
- **Click on the OK button to close the Layers dialog.**

Now we'll add a Darlington PNP-NPN configuration with a long transmission line and a capacitive load to our new Layer 2.

Use the Parts Browser to place Q2 , Q10 and X1.

- **Capture the schematic as shown below.**

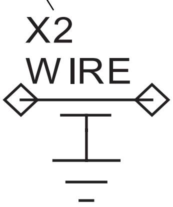


The QN3904 is in the BJTs NPN/Switching category in the Parts Browser. The QN3906 is in the BJTs PNP/Gen. Purpose category. The WIRE is in the Interconnects/Generic category. V2 and V3 are both 10 Volts DC, but notice the polarities.

Your schematic may vary from the one shown above, depending upon the sequence in which parts are placed and connections are made. Variations in the node numbers or reference designators will not affect the simulation results.

Subcircuit Parameters

WIRE symbol



Notice that the reference designator for the WIRE device is “X1.” In IsSpice4, “X” represents a subcircuit device.

The X1 WIRE symbol introduces a new concept. Unlike the active devices, which are IsSpice4 primitive elements (basic building blocks), the transmission line wire is an IsSpice4 “subcircuit” (collection of IsSpice4 primitive elements that are grouped in order to emulate a particular device).

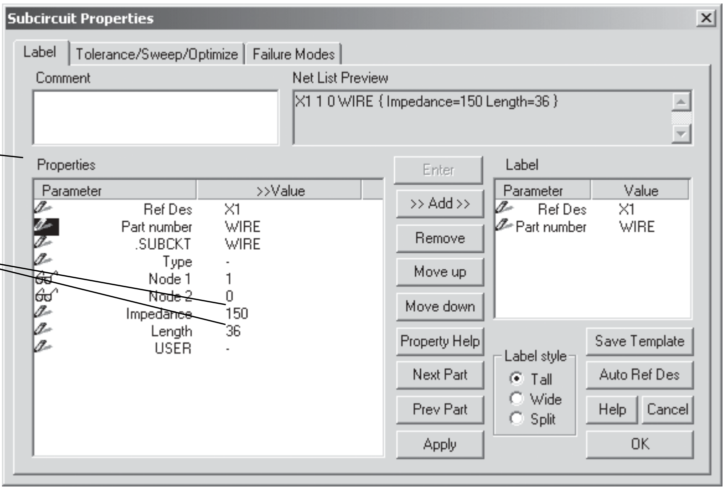
Some subcircuits, called “generic subcircuits,” have user-defined parameters that characterize their behavior. These parameters appear within curly braces, e.g. {Impedance=???}.

The WIRE device is a generic subcircuit. It can be used to represent a number of transmission lines just by inserting different values for the user-defined parameters.

- **Double-click on X1, which is the WIRE in the schematic.**

These are the parameters for the WIRE device.

We’ll enter the two parameters here.



Net List Preview: X1 1 0 WIRE { Impedance=150 Length=36 }

Properties

Parameter	>>Value
Ref Des	X1
Part number	WIRE
.SUBCKT	WIRE
Type	-
Node 1	1
Node 2	0
Impedance	150
Length	36
USER	-

Label

Parameter	Value
Ref Des	X1
Part number	WIRE

Label style: ● Tall ○ Wide ○ Split

- **Click on the “???” string (unless already displaying 150) in the Impedance parameter value field (Z) in the Subcircuit Properties dialog.**

- **Replace the ??? string with 150.**

- **Click on the “???” string (unless already displaying 36) in the Length parameter value field.**

- **Replace the ??? string with 36.**

Length is the wire length in inches, and Impedance is the impedance in ohms. We have assigned values that represent a clip lead model. All user-defined parameters for IsSpice4 symbols and generic subcircuits are entered in this manner.

- **Click on OK to close the Subcircuit Properties dialog.**

Note: Any occurrence of “???” in the Netlist will abort the simulation and bring up the Net List Errors dialog with a parts in trouble list.

Test Configurations

When a schematic is initially captured, the default test configuration name is "Config 1".

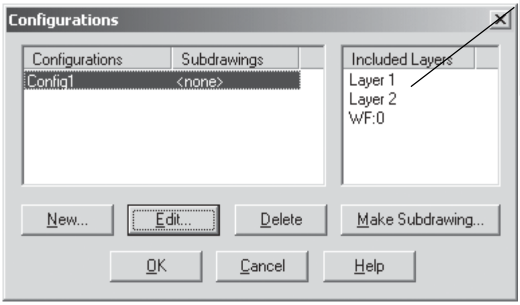
A Test Configuration consists of a specific set of layers. You may include any combination of existing layers into each configuration. Each configuration should be thought of as a set of transparencies that are stacked together. When a netlist is saved, the stacked set of transparencies, or layers, are combined (flattened) to form the active design.

Different configurations can be set up to drive different simulation scenarios. For example, you might dedicate one configuration for a Monte Carlo analysis and another for a transient analysis. You may have one configuration containing test circuitry, and another that omits "non-design" elements for PCB layout.

The configuration feature alleviates the need for you to manage separate versions of the same overall schematic for different analyses, test cases or end use (e.g., PCB, mil. standard, etc.).

To create a new test configuration:

- **Select the Configurations, Edit from the Options menu.**



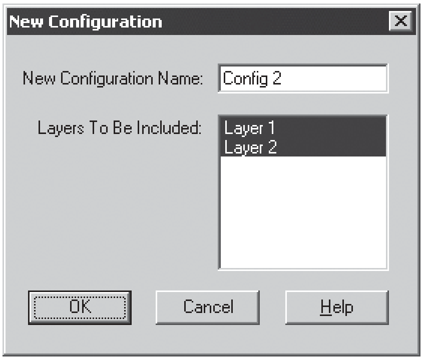
These are all of the existing layers for the selected configuration

Layer 1 contains the original RLC circuit (i.e., V1, R1, L1, C1, and the "Iinput" and "Voutput" test points). Layer 2 contains all of the other parts just placed. WF:0 contains the cross-probed waveforms placed on the schematic.

- **Click on the New button.**

When a new configuration is created, all existing schematic layers are automatically added to it.

As new layers are created, they are automatically added to the <u>active</u> configuration.


New Configuration

New Configuration Name: Config 2

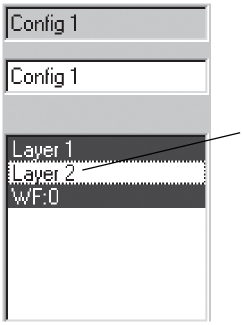
Layers To Be Included: Layer 1, Layer 2

- **The name "Config 2" is automatically suggested for you. Click on the OK button to accept it.**

If we were to add any additional layers, they would automatically be added to the currently selected configuration, Config 2.

Config 1 and Config 2 now contain the same layers, i.e., Layer 1, and Layer 2, since all layers are selected by default.

To edit a test configuration:


Config 1

Config 1

Layer 1, Layer 2, WF:0

- **Click on the "Config 1" configuration and click on the "Edit..." button.**

- **Click on the "Layer 2" layer to deselect it, so that only Layer 1 is selected.**

- **Click on the OK button to close the Edit Configuration dialog.**

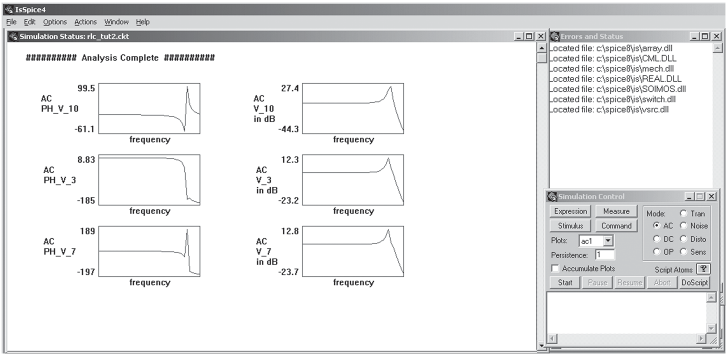
- **Click on the OK button to close the Configurations dialog.**

Notice that the parts on Layer 2 don't appear on the drawing because Config 1 is selected, which doesn't have parts in Layer 2.

Let's run a simulation of Config 1.

- **Select the Simulate function from the Actions menu.**

This is the simulation of Config 1, which only contains Layer 1 and WF:0.



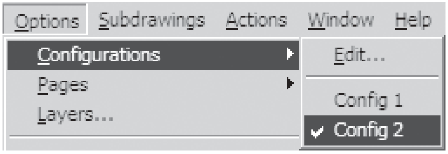
```
########## Analysis Complete ##########
```

```
Located file: c:\spice8\is\array.dll
Located file: c:\spice8\is\CML.DLL
Located file: c:\spice8\is\mech.dll
Located file: c:\spice8\is\REAL.DLL
Located file: c:\spice8\is\SOIMOS.dll
Located file: c:\spice8\is\switch.dll
Located file: c:\spice8\is\vsrc.dll
```

Mode: ● AC
Plots: ac1
Persistence: 1
□ Accumulate Plots

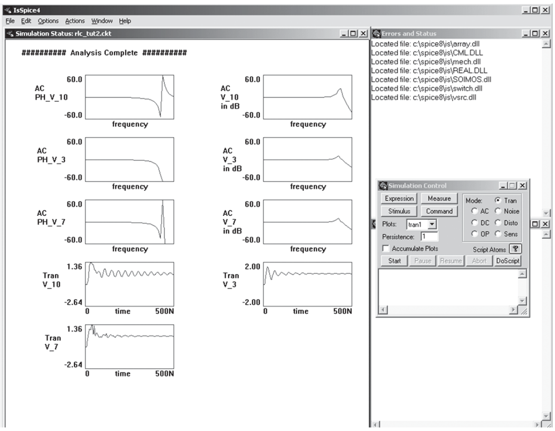
Now, let’s run a simulation of Config 2.

- **Point to the Configurations function in the Options menu and select Config 2.**



- **Select the Simulate function from the Actions menu.**

This is the simulation of Config 2, which contains Layer 1, WF:0 and Layer 2.



```
########## Analysis Complete ##########
```

```
Located file: c:\spice8\is\array.dll
Located file: c:\spice8\is\CML.DLL
Located file: c:\spice8\is\mech.dll
Located file: c:\spice8\is\REAL.DLL
Located file: c:\spice8\is\SOIMOS.dll
Located file: c:\spice8\is\switch.dll
Located file: c:\spice8\is\vsrc.dll
```

Mode: ● Tran
Plots: tran1
Persistence: 1
□ Accumulate Plots

Displaying Operating Point Values

Important: IsSpice4 must remain open in order to obtain the operating point values.

ICAP/4 has the ability to display the operating point values of the circuit directly on the schematic.

- **Bring the schematic back to view.**

To turn on (display) the OP values on the schematic:

- **From the Options menu, Refresh OP values then Options, Display on Drawing. Then, put a check besides OP voltages or OP current&power.**
- **Or in the center of the default toolbar press refresh OP values, Show OP Volts, and show OP Curr/Pow buttons.**

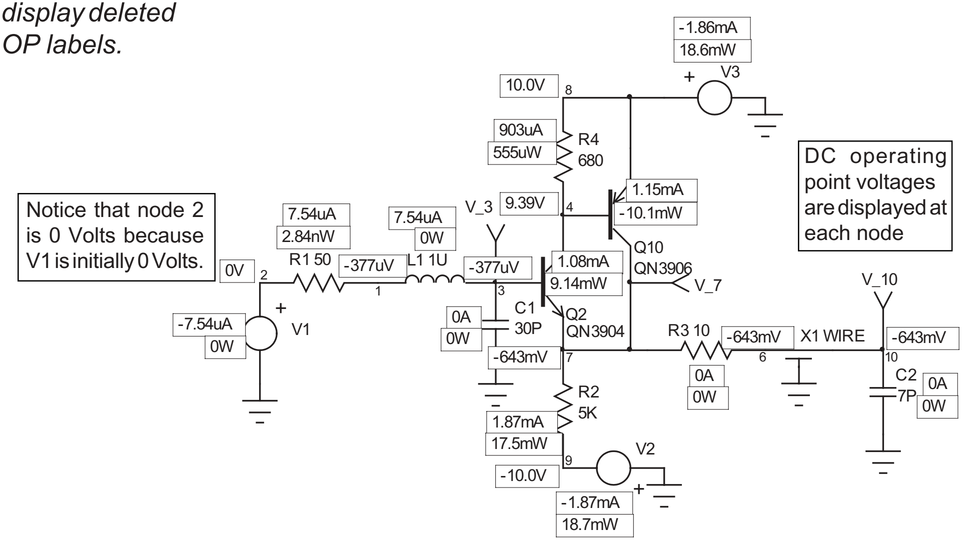
You may relocate (drag) node labels to suit your taste.

Once the OP values are displayed on the schematic, you can select and move them, or you may delete them (press the delete key). The position of all OP labels is remembered between sessions. If you forget which label belongs to what entity, just move your cursor over the label and a tool tip will appear specifying what entity it belongs to.

Note: *Remember to use Refresh OP values to update labels with new number of significant digits, or re-display deleted OP labels.*

Voltage labels end with V, Current labels end with A, and Power labels end with W. Use Options, Drawing, Significant Digits... to control the number of digits shown in the labels.

The schematic will look similar to this:



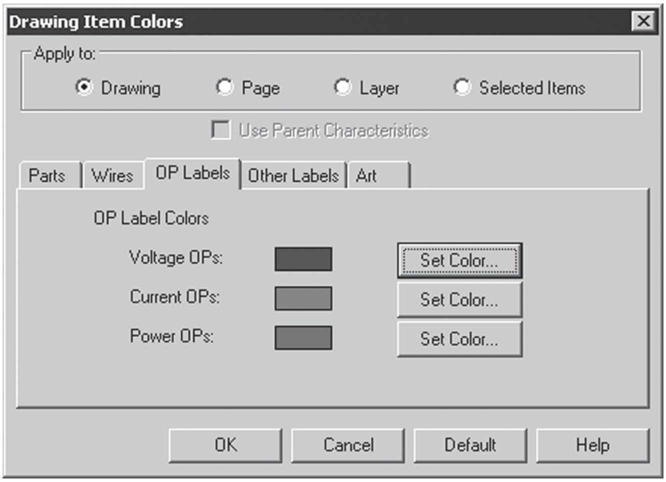
Notice that node 2 is 0 Volts because V1 is initially 0 Volts.

DC operating point voltages are displayed at each node

If you want to modify the OP value labels' display color:

- **Select Drawing Item Colors from the Options menu.**
- **Click on the OP Labels tab to open it.**
- **Click on the Set Color button, and select your display color.**
- **Click on the OK button to dismiss the dialog.**

Drawing Item Colors icon



Drawing Item Colors

Apply to: ● Drawing ○ Page ○ Layer ○ Selected Items

□ Use Parent Characteristics

Parts | Wires | OP Labels | Other Labels | Art

OP Label Colors

Voltage OPs: ▆ Set Color...

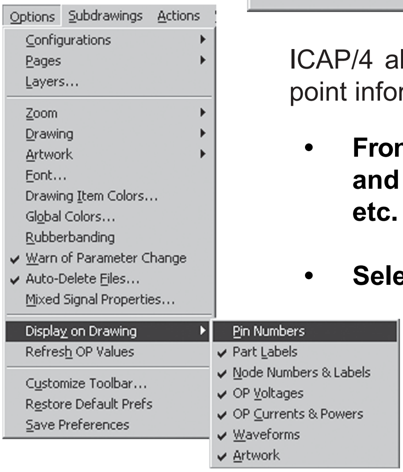
Current OPs: ▆ Set Color...

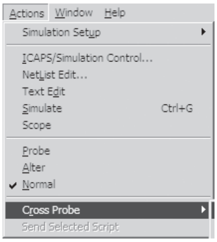
Power OPs: ▆ Set Color...

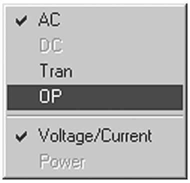
OK | Cancel | Default | Help

ICAP/4 also has the ability to display detailed operating point information for a specific schematic part.

- **From the Options Menu, select Display on Drawing and check or un-check pin numbers, part labels, etc. to display or make invisible.**
- **Select Actions, Probe to activate the probe tool.**

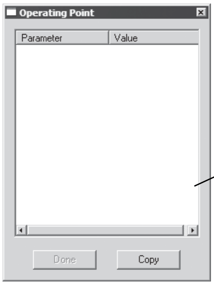






- **Select the OP option in the Actions, Cross Probe menu.**

- **Select Actions, Probe to activate the probe tool.**



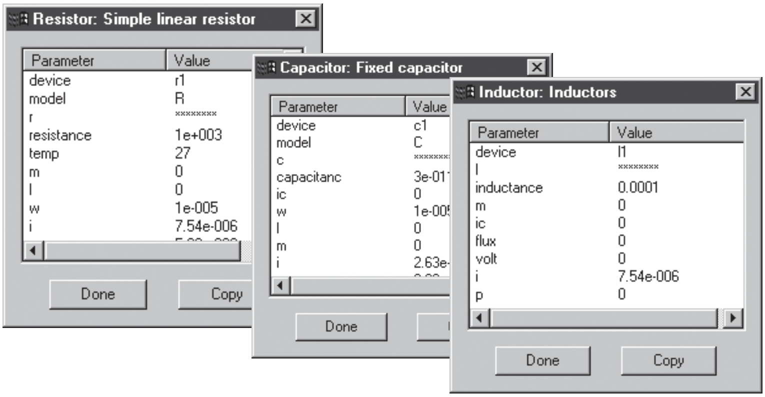
A blank OP dialog box is displayed, since no part was selected when the OP function was chosen.

You can also display the operating point information for the other parts by clicking on them.

Let’s look at the operating point information for resistor R1.

- **Using the OP probe tool, click on R1 to display its operating point information.**

You may click on a node to see its DC operating point value.



Resistor: Simple linear resistor

Parameter	Value
device	r1
model	R
r	********
resistance	1e+003
temp	27
m	0
l	0
w	1e-005
i	7.54e-006

Capacitor: Fixed capacitor

Parameter	Value
device	c1
model	C
c	*******
capacitanc	3e-01
ic	0
w	1e-00
l	0
m	0
i	2.63e-

Inductor: Inductors

Parameter	Value
device	l1
l	********
inductance	0.0001
m	0
ic	0
flux	0
volt	0
i	7.54e-006
p	0

When you quit SpiceNet, it automatically closes. IsEd and IntuScope do not close.

- **Press the Esc key to return to normal mode.**

- **Click on the Done button to close the Operating Point dialog.**

- **Select Save from the File menu if you want to save your work.**

- **Select Exit / Quit from the File menu to quit SpiceNet.**

This concludes the SpiceNet tutorial. In the next tutorial, we’ll discuss the basic functions of the IsSpice4 Simulator.

Tutorial #4 - IsSpice4

Unlike many analog/mixed-signal simulators, IsSpice4 furnishes a uniquely interactive version of SPICE.

In this tutorial we will go through the following actions:

- Starting IsSpice4 and its operation of its Simulation Control, real-time waveform, output and error windows
- Interactive sweep and measuring circuit parameters
- Interactive run of different analyses
- Automatic build of curve families in IntuScope

A detailed description of things like menu items, dialog boxes and specific functionality of IsSpice4, including syntax reference for IsSpice4 ICL commands, is found in the in-application help from the SpiceNet or IsSpice4 windows' help pulldown.

Starting IsSpice4

IsSpice4 makes an output file just like other SPICE programs.

Let's first explore some of the IsSpice4 features by simulating a simple low-pass filter.

- **Open the AKTP2.DWG design under the working SPICE8/IS/AKTP2 directory.**

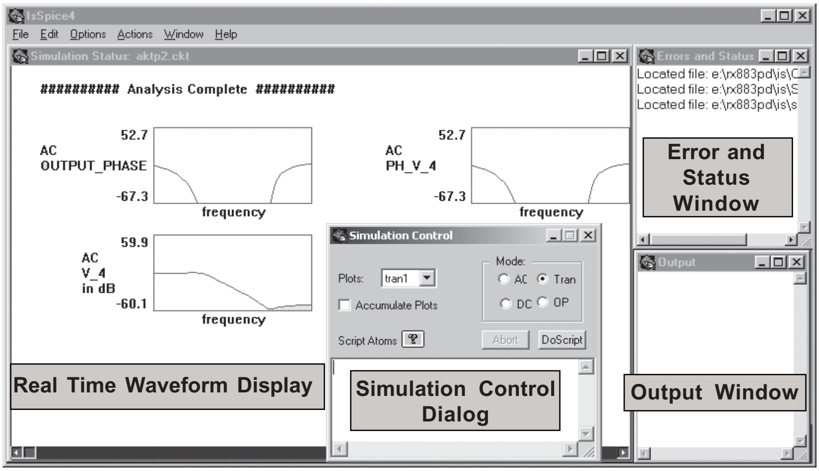
- **Select Simulate from the Actions menu.**

IsSpice4 will load the SPICE netlist and run the designated analysis. Once the simulation is complete, an output file will be created and the Simulation Control dialog will be displayed. You may optionally interact with the simulator by entering command scripts into the Script window of the Simulation Control dialog.

Reviewing The IsSpice4 Display

The IsSpice4 display contains four different windows: a Real-Time waveform display, a Simulation Control dialog, an Errors and Status window, and an Output window. In addition, the Simulation Control window has several functions that can activate other windows. These will be covered as we proceed.

Note: Your waveform names and display may appear slightly different.



########## Analysis Complete ##########

Located file: e:\rx883pd\is\C
Located file: e:\rx883pd\is\S
Located file: e:\rx883pd\is\s

Error and Status Window

Real Time Waveform Display

Simulation Control Dialog

Output Window

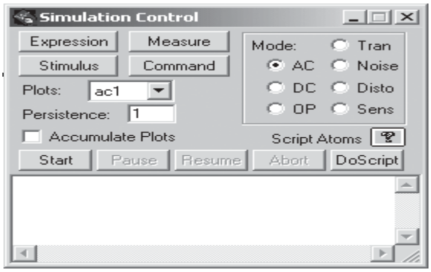
After the simulation is complete, a display similar to the one above will appear. The Real-Time display (in blue) shows the circuit performance as a simulation runs. The Simulation Control dialog is used to control simulation flow interactively (if desired), and it provides access to past simulation data. The Errors and Status window provides corresponding data about a simulation. The Output window functions similar to a traditional SPICE output file. It stores data in textural format and is produced by the simulator.

- You can always rearrange the position of each window, then select Save Preferences under the Options menu.

Working With the Simulation Control Dialog

This dialog is used to control the simulation flow. It provides access to simulation data, and to interactive parameter sweeping features, all following an initial simulation run. For this currently active simulation, AC is checked in the Mode section. The Start, Pause, Resume and Abort buttons are used to control the simulation.

- **Click the Start button.**



Plots: ac1

Persistence: 1

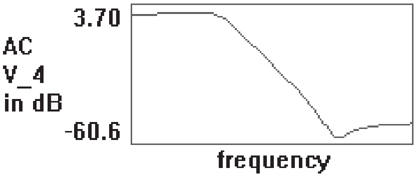
□ Accumulate Plots

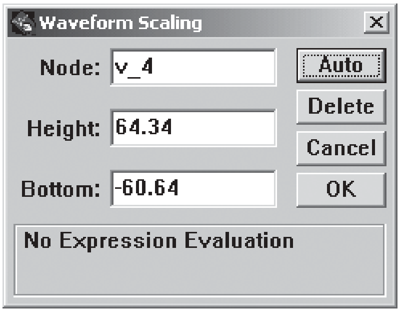
Mode: ○ Tran ● AC ○ Noise ○ DC ○ Disto ○ OP ○ Sens

The Start button clears the Real-Time waveform display and immediately runs the last performed analysis.

Scaling Waveforms

Once a design simulation is run, IsSpice4 automatically scales the Real-Time waveforms based on the .OPTIONS parameters Vscale, Iscale, and Logscale. However, at any time before, during, or after a simulation, you can re-scale any waveform. All waveforms can be automatically scaled simultaneously using the AutoScale function in the Options menu (<Ctrl>+T).




Waveform Scaling

Node: v_4

Height: 64.34

Bottom: -60.64

No Expression Evaluation

Double-click on a waveform to re-scale it individually. The Autoscale function in the Options menu autoscales all waveforms simultaneously.

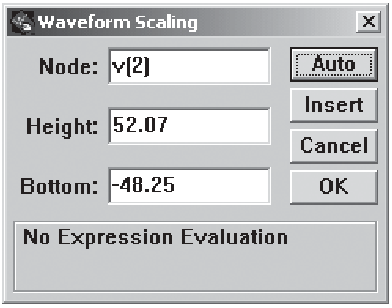
- **Double-click directly on the waveform V_4 in the blue waveform window. Click the Auto button. Click OK.**
- **Do the same for the phase waveform (PH_V_4).**

Adding and Deleting Waveforms

Before or after the initial simulation, you will be able to edit the display by deleting or adding waveforms.

To Add a Waveform:

- **Double-click in the blank area underneath the V_4 waveform.**
- **Enter V(2) into the blank "Node:" field. Click the Auto button and select OK.**


Waveform Scaling

Node: v(2)

Height: 52.07

Bottom: -48.25

No Expression Evaluation

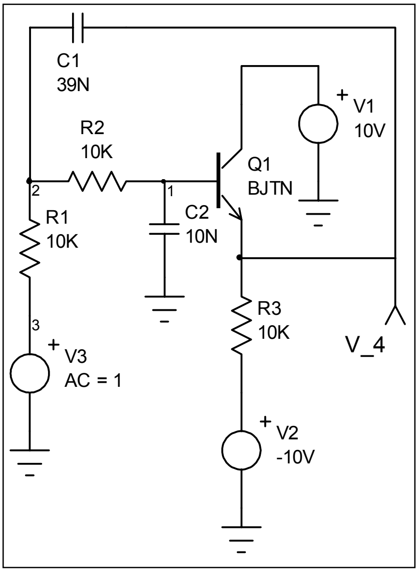
V(2) will be displayed. Since the netlist includes the “save all...” function, we have access to all of the node voltages. Therefore, any circuit node number can be entered into the ”Node:” field. Note: only waveforms from the active analysis can be re-scaled.

To delete a waveform:

- **Double click the desired waveform.**
- **Click the Delete button in its Waveform Scaling dialog.**
- **Hold the Ctrl key and press T (<Ctrl>+T) to autoscale the waveform.**

Saving Vectors

The Real Time display will initially show the vectors listed in the .PRINT and .VIEW statements (created by test point symbols), and the Interactive Command Language (ICL) view statement. After the simulation is complete, all Print Expression vectors created with the ICL alias and view statements will be displayed.

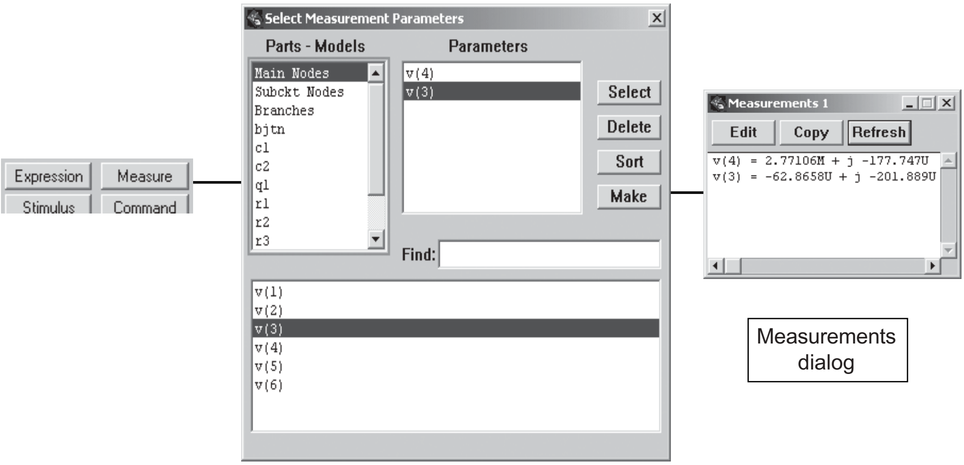


For AC, noise and distortion analyses, however, only the DB and phase quantities of .PRINT vectors are available as real-time waveforms. Other quantities, such as the real and imaginary parts, must be displayed using the ICL Alias command. Note: Print Expression vectors created with the Alias command are always displayed after the simulation is complete.

You can also add any waveform that has been saved with the ICL save statement to the Real Time display. All of the voltages, currents and device power dissipations from the circuit are available for all analyses, including various ICL statements.

Measuring Circuits Interactively

The operating point of the circuit can greatly affect the simulation results, especially for the AC analysis. With this in mind, the Measurements dialog can be used to closely examine operating point, and real-time voltages and currents.



```
v(4) = 2.77106M + j -177.747U
v(3) = -62.8658U + j -201.889U
```

Measurements dialog

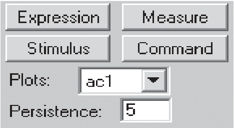
- **Click the Measure button in the Simulation Control dialog to activate the Select Measurement Parameters dialog.**
- **Click on the “Main Nodes” entry. Double-click on the “V(4)” entry in the bottom field. Double-click on the “V(3)” entry.**
- **Click on the Make button.**
- **Click on the Start button to run an AC analysis.**

The Measurements dialog will display the operating point voltage at node 4 and node 3 using real and imaginary quantities. Then, as the simulation runs, the quantities will be updated. You can use this feature with any analysis to display the operating point and exact values of key circuit parameters.

- **Close the Measurement dialog.**

Sweeping Circuit Parameters

With IsSpice4, you can interactively explore the circuit design by sweeping different parameters and watching the results as real-time waveforms appear. This type of feedback enables you to interactively optimize circuit performance as shown below.

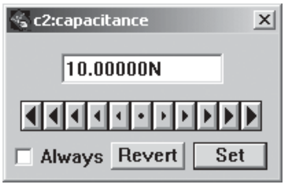


Plots: ac1

Persistence: 5

- **Set the Persistence value in the Simulation Control dialog to 5.** This will allow 5 waveforms to be displayed in a graph before the display is redrawn.

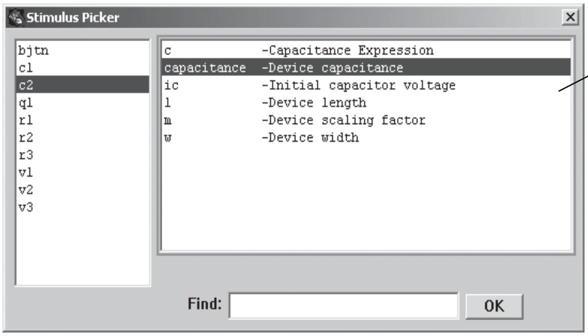
- **Click on the Stimulus button in the Simulation Control dialog to activate the Stimulus Picker dialog.**



10.00000N

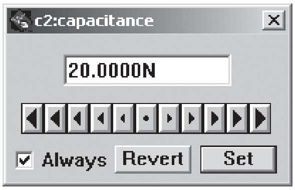
□ Always

Stimulus dialogs



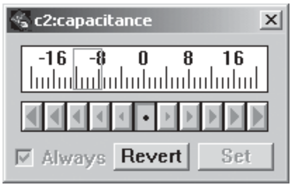
bjtn	c -Capacitance Expression
c1	capacitance -Device capacitance
c2	ic -Initial capacitor voltage
q1	l -Device length
r1	m -Device scaling factor
r2	w -Device width
r3	
v1	
v2	
v3	

Stimulus Picker dialog



20.0000N

☑ Always



Let's sweep the capacitor value for C2 and see what happens.

- **Click on the entry "c2". Then double-click on the "Capacitance - Device Capacitance" entry, or press the OK button to display the Stimulus dialog. Close the Stimulus Picker dialog by pressing the Close icon ✕ in the upper right hand corner.**

- **Place the mouse on the right most (largest) arrow and click once to increase the largest digit. Click the Set button to run the analysis with the values shown.**

To change the order of magnitude, click on the dot in the center and select from the sliding scale.

The C2 value will change to 20nF and the simulation will be rerun. Note that you can increase or decrease the value by 5 orders of magnitude.

- **Check the Always option. Place the mouse on the right-most arrow and hold the mouse button down.**

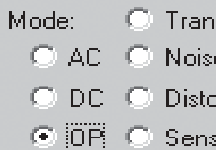
Stimulus picker left arrows are used to decrease the value; the right arrows are used to increase the value.

After the simulation runs, the value will change and the simulation will rerun. By holding down the arrow, we can sweep the value and watch the results.

- **When you are finished, click the Revert button in the "c2: capacitance" dialog. Then click the Set button to set the circuit back to the way it was when you started.**

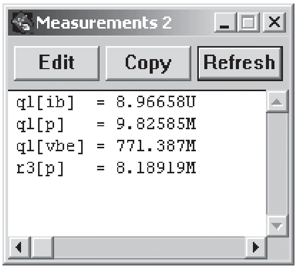
Viewing Operating Point Changes

You can combine interactive design stimulus with interactive measurements to thoroughly evaluate the operating-point performance of a design.


Mode: ○ Tran
○ AC ○ Nois
○ DC ○ Dist
● OP ○ Sens

- **Click on the OP option in the "Mode:" section of the Simulation Control dialog.**

  - **Click the Measure button. Select "q1." Then, double click the "ib- Current at base node," "p - Power dissipation," and "vbe - B-E voltage" entries. Select "r3." Double click the "p - Device" power entry. Click on the Make button.**



```
q1[ib]  = 8.96658U
q1[p]   = 9.82585M
q1[vbe] = 771.387M
r3[p]   = 8.18919M
```

  - **Click the Stimulus button. Select "r3." Then, double-click on the "resistance-Resistance" entry. Close the Stimulus Picker dialog.**

  - **Check the Always option. Change the value.**

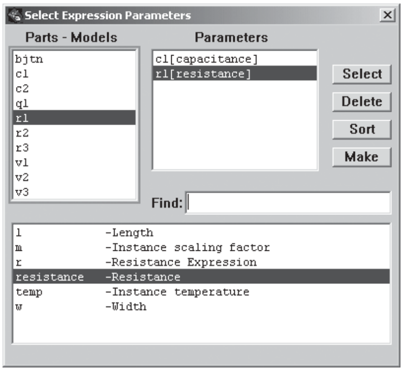
As you change the value, you will see the operating point information for the selected parameters update.

- **When you are finished, click the Revert button in the "r3: resistance" dialog. Then, click the Set button. Close the Stimulus and Measurement dialogs. Click the AC button in the "Mode:" section.**

Sweeping Groups of Parameters

The Expression dialog works in a manner similar to the Stimulus dialog. However, several circuit variables may be swept in tandem.

- **Click on the Expression button in the Simulation Control dialog to activate the Select Expression Parameters dialog.**



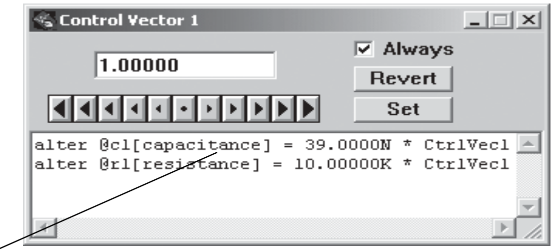
```
bjtn
c1
c2
q1
r1
r2
r3
v1
v2
v3
```

```
c1[capacitance]
r1[resistance]
```

```
l           -Length
m           -Instance scaling factor
r           -Resistance Expression
resistance  -Resistance
temp        -Instance temperature
w           -Width
```

At this point, we could select several parameters and sweep them all together. However, for this example, we will go one step further.

The Expression dialog allows swept parameters to be functions of other circuit variables. For example, the cutoff frequency, fo, of the AKTP2 low-pass filter is ∝ 1/(R1*R2*C1*C2). If we change the component ratios, we will notice a significant change in the frequency response.



```
alter @c1[capacitance] = 39.0000N * CtrlVec1
alter @r1[resistance] = 10.00000K * CtrlVec1
```

Type Here

- **Click on "c1." Select the "capacitance-Device capacitance" entry. Click on "r1." Select the "resistance-Resistance" entry. Click the Make button. Change the entries to those shown above.** The default setting will show the default component value multiplied by CtrlVec1.

You can use virtually any device or model parameter in the Expression dialog. In this case, we will make C1 and R1

ISSPICE4 Stimulus and Expressions give you complete parameter sweeping flexibility.

functions of C2 and R2 respectively. As we sweep the value of CtrlVec1, we will change the ratios of the values.

- **Change CtrlVec1 to 2.0 and click the Set button.**

Change the CtrlVec1 value and see what it does to the waveforms.

- **When you are finished, set CtrlVec1 back to 1 and run the simulation to reset the values back to their defaults.**

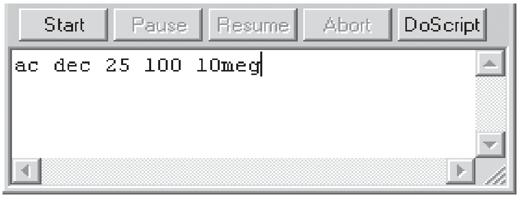
- **Close the CtrlVec1 dialog.**

Scripting: Introduction to ICL

The bottom half of the Simulation Control dialog is used for entry and execution of simulation scripts. This allows you to run analyses that weren't included in the initial IsSpice4 netlist, or send waveforms to IntuScope using the sendscript plot function. Unlike typical SPICE simulators, you don't have to quit the simulator in order to change analysis specifications. Simulation scripts can use any of the Interactive Command Language (ICL) functions.

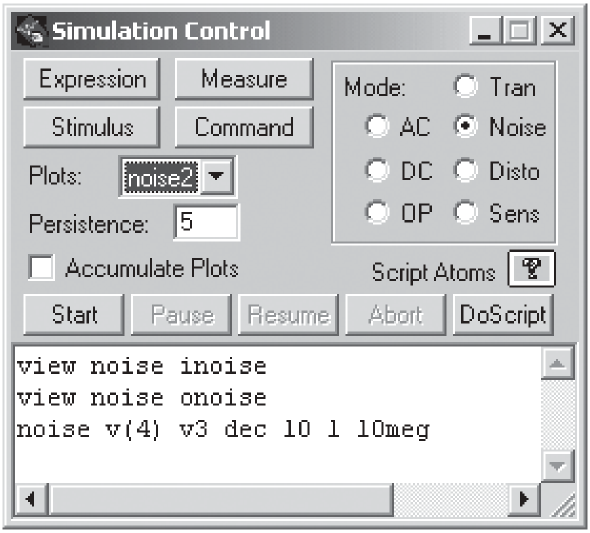
Let's change the AC analysis specification.

- **Type the line "ac dec 25 100 10meg" into the Simulation Control dialog. Click the DoScript button.**



```
ac dec 25 100 10meg
```

Now let's run a new analysis that we didn't specify in the original input netlist.



Plots: noise2

Persistence: 5

□ Accumulate Plots

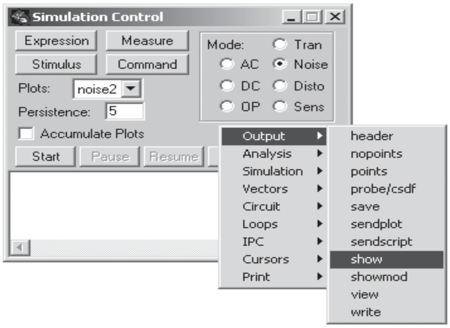
Mode: ○ Tran ○ AC ● Noise ○ DC ○ Disto ○ OP ○ Sens

```
view noise inoise
view noise onoise
noise v(4) v3 dec 10 1 10meg
```

- **Delete the ac line and enter the three lines shown in the figure to the left. Click the DoScript button.**
- **Change plot to noise1**

A noise analysis will be run, and the input and output noise waveforms will be displayed.

The Script Atoms pop-up contains all of the available ICL functions, which include most of the traditional SPICE functions.



Plots: noise2

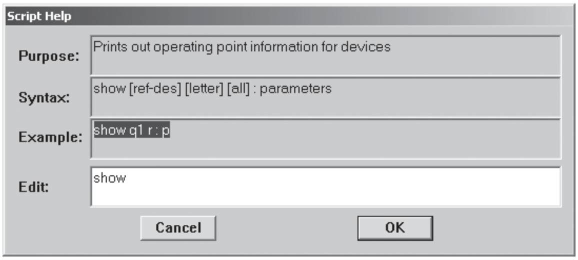
Persistence: 5

□ Accumulate Plots

Mode: ○ Tran ○ AC ● Noise ○ DC ○ Disto ○ OP ○ Sens

- **Delete the three lines in the script window.**
- **Select the Script Atoms, Output, Show function.**
- **Select all of the text in the Example: field. Press (<Ctrl>+ C). Select all of the text in the Edit: field. Press (<Ctrl>+ V) to paste the Example: text into the Edit: field. Select OK.**

Copy the help text (<Ctrl>+C) from the Example or Syntax fields. Select all of the text in the Edit field and paste (<Ctrl>+V) or enter your own options.



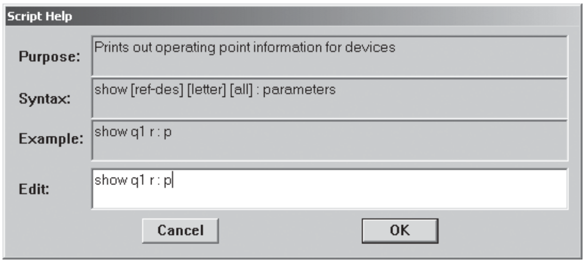
Purpose: Prints out operating point information for devices

Syntax: show [ref-des] [letter] [all] : parameters

Example: show q1 r : p

Edit: show

Move the Edit text to the script window by selecting OK.



Purpose: Prints out operating point information for devices

Syntax: show [ref-des] [letter] [all] : parameters

Example: show q1 r : p

Edit: show q1 r : p

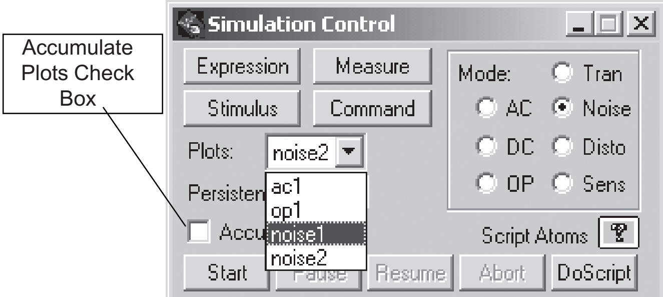
All of the information in the fields of the help dialog can be copied to the Edit: field at the bottom. The OK button causes the "Edit" field contents to be copied to the script window at the cursor position.

- **Click the DoScript button. Change the "show" function to "show all" and click the DoScript button again. This prints out all operating point information for devices to the IsSpice4 Output Window.**

Note: The show and showmod functions are used to display SPICE 2 style operating-point information in the Output window.

Saving and Viewing Past Simulation Data

IsSpice4 saves the data for each analysis it runs. Initially, one or more plot names are given to each analysis performed when you start IsSpice4. Some analyses (ac, dc, tran) receive one plot name. Other analyses (noise, distortion) will generate more than one set of vectors, and thus require one plot name for each set of vectors. For example, the noise analysis calculates the total integrated noise for each component over a frequency range, plus noise spectral density curves (input/output noise waveforms). Each set of data is given its own plot name. The same goes for the distortion analysis, which may run several passes for different numerical evaluations (f1+f2, f1-f2, 2f1-f2, etc.), depending on the options selected. Each set of vectors from each pass will be given its own plot name. The "display" command is useful for determining the contents of a plot.



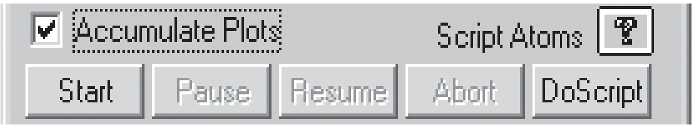
Plots: noise2

Mode: ○ Tran ○ AC ● Noise ○ DC ○ Disto ○ OP ○ Sens

□ Accu

By changing the plot name, we can access the data from past analyses.

- **Display the AC analysis data by selecting the "ac1" entry in the Plots pop-up in the Simulation Control dialog.**



☑ Accumulate Plots

- **Check the Accumulate Plots box. Change the value of a resistor using the Interactive Stimulus functions reviewed earlier.**

Your plot names may be different.

The data for each simulation will be given a new plot name. Note: As plots are accumulated, the amount of memory used by IsSpice4 increases.

- **To review the data from the previous AC analysis, pull down the Plots pop-up and select another plot.**

Note: IntuScope has access to the plot vector set shown in the Plots pop-up via the IsSpice4 Plots function in the Waveforms menu. Changing the plot changes the data set that these programs can access.

- **Click on the Command button.**

This window acts just like the script window. It is useful when the script window is occupied, and you don't want to delete the script, but rather run a few simple ICL commands.

- **Type "display" into the Command window. Click the DoScript button.**

This will show the vectors contained in the selected plot via the Output window.

- **Change to a different plot name. Click the DoScript button again in the Command window.**

- **When you are finished exploring IsSpice4, select Quit from SpiceNet's Actions menu to quit the session.**

Tutorial #5 - IntuScope

IntuScope is continuously enhanced to give you enhanced waveform selection, manipulation, scaling, viewing and math operations.

After a simulation is performed, you will want to view, analyze or manipulate the output data. In IsSpice4, the circuit waveforms are displayed in real time. However, the IntuScope program is specially designed for more in-depth waveform analysis.

IntuScope can read the simulation data directly from the IsSpice4 simulator to gain access to all voltages, device currents, and device power dissipations in the circuit. The data may be from AC, DC, Noise or Transient analyses.

IntuScope can also display waveforms from an IsSpice4 output file.

IntuScope is a versatile program, and has a wealth of functionality. And even though this tutorial is fairly complete, we will not discuss all of the capabilities of IntuScope. IntuScope is menu and icon driven, plus contains context-sensitive help. Experimentation is encouraged.

Starting IntuScope

- **Start ICAPS and open the Sample design (Sample.DWG) in the :\spice8\SN\Sample2 directory.**

Run Simulation icon

- **From the Actions menu, select Simulate, or click on the Run Simulation icon.**

The simulation is launched and the IsSpice4 screen appears.

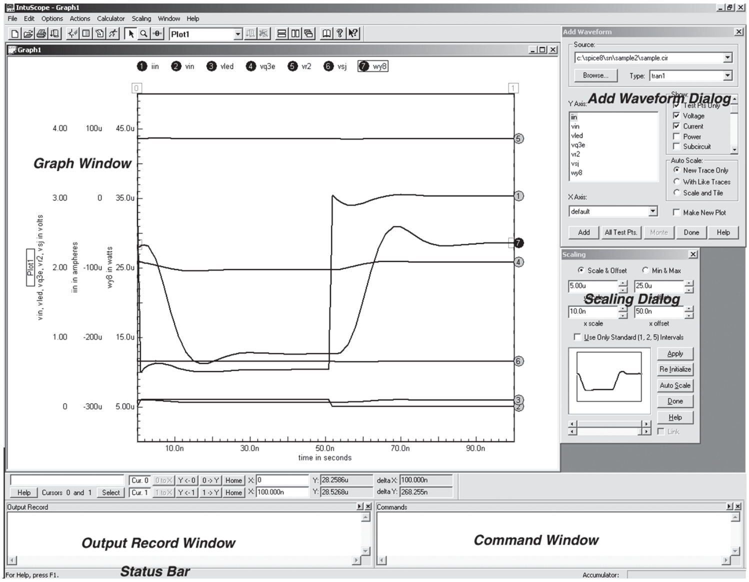
- **From the Actions menu or top toolbar, select "Scope" to launch the IntuScope program.**

The IntuScope Screen Layout

The IntuScope program contains a variety of windows. However, most of the action takes place in the Graph Window, Add Waveform dialog and Scaling dialog boxes.

When IntuScope is launched, a new waveform Graph window is opened automatically. Under IntuScope's Windows pulldown, use the Add Waveform dialog to plot your waveforms in the Graph Window (if not cross-probing directly from the schematic). Many of the fields at the bottom of IntuScope will be grayed out until a waveform is displayed.

The Output Record Window (lower left corner) shows the result of an ICL script that has been run, or any errors as a result of that run say if the user created a waveform script with error. The Commands Window allows you to optionally load an existing script or type in your own. The Status Bar provides brief instructions as to how to accomplish your current task in IntuScope, if necessary. The Accumulator Window displays the result of the measured values.



Getting Help

IntuScope has a complete online help system that contains information on all menu functions and commonly used operations.

The IntuScope Help Menu is dynamic. It allows you to choose from a list of HELP options, including movies, tutorials and documents.

From the Help menu:

- **Select IntuScope Help for help on all menu functions.**
- **Optionally, select Script References for descriptions and examples of the ICL scripts.**
- **Optionally, select Script Syntax for help on how to use scripts to create simulation templates.**

Opening a New Graph Window

Although unlimited new graph windows can be opened in IntuScope, the number of waveforms and plots created is limited to the viewing space of the graph window. When IntuScope is launched, a new graph window will automatically open.

- **Under the File Menu, select New Graph to open a new graph window.**
- **Under the Window menu, select Add Waveform to display the Add Waveform Dialog.**

Selecting a Waveform Data Type

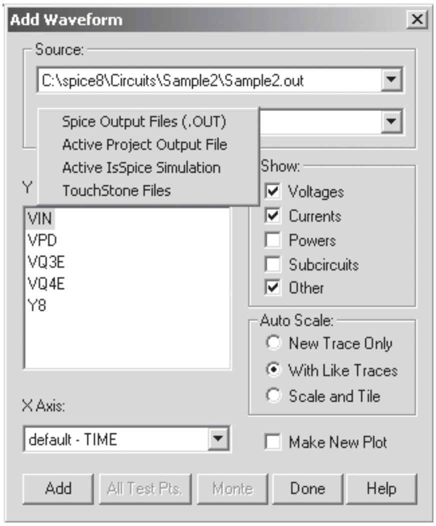
IntuScope can access waveform data from any ASCII text output file or directly from the IsSpice4 simulator. IntuScope will automatically choose the best data source available.

There are three main types of waveform data: Active IsSpice Simulations, Active Project Output Files and Spice Output Files. RF Deluxe users, and higher ICAP/4 packages, can also open Touchstone files, and plot S parameters.

To select waveforms from an active IsSpice4 simulation:

- **Open up a drawing file in SpiceNet to activate a project.**
- **Run a simulation, which places all non-interpolated simulation data in memory, as well as creates an ASCII text .OUT file containing interpolated simulation data.**
- **In IsSpice4, select the Action menu, then Scope, to launch IntuScope. By default IntuScope will link to the raw simulation data as long as IsSpice4 is active, i.e., open.**
- **Plot any saved vector (wave) by clicking it in the Y Axis field of the Add Waveform dialog, then click the Add button.**

To select an Active Project Output File:



- **Open a drawing file to activate the project. Run a simulation, which creates a .OUT file if one does not already exist.**
- **Launch IntuScope.**
- **Click on the Browse button in the Add Waveform dialog.**
- **Select the Active Project Output File to display only vectors saved in the .OUT file.**
- **Highlight any vector under the Y Axis and click the add button to plot the vector.**

Only interpolated test point waveform vectors and vectors specified with ICL commands are saved in the .OUT file. **The Active Project is your currently open schematic drawing file.** The Active Project Output File links to the .OUT file of the Active Project. The IntuScope vector list will automatically update if the .OUT file is changed by a new simulation run. Open a schematic, then IntuScope without simulating, and IntuScope will by default link to the Active Project .OUT file if the .OUT file exists. The Active Project Output file only stores data for the specific simulation, but the Active IsSpice Simulation holds all the data from multiple simulation runs such as Tran1, Tran2, AC1, etc.

To obtain data from different types of simulation runs in IntuScope:

- **Select the desired waveform vector set (simulation type) from the Type drop down menu of the Add Waveform Dialog.**

All available vectors for that selected vector set will show up in the Y Axis box.

- **Click on a desired vector in the Y axis list, then the Add button to plot a waveform graph.**

Here are some tips to make it easier to find vectors you want.

- **In the Add Waveform dialog, find the Show section. Check boxes for Voltage, Current, Power, Subcircuit and Other waveform types should be available.**
- **Only waveform vector types with check marks next to them will be available in the Y and X Axis list. Uncheck the checkbox of any vector types you want to hide.**
- **For an Active IsSpice Simulation data source, a Test Pts Only checkbox will be added to the top of the show list. Check this to remove all non-testpoint vectors.**

The Add Waveform Dialog provides control of how new waveforms are plotted.

- **Use the "Auto Scale" radio buttons to scale the y-axis of the new waveform when it is plotted. Options include:**

  - **"New Trace Only," which creates a new y-axis scale for the new trace without impacting any other traces that may already be in the current plot.**
  - **"With Like Traces" re-scales all traces, including the new one, whose y-axes have the same units as the new trace. This way, all can share a single y scale.**
  - **"Scale and Tile" re-scales all traces, including the new one, so that the traces do not overlap each other.**

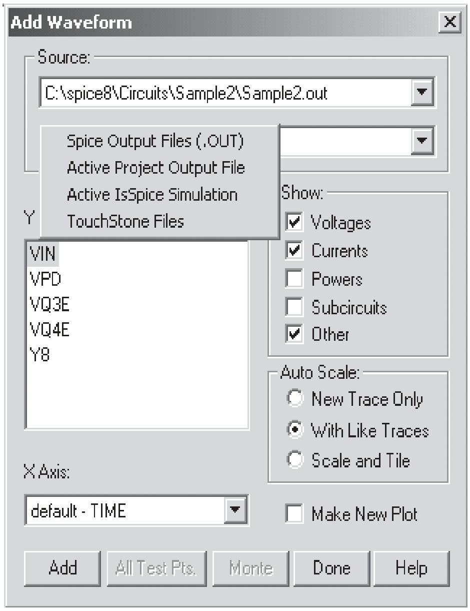
- **Check Make New Plot to make every new graph appear in a new plot.**

Selecting An Output File

If the simulator is running while IntuScope is launched, IntuScope will retrieve non-interpolated (raw) data from the IsSpice4 simulator. Otherwise, IntuScope will retrieve the linearly interpolated(re-sampled) data in the active project's output file.

IntuScope is capable of displaying data, derived from different output files, simulation analysis and X and Y axes, all on one graph. However, it is recommended to separate the information logically, using several graph windows.

To display the data that is not from the default output file:



Add Waveform

Source: C:\spice8\Circuits\Sample2\Sample2.out

Y: VIN, VPD, VQ3E, VQ4E, Y8

Show: ☑ Voltages ☑ Currents □ Powers □ Subcircuits ☑ Other

Auto Scale: ○ New Trace Only ● With Like Traces ○ Scale and Tile

X Axis: default - TIME

□ Make New Plot

Spice Output Files (.OUT)
Active Project Output File
Active IsSpice Simulation
TouchStone Files
Add
All Test Pts.
Monte
Done
Help

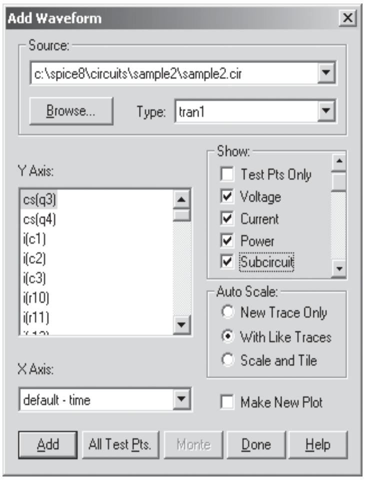
- **In the Add Waveform dialog, click on the Browse button.**
- **Select Spice Output Files (.OUT).**
- **Select any .OUT file. .OUT files are generated when a simulation is completed and placed in the same folder where the DWG was run.**

  If the desired file is the current Active Project .OUT file:
- **Choose the Active Project Output File after clicking on the Browse button.**

The .OUT file is an ASCII text file stored on a computer. Even after restarting the computer, you are still able to plot waveforms with the data from the .OUT file without re-simulation. Warning! Transient simulation uses a dynamic time-stop algorithm. The variable stop data is interpolated into the .OUT file. Transient data is more accurate when plotted from the Active IsSpice Simulation.

Displaying A Waveform

Show section enables you to limit the listed vectors to the ones you prefer.



