

PrimeWave Design Environment™

User Guide

Version S-2021.09, September 2021

Copyright and Proprietary Information Notice

© 2021 Synopsys, Inc. This Synopsys software and all associated documentation are proprietary to Synopsys, Inc. and may only be used pursuant to the terms and conditions of a written license agreement with Synopsys, Inc. All other use, reproduction, modification, or distribution of the Synopsys software or the associated documentation is strictly prohibited.

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at <https://www.synopsys.com/company/legal/trademarks-brands.html>.

All other product or company names may be trademarks of their respective owners.

Free and Open-Source Licensing Notices

If applicable, Free and Open-Source Software (FOSS) licensing notices are available in the product installation.

Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

www.synopsys.com

Contents

New in This Release	18
Related Products, Publications, and Trademarks	18
Conventions	19
Customer Support	20
Statement on Inclusivity and Diversity	20
1. Setting Up and Invoking the PrimeWave Design Environment	21
Invoking the PrimeWave Design Environment from the Custom Compiler Tool	21
Specifying a File Text Editor	24
Opening a Design	25
2. Setting Up Testbenches	27
Working with Multiple Testbenches	28
Creating Test Suites	29
Specifying Test Suite Options	31
Adding Testbenches	32
Adding Testbenches from States	33
Deleting Testbenches	36
Ejecting Testbenches	36
Cloning Testbenches	37
Copying Testbench Categories	37
Saving Test Suites	37
Saving Test Suite Results	39
Saving Multiple Testbench History	39
Loading Test Suite Results	41
Comparing Testbench Settings	41
Choosing a Design	44
Choosing a Simulator	45
Setting Simulator Options	49
Choosing a PrimeSim Simulation Engine	50
PrimeSim HSPICE Simulation Options	51

Contents

PrimeSim SPICE Simulation Options	53
PrimeSim Pro Simulation Options	56
PrimeSim XA Simulation Options	59
Setting Environment Options	61
Specifying PrimeSim Environment Options	62
Specifying FineSim Environment Options	64
Specifying FineSim VCS Environment Options	67
Specifying VCS PrimeSim AMS Environment Options	68
Specifying Model Files	70
Setting Up Model Files	71
Editing Model Files	74
Reordering Model Files	75
Removing Model Files	75
Copying and Pasting Model Files	75
Specifying Parasitic Back-Annotation Flow	75
Specifying Parasitic Back-Annotation Flow for PrimeSim HSPICE	75
Specifying Parasitic Back-Annotation Flow for FineSim	78
Sequential Testbenches	82
getVal() Syntax	88
Saving and Plotting Terminal Node Voltages	88
Saving Terminal Node Voltages	89
Plotting Terminal Node Voltages	89
Specifying Include Files	89
Editing Include Files	92
Creating Include Files from the Edit Menu	92
Removing Include Files	93
Specifying External Images	93
Specifying Digital Timing	94
Specifying Convergence Aids	95
Adding Stimulus	96
Adding Instances	96
Editing Instances	97
Copying Instances	97
Deleting Instances	97
Creating, Renaming, and Deleting Stimulus Sets	97
Previewing Netlists	98

Contents

Specifying Temperature	98
Working with History Points	98
Saving History Points	99
Loading History Points	100
Viewing Results of a History Point	100
Launching a Terminal from a History Point	100
Renaming History Points	101
Locking History Points	101
Unlocking History Points	101
Deleting History Points	102
Deleting Results of a History Point	102
Saving States	102
Loading States	104
Saving Tcl Scripts	107
3. Working With Design Variables	109
Adding and Editing Design Variables	110
Editing Design Variables in a Text Editor	112
Adding Cell View Variables	112
Adding String Variables	114
Deleting Design Variables	115
Globalizing Local Design Variables	116
Localizing Global Design Variables	116
Pushing Design Variables to All Testbenches	117
Adding Design Variable Sets	118
Copying Design Variable Sets	118
Choosing Design Variable Sets	119
Deleting Design Variable Sets	119
Copying Design Variables from a Design	119
Copying Design Variables to a Design	119
Searching for Design Variables	120
Parameterizing Designs	120
Creating Design Variables from Existing Design Parameters	121
Netlisting Parameterization Setup	125

Contents

Parameterizing Files	125
Parameterizing Include Files	126
Parameterizing SPF/DSPF Files	128
Parameterizing Stimulus Files	130
VAR() Expressions	131
<hr/>	
4. Working with Analyses	133
Setting Up Analyses	133
Enabling Save and Restore Times	133
Enabling Save Times	134
Enabling Restore Times	134
Enabling and Disabling Analyses	135
Editing Analysis Values	135
Deleting Analyses	136
<hr/>	
5. Setting Up PrimeSim HSPICE Analyses	137
PrimeSim HSPICE Transient Analysis	138
PrimeSim HSPICE Operating Point Analysis	140
PrimeSim HSPICE DC Analysis	141
PrimeSim HSPICE AC Analysis	142
PrimeSim HSPICE Noise Analysis	143
PrimeSim HSPICE FFT Analysis	145
PrimeSim HSPICE Linear Network Parameter Analysis	147
PrimeSim HSPICE AC Match Analysis	148
PrimeSim HSPICE DC Match Analysis	150
PrimeSim HSPICE Transient Noise Analysis	151
PrimeSim HSPICE Statistical Eye Analysis	153
PrimeSim HSPICE Loop Stability Analysis (LSTB)	158
PrimeSim HSPICE RF Harmonic Balance Analysis	159
PrimeSim HSPICE RF Harmonic Balance Oscillator Analysis	161
PrimeSim HSPICE RF Harmonic Balance AC Analysis	162
PrimeSim HSPICE RF Harmonic Balance Transfer Function Analysis	163
PrimeSim HSPICE RF Harmonic Balance Noise Analysis	165

Contents

PrimeSim HSPICE RF Shooting Newton Analysis	167
PrimeSim HSPICE RF Shooting Newton Oscillator Analysis	168
PrimeSim HSPICE RF Shooting Newton AC Analysis	169
PrimeSim HSPICE RF Shooting Newton Transfer Function Analysis	170
PrimeSim HSPICE RF Shooting Newton Noise Analysis	172
PrimeSim HSPICE RF Shooting Newton with Fourier Transform Analysis	174
PrimeSim HSPICE RF Phase Noise Analysis	176
PrimeSim HSPICE RF Periodic Time-Dependent Noise Analysis	178
PrimeSim HSPICE RF Envelope Analysis	180
PrimeSim HSPICE RF Envelope Oscillator Analysis	181
PrimeSim HSPICE RF Envelope Fast Fourier Transform Analysis	182
PrimeSim HSPICE Bias Check Analyses	184
<hr/>	
6. Setting Up PrimeSim SPICE Analyses	187
PrimeSim SPICE Transient Analysis	188
PrimeSim SPICE Operating Point Analysis	190
PrimeSim SPICE AC Analysis	192
PrimeSim SPICE DC Analysis	193
PrimeSim SPICE Noise Analysis	194
PrimeSim SPICE Linear Network Parameter Analysis	196
PrimeSim SPICE Loop Stability Analysis (LSTB)	197
PrimeSim SPICE AC Match Analysis	199
PrimeSim SPICE DC Match Analysis	200
PrimeSim SPICE PZ Analysis	202
PrimeSim SPICE XF Analysis	203
PrimeSim SPICE Sensitivity Analysis	205
PrimeSim SPICE Include Analysis	206
PrimeSim SPICE RF Shooting Newton Analysis	207
PrimeSim SPICE RF Shooting Newton AC Analysis	209
PrimeSim SPICE RF Shooting Newton Transfer Function Analysis	212
PrimeSim SPICE RF Shooting Newton Noise Analysis	214
PrimeSim SPICE RF Shooting Newton Linear Analysis	216

Contents

PrimeSim SPICE RF Shooting Newton Stability Analysis	218
PrimeSim SPICE RF Harmonic Balance Analysis	220
PrimeSim SPICE RF Harmonic Balance AC Analysis	222
PrimeSim SPICE RF Harmonic Balance XF Analysis	225
PrimeSim SPICE RF Harmonic Balance Noise Analysis	227
PrimeSim SPICE RF Harmonic Balance Linear Analysis	229
PrimeSim SPICE RF Harmonic Balance Stability Analysis	231
PrimeSim SPICE RF Envelope Following Analysis	233
PrimeSim SPICE Circuit Checks	234
<hr/>	
7. Setting Up PrimeSim Pro Analyses	237
PrimeSim Pro Transient Analysis	238
PrimeSim Pro DC Analysis	240
PrimeSim Pro AC Analysis	241
PrimeSim Pro Operating Point Analysis	242
PrimeSim Pro Circuit Checks	242
<hr/>	
8. Setting Up PrimeSim XA Analyses	245
PrimeSim XA Analyses	246
PrimeSim XA Transient Analysis	246
PrimeSim XA Operating Point Analysis	246
PrimeSim XA (Eldo) Analyses	247
PrimeSim XA (Eldo) Transient Analysis	248
PrimeSim XA (Eldo) Operating Point Analysis	248
<hr/>	
9. Setting Up VCS PrimeSim AMS Analyses	250
VCS PrimeSim AMS Transient Analysis	250
VCS PrimeSim AMS AC Analysis	251
VCS PrimeSim AMS Operating Point Analysis	252
<hr/>	
10. Setting Up FineSim Analyses	253
FineSim Analyses	253
FineSim Transient Analysis	254

Contents

FineSim Operating Point Analysis	255
FineSim DC Analysis	256
FineSim AC Analysis	257
FineSim Noise Analysis	258
FineSim FFT Analysis	259
FineSim Loop Stability Analysis	261
FineSim Circuit Check Analysis	261
FineSim VCS Analyses	264
FineSim VCS Transient Analysis	264
FineSim VCS Operating Point Analysis	265
Setting Performance Options	265
Specifying FineSim Analog Options	269
11. Setting Up FineSim VCS AMS Analyses	271
FineSim VCS Transient Analysis	271
FineSim VCS Operating Point Analysis	272
12. Specifying Outputs and Output Options	273
Setting Up Outputs	273
Specifying Outputs	274
Setting Specification Goals	277
Setting Up Scatter Charts	279
Setting Up Histograms	280
Setting Up Q-Q Plots	281
Setting Up Parametric Reduction	282
Adding External Images to Output	283
Specifying Output Logic Radix	283
Specifying Output Plot Type	284
Adding Outputs from Designs	284
Adding Outputs from Designs without Automatically Plotting Waveforms	285
Saving, Loading, and Editing Outputs as Text	285
Reordering Outputs	285
Reordering Output Columns	286
Controlling Output Column Visibility	286
Editing Outputs in a Text Editor	286

Contents

Editing Output Image Titles and Captions	286
Deleting Outputs	287
Showing Outputs in Designs	288
Adding Output Sets	288
Copying Output Sets	288
Choosing Output Sets	289
Deleting Output Sets	289
Specifying ACE Scripts	289
PrimeWave Design Environment Provided ACE Utilities	291
Sample ACE Script	292
Specifying MATLAB Scripts	293
Sample MATLAB Script	296
Debugging MATLAB Script Issues	297
Specifying Output Options	299
PrimeSim HSPICE Output Options	299
FineSim Output Options	301
PrimeSim XA Output Options	301
FineSim VCS Output Options	302
VCS PrimeSim AMS Output Options	304
13. Creating Netlists and Running Simulations	307
Creating Netlists	307
Netlisting a Design	307
Netlisting and Running Simulations	308
Viewing and Editing Netlists	309
Starting and Stopping Simulations	309
Running Simulations in Append Mode	310
Setting VCS PrimeSim AMS Run Mode	314
Batch Run Mode	314
GUI Run Mode	314
Setting Mixed-Signal Simulator Options	316
Specifying System-Verilog Flow Mixed-Signal Simulator Options	316
Specifying VCS Options	318
Specifying Mixed Signal Controls	318
Specifying A/D Options	318

Contents

Specifying E/R Options	320
Specifying E/N Options	322
Specifying Resistance Options	323
Specifying User Commands	323
Specifying Verilog-AMS Flow Simulator Options	323
Specifying VCS Options	325
Specifying Mixed Signal Controls	325
Specifying Connect Modules	325
Specifying Connect Rules	326
Specifying User Commands	326
Setting Block-Level Options	326
Setting Block-Level VCS PrimeSim AMS Options	326
Setting Block-Level FineSim VCS Options	328
Setting Block-Level FineSim Options	329
Example	330
Displaying the Simulation Log	331
Saving Simulation History	331
Displaying Simulation Jobs	332
Configuring Concurrent Job Limit and Queuing Engine Settings	333
Using Remote Host Without Load-Queueing Systems	334
Selecting Columns for Display	334
Grouping Jobs	336
Operating on Selected Jobs	336
Filtering Job Groups	336
14. Analyzing Simulation Data Using the Results Analyzer and Results Compare	338
Using the Results Analyzer	338
Opening the Results Analyzer	339
Choosing Simulation Result Data Sources	339
Using the Plotting Assistant	340
Plotting Signals With the Plotting Assistant	340
Creating DC Operating Point Expressions for an Instance Included in a DSF Netlist File	343
Accessing the Calculator	348
Creating .MEASURE Statements	348
Analyzing Statistical Data	349
Creating Histograms	349
Creating Scatter Plots	350

Contents

Using Parametric Reduction	351
Using Results Compare	354
Viewing Comparison Summary Results	355
Comparing Measurement Results	357
Comparing Iteration Results	358
Measurement and Iteration View Coloring Rules	359
<hr/>	
15. Using the ResultsView	361
Opening the ResultsView	362
ResultsView User Interface	363
ResultsView Menus	366
ResultsView Toolbar Buttons	368
Results Tables	369
Viewing Summary and Testbench Yield Results	370
Viewing Measurement and Statistical Results	372
Viewing Consolidated and Pass/Fail Results	376
Table Menu Options	378
Outputs Tree View	380
Viewing Incremental Results	381
Plotting Results From the ResultsView	381
Plot All	383
Plot	383
Plot Across	384
Plot Histogram	384
Displaying Netlists	385
Displaying Output Logs	385
Displaying Images	385
Launching the Terminal From a Result	385
Rerunning Iterations	386
Opening New Sessions	386
Exporting ResultsView Results	386
Setting Export Options	387
Exporting Results to a CSV File	388
Exporting Results to an HTML Datasheet	389
Selecting Items for Column Headers	390

Contents

Filtering Column Results	392
Showing and Hiding Columns in the Results Table	392
Reordering Table Results	393
Highlighting Whole Rows and Columns	394
Adding Descriptions	394
<hr/>	
16. Using Charts to Visualize Simulation Results	396
Chart Options	396
Chart Toolbar Buttons	397
Chart Menu Options	397
Scatter Charts	398
Multiple-Axis Charts	401
Histogram Charts	403
Q-Q Charts	407
Zooming In and Out	409
Editing Chart Properties	410
Enabling Data Point Information Balloons	410
Specifying Parametric Reduction From Charts	411
<hr/>	
17. Plotting, Printing, and Reviewing Simulation Results	413
Plotting Outputs	414
Creating Custom Plot Modes	415
Plotting Output Sets	416
Plotting Signals Interactively	416
Printing and Annotating Node Voltages and Operating Points	417
Printing and Annotating Operating Points	417
Tcl Preference to Control Operating Point Rendering	419
Using OP Points Results Tables	419
PrimeSim HSPICE Printing and Annotating	430
Printing PrimeSim HSPICE Voltages and Operating Points	431
Printing PrimeSim HSPICE AC and DC Mismatches	431
Printing the PrimeSim HSPICE Noise Summary	432
Annotating PrimeSim HSPICE Voltages and Operating Points	435
FineSim Printing and Annotating	436