Add Waveform

Source: c:\spice8\circuits\sample2\sample2.cir

Browse... Type: tran1

Y Axis: cs(q3) cs(q4) i(c1) i(c2) i(c3) i(r10) i(r11)

Show: ☐ Test Pts Only ☑ Voltage ☑ Current ☑ Power ☑ Subcircuit

Auto Scale: ○ New Trace Only ● With Like Traces ○ Scale and Tile

X Axis: default - time

☐ Make New Plot

To display a waveform:

- **In the Add Waveform dialog, click on the Browse Button and select a proper source.**

- **Select a desired waveform vector set from the Type pulldown menu.**

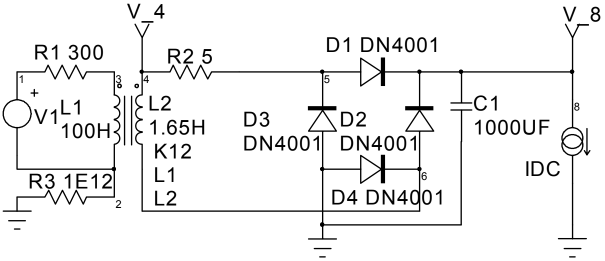
- **Select the Y and X axis vector for the waveform you want to plot.**

- **Click on the Add button.**

Combining and Separating Waveforms

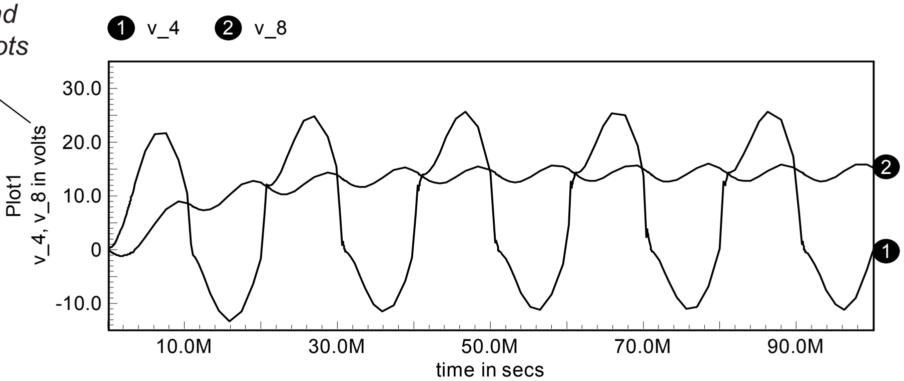
IntuScope enables you to combine separated waveforms into one plot, and to separate combined waveforms into several new plots.

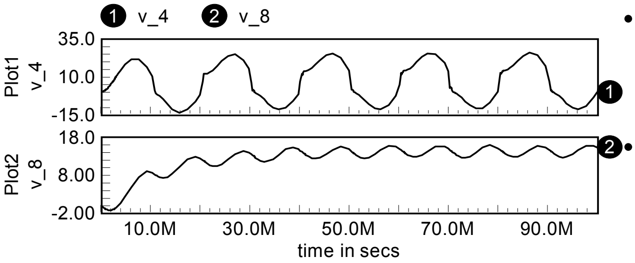
- **Open this drawing :\spice8\Circuits\Demo\ac-dc.DWG**



- **Click on the run simulation icon.**
- **From Actions menu select “Scope” to launch IntuScope.**
- **In the Add Waveform dialog’s Show section, check Test Pts Only, Voltage and Current.**
- **Make sure Make New Plot in the Add Waveform dialog is not checked, and with like traces checked.**
- **Double click V_4 and V_8 in the vector list to plot two waveforms in one plot.**

V_4 and V_8 plots





To separate the graphs into two new plots:

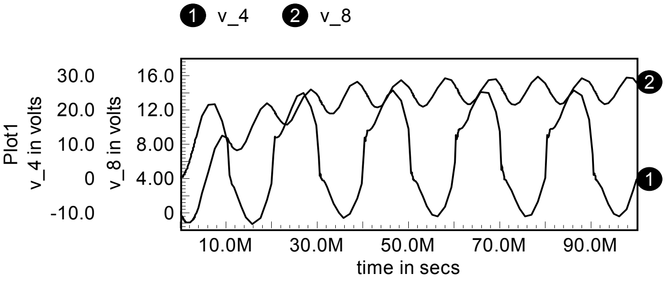
- **Click on any part of the v_8 waveform (not the waveform ID tag).**
- **Drag the cursor outside of the current plot. When the cursor changes from**

  ⊘ **to** ↖ **release the mouse button.**

To combine two separate graphs into one plot:

- **Click any part of either waveform (not the waveform ID tag).**
- **Drag the cursor from the current plot to the plot where you desire the waveform.**
- **Release the mouse to add the waveform when the cursor changes from**

  ⊘ **to** ↖



Copying, Deleting and Renaming Waveforms

- **Select a waveform by clicking any part of the waveform, except the ID tag.**

↕ *Move waveform up or down.*

The waveform ID tag ❷ should be filled in when selected.

To Copy A Waveform:

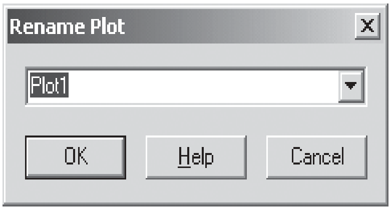
- **Hold down the Ctrl key while dragging and dropping the waveform to another plot.**
- **Release the mouse when the cursor changes from ⊘ to ↖ .**

⊘ *Waveform cannot be dropped here.*

To Delete A Waveform:

- **Press the Delete Key to delete the selected waveform.**

✥ *Select waveform movement.*



Rename Plot

Plot1

↔ *Move waveform left or right.*

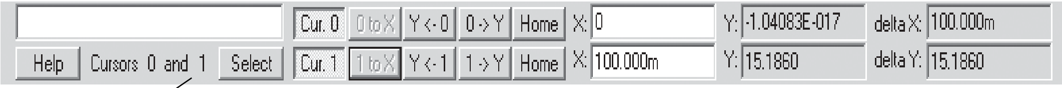
To Rename A Plot:

- **Select the Edit menu, Rename Plot.**
- **Or, double click on its label on the left of the plot, such as Plot1.**
- **Type a desired name in the Rename Plot dialog.**

🔍 *Zoom in on an area of a waveform.*

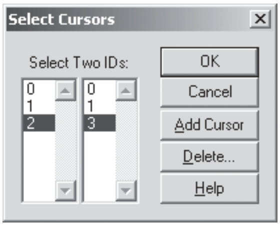
Cursor Control

The Cursor Control bar helps you to control the two cursors assigned to it. It is located at the lower left hand corner of the IntuScope screen. Cursors provide numerical information.



Cursor Control bar

To turn Cursor Control on:



- **Select the Windows menu and check Cursor Control, or press (<Ctrl>+K) to toggle Cursor Control ON if it is not already ON.**
- **Select the Calculator menu, Cursors. Check Show Default Cursors, then check Show Non-Default Cursors.**
- **Click Select in the Cursor Control bar toward the bottom of IntuScope.**
- **Click on the Add Cursor button twice to add Cursors 2 and 3.**
- **Highlight 2 and 3 to activate Cursor 2 and Cursor 3.**
- **Click OK.**

Now Cursor 0 and 1 have been replaced by Cursor 2 and 3 in the Cursor Control Bar as the active cursors. The X, Y, and Delta values are shown for the new active cursors.

- **Click Cursor 2 to hide it. Click it again to show it.**

There are several ways to move cursors along the waveforms:

To move cursors with the mouse:

- **Click on either cursor control box in your open graph.**

- **Hold and drag the mouse to reposition the cursor.**

To move cursors by assigning X, Y values:

- **Enter an X or Y value in the accumulator field above the HELP button in the Cursor Control Bar.**

- **Click the “2 to X” or “3 to X” buttons to assign your value. to the corresponding cursor. The cursor will move to the position of the value entered.**

To move cursors one data point at a time to the left or right:

- **Empty the accumulator field.**
- **Click the Y < # (# means cursor #), or # > Y button.**

To switch cursors from one waveform to another within the same plot and in the same units:

- **Click in the Cursor box** (**Note:** the empty red box with the small red horizontal line that intercepts the waveform). Arrows with four directions and a black horizontal line should appear.
- **Move the black horizontal line up or down to other waveforms within the same plot. Release the mouse.**
- **Click the Home button in the Cursor Control bar to reset the cursors one by one.**
- **OR press the Home key on the keyboard to reset both cursors at once.**

Hiding and Selecting A Waveform

Several waveforms may be displayed in one plot and overlap each other.

To hide an unwanted waveform:

- **Double click on its I.D. tag (# on far right of waveform) to make it invisible.**
- **Double click on the tag to re-display the waveform.**

To select the desired waveform:

- **Click on the preferred waveform or its I.D tag.**

The position of I.D tags is also adjustable:

- **Click on the desired waveform I.D tag.**
- **Hold the Alt key down and drag the tag left or right along the waveform.**

Translating and Scaling A Waveform

Once a waveform is plotted, IntuScope can translate, move, and scale a waveform.

To Translate a Waveform:

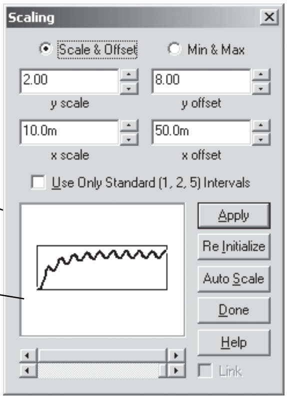
- **Click on a waveform's I.D tag and hold down the left mouse button. Drag the waveform up or down until it is in the desired position.**
- **Or, select a waveform and press the up arrow or down arrow keys to move the waveform up or down.**

To Scale a Waveform:

- **Click on a waveform's I.D tag.**
- **Press and hold down the Shift Key and the up arrow to enlarge or down arrow to shrink.**

Working With The Scaling Window

Scaling dialog is used to view and alter the scaling of the active waveform.



Scaling

● Scale & Offset ○ Min & Max

y scale: 2.00

y offset: 8.00

x scale: 10.0m

x offset: 50.0m

□ Use Only Standard (1, 2, 5) Intervals

□ Link

Scaling dialog

Picture field

- **There are two ways to bring up the Scaling dialog.**

  1) Select the Window menu, then Scaling.
  2) Select the Scaling menu, then Scaling Dialog.

- **Hold down the Shift key while clicking directly on desired waveforms to add multiple waveforms for group scaling.**

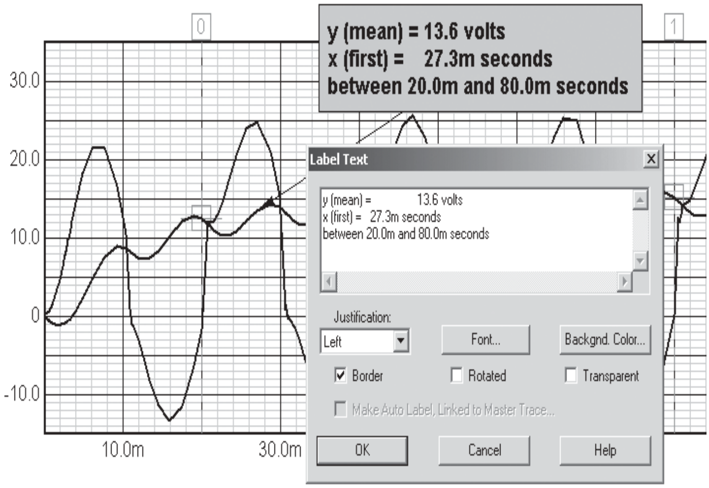
Note: Both the selected and non-selected traces' x -axis scales will be changed. They all share the same x-axis.

Labeling and the Waveform Calculator

IntuScope provides a wealth of waveform calculator functions (reference the Calculator menu pulldown). Amongst these, labeling enables you to label typical values such as average, peak to peak and minimum, directly on graphs.

- **Click on the home button for both of the active cursors.**
- **Select Calculator menu, Labels, Average. OR Right click on the Graph window and select Label, Average.**
- **Double click on the label to bring up Label Text dialog.**
- **Select Rotate and Transparent.**
- **Highlight the text and click on the Font button.**
- **Select Font Style, Bold and Font Size, 12. Click OK.**
- **Click OK to apply the changes and close the Label Text dialog.**
- **Drag and drop the Label Box to a free space in your graph.**

To label a specific point on your drawing:



y (mean) = 13.6 volts
x (first) = 27.3m seconds
between 20.0m and 80.0m seconds

Label Text

y (mean) = 13.6 volts
x (first) = 27.3m seconds
between 20.0m and 80.0m seconds

Justification: Left

☑ Border ☐ Rotated ☐ Transparent

☐ Make Auto Label, Linked to Master Trace...

- **Select Calculator menu, Labels and Trace Comment.** The circle end point can be dragged along the waveform.
- **Select Options, Save Preferences to save your settings.**

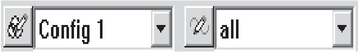
Note: Labels can be placed by selecting Edit menu, Text Label; Calculator menu, Labels, or by right-clicking in the graph.

Working with IntuScope Scripts

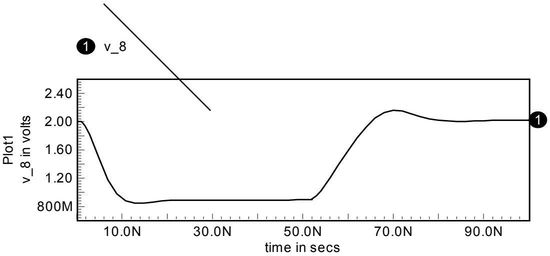
Intusoft has extended the Interactive Control Language (ICL) script commands and functions to work in IsSpice4, IntuScope and SpiceNet. These Scripts enable automatic extraction of important waveform information; ability to build and save new waveforms; compare data from previous simulations; and save results within the schematic database. The results of the simulations are automatically post-processed and made available for further analysis.

You can perform an exercise to get familiar with scripts.

- **Open Sample.DWG circuit in the following directory: :spice8\Circuits\Demo\Sample.DWG**
- **Set configuration and simulation setup drop-down boxes, respectively to "Config1" and "all" on the top schematic toolbar.**


Config 1
all

V_8 graph



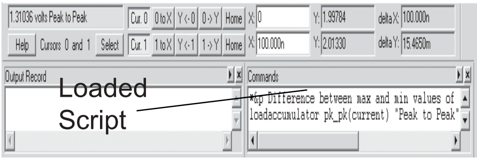
- **Click the Run Simulation icon.**
- **Select the toolbar icon, Scope.**
- **In IntuScope, double click on V_8 and choose default for the X Axis.**

- **In IntuScope's File Menu, select Load Script.**
- **In the Load Command Script Dialog, choose the Measurements folder and Peak to Peak.scp.**

If you don't see the same script file, use its path to browse: :spice8\In\scripts\measurements\Peak to Peak.scp The Script will be loaded in the Commands Window at the lower-right corner of the IntuScope program.

- **Press <Ctrl>+R to display the result.**

The result will be displayed in IntuScope's accumulator field above the Help button.



1.31036 volts Peak to Peak

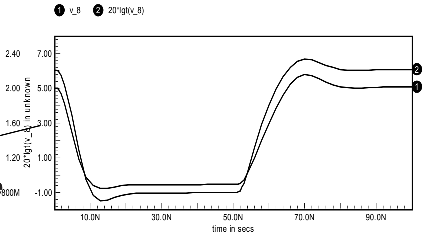
Cursors 0 and 1

X: 0 Y: 1.99784 delta X: 100.000n

X: 100.000n Y: 2.01330 delta Y: 15.4650m

```
*&p Difference between max and min values of
loadaccumulator pk_pk(current) "Peak to Peak"
```

Loaded Script



*The plot of 20*lgt(V_8) vs. time*

- **Click the Calculator menu, Functions, 20 Lgt.**

IntuScope has many useful scripts commonly used. You may select the scripts from the Calculator menu as desired. You can create or modify a script in IntuScope's Command Window for further analysis. For additional information on scripts, including format, syntax and dozens of examples, go to <http://www.intusoft.com/in/script/ScopeIndex.html>.

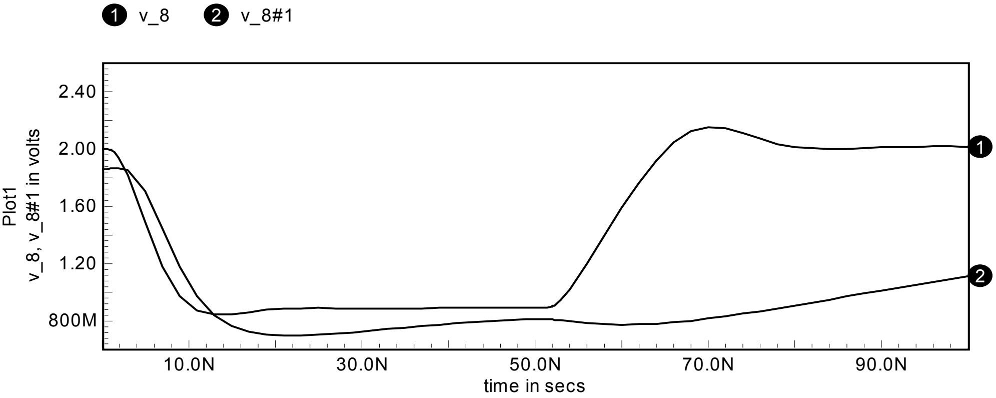
Updating A Waveform

IntuScope enables you to update current waveforms and make comparisons between the new and previous ones: e.g., after changing a component's parametric value.

Let's use the same circuit, Sample.DWG, to do the following exercise.

- **Select New Graph from the File Menu.**
- **In IntuScope's Add Waveform dialog, select Test Pts Only in the Show section.**
- **Click on V_8 to display the V_8 waveform.**
- **In the Action Menu, select Update, Add Updated Traces (Note two other choices above and below this one).**
- **Switch IntuScope back to the SpiceNet window.**

- **In the SpiceNet schematic, change the R18 value from 1K to 10K.**
- **Click on the Run Simulation icon.**
- **Switch the window back to IntuScope from IsSpice4 by pressing the ALT +TAB key, or IntuScope's desktop icon.**
- **Select the Actions menu, then Update. Use Selected Mode or (<Ctrl>+U) to display the updated waveforms.**



The new V_8 waveform will be displayed in the same plot with the old one as shown above. You can easily compare and analyze how different R18 values effect the behavior of V_8.

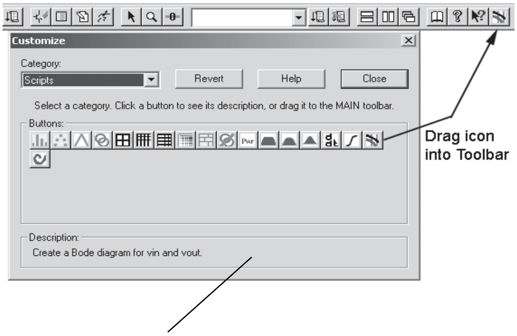
Make sure the desired update method is checked before you update. Check "Update Current Document" to replace old waveforms with new ones when you update. Check "Add Updated Traces" to display new waveforms next to previously plotted waveforms. Finally, check "Add Updated Document" to plot new waveforms in a separate graph.

You can have the update happen during mode change by checking "Update on Mode Change."

You can also toggle "Use Original Analysis Type" or "Use Selected Analysis Type." "Use Selected Analysis Type" is useful for multi-pass simulations since you can plot a group of waveforms for one analysis type, change to another analysis type, and update all existing waveforms based on currently selected analysis type. Choose Save Preferences under the Options Menu to make these update methods the default setting for every time you open IntuScope.

Customizing The IntuScope Toolbar

- **In the IntuScope Options Menu, check Show Toolbar.**
- **Select Customize Toolbar.**
- **Select Category, Scripts.**



Category: Scripts

Select a category. Click a button to see its description, or drag it to the MAIN toolbar.

Description: Create a Bode diagram for vin and vout.

To add an Icon to the toolbar:

- **Drag the bode plot button from the Customize dialog to the toolbar.**

The destination will be marked by a vertical line.

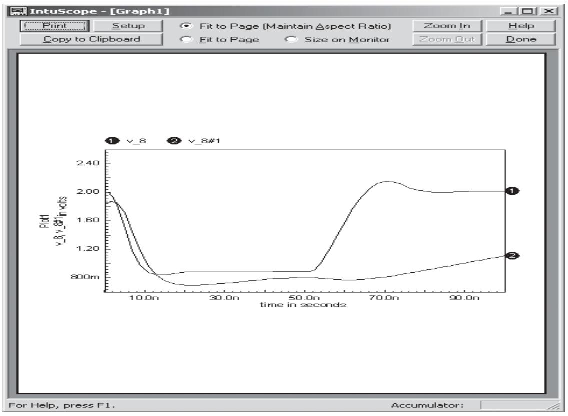
Description of the chosen icon.

To Remove an icon:

- **Drag the Bode plot icon off the toolbar onto the open Customize dialog.**

Print Preview

- **Select File menu, Print to display the print preview window.**



● Fit to Page (Maintain Aspect Ratio)
○ Fit to Page
○ Size on Monitor

For Help, press F1.

Accumulator:

- **Click on the Copy to Clipboard button to place the image on preview page into the system clipboard as a Windows Meta File (.WMF).**

Enter exact values within the bounding box or just drop and drag it to manipulate the size of your graph.

Encapsulated Postscript File (.EPS file) may also be supported depending on your printer. The clipboard allows IntuScope graphs to be imported into most word processing or desktop publishing programs.

- **Click on Print button to print everything shown on the graph except cursors.**

Working With Smith Charts

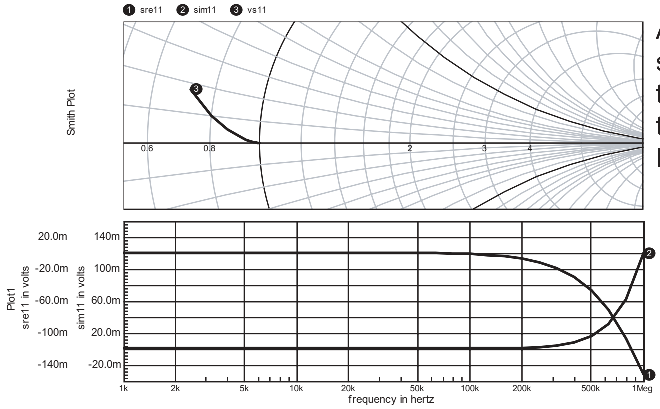
The Smith Chart function built into IntuScope enables you to determine the effects of altering network geometry during a Two-port network analysis.

Let's get familiar with Smith Charts by doing an exercise.

- **Open the TwoPort.DWG circuit with the following path :spice8\Circuits\Networks\TwoPort.DWG**
- **Set the Configuration and Simulation setups to be Sparams and 1k-1meg respectively.**
- **Click the Simulation icon.**
- **From the IsSpice4 Actions menu, select Scope.**
- **From Scope, select Calculator, Smith Chart, vs11**

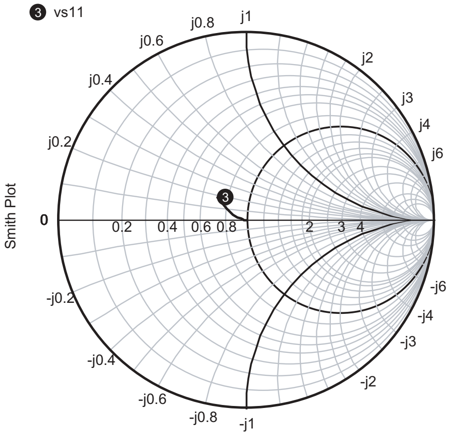
You will see a graph plotted similar to the one on the following page.

From the plots you can see how the real part (waveform1) and an imaginary part (waveform2) of S11 behave within a defined frequency range, from 1kHz to 1MegHz. You may also be able to see how vs11 moves in a Smith Chart at this frequency range.

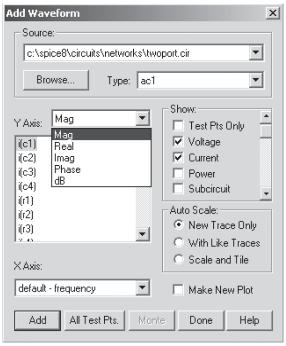


A Smith Chart and a plot of the sre11 and sim11 waveforms are the result of selecting vs11 in the Smith Chart submenu of IntuScope's calculator.

To View the full Smith Chart:



- **Click the Semi-Smith Chart.**
- **Hold Down the Shift key while clicking on the Auto Scale button inside the Scaling dialog to display a full Smith Chart.**



Add Waveform

Source:

c:\spice8\circuits\networks\twoport.cir

Browse... Type: ac1

Y Axis: Mag

Mag
Real
Imag
Phase
dB

i(c1)
i(c2)
i(c3)
i(c4)
i(r1)
i(r2)
i(r3)

Show:
□ Test Pts Only
☑ Voltage
☑ Current
□ Power
□ Subcircuit

Auto Scale:
● New Trace Only
○ With Like Traces
○ Scale and Tile

X Axis: default - frequency

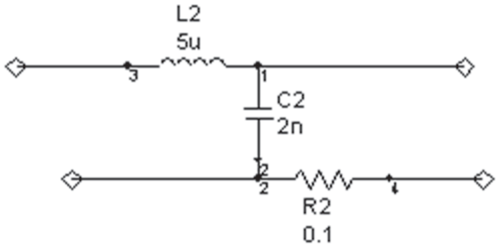
□ Make New Plot

Add | All Test Pts. | Monte | Done | Help

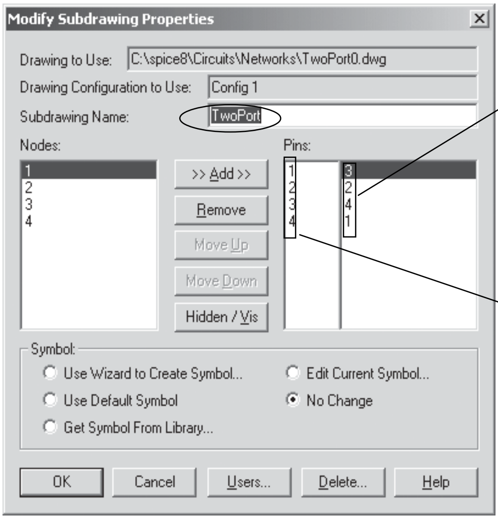
IntuScope enables you to choose the type of waveforms displayed in the pulldown menu under the Y Axis section.

You are also able to create or modify the circuit inside of a TwoPort network box.

- **In SpiceNet, select File menu, Open.**



- **Open :\spice8\Circuits\Networks\TwoPort0.DWG**
- **Change C2 2n value to 5n.**
- **Select Subdrawings menu, Modify Subdrawing.**
- **Subdrawing Name must be TwoPort to replace the TwoPort part in TwoPort.DWG.**
- **Nodes 3, 2, 4 and 1 on the schematic above are nodes 1, 2, 3, and 4 on the created box symbol. How this works is that we select the schematic nodes on the left and add it to the pin list on the right going in a counterclockwise direction, starting from the top left corner of the box symbol.**



Drawing to Use: C:\spice8\Circuits\Networks\TwoPort0.dwg
Drawing Configuration to Use: Config 1
Subdrawing Name: TwoPort

Nodes:
1
2
3
4

Pins:

1	3
2	2
3	4
4	1

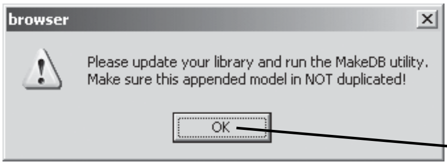
Symbol:
○ Use Wizard to Create Symbol...
○ Edit Current Symbol...
○ Use Default Symbol
● No Change
○ Get Symbol From Library...

Schematic Nodes of the modified drawing

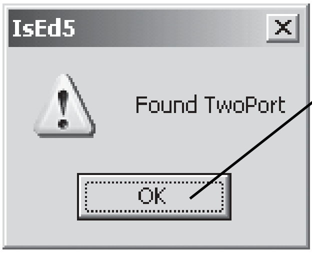
Nodes named for the TwoPort network box symbol.

- **For Symbol, select “No Change” then OK.**
- **Select File menu, Export.**
- **Choose SubCkt from the pulldown menu and click OK.**

For users who don’t have Library Manager:



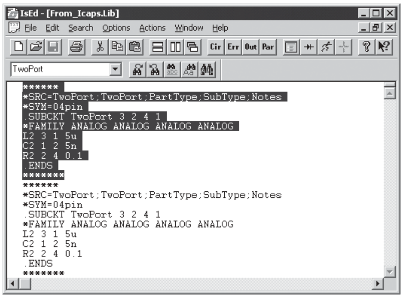
Please update your library and run the MakeDB utility.
Make sure this appended model in NOT duplicated!



Found TwoPort

- **Click OK for both message dialogs.**

Now you will see a IsEd-From_Icaps.Lib dialog. Every time you export SubCkt, the exported model will be added to the bottom.

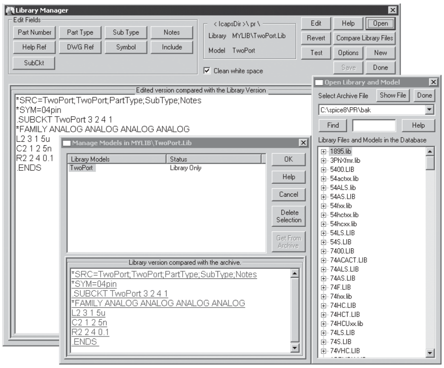


```
******
*SRC=TwoPort;TwoPort;PartType;SubType;Notes
*SYM=04pin
.SUBCKT TwoPort 3 2 4 1
*FAMILY ANALOG ANALOG ANALOG ANALOG
L2 3 1 5u
C2 1 2 5n
R2 2 4 0.1
.ENDS
*******
******
*SRC=TwoPort;TwoPort;PartType;SubType;Notes
*SYM=04pin
.SUBCKT TwoPort 3 2 4 1
*FAMILY ANALOG ANALOG ANALOG ANALOG
L2 3 1 5u
C2 1 2 5n
R2 2 4 0.1
.ENDS
*******
```

- **Highlight the top duplicate TwoPort netlist and delete it.**
- **Save the changes**
- **In SpiceNet select Update Part Database... from the file menu.**

For users who have Library Manager:

- **Library Manager will be automatically activated once you click OK in the SubCkt dialog.**
- **If the library listing does not exist, give it a name: TwoPort.lib**
- **Click Yes to rebuild the database.**
- **Click Save to save the new netlist displayed in the Library Manager window.**



< IcapsDir >\ pr \

Library MYLIB\TwoPort.Lib

Model TwoPort

☑ Clean white space

Edited version compared with the Library Version

```
*SRC=TwoPort;TwoPort;PartType;SubType;Notes
*SYM=04pin
.SUBCKT TwoPort 3 2 4 1
*FAMILY ANALOG ANALOG ANALOG ANALOG
L2 3 1 5u
C2 1 2 5n
R2 2 4 0.1
.ENDS
```

Library Models	Status
TwoPort	Library Only

Library version compared with the archive.

```
*SRC=TwoPort;TwoPort;PartType;SubType;Notes
*SYM=04pin
.SUBCKT TwoPort 3 2 4 1
*FAMILY ANALOG ANALOG ANALOG ANALOG
L2 3 1 5u
C2 1 2 5n
R2 2 4 0.1
.ENDS
```

Select Archive File C:\spice8\PR\bak

Library Files and Models in the Database

1895.lib
3PhXfmr.lib
5400.LIB
54actxx.lib
54ALS.lib
54AS.LIB
54fxx.lib
54hctxx.lib
54hcxx.lib
54LS.LIB
54S.LIB
7400.LIB
74ACACT.LIB
74ALS.LIB
74AS.LIB
74F.LIB
74fxx.lib
74HC.LIB
74HCT.LIB
74HCUxx.lib
74LS.LIB
74S.LIB
74VHC.LIB

For both types of users:

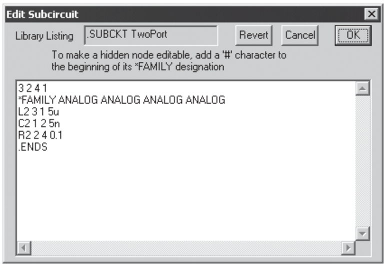
- **Close TwoPort0.dwg**
- **Open :\spice8\circuits\networks\TwoPort.dwg**
- **Select Update Cache under the Parts menu.**
- **Click Apply button to update the netlist information**

The old netlist inside of TwoPort network box has been replaced by the newly created one.

The changes made can be verified as follows:

- **In SpiceNet, double click on one of the TwoPort symbols in the schematic to bring up its Properties dialog.**
- **Select the Value field of .SUBCKT and click the Enter button.**

You will see the netlist of TwoPort network box shown in the Edit Subcircuit dialog.

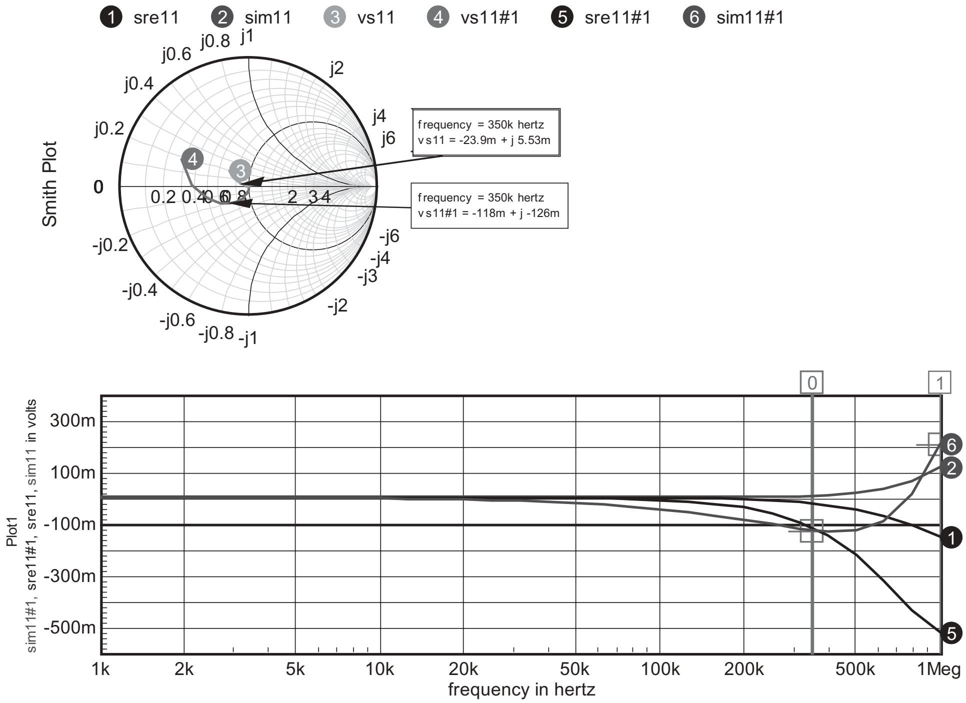


Library Listing .SUBCKT TwoPort

To make a hidden node editable, add a '#' character to the beginning of its *FAMILY designation

```
3 2 4 1
*FAMILY ANALOG ANALOG ANALOG ANALOG
L2 3 1 5u
C2 1 2 5n
R2 2 4 0.1
.ENDS
```

- **Run another simulation with Configuration and Simulation setups still set to Sparams and 1k-1meg respectively.**
- **Change to the IntuScope window.**
- **In IntuScope, press <ctrl>+u to update the waveforms.**
- **Select Plot1 at the bottom, right mouse click and check Link Master Trace.**
- **Select waveform 3 in the Smith Plot, right mouse click and check Auto Label, Cursor 0.**
- **Select waveform 4 in the Smith Plot, right mouse click and check Auto Label, Cursor 0.**
- **Move Cursor 0 in Plot1 below and the Auto Labels above will update depending on where Cursor 0 is moved to.**

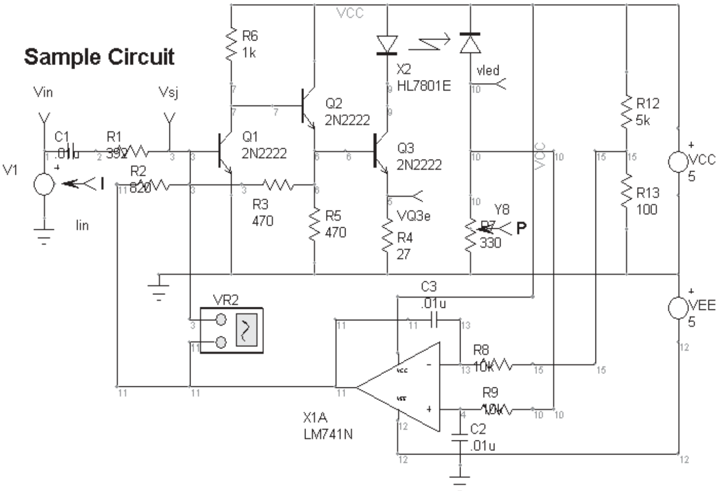


Working with SendScript

ICAP/4's SendScript feature enables users to optionally send an ICL script either from SpiceNet or IsSpice4, directly to IntuScope. The script is automatically run in the IntuScope waveform analyzer.

Let's do an exercise to get familiar with this feature.

- **Open Sample.DWG. The path is :\spice8\sn\Sample2\Sample.DWG.**



- **Select Closed Loop and Tran for Configuration and Simulation Setup, respectively.**

Closed loop
Tran

- **Click the Run Simulation icon to run the simulation.**

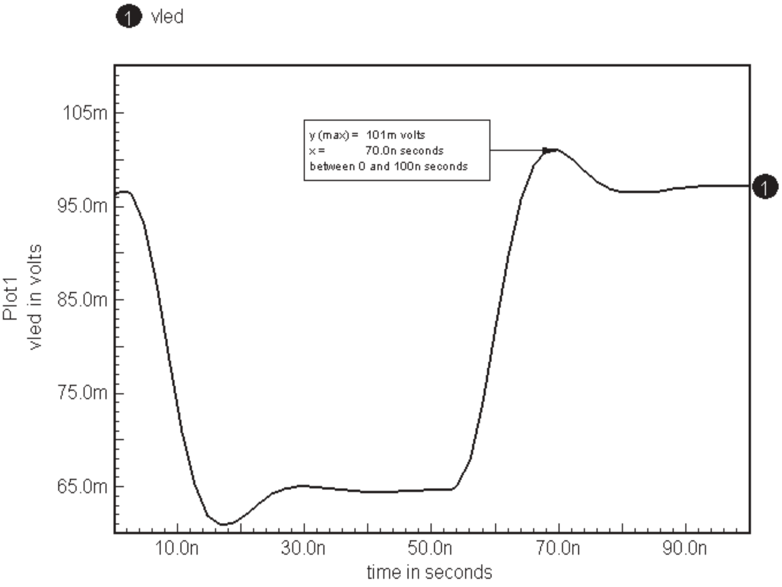
- **Press (<Ctrl>+T) to autoscale the waveforms.**

- **From the Actions menu, select Scope.**

- **In the Add Waveform dialog, select Test Pts Only under the Show field.**

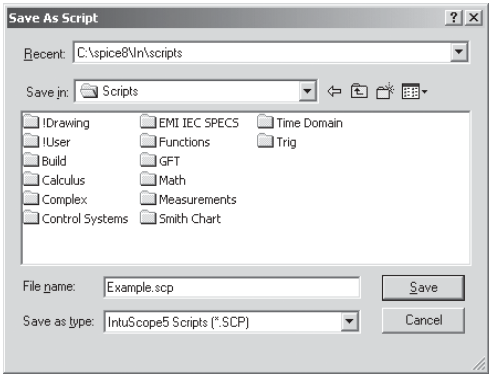
- **Double click on the Vled under the Y Axis to plot the waveform.**

- **From the Calculator menu, select Labels/Maximum to label the maximum point on the waveform.**



- **From the File menu, select <Save As Script> to create the script for actions you preformed.**

- **Save the script file as Example.SCP.**


Save As Script

Recent: C:\spice8\In\scripts

Save in: Scripts

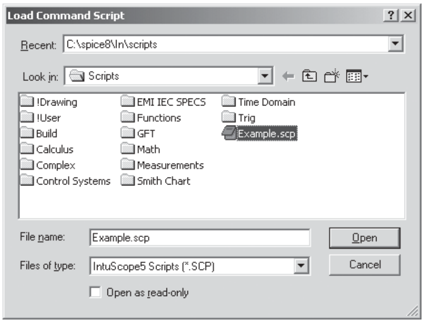
File name: Example.scp

Save as type: IntuScope5 Scripts (*.SCP)

- **Click Save to close the dialog.**

- **From the File menu, select Load Script.**

- **Select and Open Example.SCP in the pop-up dialog.**



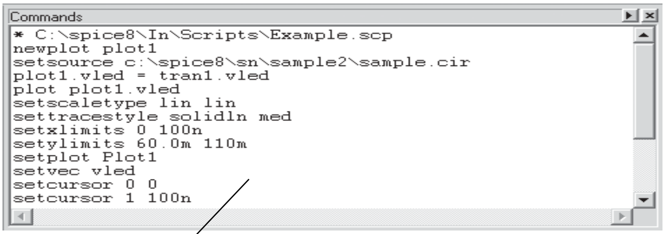
Recent: C:\spice8\In\scripts

Look in: Scripts

File name: Example.scp

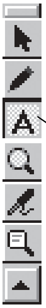
Files of type: IntuScope5 Scripts (*.SCP)

□ Open as read-only



```
* C:\spice8\In\Scripts\Example.scp
newplot plot1
setsource c:\spice8\sn\sample2\sample.cir
plot1.vled = tran1.vled
plot plot1.vled
setscaletype lin lin
settracestyle solidln med
setxlimits 0 100n
setylimits 60.0m 110m
setplot Plot1
setvec vled
setcursor 0 0
setcursor 1 100n
```

Now the example script has been loaded into the Commands window.



Text icon

- **Highlight the script in the Commands window and press (<Ctrl>+C) to copy to clipboard.**

- **Switch back to the SpiceNet window.**

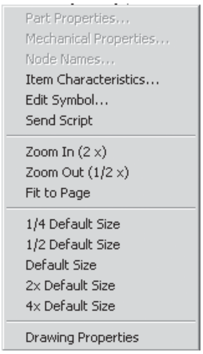
- **Click on the Text icon on the right toolbar.**

- **Put the cursor in a free space in the schematic and press (<Ctrl>+V) to paste the script into the schematic.**

- **Type Scope in the first line to send the script to IntuScope.**

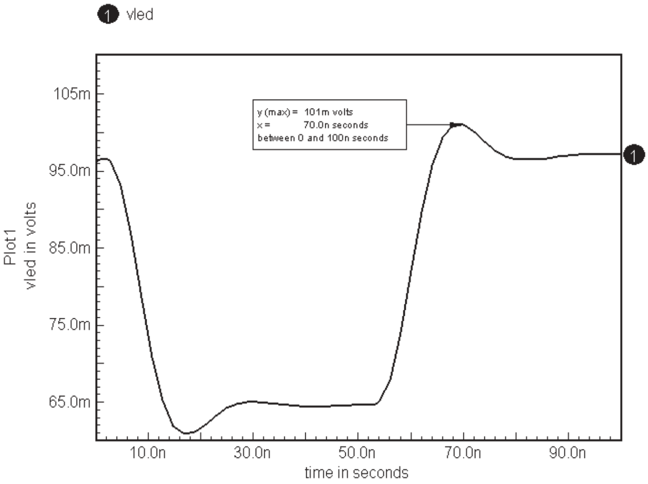
```
Scope
* D:\spice8\In\Scripts\Example.SCP
newplot plot1
setsource d:\spice8\sn\sample2\sample.cir
plot1.vled = tran1.vled
plot plot1.vled
setscaletype lin lin
settracestyle solidln med
setxlimits 0 100n
setylimits 60.0m 110m
setplot Plot1
setvec vled
setcursor 0 0
setcursor 1 100N
setlabeltype plaintext
makelabel max
setlabel border true
setlabel justify left
setlabel rotated false
movelabel 0 0
```

- **Press <Esc> key to go back to Normal Mode.**
- **With the mouse cursor over the text block, click the right mouse button and select Send Script.**



The script you just created in SpiceNet has been sent to IntuScope and run in the waveform analyzer.

- **Switch to the IntuScope window. The Vled graph has been plotted as shown below.**

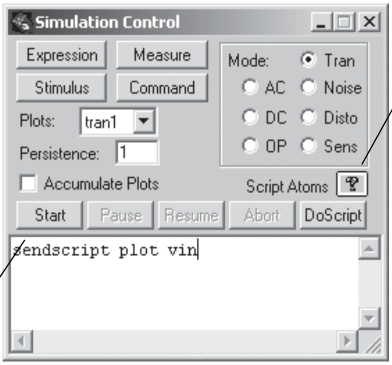


The graph is the same as the one plotted without using SendScript.

- **Switch from the IntuScope window to the IsSpice4 window.**

- **Enter a different script as shown below, which plots Vin in the simulation control dialog.**

Note: *Always keep IntuScope activated before sending scripts.*



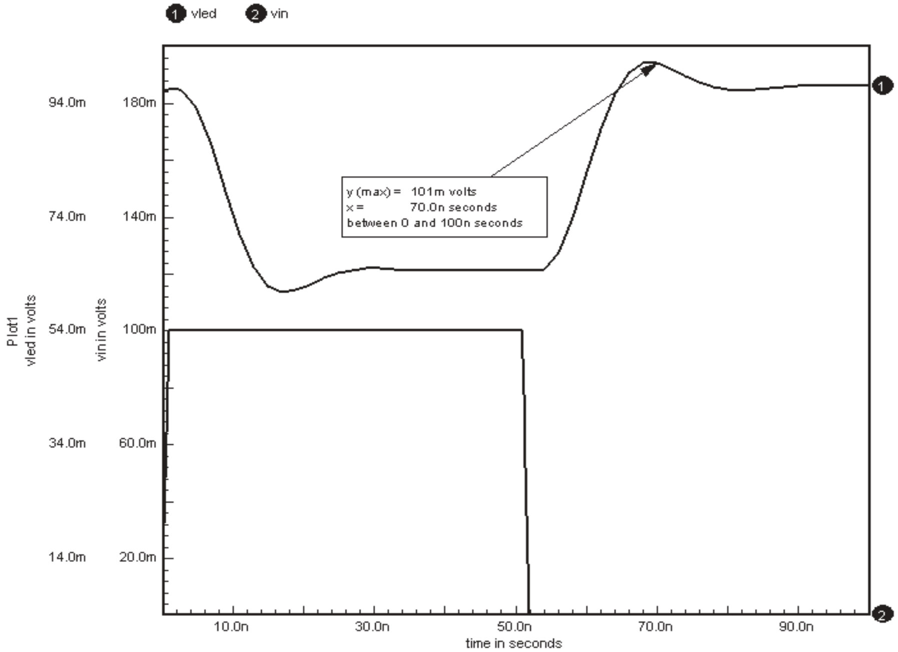
```
sendscript plot vin
```

Click button for help on script syntax and reference

sendscript plot vin

- **Click on the DoScript button to run the script in IntuScope.**
- **In IntuScope, drag and drop the label to a desired position.**

The graph looks similar to the one shown.



Quitting IntuScope

Let's exit out of the IntuScope program

- **Select File menu, Exit/Quit.**
- **Select Save to begin the graph saving process or select "No" to discard the graphs and exit IntuScope.**

To quit SpiceNet and IsSpice4:

- **Select Exit/Quit from the File menu in SpiceNet.**

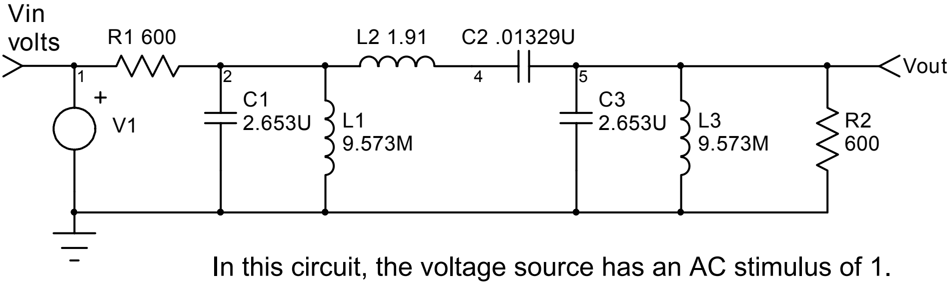
Congratulations, you have completed the IntuScope tutorial. IntuScope contains a tremendous number of capabilities, only a few of which have been discussed here. The vast array of functions are accessible through the pulldown menus, and are explained in the in-application help system.

Tutorial #6 - Monte Carlo, RSS Analysis and Optimization

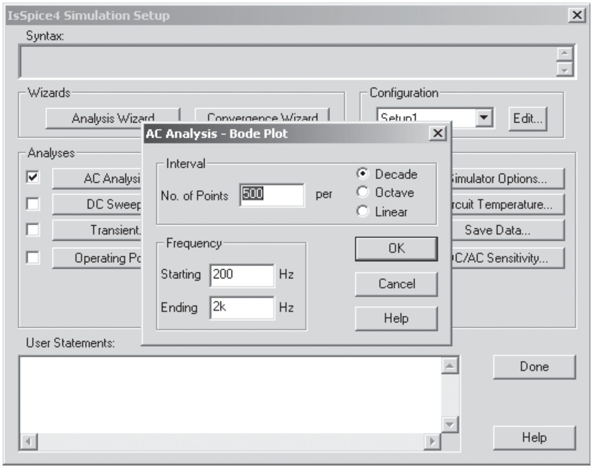
This tutorial focuses on the Monte Carlo and Root Summed Square (RSS) analysis capabilities of ICAP/4. A third-order Butterworth approximation of a passive bandpass filter is used for the analysis. To proceed, you must be familiar with all of the programs in the ICAP/4 package. For a more in-depth discussion of the Monte Carlo analysis procedures, please refer to the IsSpice4 User's Guide.

Setting Up A Design for Monte Carlo Analysis

- **Open the BUTTFIL1.DWG drawing using the :\spice8\Circuits\butfil\BUTTFIL1.DWG path**



In this circuit, the voltage source has an AC stimulus of 1.



Configuration: Setup1

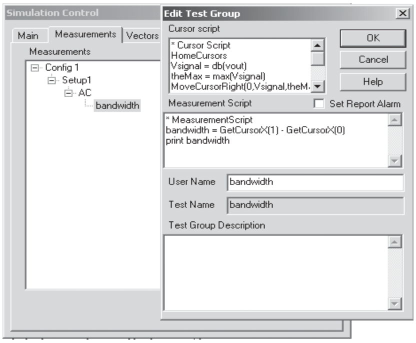
☑ AC Analysi
□ DC Sweep
□ Transient
□ Operating Po

No. of Points 500 per ● Decade ○ Octave ○ Linear

Starting 200 Hz

Ending 2k Hz

- **In SpiceNet, select Actions, Simulation Setup, Edit.**
- **In SpiceNet's IsSpice4 Simulation Setup dialog, click AC analysis.**
- **In the AC Analysis Bode Plot dialog, enter (if not entered) 500 points per decade between 200 Hz and 2kHz.**
- **Click OK to close AC Analysis - Bode Plot dialog.**
- **Click Done to close IsSpice4 Simulation Setup dialog.**



```
* Cursor Script
HomeCursors
Vsignal = db(vout)
theMax = max(Vsignal)
MoveCursorRight(0,Vsignal,theM
```

```
* MeasurementScript
bandwidth = GetCursorX(1) - GetCursorX(0)
print bandwidth
```

User Name: bandwidth

Test Name: bandwidth

- **From SpiceNet's Actions menu, select Actions, ICAPS. Or, click on the ICAPS icon (blue/white square) to get the Simulation Control dialog.**
- **Select the Measurements Tab.**
- **Select "bandwidth" in the Measurement Tree.**
- **Click on Edit Test Group button to see the Measurement Script.**

```
*MeasurementScript
bandwidth = GetCursorX(1) -
GetCursorX(0)
print bandwidth
```

```
*Cursor Script
HomeCursors
Vsignal = db(vout)
theMax = max(Vsignal)
MoveCursorRight(0,Vsignal,theMax)
Fmax = GetCursorX(0)
SetCursor(1,Fmax)
MoveCursorRight(1,Vsignal,theMax-3)
MoveCursorLeft(0,Vsignal,theMax-3)
```

- **Click the Cancel button to dismiss the Edit Test Group dialog.**

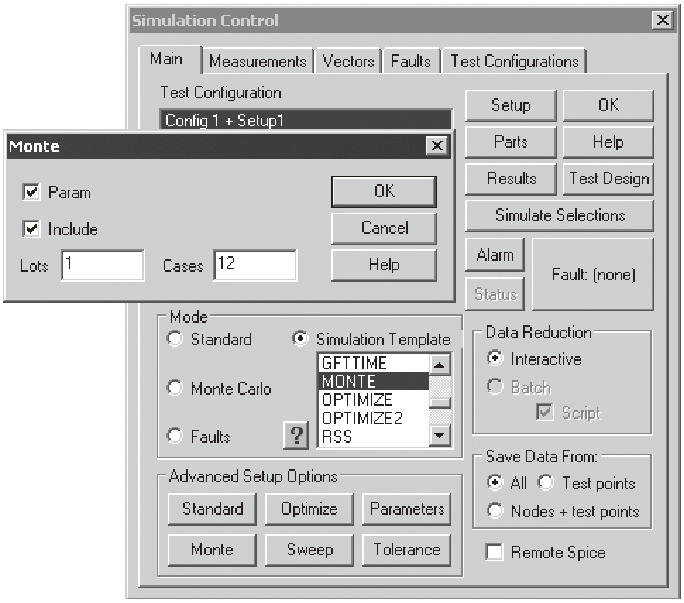
In order to run a Monte Carlo analysis, tolerance values need to be specified. These tolerances will be used to vary the component values during the Monte Carlo analysis. Three tolerance values are chosen for this simulation:

Resistors	**1% tolerance**
Inductors	**5% tolerance**
Capacitors	**10% tolerance**

All these setups should be ready for you once you open the circuit. However, if they are not, you are able to enter them yourself by double clicking on a passive component, going to the tolerance/sweep/optimize tab, entering the desired tolerance, and click apply tolerance to all similar parts button.

Monte Carlo Simulation

- **Select the Main tab in the Simulation Control dialog.**



Simulation Control

Test Configuration: Config 1 + Setup1

Monte

☑ Param

☑ Include

Lots 1 Cases 12

Mode: ○ Standard ● Simulation Template ○ Monte Carlo ○ Faults

GFTTIME, MONTE (selected), OPTIMIZE, OPTIMIZE2, RSS

Fault: (none)

Data Reduction: ● Interactive ○ Batch ☑ Script

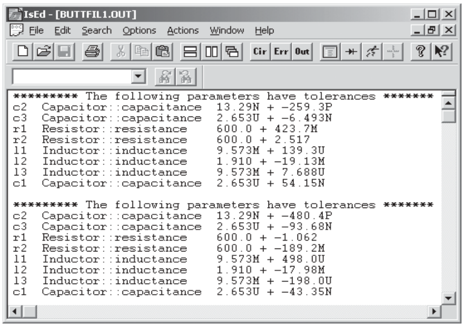
Save Data From: ● All ○ Test points ○ Nodes + test points

□ Remote Spice

- **In the Simulation Control dialog's Advanced Setup Options section, click on the Monte button.**
- **Enter a Lots value of 1, and a Cases value of 12, then click on OK.**
- **Click on Monte in the list to the right of the '?' button. This Monte Carlo enables post processing in IntuScope and the plotting of histogram or probability plots. Any "Measurements" established from the Simulation Control's "Measurements" tab will be recorded in the "Results" box, but "Mean" and "3-Sigma" data is not recorded. This statistical data is obtained from the slower running "Monte" radio button selected on the left. This non simulation-template Monte does not produce statistical waveform data in IntuScope.**
- **Click on the Simulate Selections button to launch the Monte Carlo simulation.**

This will invoke 12 Monte Carlo cases, or runs. The list of simulations will appear in the Output window. You can view any test point waveform for any simulation case by selecting the case in the Plots popdown list.

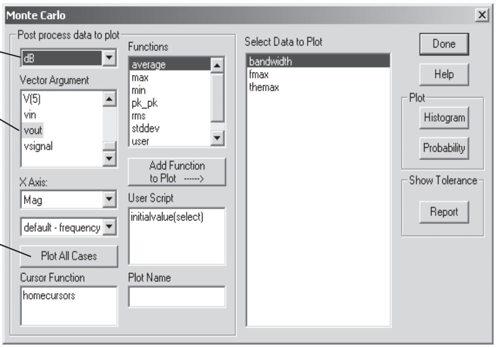
- **In IsSpice4’s Action menu, select Text Edit.**
- **Click on Output file icon** Out **to view the output file.**



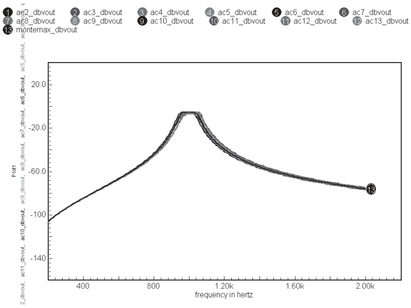
```
********* The following parameters have tolerances *******
c2  Capacitor::capacitance  13.29N + -259.3P
c3  Capacitor::capacitance  2.653U + -6.493N
r1  Resistor::resistance    600.0 + 423.7M
r2  Resistor::resistance    600.0 + 2.517
l1  Inductor::inductance    9.573M + 139.3U
l2  Inductor::inductance    1.910 + -19.13M
l3  Inductor::inductance    9.573M + 7.688U
c1  Capacitor::capacitance  2.653U + 54.15N

********* The following parameters have tolerances *******
c2  Capacitor::capacitance  13.29N + -480.4P
c3  Capacitor::capacitance  2.653U + -93.68N
r1  Resistor::resistance    600.0 + -1.062
r2  Resistor::resistance    600.0 + -189.2M
l1  Inductor::inductance    9.573M + 498.0U
l2  Inductor::inductance    1.910 + -17.98M
l3  Inductor::inductance    9.573M + -198.0U
c1  Capacitor::capacitance  2.653U + -43.35N
```

- **In IsSpice4’s Action menu, select Scope (or with icon in SpiceNet).**
- **In IntuScope’s Window menu, click Add Waveform dialog.**
- **In IntuScope’s Add Waveform dialog, Click on the Monte button.**

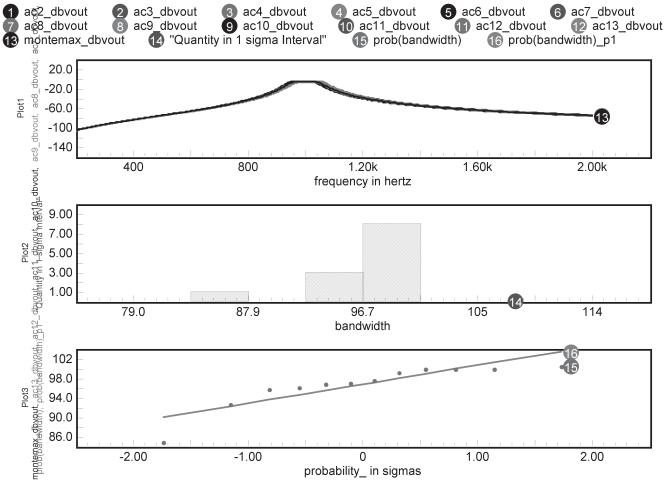
- **Select dB.**
- **Select the vout Vector Argument.**
- **Click plot all cases.**

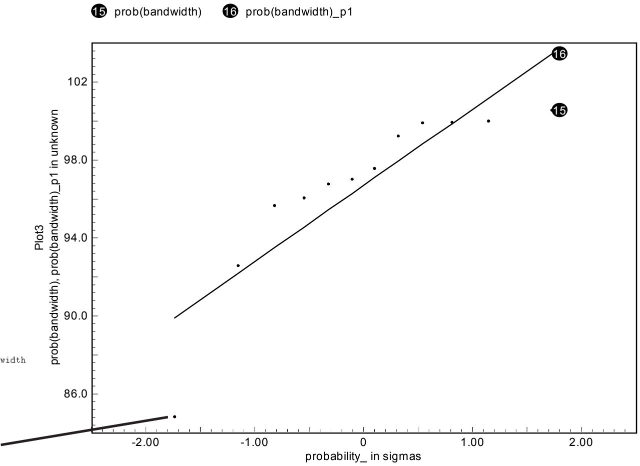


- **Under IntuScope’s Option menu select Graph Type, Linear to change to linear scale.**
- **Highlight “bandwidth” in the Monte Carlo dialog box, then select the Histogram & Probability boxes to the right.**

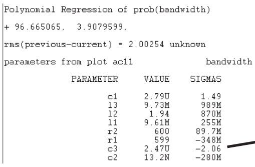


Two new graphs will be plotted.





OUTPUT RECORD



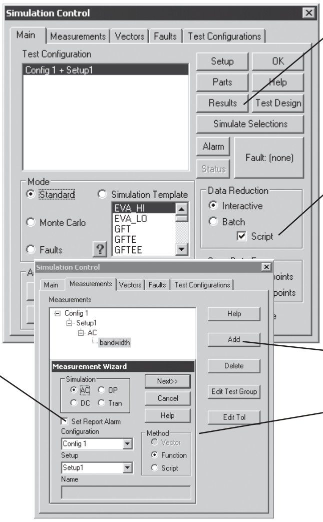
```
Polynomial Regression of prob(bandwidth)

+ 96.665065,  3.9079599,

rms(previous-current) = 2.00254 unknown

parameters from plot ac11               bandwidth

            PARAMETER    VALUE   SIGMAS

                   c1    2.79U     1.49
                   l3    9.73M     989M
                   l2     1.94     870M
                   l1    9.61M     255M
                   r2      600    89.7M
                   r1      599    -348M
                   c3    2.47U    -2.06
                   c2    13.2N    -280M
```

- **Select the Probability plot to make it active.**
- **Click on Y<- 0 and 0->Y to move cursor 0 along the datapoints of the Probability plot. Each datapoint represents results from a simulation run.**
- **Click on the Report button in the Monte Carlo Dialog's Show Tolerance Section on the right. For the simulation run that cursor 0 selects, parts with tolerance will display "value used" and "sigma value" in the Output Record field.**

Preparing for RSS Analysis

Now we will define a few circuit measurements performed automatically when the RSS simulation is run. The analysis will be performed in a standard time domain, and the measurement limits will be set using the nominal calculated values.

This overview shows you where the commands are located in the Simulation Control dialog, prior to using them in the tutorial on the next page.

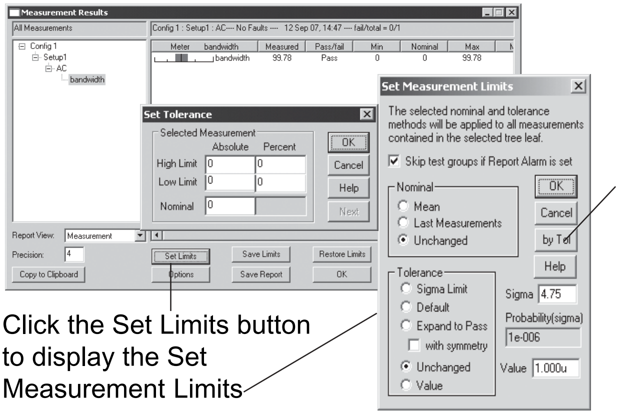


The Results button enables you to view measurement results.

If Set Report Alarm is checked, the alarm will be shown when any of the measurements are out of tolerance. It is optional.

Scripted measurement is evaluated if the Script button is checked.

Select the Add button to display the Measurement Wizard dialog.



Click by Tol to display the Set Tolerance dialog.

Click the Set Limits button to display the Set Measurement Limits dialog.

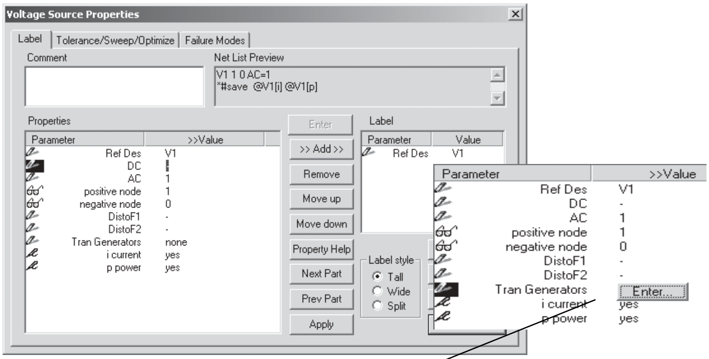
Specifying Measurements

Since we'll be making time domain measurements, we must first specify a transient time domain signal source and transient analysis request.

- **Quit the IntuScope program and discard the data.**

- **In SpiceNet, Click the OK button to close the Simulation Control Dialog.**

- **Double-click the voltage source V1 and click on the Value field for the Tran Generators parameter.**



Voltage Source Properties

Comment:

Net List Preview:

```
V1 1 0 AC=1
*#save @V1[i] @V1[p]
```

Properties

Parameter	>>Value
Ref Des	V1
DC	
AC	1
positive node	1
negative node	0
DistoF1	-
DistoF2	-
Tran Generators	none
i current	yes
p power	yes

Label

Parameter	Value
Ref Des	V1

Label style: ● Tall ○ Wide ○ Split

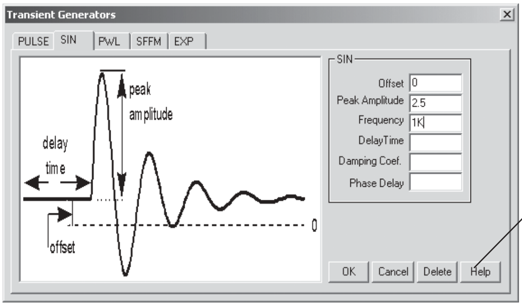
Parameter	>>Value
Ref Des	V1
DC	-
AC	1
positive node	1
negative node	0
DistoF1	-
DistoF2	-
Tran Generators	Enter...
i current	yes
p power	yes

This setting will generate a sine wave that has no DC offset, 5.0 Volts p-p, and a frequency of 1kHz.

The "Enter..." button will appear.

- **Click the Enter button to enter the Transient Generators dialog to define the characteristics of our voltage source.**

- **Click the "Sin" tab and enter 0 for Offset, 2.5 Peak Amplitude, and 1K Frequency values as shown.**



Transient Generators

SIN

Offset	0
Peak Amplitude	2.5
Frequency	1K
DelayTime	
Damping Coef.	
Phase Delay	

Click Help to view the default values.

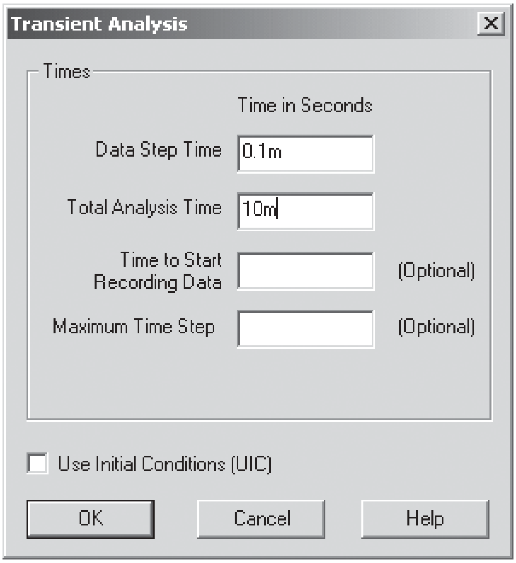
- **Click OK and OK again to close both dialogs.**

Next, we'll specify the Transient analysis request.

- **Select the Actions menu, Simulation Setup and Edit to open the Simulation Setup dialog.**

- **Click on the Transient analysis button.**

- **Enter 0.1m for the data step time and 10m for the total analysis time as shown.**



Transient Analysis

Times

Time in Seconds

Data Step Time: 0.1m

Total Analysis Time: 10m

Time to Start Recording Data: (Optional)

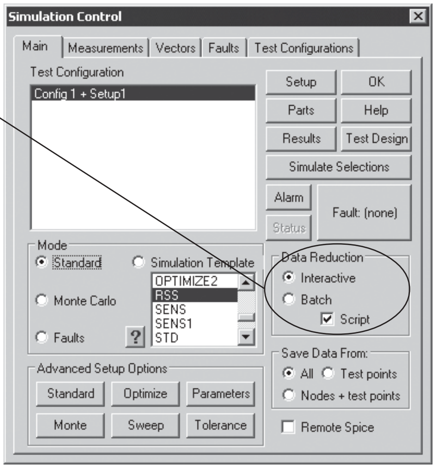
Maximum Time Step: (Optional)

☐ Use Initial Conditions (UIC)

- **Click on the OK button to close the Transient Analysis Dialog.**
- **Uncheck the AC Analysis and press the Done button to dismiss the simulation setup dialog.**

Now we're ready to specify our measurements.

- **From the Actions menu, select ICAPS to open the Simulation Control dialog. Or click the ICAPS icon from the toolbar.**



Mode: ● Standard ○ Simulation Template ○ Monte Carlo ○ Faults

Data Reduction: ● Interactive ○ Batch ☑ Script

Save Data From: ● All ○ Test points ○ Nodes + test points

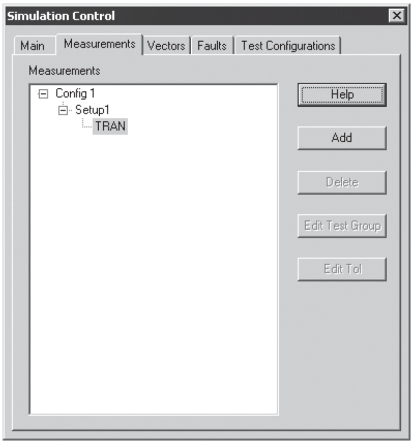
□ Remote Spice

The "Interactive" Data Reduction mode allows the simulator screen to appear while the simulation is running. The "Batch" mode prevents the simulator screen from appearing so simulations run more quickly.

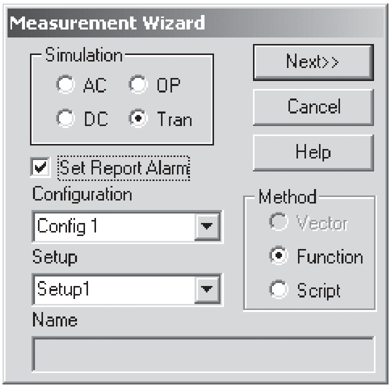
- **Make sure that the Script box is checked. It enables the automated ICL measurement features.**

Next, let's select the circuit vectors that will be measured.

- **Click on the Measurements tab toward the top of the Simulation Control dialog.**



- **Select the "TRAN" branch in the measurement tree, then click on the Add button to open the first Measurement Wizard.**



Simulation: ○ AC ○ OP ○ DC ● Tran

☑ Set Report Alarm

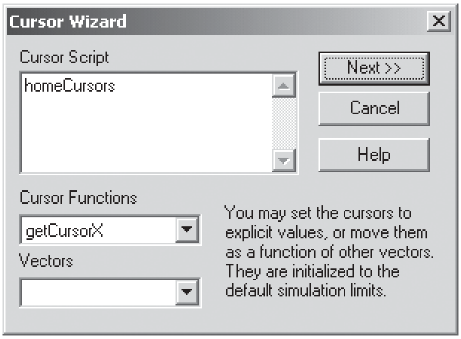
Configuration: Config 1

Setup: Setup1

Name:

Method: ○ Vector ● Function ○ Script

- **Click on the Tran radio button if it is not already selected.**
- **Check the Set Report Alarm box.**
- **Click on the Next button to open the Cursor Wizard.**



Cursor Script: homeCursors

Cursor Functions: getCursorX

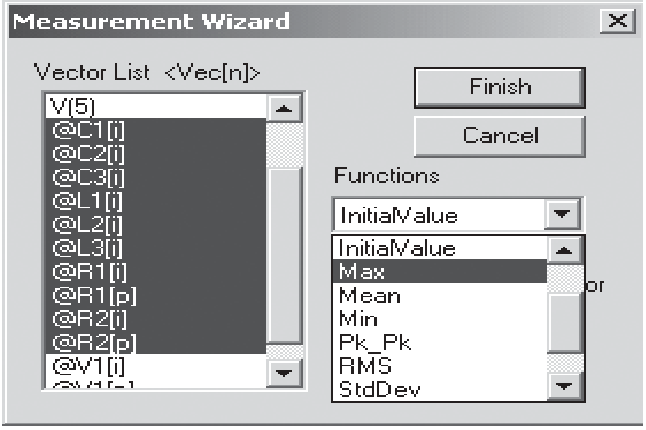
Vectors:

You may set the cursors to explicit values, or move them as a function of other vectors. They are initialized to the default simulation limits.

ICL Cursor Basics:

A cursor is an abstract marker that is connected to the IsSpice4 default plot vector (i.e., x axis data, time for transient, frequency for AC). Cursors 0 and 1 are the default, and most measurements are made between these default cursors. The "homeCursors" function sets cursor 0 to the beginning of the plot, and cursor 1 to the end of the plot. For more information about the cursors, press the Help button. For now, let's use the default, "homeCursors."

- **Click on the Next button to open the final Measurement Wizard dialog.**
- **Select Max from the Functions drop down list.**
- **Then select the vectors for which to measure the maximum value: @C1[i] through @R2[p]. You can left click on the first entry, then hold down the Shift key and click on the last one. This also selects all that lie between.**



- **Then, click on the Finish button to close the dialog.**

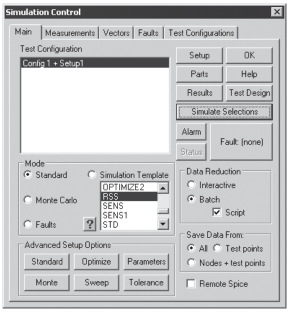
The new branch named "Max" appears under the "TRAN" branch.

Now let's add a peak_peak measurement for the same vectors as we selected for the maximum value measurement.

- **To add a peak_peak measurement, follow the same steps used above to add the Max measurement.**

Running the Standard Simulation

Now that we've specified the measurements that will be made when the simulation is performed, let's return to the Main tab of the Simulation Control dialog and launch a standard simulation.



Test Configuration: Config 1 + Setup1

Fault: (none)

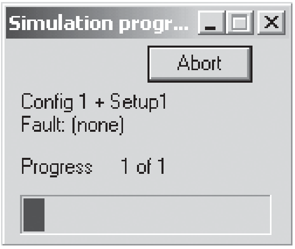
Mode: ● Standard ○ Monte Carlo ○ Faults ○ Simulation Template (OPTIMIZE2, RSS, SENS, SENS1, STD)

Data Reduction: ○ Interactive ● Batch ☑ Script

Save Data From: ● All ○ Test points ○ Nodes + test points

□ Remote Spice

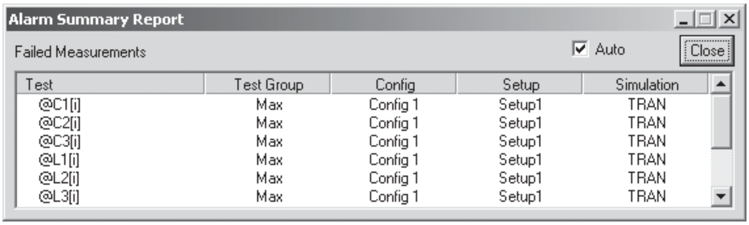
- **Click on the Main tab in the Simulation Control dialog.**
- **Select "Standard" radio button and "Batch" radio button. "Batch" will speed up the simulation by turning off the IsSpice4 real-time waveform display.**
- **Click on the Measurements Tab of the Simulation Control dialog.**
- **Click on the Simulate Selections button in the Main Tab.**



Config 1 + Setup1
Fault: (none)

Progress 1 of 1

The status dialog on the left appears during the simulation. This illustrates which configuration is being simulated, and the name of the component failure mode that has been inserted. In this case, there is no fault inserted.

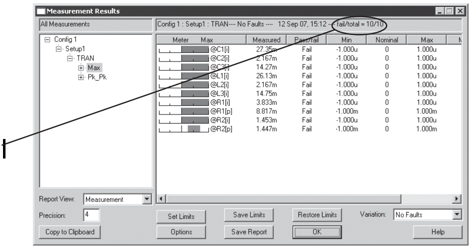


Failed Measurements ☑ Auto

Test	Test Group	Config	Setup	Simulation
@C1[i]	Max	Config 1	Setup1	TRAN
@C2[i]	Max	Config 1	Setup1	TRAN
@C3[i]	Max	Config 1	Setup1	TRAN
@L1[i]	Max	Config 1	Setup1	TRAN
@L2[i]	Max	Config 1	Setup1	TRAN
@L3[i]	Max	Config 1	Setup1	TRAN

The Alarm Summary dialog reports which electrical measurements you specified failed. Since the test limits for the Max or pk-pk measurements were not yet set, you can ignore this dialog. Component Stress Alarms are covered in the next tutorial.

Because we have not yet set the pass/fail limits, all ten measurements fail.



Config 1 : Setup1 : TRAN---- No Faults ---- 12 Sep 07, 15:12 --- fail/total = 10/10

Meter Max	Measured	Pass/fail	Min	Nominal	Max
@C1[i]	27.35m	Fail	-1.000u	0	1.000u
@C2[i]	2.167m	Fail	-1.000u	0	1.000u
@C3[i]	14.27m	Fail	-1.000u	0	1.000u
@L1[i]	26.13m	Fail	-1.000u	0	1.000u
@L2[i]	2.167m	Fail	-1.000u	0	1.000u
@L3[i]	14.75m	Fail	-1.000u	0	1.000u
@R1[i]	3.833m	Fail	-1.000u	0	1.000u
@R1[p]	8.817m	Fail	-1.000m	0	1.000m
@R2[i]	1.453m	Fail	-1.000u	0	1.000u
@R2[p]	1.447m	Fail	-1.000m	0	1.000m

Report View: Measurement

Precision: 4

Variation: No Faults

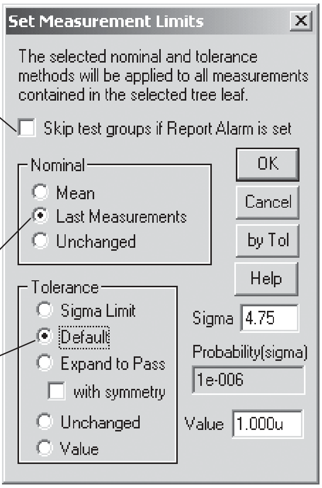
- **Click on the Results button.**
- **Select the Max test group in the tree.**

All the results of the simulation for the maximum value of each vector are displayed.

Setting the Standard Simulation’s Test Limits

Let’s set some test limits based upon the calculated values from the simulation. This will establish a pass/fail region for each measured vector (e.g., node voltage, pk-pk value, rise time, etc.). There are a number of ways to set test limits. Here, we use a simple default (+/- 10%).

- **Click on the Set Limits button in the Measurement Results dialog.**



The selected nominal and tolerance methods will be applied to all measurements contained in the selected tree leaf.

□ Skip test groups if Report Alarm is set

Nominal
○ Mean
● Last Measurements
○ Unchanged

Tolerance
○ Sigma Limit
● Default
○ Expand to Pass
□ with symmetry
○ Unchanged
○ Value

Sigma 4.75

Probability(sigma) 1e-006

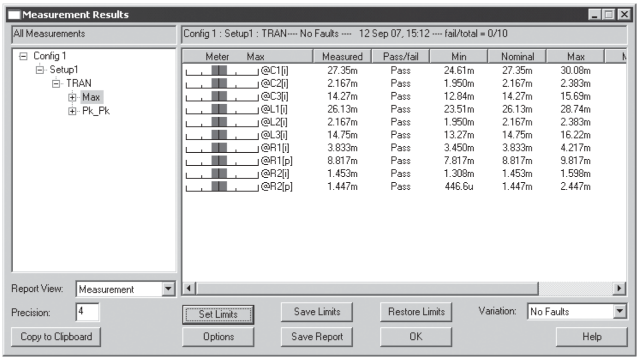
Value 1.000u

- **Uncheck the “Skip test groups if Report Alarm is set” check box, since this is a Stress Alarm measurement.**

- **Select the Last Measurements function in the Nominal group, and the Default function in the Tolerance group.**

- **Click on the OK button.**

This sets the test limits for each measurement based upon its calculated value. Notice that all of the maximum value tests now pass.



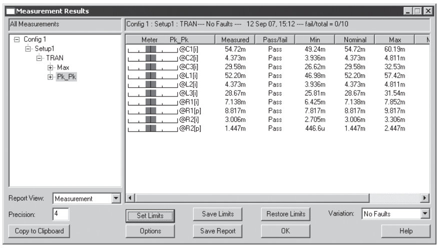
Config 1 : Setup1 : TRAN---- No Faults ---- 12 Sep 07, 15:12 ---- fail/total = 0/10

Meter Max	Measured	Pass/fail	Min	Nominal	Max
@C1[i]	27.35m	Pass	24.61m	27.35m	30.08m
@C2[i]	2.167m	Pass	1.950m	2.167m	2.383m
@C3[i]	14.27m	Pass	12.84m	14.27m	15.69m
@L1[i]	26.13m	Pass	23.51m	26.13m	28.74m
@L2[i]	2.167m	Pass	1.950m	2.167m	2.383m
@L3[i]	14.75m	Pass	13.27m	14.75m	16.22m
@R1[i]	3.833m	Pass	3.450m	3.833m	4.217m
@R1[p]	8.817m	Pass	7.817m	8.817m	9.817m
@R2[i]	1.453m	Pass	1.308m	1.453m	1.598m
@R2[p]	1.447m	Pass	446.6u	1.447m	2.447m

- **Similarly, select the "Pk-Pk" branch of the tree and then set the test limits to the "Last Measurement" function in the Set Limits dialog.**

Since this is not a Stress Alarm measurement there is no need to uncheck the "Skip test groups..." checkbox. This sets the test limits for each measurement based upon the nominal simulation value. Notice that all of the peak-to-peak value tests now pass.

If you perform any design changes and run a simulation, the Results dialog will show the effects of the change.



Config 1 : Setup1 : TRAN---- No Faults ---- 12 Sep 07, 15:12 ---- fail/total = 0/10

Meter Pk_Pk	Measured	Pass/fail	Min	Nominal	Max
@C1[i]	54.72m	Pass	49.24m	54.72m	60.19m
@C2[i]	4.373m	Pass	3.936m	4.373m	4.811m
@C3[i]	29.58m	Pass	26.62m	29.58m	32.53m
@L1[i]	52.20m	Pass	46.98m	52.20m	57.42m
@L2[i]	4.373m	Pass	3.936m	4.373m	4.811m
@L3[i]	28.67m	Pass	25.81m	28.67m	31.54m
@R1[i]	7.138m	Pass	6.425m	7.138m	7.852m
@R1[p]	8.817m	Pass	7.817m	8.817m	9.817m
@R2[i]	3.006m	Pass	2.705m	3.006m	3.306m
@R2[p]	1.447m	Pass	446.6u	1.447m	2.447m

- **Click on the OK button to close the Results dialog.**

Running the RSS Simulation

In the RSS analysis, after first running the reference simulation, each parameter is perturbed. For each measurement vector, the difference between the reference and the perturbed result, and the sensitivities are saved. The square root of the result is saved in an “rss” plot. The RSS results are saved in the IsSpice4 output file for later read back. You can set the measurement test limits by expanding the measurements to "pass with symmetry" in the Results dialog.

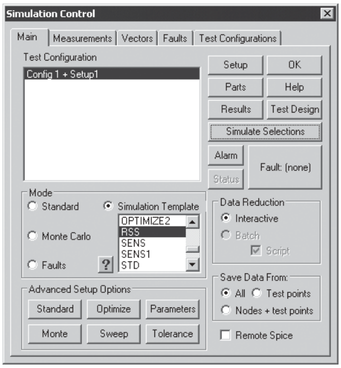
Gaussian distribution is used by default.

Each tolerance is taken as the 3 sigma value, or 3 standard deviations, and we report the RSS results as though they were also 3 sigma. The parameter perturbation is set to 1 sigma, a compromise between a negligibly small value and the entire tolerance band. Making the perturbation this way eliminates some errors due to local maximum values occurring nearby.

Now let’s run the RSS analysis and use its results to set the test limits and see how they change.

For more information about RSS, EVA and WCS analyses, as well as simulation templates, refer to the in-application help.

- **Select the RSS simulation template in the Simulation Template List Box. Note that the selected simulation template will not be run unless the radio button to the left of the List Box and above the ‘?’ button is filled.**



Test Configuration: Config 1 + Setup1

Mode: ● Simulation Template (RSS selected)

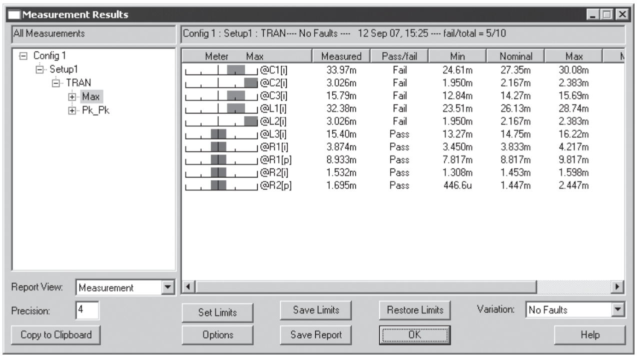
Data Reduction: ● Interactive

Save Data From: ● All

- **Click on the Simulate Selections button to launch the simulator.**

- **In the Simulation Control dialog, click on the Results button to open the Measurement Results dialog.**

- **Click on the “Max” tab to view its results.**



Config 1 : Setup1 : TRAN---- No Faults ---- 12 Sep 07, 15:25 ---- fail/total = 5/10

Meter Max	Measured	Pass/fail	Min	Nominal	Max
@C1[i]	33.97m	Fail	24.61m	27.35m	30.08m
@C2[i]	3.026m	Fail	1.950m	2.167m	2.383m
@C3[i]	15.79m	Fail	12.84m	14.27m	15.69m
@L1[i]	32.38m	Fail	23.51m	26.13m	28.74m
@L2[i]	3.026m	Fail	1.950m	2.167m	2.383m
@L3[i]	15.40m	Pass	13.27m	14.75m	16.22m
@R1[i]	3.874m	Pass	3.450m	3.833m	4.217m
@R1[p]	8.933m	Pass	7.817m	8.817m	9.817m
@R2[i]	1.532m	Pass	1.308m	1.453m	1.598m
@R2[p]	1.695m	Pass	446.6u	1.447m	2.447m

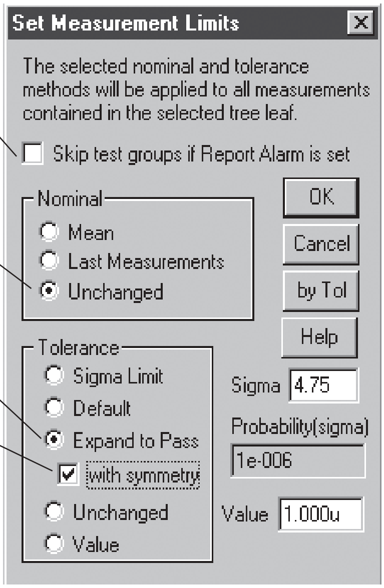
Report View: Measurement

Precision: 4

Variation: No Faults

Notice that several of the measurements now fail, based upon the calculated RSS data. Let’s expand the limits and see how they change.

- **Click on the Set Limits button to open the Set Measurement Limits dialog.**



The selected nominal and tolerance methods will be applied to all measurements contained in the selected tree leaf.

☐ Skip test groups if Report Alarm is set

Nominal
○ Mean
○ Last Measurements
● Unchanged

Tolerance
○ Sigma Limit
○ Default
● Expand to Pass
☑ with symmetry
○ Unchanged
○ Value

Sigma 4.75

Probability(sigma) 1e-006

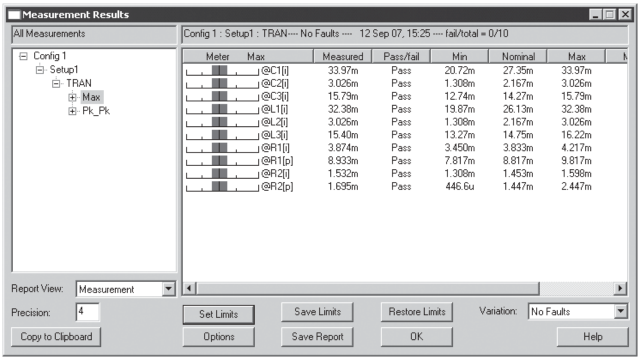
Value 1.000u

- **Uncheck the “Skip test groups” checkbox.**

- **Select Unchanged in the Nominal group and the Expand to Pass function in the Tolerance group. Check the “with symmetry” box.**

- **Click on the OK button to close the dialog.**

The new results are displayed. Notice that all of the maximum value tests now pass, and how the limits have changed.



Measurement Results

All Measurements

Config 1 : Setup1 : TRAN---- No Faults ---- 12 Sep 07, 15:25 ---- fail/total = 0/10

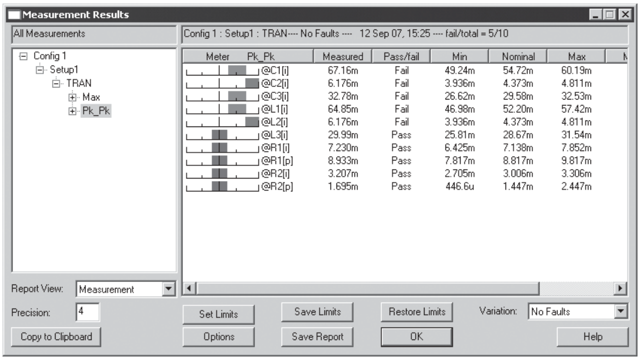
Meter	Max	Measured	Pass/fail	Min	Nominal	Max
	@C1[i]	33.97m	Pass	20.72m	27.35m	33.97m
	@C2[i]	3.026m	Pass	1.308m	2.167m	3.026m
	@C3[i]	15.79m	Pass	12.74m	14.27m	15.79m
	@L1[i]	32.38m	Pass	19.87m	26.13m	32.38m
	@L2[i]	3.026m	Pass	1.308m	2.167m	3.026m
	@L3[i]	15.40m	Pass	13.27m	14.75m	16.22m
	@R1[i]	3.874m	Pass	3.450m	3.833m	4.217m
	@R1[p]	8.933m	Pass	7.817m	8.817m	9.817m
	@R2[i]	1.532m	Pass	1.308m	1.453m	1.598m
	@R2[p]	1.695m	Pass	446.6u	1.447m	2.447m

Report View: Measurement

Precision: 4

Variation: No Faults

- **Select the "Pk-Pk" branch of the tree.**



Measurement Results

All Measurements

Config 1 : Setup1 : TRAN---- No Faults ---- 12 Sep 07, 15:25 ---- fail/total = 5/10

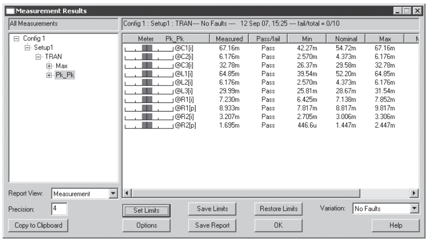
Meter	Pk_Pk	Measured	Pass/fail	Min	Nominal	Max
	@C1[i]	67.16m	Fail	49.24m	54.72m	60.19m
	@C2[i]	6.176m	Fail	3.936m	4.373m	4.811m
	@C3[i]	32.78m	Fail	26.62m	29.58m	32.53m
	@L1[i]	64.85m	Fail	46.98m	52.20m	57.42m
	@L2[i]	6.176m	Fail	3.936m	4.373m	4.811m
	@L3[i]	29.99m	Pass	25.81m	28.67m	31.54m
	@R1[i]	7.230m	Pass	6.425m	7.138m	7.852m
	@R1[p]	8.933m	Pass	7.817m	8.817m	9.817m
	@R2[i]	3.207m	Pass	2.705m	3.006m	3.306m
	@R2[p]	1.695m	Pass	446.6u	1.447m	2.447m

Report View: Measurement

Precision: 4

Variation: No Faults

- **Click on Set Limits button, uncheck "skip test group" checkbox, select Unchanged in Nominal group, and in Tolerance group select "Expand to Pass" "with symmetry". Click OK button.**



Measurement Results

All Measurements

Config 1 : Setup1 : TRAN---- No Faults ---- 12 Sep 07, 15:25 ---- fail/total = 0/10

Meter	Pk_Pk	Measured	Pass/fail	Min	Nominal	Max
	@C1[i]	67.16m	Pass	42.27m	54.72m	67.16m
	@C2[i]	6.176m	Pass	2.570m	4.373m	6.176m
	@C3[i]	32.78m	Pass	26.37m	29.58m	32.78m
	@L1[i]	64.85m	Pass	39.54m	52.20m	64.85m
	@L2[i]	6.176m	Pass	2.570m	4.373m	6.176m
	@L3[i]	29.99m	Pass	25.81m	28.67m	31.54m
	@R1[i]	7.230m	Pass	6.425m	7.138m	7.852m
	@R1[p]	8.933m	Pass	7.817m	8.817m	9.817m
	@R2[i]	3.207m	Pass	2.705m	3.006m	3.306m
	@R2[p]	1.695m	Pass	446.6u	1.447m	2.447m

Report View: Measurement

Precision: 4

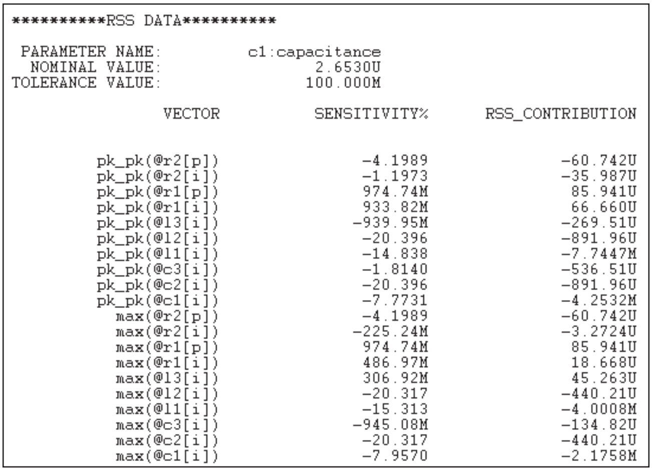
Variation: No Faults

All of the peak-to-peak value tests now pass because the minimum and maximum test limits have been "pushed out".

- **Click on the OK button to close the Measurement Results dialog.**

Again, if you now change a simulation parameter and rerun a simulation, the Results dialog will show the effects of this change. Presently, the test limits will account for the RSS limits of the measured vectors.

The RSS numerical simulation results may also be viewed in the IsSpice4 output text file.



```
**********RSS DATA**********

 PARAMETER NAME:          c1:capacitance
  NOMINAL VALUE:                 2.6530U
TOLERANCE VALUE:               100.000M

               VECTOR          SENSITIVITY%      RSS_CONTRIBUTION


         pk_pk(@r2[p])               -4.1989              -60.742U
         pk_pk(@r2[i])               -1.1973              -35.987U
         pk_pk(@r1[p])               974.74M               85.941U
         pk_pk(@r1[i])               933.82M               66.660U
         pk_pk(@l3[i])              -939.95M              -269.51U
         pk_pk(@l2[i])               -20.396              -891.96U
         pk_pk(@l1[i])               -14.838              -7.7447M
         pk_pk(@c3[i])               -1.8140              -536.51U
         pk_pk(@c2[i])               -20.396              -891.96U
         pk_pk(@c1[i])               -7.7731              -4.2532M
           max(@r2[p])               -4.1989              -60.742U
           max(@r2[i])              -225.24M              -3.2724U
           max(@r1[p])               974.74M               85.941U
           max(@r1[i])               486.97M               18.668U
           max(@l3[i])               306.92M               45.263U
           max(@l2[i])               -20.317              -440.21U
           max(@l1[i])               -15.313              -4.0008M
           max(@c3[i])              -945.08M              -134.82U
           max(@c2[i])               -20.317              -440.21U
           max(@c1[i])               -7.9570              -2.1758M
```

- **Close the Measurement Results dialog and the Simulation Control dialog.**

- **From the Actions menu, select the Text Edit function to launch the text editor.**

- **Select the OUT icon Out in the IsEd toolbar to display the text results.**

This concludes the Monte Carlo and RSS tutorial.

- **Close all open dialogs and exit the SpiceNet program.**

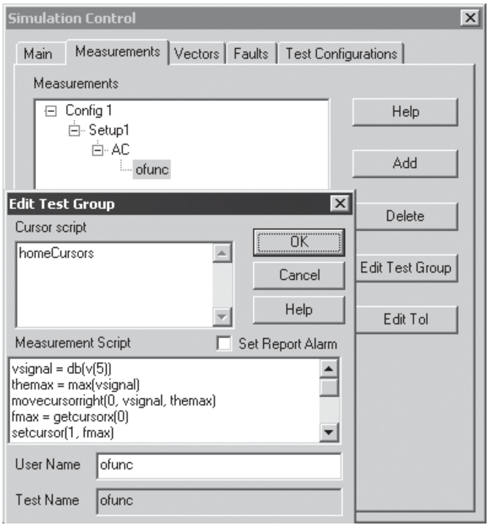
You may discard the changes that have been made to the drawing file.

Optimizing Circuit Performance

With ICAP/4's optimization feature, you can specify which parameters to change and by what percentage. When a simulation runs, the parameters are varied to try to minimize a design objective (function), which is a function (e.g., phase margin), bandwidth, etc.) that can be reduced to a scalar unit.

Let's try an exercise with the goal of obtaining optimized values for all the active components with 30% tolerance so that the bandwidth is 22.45K.

- **Open ..\spice8\Circuits\Optimizer\bwopt.dwg.**
- **Double click on L1 in the SpiceNet schematic to bring up its property dialog.**
- **Select Tolerance\Sweep\Optimizer tab, and enter 30% for the value under the Optimize column.**
- **Repeat the former steps for L2, L3, L4, C1, C2, C3 and C4. OR simply click Next Part button and enter the values.**



Simulation Control

Main | Measurements | Vectors | Faults | Test Configurations

Measurements

- Config 1
  - Setup1
    - AC
      - ofunc

Help | Add | Delete | Edit Test Group | Edit Tol

Edit Test Group

Cursor script

homeCursors

OK | Cancel | Help

Measurement Script □ Set Report Alarm

```
vsignal = db(v(5))
themax = max(vsignal)
movecursorright(0, vsignal, themax)
fmax = getcursorx(0)
setcursor(1, fmax)
```

User Name: ofunc

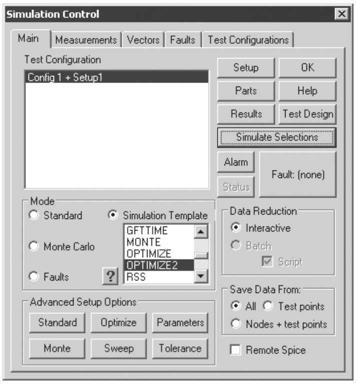
Test Name: ofunc

- **Click on ICAPS icon**
- **Select Measurement tab, and select ofunc.**
- **Click on Edit Test group button.**

You will see this script in the Edit Test Group window:

homeCursors;
In the Measurement Script field:
vsignal = db(v(5))
themax = max(vsignal)
movecursorright(0, vsignal, themax)
fmax = getcursorx(0)
setcursor(1, fmax)
movecursorright(1, vsignal, themax-3)
movecursorleft(0, vsignal, themax-3)
bandwidth = getcursorx(1)-getcursorx(0)
ofunc = bandwidth-22.45e3
*ofunc = ofunc*ofunc*
print ofunc

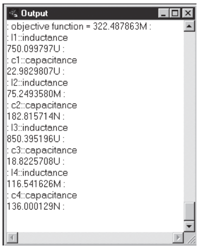
These scripts can be modified according to your needs. All measurements such as min, max, rms, pk_pk, etc. are measured between the default cursors, cursor 0 and 1. The other cursors are for labeling and measuring delta values.



- **Click the OK button to close the Edit Test Group dialog.**
- **Select Main tab.**
- **Select the Optimize2 simulation template.**
- **Click on the Simulation Selections button.**

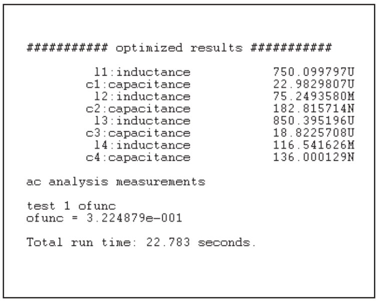
All the modified simulation parameters and parameter changes that give the bestval are shown in the Output dialog. The results also can be viewed in tabular form at the end of the .OUT file.

From Output dialog:



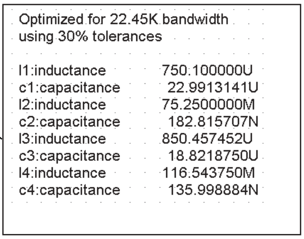
```
: objective function = 322.487863M :
: l1::inductance
750.099797U :
: c1::capacitance
22.9829807U :
: l2::inductance
75.2493580M :
: c2::capacitance
182.815714N :
: l3::inductance
850.395196U :
: c3::capacitance
18.8225708U :
: l4::inductance
116.541626M :
: c4::capacitance
136.000129N :
```

From .OUT file:



```
########### optimized results ###########

        l1:inductance          750.099797U
      c1:capacitance          22.9829807U
        l2:inductance          75.2493580M
      c2:capacitance          182.815714N
        l3:inductance          850.395196U
      c3:capacitance          18.8225708U
        l4:inductance          116.541626M
      c4:capacitance          136.000129N

ac analysis measurements

test 1 ofunc
ofunc = 3.224879e-001

Total run time: 22.783 seconds.
```

recommended values



```
Optimized for 22.45K bandwidth
using 30% tolerances

l1:inductance        750.100000U
c1:capacitance        22.9913141U
l2:inductance        75.2500000M
c2:capacitance        182.815707N
l3:inductance        850.457452U
c3:capacitance        18.8218750U
l4:inductance        116.543750M
c4:capacitance        135.998884N
```

Compare the optimized values with the recommended values listed on the schematic. You will see that these two results should be almost the same.

Tutorial #7 - Design Validation and Automated Measurements

Tutorial#7 is for all ICAP/4 Windows products, ICAP/4 Professional and Test Designer users. In this tutorial, we'll address the following ICAP/4 features:

1. Setting and charting desired electrical measurements
2. Data, setting limits, and reports for measured data
3. Alarm warnings for desired measurements
4. Monte Carlo analysis using Interactive Command Language (ICL) simulation template

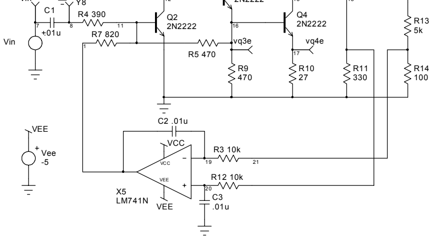
Access to these features is controlled by the Measurements tab in the Simulation Control dialog. This tab will display your test configurations and simulation setups into a tree. You can then add various measurements to the branches of the tree. There are three basic kinds of tests that this tutorial will cover in detail:

1. Operating Point
2. Vector Measurement
3. Measurement scripts using ICL

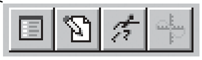
Identifying Measurements

The first thing we need to do is define desired measurements, and circuit locations where they will be taken:

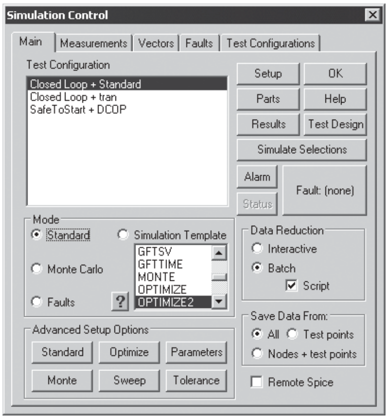
- **Launch SpiceNet and open the Sample2.DWG schematic in the spice8\circuits\sample2 folder.**



- **Select the “ICAPS...” function in the Actions menu to open the Simulation Control dialog, or select the ICAPS icon from the toolbar.**



- **Select Test Configuration: Closed Loop + Standard**
- **Select Mode: Standard**



Simulation Control

Main | Measurements | Vectors | Faults | Test Configurations

Test Configuration:
- Closed Loop + Standard
- Closed Loop + tran
- SafeToStart + DCOP

Setup | OK | Parts | Help | Results | Test Design | Simulate Selections | Alarm | Status | Fault: (none)

Mode: ● Standard ○ Simulation Template ○ Monte Carlo ○ Faults

Simulation Template: GFTSV, GFTTIME, MONTE, OPTIMIZE, OPTIMIZE2

Data Reduction: ○ Interactive ● Batch ☑ Script

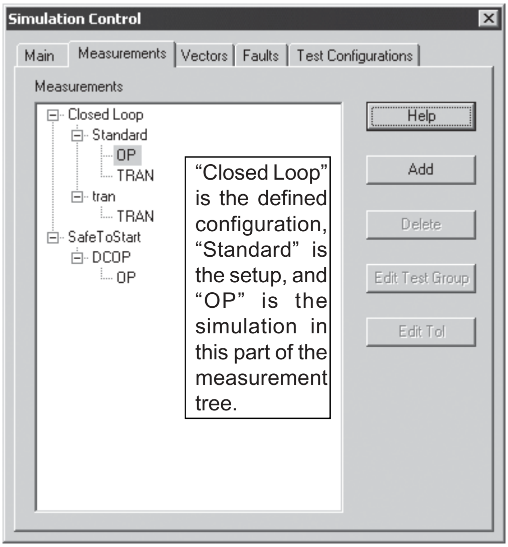
Advanced Setup Options: Standard | Optimize | Parameters | Monte | Sweep | Tolerance

Save Data From: ● All ○ Test points ○ Nodes + test points

□ Remote Spice

Side Note: The Save Data functions (see lower right) don’t save data within subcircuits. Use “All” (default) unless a large simulation requires more RAM than is available.

- **Click on the Measurements tab.**



Simulation Control

Main | Measurements | Vectors | Faults | Test Configurations

Measurements:
```
⊟ Closed Loop
  ⊟ Standard
     OP
     TRAN
  ⊟ tran
     TRAN
⊟ SafeToStart
  ⊟ DCOP
     OP
```

Help | Add | Delete | Edit Test Group | Edit Tol

“Closed Loop” is the defined configuration, “Standard” is the setup, and “OP” is the simulation in this part of the measurement tree.

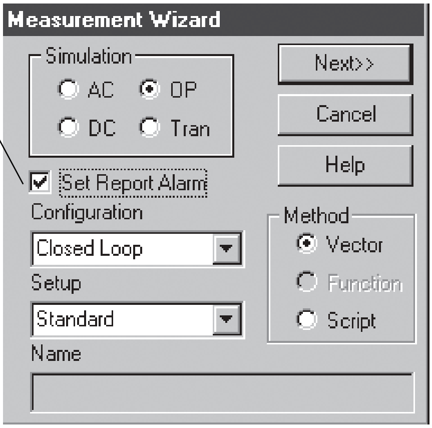
"OP" is the circuit's DC operating point.

"TRAN" is a time-based measurement.

- **Select the Closed Loop/Standard/OP branch in the measurement tree, then click on the Add button to open the Measurement Wizard dialog.**
- **Check Set Report Alarm to activate the alarm feature for this set of measurements. Press the "Next>>" button.**

Alarms can be set to monitor current dissipation, power, etc. so that you will automatically be notified if an out-of-tolerance or hazardous condition exists during any simulation.

The Configuration, Setup, and Simulation that you've selected in the measurement tree are already initialized for you.



Simulation: ○ AC ● OP ○ DC ○ Tran

☑ Set Report Alarm

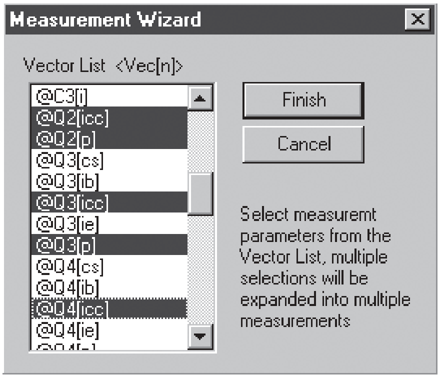
Configuration: Closed Loop

Setup: Standard

Name:

Method: ● Vector ○ Function ○ Script

- **Press and hold the Ctrl key and press the left mouse button to select the 14 resistor and transistor power dissipations (power only) in the list (@R3[p] through @R14[p], then also individually select @Q2[p] through @Q4[p].**
- **Similarly, select all 3 of the transistor collector current values in the list (@Q2[icc] through @Q4[icc]).**
- **Click on the Finish button.**

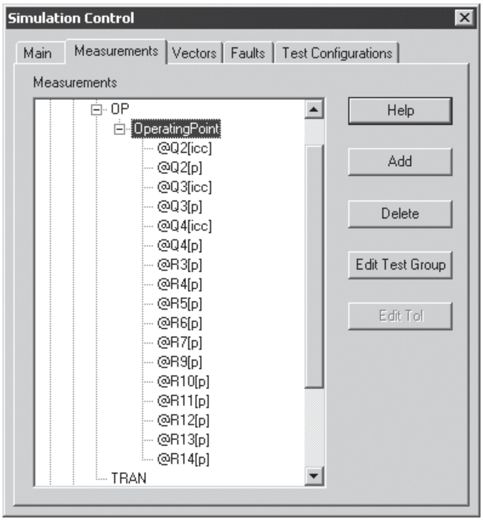


Vector List <Vec[n]>

@C3[i]
@Q2[icc]
@Q2[p]
@Q3[cs]
@Q3[ib]
@Q3[icc]
@Q3[ie]
@Q3[p]
@Q4[cs]
@Q4[ib]
@Q4[icc]
@Q4[ie]

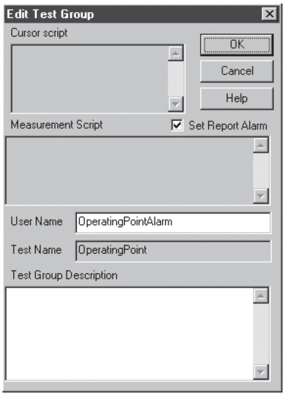
Select measuremt parameters from the Vector List, multiple selections will be expanded into multiple measurements

This creates a new leaf called “Operating Point” that contains all of the waveform vectors just selected.



Any vector(s) in the list may be selected.

- **Select the Operating Point branch and click the Edit Test Group button to open the Edit Test Group dialog.**



☑ Set Report Alarm

User Name: OperatingPointAlarm

Test Name: OperatingPoint

- **Change the User Name to “OperatingPointAlarm.”**

- **Click on the OK button to dismiss the dialog.**

- **Click on the Add button again, and click on the “Next>>” button to proceed to the vector selection Measurement Wizard.**

You can double-click on any leaf in order to view its contents.

- **Scroll down the vector list to the entry “V(1).” Click the left mouse button to select V(1).**

- **Scroll down the list until you get to the “V(VEE)” entry. Press and hold the Shift key and click the left mouse button on the “V(VEE)” entry. This will select the entire list of node voltages between “V(1)” and “V(VEE),” inclusive.**

15 vectors should be selected.

- **Scroll down the list until you get to the “@Vcc[i]” entry. Press and hold the Control key and click on the “@Vcc[i]” entry to add Vcc current to the selected list.**

- **Click on the Finish button to dismiss the Measurement Wizard.**

- **Click on the Main tab, then click on the Simulate Selections button.**

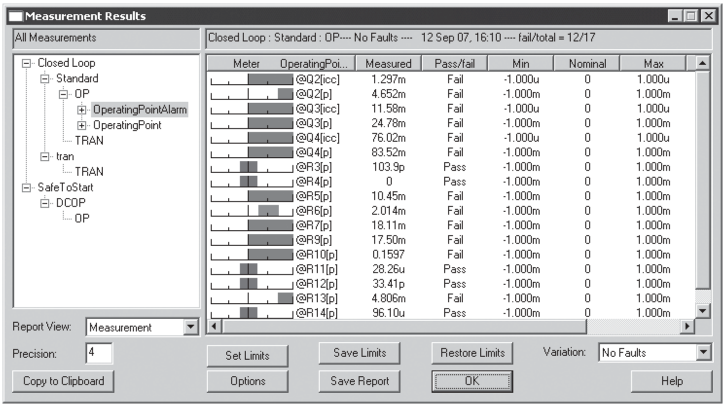
Simulation progress and Alarm Summary Report dialogs will appear once the simulation is done.