Contents

Using the Simulation Check Viewer	438
Cluster Analysis Using the Simulation Check Viewer	441
Working with Monte Carlo Data Mining Results	443
Viewing Data Mining Results	444
Probing Data Mining Results	447
Viewing Data Mining Data Files	447
Debugging Using Extreme Case Results	448
Saving Simulation Results	451
Loading Simulation Results	451
Specifying Results Options	451
Specifying General Result Options	452
Specifying Report Options	454
Specifying Image Saving Options	455
Specifying Simulation Result Options	456
Using the Timestamps Dashboard	457
Opening the Timestamps Dashboard	458
Using Full Scan	460
Using Quick Scan	461
Managing Include Files	461
Filtering	463
Viewing the Full Design	463
Showing Only Out-of-Date Nodes	464
Executing Associated Actions	464
<hr/>	
18. Running Advanced Analyses	465
Setting Up Parametric Analyses	465
Adding New Sweeps	466
Adding Sweeps to the Design Variables Table	466
Adding Sweeps Using the Parametric Analyses Dialog Box	467
Editing Sweeps	471
Deleting Sweeps	471
Correcting Invalid Sweeps	472
Using the Tandem Parameter Generator	472
Performing Parametric Analyses	473
Printing and Annotating Parametric Results	474
Setting Up Corner Analyses	474

Contents

Specifying Corner Parameters	475
Specifying Design Variables for Corners	475
Specifying Model Files for Corners	477
Defining Corners	478
Adding a Single Corner	478
Adding Multiple Corners	479
Specifying Corner Ranges	481
Specifying Corner Variables in Tandem	482
Setting Up a Corner Sweep Through Multiple Cellviews	485
Removing Corner Parameters	488
Enabling and Disabling Corner Parameters	488
Deleting Corners	488
Copying Corners	489
Removing Duplicate Corners	489
Enabling and Disabling Corners	490
Enabling and Disabling Single Corner	490
Enabling and Disabling Multiple Corners	490
Changing Corner Values	491
Renaming Corners	492
Enabling or Disabling the Corner Analysis Setup	492
Setting up Specifications for Each Corner Condition	493
Sorting Corners	493
Defining Corner Groups	493
Importing and Exporting Corner Information to a File or State	494
Setting Up Aging Analyses	496
Setting Up Monte Carlo Analyses	498
Setting Up Monte Carlo Analyses for FineSim	502
Using Sigma Amplification in Monte Carlo Analysis	508
Setting Up Worst-Case Analysis	509
Setting Up Chained Testbenches	513
Opening the Chain Testbenches Tool	514
Specifying a Source Testbench Using the Chain Testbenches Tool	514
Specifying a Destination Testbench Using the Chain Testbenches Tool	516
Specifying Destination Variables Using the Chain Testbenches Tool	516
Chaining Corner Variables	517
Chaining Design Variables	520
Chaining Sweep Variables	521
Computing Interface Elements	522

Contents

Preparing for Multi-Technology Simulation of 3-D Integrated Circuits (3DICs) 522

A. PrimeWave Design Environment Scripting	528
Invoking the PrimeWave Design Environment Shell	528
Creating PrimeWave Design Environment Scripts	529
Example 1: Initializing New PrimeWave Design Environment Sessions	530
Example 2: Setting Up and Executing Simulations	531
Example 3: Measuring a Bandgap Power Up	531
Example 4: Measuring the Effect of Temperature on a Bandgap Output Versus Resistor Network Selected	534
Running PrimeWave Design Environment Scripts	536
Setting Shell Preferences	537
B. PrimeWave Design Environment Shortcut Keys	538
PrimeWave Design Environment Function Shortcut Keys	538
PrimeWave Design Environment Keypad Shortcut Keys	538
C. Tools for Job Status and Monitoring	541
Job Monitor Tool	541
Displaying the Job Monitor Tool	542
Viewing Running Job Processes	542
Applying Job Process Controls	543
Managing Job Options	544
Customizing the Job Name	546
Filtering Job Groups	546
Displaying a Netlist	547
Adding Job Classes With <code>xt::createJobClass</code> Tcl	547
Sample Job Class for Design Rule Checking on Linux Systems	547
Sample Job Class for HSPICE Simulation on Solaris Systems	549
Adding Job Process Waiting With <code>xt::wait</code> Tcl	550
Sentry Monitor Library Tool	551
Watchdog Tool	551
Installing the Watchdog Tool	552
Watchdog Outputs	553
Database	553
Database Structure	553

Contents

Log Files	554
Watchdog Report Statistics	555
Query on Database	556
D. MOSRA Integration with the PrimeWave Design Environment	557
Introduction to MOSRA	557
MOSRA Aging Models	558
PrimeWave Design Environment Integration of the MOSRA Flow	558
Stress Simulation	559
Degraded Simulation	559
Combined Stress + Degraded Simulation	560
Graphical User Interface	560
Prerequisites	560
Recommended Steps	563
PrimeWave Design Environment MOSRA Flow Example	564
Configuring a Combined Stress + Degraded Simulation	565
Running a Combined Stress + Degraded Simulation	572
Configuring a Stress-Only Simulation	572
Running a Stress-Only Simulation	573
Inspecting and Cross-Selecting Results to Source Design	574
Stress + Degraded Simulation: Simulation Waveforms and Measurements	579
Configuring the 'Degraded' Simulation	581
Running the Degraded Simulation	582
Observing Device Behaviors at Multiple Aging Points	583
Looking at the PrimeWave Design Environment Results Table	585
Saving and Loading State	586
Conclusion	587

About This User Guide

This guide covers instructions for defining and working in the PrimeWave Design Environment, which is a custom environment for netlisting and simulation. The PrimeWave Design Environment includes the use of the PrimeSim Continuum simulators and the PrimeWave Design Environment waveform viewer for pre- and post-layout circuit simulation and analysis.

This preface includes the following sections:

- [New in This Release](#)
 - [Related Products, Publications, and Trademarks](#)
 - [Conventions](#)
 - [Customer Support](#)
 - [Statement on Inclusivity and Diversity](#)
-

New in This Release

Information about new features, enhancements, and changes, known limitations, and resolved Synopsys Technical Action Requests (STARs) is available in the PrimeWave Design Environment Release Notes on the SolvNetPlus site.

Related Products, Publications, and Trademarks

For additional information about the PrimeWave™ Design Environment tool, see the documentation on the Synopsys SolvNetPlus support site at the following address:

<https://solvnetplus.synopsys.com>

You might also want to see the documentation for the following related Synopsys products:

Synopsys® PrimeWave™ Design Environment
Synopsys® Custom Compiler™
PrimeSim™ Continuum
PrimeSim™
PrimeSim™ Pro
PrimeSim™ SPICE
PrimeSim™ HSPICE®
PrimeSim™ XA
PrimeSim™ CCK

PrimeSim™ MOSRA
VCS® PrimeSim™ AMS
Synopsys® FineSim®
Synopsys® HSPICE®
Synopsys® VCS®
Synopsys® Verdi®
Cadence® Spectre® Circuit Simulator

MathWorks® MATLAB

Conventions

The following conventions are used in Synopsys documentation.

Convention	Description
Courier	Indicates syntax, such as <code>write_file</code> .
<i>Courier italic</i>	Indicates a user-defined value in syntax, such as <code>write_file design_list</code>
Courier bold	Indicates user input—text you type verbatim—in examples, such as <code>prompt> write_file top</code>
Purple	<ul style="list-style-type: none">Within an example, indicates information of special interest.Within a command-syntax section, indicates a default, such as <code>include_enclosing = true false</code>
[]	Denotes optional arguments in syntax, such as <code>write_file [-format fmt]</code>
...	Indicates that arguments can be repeated as many times as needed, such as <code>pin1 pin2 ... pinN</code> .
	Indicates a choice among alternatives, such as <code>low medium high</code>
\	Indicates a continuation of a command line.
/	Indicates levels of directory structure.
Bold	Indicates a graphical user interface (GUI) element that has an action associated with it.
Edit > Copy	Indicates a path to a menu command, such as opening the Edit menu and choosing Copy .

Convention	Description
Ctrl+C	Indicates a keyboard combination, such as holding down the Ctrl key and pressing C.

Customer Support

Customer support is available through SolvNetPlus.

Accessing SolvNetPlus

The SolvNetPlus site includes a knowledge base of technical articles and answers to frequently asked questions about Synopsys tools. The SolvNetPlus site also gives you access to a wide range of Synopsys online services including software downloads, documentation, and technical support.

To access the SolvNetPlus site, go to the following address:

<https://solvnetplus.synopsys.com>

If prompted, enter your user name and password. If you do not have a Synopsys user name and password, follow the instructions to sign up for an account.

If you need help using the SolvNetPlus site, click REGISTRATION HELP in the top-right menu bar.

Contacting Customer Support

To contact Customer Support, go to <https://solvnetplus.synopsys.com>.

Statement on Inclusivity and Diversity

Synopsys is committed to creating an inclusive environment where every employee, customer, and partner feels welcomed. We are reviewing and removing exclusionary language from our products and supporting customer-facing collateral. Our effort also includes internal initiatives to remove biased language from our engineering and working environment, including terms that are embedded in our software and IPs. At the same time, we are working to ensure that our web content and software applications are usable to people of varying abilities. You may still find examples of non-inclusive language in our software or documentation as our IPs implement industry-standard specifications that are currently under review to remove exclusionary language.

1

Setting Up and Invoking the PrimeWave Design Environment

This chapter describes how to invoke and set up the PrimeWave Design Environment.

The PrimeWave Design Environment provides a fully featured simulation environment, in which simulators can be used to verify analog and mixed-signal designs. You can set up simulator control statements, such as analyses, models, and options, run simulations, and analyze the results. The PrimeWave Design Environment also provides parametric sweep capabilities, a corner analysis tool, and an interface to Monte Carlo analysis in your simulator. An integration with a waveform viewer, the WaveView tool, supports postprocessing. The PrimeWave Design Environment also includes the Results Analyzer tool to assist with plotting and expression building and several other built-in functions.

This chapter contains the following topics:

- [Invoking the PrimeWave Design Environment from the Custom Compiler Tool](#)
- [Specifying a File Text Editor](#)
- [Opening a Design](#)

Invoking the PrimeWave Design Environment from the Custom Compiler Tool

Note:

Before you invoke the PrimeWave Design Environment, specify the editor you want to use to edit netlists. See [Specifying a File Text Editor](#).

Starting from the S-2021.09 PrimeWave Design Environment release, you must install the PrimeWave Design Environment to run the following PrimeWave Design Environment features:

- Simulation Environment - PrimeWave
- Plot Signals in the Schematic Editor

- Annotate Operating Point in the Schematic Editor
- AMS visualization
- Voltage-dependent rule checking (VDRC) flow
- Electromigration back annotation (EMBA)
- Electromigration and IR Drop Analysis (EMIR)
- In-design electromigration
- Power devices

To use these features in the PrimeWave Design Environment tool, install the PrimeWave Design Environment tool and set \$PATH to the bin directory in the PrimeWave Design Environment installation, before starting the Custom Compiler tool. Note that the PrimeWave Design Environment version must be compatible with the Custom Compiler version.

To use the PrimeWave Design Environment from the Custom Compiler tool, both must be in \$PATH.

For example, when using Bourne-compatible shells such as sh, ksh, bash, ash, and zsh:

```
export PATH=/path/to/primewave/S-2021.09/bin:$PATH
```

```
export PATH=/path/to/customcompiler/S-2021.09/bin:$PATH
```

When using csh-compatible shells such as csh and tcsh:

```
setenv PATH /path/to/primewave/S-2021.09/bin:$PATH
```

```
setenv PATH /path/to/customcompiler/S-2021.09/bin:$PATH
```

For Feature and Service Pack releases, it is recommended that you install and use the same versions of the Custom Compiler and PrimeWave Design Environment tools, which will always be compatible. For other release types and cases where there is a need to use different versions of the tools together, the Custom Compiler software will automatically check whether the PrimeWave Design Environment version being run is compatible. If found incompatible, a warning message ("PRIMEWAVE-001") will be printed in the console and in the log file, to notify you to take appropriate action.

To invoke the PrimeWave Design Environment:

1. If Custom Compiler is not yet open, invoke Custom Compiler by entering the following on the command line:

```
custom_compiler &
```

The Custom Compiler home page opens.

2. Click the **PrimeWave Design Environment** button.



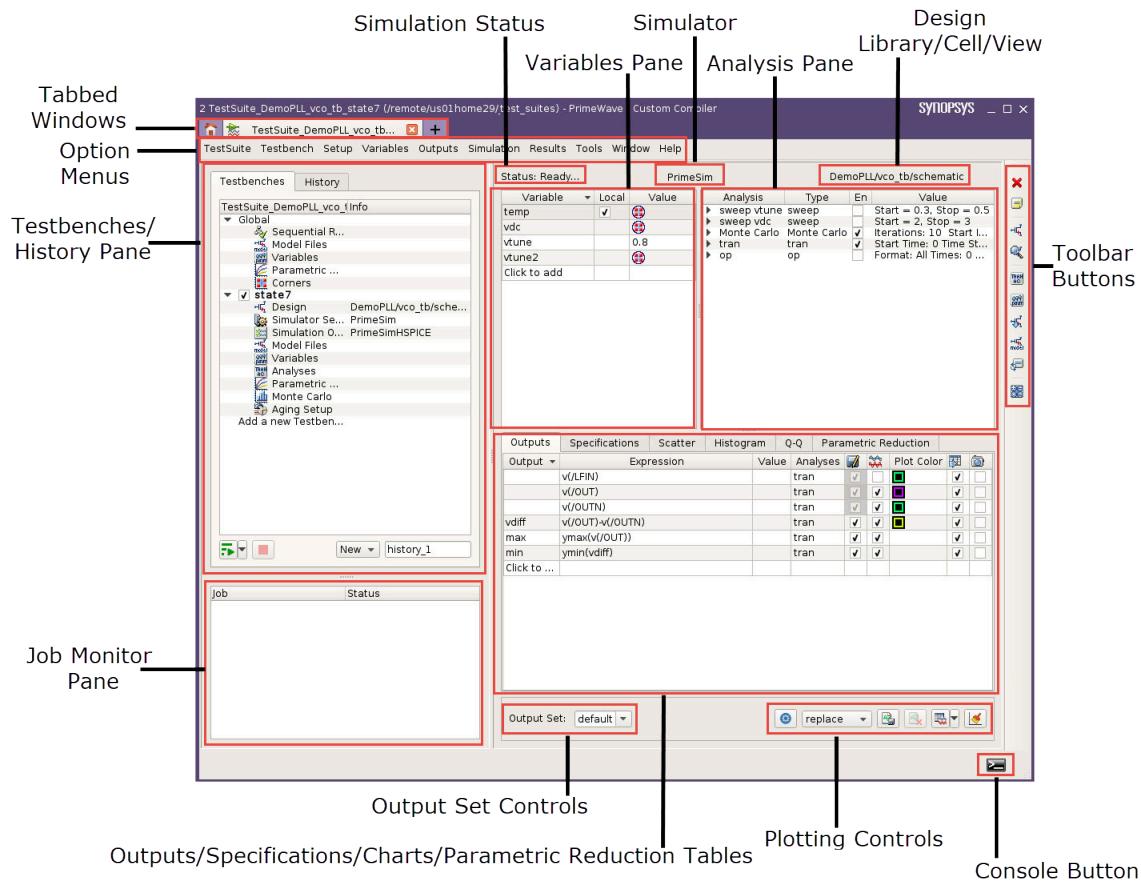
The PrimeWave Design Environment window opens.