Setting Alarm Limits

Let’s set the alarm limits.

- **Click on the Results button in the Simulation Control dialog to open the Measurement Results dialog.**

- **Select the OperatingPointAlarm test group.**



Closed Loop : Standard : OP---- No Faults ---- 12 Sep 07, 16:10 ---- fail/total = 12/17

Meter	OperatingPoi...	Measured	Pass/fail	Min	Nominal	Max
	@Q2[icc]	1.297m	Fail	-1.000u	0	1.000u
	@Q2[p]	4.652m	Fail	-1.000m	0	1.000m
	@Q3[icc]	11.58m	Fail	-1.000u	0	1.000u
	@Q3[p]	24.78m	Fail	-1.000m	0	1.000m
	@Q4[icc]	76.02m	Fail	-1.000u	0	1.000u
	@Q4[p]	83.52m	Fail	-1.000m	0	1.000m
	@R3[p]	103.9p	Pass	-1.000m	0	1.000m
	@R4[p]	0	Pass	-1.000m	0	1.000m
	@R5[p]	10.45m	Fail	-1.000m	0	1.000m
	@R6[p]	2.014m	Fail	-1.000m	0	1.000m
	@R7[p]	18.11m	Fail	-1.000m	0	1.000m
	@R9[p]	17.50m	Fail	-1.000m	0	1.000m
	@R10[p]	0.1597	Fail	-1.000m	0	1.000m
	@R11[p]	28.26u	Pass	-1.000m	0	1.000m
	@R12[p]	33.41p	Pass	-1.000m	0	1.000m
	@R13[p]	4.806m	Fail	-1.000m	0	1.000m
	@R14[p]	96.10u	Pass	-1.000m	0	1.000m

Report View: Measurement

Precision: 4

Variation: No Faults

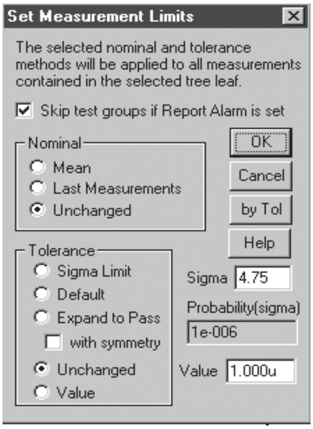
Since we haven’t set the test limits for the alarms, 12 of the 17 tests have failed.

- **Click on the Set Limits button to open the Set Measurement Limits dialog.**

- **Uncheck the “Skip test groups…” check box.**

- **Select Unchanged function in the Nominal group and the Unchanged function in the Tolerance group.**

Click on the Help button for a detailed explanation of how to set the Absolute and Percent Tolerance values.



Set Measurement Limits

The selected nominal and tolerance methods will be applied to all measurements contained in the selected tree leaf.

☑ Skip test groups if Report Alarm is set

Nominal
○ Mean
○ Last Measurements
● Unchanged

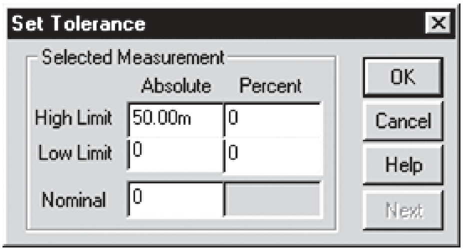
Tolerance
○ Sigma Limit
○ Default
○ Expand to Pass
□ with symmetry
● Unchanged
○ Value

Sigma: 4.75

Probability(sigma): 1e-006

Value: 1.000u

- **Click on the “by Tol” button to open the Set Tolerance dialog, which allows you to set asymmetrical tolerances by percent or value.**



Set Tolerance

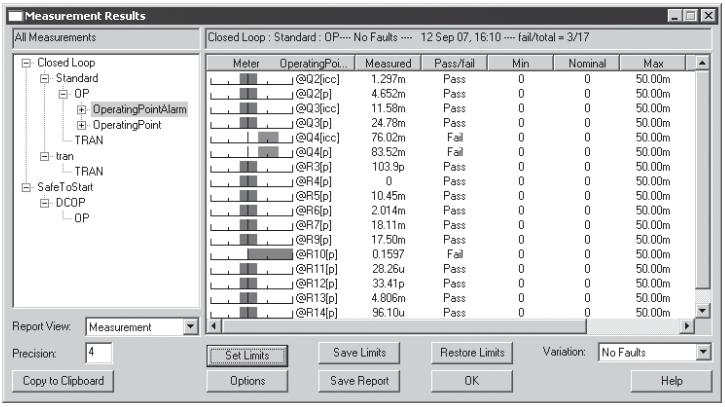
Selected Measurement

	Absolute	Percent
High Limit	50.00m	0
Low Limit	0	0
Nominal	0	

- **Set the High Limit (Absolute) value to 50.00m**

- **Click on the OK button to set the alarm limit to 50m for each measurement.**

Notice that now only 3 of the 17 tests have failed. This is because we have not yet set all of the test limits for the alarms.



Measurement Results

All Measurements

Closed Loop : Standard : OP---- No Faults ---- 12 Sep 07, 16:10 ---- fail/total = 3/17

Closed Loop
- Standard
  - OP
    - OperatingPointAlarm
    - OperatingPoint
    - TRAN
  - tran
    - TRAN
- SafeToStart
  - DCOP
    - OP

Meter	OperatingPoi...	Measured	Pass/fail	Min	Nominal	Max
	@Q2[icc]	1.297m	Pass	0	0	50.00m
	@Q2[p]	4.652m	Pass	0	0	50.00m
	@Q3[icc]	11.58m	Pass	0	0	50.00m
	@Q3[p]	24.78m	Pass	0	0	50.00m
	@Q4[icc]	76.02m	Fail	0	0	50.00m
	@Q4[p]	83.52m	Fail	0	0	50.00m
	@R3[p]	103.9p	Pass	0	0	50.00m
	@R4[p]	0	Pass	0	0	50.00m
	@R5[p]	10.45m	Pass	0	0	50.00m
	@R6[p]	2.014m	Pass	0	0	50.00m
	@R7[p]	18.11m	Pass	0	0	50.00m
	@R9[p]	17.50m	Pass	0	0	50.00m
	@R10[p]	0.1597	Fail	0	0	50.00m
	@R11[p]	28.26u	Pass	0	0	50.00m
	@R12[p]	33.41p	Pass	0	0	50.00m
	@R13[p]	4.806m	Pass	0	0	50.00m
	@R14[p]	96.10u	Pass	0	0	50.00m

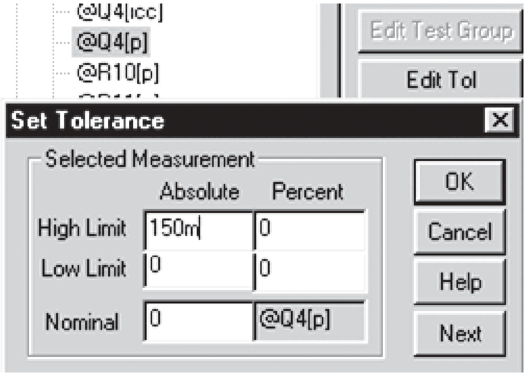
Report View: Measurement

Precision: 4

Variation: No Faults

- **Click on the OK button to close the Measurement Results dialog.**

- **Select the Measurements tab and double click the OperatingPointAlarm leaf to open up the leaf.**



Selected Measurement

	Absolute	Percent
High Limit	150m	0
Low Limit	0	0
Nominal	0	@Q4[p]

- **Select "@Q4[p]"**
- **Click on the EditTol button.**
- **Set the High Limit to 150m.**
- **Click on the OK button.**

- **Select the Main tab and click on the Alarm button in order to refresh the alarm measurements.**

@Q4[p] is no longer present in the alarm dialog, because we increased its limit.

- **Similarly, change the "@Q4[icc]" tolerance limit to 100m, and the "@R10[p]" tolerance limit to 200m in order to cause the alarms to pass.**

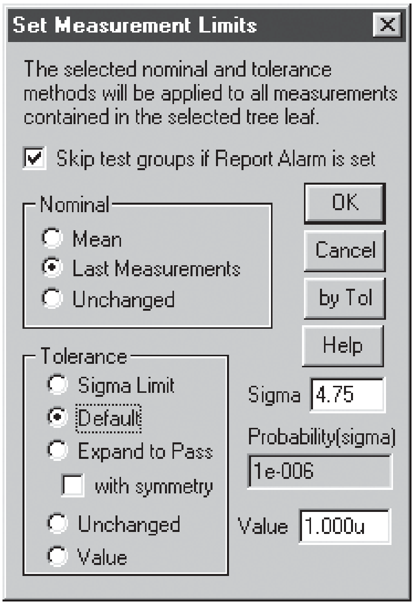
You can look in the Measurement Results dialog to see the calculated values in order to set the alarm limits. Typically, each alarm limit is set such that the alarm is not triggered unless a circuit fault condition exists. If you do this while the Simulation Results dialog is open, you must close the dialog and reopen it in order to update the test results.

Setting Operating Point Measurements

Let's set the limits for the Operating Point (OP) test vectors so that all tests will pass. In one step, we'll set all test limits to default limits about their last measured values.

- **Click on the Results button (in the Main tab) to open the Measurement Results dialog, select the OperatingPoint test group.**

- **Click on the Set Limits button.**
- **Select the Last Measurements function in the Nominal group and the Default function in the Tolerance group.**
- **Click on the OK button to dismiss the dialog.**



Set Measurement Limits

The selected nominal and tolerance methods will be applied to all measurements contained in the selected tree leaf.

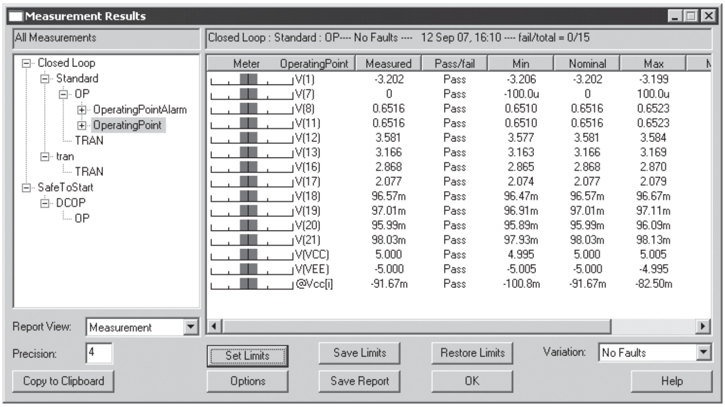
☑ Skip test groups if Report Alarm is set

Nominal
○ Mean
● Last Measurements
○ Unchanged

Tolerance
○ Sigma Limit
● Default
○ Expand to Pass
□ with symmetry
○ Unchanged
○ Value

Sigma: 4.75
Probability(sigma): 1e-006
Value: 1.000u

The tolerances are now set to default limits about the last measurements, so everything will pass.



Measurement Results

All Measurements

Closed Loop : Standard : OP---- No Faults ---- 12 Sep 07, 16:10 ---- fail/total = 0/15

Meter	OperatingPoint	Measured	Pass/fail	Min	Nominal	Max
	V(1)	-3.202	Pass	-3.206	-3.202	-3.199
	V(7)	0	Pass	-100.0u	0	100.0u
	V(8)	0.6516	Pass	0.6510	0.6516	0.6523
	V(11)	0.6516	Pass	0.6510	0.6516	0.6523
	V(12)	3.581	Pass	3.577	3.581	3.584
	V(13)	3.166	Pass	3.163	3.166	3.169
	V(16)	2.868	Pass	2.865	2.868	2.870
	V(17)	2.077	Pass	2.074	2.077	2.079
	V(18)	96.57m	Pass	96.47m	96.57m	96.67m
	V(19)	97.01m	Pass	96.91m	97.01m	97.11m
	V(20)	95.99m	Pass	95.89m	95.99m	96.09m
	V(21)	98.03m	Pass	97.93m	98.03m	98.13m
	V(VCC)	5.000	Pass	4.995	5.000	5.005
	V(VEE)	-5.000	Pass	-5.005	-5.000	-4.995
	@Vcc[i]	-91.67m	Pass	-100.8m	-91.67m	-82.50m

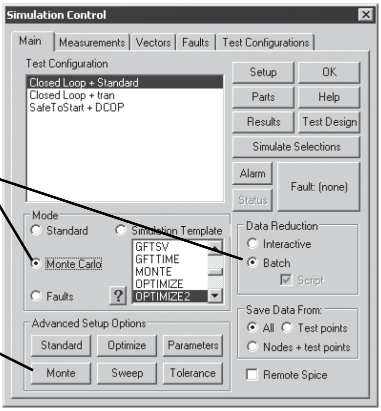
Report View: Measurement
Precision: 4
Variation: No Faults

- **Click on the OK button to dismiss the Measurement Results dialog.**

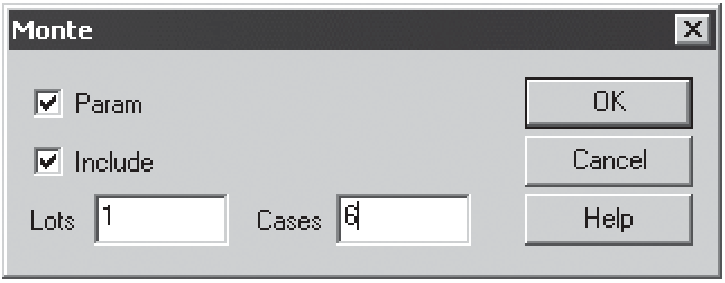
Running a standard Monte Carlo Analysis

You can improve these test limits by running a “standard” Monte Carlo analysis, which as mention previously provides Mean and 3-Sigma data for any device/node measurement specified by the user. It will not provide special statistical waveform data in IntuScope (i.e., curve family, histogram and probability plots).

- **Click on the Monte Carlo Mode in the Main tab.**
- **Select Batch in the Data Reduction box to the right.**
- **Then click on the Monte button to open up the Monte dialog.**



- **Change the number of Cases to 6 and click on the OK button to dismiss the dialog.**


Monte

☑ Param

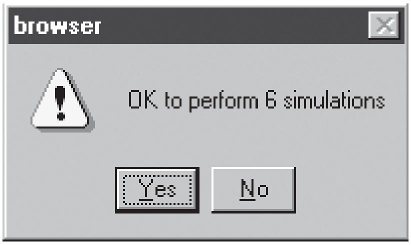
☑ Include

Lots 1 Cases 6

- **Click on the Simulate Selections button to run the simulation for Monte Carlo analysis.**

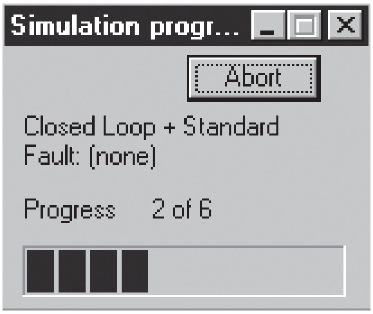
The following dialog appears. Because simulations may sometimes require large amounts of time, you are given the option to continue or to cancel the simulation request.

- **Click on the Yes button to run the simulations.**



OK to perform 6 simulations

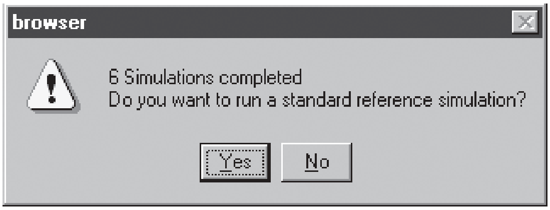
During the simulation, the "Simulation progress" dialog appears to inform you of the progress.



Closed Loop + Standard
Fault: (none)

Progress 2 of 6

After all of the simulations have been run, you will be asked if you want to run a standard reference simulation. If you select Yes, an additional simulation will be performed, and the data from that run may then be used to set the nominal measurement limits.



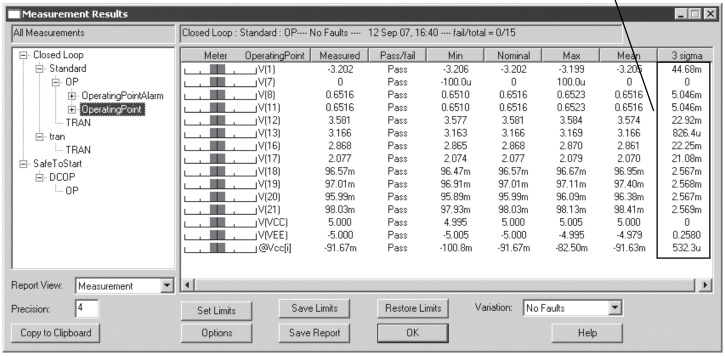
6 Simulations completed
Do you want to run a standard reference simulation?

- **Click on the Yes button so we'll have the nominal simulation results saved for the "Last Measurements" data.**

- **Click on the Results button in the Simulation Control Dialog.**

- **Select the OperatingPoint branch in the measurement tree.**

- **Scroll the report to the right and view the Monte Carlo statistics.**



Measurement Results

All Measurements

Closed Loop : Standard : OP---- No Faults ---- 12 Sep 07, 16:40 ---- fail/total = 0/15

```
Closed Loop
  Standard
    OP
      OperatingPointAlarm
      OperatingPoint
    TRAN
  tran
    TRAN
SafeToStart
  DCOP
    OP
```

Meter	OperatingPoint	Measured	Pass/fail	Min	Nominal	Max	Mean	3 sigma
	V(1)	-3.202	Pass	-3.206	-3.202	-3.199	-3.205	44.68m
	V(7)	0	Pass	-100.0u	0	100.0u	0	0
	V(8)	0.6516	Pass	0.6510	0.6516	0.6523	0.6516	5.046m
	V(11)	0.6516	Pass	0.6510	0.6516	0.6523	0.6516	5.046m
	V(12)	3.581	Pass	3.577	3.581	3.584	3.574	22.92m
	V(13)	3.166	Pass	3.163	3.166	3.169	3.166	826.4u
	V(16)	2.868	Pass	2.865	2.868	2.870	2.861	22.25m
	V(17)	2.077	Pass	2.074	2.077	2.079	2.070	21.08m
	V(18)	96.57m	Pass	96.47m	96.57m	96.67m	96.95m	2.567m
	V(19)	97.01m	Pass	96.91m	97.01m	97.11m	97.40m	2.568m
	V(20)	95.99m	Pass	95.89m	95.99m	96.09m	96.38m	2.567m
	V(21)	98.03m	Pass	97.93m	98.03m	98.13m	98.41m	2.569m
	V(VCC)	5.000	Pass	4.995	5.000	5.005	5.000	0
	V(VEE)	-5.000	Pass	-5.005	-5.000	-4.995	-4.979	0.2580
	@Vcc[i]	-91.67m	Pass	-100.8m	-91.67m	-82.50m	-91.63m	532.3u

Report View: Measurement

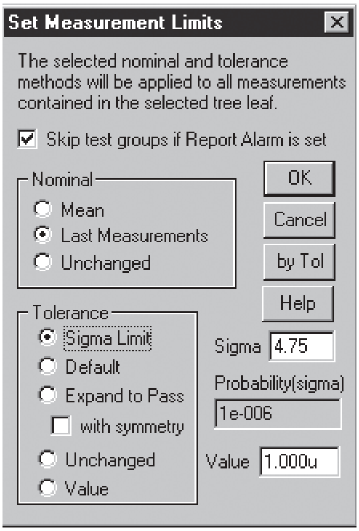
Precision: 4

Variation: No Faults

- **Click on the Set Limits button.**

- **Select the Last Measurements button in the Nominal group and Sigma Limit in the Tolerance group.**

- **Click on the OK button to close the Set Measurement Limits dialog.**



Set Measurement Limits

The selected nominal and tolerance methods will be applied to all measurements contained in the selected tree leaf.

☑ Skip test groups if Report Alarm is set

Nominal
○ Mean
● Last Measurements
○ Unchanged

Tolerance
● Sigma Limit
○ Default
○ Expand to Pass
□ with symmetry
○ Unchanged
○ Value

Sigma 4.75

Probability(sigma) 1e-006

Value 1.000u

You may change the default Sigma value, 4.75. The probability calculator shows the probability for one side of a normal distribution. The tolerances for all measurements are now set, based upon the Monte Carlo results.

- **Click on the OK button to close the Measurement Results dialog.**

Expanding Measurement Limits

You may also set limits based upon other measurements, such as temperature tests and power line variations. It is recommended that you perform these test limit expansions for all parametric variations for which you are aware.

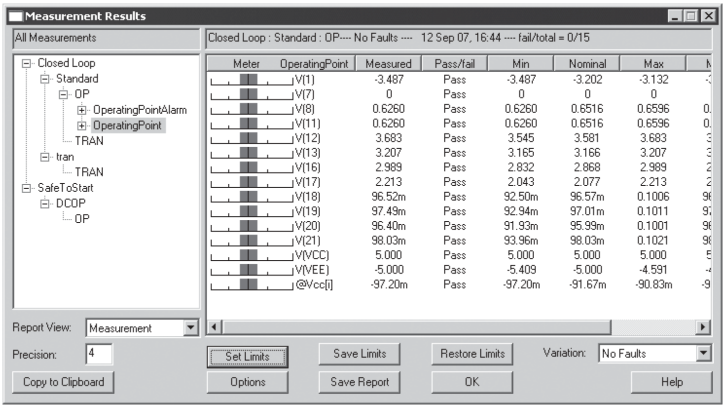
Let's run a simulation with the circuit temperature set 10 Degrees higher than room temperature.

- **Click on the Setup button in the Main tab.**
- **Click on the Circuit Temperature button and enter a value of 40.**
- **Click on the OK button to close the Circuit Temperature dialog.**
- **Click on the Done button to close the IsSpice4 Simulation Setup dialog.**
- **Then, click on the Standard mode radio button in the Simulation Control dialog.**
- **Select the "Closed Loop + Standard" test configuration.**
- **Click on the Simulate Selections button to run the simulations.**
- **Click on the Results button after the simulation.**
- **Select the OperatingPoint branch and click on the Set Limits button.**
- **Select the Unchanged function in the Nominal group and the Expand to Pass function in the Tolerance group.**

This will retain all previous tolerances except for those that fail; it expands failed tolerance values in order to pass the last test.

- **Click on the OK button to dismiss the dialog.**

The Result is shown below.

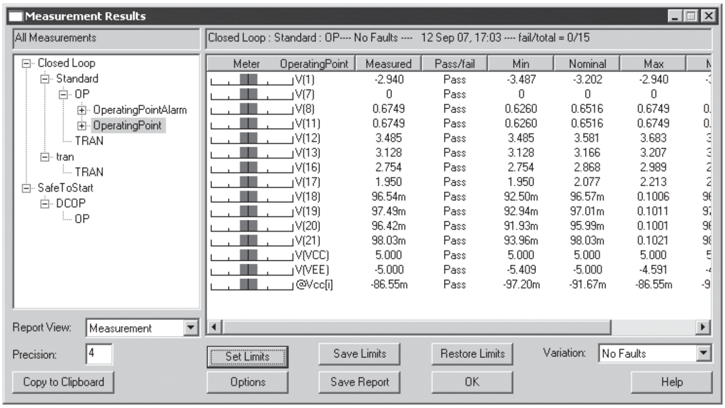


Closed Loop : Standard : OP---- No Faults ---- 12 Sep 07, 16:44 ---- fail/total = 0/15

Meter OperatingPoint	Measured	Pass/fail	Min	Nominal	Max
V(1)	-3.487	Pass	-3.487	-3.202	-3.132
V(7)	0	Pass	0	0	0
V(8)	0.6260	Pass	0.6260	0.6516	0.6596
V(11)	0.6260	Pass	0.6260	0.6516	0.6596
V(12)	3.683	Pass	3.545	3.581	3.683
V(13)	3.207	Pass	3.165	3.166	3.207
V(16)	2.989	Pass	2.832	2.868	2.989
V(17)	2.213	Pass	2.043	2.077	2.213
V(18)	96.52m	Pass	92.50m	96.57m	0.1006
V(19)	97.49m	Pass	92.94m	97.01m	0.1011
V(20)	96.40m	Pass	91.93m	95.99m	0.1001
V(21)	98.03m	Pass	93.96m	98.03m	0.1021
V(VCC)	5.000	Pass	5.000	5.000	5.000
V(VEE)	-5.000	Pass	-5.409	-5.000	-4.591
@Vcc[i]	-97.20m	Pass	-97.20m	-91.67m	-90.83m

Report View: Measurement
Precision: 4
Variation: No Faults

- **Similarly, rerun the simulation for low temperature (15 Degrees) and expand tolerance to pass.**



Closed Loop : Standard : OP---- No Faults ---- 12 Sep 07, 17:03 ---- fail/total = 0/15

Meter OperatingPoint	Measured	Pass/fail	Min	Nominal	Max
V(1)	-2.940	Pass	-3.487	-3.202	-2.940
V(7)	0	Pass	0	0	0
V(8)	0.6749	Pass	0.6260	0.6516	0.6749
V(11)	0.6749	Pass	0.6260	0.6516	0.6749
V(12)	3.485	Pass	3.485	3.581	3.683
V(13)	3.128	Pass	3.128	3.166	3.207
V(16)	2.754	Pass	2.754	2.868	2.989
V(17)	1.950	Pass	1.950	2.077	2.213
V(18)	96.54m	Pass	92.50m	96.57m	0.1006
V(19)	97.49m	Pass	92.94m	97.01m	0.1011
V(20)	96.42m	Pass	91.93m	95.99m	0.1001
V(21)	98.03m	Pass	93.96m	98.03m	0.1021
V(VCC)	5.000	Pass	5.000	5.000	5.000
V(VEE)	-5.000	Pass	-5.409	-5.000	-4.591
@Vcc[i]	-86.55m	Pass	-97.20m	-91.67m	-86.55m

Report View: Measurement
Precision: 4
Variation: No Faults

- **Change the circuit temperature back to its nominal value (27 Degrees Celsius).**

- **Change the value of Vcc by plus and minus 5 percent and expand the tolerances to pass.**

You can do this by:

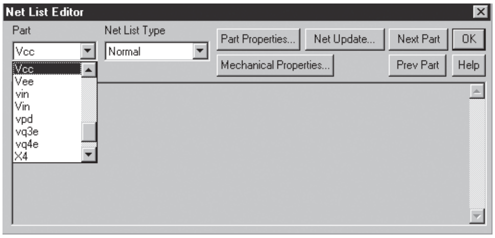
1. Define parametric “faults” for Vcc.
2. Manually changing Vcc DC value in the Part Properties dialog.

We’ll discuss how to perform the first of these activities in the following section.

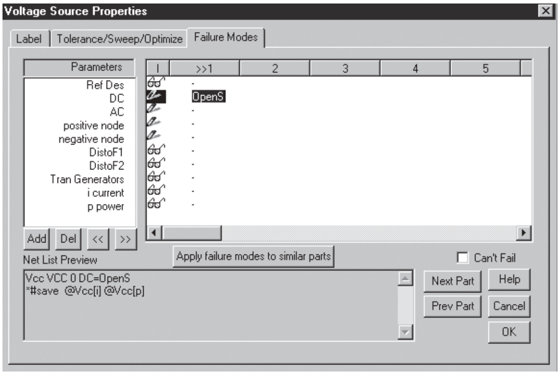
Defining Parametric Fault Variations

To change Vcc by introducing a “soft” out-of-tolerance fault:

- **Return to the Main tab in the Simulation Control dialog and click on the Parts button to open the Net List Editor dialog.**



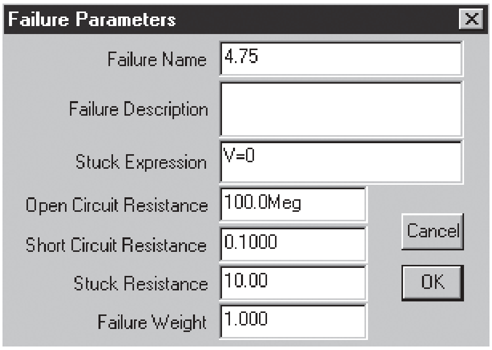
- **Select Vcc from the Part list box. Then, click on the Part Properties button to open the Voltage Source Properties dialog for Vcc.**
- **Click on the Failure Modes tab.**



```
Vcc VCC 0 DC=OpenS
*#save  @Vcc[i] @Vcc[p]
```

- **Click on the column 1 header (the “>>1” button at the top) in order to open the Failure Parameters dialog.**

- **Enter the name “4.75” in the Failure Name field.**



Failure Parameters

Failure Name	4.75
Failure Description	
Stuck Expression	V=0
Open Circuit Resistance	100.0Meg
Short Circuit Resistance	0.1000
Stuck Resistance	10.00
Failure Weight	1.000

Now the column 1 header now has the name “4.75.”

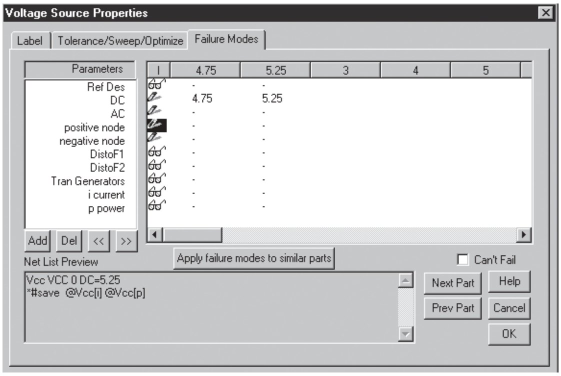
- **Then click on the OK button to dismiss the Failure Parameters dialog.**

- **Click on the value field for the “4.75” column, “DC” row entry and enter the Vcc fault value, 4.75.**

Low and high Vcc failure modes have just been defined to be 4.75 and 5.25 respectively.

- **Make sure that the Can’t Fail box is unchecked.**

- **Similarly, define another failure mode for the Vcc high voltage test (5.25) and click on the OK button to return to the Net List Edit dialog.**



Voltage Source Properties

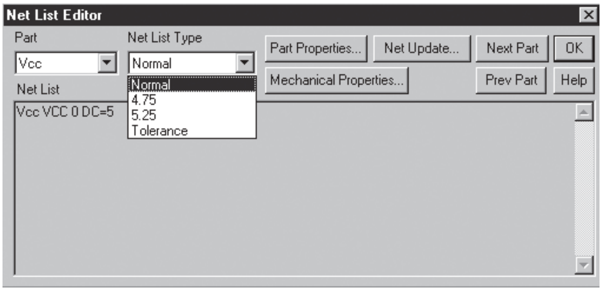
Parameters	4.75	5.25	3	4	5
Ref Des	-	-			
DC	4.75	5.25			
AC	-	-			
positive node	-	-			
negative node	-	-			
DistoF1	-	-			
DistoF2	-	-			
Tran Generators	-	-			
i current	-	-			
p power	-	-			

□ Can't Fail

Net List Preview

```
Vcc VCC 0 DC=5.25
*#save  @Vcc[i] @Vcc[p]
```

- **Ensure Vcc is selected in the Part pulldown. Click on the Netlist Type pulldown list to view the defined failure modes for Vcc.**



Part: Vcc

Net List Type: Normal

Normal
4.75
5.25
Tolerance

Net List:

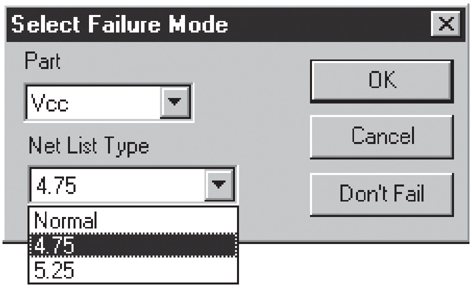
```
Vcc VCC 0 DC=5
```

- **Click on the OK button to exit the Net List Editor dialog.**

- **Click on the “Faults: (none)” button in the Main tab of the Simulation Control dialog.**

Since we were just working with Vcc, it is already selected in the Part list.

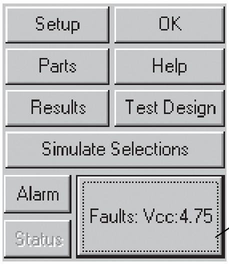
- **Select the “4.75” entry from the Net List Type list and click on the OK button to dismiss the dialog.**



Part: Vcc

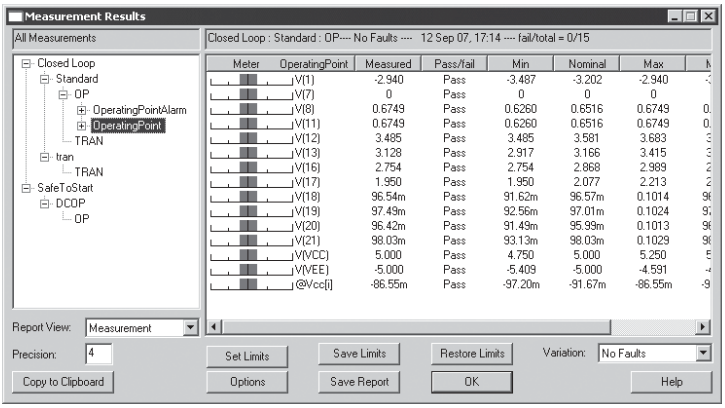
Net List Type: 4.75

Normal
4.75
5.25



Notice that the name of the Fault button has become “Faults:Vcc:4.75” in order to let you know which fault has been selected. It will be simulated this way when the simulator is launched.

- **Click on the Simulate Selections button.**
- **Click on the Results button and select the OperatingPoint branch. ICAP/4Windows just shows you the results for the last simulation. If you are using a higher package like ICAP/4Professional or Test Designer then in the variation drop down choose Vcc::4.75 to view the results. The higher packages accumulate results for multiple fault cases and allow you to switch between them without resimulating.**
- **Click on the Set Limits button and select the Expand to Pass function in the Tolerance section.**
- **Next, repeat this procedure for the 5.25 volt fault and rerun the simulation and expand the tolerances to pass.**



All Measurements

Closed Loop : Standard : OP---- No Faults ---- 12 Sep 07, 17:14 ---- fail/total = 0/15

```
⊟ Closed Loop
   ⊟ Standard
      ⊟ OP
         ⊞ OperatingPointAlarm
         ⊞ OperatingPoint
         TRAN
   ⊟ tran
         TRAN
⊟ SafeToStart
   ⊟ DCOP
         OP
```

Meter	OperatingPoint	Measured	Pass/fail	Min	Nominal	Max
	V(1)	-2.940	Pass	-3.487	-3.202	-2.940
	V(7)	0	Pass	0	0	0
	V(8)	0.6749	Pass	0.6260	0.6516	0.6749
	V(11)	0.6749	Pass	0.6260	0.6516	0.6749
	V(12)	3.485	Pass	3.485	3.581	3.683
	V(13)	3.128	Pass	2.917	3.166	3.415
	V(16)	2.754	Pass	2.754	2.868	2.989
	V(17)	1.950	Pass	1.950	2.077	2.213
	V(18)	96.54m	Pass	91.62m	96.57m	0.1014
	V(19)	97.49m	Pass	92.56m	97.01m	0.1024
	V(20)	96.42m	Pass	91.49m	95.99m	0.1013
	V(21)	98.03m	Pass	93.13m	98.03m	0.1029
	V(VCC)	5.000	Pass	4.750	5.000	5.250
	V(VEE)	-5.000	Pass	-5.409	-5.000	-4.591
	@Vcc[i]	-86.55m	Pass	-97.20m	-91.67m	-86.55m

Report View: Measurement

Precision: 4

Variation: No Faults

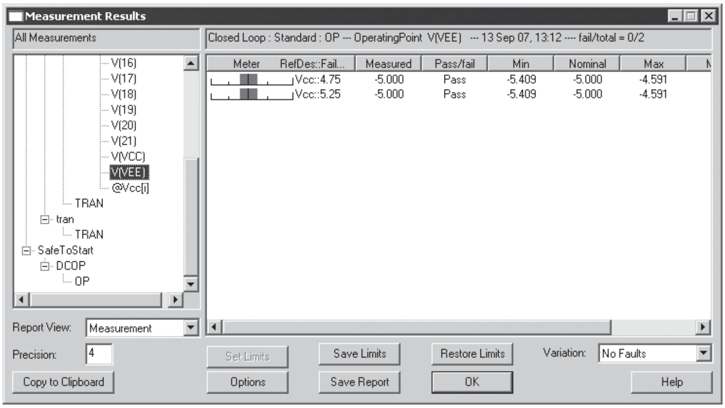
Notice that the Vcc limits are now 4.75 and 5.25 Volts, as expected. While Vcc didn't actually fail, the use of these values as pseudo-failures was a convenient method since we didn't have to change the Vcc value manually and then remember to restore it.

Another way of introducing parametric variations is by way of a Monte Carlo analysis.

- **Close the Measurement Results dialog (if it's still open), close the Simulation Control dialog and double-click on the Vee voltage source in the schematic.**
- **Select the Tolerance/Sweep/Optimize tab.**

Notice that it has already been done. A 5% tolerance has already been assigned to Vee. The Monte Carlo limits were already set in the previous “Running a standard Monte Carlo Analysis” section.

- **You can verify this by looking at the measurement limits for Vee in the Measurement Results dialog. In Test Designer and ICAP/4Professional you can select V(VEE), and view all the fault variations together.**



Closed Loop : Standard : OP --- OperatingPoint V(VEE) --- 13 Sep 07, 13:12 ---- fail/total = 0/2

Meter	RefDes::Fail...	Measured	Pass/fail	Min	Nominal	Max	N
	Vcc::4.75	-5.000	Pass	-5.409	-5.000	-4.591	
	Vcc::5.25	-5.000	Pass	-5.409	-5.000	-4.591	

Report View: Measurement

Precision: 4

Variation: No Faults

Notice that the tolerance is wider than 5% because we used 4.75 sigma limits to expand tolerances, but defined the tolerance using 3 sigma limits.

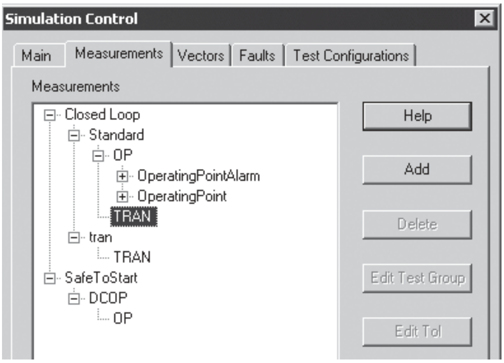
It is a commonly accepted practice to assign the 3 sigma limit to the manufacturer’s tolerance for a single component. However, when dealing with an assembly that must maintain a high degree of functional reliability when manufactured with these parts, the test limits must be greater than 3 sigma. Because of this, you may wish to assign a tolerance (in the Tolerance Properties dialog) that is less than the manufacturer’s limit in order to more closely approximate real-world behavior, rather than using the typical 3 sigma assignment.

Measuring Vectors

AC, DC, and TRAN simulations produce IsSpice4 vectors of a size greater than 1 because they include many data points. There are a set of common measurement functions that can be applied to these vectors. This set includes rise time, standard deviation and many others.

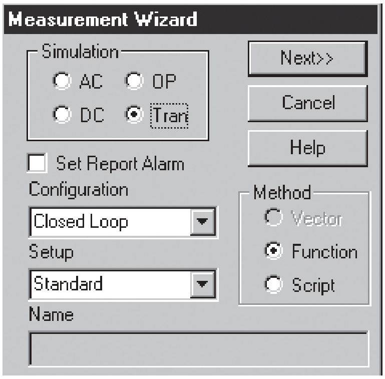
To perform these measurements:

- **Select the Measurements tab and highlight the "Closed Loop/Standard/TRAN" branch in the Measurements tree.**

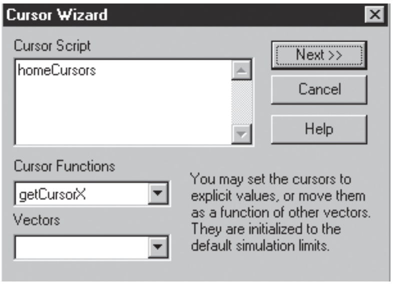


In the Method group, "Vector" is grayed out and "Function" is selected.

- **Click on the Add button to open the Measurement Wizard dialog.**



- **Click on the Next button to open the Cursor Wizard.**

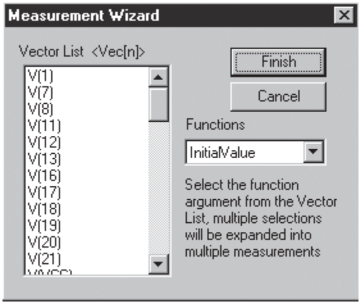


You may set the cursors to explicit values, or move them as a function of other vectors. They are initialized to the default simulation limits.

Cursor Basics:

A cursor is an abstract marker that is connected to the IsSpice4 default plot vector (i.e., x axis data, time for transient, and frequency for AC). Cursors are numbered, beginning with zero. There are two default cursors, zero (0) and one (1). Many measurement functions use these cursors. For example, standard deviation is measured between cursor 0 and cursor 1. The "homeCursors" function sets cursor 0 to the beginning of the plot, and sets cursor 1 to the end of the plot. This methodology allows you to record measurements across a set of waveforms. For more information about the cursors, press the Help button. For now, let's use the default "homeCursors."

- **Click on the Next button to open the Measurement Wizard.**



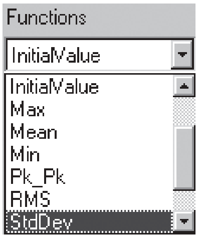
Select the function argument from the Vector List, multiple selections will be expanded into multiple measurements

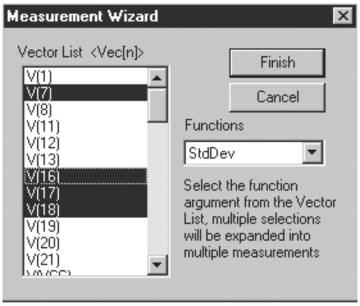
Standard deviation is the result you get when you use an AC-coupled true RMS meter, as the low pass bandwidth approaches zero.

To perform Standard Deviation Measurements:

- **Select "StdDev" from the Functions drop down list.**

- **Then, select the vectors for which we want to measure the standard deviation: v(7), v(16), v(17) and v(18).**





Select the function argument from the Vector List, multiple selections will be expanded into multiple measurements

- **Click on the Finish button to dismiss the dialog.**

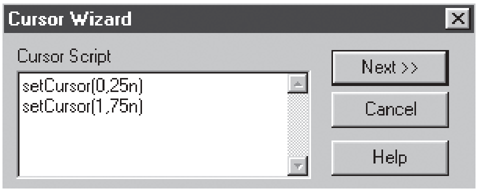
To Perform Rise Time Measurements:

Note: Make sure the TRAN leaf is still selected.

- **Click on the Add button in the Measurements tab.**

- **Click on the Next button in the Measurement Wizard.**

- **Replace the "homecursors" entry (in the Cursor Script edit box) with the following:**
  setCursor(0,25n)
  setCursor(1,75n)

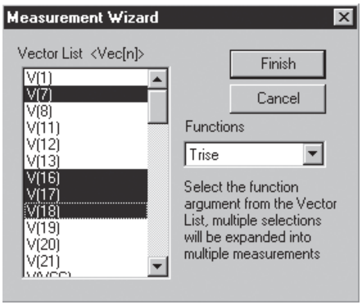


```
setCursor(0,25n)
setCursor(1,75n)
```

You may type the lines directly, or use the Cursor Functions list to paste the "setCursor" command into the Cursor Script edit box. This script sets the cursors to 25ns and 75ns, respectively. It is used to find the initial and final values on the waveform.

- **Click on the Next button to open the Measurement Wizard.**
- **Select "Trise" from the Functions drop down list.**
- **Then, select the vectors for which we want to measure the rise time: v(7), v(16), v(17) and v(18).**

- **Click on the Finish button to dismiss the dialog.**



Functions: Trise

Select the function argument from the Vector List, multiple selections will be expanded into multiple measurements

To make a scripted measurement:

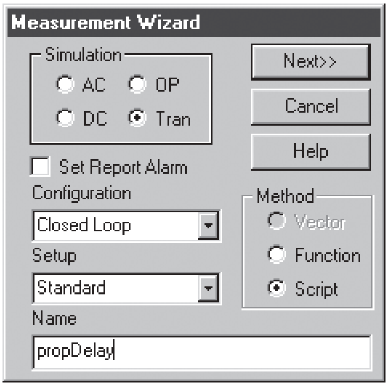
The full power of the IsSpice4 ICL (interactive command language) is at your disposal for making measurements. Several of the more common measurements are documented in the help dialog. In this example, we will perform a propagation delay measurement between the input at node 7 and the output at node 18.

- **Again, select the TRAN branch and press the Add button.**

- **Then, select "Script" radio button in the Method box and enter the name of your script, "propDelay" in the Name field.**

- **Click on the Next button.**

- **Click on the Help button, click on the Measurement Examples item at the top middle of the help screen, and select the Propagation Delay example.**



Simulation: ○ AC ○ OP ○ DC ● Tran

□ Set Report Alarm

Configuration: Closed Loop

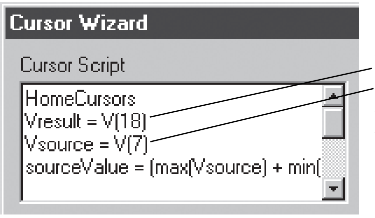
Setup: Standard

Method: ○ Vector ○ Function ● Script

Name: propDelay

- **Copy the Cursor Wizard Script (<Ctrl>+C) and paste (<Ctrl>+V) it into the Cursor Script edit box in place of the "homeCursors" script.**

- **Manually replace the "???" entries with v(18) and v(7), respectively.**



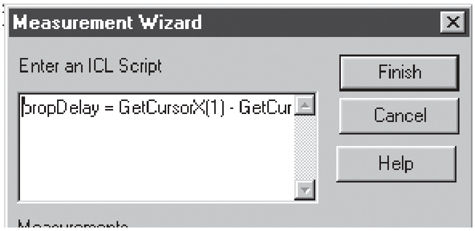
```
HomeCursors
Vresult = V(18)
Vsource = V(7)
sourceValue = (max(Vsource) + min(
```

The pasted script appears as follows:

```
HomeCursors
Vresult = ???
Vsource = ???
sourceValue = (max(Vsource) + min(Vsource))/2
resultValue = (max(Vresult) + min(Vresult))/2
MoveCursorRight(0, Vsource, sourceValue)
Tsource = GetCursorX(0)
SetCursor(1, Tsource)
MoveCursorRight(1, Vresult, resultValue)
```

- **Click on the Next button and copy and paste the Script Wizard script text under the Propagation Delay Example from the help dialog, into the “Enter an ICL Script” field.**

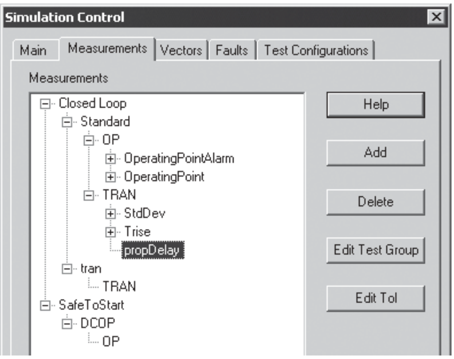
propDelay = GetCursorX(1) - GetCursorX(0)



Enter an ICL Script

```
propDelay = GetCursorX(1) - GetCur
```

- **Click on the Finish button to dismiss the dialog.**



Measurements

```
Closed Loop
  Standard
    OP
      OperatingPointAlarm
      OperatingPoint
    TRAN
      StdDev
      Trise
      propDelay
  tran
    TRAN
SafeToStart
  DCOP
    OP
```

To design new scripts, run the simulator in interactive mode and debug your script in the IsSpice4 interactive script window. The debugged script can be saved and later recalled using the IsEd text editor.

In order to get proper results, the test limits for all the added measurements should be established. Ordinarily, you would define all of the measurements and then set up tolerances so that the Monte Carlo and parameter variation simulations would be run only once. In our example, we will add tolerance directly.

- **Open the Result dialog, and individually select the desired branches.**

- **Click on the Set Limits button, select the Last Measurements from the Nominal group. Select Expand to Pass from the Tolerance group.**

- **Switch back to the Measurements tab and fold the measurements tree out to "Closed Loop/Standard/ TRAN/ StdDev/v(7)."**

- **Click on the Edit Tol button and select the absolute value for each of these measurements to be 2mv.**

- **Use the Next button to cycle through the measurements in each group.**

- **Repeat this process for Trise and prop delay, but use .5ns and 10%.**

Summary

You have established a baseline set of measurements and tolerance limits for a design. Make sure that you save the design example. You can now make design changes and see whether or not the design passes your performance tests. You can also simulate device failures individually, see if other parts are over stressed, and note if the failures are detectable by the tests that have been defined.

It should be noted that ICAP/4's Design Validator feature set is general in nature. The preceding capabilities allow you to make virtually any type of measurement and to compare the stored data with the results of virtually any design change or set of design changes.

If you have purchased the Test Designer package, please proceed to the next tutorial. This will show you how to automate many failure analysis steps, as well as synthesize and sequence tests into a fault tree.

Tutorial #8 - Test Designer

***Important*: This tutorial is for Test Designer users only.**

Overview

One of the most important applications of circuit testing is the detection and isolation of faulty components. Test Designer facilitates rapid development of effective detection and isolation testing strategies. It provides an environment for automated computation of a test's failure detection characteristics and automated generation of test sequences that are optimized to isolate failures using the minimum number of tests. This tutorial will introduce you to these powerful capabilities.

A testing strategy is commonly presented as a Fault Tree, which is similar in appearance to a hierarchical directory tree. Each node of the fault tree represents a test, and has one input connection and two output branches. The input contains a list of the faults (or remaining faults). Two output branches link to two tree nodes. One tree node defines the test with a PASS test outcome, and the other defines the test with a FAIL test outcome. Traversing the tree from the root node to a terminal leaf node of the fault tree, defines the sequence of tests and their associated PASS/FAIL outcomes. Through an extensive test algorithm, the process leads to isolation of one or more failure modes, or verification that no fault exists.

In a perfect world, a fault tree would contain all tests needed to uniquely isolate each component failure in a circuit. However, in the real world, this is not always possible. Therefore, a terminal leaf node of the fault tree may isolate to a group of failure modes, beyond which no further isolation is possible. This failure mode group is known as an Ambiguity Group, and contains the failure mode that actually exists at this terminal node of the test sequence. The term "ambiguity group" is often generalized to represent any group of failures, one of which caused traversal through the fault tree to the associated node.

A fault tree defines a test sequence that leads to one of several conclusions, e.g., the circuit contains no failed components, transistor Q5 has a shorted base/emitter, etc. The initial set of conclusions, which is operated on by the fault tree, consists of all defined component failure modes for the circuit, and the conclusion that no failures exist. Each test in a fault tree has an associated set of failures, or conclusions, that will cause it to have an outcome of FAIL. If a test fails, then the circuit failure mode is assumed to be among the set of failures that are detected by the test. If the test passes, the failure mode of the circuit (if there is one) will not be in the set of failure modes that are detected by the test. By applying this assumption on a test-by-test basis as the fault tree is traversed, it is possible to arrive at a fault isolation.

Consider a testing sequence as defined by a fault tree. The assumption going into the first test is that either there is no failure in the circuit under test, or one and only one of the failure modes defined for the circuit exists. We perform the first test defined at the root node of the fault tree. If the test fails, we eliminate all failures that are not detected by the test, plus the no-fault conclusion from the conclusion space. The conclusion space now consists of only those failure modes that are detected by the root node test. The “test failed” branch leading from the root node of the fault tree identifies the next test to be performed. However, the conclusion space for this second test consists only of the failure modes detected by the first test. If the second test also fails, the conclusion space is further reduced to the set of failures defined by the intersection of the set of failures in the conclusion space “handed” to the second test, and the set of failures detected by the second test.

If the first test (defined in the root node of the fault tree) passes, all detected failure modes from the first test are removed from the conclusion space. The “pass branch” from this node defines the next test to be performed. This second test then operates upon this reduced conclusion space.

Let’s use Test Designer to build a fault tree by completing the following steps:

Step	Description	Window/Dialog
1.	Capture a schematic drawing of the circuit.	Schematic
2.	Define or modify failure modes and tolerances, if required (reasonable defaults are provided).	Tolerance/Failure Modes tab
3.	Define the simulation setups (analyses that will be performed).	Simulation Setup
4.	Define the various test configurations in the schematic (pair schematic configurations with simulation setups).	Simulation Control/ Test configuration
5.	Define the measurements that will form the basis of individual tests. Typically, you will include more measurements than you will include in the final fault tree. Making extra information available here allows you to adjust the tests later, without having to re-run simulations in order to acquire more data.	Measurements tab
6.	Create test limits by adding tolerances that define the pass/fail boundaries for the measurements, directly or by using Monte Carlo, temperature variation, power supply variation, etc.	Results
7.	Simulate component and model, hard or soft parametric faults.	Simulation Control
8.	Iterate back through step 3 in order to add tests that detect non-convergent and overstressed failure modes. Later, these tests are used in order to detect these failure modes early in the test sequence, so that other parts in the circuit are not damaged due to the presence of one of these faults.	
9.	Adjust the fault modes so that the active modes are the ones that are being tested in each group. Remove faults that are not detectable or are not required for consideration - for example, open circuit failures for bypass capacitors.	Simulation Control/ Faults tab

Now you are ready to design a test sequence. This will be an iterative process that requires your judgment in order to make an optimal decision about the quality of the test flow, plus the number and types of tests and test points that will be used. It's usually best to select from all of the tests that you previously designated in order to see whether the tests are sufficient. In later phases, you may remove unwanted tests.

Step	Description	Window/Dialog
10.	Remove unnecessary tests.	Measurements tab
11.	Identify each test group and its measurement components. A test group stands alone. It may be part of a larger test, or it may be complete by itself. Examples include: Safe to Start (Safe-to-Turn-On); Product Acceptance Test; Depot Maintenance and Fault Isolation; and Built-In Test.	Measurements tab
12.	Choose a test “guard band;” this is a region located just outside of the test limits that will be used for screening tests that are being considered for assignment to a fault tree node. If any failure modes in the conclusion space passed to the test, cause a measurement value for a test to fall within the guard band, that test will not be selected for use at that node of the fault tree. The guard band ensures selection of tests that demonstrate a high detection confidence level. The software scales guardbands to the active measurement tolerance band. For example, a guard band of 0.5 (this is the default) is 1/2 the difference between the Max and Min test limit.	Test Design
13.	Decide on the measurement mode (tertiary is the default). There are four available measurement modes: Binary: The test fails if the measurement is above the upper limit or below the lower limit. Tertiary: The test fails high, low or passes. Histogram: The test detects failures that are within a histogram bin. Vector: The test detects failures centered about each known failed result.	Test Design
14.	Select tests from groups that can be run at the same time. A maximum entropy strategy is built in. The “Auto Build” function executes this strategy and can complete the fault tree at any time.	Test Design
15.	Iterate back through step 12 until the fault tree is complete.	
16.	Iterate back to step 10 until you are satisfied with each test flow (remove measurements and faults until you reach the optimum balance between fault detection and test complexity).	

Fault-tree based tests also provide fault detection for product acceptance tests. Each fault tree has an input, which is the list of failures that are to be detected, including “no fault.” Perfect fault detection requires the final tree exit to contain only the “no fault” failure mode. The best fault detection sequence is the one that arrives at this “no fault” conclusion with the least amount of work.

"No fault" is the situation in which the circuit functions properly.

Probability weight can be set using the Set Limits function in the Results dialog.

Test Designer accomplishes this task by sequencing tests based upon an entropy algorithm, which measures the probability of the pass/fail outcomes. For fault detection, setting the "no fault" probability weight to a high value will bias the test flow such that the no-fault outcome will occur with the minimum number of tests; this is the least amount of work. Fault isolation information is still available and can also be used to repair faulty units, or simply as quality control feedback if the circuit can't be repaired (e.g., an integrated circuit).

The philosophy of generating a test sequence based on known fault behavior assumes that most failure modes can be predicted, and that they tend to be catastrophic. The assumption doesn't need to be perfect in order to be effective; however, it must account for the majority of the failure rate. When most of the failure rate is predictable, then the fault isolation process will only be incorrect in unusual situations. In other words, the strategy will be correct most of the time, which is the desired result. Two special cases that are handled differently are:

1. Global parametric failures within IC's: For example, threshold voltage in a MOS circuit would cause multiple failures and would produce poor results using this strategy. The good news is that this kind of process failure doesn't need to be detected using circuit functional tests because it is detected using process control devices on the IC wafer. The circuit is presumed to be designed so that it will work when process variations are within acceptable limits.

2. When parametric culling is employed: An example of this is selecting from the central part of the statistical distribution in order to get high precision parts. The appropriate yield prediction technique is the Monte Carlo analysis. The division between catastrophic failures and functioning circuits is still well-defined using the fault model approach.

   Thus far, work that has been used to model IC failure modes from local defect models has confirmed the tendency toward catastrophic failure modes. This validates the fault model approach, which is used in Test Designer.

Beginning the Test Designer Tutorial

This tutorial is a continuation from the Design Validator tutorial, and assumes that you have walked through that tutorial.

In this Test Designer tutorial we begin at Step 11 (see the overview section of this tutorial) and make a single pass through the test sequence. We'll find several failure modes that must be tested in a "safe to start" test, and eliminate them from the first pass. You can take a look at the Sample3.DWG file (in the Sample3 directory) if you'd like to see how these tests were handled.

- **Open the Sample2a.DWG file (in the Circuits directory).**

Sample2a.DWG is the drawing that is obtained after the steps in the Design Validator tutorial have been performed on the sample2.DWG file.

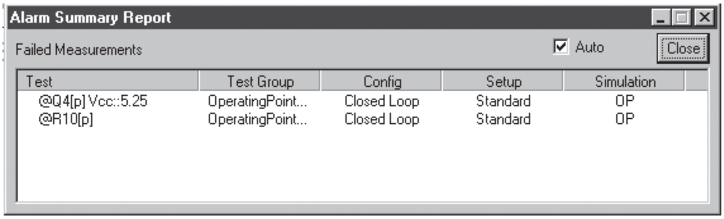
Setting Alarms, Faults, and Limits

- **Select the ICAPS function from the Actions menu to open the Simulation Control dialog.**

Notice that 2 alarms have been activated.

You can remove an alarm by changing its tolerance, or by removing the fault from the fault list.

- **Click on the Alarm button to view the alarm report.**



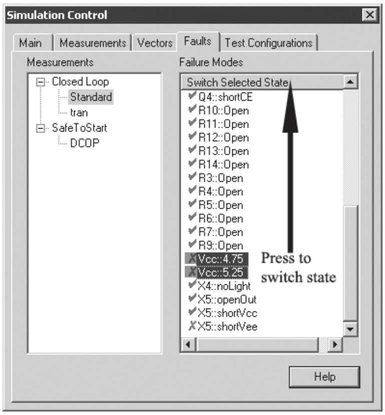
Alarm Summary Report

Failed Measurements ☑ Auto

Test	Test Group	Config	Setup	Simulation
@Q4[p] Vcc::5.25	OperatingPoint...	Closed Loop	Standard	OP
@R10[p]	OperatingPoint...	Closed Loop	Standard	OP

In order to remove an alarm, you must either change the tolerance or remove the fault from the fault list. Let's remove the failure for @Q4[p]Vcc:5.25.

The "Closed Loop/Standard" branch has already been selected for you.



Click on the Faults tab. Select the Closed Loop/Standard leaf in the Measurements tree Scroll down the failure modes list to select "Vcc::4.75"and "Vcc::5.25".

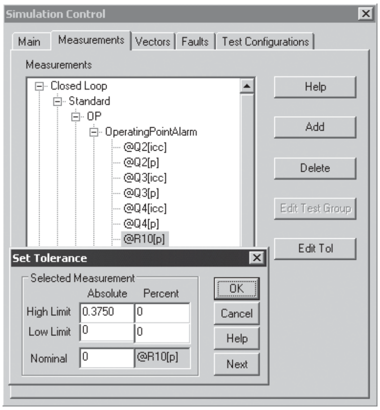
Click on the Switch Selected State button.

This will remove these faults from the active faults list for the selected test ("Closed Loop/Standard"). Notice that the "check" symbols have been changed to "X"s.

"Checks" indicate the faults that will be inserted when the simulations are performed. "X"s indicate faults that will not be simulated.

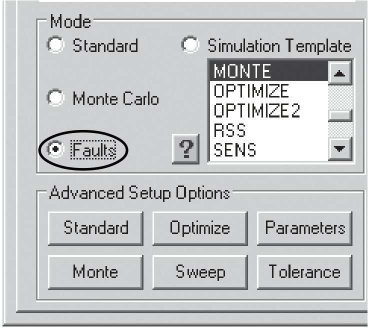
- **Click on the Main tab.**
- **Click on the Alarm button to view the Alarm summary report again.**

The "@Q4[p]Vcc:5.25" alarm has been deleted. Let's remove the "@R10[p]" fault from the Alarm window.



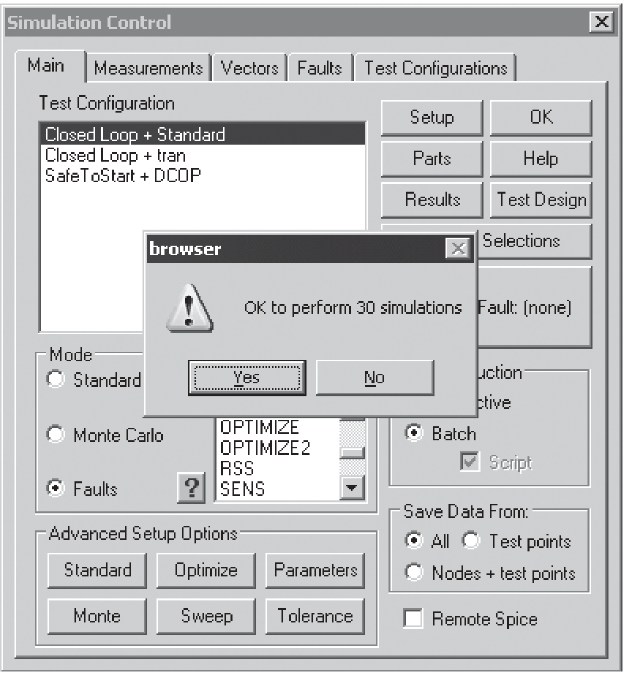
High Limit 0.3750 0
Low Limit 0 0
Nominal 0 @R10[p]

- **Click on the Close button to dismiss the alarm dialog.**
- **Click on the Measurements tab and expand the Closed Loop/Standard/OP/OperatingPointAlarm leaf.**
- **Select the "@R10[p]" entry.**
- **Click on the Edit Tol button and change the High limit (Absolute) to 0.3750.**
- **Click on the OK button to dismiss the dialog.**



- **Click on the Main tab.**
- **Click on the Faults radio button in the "Mode" group.**
- **Select the "Closed Loop + Standard" test configuration.**
- **Select Batch radio button.**
- **Click on the Simulate Selections button.**
- **Click on the Yes button.**

30 faults in the Faults tab were selected for this test.



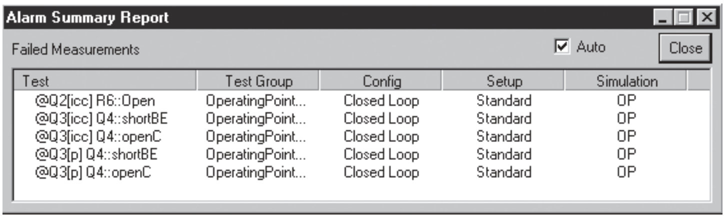
OK to perform 30 simulations

- **After simulation, click on the Yes button to perform the reference (nominal) simulation.**

Deleting Failure Modes that Overstress Components

The Alarm Summary Report dialog for the OP Alarm group of measurements is displayed when the fault simulations have been completed.

Notice that the first failure is Q2[icc], caused by the"R6::Open" failure mode.



Alarm Summary Report

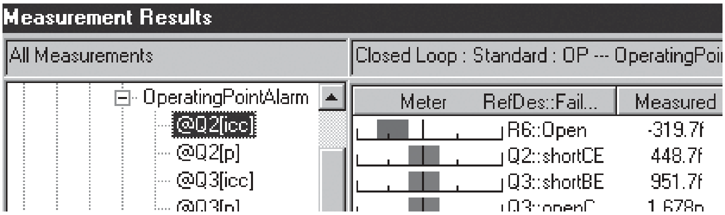
Failed Measurements ☑ Auto Close

Test	Test Group	Config	Setup	Simulation
@Q2[icc] R6::Open	OperatingPoint...	Closed Loop	Standard	OP
@Q3[icc] Q4::shortBE	OperatingPoint...	Closed Loop	Standard	OP
@Q3[icc] Q4::openC	OperatingPoint...	Closed Loop	Standard	OP
@Q3[p] Q4::shortBE	OperatingPoint...	Closed Loop	Standard	OP
@Q3[p] Q4::openC	OperatingPoint...	Closed Loop	Standard	OP

- **Click on the Results button and select the first failed measurement in the report, e.g., "@Q2[icc]" in the OperatingPointAlarm measurements group.**

The "R6::Open" failure mode makes this go a bit negative, so it's not really over stressed. The "R5::Open" test failed since the simulator was not able to converge onto a solution for this case.

You might need to move the Alarm Summary Report dialog out of the way so that you can see the Results dialog.



Measurement Results

All Measurements

Closed Loop : Standard : OP --- OperatingPoi

OperatingPointAlarm
@Q2[icc]
@Q2[p]
@Q3[icc]
@Q3[p]

Meter	RefDes::Fail...	Measured
	R6::Open	-319.7f
	Q2::shortCE	448.7f
	Q3::shortBE	951.7f
	Q3::openC	1.678n

- **Click on the Measurements tab and fold out the "Closed Loop/Standard/OP/OperatingPointAlarm" leaf.**
- **Select the "@Q2[icc]" entry and click on the Edit Tol button.**
- **Change its Low Limit (absolute) value to 1.000m so that it passes the test.**
- **Click on the OK button to dismiss the dialog.**
- **Select the Measurement Results dialog.**

Look in the Measurement Results dialog. The largest measured value for @Q3[icc] was about 0.1 volts ("Q4:ShortBE" failure mode). The largest measured value for @Q3[p] was about .24 volts ("Q4:ShortBE" failure mode).

Next, we'll alter the tolerances for "@Q3[icc]" and "@Q3[p]" so that these tests pass also.

- **Select the "@Q3[icc]" entry in the Measurements tab, and click on the Edit Tol button.**

- **Change the High Limit (absolute) value to 150m.**

The value of 150m was selected because the largest measured "@Q3[icc]" value was approximately 100m (see the "Q4:ShortBE" failure mode in the Measurement Results dialog).

- **Click on the OK button to dismiss the dialog.**

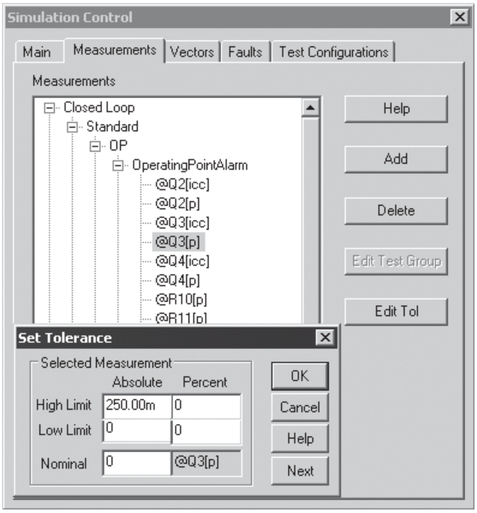
- **Select the "@Q3[p]" entry in the Measurements tab, and click on the Edit Tol button.**

- **Change the High Limit (absolute) value to 250m.**

The value of 250m was selected because the largest measured "@Q3[p]" value was approximately 240m (see the "Q4:ShortBE" failure mode in the Measurement Results dialog).

- **Click on the OK button to dismiss the dialog.**

The Edit Tol button is only available in the Measurements tab.



	Absolute	Percent
High Limit	250.00m	0
Low Limit	0	0
Nominal	0	@Q3[p]

Alarm summary shows that C2::open and R5::open failed. This is because the Test Failed to converge to a solution.

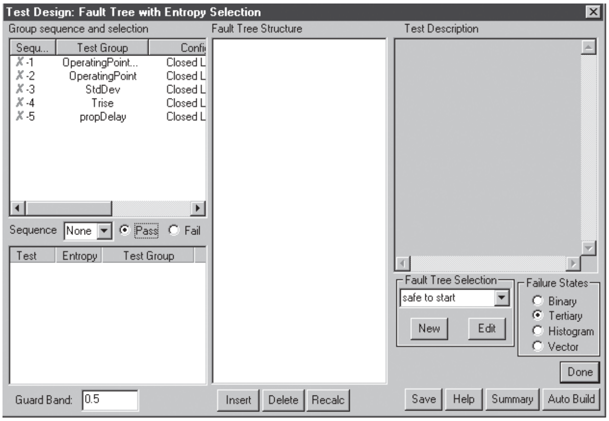
While the power and current are a bit excessive, they're not destructive.

<u>Note</u>: If the Measurement Results dialog was open when you changed tolerance limits, it must be refreshed by closing it and reopening it, or by selecting a different leaf.

- **Click on the Alarm button in the Main tab.**

Building the Fault Tree

- **Click on the Test Design button in the Main tab to open the Test Design dialog.**



Test Design: Fault Tree with Entropy Selection

Group sequence and selection

Sequ...	Test Group	Confi
✗-1	OperatingPoint...	Closed L
✗-2	OperatingPoint	Closed L
✗-3	StdDev	Closed L
✗-4	Trise	Closed L
✗-5	propDelay	Closed L

Sequence None ● Pass ○ Fail

Test	Entropy	Test Group

Fault Tree Selection: safe to start

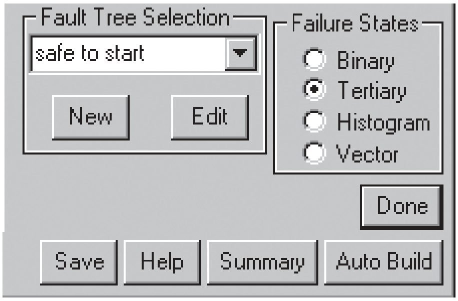
Failure States: ○ Binary ● Tertiary ○ Histogram ○ Vector

Guard Band: 0.5

Test Design windows and functions are documented in the on-line help.

- **Select the "Standard" test from the Fault Tree Selection list.**

If the list is empty, the Fault Tree dialog will automatically prompt you to enter a name.

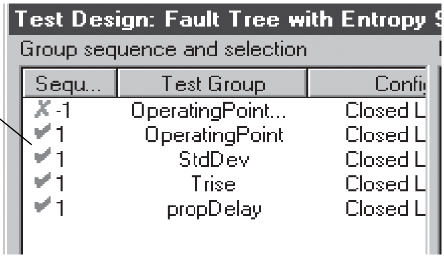


Fault Tree Selection: safe to start

Failure States: ○ Binary ● Tertiary ○ Histogram ○ Vector

- **Next, renumber the sequence numbers in the Group Sequence and Selection list, as shown below.**

To change a Sequence value, double-click on the number. This will activate an editing field. Then, enter a new value.



Test Design: Fault Tree with Entropy S

Group sequence and selection

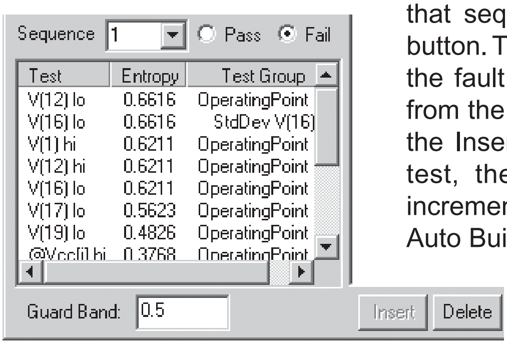
Sequ...	Test Group	Confi
✗ -1	OperatingPoint...	Closed L
✓ 1	OperatingPoint	Closed L
✓ 1	StdDev	Closed L
✓ 1	Trise	Closed L
✓ 1	propDelay	Closed L

- **Select the "1" entry in the "Sequence" drop-down list.**

- **Then click on the Auto Build button.**

The Auto Build button automatically creates the entire fault tree.

If desired, Test Designer also allows you to manually build the fault tree. You may choose a test sequence, select a test within that sequence and then click on the Insert button. The test information will be added to the fault tree, and the test will be removed from the test list. If you sequentially press the Insert button without first selecting a test, the fault tree will be created by incrementally stepping through the steps of Auto Build function.



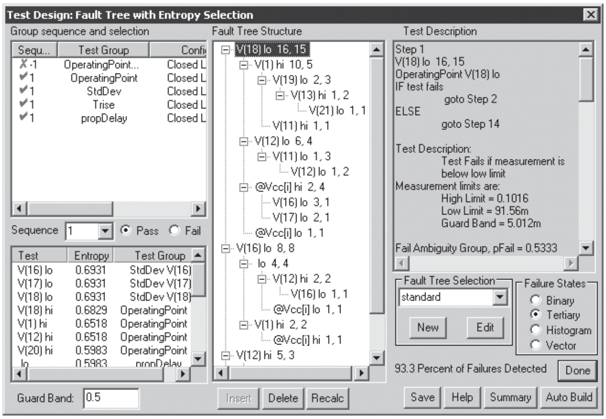
Sequence: 1 ○ Pass ● Fail

Test	Entropy	Test Group
V(12) lo	0.6616	OperatingPoint
V(16) lo	0.6616	StdDev V(16)
V(1) hi	0.6211	OperatingPoint
V(12) hi	0.6211	OperatingPoint
V(16) lo	0.6211	OperatingPoint
V(17) lo	0.5623	OperatingPoint
V(19) lo	0.4826	OperatingPoint
@Vcc[i] hi	0.3768	OperatingPoint

Guard Band: 0.5

The "Fault Tree Structure" labels the fault tree nodes starting with the test vector name, followed by the size of the pass, then the size of the fail ambiguity groups. When you select a fault tree node, its detailed information is displayed in the Test Description text field. The Save button creates an output file that can be used to code your test program. The Recalc button recalculates the fault tree. It's useful if you've made changes to the circuit or to measurement tolerances. A dialog will inform you if any changes have been made, and will suggest actions if the fault tree no longer makes any sense.

Notice that we achieved 93.9 percent failure coverage.



Test Design: Fault Tree with Entropy Selection

Group sequence and selection

Sequ...	Test Group	Confi
✗ -1	OperatingPoint...	Closed L
✓ 1	OperatingPoint	Closed L
✓ 1	StdDev	Closed L
✓ 1	Trise	Closed L
✓ 1	propDelay	Closed L

Sequence: 1 ● Pass ○ Fail

Test	Entropy	Test Group
V(16) lo	0.6931	StdDev V(16)
V(17) lo	0.6931	StdDev V(17)
V(18) lo	0.6931	StdDev V(18)
V(18) hi	0.6829	OperatingPoint
V(1) hi	0.6518	OperatingPoint
V(12) hi	0.6518	OperatingPoint
V(20) hi	0.5983	OperatingPoint
lo	0.5983	propDelay

Guard Band: 0.5

Fault Tree Structure

```
V(18) lo  16, 15
    V(1) hi  10, 5
        V(19) lo  2, 3
            V(13) hi  1, 2
                V(21) lo  1, 1
        V(11) hi  1, 1
    V(12) lo  6, 4
        V(11) lo  1, 3
            V(12) lo  1, 2
    @Vcc[i] hi  2, 4
        V(16) lo  3, 1
        V(17) lo  2, 1
    @Vcc[i] lo  1, 1
V(16) lo  8, 8
    lo  4, 4
        V(12) hi  2, 2
            V(16) lo  1, 1
        @Vcc[i] lo  1, 1
    V(1) hi  2, 2
        @Vcc[i] hi  1, 1
V(12) hi  5, 3
```

Test Description

```
Step 1
V(18) lo  16, 15
OperatingPoint V(18) lo
IF test fails
            goto Step 2
ELSE
            goto Step 14

Test Description:
            Test Fails if measurement is
            below low limit
Measurement limits are:
            High Limit = 0.1016
            Low Limit = 91.56m
            Guard Band = 5.012m

Fail Ambiguity Group, pFail = 0.5333
```

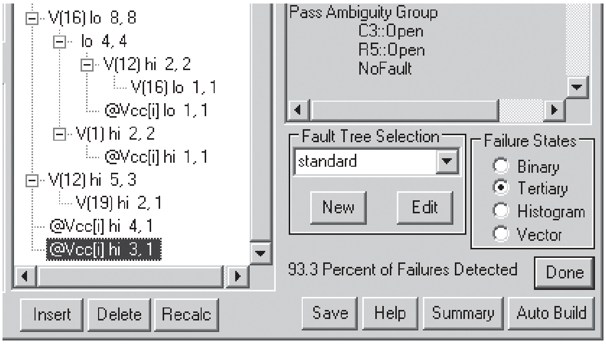
Fault Tree Selection: standard

Failure States: ○ Binary ● Tertiary ○ Histogram ○ Vector

93.3 Percent of Failures Detected

To see which failure modes haven't been detected:

- **Select the "Vcc[i]hi 3,1" item in the Fault Tree.**
- **Then scroll the Test Description list to view the "Pass Ambiguity Group" information.**



```
V(16) lo  8, 8
    lo  4, 4
        V(12) hi  2, 2
            V(16) lo  1, 1
        @Vcc[i] lo  1, 1
    V(1) hi  2, 2
        @Vcc[i] hi  1, 1
V(12) hi  5, 3
    V(19) hi  2, 1
@Vcc[i] hi  4, 1
@Vcc[i] hi  3, 1
```

```
Pass Ambiguity Group
            C3::Open
            R5::Open
            NoFault
```

Fault Tree Selection: standard

Failure States: ○ Binary ● Tertiary ○ Histogram ○ Vector

93.3 Percent of Failures Detected

Note: The Test Design dialog must be closed in order to access the Simulation Control dialog, etc.

The "C3::Open" and "R5::Open" failure modes were not detected. It can be seen that R5 isn't really needed to bias Q3 in this test. In order to detect "Q3::Open," the laser would have to be extinguished. The feedback would prevent us from setting up a meaningful test, so we should accept "R5::Open" as an uninteresting failure, or devise an open loop test. C3 and C2 affect stability. Open circuit failures could make the circuit oscillate, and possibly destroy the laser diode. We need a test that performs a controlled start-up so we can see the onset of instability. Applying power, monitoring peak-to-peak ripple and maximum laser current will probably do the trick. Next, we need a safe-to-start test in order to detect the faults that might cause an overstress condition (indicated by the previous alarms).

- **Click on the Done button to dismiss the Test Design dialog.**

Adding Safe-to-Start Tests

We already have a test configuration for this case. It reduces Vcc to 2 volts and uses a current limited power supply to protect the laser diode.

- **Click on the Measurements tab and select the "SafeToStart/DCOP/OP" leaf.**

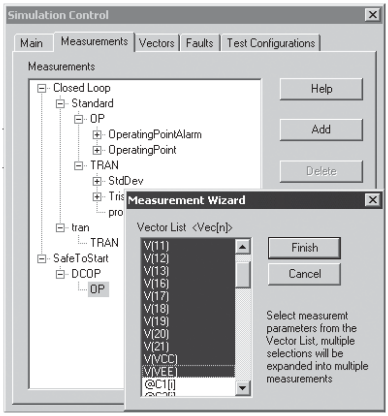
- **Click on the Add button.**

- **Click on the Next button.**

- **Select all of the node voltages in the vector list (e.g., V(1) through V(VEE) inclusive).**

- **Click on the Finish button.**

- **Click on the Main tab then click on the OK button to dismiss the Simulation Control dialog.**



Select measuremt parameters from the Vector List, multiple selections will be expanded into multiple measurements

You can define a tolerance for the current limited power supply so that the Monte Carlo analysis will provide all of the test limits based upon this tolerance variation.

- **Select the SafeToStart configuration from the Options/ Configurations menu.**

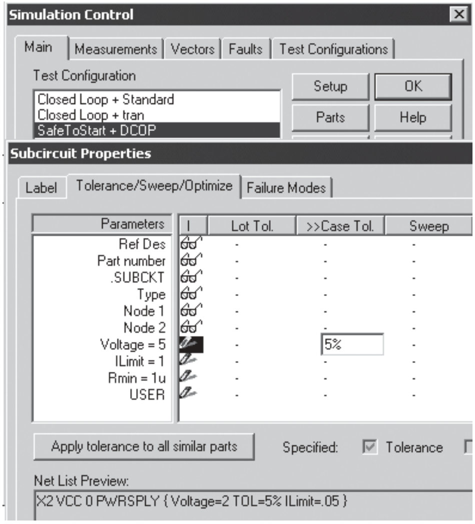
- **Double-click on X2, the power supply in the upper left corner of the schematic.**

- **Click on the Tolerance/Sweep/Optimize tab.**

- **Click on the "Case" column in the box, which is directly across from the "Voltage" parameter.**

- **Enter "5%" in that box and click on the OK button.**

We've entered a 5% tolerance for the output voltage of power supply X2



Test Configuration

Closed Loop + Standard
Closed Loop + tran
SafeToStart + DCOP

Parameters	I	Lot Tol.	>>Case Tol.	Sweep
Ref Des		-	-	-
Part number		-	-	-
.SUBCKT		-	-	-
Type		-	-	-
Node 1		-	-	-
Node 2		-	-	-
Voltage = 5		-	5%	-
ILimit = 1		-	-	-
Rmin = 1u		-	-	-
USER		-	-	-

Specified: ☑ Tolerance □

Net List Preview:
X2 VCC 0 PWRSPLY { Voltage=2 TOL=5% ILimit=.05 }

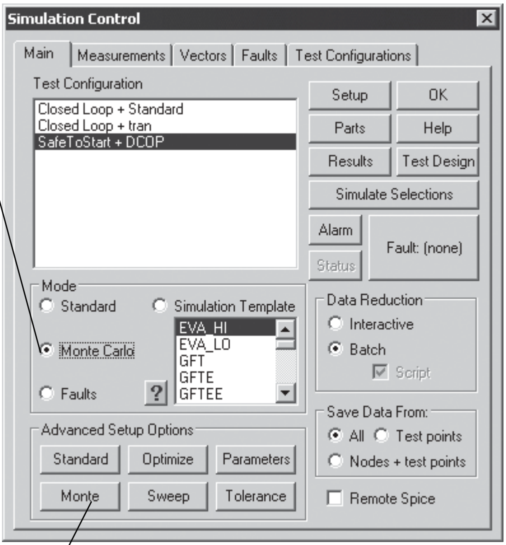
Next, let's set the test tolerances based upon the Monte Carlo results, just as we did in the previous Design Validator tutorial.

- **Select the ICAPS function from the Actions menu.**

- **Select the "SafeToStart + DCOP" test configuration.**

- **Select the standard Monte Carlo Mode in the Main tab.**

Make sure that the “SafeToStart + DCOP” test configuration is selected.



Test Configuration

Closed Loop + Standard
Closed Loop + tran
SafeToStart + DCOP

Mode

○ Standard ○ Simulation Template
● Monte Carlo
○ Faults

EVA_HI
EVA_LO
GFT
GFTE
GFTEE

Data Reduction

○ Interactive
● Batch
☑ Script

Save Data From:

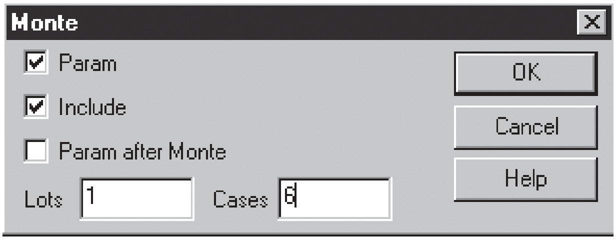
● All ○ Test points
○ Nodes + test points

☐ Remote Spice

- **Click on the Monte button.**

Entering a value of 6 here will cause six simulations to be performed when the Monte Carlo analysis is launched.

- **Change the number of Cases to 6 and click on the OK button to dismiss the dialog.**



☑ Param

☑ Include

☐ Param after Monte

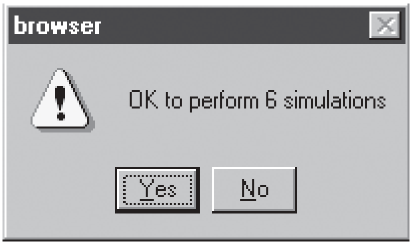
Lots 1 Cases 6

- **Make sure that the Batch (in the Data Reduction group) are selected.**

We’re now ready to run the Monte Carlo analysis.

- **Click on the Simulate Selections button to launch the simulations.**

The following dialog appears. Because simulations may sometimes require large amounts of time, you are given the option to continue or to cancel the simulation request.


browser

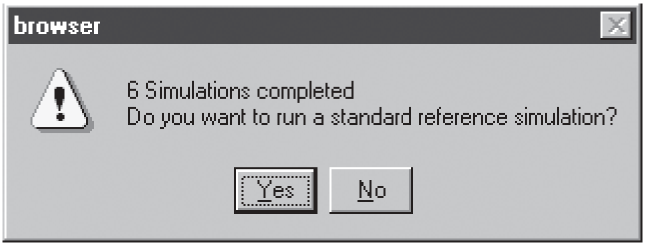
OK to perform 6 simulations

Vectors V(1) through V(VEE) will be measured during this test.

- **Click on the Yes button to run the simulations.**

Since the Batch mode radio button was selected before the simulations were launched, the six simulations are run in the background in order to optimize the simulation time. The simulator screen is not displayed and updated as the simulations proceed, but the "Simulation progress" dialog appears in order to inform you of the progress as the simulations are performed.

After all of the simulations have been run, you will be asked if you want to run a standard reference simulation. If you select Yes, an additional simulation will be performed, and the data from that run may then be used to set the nominal measurement limits.


browser

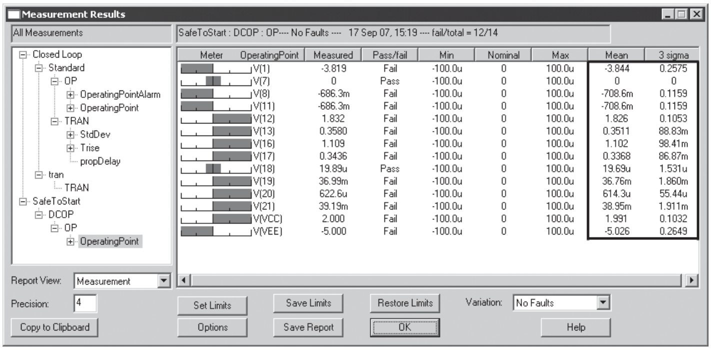
6 Simulations completed
Do you want to run a standard reference simulation?

- **Click on the Yes button so that we'll have a nominal simulation saved for the "Last Measurements" data.**

- **Click on the Results button in the Main tab.**

- **Select the "SafeToStart/DCOP/OP/OperatingPoint" leaf in the measurement tree.**

- **Scroll the report to the right and view the Monte Carlo statistics.**



SafeToStart : DCOP : OP---- No Faults ---- 17 Sep 07, 15:19 ---- fail/total = 12/14

Meter	OperatingPoint	Measured	Pass/fail	Min	Nominal	Max	Mean	3 sigma
	V(1)	-3.819	Fail	-100.0u	0	100.0u	-3.844	0.2575
	V(7)	0	Pass	-100.0u	0	100.0u	0	0
	V(8)	-686.3m	Fail	-100.0u	0	100.0u	-708.6m	0.1159
	V(11)	-686.3m	Fail	-100.0u	0	100.0u	-708.6m	0.1159
	V(12)	1.832	Fail	-100.0u	0	100.0u	1.826	0.1053
	V(13)	0.3580	Fail	-100.0u	0	100.0u	0.3511	88.83m
	V(16)	1.109	Fail	-100.0u	0	100.0u	1.102	98.41m
	V(17)	0.3436	Fail	-100.0u	0	100.0u	0.3368	86.87m
	V(18)	19.89u	Pass	-100.0u	0	100.0u	19.69u	1.531u
	V(19)	36.99m	Fail	-100.0u	0	100.0u	36.76m	1.860m
	V(20)	622.6u	Fail	-100.0u	0	100.0u	614.3u	55.44u
	V(21)	39.19m	Fail	-100.0u	0	100.0u	38.95m	1.911m
	V(VCC)	2.000	Fail	-100.0u	0	100.0u	1.991	0.1032
	V(VEE)	-5.000	Fail	-100.0u	0	100.0u	-5.026	0.2649

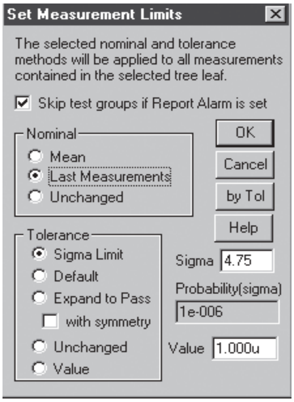
Report View: Measurement

Precision: 4

Variation: No Faults

- **Click on the Set Limits button.**

- **Select the Last Measurements button in the Nominal group and Sigma Limit in the Tolerance group.**



The selected nominal and tolerance methods will be applied to all measurements contained in the selected tree leaf.

☑ Skip test groups if Report Alarm is set

Nominal
○ Mean
● Last Measurements
○ Unchanged

Tolerance
● Sigma Limit
○ Default
○ Expand to Pass
□ with symmetry
○ Unchanged
○ Value

Sigma 4.75

Probability(sigma) 1e-006

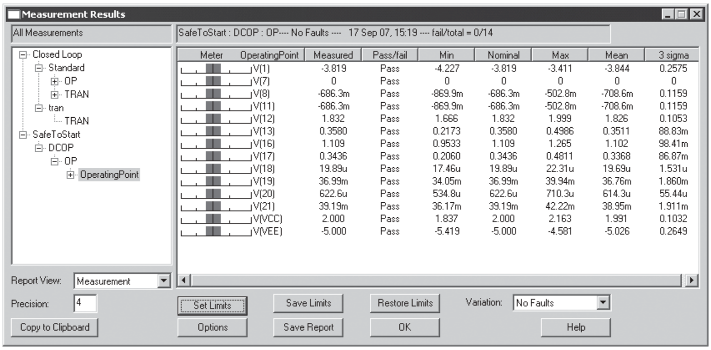
Value 1.000u

- **Click on the OK button to dismiss the Set Measurement Limits dialog.**

The default Sigma value is 4.75, but you may change it. The probability calculator shows the probability for one side of a normal distribution.

The tolerances for all measurements are now set, based upon the Monte Carlo results.

The test limits are now set using the Monte Carlo analysis values (mean and 4.75 sigma).



SafeToStart : DCOP : OP---- No Faults ---- 17 Sep 07, 15:19 ---- fail/total = 0/14

Meter OperatingPoint	Measured	Pass/fail	Min	Nominal	Max	Mean	3 sigma
V(1)	-3.819	Pass	-4.227	-3.819	-3.411	-3.844	0.2575
V(7)	0	Pass	0	0	0	0	0
V(8)	-686.3m	Pass	-869.9m	-686.3m	-502.8m	-708.6m	0.1159
V(11)	-686.3m	Pass	-869.9m	-686.3m	-502.8m	-708.6m	0.1159
V(12)	1.832	Pass	1.666	1.832	1.999	1.826	0.1053
V(13)	0.3580	Pass	0.2173	0.3580	0.4986	0.3511	88.83m
V(16)	1.109	Pass	0.9533	1.109	1.265	1.102	98.41m
V(17)	0.3436	Pass	0.2060	0.3436	0.4811	0.3368	86.87m
V(18)	19.89u	Pass	17.46u	19.89u	22.31u	19.69u	1.531u
V(19)	36.99m	Pass	34.05m	36.99m	39.94m	36.76m	1.860m
V(20)	622.6u	Pass	534.8u	622.6u	710.3u	614.3u	55.44u
V(21)	39.19m	Pass	36.17m	39.19m	42.22m	38.95m	1.911m
V(VCC)	2.000	Pass	1.837	2.000	2.163	1.991	0.1032
V(VEE)	-5.000	Pass	-5.419	-5.000	-4.581	-5.026	0.2649

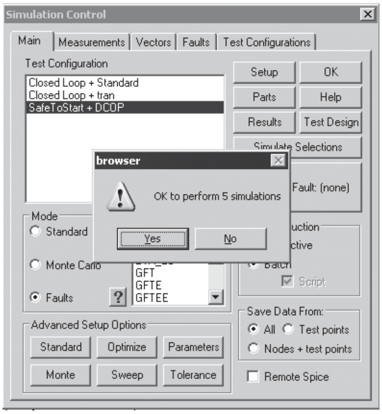
__Note:__ Be sure to close any Measurement Results dialogs before you run the simulations, or the results update might make the simulation run time very long.

- **Click on the OK button to close the Measurement Results dialog.**

Next, let's run the Test Designer simulations to get the fault data.

- **Make sure that the "SafeToStart + DCOP" test configuration is selected.**
- **Click on the Faults radio button in the Mode section.**
- **Click on the Simulate Selections button to run the Test Designer simulations.**
- **Click on the Yes button to run the 5 simulations.**
- **Click on the Yes button to run the reference simulation.**
- **Click on the OK button to dismiss the Simulation Control dialog.**

There are five (5) failure modes that will be inserted for the SafeToStart tests: C1 short, Q2 short BE, Q3 short CE, R5 open, and X5 short Vee (look in the Faults tab).



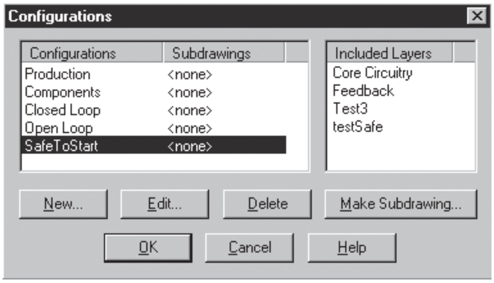
OK to perform 5 simulations

Adding a Special Non-Destructive Test

In order to detect the “C2::Open” and “C3::Open” failure modes, we need to add another test. This test will merely apply power using a ramped power supply. We’ll monitor the voltage at the summing junction, and the error voltage of the op amp. We’ll create a new configuration for this test.

- **Select Configurations/Edit from the Options menu.**

These are the existing configurations.



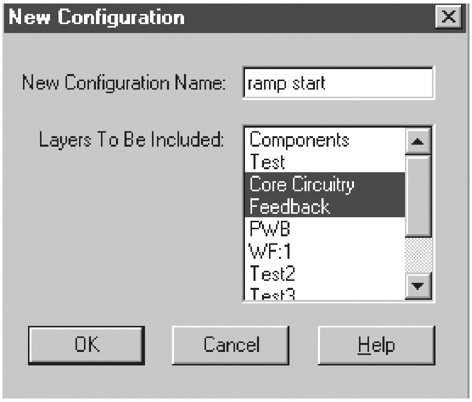
Configurations	Subdrawings
Production	<none>
Components	<none>
Closed Loop	<none>
Open Loop	<none>
SafeToStart	<none>

Included Layers:
Core Circuitry
Feedback
Test3
testSafe

- **Click on the New button.**

- **Enter the name “ramp start” in the New Configuration Name field.**

- **Deselect all layers but the “Core Circuitry” and “Feedback” layers.**



New Configuration Name: ramp start

Layers To Be Included:
Components
Test
Core Circuitry
Feedback
PWB
WF:1
Test2
Test3

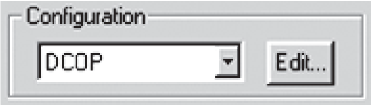
- **Click on the OK button to dismiss the New Configuration dialog.**

- **Click on the OK button to dismiss the Configurations dialog.**

Next, we'll create a new simulation setup called "ramp."

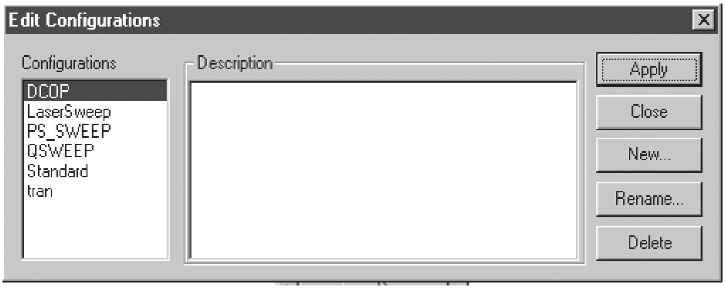
- **Select the Simulation Setup/Edit function from the Actions menu.**

- **Click on the Edit button.**



Configuration: DCOP

- **Click on the New button.**

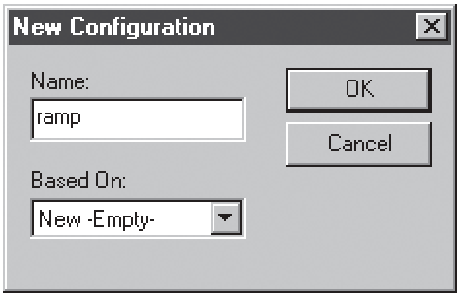


Edit Configurations

Configurations: DCOP, LaserSweep, PS_SWEEP, QSWEEP, Standard, tran

Description:

- **Enter the name "ramp" in the Name field.**



New Configuration

Name: ramp

Based On: New -Empty-

- **Click on the OK button to dismiss the New Configuration dialog.**

- **Click on the Apply button to dismiss the Edit Configurations dialog.**

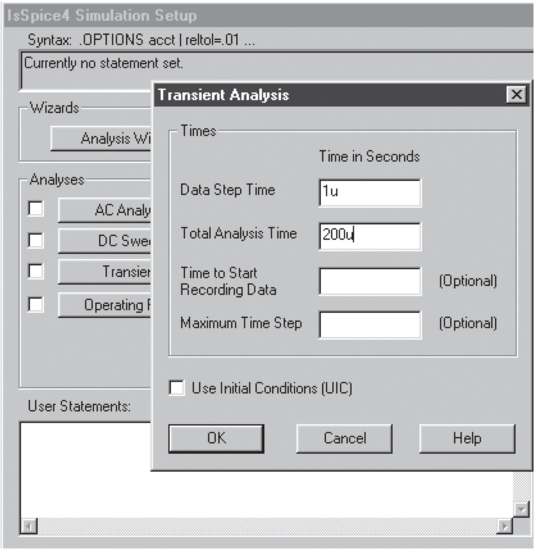
Now, let's add the definition for the new "ramp" simulation setup that we've just created.

- **Click on the Transient button in the IsSpice4 Simulation Setup dialog.**

- **Enter a Data Step Time value of "1u" and a Total Analysis Time value of "200u."**

- **Click on the OK button to dismiss the Transient Analysis dialog.**

The "ramp" simulation setup will perform a 200 microsecond transient analysis.



IsSpice4 Simulation Setup

Syntax: .OPTIONS acct | reltol=.01 ...

Currently no statement set.

Transient Analysis

Data Step Time: 1u

Total Analysis Time: 200u

Time to Start Recording Data: (Optional)

Maximum Time Step: (Optional)

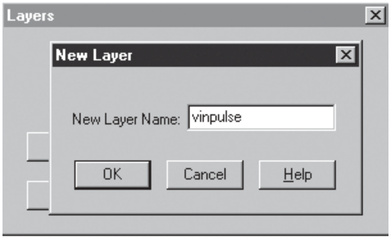
□ Use Initial Conditions (UIC)

- **Click on the Done button to dismiss the IsSpice4 Simulation Setup dialog.**

Next, we'll create a new schematic layer called "vinpulse."

- **Select the Layers function from the Options menu.**

- **Click on the New button.**



New Layer Name: vinpulse

- **Enter the name "vinpulse" into the New Layer Name field.**

- **Click on the OK button to dismiss the New Layer dialog.**

- **Click on the OK button to dismiss the Layers dialog.**

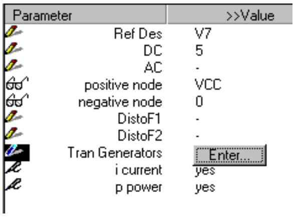
On this new "vinpulse" layer, we'll connect a new VCC source and a new VEE source. These two sources are pulse sources with 100usec rise times. The "ramp start" configuration will then contain the core circuitry, feedback and vinpulse layers.

This will place voltage source V7 on the VCC node, and place a ground connection on its low side.

- **Click on the VCC node in the schematic.**

- **Select the Voltage Source function from the Parts menu.**

  - **Select the Ground function from the Parts menu.**

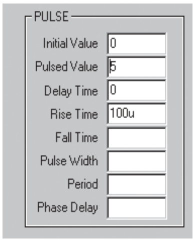


Parameter	>>Value
Ref Des	V7
DC	5
AC	-
positive node	VCC
negative node	0
DistoF1	-
DistoF2	-
Tran Generators	Enter...
i current	yes
p power	yes

  - **Double-click on V7 and enter a DC value of 5.**

  - **Click on the Tran Generators value field.**

  - **Click on the Enter button.**



PULSE

Initial Value	0
Pulsed Value	5
Delay Time	0
Rise Time	100u
Fall Time	
Pulse Width	
Period	
Phase Delay	

- **Enter the following values into the first four parameter fields in the Pulse tab: 0,5,0 and 100u.**

- **Click on the OK button to dismiss the Transient Generators dialog.**

- **Click on the OK button to dismiss the properties dialog.**

- **Click on the left pin of capacitor C1.**

This will create R15, rotate it, and connect it to ground.

- **Select the Passive/Resistor function from the Parts menu.**

- **Select the Rotate function from the Edit menu.**

- **Select the Ground function from the Parts menu.**

- **Double-click on R15, then select the Label tab and enter a value of 0.1.**

- **Click on the OK button to dismiss the dialog.**

This will create a continuation node called "VEE", place voltage source V8 on the VEE node, and place a ground connection on its low side.

- **Click on an empty area of the schematic (about 1/2 inch below the ground for R15).**

- **Select the Continuation function from the Parts menu.**

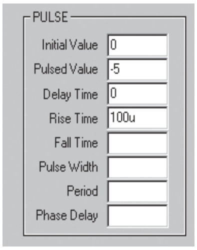
- **Select the Voltage Source function from the Parts menu.**

- **Enter "VEE" in the continuation's name field and click on the OK button.**

- **Select the Ground function from the Parts menu.**

- **Double-click on V8 and enter a value of -5.**

- **Click on the Tran Generators value field and then click on the Enter button.**



PULSE

Initial Value	0
Pulsed Value	-5
Delay Time	0
Rise Time	100u
Fall Time	
Pulse Width	
Period	
Phase Delay	

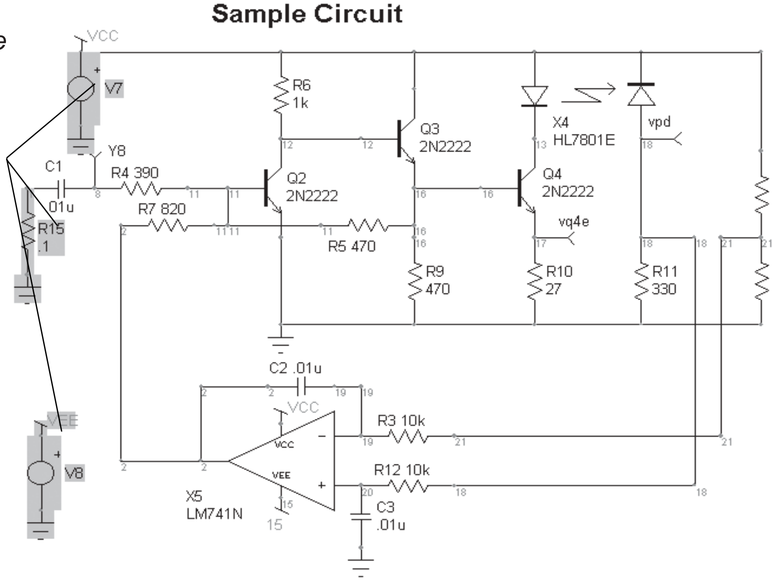
- **Enter the following values into the first four parameter fields in the Pulse tab for V8: 0,-5,0 and 100u.**

- **Click on the OK button to dismiss the Transient Generators dialog.**

- **Click on the OK button to dismiss the Properties dialog.**

The schematic will then appear as shown below.

These are the parts that we have just added.



Next, we'll add the new configuration, "ramp start + ramp" to the list of available test configurations.

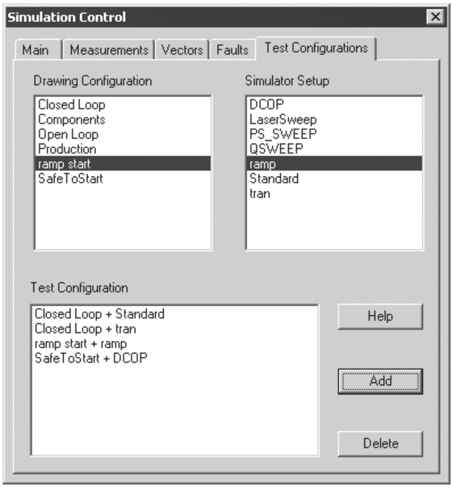
- **Select the ICAPS function from the Actions menu.**

- **Click on the Test Configurations tab.**

- **Select the "ramp start" drawing configuration and the "ramp" Simulator setup.**

- **Click on the Add button.**

These are all of the available test configurations



Drawing Configuration

- Closed Loop
- Components
- Open Loop
- Production
- ramp start
- SafeToStart

Simulator Setup

- DCOP
- LaserSweep
- PS_SWEEP
- QSWEEP
- ramp
- Standard
- tran

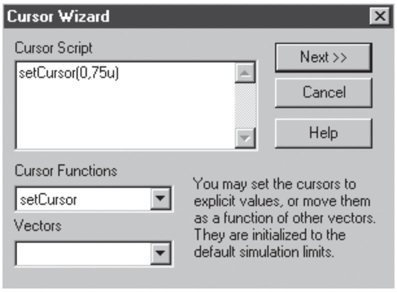
Test Configuration

- Closed Loop + Standard
- Closed Loop + tran
- ramp start + ramp
- SafeToStart + DCOP

Now we'll define a measurement for this configuration.

- **Click on the Measurements tab.**

Note: A dialog will inform you if any component's reference designator has changed. If this happens, dismiss the dialog.



Cursor Script: setCursor(0,75u)

Cursor Functions: setCursor

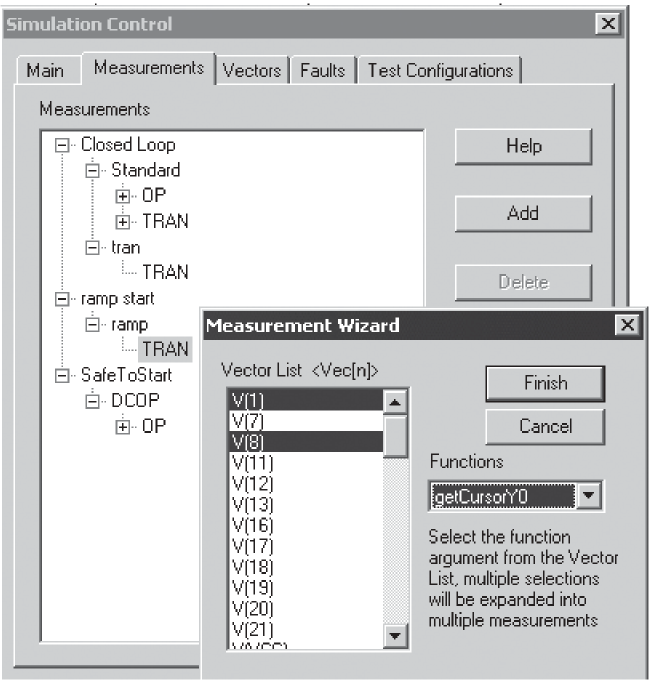
You may set the cursors to explicit values, or move them as a function of other vectors. They are initialized to the default simulation limits.

- **Select the "ramp start/ramp/TRAN" leaf and click on the Add button.**
- **Click on the Next button.**
- **Delete the "home Cursors" script from the Cursor Script window, and enter "setCursor(0,75u)".**

This will set cursor zero's position to 75u.

- **Click on the Next button.**
- **Select vectors V(1) and V(8) from the Vector List.**
- **Select the "getCursorY0" function from the Functions list.**

V(1) and V(8) will be measured during this test.



Functions: getCursorY0

Select the function argument from the Vector List, multiple selections will be expanded into multiple measurements

- **Click on the Finish button.**

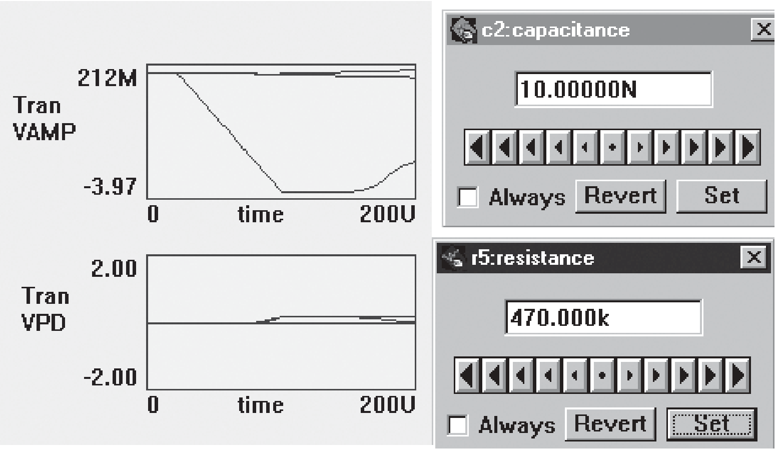
Note that for your final test, you'll want to turn off power after 100us in order to protect against excessive current through the laser diode.

Using IsSpice4 Interactively to Determine Tolerances

Interactive parameter variation is discussed in the IsSpice4 Tutorial in the Getting Started manual.

You can use IsSpice4 to interactively vary parameters in order to observe the effects of troublesome failure modes (e.g., "R5::Open"). The Persistence function in IsSpice4 lets you superimpose the different results in order to see if there is a good discriminant. This approach was used to decide which test point(s) should be used, and at what time the measurements should be made.

C2 and R5 values were changed in order to determine which test point and at what time the measurements should be made.



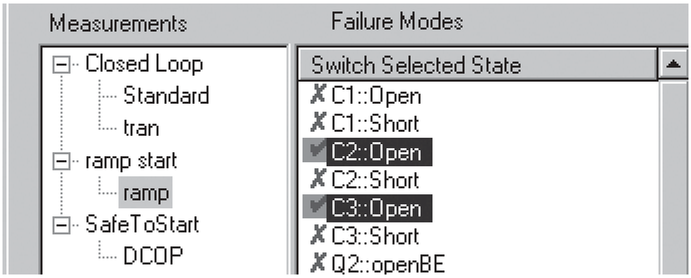
c2:capacitance: 10.00000N

r5:resistance: 470.000k

We are only interested in "C2::Open," "C3::Open" and "R5::Open" for this test.

- **Select the Faults tab and select the "ramp start/ramp" leaf.**
- **Select all of the faults in the Failure Modes list.**
- **Click on the Switch Selected State button to turn off all of the failure modes.**
- **Select "C2::Open," "C3::Open" and "R5::Open," then click on the Switch Selected State button again in order to activate these faults.**

Failure modes C2::Open, C3::Open and R5::Open will be inserted for the ramp test.

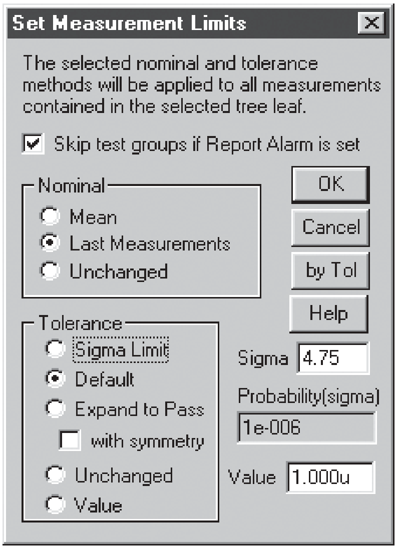


Making the Final Product Acceptance Test

Next, we'll run the Test Designer simulations for the 3 faults and then run a reference simulation.