Note:

Some of the features shown below are limited availability. For information about these features, refer to SolvNetPlus article #000036534 "[How to Enable the PrimeWave Design Environment Flow-Based Interface](#)" or consult your Synopsys representative.

Chapter 1: Setting Up and Invoking the PrimeWave Design Environment

Invoking the PrimeWave Design Environment from the Custom Compiler Tool



If you have existing PrimeWave Design Environment states that are saved in the cell directory of your design, you can also start the PrimeWave Design Environment by double-clicking the state name in the **Views** column of the Library Manager. The PrimeWave Design Environment window opens with the state loaded.

Specifying a File Text Editor

To specify an external file editor, which is used to edit netlists in the PrimeWave Design Environment, set the `EDITOR` environment variable before you start the PrimeWave Design Environment.

- For csh, type the following on the command line before you start (or add it to your `.cshrc` startup file):

```
setenv EDITOR vi
```

- For bash, type the following on the command line before you start (or add it to your `.bashrc` startup file):

```
export EDITOR=vi
```

You can also specify the editor in the **General Options** dialog box, which you can access by choosing **Options > General** from the **Home** tab. Switch to the **Editors** tab and specify an editor in the **Text Editor** field. The `EDITOR` environment variable is referenced by default.

Opening a Design

Note:

Design libraries must be defined in the `lib.defs` file (or library definition file referenced by the `lib.defs` file) before you can open a design in Custom Compiler and then the PrimeWave Design Environment. See the *Custom Compiler Environment User Guide* for more information on how set up a library definition file.

Note:

If you move your design library, you can update the location by choosing **Edit > Update Library References** from the Library Manager dialog box.

To open a design:

1. Click the **Library Manager** button on the Custom Compiler home page.



The Library Manager opens in a new tab.

2. Select a library and cell.
3. (Optional) Select a cell category.
4. Double-click an associated view to open the design in the Schematic Editor.

The Schematic Editor opens with the design loaded in editing mode.

You can now invoke the PrimeWave Design Environment from the Schematic Editor using **Tools > PrimeWave**. The top-level design in the Schematic Editor window becomes the design under test in your PrimeWave Design Environment session.

2

Setting Up Testbenches

This chapter contains information on how to set up testbenches in the PrimeWave Design Environment.

This chapter contains the following major topics:

- [Working with Multiple Testbenches](#)
- [Choosing a Design](#)
- [Choosing a Simulator](#)
- [Setting Simulator Options](#)
- [Choosing a PrimeSim Simulation Engine](#)
- [Setting Environment Options](#)
- [Specifying Model Files](#)
- [Specifying Parasitic Back-Annotation Flow](#)
- [Sequential Testbenches](#)
- [Saving and Plotting Terminal Node Voltages](#)
- [Specifying Include Files](#)
- [Specifying External Images](#)
- [Specifying Digital Timing](#)
- [Specifying Convergence Aids](#)
- [Adding Stimulus](#)
- [Specifying Temperature](#)
- [Working with History Points](#)
- [Saving States](#)

- [Loading States](#)
 - [Saving Tcl Scripts](#)
-

Working with Multiple Testbenches

Design verification is an essential step during circuit design. As design complexity increases, the number of testbenches also increases, making the verification a time-consuming process. The PrimeWave Design Environment Multiple Testbench (MTB) environment in Custom Compiler allows you to set up multiple testbenches using the same cellView or different cellViews. You can generate a comprehensive verification test suite by setting up corners and sweeps across multiple testbenches and viewing results in one location.

If there are errors when you load an MTB session, they are indicated by the exclamation point icon (!) at the global scope and at the local scope for the testbenches. This icon on the testbenches is an indicator that there is a problem you need to resolve. The tooltip on the exclamation icon gives a brief description of the error.

In most cases, the error is due to an invalid path definition for the model files. To correct the error, choose **Setup > Model Files** from the PrimeWave Design Environment main window.

This section contains information on the following topics:

- [Creating Test Suites](#)
- [Specifying Test Suite Options](#)
- [Adding Testbenches](#)
- [Adding Testbenches from States](#)
- [Deleting Testbenches](#)
- [Ejecting Testbenches](#)
- [Cloning Testbenches](#)
- [Copying Testbench Categories](#)
- [Saving Test Suite Results](#)
- [Saving Multiple Testbench History](#)
- [Loading Test Suite Results](#)

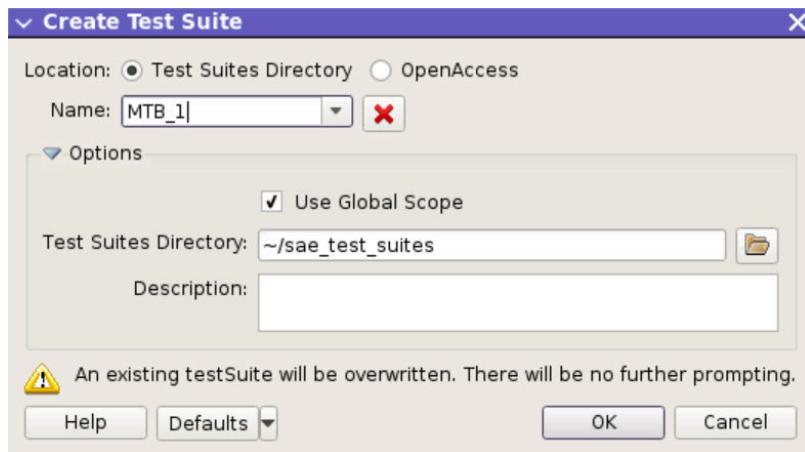
Creating Test Suites

A test suite is a single location where you can gather multiple testbenches together for simulation.

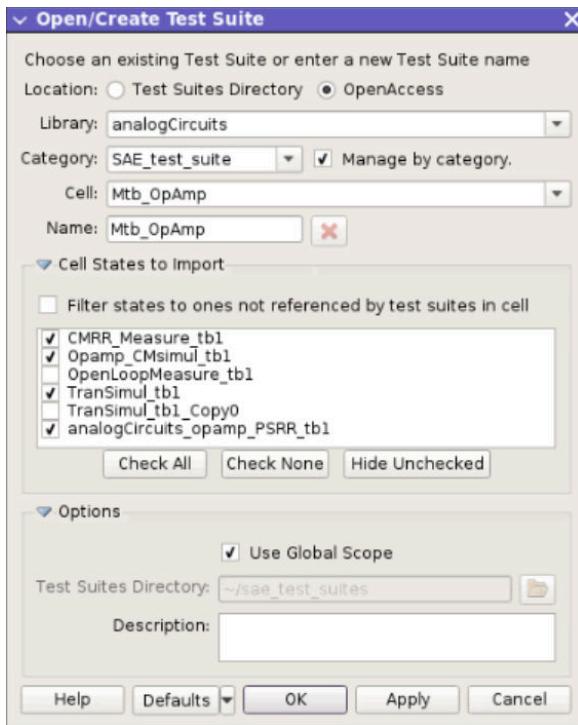
To create a test suite:

1. Choose **Tools > PrimeWave** from the Custom Compiler **Home** page, or choose **TestSuite > Open** from the PrimeWave Design Environment menu.

The **Open/Create Test Suite** dialog box opens.



2. Select the location of the test suite, the **Test Suites Directory** or **OpenAccess**. If you choose OpenAccess, skip to [Step 4](#).
3. Enter a name for the test suite in the **Name** text field.
4. If you choose **OpenAccess**, you can select **Cell States to Import** if you choose a cell that already contains designs and single-testbench cell states. If you only want to import states that are not already referenced by other MTB views within the target cell, select the **Filter states to ones not referenced by test suites in cell** option, which reduces the cell state list accordingly.



5. (Optional) Uncheck **Use Global Scope to disable global settings.**

Two modes of MTB are supported: ones with a global scope and ones without global scope, which you need to define when you create an MTB.

When **Use Global Scope** is enabled, you can share design variables and device models among testbenches (for example, common design variables for VDD/VSS). Corners analyses are only available at the global scope; they apply across all testbenches.

When **Use Global Scope** is disabled, each testbench is self-contained, and no data (design variables) or analyses (corners or sweeps) are shared across testbenches.

6. Choose a directory in which to save your test suite.

The test suite is saved in the `~/SAE_test_suites/` directory by default.

7. (Optional) Add a comment to the test suite to be created.

8. Click **OK.**

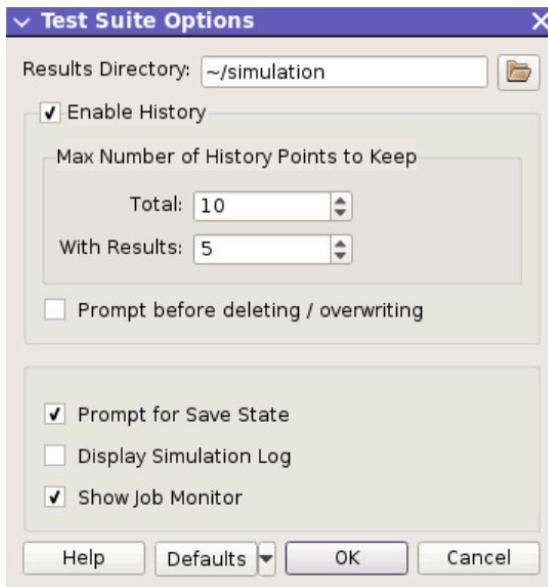
The test suite is saved and an empty MTB session is created.

Specifying Test Suite Options

To specify test suite options:

1. Choose **TestSuite > Options**.

The **Test Suite Options** dialog box opens.



2. Enter a path for the **Results Directory**, which is the location where the results hierarchy for simulations across all testbenches are created.

The default root run directory is `~/simulation`.

3. (Optional) Uncheck **Enable History** to disable the creation of history points.

When **Enable History** is checked, simulations are run in a new history point when no history point is active or the current history point when there is an active history. For more details, see [Saving Multiple Testbench History](#).

4. (Optional) Adjust the **Max Number of History Points to Keep**. Check **Prompt before deleting/overwriting** to show a prompt before history points are deleted or overwritten.
5. (Optional) Check **Prompt for Save State** to show a prompt to save the current state when closing or resetting the session.
6. (Optional) Check **Display Simulation Log** to show the simulation log file as a simulation runs.

7. (Optional) Check **Show Job Monitor** to display the Job Monitor when simulations run.
8. Click **OK** to save your changes.

Adding Testbenches

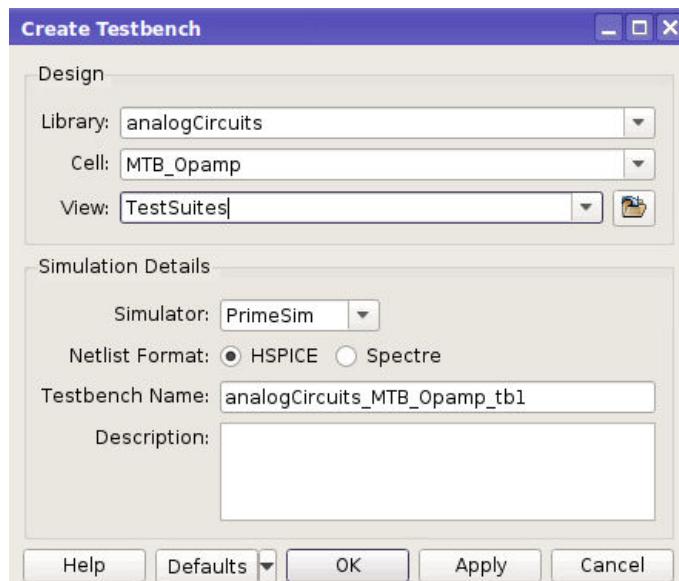
Note:

If you have only a single testbench and have not yet created a test suite, see [Creating Test Suites](#).

To add a testbench to your test suite:

1. Choose **Testbench > Add** or **Testbench > Add from State**.

The **Create Testbench** dialog box opens.



2. Choose a **Library**, **Cell**, and **View**.
3. Choose a **Simulator** for the added testbench.
4. Choose a **Netlist Format** for the added testbench.
5. Enter a **Testbench Name** for the testbench.

By default, a name is created using the following format:

<library_name>_<cell_name>_tb<#>

6. Add a **Description** of the testbench.

- Click **OK** to add the testbench to your test suite.

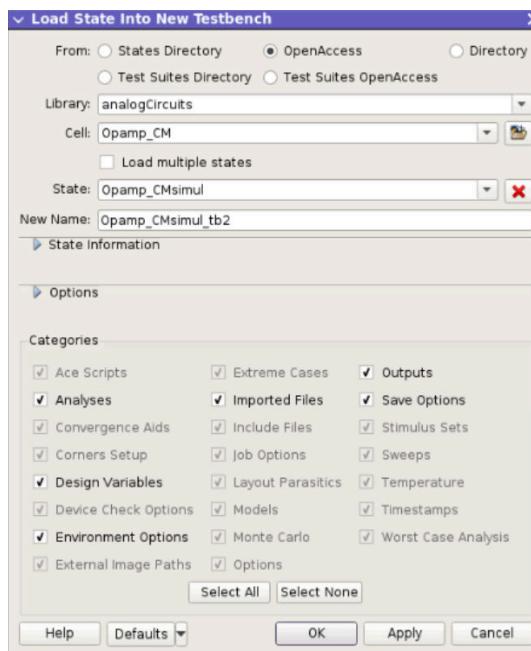
If you want to add more than one testbench, you can instead click **Apply**, then repeat [Step 2 to Step 6](#).

Adding Testbenches from States

To add a testbench from a state:

- Choose **Testbench > Add from State**.

The **Load State into New Testbench** dialog box opens.



- Choose the location of the saved state that you want to load:

- States Directory**

When you choose this option, information is loaded from a state that is specific to a single testbench. The state might be one that you created while working with just a single testbench. Once you specify the **Library**, **Cell**, and **Simulator**, you can choose from a list of available states in the **State** menu.

Choose which state you want to load from the **State** menu.

The **New Name** field is autopopulated with a unique testbench name not currently found in the session. You can overwrite this autopopulated value.

State Information is located in the following directory hierarchy:

```
<top_level_states_dir>/<Library_Name>/<Cell_Name>/ <simulator_name>/  
<state_name>
```

You can set the `<top_level_states_dir>` in the **States Directory** text box. If you do not see this text box, click the arrow next to the **Options** section heading to expand the section.

- **OpenAccess**

When you choose this option, you are accessing information stored in the cell directories of your design libraries. These states might be ones that you created while working with just a single testbench.

Once you specify the **Library** and **Cell**, you can choose from a list of available states in the **State** menu.