- **Click on the Main tab.**
- **Select the "ramp start + ramp" test configuration.**
- **Make sure that the Faults mode button is selected, then click on the Simulate Selections button.**
- **Click on the Yes button to run the 3 simulations.**
- **Click on the Yes button to run the reference simulation.**
- **Click on the Results button and select the "ramp start/ ramp/TRAN/getCursorY0" leaf.**
- **Click on the Set Limits button.**

Limits that are set within the Measurement Results dialog apply to all tests within the selected leaf.


Set Measurement Limits

The selected nominal and tolerance methods will be applied to all measurements contained in the selected tree leaf.

☑ Skip test groups if Report Alarm is set

Nominal
○ Mean
● Last Measurements
○ Unchanged

Tolerance
○ Sigma Limit
● Default
○ Expand to Pass
□ with symmetry
○ Unchanged
○ Value

Sigma 4.75
Probability(sigma) 1e-006
Value 1.000u

- **Set the Set Limits options to Last Measurement for nominal and Default for tolerance.**

- **Click on the OK button to close the Set Measurement Limits dialog.**

- **Click on the OK button to dismiss the Measurement Results dialog.**

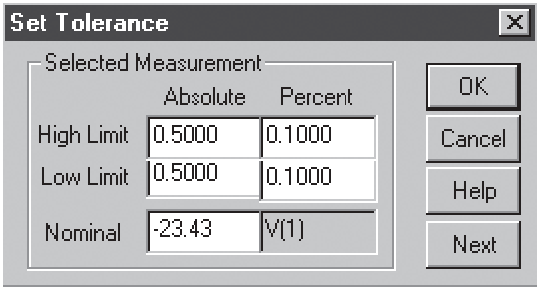
Next, we'll manually set the limits for the V(1) and V(8) measurements.

- **Click on the Measurements tab.**

- **Double-click on the "ramp start/ramp/TRAN/getCursorY0" leaf in order to expand it.**

- **Select vector V(1), then click on the Edit Tol button and manually set the tolerances as shown below.**

These are the limits for V(1).



Set Tolerance

Selected Measurement

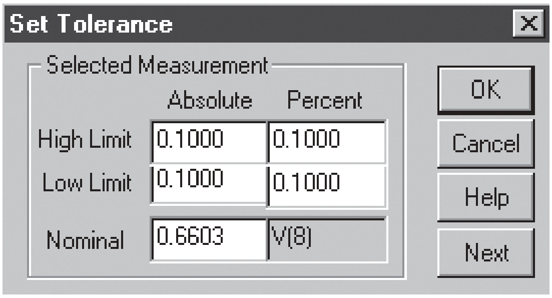
	Absolute	Percent
High Limit	0.5000	0.1000
Low Limit	0.5000	0.1000
Nominal	-23.43	V(1)

These values would be acceptable, based upon the interactive simulation data.

- **Click on the OK button to dismiss the Set Tolerance dialog.**

- **Select vector V(8), then click on the Edit Tol button and manually set the tolerances as shown below.**

These are the limits for V(8).



	Absolute	Percent
High Limit	0.1000	0.1000
Low Limit	0.1000	0.1000
Nominal	0.6603	V(8)

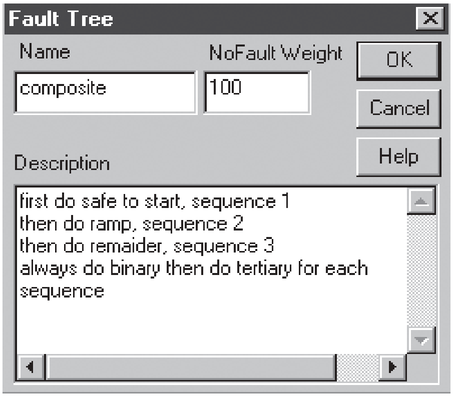
- **Click on the OK button to dismiss the Set Tolerance dialog.**

- **Select the Main tab and click on the Test Design button.**

- **Make a new fault tree by clicking on the New button in the Fault Tree Selection group.**

- **Enter the information that is shown below.**

These comments represent the steps we'll perform next.



Name: composite

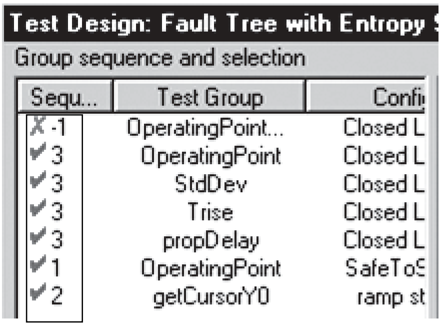
NoFault Weight: 100

Description:
first do safe to start, sequence 1
then do ramp, sequence 2
then do remaider, sequence 3
always do binary then do tertiary for each sequence

- **Click on the OK button to dismiss the dialog.**

- **Click on each sequence value and enter the values that are shown below.**

Tests can be grouped in different “sequences”.

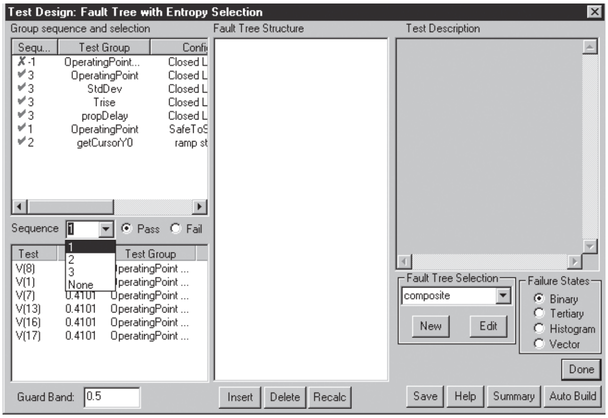


Test Design: Fault Tree with Entropy

Group sequence and selection

Sequ...	Test Group	Confi
✗ -1	OperatingPoint...	Closed L
✓ 3	OperatingPoint	Closed L
✓ 3	StdDev	Closed L
✓ 3	Trise	Closed L
✓ 3	propDelay	Closed L
✓ 1	OperatingPoint	SafeToS
✓ 2	getCursorY0	ramp st

- Select sequence #1 from the Sequence list.



Test Design: Fault Tree with Entropy Selection

Group sequence and selection

Sequ...	Test Group	Confi
✗ -1	OperatingPoint...	Closed L
✓ 3	OperatingPoint	Closed L
✓ 3	StdDev	Closed L
✓ 3	Trise	Closed L
✓ 3	propDelay	Closed L
✓ 1	OperatingPoint	SafeToS
✓ 2	getCursorY0	ramp st

Sequence: 1 ● Pass ○ Fail

Test		Test Group
V(8)		peratingPoint ...
V(1)		peratingPoint ...
V(7)	0.4101	OperatingPoint ...
V(13)	0.4101	OperatingPoint ...
V(16)	0.4101	OperatingPoint ...
V(17)	0.4101	OperatingPoint ...

Guard Band: 0.5

Fault Tree Structure

Test Description

Fault Tree Selection: composite

Failure States: ● Binary ○ Tertiary ○ Histogram ○ Vector

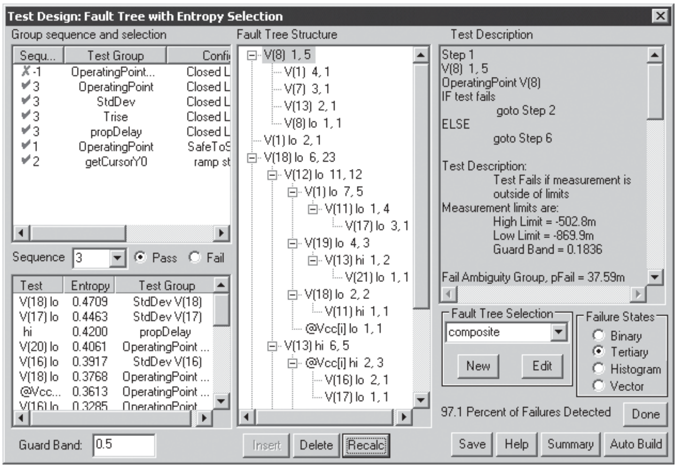
- Click on the Binary button.

- Click on the AutoBuild button to build the fault tree for sequence 1.

- Click on the Tertiary button, then click on the AutoBuild button to rebuild the fault tree using tertiary logic.

- Similarly, create fault trees for sequences 2 and 3.

Your Test Design window should appear similar to this.



Test Design: Fault Tree with Entropy Selection

Group sequence and selection

Sequ...	Test Group	Confi
✗ -1	OperatingPoint...	Closed L
✓ 3	OperatingPoint	Closed L
✓ 3	StdDev	Closed L
✓ 3	Trise	Closed L
✓ 3	propDelay	Closed L
✓ 1	OperatingPoint	SafeToS
✓ 2	getCursorY0	ramp st

Sequence: 3 ● Pass ○ Fail

Test	Entropy	Test Group
V(18) lo	0.4709	StdDev V(18)
V(17) lo	0.4463	StdDev V(17)
hi	0.4200	propDelay
V(20) lo	0.4061	OperatingPoint ...
V(16) lo	0.3917	StdDev V(16)
V(18) lo	0.3768	OperatingPoint ...
@Vcc...	0.3613	OperatingPoint ...
V(16) lo	0.3285	OperatingPoint

Guard Band: 0.5

Fault Tree Structure

```
⊟ V(8)  1, 5
     V(1)  4, 1
     V(7)  3, 1
     V(13)  2, 1
     V(8) lo  1, 1
  V(1) lo  2, 1
⊟ V(18) lo  6, 23
   ⊟ V(12) lo  11, 12
      ⊟ V(1) lo  7, 5
         ⊟ V(11) lo  1, 4
              V(17) lo  3, 1
      ⊟ V(19) lo  4, 3
         ⊟ V(13) hi  1, 2
              V(21) lo  1, 1
      ⊟ V(18) lo  2, 2
           V(11) hi  1, 1
        @Vcc[i] lo  1, 1
   ⊟ V(13) hi  6, 5
      ⊟ @Vcc[i] hi  2, 3
           V(16) lo  2, 1
           V(17) lo  1, 1
```

Test Description

```
Step 1
V(8)  1, 5
OperatingPoint V(8)
IF test fails
            goto Step 2
ELSE
           goto Step 6

Test Description:
           Test Fails if measurement is
           outside of limits
Measurement limits are:
           High Limit = -502.8m
           Low Limit = -869.9m
           Guard Band = 0.1836

Fail Ambiguity Group, pFail = 37.59m
```

Fault Tree Selection: composite

Failure States: ○ Binary ● Tertiary ○ Histogram ○ Vector

97.1 Percent of Failures Detected

You can compare your results with the sample2c.dwg file.

Optimizing the Test

You might not have probe access to some nodes, and other nodes might be sensitive to noise, so you may wish to delete them from consideration. Vectors can be removed from consideration via the Vectors tab.

Some nodes may be sensitive to noise, or may not be accessible for probing.

- **Eliminate undesirable test points, e.g., the summing junction V(11), by selecting them in the Vectors tab and clicking on the Switching Selected State button.**

- **Then rebuild the fault tree.**

Notice that up to a point, the detection capability doesn't change; however, fault isolation becomes more granular. That's the trade-off decision you need to make, taking into account the constraints that have been placed on your test design.

lo 0.3251 propDelay

Guard Band: 2

- **Try increasing the value of the guard band to 0.5 or 2.0.**

If the fault tree doesn't change, the tests with higher guard bands are better, provided the test decision point is centered in the guard band.

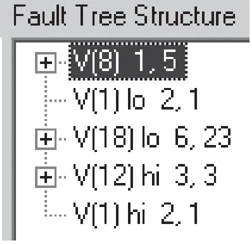
Entropy isn't compared across failure states. Sometimes, a higher entropy test is possible using binary states instead of tertiary states.

- **You may press the Binary button at any time in order to see if you should use the binary states.**

A procedure that works frequently is to exhaust the binary states using the Auto Build function, then finish with the tertiary states, again using the Auto Build function. This is the procedure that has been performed in our example. Be aware that vector and histogram states require much better fault modeling in order to produce robust test programs.

- **Click on the expanded branches of the fault tree to collapse it so that it shows only the root and its siblings.**

This is the collapsed fault tree.



Fault Tree Structure

```
⊞ V(8) 1, 5
   V(1) lo 2, 1
⊞ V(18) lo 6, 23
⊞ V(12) hi 3, 3
   V(1) hi 2, 1
```

> You now have the necessary tests to make a product acceptance test, without fault isolation. If you add to this test suite the functional performance tests that your product must pass, then you have created a test that detects all catastrophic faults and screens the product for its functional capability.

Summary

You have just completed a product acceptance test and a fault isolation test using the failure modes, which are recommended by the U.S. Navy for their Consolidated Automatic Support System (CASS) programs. Interestingly, you have identified several components that are not required for circuit functionality. If these procedures are used properly, they satisfy contract requirements for fault isolation.

There are many ways to use the Test Design features of the program. We've pointed out a few; but there are many others, including the ability to perform a manual fault tree design without the use of our entropy recommendations. The Options button in the Results dialog is just the beginning of a new way to develop an automatic test. It finds all possible equations that identify failures, and reports these equations in a file. If the cost of making all measurements is equal, then we can use all of the equations in order to make a fault isolation decision. This furnishes a very robust mechanism, avoiding the pitfalls associated with overestimating the accuracy of the tests. This approach redefines the idea of a "best" solution, and requires further research to prove its value.

Analog Simulation Examples

AC Analysis - A Butterworth Filter

In this example, we will introduce the PWL voltage source and demonstrate the effect of sampling theory, including how it applies to IsSpice4.

<u>**New Topics Covered**</u>

Sources

- PWL...Piecewise Linear Voltage Controlled Source

Transient Analysis Control

- TSTEP...Time Step Control

Output Control

- Printout and Plot Resolution
- Magnitude/Phase Plot (.PRINT AC)

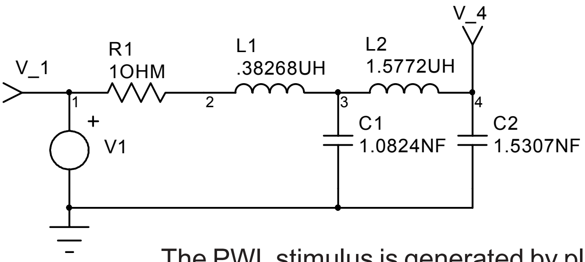
Analysis

- .PZ...Pole Zero Analysis

Specification:

Given the circuit below, we'll determine the magnitude and phase response vs. frequency (Bode Plot) and the pulse response. Using the PWL statement, the input pulse should have the following time voltage description:

Open the schematic at :\spice8\Circuits\Examples\AC Analysis.DWG



Time (s)	Voltage (v)
0	0
1u	0
2u	1
20u	1
20.1u	0

The .PRINT Statements are generated for you by the test point symbols. You DO NOT have to type them in!!

The PWL stimulus is generated by placing a voltage source on the schematic, double-clicking on it to open the properties dialog, double-clicking on Tran Generators field and then entering the PWL data points in the PWL tab.

The .TRAN and .AC statements are created using the Actions Simulation Setup Edit... function.

Be careful to enter the PWL points correctly. Be sure not to enter the letter “O” in place of the number zero.

```
AC Analysis - A Butterworth Filter
.TRAN 5US 50US
.AC DEC 20 100HZ 100MEGHZ
.PZ 2 0 4 0 VOL PZ             ; The .PZ statement is entered in the
R1 1 2 1OHM                    ; IsSpice4 Simulation Setup dialog
L1 2 3 .38268UH                  ; in the user statement field.
L2 3 4 1.5772UH
C1 3 0 1.0824NF
C2 4 0 1.5307NF
V1 1 0 PWL 0S 0V 1US 0V 2US 1V 20US 1V 20.1US 0V AC 1
.END
```

Note: The netlist generated by the schematic will look different.

- **After entering the schematic, simulate the circuit. Be careful to enter the piecewise linear voltage source properly.**

Refer to the IsSpice4 User’s Guide to clarify any syntax questions. If any errors appear, check the error file by clicking the ERR button in IsEd.

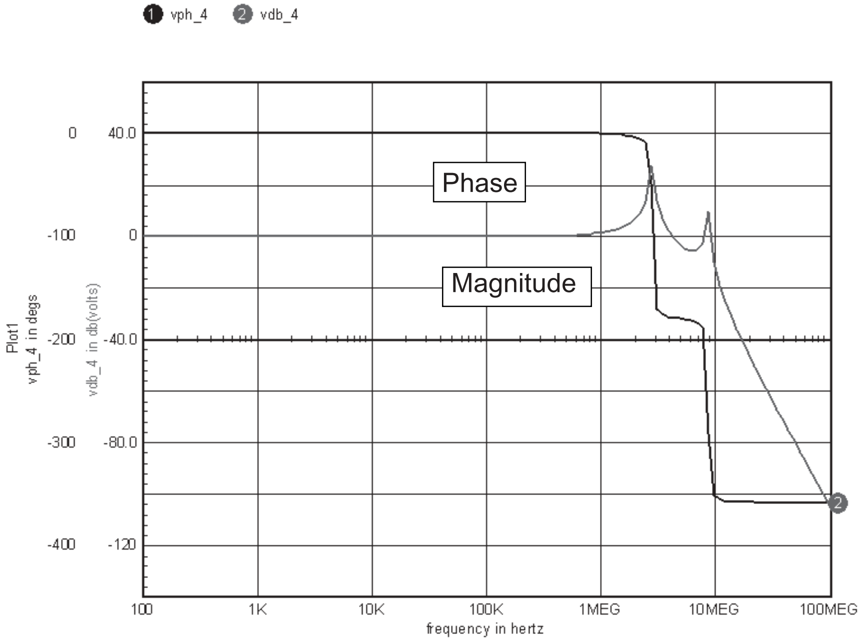
Check the .ERR file for a listing of the IsSpice4 *error and warning messages if your simulation does not run.*

The voltage source, V1, has a piecewise linear transient signal generator, as well as an AC source magnitude value. This is perfectly all right. Sources can be assigned values for any or all of the five main types of analyses: AC, noise, distortion, DC or transient, on the same line. The values will each be used with their respective analyses.

Bode Plot

After running a successful simulation, look at the plot of VDB(4) and VP(4). You can look at various waveforms and variations (magnitude, real, imaginary, phase, etc.) of the saved waveforms using the IntuScope Add Waveform dialog. The schematic test points automatically cause the phase and magnitude in dB to be displayed in IsSpice4. The number of points we decided to print per decade (20) is determined by the .AC statement. This point interval determines how the output data will appear.

The magnitude/phase response for this example is shown on the next page. Your X/Y axis labels may be different.



Interpolated data is used for the waveforms when the source is "Active Project Output File." Non-interpolated "raw" data is used for the waveforms when the source is "Active IsSpice Simulation."

Non-interpolated data is normally more accurate and can be controlled by the TMAX and RELTOL parameters.

Pulse Response

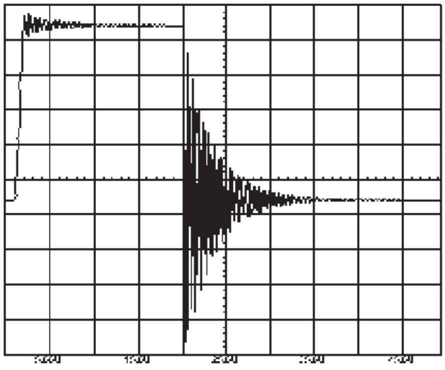
Proper specification of control statements is critical to the transient analysis. The minimum entries required for the transient control statement are TSTOP and TSTEP. TSTOP is the total analysis time. TSTEP is the sampling interval. The Nyquist sampling theory will dictate the appropriate value of TSTEP. TSTEP must be at least twice the frequency of the quickest event in a simulation in order for the event to be captured. However, in certain circumstances, trade-offs must be made between acceptable accuracy and simulation time.

Examine the transient waveforms. Notice how the steep fall time causes ringing at the output. Using inadequate sampling can greatly effect the results you will receive from IsSpice4. The graphs shown next were created by changing the TSTEP parameter in the .TRAN statement, thus altering the interpolated data recorded by the simulation. It clearly shows the effect of variations on the data printout interval, TSTEP. Too few points will cause data to be aliased or distorted. Too many data points will increase the total simulation time and may provide data beyond the scope of interest. Note that no matter how many data points were taken, the final value was always the same.

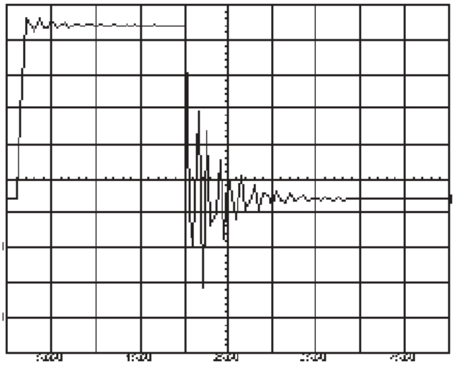
Additional Notes On Accuracy

There are other parameters besides TSTEP that can affect the accuracy of the output from a simulation. Some of these are the TMAX parameter in the .TRAN control statement and the RELTOL parameter in the .OPTIONS control statement.

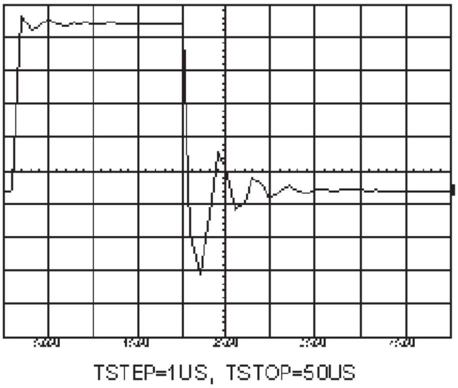
The TMAX parameter in the .TRAN control statement controls the maximum internal timestep of the simulator. The IsSpice4 simulator does not necessarily make calculations at every data point requested by TSTEP. The internal timestep of the simulator is variable, and changes depending on the conditions present



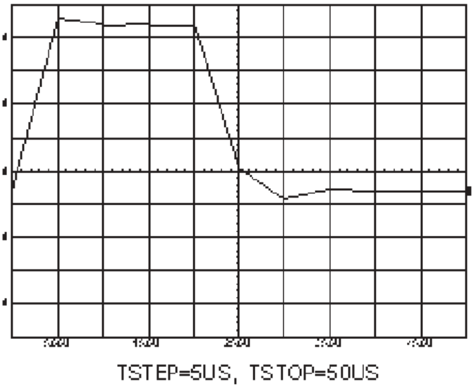
TSTEP=.1US, TSTOP=50US



TSTEP=.5US, TSTOP=50US



TSTEP=1US, TSTOP=50US



TSTEP=5US, TSTOP=50US

in the circuit. All output data is interpolated from internally calculated points. The TMAX parameter is used to control the internal time step by limiting it to a specified maximum value. This will allow the variable time points being used for interpolation to be closer to the actual requested constant data point interval. (TMAX should be set to one-half or one-fourth of **TSTEP** if interpolation problems are suspected.)

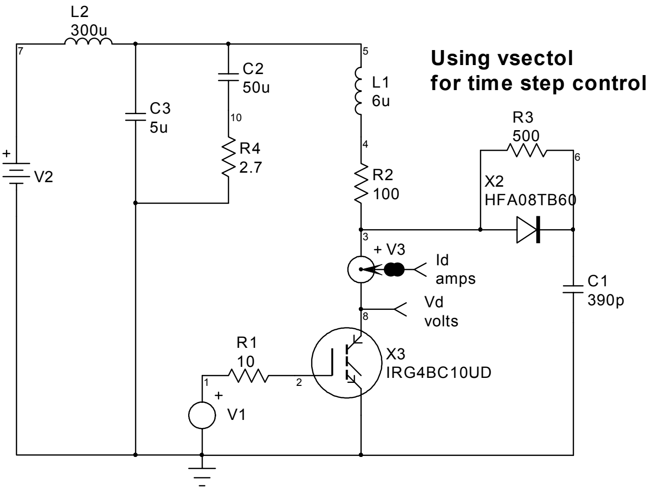
Use the Convergence Wizard to resolve convergence errors.

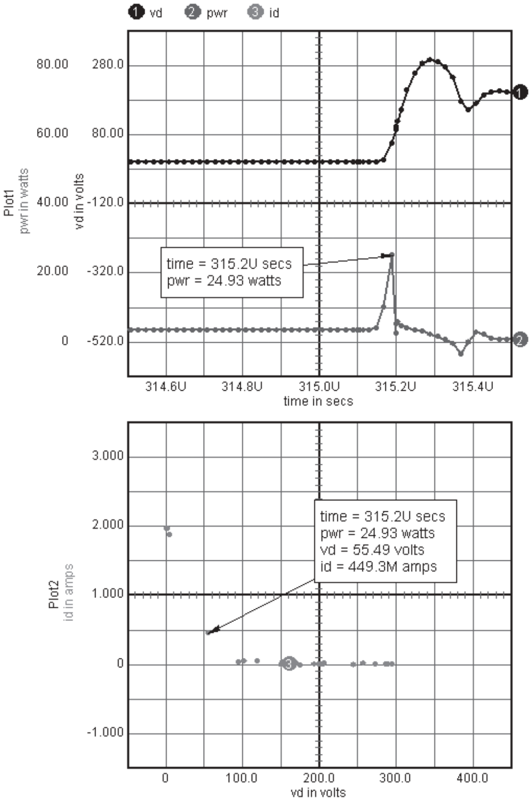
The RELTOL parameter in the .OPTIONS statement controls the overall accuracy of the simulator. The default value is set to 0.001. For most applications, RELTOL can be set to .01 (or 0.1% accuracy) to speed up the simulation without significantly affecting accuracy.

VSECTOL

A new VSECTOL option reduces the usual SPICE time step if the product of the absolute value of the error in the predicted voltage (prediction – solution), and the time step, exceeds the VSECTOL specification. As in most IsSpice4 options, its default is zero, which turns the option off. In testing the VSECTOL option, we ran some cases setting RELTOL=0.5, and at the same time set VSECTOL to a reasonable value for the circuit. The figure referenced below shows this new time step control option for the simulation, based on node voltage accuracy.

For high switching circuits like **:\spice8\Circuits\Snubber\snubber.dwg**, the standard SPICE time-step algorithms wont ensure that you acquire enough data points to accurately view the design’s switching characteristics. Normally, you must specify a very small TMAX (maximum time between data points) to get adequate accuracy. However, VSECTOL is used to maintain volt-second accuracy. It ensures adequate detail in the switching transitions while avoiding unnecessary data points that otherwise greatly increase total simulation time. Remember to always set Bypass=OFF when using VSECTOL. Bypass normally controls whether the device operating point is calculated for each time step. When Bypass is ON, these load operations are skipped, if errors are low enough. However, Bypass=ON does create problems for the VSECTOL algorithm.


Using vsectol
for time step control
L2
300u
7
C3
5u
C2
50u
10
R4
2.7
+
V2
5
L1
6u
4
R2
100
R3
500
6
X2
HFA08TB60
3
+ V3
Id
amps
Vd
volts
8
C1
390p
R1
10
1
2
+
V1
X3
IRG4BC10UD


1 vd
2 pwr
3 id
Plot1
pwr in watts
vd in volts
80.00
60.00
40.00
20.00
0
280.0
80.00
-120.0
-320.0
-520.0
time = 315.2U secs
pwr = 24.93 watts
314.6U
314.8U
315.0U
315.2U
315.4U
time in secs
Plot2
id in amps
3.000
2.000
1.000
0
-1.000
time = 315.2U secs
pwr = 24.93 watts
vd = 55.49 volts
id = 449.3M amps
0
100.0
200.0
300.0
400.0
vd in volts

Results using VSECTOL to control the time step produces higher accuracy during the turn-off transition and uses less computational resources when there is no switching activity.

Transient timepoints = 26015
Accepted timepoints = 14217
Rejected timepoints = 11798
Total Analysis Time = 35.15

Pole-Zero Analysis

Use the User Statements field in the IsSpice4 Simulation Setup dialog to enter the .PZ statement.

IsSpice4 has the capability to calculate the poles and zeros of a transfer function. The syntax for the Pole-Zero analysis is listed in the IsSpice4 User's Guide. In this example, the poles and zeros are requested for the transfer function defined by the input nodes, 2 & 0, and output nodes, 4 & 0. The transfer function is defined by the VOL entry to be: (output voltage)/(input voltage). We could have instead used the CUR entry to define the transfer function as (output voltage)/(input current). The .PZ entry causes the calculated poles and zeros to be placed in the output file. An example is shown below.

```
-------------- Pole - Zero Analysis -------------

Transfer function of (V(4) - V(0)) / (V(2) - V(0)) in Volts / Volt

Poles           Real                    Imag                 Number
               ------                  ------                ------
             -9.87068e+005          5.557209e+007              1
             -9.87068e+005          -5.55721e+007              1
             -3.19507e+005          1.798896e+007              1
             -3.19507e+005          -1.79890e+007              1
```

- **Try expanding the frequency range for the .AC analysis by changing the .AC control statement to the following:**

  .AC DEC 20 100KHZ 1GHZ

- **Resimulate the circuit with the new .AC statement and examine the output.**

Transient Analysis - A Rectifier Circuit

Since not all circuits are made with passive components, it is time to advance and see how to use active devices. This example will demonstrate how to call and use IsSpice4's built-in diode model.

New Topics Covered

Sources

- V...Independent Voltage Source
  SIN...Sinusoidal Transient Statement

Semiconductors

- D...Diode call

Miscellaneous

- .MODEL statement

Specification:

The circuit we will construct and simulate is shown on the next page. The input stimulus consists of a sinusoidal signal with 0 volt offset and a peak value of 1 volt at 1kHz. The SIN stimulus is generated by placing a voltage source on the schematic, double-clicking on it to open the properties dialog, double-clicking on Tran Generators field and then entering the SIN data into the SIN tab.

The DC voltage source will have a value of .5 volts. Data will be collected every 10µs for 2ms at nodes 1 and 2 using a transient analysis and two test points. We will use the 1N4148 diode. This diode can be found using the X Parts Browser.. function under the Parts menu.

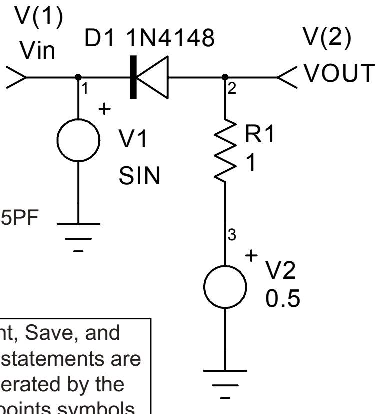
- **Enter the design and run the simulation.**

Note: It is not necessary to type the .MODEL statements. SpiceNet will retrieve the models from the ICAPS SPICE model libraries in the PR directory automatically.

Open up the drawing
:\spice8\Circuits\Examples\TRANAnalysis.DWG

The IsSpice4 *model and .PRINT statements are only shown for clarity. They DO NOT need to be typed in!*

```
Transient - A Rectifier Circuit
*#save V(1) @V1[i] @V1[p] V(2) V(3) @R1[i] @R1[p] @V2[i]
*#save @V2[p] @D1[id] @D1[p] @D3[id] @D3[p]
*#alias vin  v(1)
*#view  tran vin
*#alias vout  v(2)
*#view  tran vout
.TRAN 10u 2m
*#save all allcur allpow
.PRINT  TRAN Vin
.PRINT  TRAN Vout
D1 2 1 DN4001
.MODEL DN4001 D BV=75V CJO=55PF
IS=1.38E-09 M=.38 N=1.7
+ RS=.04 TT=5U VJ=.34V
V1 1 0 SIN 0 1 1k
R1 2 3 1
V2 3 0 DC=0.5
.END
```



Print, Save, and Alias statements are generated by the test points symbols

In the above netlist, you will notice the .MODEL statements. All semiconductor devices used in an IsSpice4 simulation must have a .MODEL statement associated with them. This statement is used to describe the semiconductor device by passing a set of parameters that fill predefined equations. These equations are then used to simulate the device. Semiconductor devices require two statements in a netlist. One line, the diode call line, describes the connections (i.e. D1 2 1 DN4148). The second is the associated .MODEL statement, which is referred to by the diode (i.e. .MODEL DN4148 D (CJO=...). The model name in the .MODEL line must match the one on the diode call line (DN4148). The schematic handles this syntax for you.

As explained in the previous example, the .TRAN parameters are extremely important when performing a transient analysis. The output is a direct result of the TSTEP entry in the .TRAN statement, and should be set to adequately sample the waveform. Unfortunately, the highest output frequency is not always known. Hence, the .TRAN statement should be set to examine the expected output from a circuit. Unresolved artifacts can always be examined with another simulation at an reduced sampling interval.

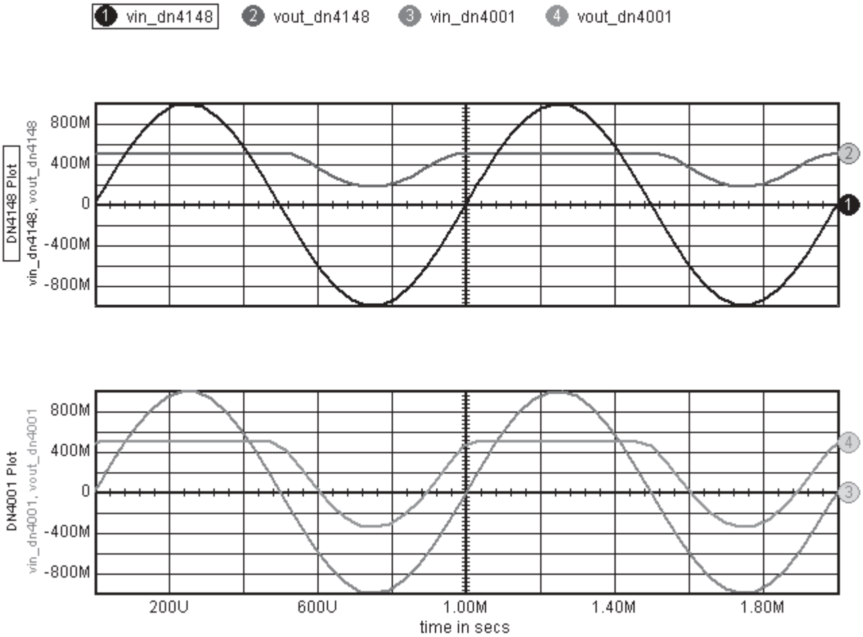
In this circuit, we used a 1kHz SIN wave. One period occurs every 1ms. Since the main interest of the circuit focuses on the response to the 1kHz sin wave, we must choose a sampling interval that will generate acceptable output. By adhering strictly to the Nyquist theory, the output should be sampled at least every 0.5ms. The choice of 10μs was used to obtain detailed output without causing excessive simulation time.

- **To continue the example, replace the DN4148 diode with a DN4001 diode.**

Note: After placing more than one model for the same kind of part, you can switch between the different models by using the .Model field in the part's properties dialog.

- **Rerun the simulation.**

The results of the simulation for the circuit containing the DN4148 and DN4001 diode are shown below.



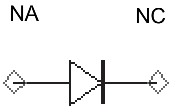
Proper Use of Semiconductor Devices

When using semiconductor devices, care should be taken to ensure that the proper connections are made. An incorrectly connected device may satisfy simulator connectivity requirements, but may cause your circuit to run improperly or cause IsSpice4 to abort the analysis.

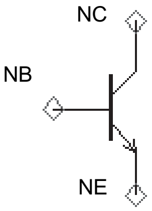
Normally, models are retrieved automatically from libraries.

Listed next are five types of semiconductor devices that are available in IsSpice4, their connections, and an abbreviated example of each device's model statement. The model parameters can be edited directly in the schematic using the part properties dialog. Any changes will be local to the schematic. The semiconductor devices are called using the D, Q, J, M, or Z keyletters, and must be properly defined by an appropriate .MODEL TYPE parameter. For example, Q should list an NPN or PNP type.

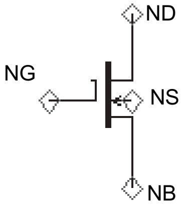
```
D1  NA  NC  DN4148
.MODEL DN4148  D
```



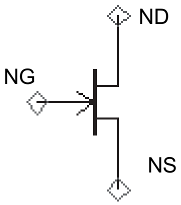
```
Q1 NC NB NE QN2222
.MODEL QN2222 NPN
```



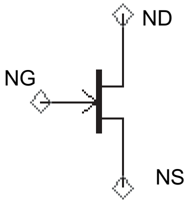
```
M1 ND NG NS NB  N1
.MODEL N1 NMOS
```



```
J1 ND NG NS J2N
.MODEL J2N  PJF
```



```
Z1 ND NG NS J2N
.MODEL GAS NMF
```



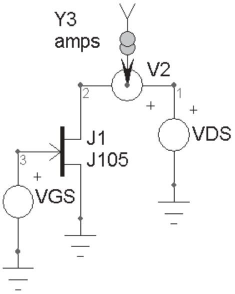
DC Sweep - Device Characteristics

In this example, we will explore the use of the IsSpice4 JFET model and how to obtain its forward DC characteristics.

Specifications:
A DC analysis will be performed and the DC sources VDS (Drain-Source voltage), and VGS (Gate-Source voltage), will be swept to obtain the drain current characteristics. VDS will be swept from 0 to 5V in .2V steps, and VGS will be swept from -5v to -7V in -.5V steps. The voltage source, V2, will be used to measure the drain current. Output will be obtained for the Drain-Source voltage, VDS, and the Drain-Source current, I(V2).

Note: You must make the reference designations in the voltage source properties dialogs match the ones shown in the schematic below if the .DC statement is to work properly.

Open up the drawings at
:\spice8\Circuits\Examples\DCSweep.DWG



<u>**New Topics Covered**</u>

Sources
- Current meter

Semiconductors
- JFET

Analysis
- DC Sweep

Output Control
- .PRINT DC

The IsSpice4 *model and .PRINT statements are only shown for clarity. They DO NOT need to be typed in!*

```
DC Sweep - Device Characteristics
.*#save all allcur allpow
.PRINT  DC IY1
VGS 3 0
J2 1 3 0 J105
.MODEL J105 NJF AF=1 BETA=5M CGD=1.40E-10 CGS=8.00E-11
+ FC=.5 IS=9.48E-14 KF=3.886E-16 LAMBDA=.035 RD=4.20E-01
+ RS=3.78E-01 VTO=-7.25 PB=1.00
V2 2 1
VDS 2 0
.END
```

DC Analysis

The .DC control statement is used to specify a DC analysis. In a DC analysis, any voltage or current source may be swept. Node voltages or branch currents requested in the .PRINT DC statements will be recorded. The DC analysis statement **.DC VDS 0V 5V .2V VGS -5V -7V -.5V** will sweep the voltage source VDS from 0 to 5V, in .2V steps. For each step of VDS, VGS will be swept from -5V to -7V in -0.5V increments. Data will be saved for each combination of VGS and VDS. In this case, the current through the voltage source, V2, (.PRINT DC I(V2)), will be measured. The value of VDS is obtained automatically from the DC analysis, because it is the first swept variable.

You are likely to notice that the voltage sources do not need any values. In a DC analysis, the source values are driven by the .DC control statement. The .DC statement will override any DC or transient values entered into the properties dialog during the .DC analysis, but will allow the values to be used for the AC and transient analyses.

JFET Semiconductor

Like the diode, there are two lines that identify a JFET. The call line, (J1 1 3 0 J105), and the .MODEL statement, (.MODEL J105 NJF (VT=...). The .MODEL statement DOES NOT have to be typed into the schematic. It will be added to the netlist automatically.

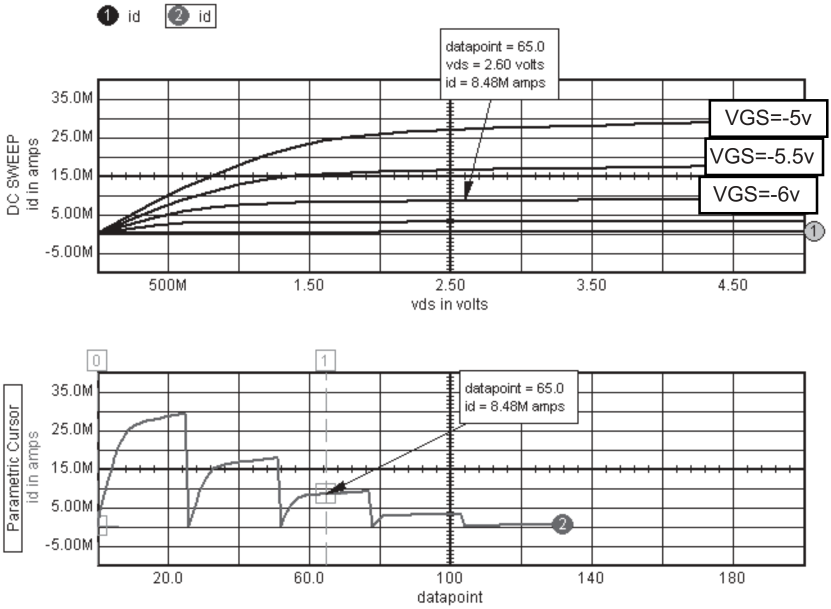
Current Measurement

Another new function performed by this example is the measurement of current. In IsSpice4, current can be measured through any element. In this example, the current meter, V2, does not contribute stimulus to the circuit since its value field is not stated. It only exists to measure current flow. A blank value field is interpreted as zero volts. For other devices, current is specified by inserting a current test point (Parts Menu, 3Y) in the symbol. You can also use the .PRINT DC statement by placing the @ sign before the model's Reference Designation, followed by the desired current (between square brackets).

. For example;   .PRINT DC @J1[ID]

Plot i_v2, then run parametric cursors by invoking IntuScope; Calculator menu; Measurements; parametric cursor. Move cursor1 along the created curve at the bottom to obtain all the values of the DC sweep curve at the top.

The output for this example is shown below.



Transmission Lines - Buffering and Impedance Matching

In this example, we will demonstrate three new elements, the voltage controlled voltage source (VCVS), and two transmission lines (T-Line - both lossless and lossy). We will use the VCVS to simulate an ideal op-amp and the T-line to investigate impedance matching.

<u>New Topics Covered</u>

Sources

- E...Voltage Controlled Voltage Source

Passive Elements

- T...Lossless Transmission Line
- O...Lossy Transmission Line

Specification:

For the circuit shown below, the input voltage source will consist of a pulse train with an initial value of .5V, a final value of 3.5V, no delay, rise and fall times of 3ns, a pulse width of 22ns, and a period of 50ns. The VCVS will have a gain of 10,000. The lossless T-line will have a time delay (TD) of 2ns, a characteristic impedance of 75Ω and a length of 1 meter. The lossy T-line will have a time delay of 2ns, an impedance (Zo) of 75Ω and a series resistance of 4Ω/m. Transient analysis data should be collected every 1ns for 100ns, starting at time 0. We will investigate the output of the op-amp and the T-line. We will also investigate the effects of losses and length for the best matching circuit.

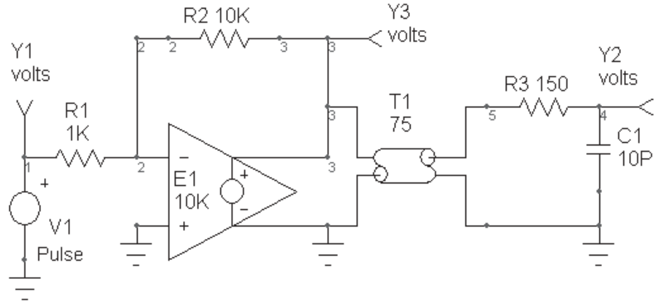
Open up the schematic at :\spice8\Circuits\Examples\TLine.DWG

"ZO=75 TD=2N" must be entered into the lossless transmission line's Properties dialog.

```
T-Lines - Buffering and Impedance Matching
*#SAVE ALL ALLCUR ALLPOW
.TRAN 1N 100N 0 1N
.PRINT TRAN  V(1)  V(3)  V(4)
R1 1 2 1K
E1 3 0 0 2 10K
R2 2 3 10K
T1 3 0 5 0   ZO=75 TD=2N  ; Symbol Ideal T line
R3 5 4 150
C1 4 0 10P
V1 1 0 PULSE .5 3.5 0 3N 3N 22N 50N
.END
```

Buffering

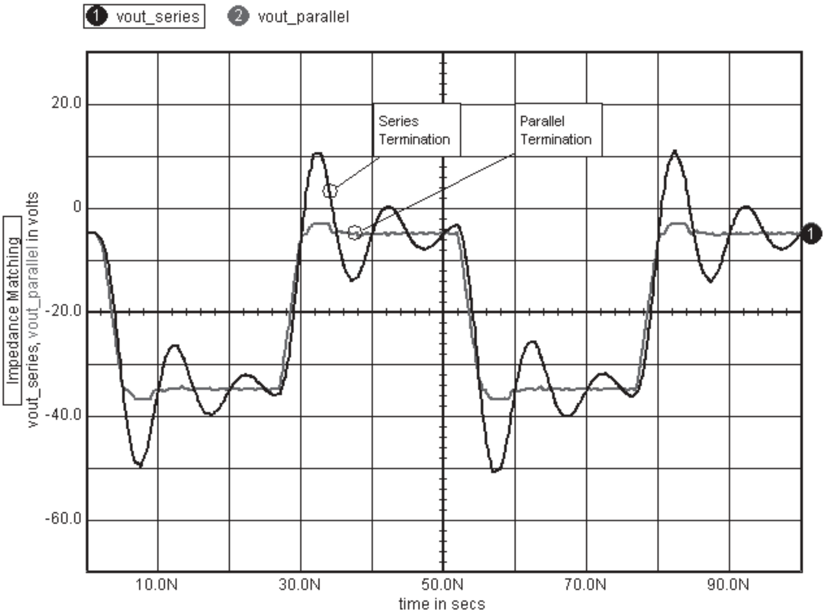
The VCVS in this circuit is modeling an ideal op-amp.



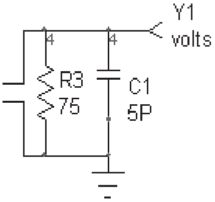
The resistors, R1 and R2, create a simple inverting amplifier configuration with a gain of 10. This inverting amplifier essentially buffers the input voltage source to the transmission line.

Impedance Matching

The 150Ω resistor and the 10pf capacitor form a series impedance match to the transmission line. As the output shows, this termination scheme is quite poor.



- **To improve the matching, replace the 150Ω resistor and 10pf capacitor with the parallel termination shown to the left. Simulate the circuit again. The new statements are shown below for reference.**



R3 4 0 75
C1 4 0 5P

Now that the circuit has better matching, let's examine the impact of a longer transmission line and the effects of a less-than ideal T-line.

- **To do this, replace the current Lossless T-line symbol with the RLCG Lossy Tline symbol.**

The symbol RLCG Lossy T Line is found under the Parts menu, Passive submenu.

- **Enter the model name "line" in the Lossy Tline's .Model pop-up field in the properties dialog.**

The model name links the lossy transmission line element to a set of model parameters. To calculate the necessary parameters, we will use the following conversions.

$$TD = \frac{LEN}{V_P} \Rightarrow V_P = \frac{LEN}{TD}$$

$$Z_O = \sqrt{\frac{L}{C}} \Rightarrow L = Z_O{}^2 C$$

$$V_P = \frac{1}{\sqrt{LC}} \Rightarrow C = \frac{1}{LV_P{}^2}$$

In this example, LEN=1 meter, TD=2ns, Zo=75Ω, Vp is phase velocity (m/s), L is inductance (H/m) and C is capacitance (f/m). Evaluating the equations yields the parameters for the following .MODEL statement.

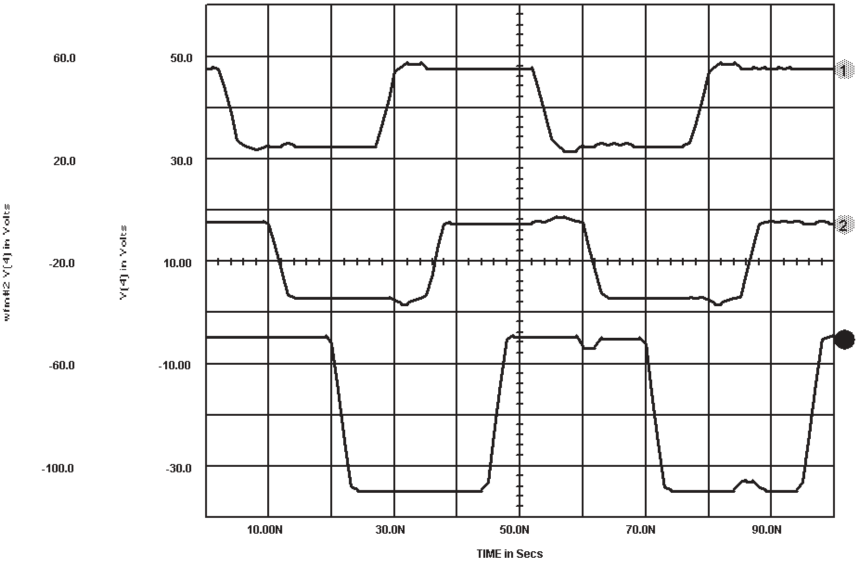
```
.MODEL LINE LTRA(R=0 L=150n G=0 C=26p LEN=1)
```

- **Enter the LTRA parameters values in the Part Properties dialog. Simulate the parallel and series termination circuits with this new transmission line. The results should be comparable.**

- **With the transmission line in this form, it is quite easy to investigate the effects of lengthening the line. To do this, simply change the value of LEN in the Part properties dialog.**

- **Simulate the series terminated circuit with different lengths. The results for simulations with 1, 5 and 10 meter segments are shown on the following page.**

Finally, we can determine the effects of losses in the line. This



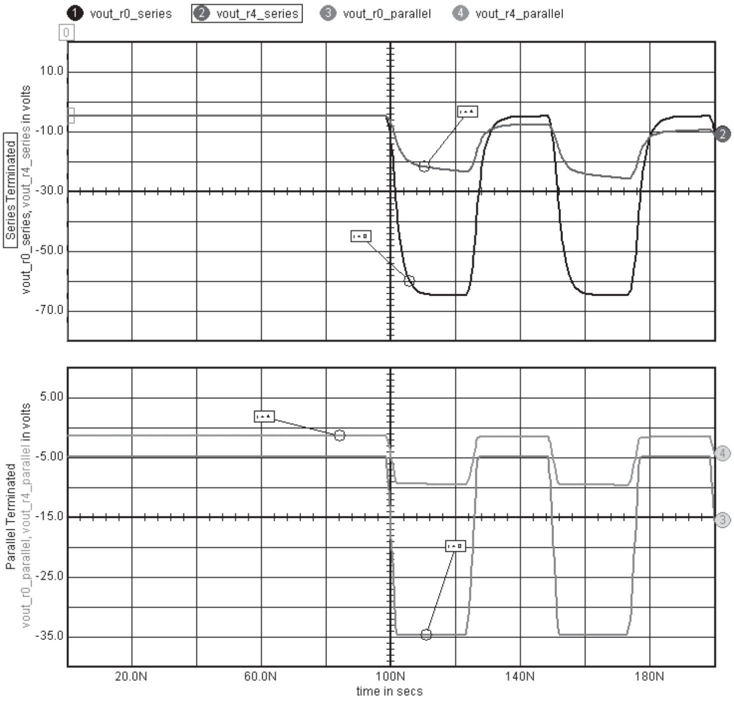
is done by setting R to a nonzero value. Unfortunately, it is not as easy as plugging in a value. If R is included, the characteristic impedance (Zo), and phase velocity (Vp) will change. This will change the values of L and C. For this example, however, we will simply add a unit resistance of 4Ω/m and simulate the circuit for a 50m line.

- **Replace the values for the lossy line parameters with the following parameter values.**

R=4 G=0 L=150n C=26p LEN=50

By increasing the length of the line, we have also increased the time delay. Because of this, the TSTOP parameter in the .TRAN statement must be increased. Replace the .TRAN line with .TRAN 1N 200N 0 1N.

- **Simulate the series-terminated circuit with the new lossy line. The results are shown next. Note the attenuation in the line caused by R.**



Noise, Distortion, and Fourier Analyses - Bipolar Amplifier

So far, we have investigated the AC, DC, and Transient analyses. All of these fundamental analysis types are important when designing and troubleshooting a circuit. However, there are numerous circuit specification requirements that are not tested by these three analysis types. This example uses a simple bipolar amplifier to perform: distortion, Fourier, noise, and DC transfer function analyses.

New Topics Covered

Analysis Control

- .FOUR...Fourier Analysis
- .DISTO...Distortion Analysis
- .NOISE...AC Small Signal Noise Analysis
- .TF...Transfer Function
- .OPTION ACCT

Open up the drawing at :\spice8\Circuits\Examples\NoiseDistoFour.DWG

ICL Control Block

- show...Device Operating Point Information
- showmod...Model Operating Point Information

Specification:

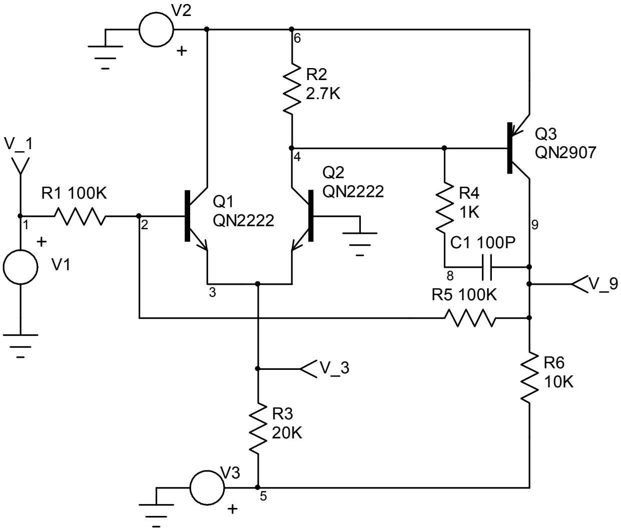
We wish to determine the harmonic content, intermodulation distortion and the equivalent input and output noise of the following amplifier circuit. The input stimulus will be a sinusoidal waveform with 0V offset, a magnitude of 1V and a frequency of 100kHz. The AC input stimulus will have a magnitude of 1 and a phase of 0. Distortion sources will have a magnitude of 1 for the primary frequency, and a magnitude of .1 for the secondary frequency. Transient data should be collected every 100ns for 20µs, starting at 20µs. AC data should be collected at 20 points per decade from 1Hz to 100MEGHz. Noise data should be collected at 10 points per decade from 1Hz to 100MEGHz. Distortion output should be collected from 10kHz to 1MEGHz at 5 points per decade. Output should be collected for node voltage 9. Transient data should be collected for node 1. The circuit accounting information, operating point information, and model parameter information should all be collected and displayed in the output file.

*All the **ICL** (control block in bold below) statements are entered in the User Statements field of the IsSpice4 Simulation Setup dialog.*

```
Advanced Analysis Types
*#SAVE ALL ALLCUR ALLPOW
.CONTROL
OP
SHOW Q : VBE VBC ICC IB IE GM GO FT TEMP
SHOWMOD ALL
.ENDC
.MODEL QN2222 NPN (IS=15.2F NF=1 BF=105
+ VAF=98.5 IKF=.5 ISE=8.2P NE=2 BR=4
+ NR=1 VAR=20 IKR=.225 RE=.373 RB=1.49
+ RC=.149 XTB=1.5 CJE=35.5P CJC=12.2P
+ TF=500P TR=85N)
.MODEL QN2907 PNP (IS=1.1E-12 BF=200 BR=6
+ NF=1.21 RC=.6 IKF=100E-3 VAF=50 XTB=1.5
+ NE=1.9 ISE=1.3E-11 CJE=23PF VJE=.85 MJE=1.25
+ CJC=19PF VJC=.5 MJC=.2 TF=5E-10 TR=34E-9)
.OPTION  ACCT
.TRAN 100N 40U 20U
.AC DEC 20 1HZ 100MEGHZ
.FOUR 100KHZ V(9)
.DISTO DEC 5 10K 1MEG .49
.PRINT DISTO V(9) VP(9)
.NOISE V(9) V1 DEC 10 1 100MEG
.PRINT NOISE INOISE ONOISE
.TF V(9) V1
*#ALIAS  VIN V(1)
*#ALIAS  VOUT V(9)
.PRINT AC  V(9)  VP(9)
.PRINT TRAN  V(1)  V(9)
R1 1 2 100K
Q1 6 2 3 QN2222
Q2 4 0 3 QN2222
R2 4 6 2.7K
V2 6 0 10
R3 3 5 20K
V3 5 0 -10
R4 4 8 1K
Q3 9 4 6 QN2907
C1 8 9 100P
R5 9 2 100K
R6 9 5 10K
V1 1 0 Sin 0 1 100K AC 1 Distof1 1 Distof2 .1
.END
```

AC 1, Distof1 1 and Distof2 .1 are entered into the V1 Properties dialog.

Sin 0 1 100K is entered using the Tran generators field in the properties dialog.



The QN2222 and QN2907 symbols are listed in the Part Browser dialog under BJT Part Types, and General purpose Sub Types.

- **After simulating the circuit, scan the output file and look at the results produced by each different analysis.**

The analyses performed by this IsSpice4 simulation generate a tremendous amount of output data. The parameter options used to generate the output may appear a little confusing. We will go over each analysis.

```
.TRAN 100N 40U 20U
.FOUR 100KHZ V(9)
```

Fourier: This analysis is specified by the Fourier dialog in the IsSpice4 Simulation Setup dialog. It defines the fundamental frequency and the output variables. Data will be taken from TSTOP-period to TSTOP, where period is one period of the fundamental frequency specified on the .FOUR line as a frequency. The first nine harmonic components and the DC component will be calculated and placed in the output file for every output variable specified on the .FOUR line. In this example, a Fourier analysis was performed on data collected from 40µs-10µs (30µs) to 40µs for node 9. The output generated for the Fourier analysis is shown next. The Fourier analysis is performed in conjunction with the transient analysis, so a .TRAN statement with a valid TSTOP interval must be included.

Fourier output data is placed in the IsSpice4 *Output file.*

```
Fourier analysis for v(9):
  No. Harmonics: 10, THD: 0.602413 %, Gridsize: 200, Interpolation Degree: 1

Fourier analysis for v(9):
  No. Harmonics: 10, THD: 0.604437 %, Gridsize: 200, Interpolation Degree: 1

Harmonic   Frequency    Magnitude     Phase      Norm. Mag    Norm. Phase
--------    ---------    ---------     -----      ---------     -----------
 0          0            0.277301          0            0               0
 1          100000      1.0159         173.164       1               0
 2            200000     0.00591471   -52.971     0.00582215    -226.13
 3            300000     0.00160939   88.8066    0.0015842     -84.357
 4            400000     0.000350401   -146.93    0.000344918    -320.1
 5            500000     8.12824e-005   -60.837  8.00104e-005    -234
 6             600000     1.72764e-005    128.956  1.70061e-005  -44.207
 7             700000     3.24918e-005    134.603     3.19833e-005  -38.561
 8             800000     1.91887e-005   128.858     1.88884e-005  -44.305
 9             900000     1.54897e-005   117.339     1.52473e-005 -55.825
```

This Disto analysis is specified by the Distortion dialog in the IsSpice4 Simulation Setup dialog.

```
 .DISTO DEC 5 10K 1MEG .49
 .PRINT DISTO V(9) VP(9)
V1 1 0 AC 1 DISTOF1 1 DISTOF2 .1
```

Distortion: The distortion analysis is a small signal analysis that will produce the second and third harmonic distortion or the intermodulation product frequencies and the second intermodulation component. The three statements required to perform the distortion analysis are shown. The .DISTO line tells IsSpice4 to perform a distortion analysis from 10kHz to 1MEGHz at 5 points per decade. The final parameter on the line is the distortion parameter “f2overf1.” This sets the second frequency component to “f2overf1∗fstart,” in this case, 49kHz. When f2overf1 is present, the intermodulation components are calculated. When this parameter is absent, only the 2nd and 3rd harmonic contents are calculated. The second line tells IsSpice4 to output the voltage, in magnitude and phase, at node 9. The final line defines the input magnitudes for both the primary frequency (distof1), and the secondary frequency (distof2), as 1 and .1, respectively. Output for the distortion analysis is tabulated in the output file for the range and points specified in the .DISTO line.

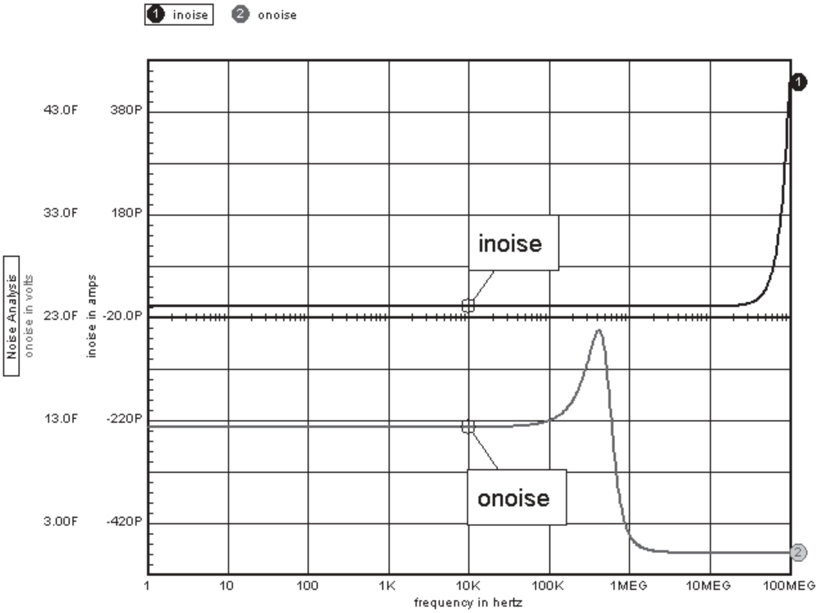
This .Noise analysis is specified by the Noise dialog in the IsSpice4 Simulation Setup dialog.

```
.NOISE V(9) V1 DEC 10 1 100MEG
.PRINT NOISE INOISE ONOISE
```

Noise: The noise analysis is a small signal analysis. It calculates the total output noise and the equivalent input noise, as well as the noise contributions of each component or model parameter. The noise analysis requires the presence of two lines in the netlist, as is shown in the text box. The first is the .NOISE control statement. This statement specifies the frequency range and how the data should be collected. This is very similar to the .AC and .DISTO statements. In this example, the .NOISE control statement will cause output to be calculated for all noise contributing sources in the circuit (resistive and active elements) at 10 points per decade from 1Hz to 100MEGHz. To generate data, SpiceNet issues the .PRINT NOISE statement when a .NOISE analysis is specified. It does not have to be typed in. The .PRINT NOISE statement works like any other .PRINT statement except the only options that are allowed are INOISE (equivalent input noise) and ONOISE (total output noise).

Noise output can be converted to dB with INTUSCOPE.

The output for the noise analysis is shown below.



This .TF analysis is specified by the DC Transfer function dialog in the IsSpice4 Simulation Setup dialog.

```
.TF V(9) V1
```

Transfer Function: The small signal transfer function analysis is used to find the small signal DC gain from any input source, to any output node, input resistance or the output resistance. Only one line is necessary to implement this analysis. The .TF line is used to specify the input source and the output node used for the calculations. The results will be placed in the output file. This example reports the DC small signal gain, from V1 to V(9), to be .997 with 180 degrees of phase shift (indicated by the minus sign). It also reports an input resistance at V1 of 100kΩ and an output resistance looking into V(9) of 30Ω. Only one .TF may be performed for each simulation. The results, as they appear in the output file, are shown below.

The Transfer Function (TF) Output is placed in the IsSpice4 *OUTPUT file.*

```
***** SMALL SIGNAL DC TRANSFER FUNCTION

   output_impedance_at_V(9)    3.038449e+001
   v1#Input_impedance      1.000559e+005
   Transfer_function       -9.96127e-001
```

Operating Point Information: The show and showmod commands are used to extract device and model operating point information. These commands are activated in the IsSpice4 Simulation Setup dialog using the Operating Point button, or by typing in the correct ICL syntax. This example tabulates the operating point information for all transistors for the parameters specified, and all model parameters in the circuit. A small section of the output is shown below;

```
show q :  vbe vbc ...
showmod all
```

```
BJT: Bipolar Junction Transistor
 device    q3                    q2                  q1
 model     qn2907                qn2222              qn2222
 vbe       0.642                 0.607               0.603
 vbc       -9.07                 -9.36               -10
 icc       0.00103               0.000248            0.000216
 ib        1.05e-005             3.24e-006           2.86e-006
 ie        -0.00104              -0.000251           -0.000218
 gm        0.0327                0.00957             0.00832
 go        1.75e-005             2.37e-006           2.04e-006
 ft        4.91e+007             2.6e+007            2.29e+007
 temp      27                    27                  27
```

Using Subcircuits - Active RC Filter

There are four types of devices in IsSpice4; passive elements, active elements, C code models and subcircuits. Thus far, we have limited our examples to primitives. In this example, we will introduce a 4th order Bessel low-pass filter circuit that uses a subcircuit representing a UA741 op-amp. Since the UA741 is too complicated to be represented by only one component, it uses a collection of primitive components put together in a topology that produces the characteristics of a UA741.

__New Topics Covered__
Subcircuits and Models

- X...Subcircuit Call
- .SUBCKT .ENDS

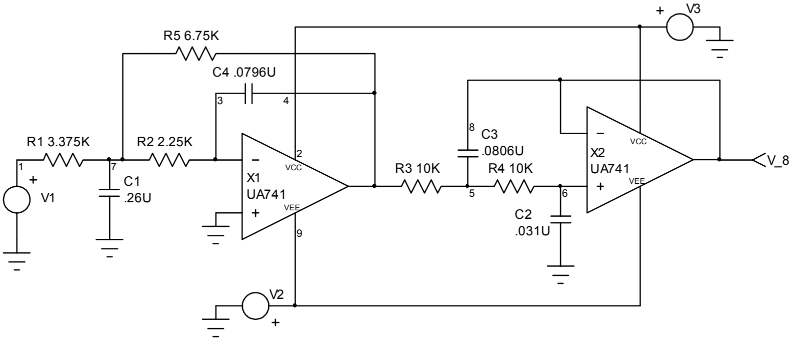
Output

- .PRINT Real Imaginary

The UA741 is located in the Part Browser dialog under the Op-amps Part Type, Intusoft Sub Type listing.

Open up the schematic at :\spice8\Cirsuits\Examples\Subcircuits.DWG

Specifications:
Given the circuit shown below, we will determine the frequency response. The AC stimulus will have be 1. Data will be collected at node 8 from 1Hz to 1GHz at 20 points per decade. The data will be displayed in magnitude and phase, and real and imaginary formats.



```
Using Subcircuits - Active RC Filter
*#SAVE ALL ALLCUR ALLPOW
.SUBCKT UA741 2    3  6   7   4
*            - IN + OUT VCC VEE
RP 4 7 10K
RXX 4 0 10MEG
IBP 3 0 80NA
RIP 3 0 10MEG
CIP 3 0 1.4PF
IBN 2 0 100NA
RIN 2 0 10MEG
CIN 2 0 1.4PF
VOFST 2 10 1MV
RID 10 3 200K
EA 11 0 10 3 1
R1 11 12 5K
R2 12 13 50K
C1 12 0 13PF
GA 0 14 0 13 2700
C2 13 14 2.7PF
RO 14 0 75
L 14 6 30UHY
RL 14 6 1000
CL 6 0 3PF
.ENDS
.AC DEC 20 1HZ 1GHZ
*#ALIAS  VOUT V(8)
.PRINT AC  V(8)  VP(8)
.PRINT AC  VR(8)  VI(8)
R1 1 7 3.375K
R2 7 3 2.25K
C1 7 0 .26U
X1 3 0 4 2 9 UA741
R3 4 5 10K
R4 5 6 10K
X2 8 6 8 2 9 UA741
C2 6 0 .031U
C3 5 8 .0806U
C4 3 4 .0796U
R5 7 4 6.75K
V2 9 0 -15
V3 2 0 15
V1 1 0 AC 1
.END
```

Aliases are generated by the Test points

Subcircuits

Important Note: As with .Model statements, .SUBCKT subcircuit netlists DO NOT have to be typed in.

Node zero is ground in subcircuits. It is the SAME ground that is in the main circuit.

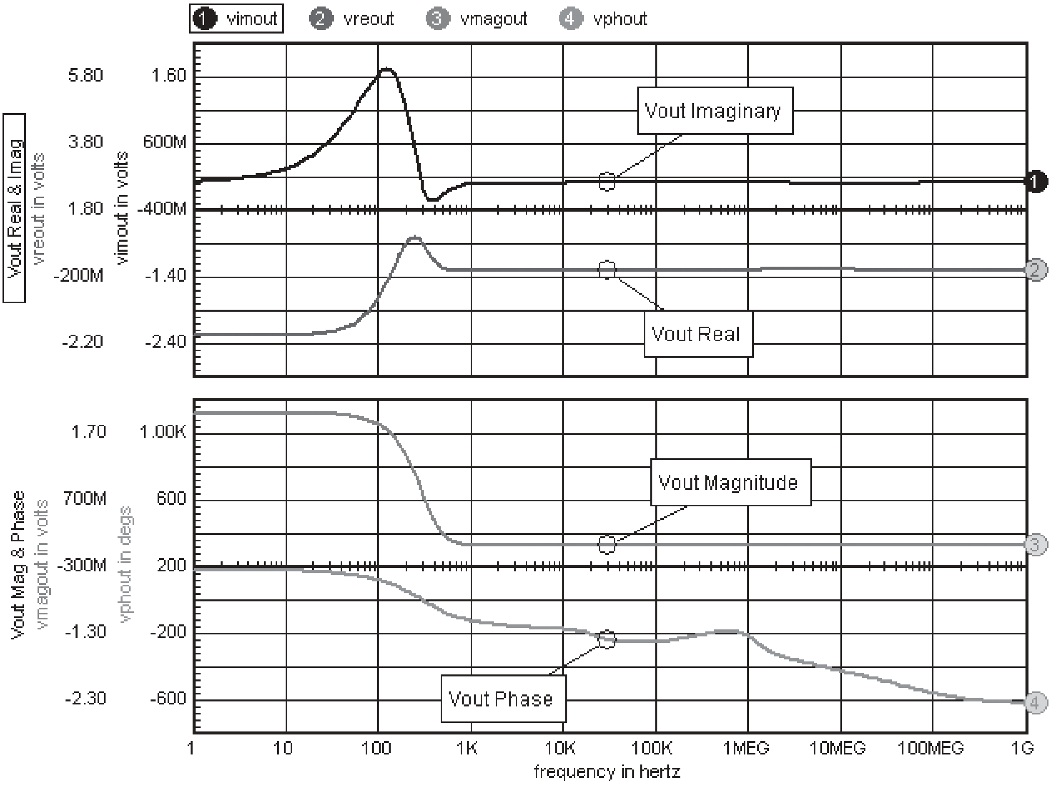
A subcircuit netlist is enclosed by a .SUBCKT statement at the beginning, and a .ENDS statement at the end. All connections into and out of a subcircuit are specified on the .SUBCKT line. The subcircuit name is also given on the .SUBCKT line. This can be seen in the partial listing of the UA741 subcircuit shown below.

```
.SUBCKT UA741 2    3  6   7   4
 *            - IN + OUT VCC VEE
 RP 4 7 10K
 RXX 4 0 10MEG
.
.
.
 L 14 6 30UHY
 RL 14 6 1000
 CL 6 0 3PF
 .ENDS
```

The node list on the .SUBCKT line is important. The order of the nodes determines the connections between the subcircuit netlist and the “X” call line. In this case, the comment line below the .SUBCKT line details the proper connections. The calling line in the main IsSpice4 netlist would look like the following;

X1 3 0 4 2 9 UA741

This call to the UA741 subcircuit would connect 3 to the inverting input, 0 to the non-inverting input and so on. The actual node numbers on the calling line are not important. Only their position is important. Also, there must be the same number of nodes on the calling line as there are on the .SUBCKT line.



Note: The SpiceNet symbol takes care of generating the proper X subcircuit call line. If the symbols are made with the proper pin order relative to the .SUBCKT netlist, then the subcircuit will connect properly to the symbol.

The AC analysis output is shown below. If your output differs or you are unable to obtain this output, look in the error file for any errors and check the circuit values.

Advanced Macro Modeling - Saturable Core

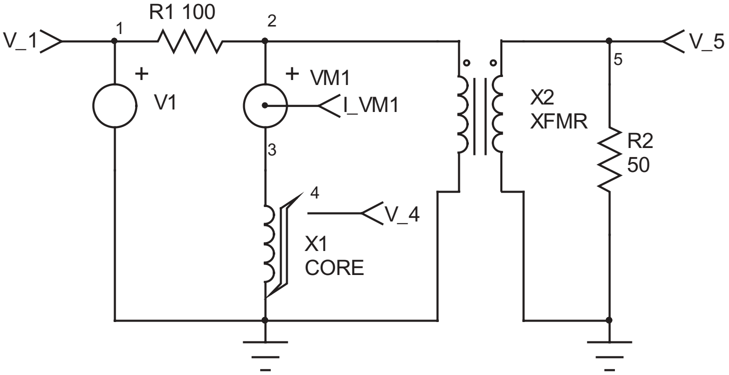
The following example uses 2 parameterized subcircuits, an ideal transformer and a nonlinear saturable core model to perform an analysis of a complete transformer circuit.

New Topics Covered
Subcircuits

- Using Multiple Subcircuits

Open up the drawing at :\spice8\Circuits\Examples\MacroModeling.DWG

Specifications:
The following circuit is used to simulate a transformer and its nonlinear core characteristics. We will investigate the input current, secondary output voltage and the core flux. A transient analysis will be performed and data will be collected every .1µs for 50µs, starting at time 0. The input stimulus will be a pulse train from -5V to 5V with no time delay, 1µs rise and fall times, a 25µs pulse width and a 50µs period.



```
Saturable Core, Advanced Macro Modeling
.SUBCKT CORE#0 1 2 3
RX 3 2 1E12
CB 3 2 1E-7  IC=-250
F1 1 2 VM1 1
G2 2 3 1 2 1
E1 4 2 3 2 1
VM1 4 5
RB 5 2 100K
RS 5 6 200
VP 7 2 250
D1 6 7 DCLAMP
VN 2 8 250
D2 8 6 DCLAMP
.MODEL DCLAMP D(CJO=191P VJ=25)
.ENDS
```

```
.SUBCKT XFMR#0 1 2 3 4
RP 1 2 1MEG
E 5 4 1 2 300M
F 1 2 VM 300M
RS 6 3 1U
VM 5 6
.ENDS
.TRAN .1U 50U
R1 1 2 100
VM1 2 3  0
X1 3 0 4 CORE#0
*{VSEC=50U IVSEC=-25U LMAG=10MHY
+ LSAT=20UHY FEDDY=20KHZ }
X2 2 0 5 0 XFMR#0
* {RATIO=.3}
R2 5 0 50
V1 1 0 PULSE -5 5 0 0 0 25U 50U
.END
```

The CORE part is located under Magnetic Cores Part Type, Generic Sub Type. The XFMR part is located under Transformers.

The magnetic core and transformer elements are generic. They can emulate many different kinds of nonlinear cores and transformers simply by specifying a set of parameters before running the simulation.

- **After placing the CORE symbol, open the Properties dialog by double-clicking on the symbol.**

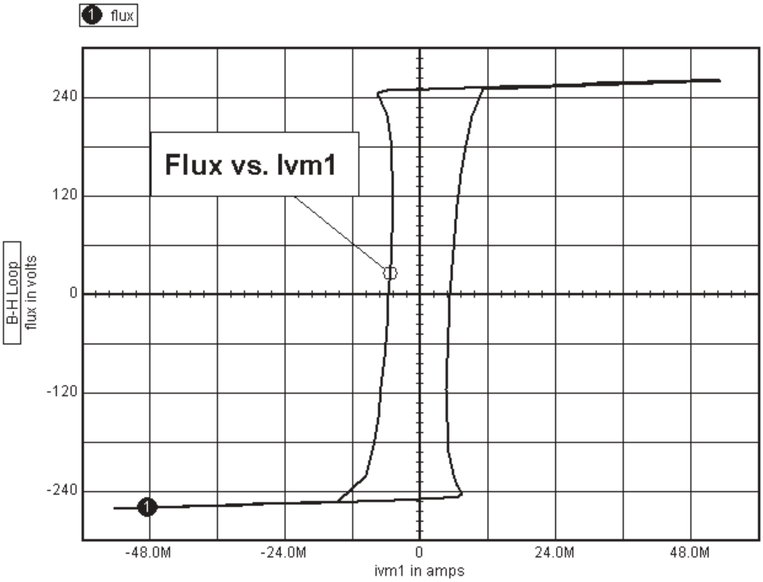
- **Enter the parameters values in the correct Properties fields using the following information:**

VSEC=50u IVSEC= -25u LMAG=10m LSAT=20u FEDDY=20k

- **You will need to do the same with the XFMR part. Use a parameter value of “RATIO=.3”.**

Most subcircuits have default values for their parameters. If so they will be listed in the properties dialog.

- **To make the X-Y plot, select the v(4) md + (vm1) Add Waveform dialog.**



If any errors occurred, remember to examine the error file for error messages.

The results for this example are shown in the following graph:

- **The waveform is created in IntuScope using an X-Y plot of the current through the core (vm#1branch = I(VM1)) versus the core flux.**

A voltage equal to the core flux is brought out from inside the subcircuit to a test point on the symbol. To measure the core flux, simply attach a test point symbol to the symbol pin.

Although this example does not introduce any new IsSpice4 syntax, it does introduce a new concept, macro modeling. Macro modeling is the name given to the process of combining IsSpice4 primitive components in such a way that they perform the function of a single device.

The Core subcircuit uses the macro modeling concept to model the nonlinear saturation effects present in magnetic core materials. A complete explanation of the core model is located in the Working with Models book.

In-line Math Equations - AM Modulator

This example introduces the nonlinear dependent source (B element). The B element can generate an output voltage, or current, based on a transfer function or mathematical expression. Equations can use a predefined set of math functions and any voltage or current in the circuit. As an example of the power of this source, we will simulate a simple AM modulator circuit.

__New Topics Covered__

Sources

- B...Nonlinear Dependent Source

Specifications:

An amplitude modulated signal, s(t), is defined by combining a carrier,

$$c(t) = A_C \cos(2\pi f_C t)$$

and a message, m(t), by the following equation;

$$s(t) = [1 + k_a m(t)] c(t)$$

or,

$$s(t) = A_C[1 + K_a m(t)] \cos(2\pi f_C t)$$

where k_a is the amplitude sensitivity. If we consider a single tone modulation such as;

$$m(t) = A_m \cos(2\pi f_m t)$$

so that the amplitude modulated signal becomes;

$$s(t) = A_C[1 + K_a Am \cos(2\pi f_m t)] \cos(2\pi f_C t)$$

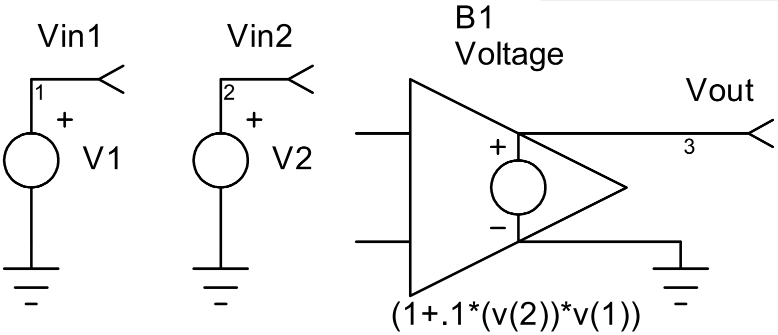
This equation can be generated by the nonlinear dependent source quite easily. The equation for the nonlinear source would be

v = (1 + V(1)) * V(2)

The netlist and circuit for the this example are shown as follows.

```
.TRAN .5u 1m 0 .5u
.PRINT  TRAN Vin1
.PRINT  TRAN Vin2
.PRINT  TRAN Vout
V1 1 0 SIN 0 1 100k
V2 2 0 SIN 1 100m 10k
B1 3 0 V=(1+.1*(v(2))*v(1))
.END
```

Open up the drawing at :\spice8\Circuits\Examples\AMModulator.DWG



We could have generated a current output with the B source by replacing “v=” with “i=”. For example, B1 3 0 I=(1+.1*(v(2))*v(1)).

The final signal generated at the output of the B source is;

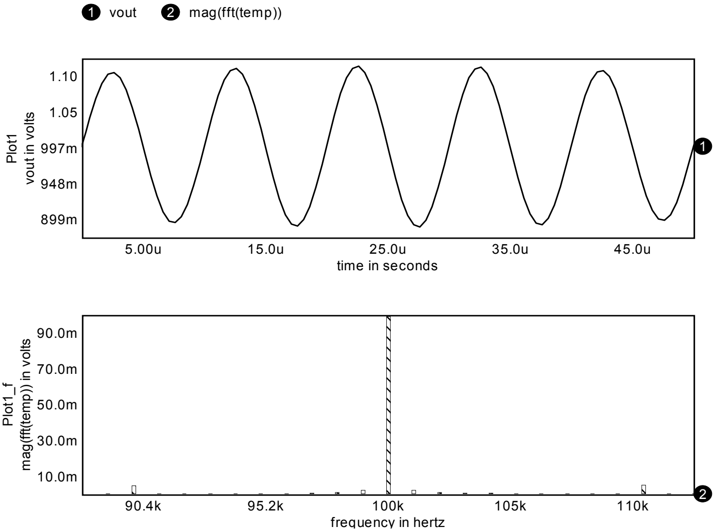
$$s(t) = 1*[1+.1\cos(2\pi 10kt)]\cos(2\pi 100\,kt)$$

where Ac=1, Am=1, ka=.1, fc=100kHz, and fm=10kHz.

A Fourier transform of this equation yields a spectrum of an AM signal. Keeping to positive frequencies, delta functions appear at fc, fc+fm and fc-fm.

Both the time domain and frequency domain waveforms of V(3), alias Vout, are shown in the graph on the next page.

IntuScope can take the FFT of any time domain waveform (FFT of s(t) shown on the next page). Both Time-to-Magnitude and Time-to-Real/Imaginary transforms are supported.



If-Then-Else - A Comparator

The B source is also capable of making decisions using an If-Then-Else expression. This is extremely useful when modeling components or portions of a circuit where the output depends on the state of the input compared to a known constant (e.g., a comparator).

<u>**New Topics Covered**</u>

Sources

- B...Nonlinear Dependent Source

Specifications:

The comparator will have an on voltage of 5V and an off voltage of 0V. The comparator will trigger when the input voltage difference is greater than zero. The easiest way to model this comparator is with an If-Then-Else function. The output of a comparator will change state depending on the condition of the two inputs. This function can be written as; **If** the difference between the inputs is greater than zero, **Then** the output is high, **Else** the output is low. The syntax for this particular device is:

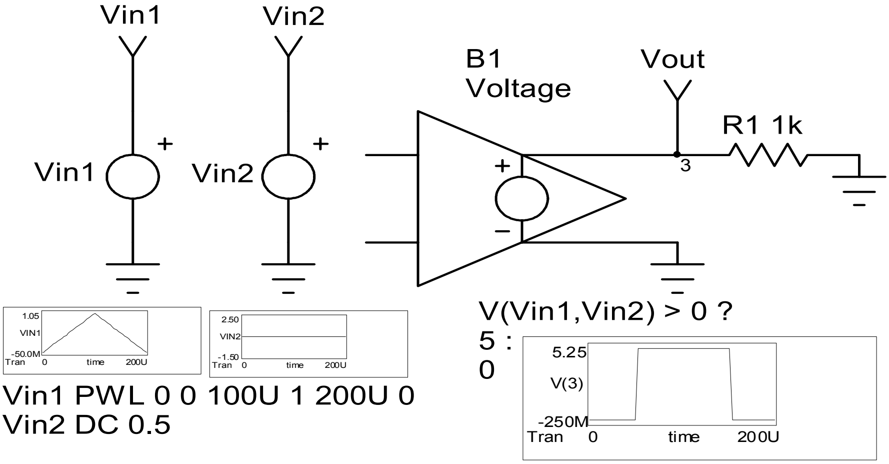
```
B1 3 0 v= V(1,2) > 0 ? 5 : 0
```

This can be read as:

If (V(1,2) > 0) Then V(3)=5

Else V(3)=0

Open up the drawing at :\Spice8\Circuits\Examples\Compar.dwg

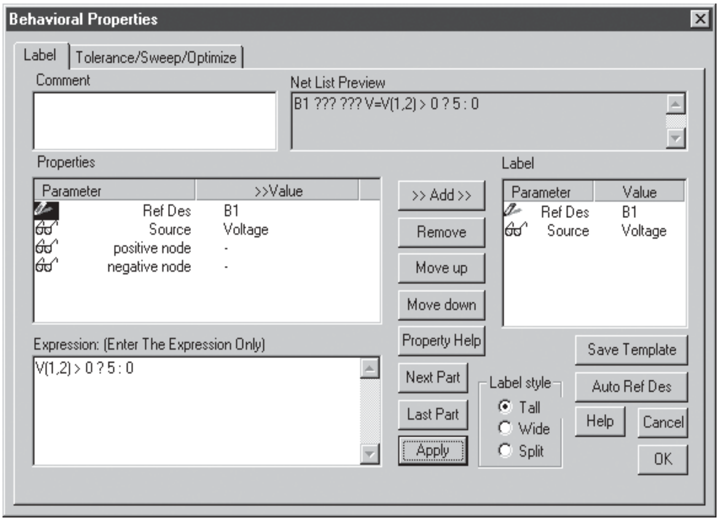


Vin1 PWL 0 0 100U 1 200U 0
Vin2 DC 0.5

V(Vin1,Vin2) > 0 ?
5 :
0

The B element can be used directly in a schematic, however, it has no nodal input connections making schematic capture arkward. This is due to the fact that the input to a B element is an arbitrary expression and can be made of node voltages, device currents, and other quantities.

An example B element dialog is shown next. It should be noted that if you refer to a node voltage or device reference designation it is best to give the node or ref-des a unique identifier, such as VTEST for a node and ROUT for a resistor ref-des. This will prevent SpiceNet from changing the node or ref-des during the course of capturing the design, and hence breaking any expressions you may have entered for B elements. In our example drawing we renamed v(1) and v(2) to vin1 and vin2 respectively.



Behavioral Properties

Label | Tolerance/Sweep/Optimize

Comment

Net List Preview

```
B1 ??? ??? V=V(1,2) > 0 ? 5 : 0
```

Properties

Parameter	>>Value
Ref Des	B1
Source	Voltage
positive node	-
negative node	-

Label

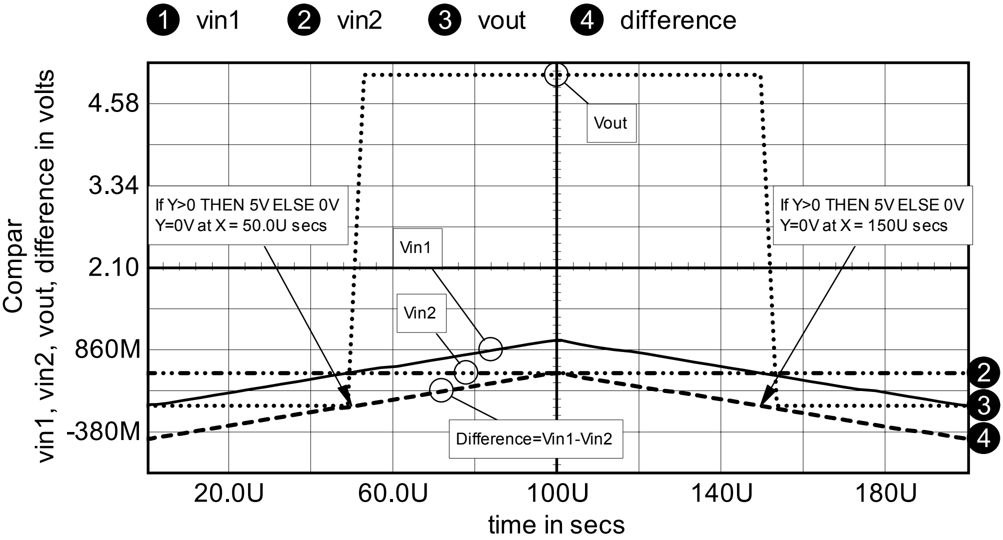
Parameter	Value
Ref Des	B1
Source	Voltage

Expression: (Enter The Expression Only)

```
V(1,2) > 0 ? 5 : 0
```

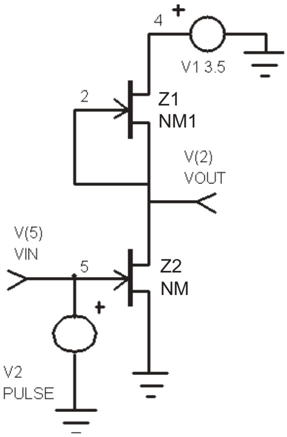
Label style
● Tall
○ Wide
○ Split

The results of the simulation are shown below.



GaAs MESFETs - An Inverter

Open up the drawing at :\spice8\ Circuits\ Examples\ GaAsMEFET.DWG



The .PRINT, statement is entered in the User Statement field in the IsSpice4 *Simulation Setup dialog.*

IsSpice4 includes GaAs MESFET models based on the Statz and Parker-Skellern models. The model parameters are detailed in the IsSpice4 User's Guide.

New Topics Covered

Sources

- Z...GaAs MESFET Model

Output

- .PRINT...Computed Device Parameters

Specifications:

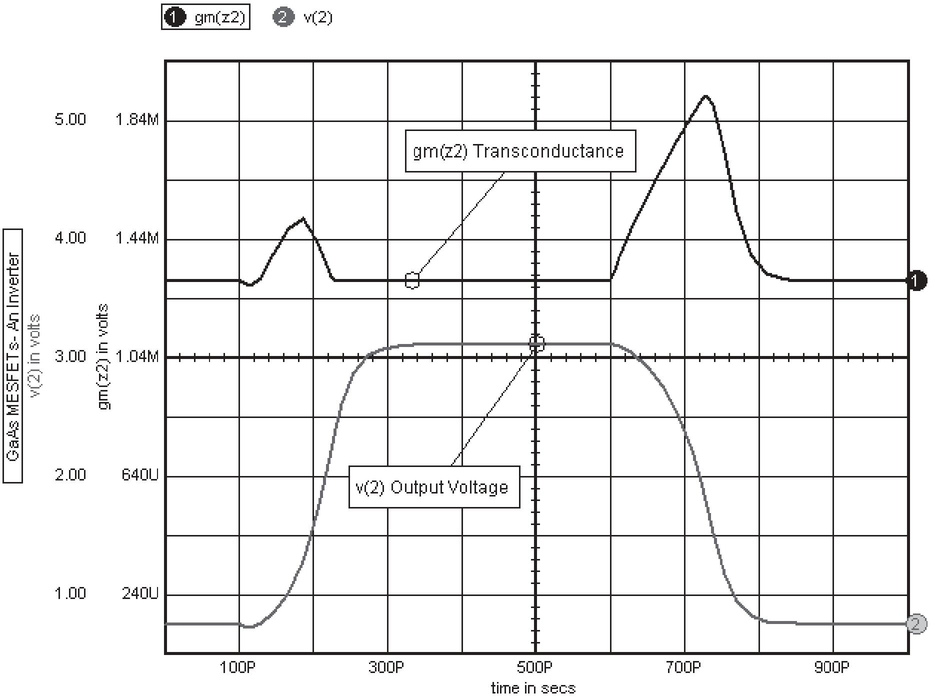
Using the circuit below, we will investigate the response of the GaAs inverter to an input clock with 125ps rise and fall times, and a pulse width of 375ps.

```
An Inverter - GaAs MESFET
*#SAVE ALL ALLCUR ALLPOW
.TRAN 10P 1N
.MODEL NM NMF (VTO=-2.0 BETA=1M LAMBDA=1E-4
+ RD=100 RS=100 CGS=40F CGD=5F PB=.6)
.MODEL NM1 NMF (VTO=-1.8 BETA=.8M LAMBDA=1E-4
+ RD=100 RS=100 CGS=70F CGD=15F PB=.6)
.PRINT TRAN @Z1[GDS] @Z2[GDS] @Z1[GM] @Z2[GM]
.VIEW TRAN V(2) 0 3.5
.VIEW TRAN V(1) -1 0
.VIEW TRAN @Z1[GDS] 0 4M
.VIEW TRAN @Z2[GDS] 0 2.50U
.VIEW TRAN @Z1[GM] 0 4M
.VIEW TRAN @Z2[GM] 0 2M
V2 1 0 PULSE 0 -1 100P 125P 125P 375P 1
Z1 4 2 2 NM1
Z2 2 5 0 NM
V1 5 0 3.5
.END
```

The GaAs MESFET is called by pressing the number 4 and the letter Z. Once the MESFET is placed on the schematic, a .MODEL statement must appear for every MESFET with a different model name. In this example, two .MODEL statements must be made, one for "MN" and one for "MN1." If the model was the same for both devices, then only one .MODEL line would be needed. You can enter the model parameters for each MESFET through the MESFET's Part Properties dialog. The parameter values can be seen from the netlist on the previous page.

The .PRINT command uses the reference designators listed in the circuit, and the MESFET parameters listed in the on-line help (press Property Help in the MESFET Properties dialog). This obtains the transconductance and drain-source conductance for both MESFETs. These device related parameters are not generated by the test point symbol. They must be typed in by hand by inserting a .PRINT statement in the IsSpice4 Simulation Setup dialog, User Statements area.

The output for this example is shown below.



Simulating at Different Temperatures - Simple Amplifier

In this example, we will demonstrate how to simulate circuits and components at different temperatures. This can be extremely useful when determining the reliability of a circuit design in a harsh environment, or when an active device in the circuit is subjected to heat stress.

<u>**New Topics Covered**</u>

Temperature

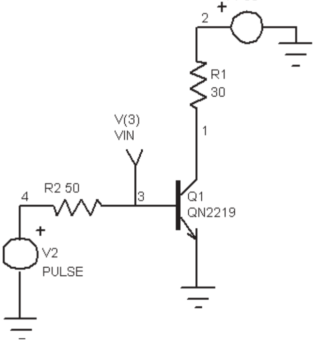
- .OPTION TEMP Parameter
- Temperature Control of Active Devices
- Resistor Temperature Coefficients

Analysis Control

- .TRAN...TSTART, TSTOP, TSTEP

Specifications:

Given the circuit shown, we will investigate the effect of temperature variations on the base of the transistor. The input stimulus will be a pulse train with an initial value of 0V, a peak value of 5V, rise and fall times of 10ns, a pulse width of 500ns and a period of 1μs. Data will be collected from 400ns to 1.2μs, in 1ns steps. Output will be generated for node 3. The first simulation will be performed at room temperature. The second simulation will be performed with the entire circuit at 75°C. The third simulation will be performed with the temperature of the transistor set to 75°C and the rest of the circuit at room temperature.



Open up the drawing at :\spice8\ Circuits\ Examples\ Temperature.DWG

Use the Properties dialog to enter the values for TC1 and TC2 for RMOD. The dialog will generate the RMOD .Model statement.

```
Simulating at Different Temperatures
*#SAVE ALL ALLCUR ALLPOW
.MODEL RMOD R(TC1=.01 TC2=1E-6)
.PRINT TRAN  V(3)
R1 1 2 30  RMOD ; enter Rmod in the .Model Value field
V1 2 0 30
R2 3 4 50  RMOD
V2 4 0 PULSE 0 5 0 10N 10N 500N 1U
Q1 1 3 0 QN2219
.END
```

- **Construct and simulate the circuit. If you encounter any errors, remember to examine the error file. Be careful to enter the schematic exactly as shown.**

Notice that the .TRAN line has four parameters. In addition to the TSTEP and TSTOP parameters, we have added the TSTART and TMAX parameters. The TSTART parameter is used to tell IsSpice4 to start collecting data after 400ns. The simulation will still start at time zero. However, IsSpice4 will not record any data until 400ns. This parameter is used to reduce the amount of data saved in the output file. The last parameter on the .TRAN line is used to set the upper limit on the internal IsSpice4 timestep. Setting this parameter to a value less than TSTEP will improve the data point interpolation and thus increase the accuracy of the results placed in the output file.

You will also notice a .MODEL statement for a resistor in the netlist. This statement is used to place the temperature coefficients on all of the resistors in the circuit. The name, “RMOD,” which is used to reference the .MODEL statement, occurs on all of the resistors. You only need to enter the RMOD model name once in one of the resistor properties dialogs, along with the TC1 and TC@ values. You can then select the RMOD model while in the other resistor properties dialogs. Thus, all of the resistors will use the same temperature coefficients.

This simulation will produce output at the default temperature setting, 27°C.

Use the IsSpice4 Simulation Setup dialog to set the circuit temperature.

- **Set the circuit temperature to 75°C by using the "Circuit Temperature..." button in the IsSpice4 Simulation Setup dialog.**

The behavior of the entire circuit will be simulated at a temperature of 75°C. All resistors and active devices in the circuit will be adjusted, based on the new simulation temperature.

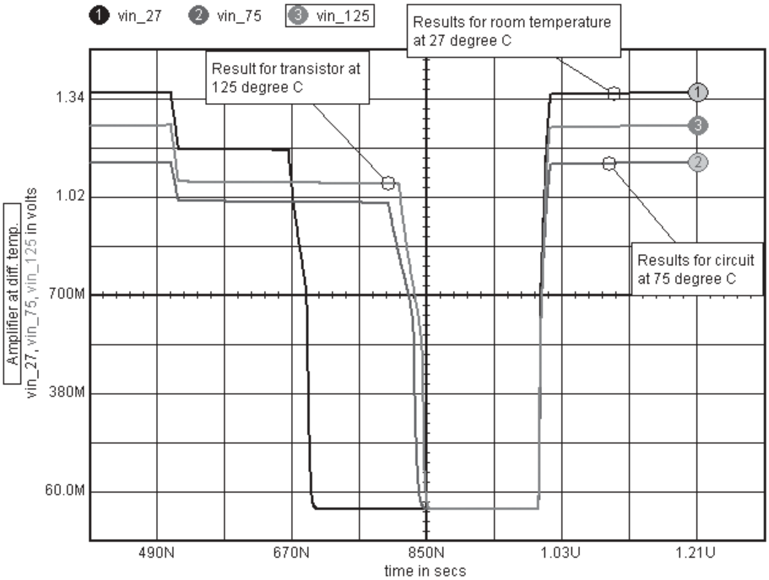
- **Uncheck the "Circuit Temperature..." button in the IsSpice4 Simulation Setup dialog.**

- **Enter "125" in the Temp field in the Properties dialog of the QN2219 transistor.**

Q10 1 3 0 QN2219 TEMP=125

This will cause the transistor to be simulated at 125°C. The rest of the circuit will be simulated at room temperature.

The results for the three simulations are shown below.



See the Alter dialog for more information on sweeping the circuit temperature.

The temperature set on an active device will override the temperature set in the .OPTIONS statement. With this feature, it is easy to simulate a device at one temperature, and the rest of the circuit at another temperature.

- **Set the operating temperature of the circuit to 50°C.**

- **Simulate the circuit and examine the output.**

UIC\IC - Establishing Initial Conditions

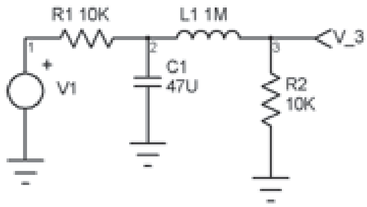
In certain circumstances, it is necessary or just convenient to set the circuit to a particular state prior to the start of a transient simulation. This is done by setting initial conditions and telling IsSpice4 to use them.

New Topics Covered
Sources

- IC=...Initial Conditions

Specifications:
Given the circuit shown, we will examine the effect of initial conditions on the state of the output.

```
Using Initial Conditions
*#SAVE ALL ALLCUR ALLPOW
.TRAN 1M 1000M
.PRINT TRAN  V(3)
V1 1 0 PULSE 0 10
L1 2 3 1M
R2 3 0 10K
R1 1 2 10K
C1 2 0 47U
.END
```



Open up the drawing at :\spice8\Circuits\Examples\Uic.DWG

- **Simulate the circuit.**

The control statement, .IC V(1)=2.5, will have the same effect as the "IC=" on the capacitor.

Notice that the output voltage requires about 800ms to reach a steady state value. In some circuits, this may be an obstacle. By setting the initial conditions of IsSpice4, we can overcome this initial charge time.

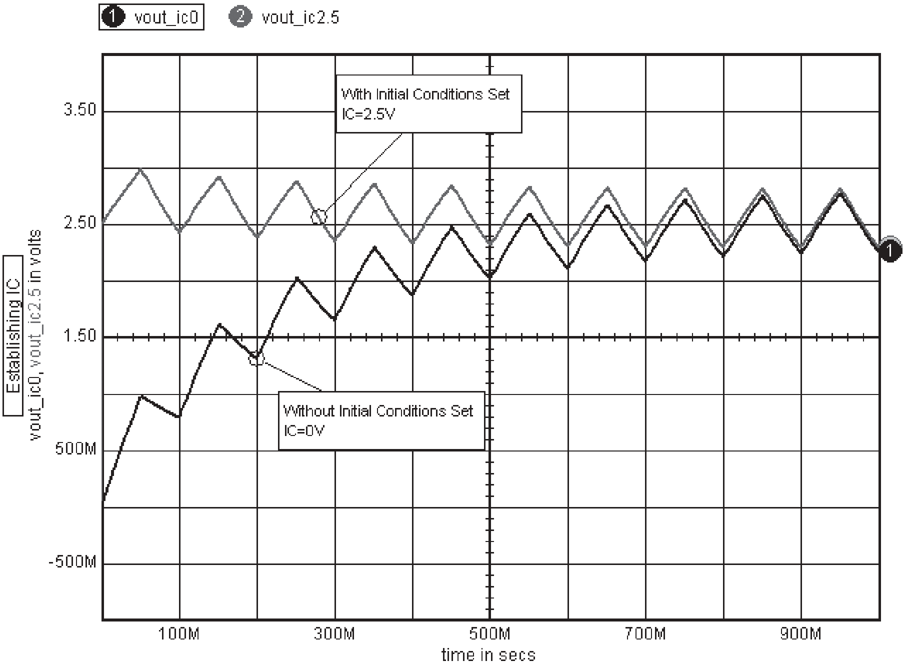
- **Enter "2.5" in the IC value field of the capacitor from the Properties dialog. Then, click on the Transient button in the IsSpice4 Simulation Setup dialog.**

The "UIC" parameter must be present on the .TRAN line in order to obtain the proper results.

- **Check the "Use Initial Conditions" (UIC) check box. Select OK and Done.**

This forces IsSpice4 to start the transient analysis with an initial voltage of 2.5V on the capacitor. It also eliminates the charge time and brings the circuit to steady state more quickly.

The results for the two simulations are shown below.



Using Phase Delayed Sources

The Sin, Pulse, and FM sources include the ability to add an initial phase delay at time = 0.

The phase delay parameter is available in the voltage and current sources' Tran Generators sub-dialog.

Note: The pulse delay is different from the Td (time delay) parameter. The Td parameter only delays, in time, the zero phase version of the pulse waveform.

<u>**New Topics Covered**</u>

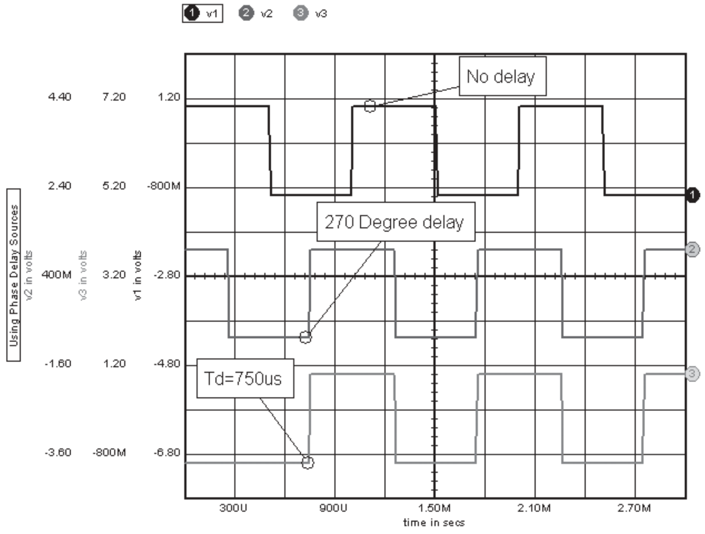
Sources

- Delaying the initial phase of a periodic stimulus

Specifications:
The following statements may be used to generate the waveform stimulus shown below.

The phase parameter is specified in degrees, and is included at the end of the statement.

```
V1 1 0 Pulse(-1 1 0 10U 10U 500U 1M)       ; No delay, WFM.1
V2 2 0 Pulse(-1 1 0 10U 10U 500U 1M -270); 270 Degree delay, WFM.2
V3 3 0 Pulse(-1 1 750U 10U 10U 500U 1M)  ; Td=750µs, WFM.3
```



Using Code Models - Laplace Equations

Code Models can be created using the Intusoft Code Modeling Kit (CMSDK)

Code modeling is a new approach to making SPICE models. The process enables you to add customized C code subroutines to IsSpice4. The C code describing the model's behavior is linked to the simulator via an external DLL file, rather than being bound within the executable program. A variety of analog, digital and mixed analog/digital code models are already included with IsSpice4. These models behave exactly like traditional SPICE models, but their model parameters are configured directly within the schematic, rather than in a separate library file.

New Topics Covered

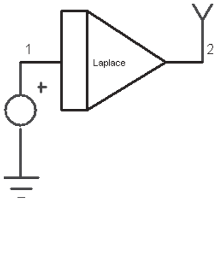
Code Models

- Laplace Code Model

Open up the drawing at :\spice8\ Circuits\ Examples\ Laplaceq.DWG

Specification:

To understand how to access and work with code models, let's experiment with a simple filter circuit using the Laplace code model. The Laplace code model allows you to enter any arbitrary s-domain transfer function as a ratio of two polynomials in s. We will take care of this element in a moment. The voltage source should have the value of "Pulse 0 1 AC 1" since we will be running AC and transient analyses. Enter the analysis statements ".AC DEC 10 .1 1K" and ".TRAN .1 10" in the IsSpice4 Simulation Setup dialog.



```
CHEBYSHEV
*#SAVE ALL ALLCUR ALLPOW
.MODEL S_001 s_xfer(in_offset=0.0 gain=1.0 num_coeff=[1.0]
+ den_coeff=[1.0 1.42562 1.5162] out_ic=0.0 denorm_freq=1.0)
.AC DEC 10 .1 10K
.TRAN .1 10
.PRINT AC  V(2)  VP(2)
.PRINT TRAN  V(2)
A6 1 2 S_001
V1 1 0 PULSE 0 1 AC 1
.END
```

The .SUBCKT netlist does not have to be typed into IsEd. Use the Laplace3 symbol.

Using Code Models

Let's enter the information to create the 3kHz Chebyshev low pass filter with the transfer function using the first method.

$$\frac{1}{s^2 + 1.42562\, s + 1.5162}$$

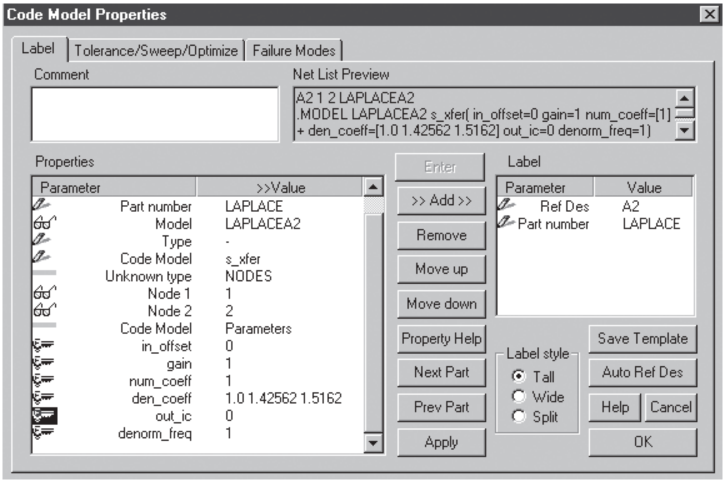
- **Select the X Part...function. From the Browser dialog, select Behavioral. Then click on the Laplace entry. Click the Place button.**

- **Double-click on the Laplace model to enter the parameter values In the Properties dialog.**

- **Type in the following parameter values that we will pass to the Laplace model.**

  num_coeff=1.0
  den_coeff=1.0 1.42562 1.5162
  (corresponding to S2=1.0 S1=1.42562 S0=1.5162)
  dnorm_freq=1.0

- **Close the dialog and run the simulation.**



Code Model Properties

Label | Tolerance/Sweep/Optimize | Failure Modes

Comment

Net List Preview

```
A2 1 2 LAPLACEA2
.MODEL LAPLACEA2 s_xfer( in_offset=0 gain=1 num_coeff=[1]
+ den_coeff=[1.0 1.42562 1.5162] out_ic=0 denorm_freq=1)
```

Properties

Parameter	>>Value
Part number	LAPLACE
Model	LAPLACEA2
Type	-
Code Model	s_xfer
Unknown type	NODES
Node 1	1
Node 2	2
Code Model	Parameters
in_offset	0
gain	1
num_coeff	1
den_coeff	1.0 1.42562 1.5162
out_ic	0
denorm_freq	1

Label

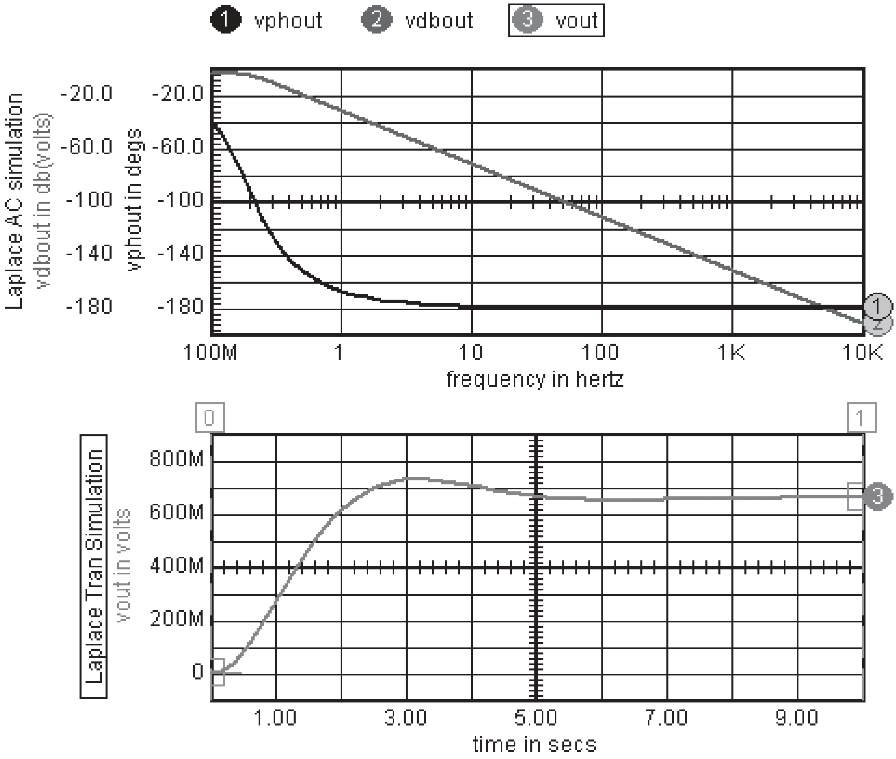
Parameter	Value
Ref Des	A2
Part number	LAPLACE

Label style: ● Tall ○ Wide ○ Split

Code models are more versatile than traditional SPICE models. SPICE model parameters are generally entered as a single real-valued number. Code model parameter values can be integer, real, boolean (True, False), complex, string, vector (a set of numbers) or an n-dimensional array of values. For example, the set of denominator coefficients for the Laplace model comprises a single model parameter.