Note:

When the **OpenAccess** option is enabled, you do not need to specify a directory because the **States Directory** resides in your design data.

The **New Name** field is autopopulated with a unique testbench name not currently found in the session. You can overwrite this autopopulated value.

(Optional) Select the **Load multiple states** option if you want to import more than one STB state into the target MTB session. This adds the **States to import** list to the **Load State into New Testbench** dialog box.

- **Directory**

When you choose this option, you can load states from any specified directory.

Enter the directory path into the **States Directory** text box or browse to the directory that contains the desired state information. If you do not see this text box, click the arrow next to the **Options** section heading to expand the section.

Once you specify a **States Directory**, any state directories that are located in that directory are listed in the state menu.

Note:

The on-disk representation of the state is identical regardless of the option that you select. Only the directory structure above the actual state directory differs with the options.

You can find information about the state in the **State Information** section. The state information includes the time and date of the last change, as well as any comments.

- **Test Suites Directory**

When you choose this option, saved state information from a selected test suite is loaded, which you can then use to populate other test suites.

With this option, you are accessing state information, which is created by choosing **Testsuite > Save As** from the PrimeWave Design Environment main window when working with multiple testbenches. The state information includes the current state for all testbenches in the test suite and is separate from the single testbench state information.

Enter the a path in the **Test Suites Directory** text box for the directory that contains the test suite information that you want to load. If you do not see the **Test Suites Directory** text box, you might need to click the down arrow next to the **Options** heading to expand that section.

Once you specify a test suite directory, any test suites that are located in that directory are listed in the **Test Suite** menu. You can then choose a **Testbench** from which to load a state.

The **New Name** field is autopopulated with a unique testbench name not currently found in the session. You can overwrite this autopopulated value.

You can find information about the state in the **State Information** section. The state information includes the time and date of the last change, as well as any comments.

- **Test Suites Open Access**

When you choose this option, you can populate other test suites using individual testbench information.

State information is saved into an Open Access location for all your testbenches in your test suite. The test suite acts as a cellview, and the testbenches are listed as views.

A special view called `_testsuite_` represents the entire suite. Double-clicking this view in the Library Manager opens the PrimeWave Design Environment in multiple testbench mode and loads the suite.

Once you specify the **Library** and **Cell**, you can choose from a list of available test suites in the **Test Suite** menu.

Note:

When this option is enabled, you do not need to specify a directory since the state directory resides in your design data.

You can then choose a testbench from the chosen test suite from which to load a state, and enter a new name for the newly created testbench.

You can find information about the state in the **State Information** section. The state information includes the time and date of the last change, as well as any comments.

3. (Optional) If you choose **Test Suites Directory** or **Test Suites Open Access**, then click **Include effective global inheritance of source test suite** to inherit the global settings from the source test suite to the imported testbench.

Otherwise, skip to the next step.

4. (Optional) Uncheck **Remove localizations that match globals**.

This option is enabled by default.

When enabled, this option automatically detects commonalities between the testbench and global scope settings, and the testbench inherits the matching global scope settings. In addition, the pertinent testbench level settings are removed.

5. In the **Categories** section, select what information in the state you want to load.

All information is loaded by default.

6. Click **OK** to load the state into the PrimeWave Design Environment.

Deleting Testbenches

To delete a testbench from your test suite, click the name of the testbench, then click **Delete**  on the toolbar on the right side of the PrimeWave Design Environment main window. You can also right-click the name of a testbench, and choose **Delete** from the menu that opens.

Ejecting Testbenches

To eject a testbench from a test suite so you can run simulations and debug just that testbench outside of the test suite, right-click the name of a testbench, and choose **Eject to Single Testbench** from the menu that opens. A new PrimeWave Design Environment window opens with just that testbench loaded. The ejected testbench includes a copy of the global settings and testbench-specific settings.

Note:

When you run a simulation for an ejected testbench, that simulation does not overwrite any simulations run when the ejected testbench was part of a test suite.

Cloning Testbenches

To create an exact copy of a testbench and add it to your existing test suite, right-click the name of the testbench you want to clone, then choose **Clone** from the menu that opens. The cloned testbench is added to the list of testbenches in your test suite.

Note:

After cloning, there is a relationship between the original and cloned testbenches, they are not strictly individuals. Any change to the schematic immediately applies to the original design and all clones.

Copying Testbench Categories

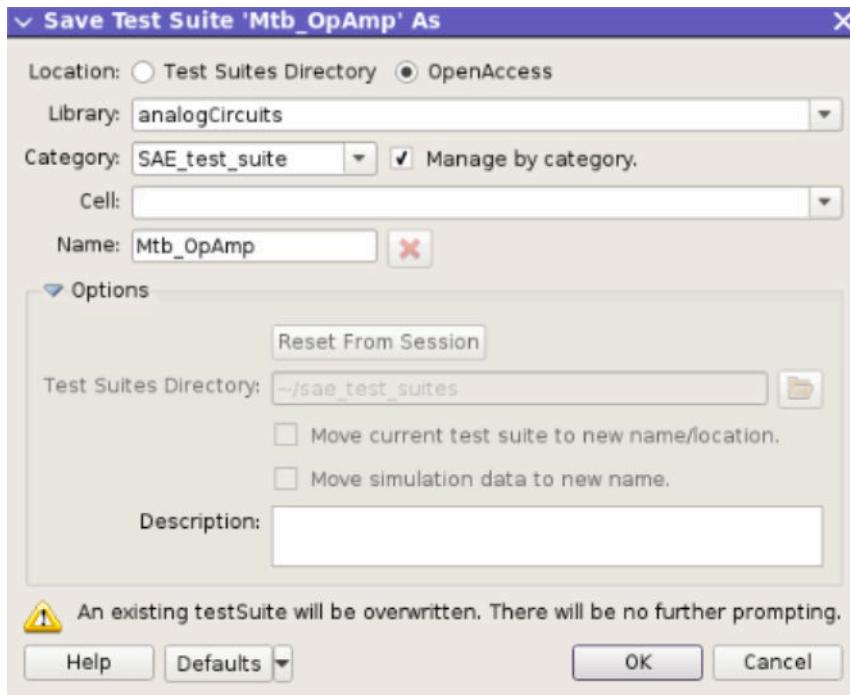
To copy one or more categories of a testbench to another testbench:

1. Click the name of the testbench from which you want to copy categories of a testbench.
 2. Choose **Testbench > Copy Categories**.
The **Copy <testbench_name> Categories** dialog box opens.
 3. Click the name of the testbench to which you want to copy categories.
 4. Choose one or more **Categories** to copy.
 5. Click **OK** to copy the categories to the specified testbench.
-

Saving Test Suites

To save a test suite, choose **TestSuite > Save**.

If you choose **TestSuite > Save As**, you can choose to save your test suite to a specific **Test Suites Directory** or to an **OpenAccess** database.



The following table describes the **Save Test Suite As** options.

Option	Description
Location	Saves the test suite to a specific Test Suites Directory or to an OpenAccess database.
Library	For OpenAccess, the library where the test suite resides.
Category	For OpenAccess, when Manage by category is checked, allows you to manage test suites by category.
Cell	For OpenAccess, the cell where the test suite resides.
Name	The name of the test suite.
Options	<p>Reset from Session If you have made any changes in the current dialog box, clicking this option resets the dialog box with the values that were there when you first opened the dialog box (reading the MTB session information).</p> <p>Move current test suite to new name/location saves the current test suite with a new name and/or in a new location.</p> <p>Move simulation data to new name saves any available simulation data with a new name.</p>

Saving Test Suite Results

To save test suite results:

1. Choose **TestSuite > Save Results As**.

The **Save Test Suite <test_suite_name> Results As** dialog box opens.

2. Choose where you want to save the test suite results:

- **Test Suite Run Directory Archive**

This option saves test suite results to the `<root_dir>/<test_suite_name>/userSavedData/testSuites` directory, which is a location that correlates to the current test suite. The `<root_dir>` is the working directory you set up in the **Test Suite Options (TestSuite > Options)**, and the `<test_suite_name>` is the name of the current test suite.

The path displayed in the **Directory** text field is created using the current root test suite simulation run directory, the test suite name, and the static `userSavedData/testSuites` path. This path is not editable.

Enter a name for the results in the **Name** text box.

- **Directory**

This option stores the test suite simulation results to a directory you specify in the **Save** text box.

3. Click **OK** to save the location for your test results.

Saving Multiple Testbench History

The history points of the MTB are taken during the following instances:

- When you perform a simulation, either at the test-suite level or for a specific testbench. The history point is named **Run Simulation**.
- When you explicitly perform an MTB save. The history point is named **MTB Save**.

When you open a test suite, you can see two tabs on the PrimeWave Design Environment main window, the **Testbenches** tab and the **History** tab. The panel under the **History** tab shows all the history points (snapshots) of the MTB available in a specific run directory.

Click the green **Netlist and Run** button  to run the simulations in `history_1` (the default name given to the first history_point). You can edit the name as per context. The drop-down beside the history name helps toggle between new and current history points.



You can specify the maximum number of history points that you want to save by using the **TestSuite Options** dialog box. Choose **TestSuite > Options**. The **Test Suite Options** dialog box opens. (See [Specifying Test Suite Options](#).)

In the **Test Suite Options** dialog box, you can specify the following:

- **Results Directory** - The root directory where the simulation results/setup snapshot is saved.
- **Enable History** - Allows you to enable or disable history point creation. Simulations are run in:
 - a new history point when no history point is active
 - the current history point when there is an active history
- **Max Number of History Points to Keep** - You can specify the number of history points (or snapshots) of the MTB that you want to save. This group box has the following fields:
 - **Total** - Specify the maximum number of history points that you want to save, which includes both the simulation setup snapshots (no simulation results) and the simulation setup and results snapshots.
 - **With Results** - Specify the maximum number of history points that you want to save with the simulation setup and results snapshots for each testbench.

The value of **Total** is greater than the value of **With Results**. When the number of history points reaches the maximum value assigned for **Total**, the oldest history point is deleted. Whereas, when the number of history points reaches the maximum value assigned for **With Results**, the results in the oldest history point are deleted. The oldest snapshot now contains only the setup Measurement Results database, which allows you to view generated scalar data in the ResultsView. The icon of the oldest history point changes to the icon that corresponds to the setup type. Only simulator-generated results are removed.

In Multiple Testbench (MTB), history points can be major or minor. The testbenches are selected from the test suite tree. You can choose to save a number of history points with or without results in the following ways:

- When you run the test suite, a major history point is created with the results. When a **Save** operation is done, a minor history point is created under the last major history point, giving you the ability to apply different measurements or specification on the same results. This allows you to still have access to results from the current run, prior to any subsequent saves.

Caution:

Changes made to the setup that are saved can cause the results to be invalid. Be aware of this situation and rerun the test suite if necessary.

- When you choose to run one or more testbenches by selecting them and using the right-click menu to invoke a run, a new major history point is not created. Instead the results in the current (or last) major history point are overwritten for the selected testbenches. This allows you to correct the setup and rerun a subset of testbenches that might have failed on the previous run.

Additionally, you can defer running one or more testbenches and run them later (for example, overnight jobs), and then display them as if you ran them with the other testbenches.

For further information, see [Working with History Points](#).

Loading Test Suite Results

To load test suite results:

- Choose **TestSuite > Load Results**.

The **Load Test Suite Results** dialog box opens.

- Choose from where you want to load test suite results:

- Test Suite Run Directory Archive**

This option loads test suite results from the `../userSavedData/testSuites` default directory architecture, which is a location that correlates to the current test suite.

Choose the name of the results you want to load from the **Name** menu.

- Directory**

This option loads the test suite simulation results from a directory you specify in the **Load** text box.

- Click **OK** to save the location for your test results.

Comparing Testbench Settings

You can compare all testbenches in the current session.

1. Choose **Tools > Testbench Setting Comparison**. The **Testbench Setting Comparison** dialog box opens.

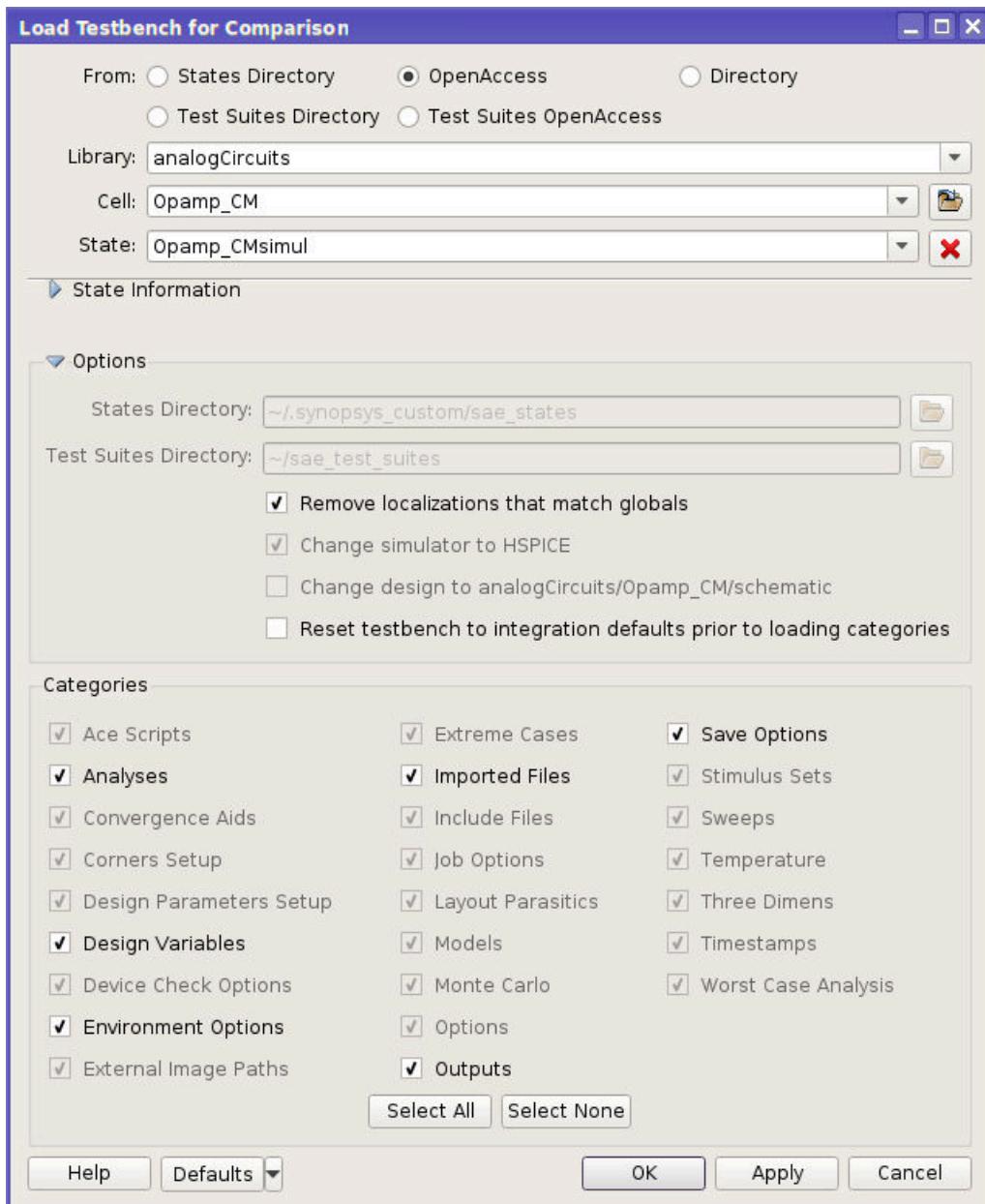
Testbench Setting Comparison - Mtb_OpAmp					
Target Reference		Opamp_CMsimul_tb1	OpenLoopMeasure_tb1	TranSimul_tb1	CMRR_Measure_tb1
Item	filter	Opamp_CMsimul_tb1	filter	filter	filter
analyses	analyses				
analysis					
ac					
card					
definition		ac		ac	ac
designVar					
enabled		1		1	1
freqPoint					
name		ac		ac	ac
numPoints		100		100	100
options					
poi					
start		1		1	1
stop		1G		1G	1G
sweep		Frequency (Hz)		Frequency (Hz)	Frequency (Hz)
sweepType		Decade		Decade	Decade
useCard		0		0	0
dc					
card					
dataDrive...					
dataDrive...					
definition	dc				
designVar					
enabled	1				
hysteresis	0				
name	dc				
numPoints					
options					
poi					
source	/V3				
start	0				
stepSize	0.05				
stop	1.2				
sweep	Source				
sweepType	Linear Steps				
useCard	0				
op					
card					
definition		op	op	op	op
enabled	1	1	1	1	1
format	All	All	All	All	All
format2	All	All	All	All	All
format3	All	All	All	All	All
format4	All	All	All	All	All
format5	All	All	All	All	All
format6	All	All	All	All	All
interpolat...	0	0	0	0	0

The dialog box displays a comparison of all the testbenches in the current session. The target reference is set to the first active testbench by default.