If desired, you can see how the model is saved in the netlist by selecting the Edit Text Files Icon from the SpiceNet toolbar.

The results for the AC and transient simulations are shown below. You can adjust the numerator and denominator coefficients in the Properties dialog to see different results.



Note: If you want to use another configuration of the Laplace model, such as a Laplace equation with a 2nd-order numerator and 4th-order denominator, you can simply enter a different set of coefficients.

Accessing Multiple Models

It is not necessary to enter model parameters for every code model instance unless you want to change the parameters.

- **Select the Laplace symbol. Copy and paste another Laplace symbol.**

- **Double-click on the second symbol and then open the Properties dialog.** Notice that the parameters for this symbol are the same as those of the first symbol. Both symbols will refer to the same models.

Open up the drawing at :\spice8\Circuits\Examples\PIDC.DWG

Shown next is a circuit for a PID controller that uses the Laplace and Differentiator code models. The netlist and results are shown for comparison. Note: a SPICE 2 style Integrator (SINT) is used in place of a simple Laplace (1/s) block.

```
PID CONTROLLER
*#SAVE  ALL ALLCUR ALLPOW
.MODEL S_001 s_xfer(in_offset=0.0 gain=2.0 num_coeff=[1.0]
+ den_coeff=[1.0 2.0] denorm_freq=1.0)
.MODEL S_002 s_xfer(in_offset=0.0 gain=10.0 num_coeff=[1.0]
+ den_coeff=[1.0 5.0] denorm_freq=1.0)
.MODEL S_003 s_xfer(in_offset=0.0 gain=1.0 num_coeff=[1.0]
+ den_coeff=[1.0 500.0M] denorm_freq=1.0)
.MODEL DIFF_001 d_dt(out_offset=0.0 gain=1.5 limit_range=1.0u)
.AC DEC 60 .1 100
.TRAN .05 10 0 .05
*INCLUDE SYS.LIB
*INCLUDE CM1.LIB
*#ALIAS  V_P  V(7)
*#ALIAS  VOUT V(10)
*#ALIAS  V_I  V(4)
.PRINT AC  V(7)  VP(7)  V(10)  VP(10)
.PRINT AC  V(5)  VP(5)  V(4)  VP(4)
.PRINT TRAN  V(7)  V(10)  V(5)  V(4)
.PRINT TRAN  V(9)
X8 9 10 6 SUM2 {K1=1 K2=-1 }
V1 9 0 AC 1 PULSE 0 1 .5
A4 1 3 S_001
A6 3  10 S_002
A7 2 1 S_003
X9 6 7 GAIN {K=.8 }
X10 6 4 SINT {K=.5 }
A8 6 5 DIFF_001
X6 7 4 5 2 SUM3 {K1=1 K2=1 K3=1 }
.END
```

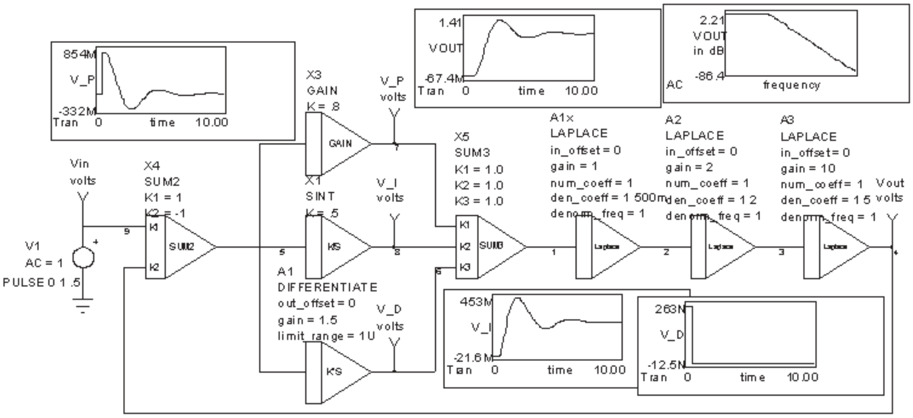


Table Model Transfer Functions

Note: To enter the proper values for the Laplace model, use the Laplace Properties dialog.

IsSpice4 allows you to specify arbitrary transfer functions using tables of X, Y points. This is accomplished with the PWL (piecewise linear) code model, which is often called a Table model.

The model parameters listed in each code model, and the parameters passed to Sum2, Sum3, Gain and SINT, must be entered into the Properties dialog of each element.

New Topics Covered

Code Models

- Table Code Model
- Entering Vector Model Parameters

Open up the drawing at :\spice8\ Circuits\ Examples\ Table.DWG

Specification:
Enter the circuit as shown below. The code model can be selected from the Parts Browse dialog under the Behavioral Type, Function subtype. The model parameters for the Table model are entered in the Properties dialog. Let's enter the data for the Table model.

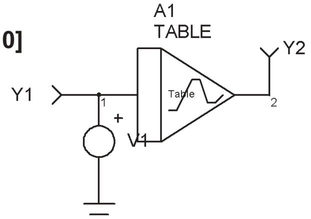
- **Double-click on the Table symbol.**

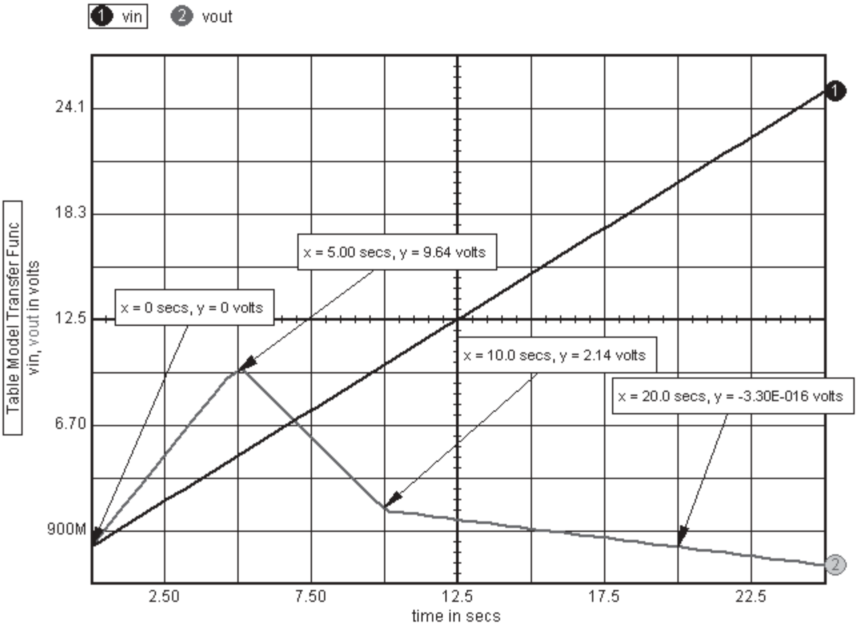
A table model's input/output relationship is described by a series of X, Y, data points. There are 4 pairs in this case, with the output linearly interpolated between points.

- **Enter the data points listed below in the PWL_001 model into the xy_array field.**

Notice that the transfer function continues past 20, using the slope that was established by the highest two points.

```
TABLE Test Circuit
*#SAVE ALL ALLCUR ALLPOW
.MODEL PWL_001  Pwl(xy_array=[0  0    5  10    10  2    20  0]
+ input_domain=10m fraction=true)
.TRAN 1 25
A3 1 2 PWL_001
V1 1 0 PULSE 0 25 0 25
.END
```





Using Table Models From Other SPICE programs

Other SPICE programs may use a similar format for table-type models. If the data points are in an X,Y sequence, you can simplify the process via the following:

- Using any Windows text editor, copy the points from the existing netlist to the clipboard using (<Ctrl>+ C).
- Switch programs to SpiceNet.

- In the PWL Properties dialog, you can paste in the data points by pressing (<Ctrl>+ V) in the xy_array=[....] parameter field.

Switches - A Comparator With Hysteresis

In this example, we will repeat a prior example using a switch.

New Topics Covered

Primitive Element

- S...Voltage Controlled Switch

The default resistance values for the switch model are: RON=1Ω and ROFF=1E12Ω

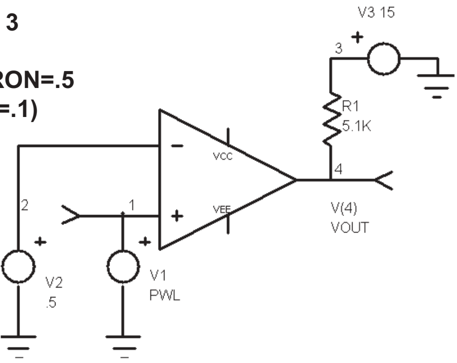
Specifications

We will construct a model of a comparator with hysteresis. As in the previous example, the on state will occur when the input voltage difference is greater than zero, and the off state will occur for every other value. Be sure to create a comparator symbol as stated in the previous comparator example.

- **Construct and simulate the circuit using a netlist approach as shown below.**

Open up the drawing at :\spice8\Circuits\Switches.DWG

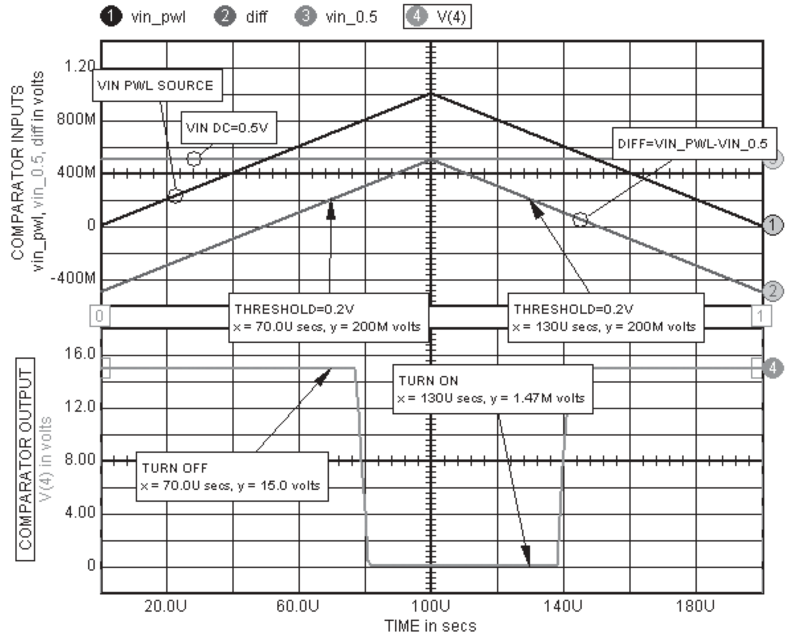
```
Switches - A Comparator With Hysteresis
*#SAVE ALL ALLCUR ALLPOW
.TRAN 1U 200U
.PRINT TRAN  V(1)  V(4)
V1 1 0 PWL 0 0 100U 1 200U 0
V2 2 0 .5
R1 4 3 5.1K
V3 3 0 15
X1 2 1 4 COMPAR
.SUBCKT COMPAR 1 2 3
S1 3 0 2 1 SWMOD
.MODEL SWMOD SW(RON=.5
+ ROFF=10G VT=.2 VH=.1)
.ENDS
.END
```



Current controlled switches are also available in IsSpice4*.*

As you can see, a different approach was used for this comparator. The switch model in IsSpice4 is used because it contains parameters to model hysteresis. Since the hysteresis is built into the switch model, the construction of a comparator with hysteresis becomes quite easy.

The .MODEL statement for the switch specifies a threshold voltage of 0.2 volts. This means that as soon as the difference between the inputs exceeds 0.2 volts, the switch will change state. The parameters RON and ROFF specify the on/off resistance levels of the switch. In this case, when the switch triggers, the output will change to a low resistance state, .5Ω. When the switch is off, the output will be in a high resistance state, 10GΩ. The hysteresis is specified by the VH parameter. With hysteresis, the comparator will be on at a voltage of .8V (Vref+VT+VH) and off at a voltage of .6V (Vref+VT-VH). Since the output in this example changes between a high resistance state and a low resistance state, the comparator simulates an open collector output stage. The results of the simulation are shown below.



Digital/Mixed Mode Examples

Analog-Digital Interfacing

For the purposes of these examples, A2D and/or D2A bridges are used to interface analog nodes with digital nodes. These bridges are not required; if they are not present, the software will automatically insert them in the netlist file, as necessary.

The purpose of this example is to introduce node translation. Node translation is performed throughout the remaining examples in this chapter. It is an important concept when dealing with mixed-mode circuits.

New Topics Covered

Code Models

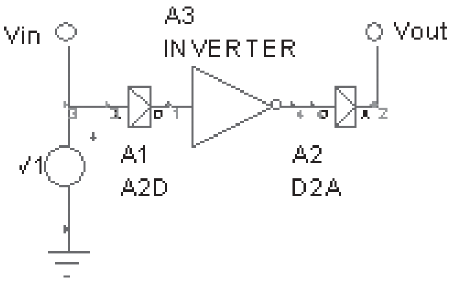
- Inverter Code Model
- Hybrid Code Models
- Configuring Code Models

Specification:

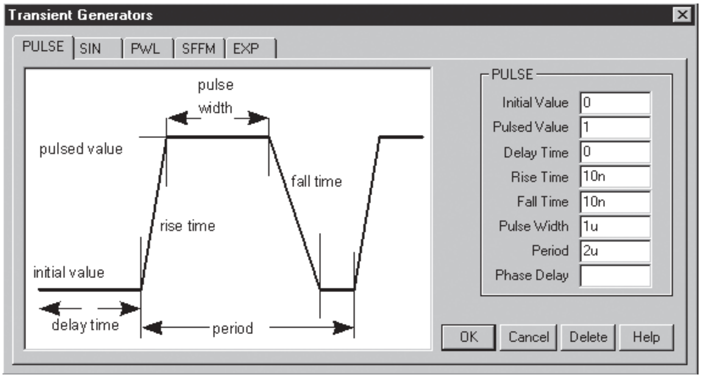
A transient analysis will be performed on a simple inverter circuit. A 500kHz square wave (V1, Pulse source), with 10ns rise and fall times, will be used as the input. Input and output node data will be collected for 10us, in 10ns steps. The rise and fall times of the D-to-A node bridge will be set to 100ns. The rise and fall delays of the inverter will be set to 500ns. Delays in the Analog-to-Digital (A-to-D bridge) will be set to a negligibly small value of 1ps.

Node Translation Using Hybrids

The A-to-D and D-to-A node bridges use the A2D and D2A symbols, respectively.



- **Double-click on the voltage source then double-click on "None" in the Tran Generators option to enter the following values.**



PULSE

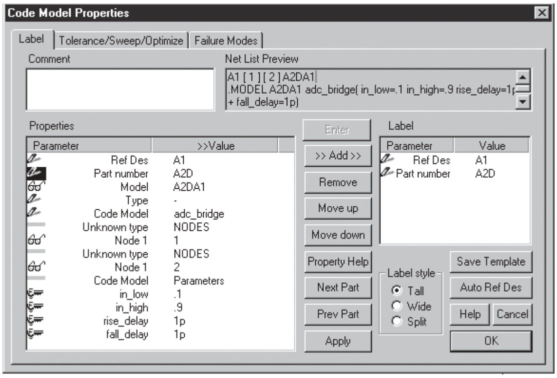
Initial Value	0
Pulsed Value	1
Delay Time	0
Rise Time	10n
Fall Time	10n
Pulse Width	1u
Period	2u
Phase Delay	

Placing and Configuring The Code Models

From the Part Browser dialog, place the symbols as shown above. The inverter is located in the Digital Part types. The A2D and D2A bridges are located under the Bridges Part Type.

- **After the parts are placed, double-click on the A2D symbol to open the Part dialog.**

- **Then click on the appropriate fields to enter the new parameter values.**



Comment

Net List Preview

```
A1 [ 1 ] [ 2 ] A2DA1
.MODEL A2DA1 adc_bridge( in_low=.1 in_high=.9 rise_delay=1p
+ fall_delay=1p)
```

Properties

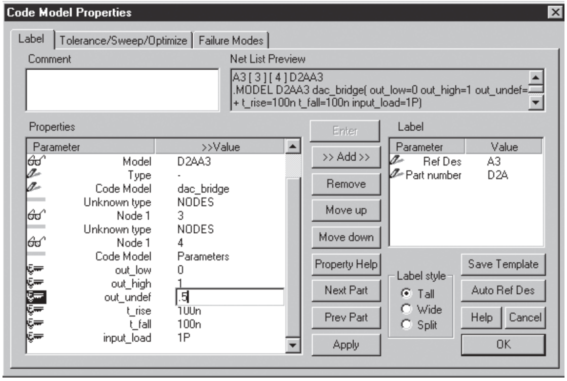
Parameter	>>Value
Ref Des	A1
Part number	A2D
Model	A2DA1
Type	-
Code Model	adc_bridge
Unknown type	NODES
Node 1	1
Unknown type	NODES
Node 1	2
Code Model	Parameters
in_low	.1
in_high	.9
rise_delay	1p
fall_delay	1p

Label

Parameter	Value
Ref Des	A1
Part number	A2D

Label style: ● Tall ○ Wide ○ Split

- **Repeat these steps for the D2A and inverter symbols. Their parameter values are shown below.**



Comment

Net List Preview

```
A3 [ 3 ] [ 4 ] D2AA3
.MODEL D2AA3 dac_bridge( out_low=0 out_high=1 out_undef=
+ t_rise=100n t_fall=100n input_load=1P)
```

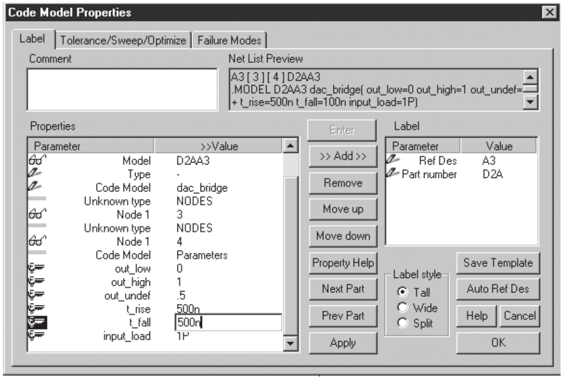
Properties

Parameter	>>Value
Model	D2AA3
Type	-
Code Model	dac_bridge
Unknown type	NODES
Node 1	3
Unknown type	NODES
Node 1	4
Code Model	Parameters
out_low	0
out_high	1
out_undef	.5
t_rise	100n
t_fall	100n
input_load	1P

Label

Parameter	Value
Ref Des	A3
Part number	D2A

Label style: ● Tall ○ Wide ○ Split



Comment

Net List Preview

```
A3 [ 3 ] [ 4 ] D2AA3
.MODEL D2AA3 dac_bridge( out_low=0 out_high=1 out_undef=
+ t_rise=500n t_fall=100n input_load=1P)
```

Properties

Parameter	>>Value
Model	D2AA3
Type	-
Code Model	dac_bridge
Unknown type	NODES
Node 1	3
Unknown type	NODES
Node 1	4
Code Model	Parameters
out_low	0
out_high	1
out_undef	.5
t_rise	500n
t_fall	500n
input_load	1P

Label

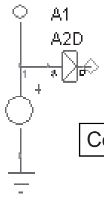
Parameter	Value
Ref Des	A3
Part number	D2A

Label style: ● Tall ○ Wide ○ Split

Node Translation (with Node Bridges)

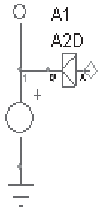
Node translation is required whenever any non-analog component is used in a simulation. Node bridge elements are used to perform the translation between different types of signals (Analog, Digital, Sampled-Data and Arrays) and to obtain output waveforms for any non-analog node. For analog and digital parts, specifically, an A2D bridge translates analog signals to digital states, while a D2A bridge translates digital states to analog voltages. For this simple example, two node translations are required. The first translates the analog input pulse from a voltage source into a digital signal that is used to drive the inverter. The second translates the digital output state of the inverter into an analog voltage so that the signal can be viewed in real-time and displayed in IntuScope.

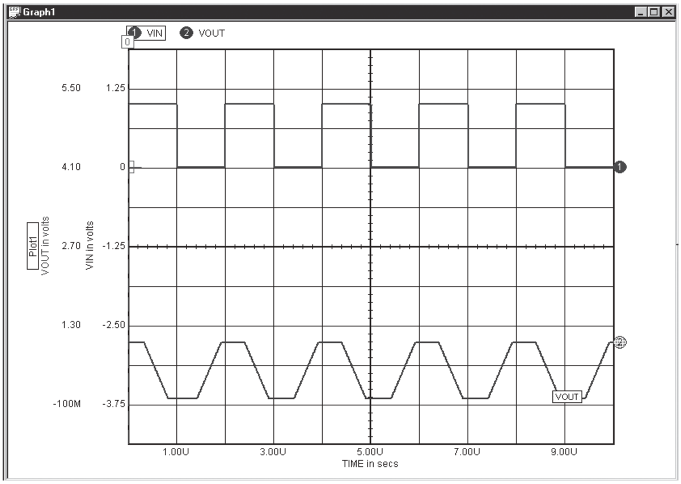
A-to-Ds and D-to-As are not bidirectional. You cannot connect a D-to-A backwards to get an A-to-D. In the figure to the left, notice that the A-to-D has been flipped in an attempt to make a D-to-A.



Correct

Incorrect





The arrow shown inside the symbol marks the correct conversion direction. The A-to-D must have an analog signal connected to its input, and a digital signal connected to its output. The opposite is true for the D-to-A.

Binary Counter

In the previous example, we focused on the important concept of node translation. In this example, we will introduce a more complex digital circuit.

New Topics Covered

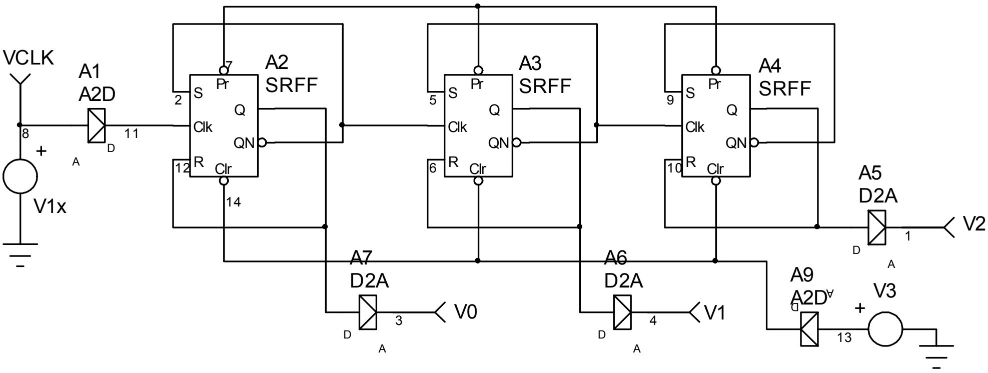
Code Models

- SR flip-flop Code Model
- Analog/Digital ground
- Initializing flip-flops

Specification:
The eight-state binary counter (shown below) will be simulated. The counter will be clocked at a frequency of 20MEG Hz (rise and fall times are set to 1ns). The counter will be initialized to a zero output state. Data will be collected for the input clock and all three outputs for 500ns, in 1ns intervals. All rise and fall delays will be set to 1ns.

Open up the drawing at :\spice8\ Circuits\ Examples\ Bincntr.DWG

```
Binary Counter
.TRAN 1N 500N
*#ALIAS  V2  V(1)
*#ALIAS  V0  V(3)
*#ALIAS  V1  V(4)
*#ALIAS  VCLK  V(8)
V3 6 0 PULSE 5 0 1N
A4 [8] [2] A2D   ; from the Bridges Part Type
A5 [20] [3] D2A ; from the Bridges Part Type
A6 [13] [4] D2A
A8 [6] [15] A2D
A10 [17] [1] D2A
A11 9 PDOWN ; from the Digital Part Type
A13 14 20 2 9 15 20 14 SRFF ; from the Digital Part Type
A14 5 13 14 9 15 13 5 SRFF
A15 10 17 5 9 15 17 10 SRFF
V1 8 0 PULSE 0 1 0 1N 1N 25N 50N
.END
```



Placing parts and Configuring the Code Models

The SR flip-flop and pulldown code models are located in the Browser dialog under the Digital Type, with Generic Subtype as SRFF. The A2D and D2A parts are located under the Bridges Type.

- **Place the parts as shown above. Double-click on each part and configure the model according to the values given below.**

The A2D Model:

in_low =100m
in_high =900m
rise_delay =1p
fall_delay =1n

The D2A Model:

out_low =0
out_high =1
out_undef =500m
t_rise =1n
t_fall =1n

The SRFF Model:

clock_delay =1n
set_delay =1n
reset_delay =1n
output initial state =0
SL = set_load =1p

rise_delay =1n
fall_delay =1n
sr_load =1p
CLKL = clk_load =1p
RSL = reset_load =1p

The Pdown Model:

load = 1p

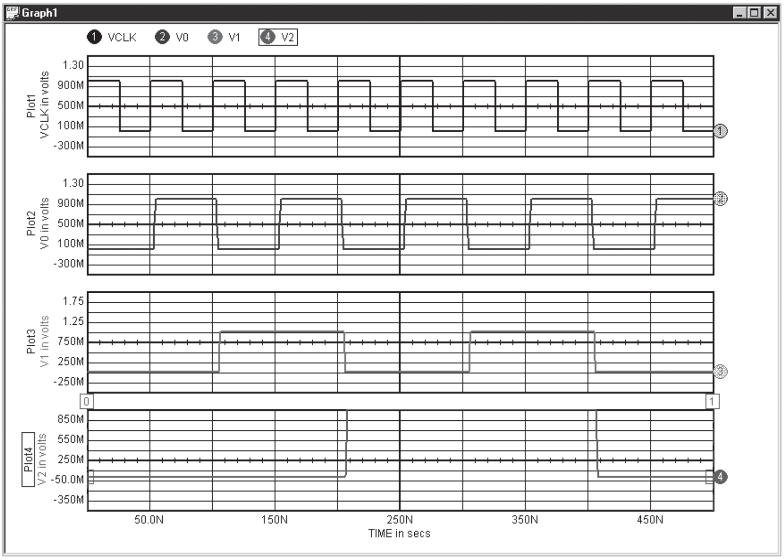
The Pulse stimulus parameters, which define the voltage sources V1 and V3, are located in the netlist.

Ground and The Pulldown Code Model

The pulldown code model is equivalent to a logic 0, or digital ground. It is used here to avoid placing a D2A between the preset and ground (the pullup is used to provide a logic 1).

Initializing the Flip-Flop

The pulldown code model is used to pull the preset of all the flip-flops low. The voltage source, V3, is used to clear all the flip-flops. The PULSE statement (used for V3) places a high at the Clr input during the first clock transition (time=0). Hence, the outputs are low until the next positive edge at time=50n. This is the preferred method for initializing flip-flops. Alternatively, the SR flip-flop code model has an "ic" model parameter that can be used to initialize the outputs of the flip-flop. If a pulldown code model is placed on the Clr and Pr inputs, the "ic" model parameter will be used to initialize the code model (0=low, 1=high, 2=unknown). The resulting waveforms would be the same, but the count would start at the first clock transition located at time=0. The output for this example is shown below.



State Machine

In this example, we will demonstrate the use of the state machine by duplicating the previous binary counter example. The state machine is quite useful since the state diagram, or state table, of a counter is available before the circuit is constructed. In fact, the state machine can be used to replace any large section of clocked combinational digital circuitry.

<u>**New Topics Covered**</u>

Code Models

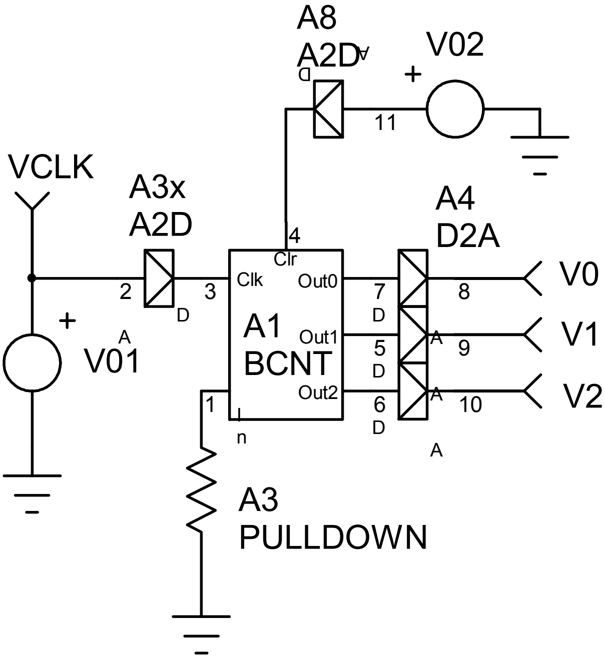
- State Machine Code Model

Open up the drawing at :\spice8\ Circuits\ Examples\ Statemch.DWG

Specification:

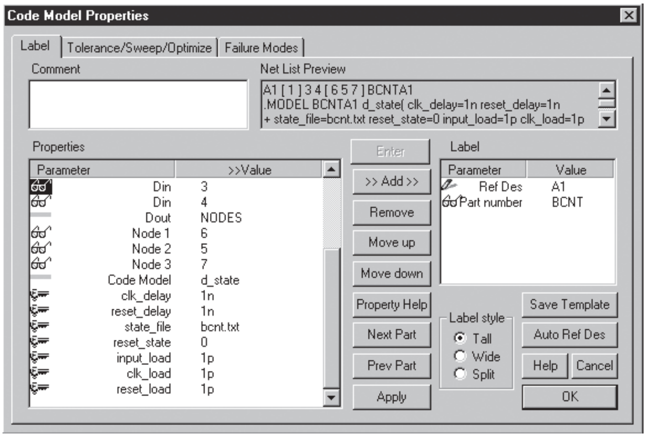
The binary counter from the previous example is shown below, which uses the state machine code model. All input and output conditions will be the same as the previous example.

All devices, except the state machine, are given the same parameters as in the previous example.



Placing Parts

All devices, except the state machine, are given the same parameters as in the previous example. The state machine for this example is located in the Digital Type, Generic Subtype as BCNT in the parts list of the Parts Browser dialog. The Properties dialog for the state machine is given below.



Code Model Properties

Net List Preview

```
A1 [ 1 ] 3 4 [ 6 5 7 ] BCNTA1
.MODEL BCNTA1 d_state( clk_delay=1n reset_delay=1n
+ state_file=bcnt.txt reset_state=0 input_load=1p clk_load=1p
```

Properties

Parameter	>>Value
Din	3
Din	4
Dout	NODES
Node 1	6
Node 2	5
Node 3	7
Code Model	d_state
clk_delay	1n
reset_delay	1n
state_file	bcnt.txt
reset_state	0
input_load	1p
clk_load	1p
reset_load	1p

Label

Parameter	Value
Ref Des	A1
Part number	BCNT

Label style: ● Tall ○ Wide ○ Split

State Machine Symbols and Models

Note: The state machine model and state transition file must be reconfigured for each different state machine.

The state machine code model can have a variable number of inputs and outputs, and an unlimited number of states. Typically, your state machine will start as a state diagram. From this diagram, the state transition table for the state machine code model will be created. The format for this file is given in the IsSpice4 User's Guide. Below is a state transition file (BCNT.TXT) for this example. The state machine's .model statement contains a parameter called state_file. Its value is a filename that contains the state transition information, in this case, BCNT.TXT. You can only change this data by editing the BCNT file.

For other state machine configurations you will have to create a new library entry like the BCNT one in the CM2.LIB file, a new symbol, and a new state transition file like BCNT.TXT

Note: You must copy the BCNT.TXT file to the directory where the circuit file is located. The state machine looks for the state transition file in the project directory.

```
****
0 0s 0s 0s 0 -> 1
       1 -> 1

1 0s 0s 1z 0 -> 2
       1 -> 2

2 0s 1z 0s 0 -> 3
       1 -> 3

3 0s 1z 1z 0 -> 4
       1 -> 4

4 1z 0s 0s 0 -> 5
       1 -> 5

5 1z 0s 1z 0 -> 6
       1 -> 6

6 1z 1z 0s 0 -> 7
       1 -> 7

7 1z 1z 1z 0 -> 0
       1 -> 0
```

State Transition File for the state machine used in the Binary Counter

The first column defines the state. The next three columns define the output for the state. The fifth column, which precedes the -> symbol, defines the expected input. The final column, after the ->, defines the state that is expected if the input is present during the positive edge of the input clock signal. You can see from the file that the output is a binary counter that changes state on either a high or low input.

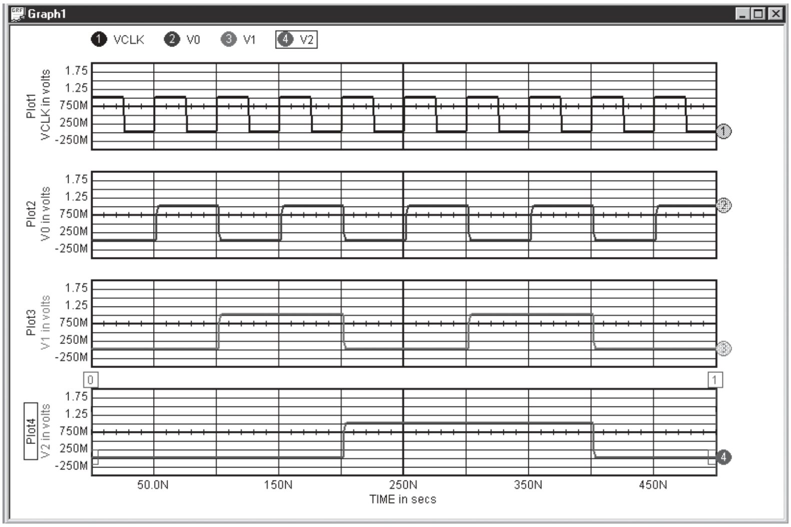
Once the state transition file is created, a symbol can be constructed to match the desired number of inputs and outputs.

In this case, the part has the pin order:

```
A47 [1] 3   4    [11 6 5] St3
*   In  Clk Clr Out2 1 0
```

The process of adding a new model library entry is covered in the on-line help.

The output for this example is shown below.



Boolean Logic Functions

Apart from the digital simulation capabilities of IsSpice4, an extremely useful feature is the ability to enter Boolean logic expressions. As with the state machine code model, this function can greatly speed the implementation of digital circuitry. Boolean functions are accessed through the B (arbitrary dependent source) element. Boolean logic is faster and more efficient than event-driven elements for most simple logic functions, which only require a few gates.

This example is split into two parts. First, we will introduce the Boolean expression simulation using a simple NAND gate and a simple NOR gate. Second, we will implement a larger Boolean expression.

New Topics Covered

Sources

- B...Nonlinear Dependent Source
- Boolean Expressions
- .OPTIONS...LTHRESH, LONE, LZERO

Specifications:

We will use the B element to construct a typical AND-NOR stage. The gates will be represented by a single Boolean expression.

Open up the drawing at :\spice8\ Circuits\ Examples\ logic.DWG

NAND and NOR Gates

The syntax for the B source is listed in the IsSpice4 User's Guide. The &, |, and ~ operators will be used to construct a NAND gate and a NOR gate.

To implement an AND function, first AND two voltages:

V(1) & V(2)

To create the NAND, NOT this combination:

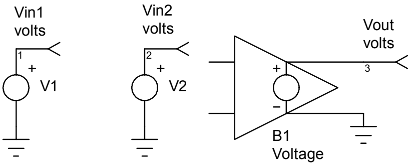
~(V(1) & V(2))

To implement a NOR function, first OR the voltages,

V(1) | V(2)

Then, to create the NOR, NOT this combination:

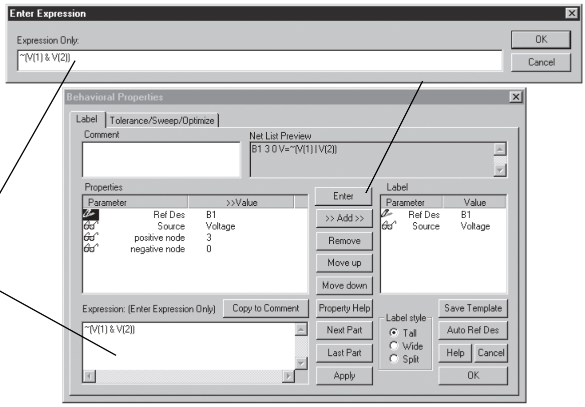
~(V(1) | V(2))



To enter an expression:

The phrase “~(V(1)&V(2))“ is entered into the B element Properties dialog in the Expression window.

- **Double Click the arbitrary dependent voltage source B to open up the Behavioral Properties dialog.**
- **Enter an expression in the Expression field.**
- **Or, click on the Enter button in the dialog and type.**



Expression Only: ~(V(1) & V(2))

Comment:

Net List Preview: B1 3 0 V=~(V(1) | V(2))

Properties

Parameter	>>Value
Ref Des	B1
Source	Voltage
positive node	3
negative node	0

Label

Parameter	Value
Ref Des	B1
Source	Voltage

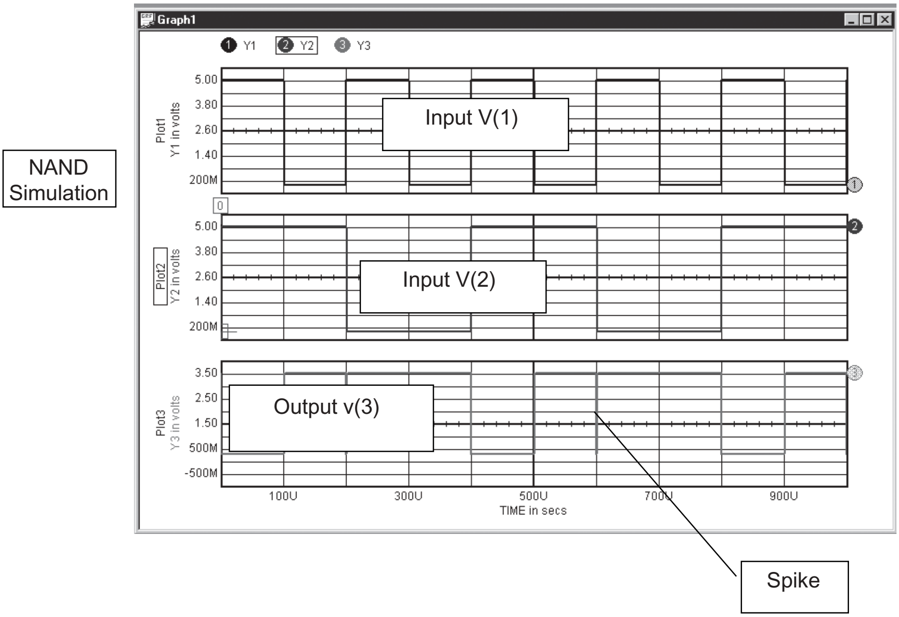
Expression: (Enter Expression Only) ~(V(1) & V(2))

Label style: ● Tall ○ Wide ○ Split

Enter the expression in either field here.

```
NAND Gate
.TRAN 1U 2.5M
BNAND 3 0 V=~(V(1)&V(2))
V1 1 0 PULSE 0 5 0 1U 1U 100U 200U
V2 2 0 PULSE 0 5 0 1U 1U 200U 400U
.END
```

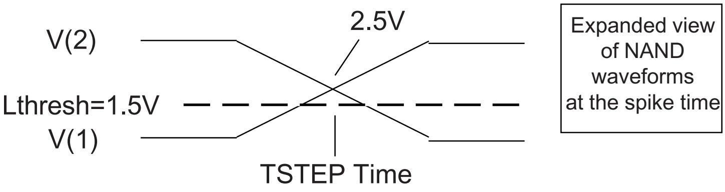
- **Construct and simulate the NAND circuit. Notice how the voltage sources and the B element are not wired together. When the simulation is complete, examine the output.**



Boolean functions are continuously evaluated, not event-driven.

Note the spike in the NAND output (waveform 3). This occurs because the TRISE and TFALL values of the inputs are equal to the TSTEP value. This causes the input signals to cross at 2.5V. At the TSTEP time, both signals are above the Lthresh level (1.5V), therefore generating the spike in the output (waveform 3). Such conditions are possible because Boolean functions are continuously evaluated.

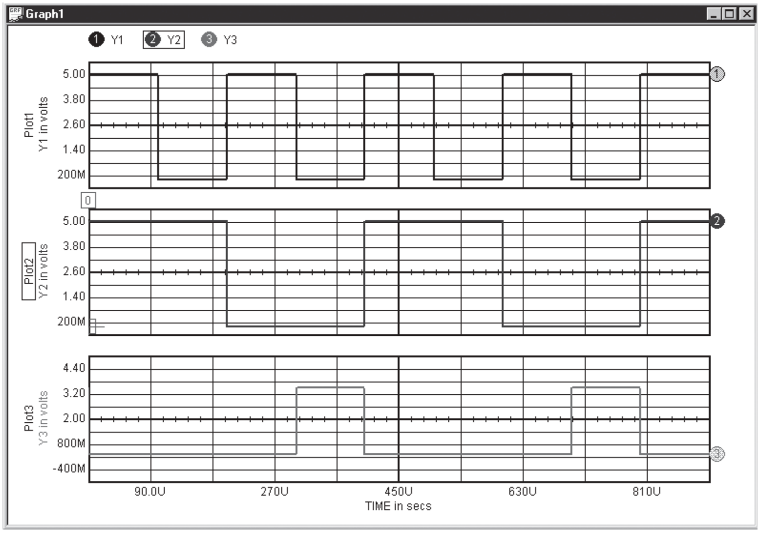
Default .OPTIONS parameters for Boolean expressions are:
LTHRESH=1.5V
LONE=3.5V
LZERO=.5V



```
NOR Gate
.TRAN 1U 2.5M
BNOR 3 0 V=~(V(1)|V(2))
C1 3 0 0.87NF
V1 1 0 PULSE 0 5 0 1U 1U 100U 200U
V2 2 0 PULSE 0 5 0 1U 1U 200U 400U
.END
```

- **Construct and simulate the NOR circuit. You can do this by simply modifying the NAND circuit.**

The B element has no physical nodal input connections. Its only input comes from the variable listed in the Boolean equation that was entered into the Properties Expression field.

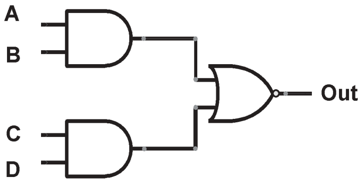


The .OPTIONS parameters, LTHRESH, LONE and LZERO control the turn-on/turn-off voltage threshold, the high level output and low level output, respectively.

- **Set the low level output to 0V, the high level output to 5V and the threshold to 0.1V by typing the following .OPTIONS line in the IsSpice4 Simulation Setup dialog. Type this in the User Statements area.**

```
.OPTIONS LTHRESH=.1  LONE=5 LZERO=0
```

More Complicated Boolean Expression
To demonstrate the power of the Boolean expression, we can implement the stage shown below with one expression.



V(Out) = ~((V(A)&V(B)) | (V(C)&V(D)))

A subcircuit could be made by implementing this gate combination. It may be called as many times as needed without being retyped. The subcircuit would be:

```
.SUBCKT ANDNOR 1 2 3 4 5
b1 5 0 v=~( (v(1)&v(2)) | (v(3)&v(4)) )
.ENDS
```

. . . .

- **Enter and simulate the following netlist. Examine the output and verify the operation of the subcircuit.**

```
Boolean Logic Expressions
.TRAN 1U 500U
VA 1 0 PULSE 0 5 0 1U 1U 10U 20U
VB 2 0 PULSE 0 5 0 1U 1U 20U 40U
VC 3 0 PULSE 0 5 0 1U 1U 40U 80U
VD 4 0 PULSE 0 5 0 1U 1U 80U 160U
X1 1 2 3 4 5 ANDNOR
.SUBCKT ANDNOR 1 2 3 4 5
B1 5 0 V=~((V(1)&V(2))|(V(3)&V(4)))
.ENDS
.END
```

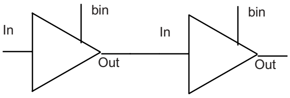
If-Then-Else - An A to D Converter

In this example, we will use the if-then-else capability to simulate an analog-to-digital converter.

Specification:

Below is a circuit file that implements an 8-bit A-to-D converter. The input stimulus is a ramp voltage created by the PULSE statement given by vin. The ramp will cause the A-to-D converter to cycle through all of its states. The A-to-D converter consists of one stage per bit. Each stage is identical and is represented by the subcircuit "adc," as shown in the netlist. In this subcircuit, B1 is used to limit the "bin" value to 1 or 0, thus producing the digital output. B2 is used to conditionally subtract the reference (1 volt) from the input. The resultant error signal is then multiplied by two, so that it can be applied to the input of the following stage of the A-to-D converter.

- **Construct and simulate the circuit shown below. You will need to construct an ADC symbol.**



Example adc symbols

```
An A to D converter
.tran 100n 1u 0 1n
vin 1 0 pulse 0 2 0 1u 0 1u
x1 1 2 10 adc
x2 2 3 11 adc
x3 3 4 12 adc
x4 4 5 13 adc
x5 5 6 14 adc
x6 6 7 15 adc
x7 7 8 16 adc
x8 8 9 17 adc
.print tran v(10) v(11) v(12) v(13)
.print tran  v(14) v(15) v(16) v(17)
.print tran v(1) v(2) v(3) v(4)
.print tran  v(5) v(6) v(7) v(8)
.subckt adc in out bin
b1 bin 0 v= (v(in) > 1) ? 1 : 0
b2 out 0 v= 2*(v(in) - v(bin))
.ends
.end
```

MIDI Synthesizer

In this example, we will simulate a MIDI synthesizer in order to examine the mixed-domain simulation capability of ICAPS. The MIDI circuit uses three types of data: Digital, Real and Analog. These types are signified by the letters D, R and A, respectively on the schematic. Translational node bridges are used to convert between signal types.

New Topics Covered

Sources

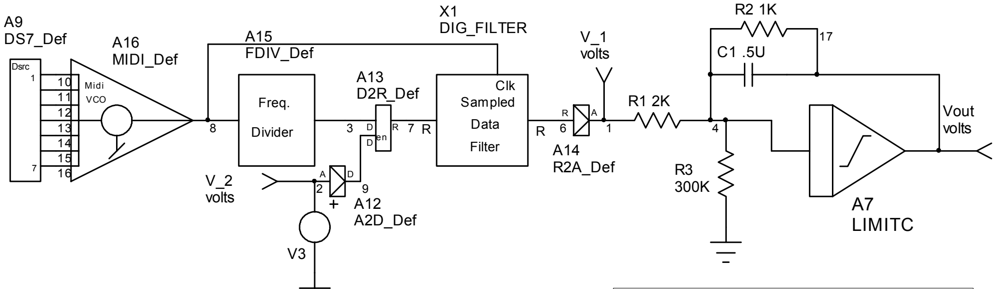
- Digital Source

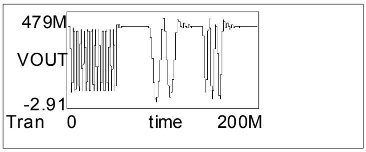
Code Models

- Numerically Controlled Oscillator (NCO)
- Digital Frequency Divider
- Sample Data Filter using Real Gain and Real Delay
- Limiter

Specifications:

Given the circuit shown below, we will perform a transient simulation and collect data every 0.1ms for 200ms at the output (node 13) and for the digital clock signal (node 17). Reltol, which is entered in the Simulator Options menu of the IsSpice4 Simulation Setup dialog, should be set to .0001.





```
MIDI
.OPTIONS RELTOL=.0001 ; Simulation Options... dialog
.MODEL DS7_001 d_source(input_file=source7.txt input_load=1.0p)
.MODEL FDIV_001 d_fdiv(div_factor=8 high_cycles=4
+ i_count=0 rise_delay=2.0N fall_delay=20.0N freq_in_load=1.0p)
.MODEL A2D_001 adc_bridge(in_low=100M in_high=900.0M
+ rise_delay=1.0N fall_delay=1.0N)
.MODEL D2R_001 d_to_real(zero=0.0 one=1.0 delay=1.0N)
.MODEL R2A_001 real_to_v(gain=1.0 transition_time=1.0N)
.MODEL MIDI_001 nco(delay=1.0N mult_factor=16.0)
.MODEL LIM_001 limit(in_offset=0.0 gain=300.0k out_lower_limit=-12.0
+ out_upper_limit=12.0 limit=12.0 limit_range=1.0u fraction=FALSE)
.MODEL D2A_001 dac_bridge(out_low=0.0 out_high=1.0 out_undef=500m
+ t_rise=1.0N t_fall=1.0N)
.TRAN .0001 .2
*#ALIAS  VOUT  V(13)
.PRINT TRAN  V(13)
A2 [1 2 3 4 5 6 7] 11Midi_001
A4 16 15 10 D2R_001
X1 10 11 8 DIG_FILTER
R1 14 12 2K
R2 12 0 300K
R3 12 13 1K
C1 12 13 .5U
A6 8 14 R2A_001
A9 11 16 FDIV_001
V1 9 0 PWL 0 1 .049 1 .05 0 .074 0 .075 1 .109 1 .11 0 .139 0
+ .14 1 .159 1 .16 0
A11[ 9] [15] A2D_001
A12 [11] [17] D2A_001
A13 12 13 LIM_001
A1 [1 2 3 4 5 6 7] DS7_001
.END
```

Placing Parts

The digital source used for this example can be found under the Digital type, Generic Subtype as Dsrc7. The MIDI VCO and frequency divider models are also found under the Digital type, Generic Subtype as NCO and Fdiv, respectively. The sampled data filter and the limiter can be found under the Behavioral Type, Function Block subtype as Dig_Filter, and Limitc, respectively. All node bridges (D-to-A, A-to-D, D-to-R, and R-to-A) are found under Bridges Type. The parameters used to configure the code models can be found in the netlist given above.

Digital Source

The MIDI note data is created using a digital stimulus source, Dsrc7. MIDI notes are composed of 7 bits, and are numbered between zero and 127 (Bit 1 = MSB). Three C notes are simulated, each an octave apart, corresponding to the note numbers 60, 36 and 48. These are given by the following digital source input file, MIDI.txt.

```
*T      1   2   3   4   5   6   7
*i
*m
*e

0       0s  1s  1s  1s  1s  0s  0s
.06 0s  1s  0s  0s  1s  0s  0s
.12 0s  1s  1s  0s  0s  0s 0s
```

MIDI.TXT is an external text file that contains a description of the digital source.

The digital values, 0s and 1s, correspond to the acceptable state values as defined in the IsSpice4 User's Guide. Please refer to this guide for a more complete description of the defined states and the digital source syntax.

Numerically Controlled Oscillator (NCO)

The MIDI VCO (NCO model) is a digitally controlled oscillator that produces a square wave with a frequency based on the MIDI note number presented at the input. The digital input source feeds this model to create the desired digital frequency. The frequency multiplier model parameter (mult_factor) is 16.

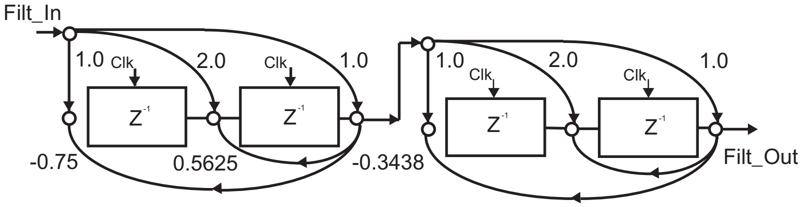
***Note:** You must copy the MIDI.TXT file to the directory where the MIDI design is located in order to make it accessible. The digital source looks for the source input file in the project directory.*

Frequency Divider

The frequency divider is a generic divide-by-N frequency divider. The model parameter (div_factor) defines the value of N. In this example, the output from the NCO is divided by 8 and fed to the sampled data filter.

Sampled Data Filter

The sampled data filter is a subcircuit composed of real gain and real delay (Z^{-1}) code models, as shown below. These models use real data, and therefore require node bridges on the input and output to convert from digital to real and from real to analog.



Note: In order to view the digital output, a D-to-A bridge was required to convert the digital clock signal to its analog equivalent.

The input is translated from a digital state to a real value using the D-to-R node bridge. The enable on the D-to-R bridge is driven by a simple PWL statement, which is converted to a digital signal with an A-to-D bridge. The output is converted to an analog signal through a real-to-analog bridge. The subcircuit for the sampled-data filter can be viewed in the IsSpice4 .OUT output file.

Limiter

The limiter code model is a simple limiting gain stage. In this example , the limiter is used to simulate the lowpass filter output stage. The final limiter output is shown below.

A detailed description of the MIDI synthesizer is located in the November 1994 *Intusoft Newsletter.*

Hot Key Table

IntuScope5 Hot Key Table

Function	Letter
add	+
Add Waveform	Ctrl+F
average	a
Bode with margins	b
Clear	Del
Close Graph	Ctrl+W
Context Help	Shift+F1
Copy	Ctrl+C
Cut	Ctrl+X
Data points	#
Differentiate	d
divide	/
Do Command Script	Ctrl+R
efficiency	e
enlarge	Shift+Up
Exit/Quit	Ctrl+Q
expandx	x
expandy	Y
FFT time to mag	f
frequency	w
Get X at Cursor 0	F7
Get X at Cursor 1	F8
Get X Delta	F10
Get Y at Cursor 0	F5
Get Y at Cursor 1	F6
Get Y Delta	F9
Integrate	i
matchscales	y
maximum	>

IntuScope5 Hot Key Table

Function	Letter
minimum	<
Move Cursor 0 <- to a(y)	F11
Move Cursor 0 -> to a(y)	F12
Move Cursor 0 to X	F3
Move Cursor 1 <- to a(y)	<-
Move Cursor 1 -> to a(y)	->
Move Cursor 1 to X	F4
multiply	*
New Graph	Ctrl+N
nudgedown	Down
nudgeup	Up
Nyquist	n
Open Graph...	Ctrl+O
Paste	Ctrl+V
peak to peak	p
Print...	Ctrl+P
Redo	Shift+Ctrl+Z
Reset Cursors	HOME
RMS	r
Save Graph	Ctrl+S
Scaling	Ctrl+L
Scaling Dialog	Ctrl+L
Schematic	Ctrl+D
Select All	Ctrl+A
Show Cursor Bar	Ctrl+K
Show Script Windows	Ctrl+J
shrink	Shift+Down
Simulate	Ctrl+G
Std Deviation	$

IntuScope5 Hot Key Table

Function	Letter
subtract	-
superfamily	s
Text Edit>Input	Ctrl+I
Tile All Traces(y only)	Ctrl+T
Undo	Ctrl+Z
Update Using Selected Mode	Ctrl+U
X at Maximum	M
X at Minimum	m

IsSpice4Hot Key Table

Function	Letter
Copy	Ctrl+C
Cut	Ctrl+X
Exit/Quit	Ctrl+Q
Print	Ctrl+P
Undo	Ctrl+Z

SpiceNet Hot Key Table

Function	Letter
Amps —> Amps control source	F
Amps —> Volts control source	H
Arbitrary Iout control source	2B
Arbitrary Vout control source	B
Bus	2W
Capacitor	C
Continuation	Z
Current source	I
Current test point	3Y
Cut	Ctrl+X
DC voltage source	V
Differential voltage test point	2Y
Diode	D
Ground	O
Ideal T line	T
Inductor	L
Mutual Inductor	K
N-BJT	Q
N-JFET	J
N-MFET	4Z
N-MOS	M
P-BJT	2Q
P-JFET	2J
P-MFET	5Z
P-MOS	2M
Power test point	4Y
RCRD Lossy T Line	U
Resistor	R
RLCG Lossy T Line	4O
Run a simulation	Ctrl+G
Switch I control	4W
Switch V control	S
Voltage test point	Y
Volts —> Amps control source	G
Volts —> Volts control source	E
Wire	W

I Index