The dialog opens with attributes having differing values among different testbenches expanded, while attributes that have the same values are collapsed.

Differing values are highlighted in blue.

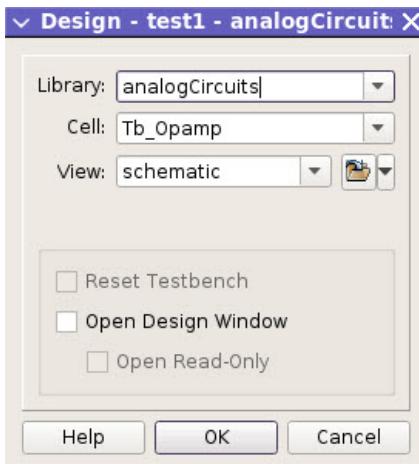
2. (Optional) Change the **Target Reference** testbench to change the target testbench items shown in the first column of the table.
3. (Optional) Edit the values in the comparison table. Note that you cannot create a new object in the comparison table, but can only edit items that are already in the OA.
4. (Optional) Click **Refresh** to reset the comparison table after adding new attributes or creating new objects in other places. The table then re-creates the comparison results.
5. (Optional) Use the filters in the first row of the table to filter the comparison results.
6. (Optional) Click **Load Testbench** to open the **Load Testbench for Comparison** dialog box, select another testbench to compare, and click **OK**.



- Click **X** to close the **Testbench Setting Comparison** dialog box.

Choosing a Design

If you do not launch the PrimeWave Design Environment from an editor, or if you want to change the design that is associated with your current session, choose **Setup > Design** from the PrimeWave Design Environment main menu bar. The **Design** dialog box opens and you can select a **Library**, **Cell**, and **View** that are associated with your design.



If you have changed the selected simulator, you can check **Reset Testbench** to reset all testbench categories to defaults for the selected simulator.

Check **Open Design Window** to open the design in a new tab when you click OK. Check **Open Read-Only** to open the design as read-only.

Choosing a Simulator

Note:

After choosing a simulator, see [Setting Simulator Options](#) for information on selecting a simulation engine and other simulation options.

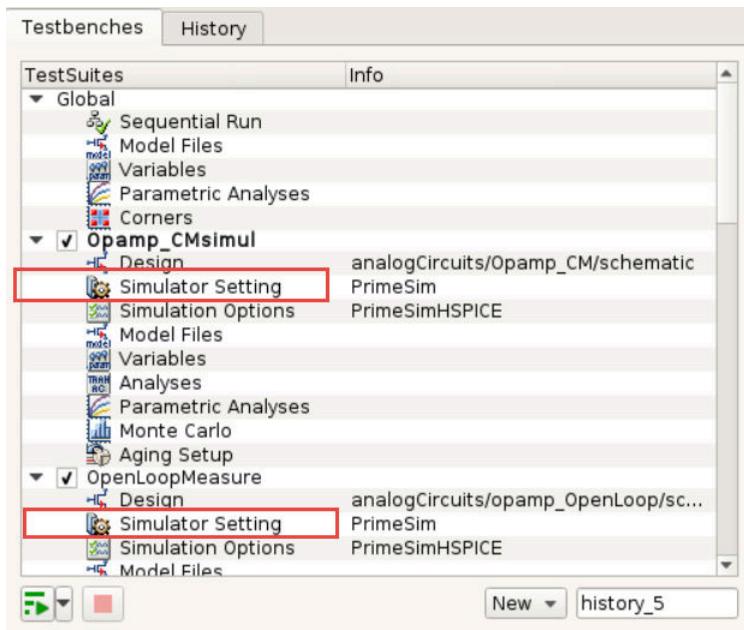
To choose a simulator for simulating your testbench:

1. Choose **Setup > Simulator** from the PrimeWave Design Environment main menu bar. Alternatively, double-click the **Simulator Setting** in the tree under the current testbench name.

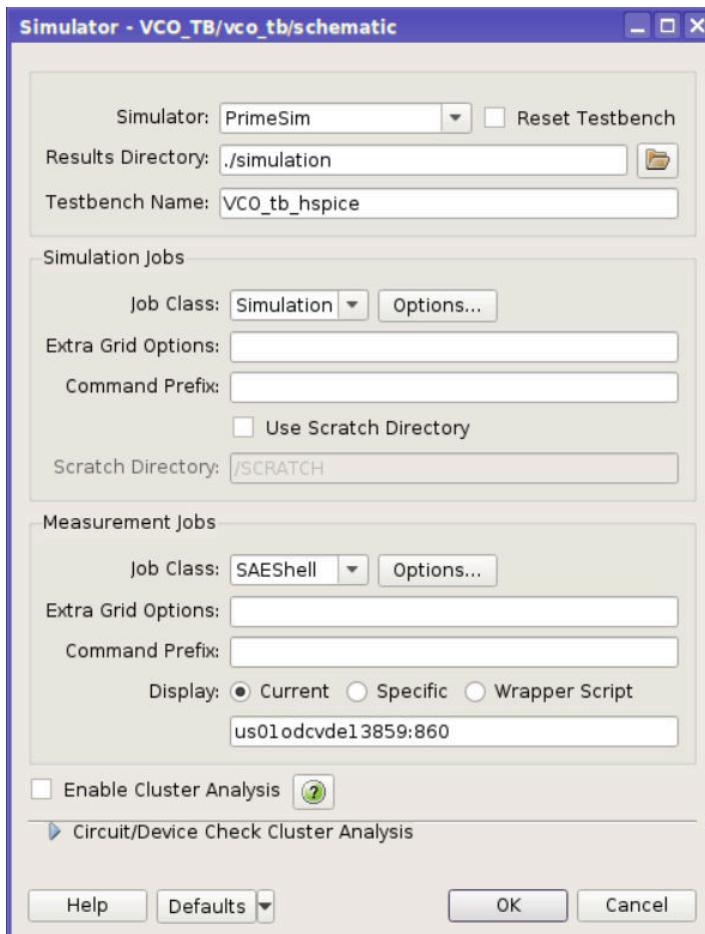
Note:

Some of the features shown below are limited availability. For information about these features, refer to SolvNetPlus article #000036534 "[How to](#)

[Enable the PrimeWave Design Environment Flow-Based Interface](#)" or consult your Synopsys representative.



The **Simulator** dialog box opens.



2. Select the **Simulator** from the list of simulators available for your chosen netlist format. (If you do not find the simulator you would like to use with the PrimeWave Design Environment, contact your Synopsys representative to determine options.)

Note:

After choosing the PrimeSim simulator, see [Setting Simulator Options](#) for information on selecting a simulation engine and other PrimeSim simulation options.

3. If you have changed the selected simulator, you can select **Reset Testbench** to reset all testbench categories to defaults for the selected simulator.
4. If necessary, specify the **Netlist Format** you are using.
5. For mixed-signal flows, specify **System-Verilog** or **Verilog-AMS**.
6. Specify a **Results Directory** where you want simulation results to be stored.

Note:

This defaults to your home directory, but for many users there is insufficient disk space in `$HOME`. Consult your IT or CAD group to define a better location, such as `./simulation` or `/slowfs/<username>/simulation`.

7. Change the **Testbench Name** if necessary. You can give a single-testbench design a testbench name using this option.
8. In the **Simulation Jobs** section, choose one of the following job management options from the **Job Class** menu:
 - **<Default>**
 - **Simulation**
 - **Netlisting**
 - **SAEShell**Click **Options** to specify such items as choosing a queue engine or job limiting settings for these job classes.
9. (Optional) Enter any additional options in the **Extra Grid Options** text box.
These options are saved as part of the PrimeWave Design Environment state.
10. (Optional) Enter any command prefixes in the **Command Prefix** text box.
11. (Optional) Check **Use Scratch Directory** to enable a scratch directory for your simulation.
Using a scratch disk can provide a performance improvement for many network environments. The `/SCRATCH` directory is created by default when this option is enabled. You can enter the path to another scratch directory location in the **Scratch Directory** text box if needed.
12. Choose a job class from the **Job Class** menu.
13. (Optional) Enter any additional options in the **Extra Grid Options** text box.
These options are saved as part of the PrimeWave Design Environment state.
14. (Optional) Enter any command prefixes in the **Command Prefix** text box.
15. Choose a **Display**, which is where measurement calculations and waveform viewer jobs are displayed:
 - **Current**
The current display.

- **Specific**

A specific display that you enter in the **Display** text box just below this option.

- **Wrapper Script**

A wrapper script you use for displaying simulations and results.

Note:

Using another display besides your current display can help increase simulation processing and plotting speed. This option is only available when you choose to save an image from a plot for one or more outputs.

16. (Optional) Select **Enable Cluster Analysis** to perform a cluster analysis of circuit/device check violations.
17. (Optional) Expand the **Circuit/Device Check Cluster Analysis** section and select your options for cluster analysis:
 - **Cluster Algorithm** is set by default to the **MiniBatchKMeans** algorithm, which is very similar to the commonly used K-means algorithm, but performs better with large data sets.
 - **Number of Clusters to Extract** (disabled when using the Affinity-Propagation Clustering algorithm)
 - **Number of Violations to Display per Cluster** sets the initial number of violations to display. Note that regardless of this setting, it is still possible to show all violation items for each cluster.
18. Click **OK** to save your settings.

Setting Simulator Options

After setting the simulator (described in [Choosing a Simulator](#)), you can choose a simulation engine and set other simulation options.

To access simulator options, choose one of the following menu items based on which integrated simulator you are using:

To access the simulator options for these integrations...	...choose this menu item	...see also
PrimeSim	Simulation > Options or press the O key	Choosing a PrimeSim Simulation Engine

To access the simulator options for these integrations...	...choose this menu item	...see also
VCS PrimeSim AMS	Simulation > Analog Global Options or Simulation > Digital Options	Setting Block-Level VCS PrimeSim AMS Options Setting Mixed-Signal Simulator Options Setting VCS PrimeSim AMS Run Mode
FineSim VCS	Simulation > Analog Global Options or Simulation > Digital Options	Setting Block-Level FineSim VCS Options Setting Mixed-Signal Simulator Options
FineSim	Simulation > Global Options or Simulation > Block Options	Setting Block-Level FineSim Options

For more information on the simulation options for your integrated simulator, see the following documentation:

Simulator	Document
PrimeSim Continuum Simulators	<i>PrimeSim Continuum User Guide</i>
VCS PrimeSim AMS, FineSim VCS	<i>VCS PrimeSim AMS User Guide</i>

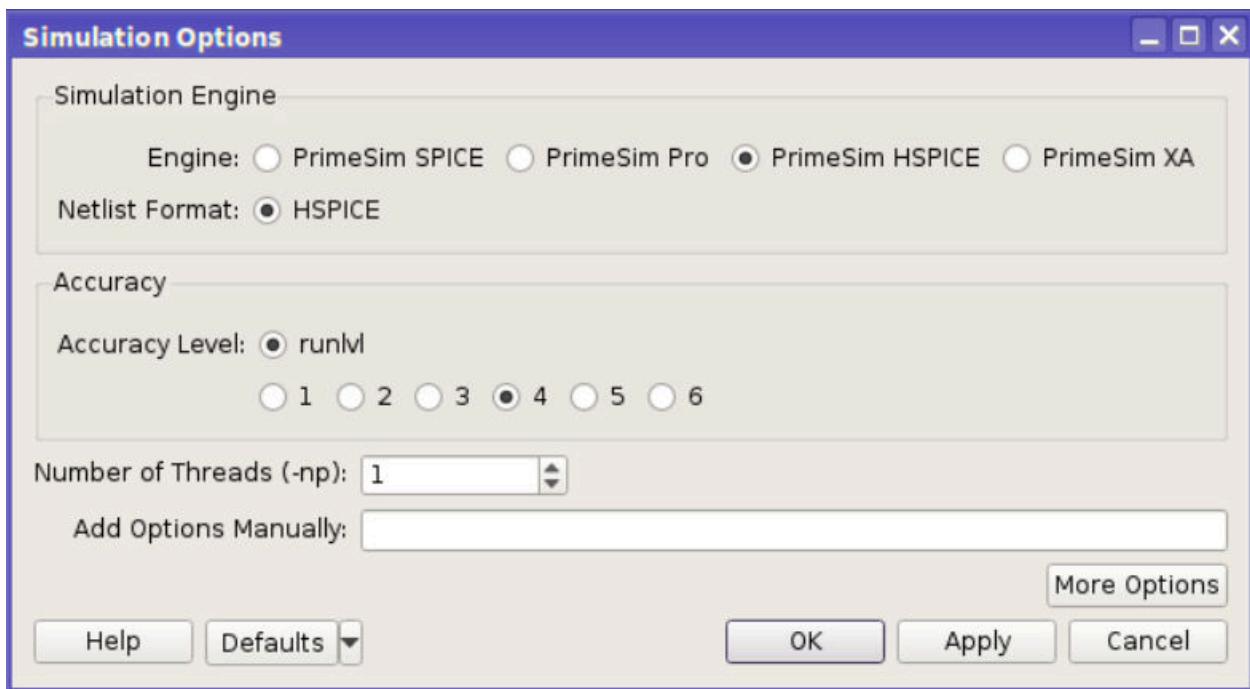
Choosing a PrimeSim Simulation Engine

After setting the simulator to PrimeSim (described in [Choosing a Simulator](#)), you can choose a PrimeSim simulation engine and set other simulation options.

To select a PrimeSim simulation engine:

1. From the main PrimeWave Design Environment window, choose **Simulation > Options**.

The **Simulation Options** window opens.



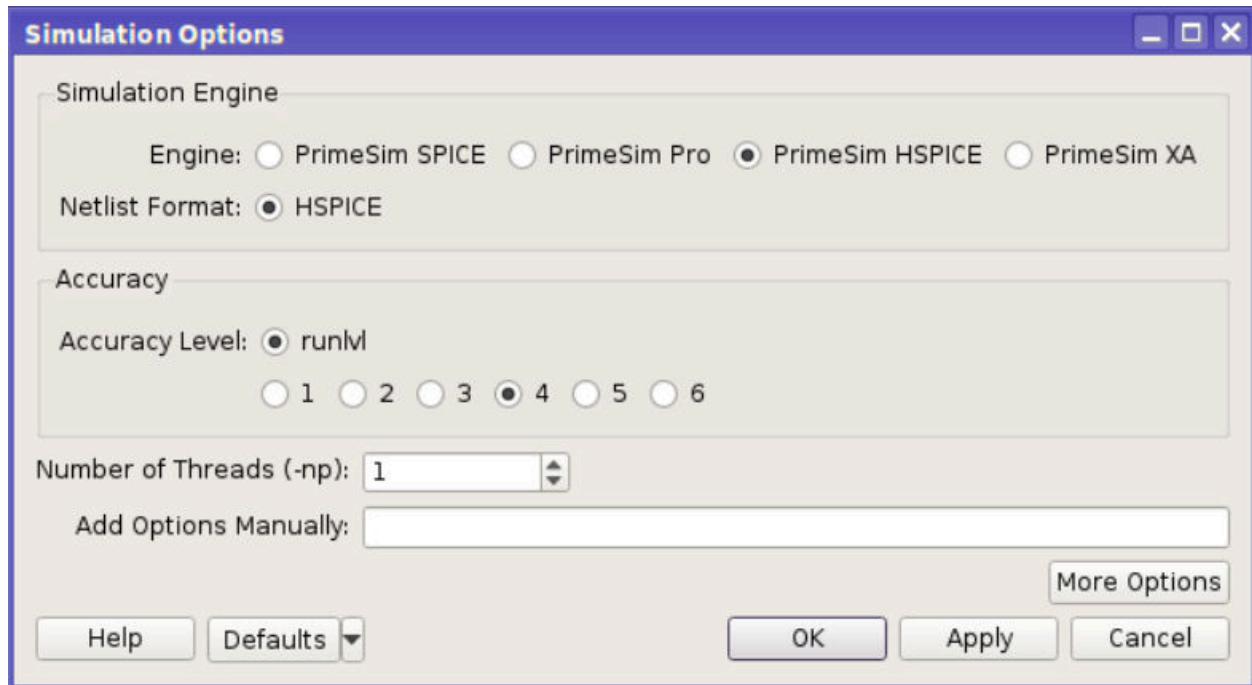
2. Select the **Engine**.
3. Set the rest of the simulation options for the simulation engine, as described in the following topics:
 - [PrimeSim HSPICE Simulation Options](#)
 - [PrimeSim SPICE Simulation Options](#)
 - [PrimeSim Pro Simulation Options](#)
 - [PrimeSim XA Simulation Options](#)

PrimeSim HSPICE Simulation Options

After setting the simulator to PrimeSim (described in [Choosing a Simulator](#)), you can choose a PrimeSim simulation engine and set other simulation options.

To set the PrimeSim HSPICE simulation options:

1. In the **Simulation Options** dialog box (**Simulation > Options**), set the **Engine** to **PrimeSim HSPICE**.

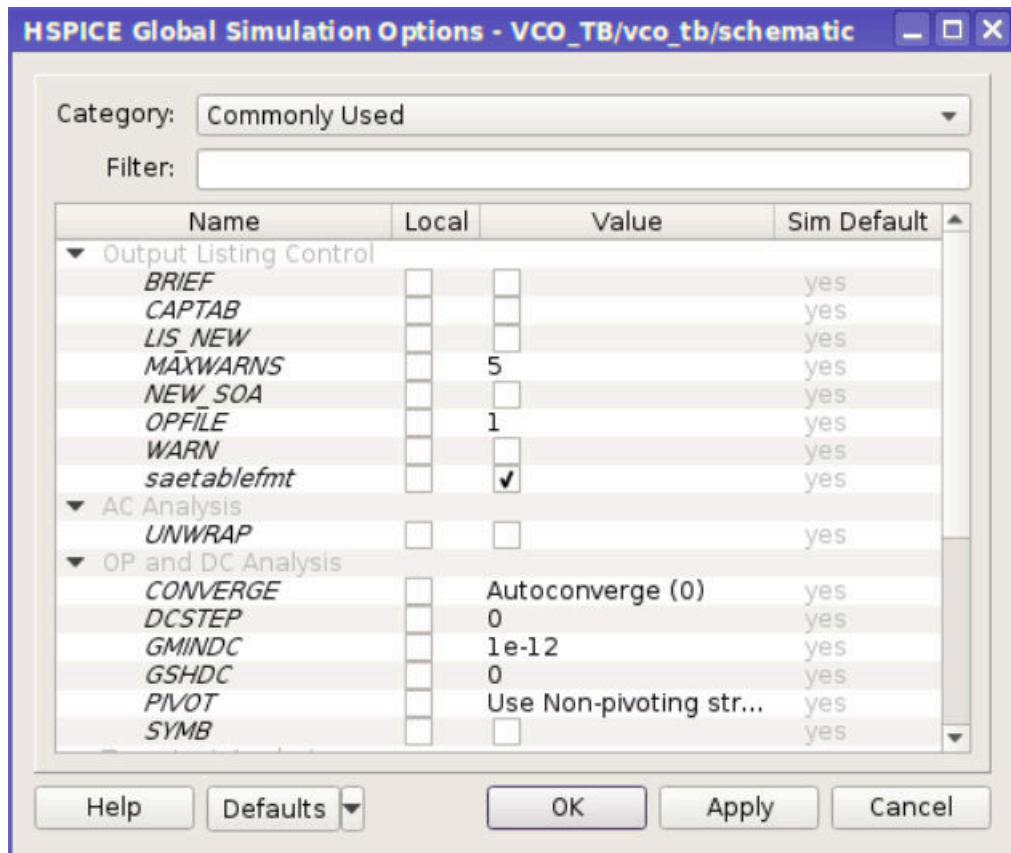


2. The **Netlist Format** is **HSPICE** by default.
3. Set the **Accuracy** options as described in the following table.

Table 1 Accuracy Options

Options	Description
Accuracy Level	runlvl : Choose from 1 through 6 . The default is 4 .

4. Set the **Number of Threads (-np)**.
 5. (Optional) To set options that are not listed in the dialog box, enter them in the **Add Options Manually** text box.
 6. (Optional) Click **More Options** to open the **Global Simulation Options** dialog box, where you can select options that are not listed in the **Simulation Options** dialog box.
- Select those options and click **OK** to close the **Global Simulation Options** dialog box.



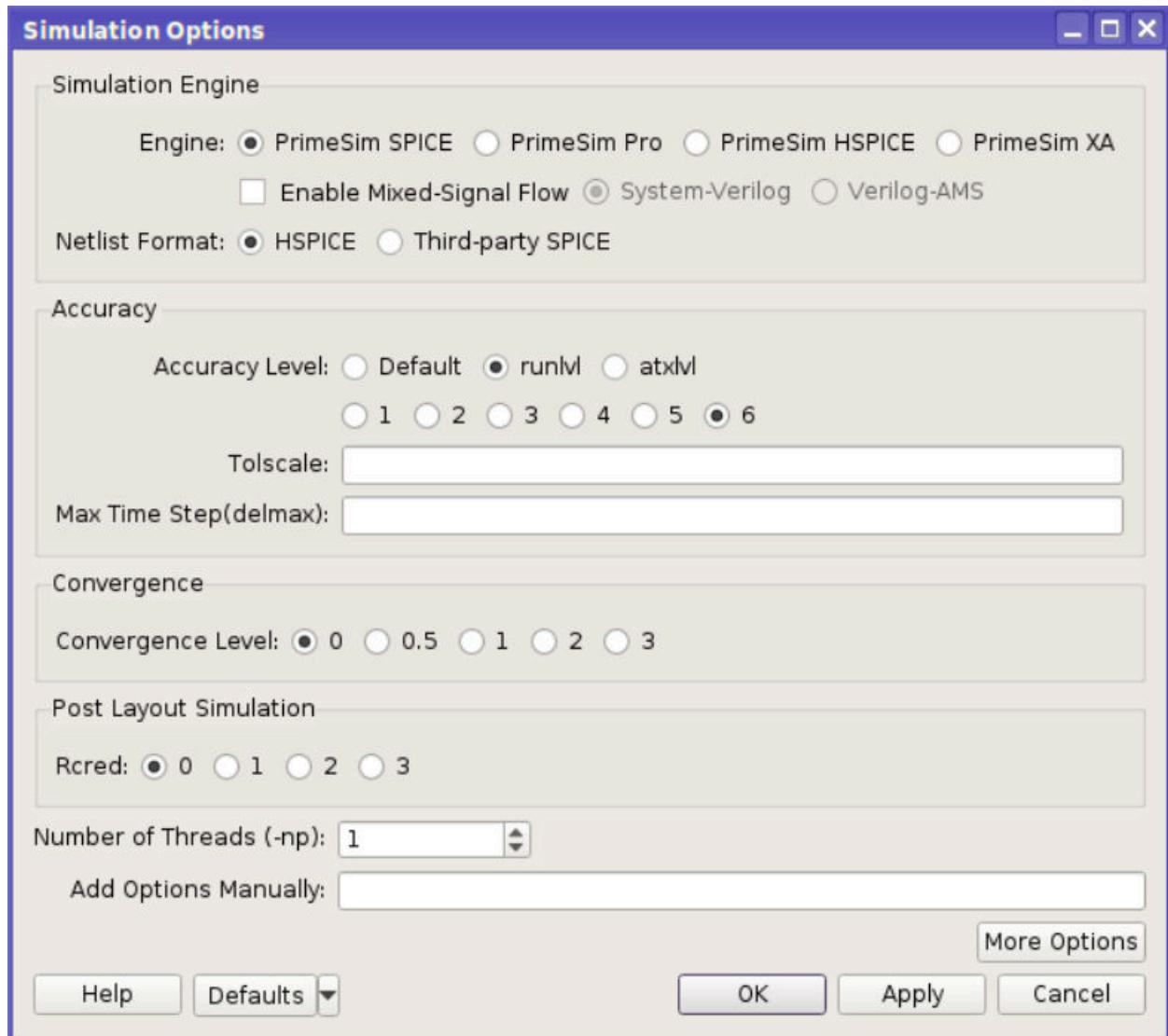
7. Click **OK** to apply the changes and close the **Simulation Options** dialog box.

PrimeSim SPICE Simulation Options

After setting the simulator to PrimeSim (described in [Choosing a Simulator](#)), you can choose a PrimeSim simulation engine and set other simulation options.

To set the PrimeSim SPICE simulation options:

1. In the **Simulation Options** dialog box (**Simulation > Options**), set the **Engine** to **PrimeSim SPICE**.



2. (Optional) Select **Enable Mixed-Signal Flow** and select **System-Verilog** or **Verilog-AMS**. See [Setting Mixed-Signal Simulator Options](#) for more information.
3. Set the **Netlist Format** to either **HSPICE** (the default) or **Third-party SPICE**.
4. Set the **Accuracy** options as described in the following table.

Table 2 Accuracy Options

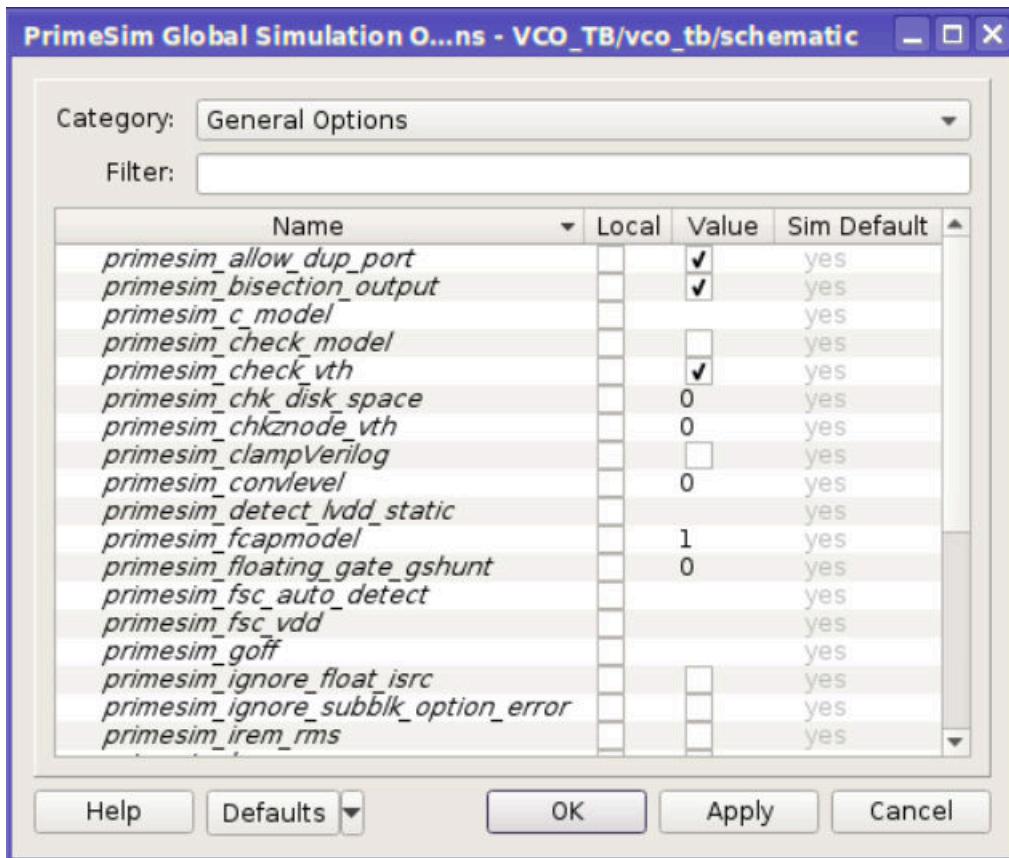
Options	Description
Accuracy Level	Default. When selected, sets the post-layout simulation Rcred value to Default , also. runlvl (the default): Choose from 1 through 6 . The default is 6 . atxlvl : Choose from 1 through 7 .
Tolscale	Sets the tolerance scale, a multiplier applied to all internal tolerance values.
Max Time Step (delmax)	Sets the maximum value of the internal time step for the simulation.

5. Set the **Convergence Level**. Possible values are 0, 0.5, 1, 2, or 3. This setting determines how vigorously the simulator attempts to solve a “time step too small” error.
6. Set the **Rcred** value. Possible values are 0 (the default), 1, 2, or 3. This sets the level of acceleration to use when performing post-layout analog simulation with RC reduction.

Note:

When **Accuracy Level** is set to **Default**, **Rcred** is automatically set to **Default**.

7. Set the **Number of Threads (-np)**.
 8. (Optional) To set options that are not listed in the dialog box, enter them in the **Add Options Manually** text box.
 9. (Optional) Click **More Options** to open the **Global Simulation Options** dialog box, where you can select options that are not listed in the **Simulation Options** dialog box.
- Select those options and click **OK** to close the **Global Simulation Options** dialog box.



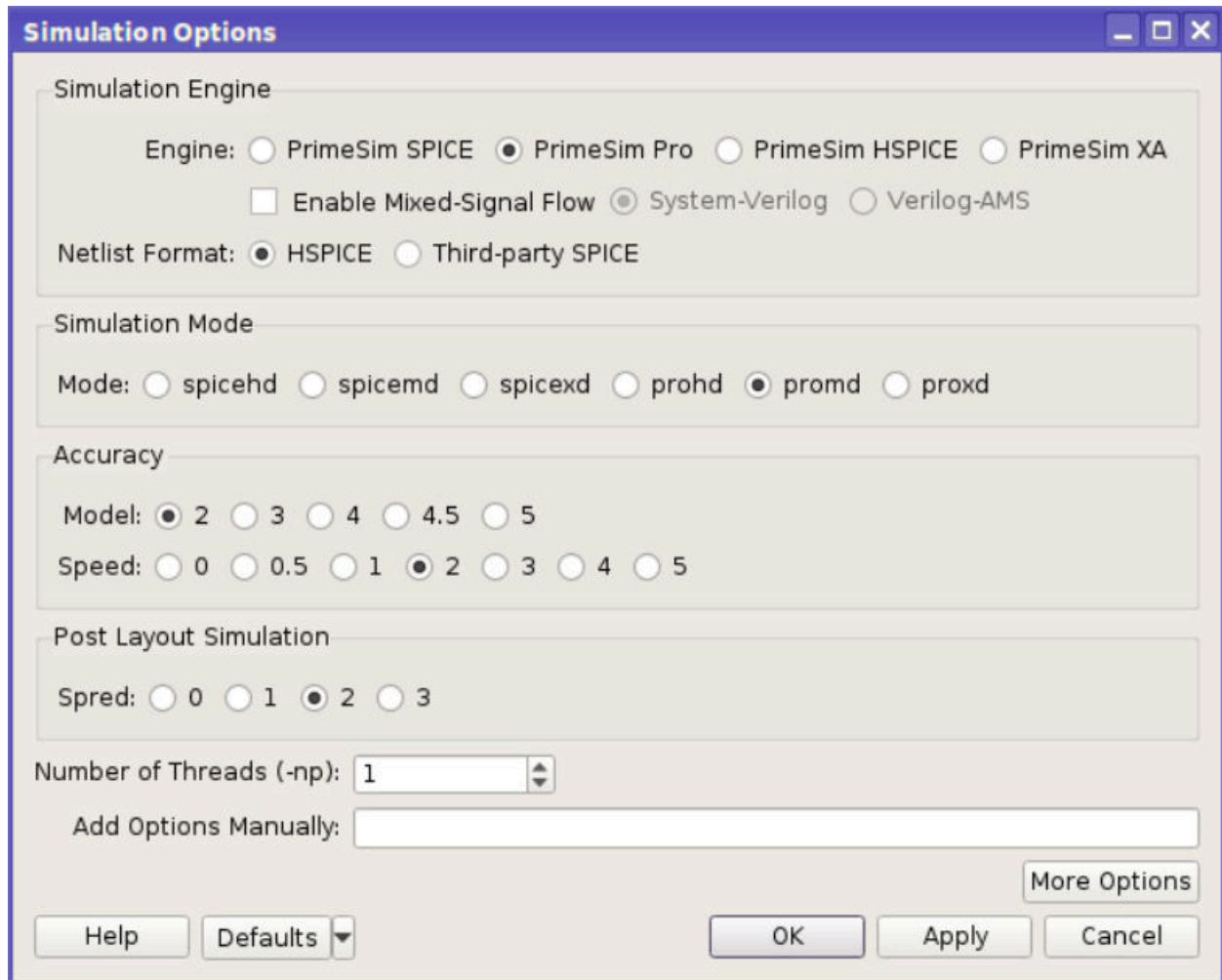
10. Click **OK** to apply the changes and close the **Simulation Options** dialog box.

PrimeSim Pro Simulation Options

After setting the simulator to PrimeSim (described in [Choosing a Simulator](#)), you can choose a PrimeSim simulation engine and set other simulation options.

To set the PrimeSim SPICE simulation options:

1. In the **Simulation Options** dialog box (**Simulation > Options**), set the **Engine** to **PrimeSim SPICE**.

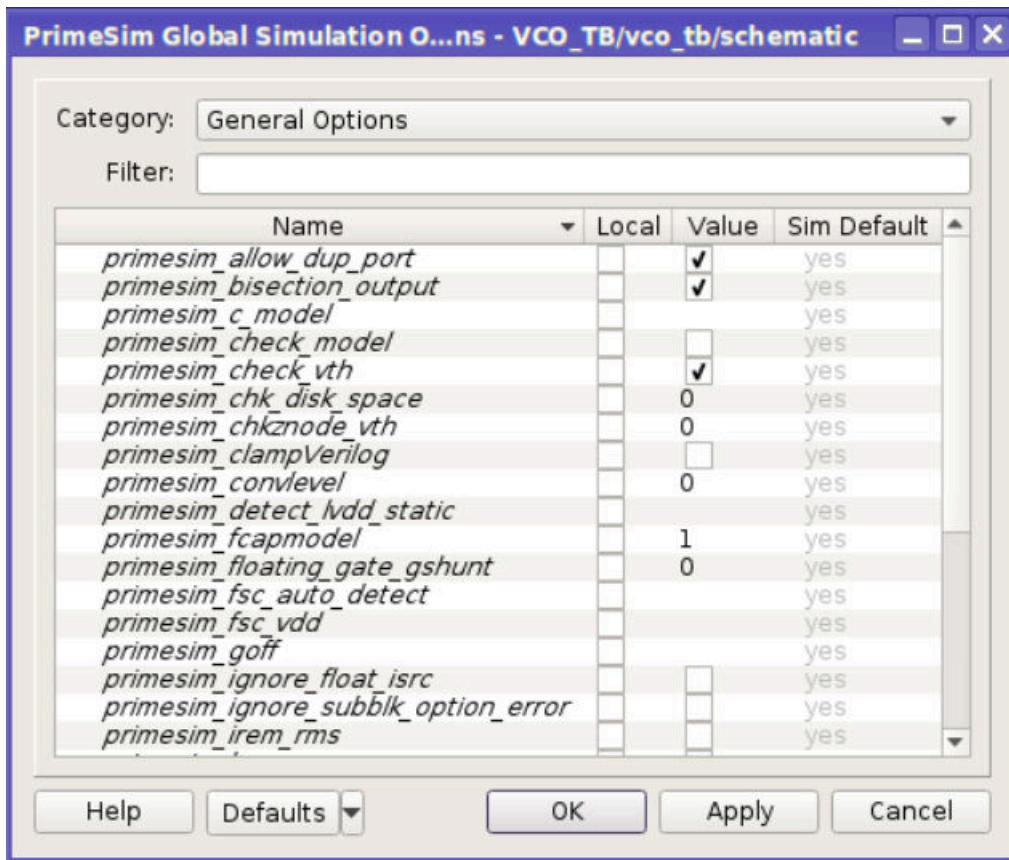


2. (Optional) Select **Enable Mixed-Signal Flow** and select **System-Verilog** or **Verilog-AMS**. See [Setting Mixed-Signal Simulator Options](#) for more information.
3. Set the **Netlist Format** to either **HSPICE** (the default) or **Third-party SPICE**.
4. Select a **Simulation Mode**. Accuracy settings **Model** and **Speed** are dependent on the **Mode** setting.

Table 3 *Simulation Mode and Accuracy Settings*

Mode	Accuracy Settings
spicehd	Model = 4 Speed = 0.5
spicemd	Model = 24 Speed = 1
spicexd	Model = 3 Speed = 1
prohd	Model = 3 Speed = 1
promd (the default)	Model = 2 Speed = 2
proxd	Model = 2 Speed = 3

5. Set the **Spred** value. Possible values are 0 (the default), 1, 2, or 3. This sets the level of acceleration to use when performing post-layout analog simulation with RC reduction.
6. Set the **Number of Threads (-np)**.
7. (Optional) To set options that are not listed in the dialog box, enter them in the **Add Options Manually** text box.
8. (Optional) Click **More Options** to open the **Global Simulation Options** dialog box, where you can select options that are not listed in the **Simulation Options** dialog box.
Select those options and click **OK** to close the **Global Simulation Options** dialog box.



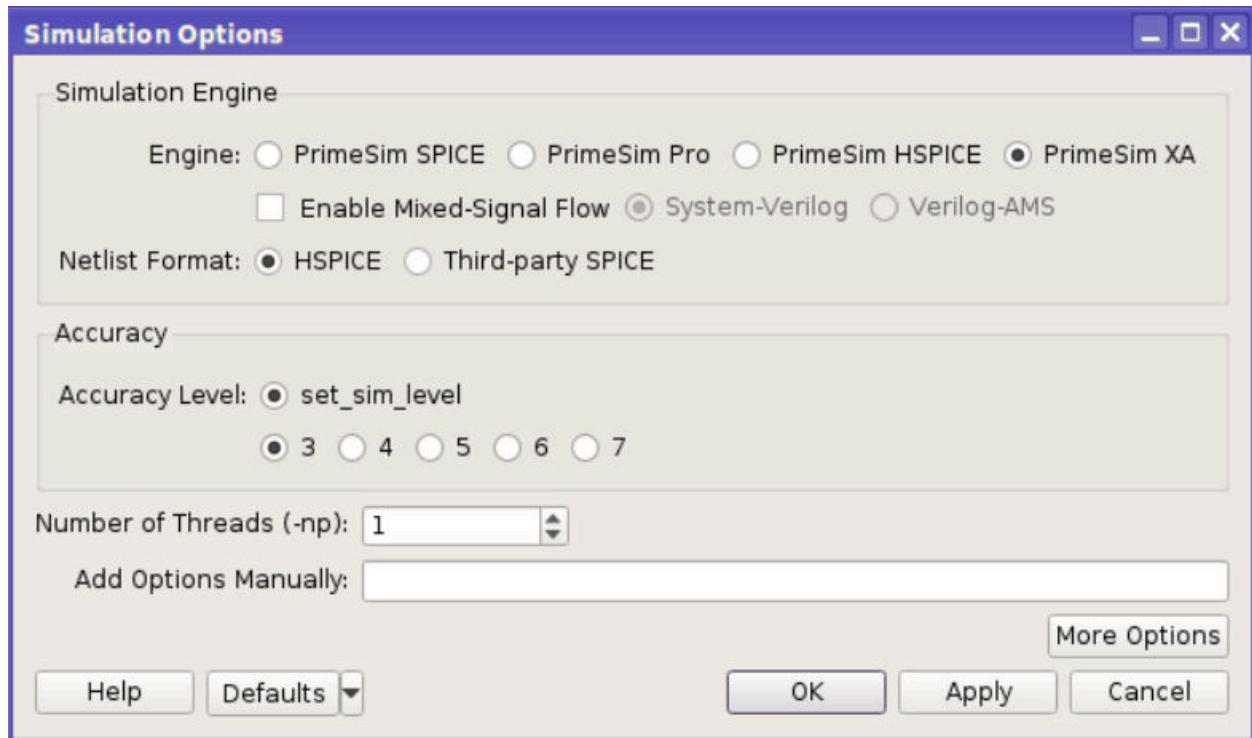
9. Click **OK** to apply the changes and close the **Simulation Options** dialog box.

PrimeSim XA Simulation Options

After setting the simulator to PrimeSim (described in [Choosing a Simulator](#)), you can choose a PrimeSim simulation engine and set other simulation options.

To set the PrimeSim SPICE simulation options:

1. In the **Simulation Options** dialog box (**Simulation > Options**), set the **Engine** to **PrimeSim SPICE**.



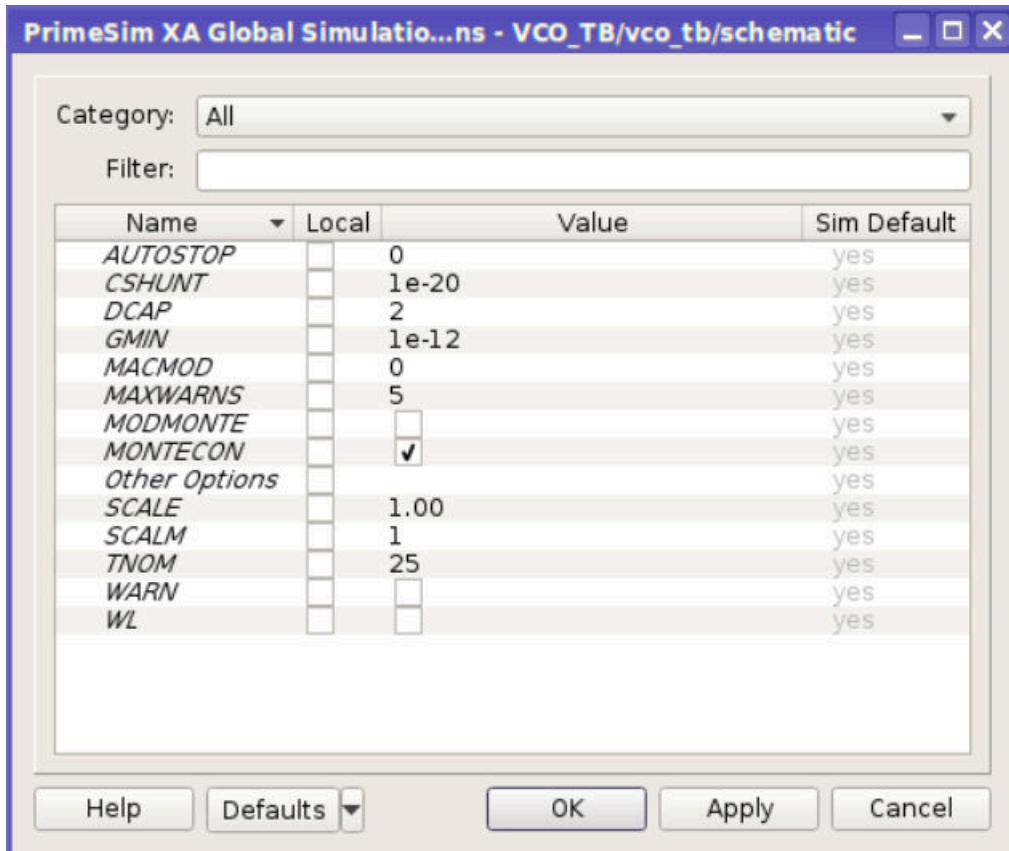
2. (Optional) Select **Enable Mixed-Signal Flow** and select **System-Verilog** or **Verilog-AMS**. See [Setting Mixed-Signal Simulator Options](#) for more information.
3. Set the **Netlist Format** to either **HSPICE** (the default) or **Third-party SPICE**.
4. Set the **Accuracy** options as described in the following table.

Table 4 Accuracy Options

Options	Description
Accuracy Level	set_sim_level : Choose from 3 through 7. The default is 3.

5. Set the **Number of Threads (-np)**.
6. (Optional) To set options that are not listed in the dialog box, enter them in the **Add Options Manually** text box.
7. (Optional) Click **More Options** to open the **Global Simulation Options** dialog box, where you can select options that are not listed in the **Simulation Options** dialog box.

Select those options and click **OK** to close the **Global Simulation Options** dialog box.



8. Click **OK** to apply the changes and close the **Simulation Options** dialog box.

Setting Environment Options

Environment options, among other uses, are simulator-independent controls that determine how the netlister traverses the design hierarchy and how paths to included files are resolved. These options apply across simulators, and are not directly mapped to netlist .OPTION statements.

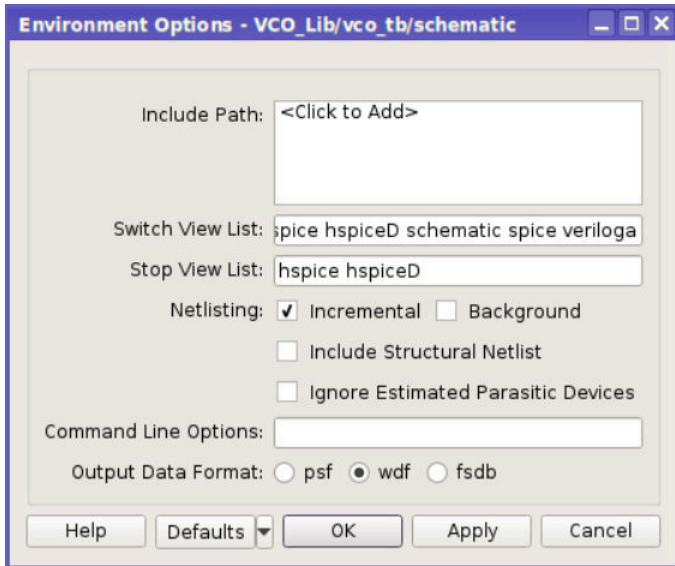
This section contains information on the following topics:

- [Specifying PrimeSim Environment Options](#)
- [Specifying FineSim Environment Options](#)
- [Specifying FineSim VCS Environment Options](#)
- [Specifying VCS PrimeSim AMS Environment Options](#)

Specifying PrimeSim Environment Options

To specify the environment options, choose **Setup > Environment Options** from the PrimeWave Design Environment main menu bar.

The **Environment Options** dialog box opens.



The following options are available:

- **Include Path**

The include path specifies a collection of directories to be searched for any file that is specified with a relative path in the environment. The **Model Include Path** in the **Model Setup** dialog box overrides these path specifications for any files that are relatively specified.

Caution:

Relative Model Include Paths in the **Model Setup** dialog box (**Setup > Model Files**) override any specified include paths in the **Environment Options**.

- **Switch View List and Stop View List**

Note:

This option is not available when simulating a configuration.

These lists override the default switch and stop view lists for the simulator when the design is netlisted. If the design under test is a configuration, these lists are ignored.

- **Incremental Netlisting**

This option enables the netlister to skip designs that have not changed since the previous netlisting job.

- **Background Netlisting**

This option enables netlisting to be performed as a background process.

- **Include Structural Netlist**

Select this option to include a structural netlist inside the final netlist.

- **Ignore Estimated Parasitic Devices**

You can enable inclusion of parasitics in the netlist using this option. When a design with parasitics is simulated in the PrimeWave Design Environment, the parasitics defined in the Schematic Editor Estimated Parasitic assistant are included in the netlist or ignored based on the **Ignore Estimated Parasitic Devices** setting in the **Environment Options** dialog box in the PrimeWave Design Environment. This setting is localized to a testbench. In MTB sessions, you can specify different values for the **Ignore Estimated Parasitic Devices** setting for different testbenches. The PrimeWave Design Environment creates separate results directories to make sure netlists and simulation results with and without parasitics are managed efficiently.

- **Command Line Options**

Enter any **Command Line Options** that you want to append to the line that the tool uses to invoke the integrated simulator.

For example, when PrimeSim HSPICE is integrated with the PrimeWave Design Environment, you can enter `-mt` to enable the multi-threading capability. The `-mt` option is added to the following command that the tool uses to invoke PrimeSim HSPICE:

```
hspice -i input.spi -mt -o ../../results/input > ../../results/run.log 2>&1
```

Caution:

Settings in this field can conflict with PrimeWave Design Environment requirements.

- **Output Data Format**

Note:

Some simulators might support varying output data subsets. The `fsdb` format is available for mixed-signal simulations, for example.

The `wdf` option outputs data in `.wdf` format, which is a Synopsys binary waveform data file for analog that includes lossless compression. It also helps the waveform viewer

load and display large analog waveforms more quickly than other formats. This is the default for PrimeSim and FineSim.

The **fsdb** option outputs data in `.fsdb` format, which is a Synopsys binary waveform data file for mixed-signal and high performance digital-centric analog. This is the default for System-Verilog mixed-signal flows.

The **psf** option outputs data in `.psf` format (3rd-party), which is a binary data format that splits files into 2GB pieces when the file gets larger than 2GB. For Synopsys simulators, `.psf` support is "as-is" and primarily available for interoperability with other tools and flows.

- **Case Sensitive** (PrimeSim HSPICE simulations)

Click **Case Sensitive** to enable case sensitivity for all parts of a testbench.

- **Multi Threading/Multi Core** (PrimeSim HSPICE simulations)

Enter the number of threads you want to use during simulation in the **Number of Threads** text box (the default is 1 thread).

If you want to enable the PrimeSim HSPICE High Performance Parallel engine, check **HPP**.

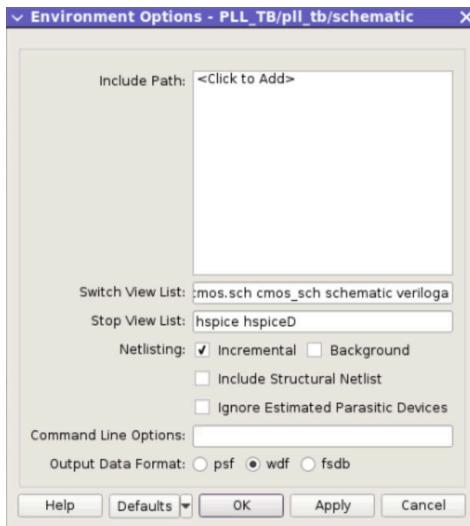
Specifying FineSim Environment Options

The **Environment Options** dialog box contains fields that are common to many simulators and you can customize this dialog box by adding simulator-specific fields.

To specify the **Environment Options** for FineSim integration:

1. Choose **Setup > Environment Options** from the PrimeWave Design Environment main menu bar.

The **Environment Options** dialog box opens.



2. (Optional) Enter **Include Path**.

FineSim supports the command-line argument `-inc` that allows you to specify a directory to be searched for device models and files specified without the absolute path. Multiple directories are supported with the `-inc` command-line argument.

The FineSim integration uses the command-line argument `-inc` for each path specified in the **Include Path** field.

Note:

FineSim supports Linux environment variables and the Linux home character (~).

3. The **Switch View List** and **Stop View List** fields contain design view names used to determine which view to netlist and whether a view is a stopping view. If the design is a config view, these are not available here but in the Hierarchy Editor, where you can select and override views to use. To learn more about switch lists and stop lists, see the *Custom Compiler Environment User Guide*.

4. Enable **Netlisting**.

The **Netlisting** options control how the netlister is run.

The **Netlisting** options are:

- **Incremental:** This option enables the netlister to skip the designs that have not changed since the previous netlisting job.
- **Background:** This option enables netlisting to be performed as a background process.

5. (Optional) Enable **Include Structural Netlist.**

Select this option to include a structural netlist inside the final netlist.

6. (Optional) Enable **Ignore Estimated Parasitic Devices.**

When a design with parasitics is simulated in the PrimeWave Design Environment, the parasitics are included in the netlist or ignored based on the **Ignore Estimated Parasitic Devices** setting in the **Environment Options** dialog box in the PrimeWave Design Environment. This setting is localized to a testbench. In MTB sessions, you can specify different values for the **Ignore Estimated Parasitic Devices** setting for different testbenches. The PrimeWave Design Environment creates separate results directories to make sure netlists and simulation results with and without parasitics are managed efficiently.

7. (Optional) Enter the **Command Line Options.**

This field gives you the ability to specify command-line options on the call to FineSim which are appended to the list that the PrimeWave Design Environment normally sends to the simulator.

8. Output Data Format

The **wdf** option (the default for FineSim) outputs data in **.wdf** format, which is a Synopsys binary waveform data file for analog that includes lossless compression. It also helps the waveform viewer load and display large analog waveforms more quickly than other formats.

The **fsdb** option outputs data in **.fsdb** format, which is a Synopsys binary waveform data file for mixed-signal and high performance digital-centric analog. This is the default for System-Verilog mixed-signal flows.

The **psf** option outputs data in **.psf** format (3rd-party), which is a binary data format that splits files into 2GB pieces when the file gets larger than 2GB. For Synopsys simulators, **.psf** support is "as-is" and primarily available for interoperability with other tools and flows.

9. Enable the **Enable Parallel Simulation group.**

Enabling the **Enable Parallel Simulation** group allows the FineSim simulation to run on multiple CPUs and machines.

The **Enable Parallel Simulation** group has the following options:

- **Total Number of Processes:** This field lets you specify the total number of processes used to run the simulation job. If you specify the value for this field, the command-line option **-np** is used in the simulation command.
- **Auto:** When **Auto** is enabled, the label of the **Total Number of Processes** field changes to **Max Number of Processes** and it becomes the maximum number

of processes used for running the simulation job. If **Auto** is enabled, FineSim automatically determines the number of processes to use, and the command-line option `-auto` is used in the simulation command.

Note:

Parallel simulation is not supported for AC or Noise analyses.

10. Click **OK**.

The environment options are now set up.

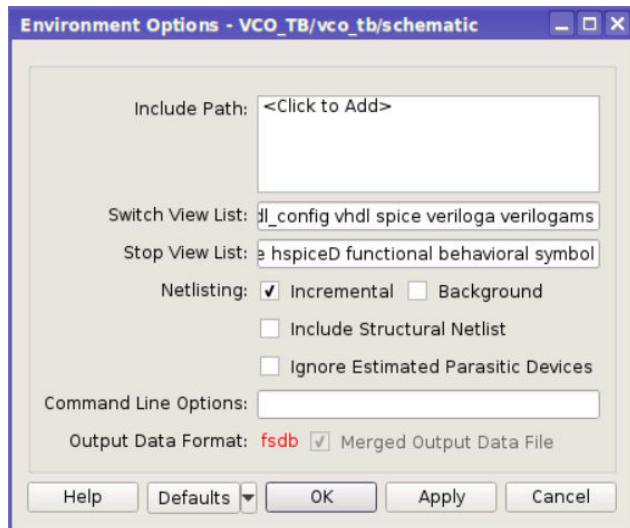
Specifying FineSim VCS Environment Options

The **Environment Options** dialog box contains fields that are common to many simulators and you can customize this dialog box by adding simulator-specific fields.

To specify the **Environment Options** for the FineSim VCS integration:

1. Choose **Setup > Environment Options** from the PrimeWave Design Environment main menu bar.

The **Environment Options** dialog box opens.



2. (Optional) Enter an **Include Path**.
3. The **Switch View List** and **Stop View List** fields contain design view names used to determine which view to netlist and whether a view is a stopping view. If the design is a config view, these are not available here but in the Hierarchy Editor, where you can select and override views to use. To learn more about switch lists and stop lists, see the *Custom Compiler Environment User Guide*.

4. Choose one of the following **Netlisting** options:
 - **Incremental:** This option enables the netlister to skip the designs that have not changed since the previous netlisting job.
 - **Background:** This option enables netlisting to be performed as a background process.

5. (Optional) Enable **Include Structural Netlist**.

Select this option to include a structural netlist inside the final netlist.

6. (Optional) Enable **Ignore Estimated Parasitic Devices**.

When a design with parasitics is simulated in the PrimeWave Design Environment, the parasitics are included in the netlist or ignored based on the **Ignore Estimated Parasitic Devices** setting in the **Environment Options** dialog box. This setting is localized to a testbench.

In MTB sessions, you can specify different values for the **Ignore Estimated Parasitic Devices** setting for different testbenches. The PrimeWave Design Environment creates separate results directories to make sure netlists and simulation results with and without parasitics are managed efficiently.

7. (Optional) Enter any command-line options for your mixed-signal simulation.

Note:

Many options are automatically added by the PrimeWave Design Environment environment based on setup and entries in other dialog boxes.

8. Click **OK** to save your settings.

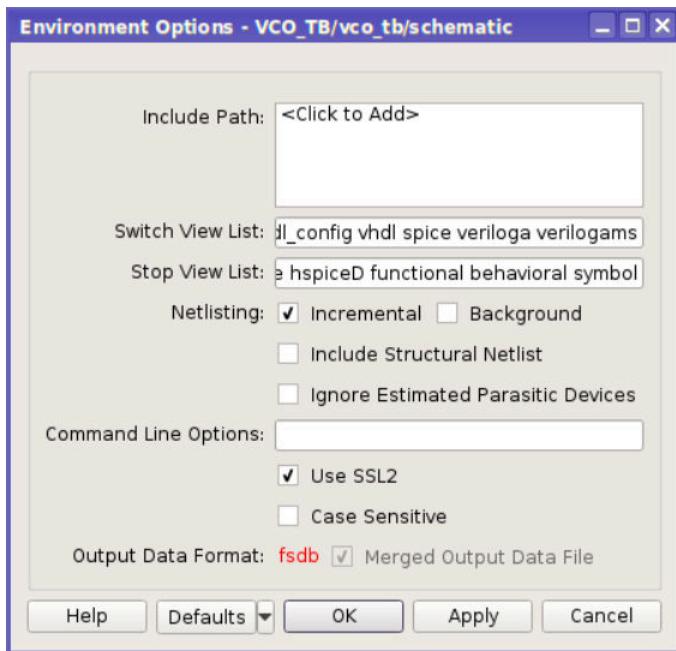
Specifying VCS PrimeSim AMS Environment Options

The **Environment Options** dialog box contains fields that are common to many simulators and you can customize this dialog box by adding simulator-specific fields.

To specify the **Environment Options** for the VCS PrimeSim AMS integration:

1. Choose **Setup > Environment Options** from the PrimeWave Design Environment main menu bar.

The **Environment Options** dialog box opens.



2. (Optional) Enter an **Include Path**.
3. The **Switch View List** and **Stop View List** fields contain design view names used to determine which view to netlist and whether a view is a stopping view. If the design is a config view, these are not available here but in the Hierarchy Editor, where you can select and override views to use. To learn more about switch lists and stop lists, see the *Custom Compiler Environment User Guide*.
4. Choose one of the following **Netlisting** options:
 - **Incremental**: This option enables the netlister to skip the designs that have not changed since the previous netlisting job.
 - **Background**: This option enables netlisting to be performed as a background process.
5. (Optional) Enable **Include Structural Netlist**.
Select this option to include a structural netlist inside the final netlist.
6. (Optional) Enable **Ignore Estimated Parasitic Devices**.
When a design with parasitics is simulated in the PrimeWave Design Environment, the parasitics are included in the netlist or ignored based on the **Ignore Estimated Parasitic Devices** setting in the **Environment Options** dialog box. This setting is localized to a testbench.

In MTB sessions, you can specify different values for the **Ignore Estimated Parasitic Devices** setting for different testbenches. The PrimeWave Design Environment creates separate results directories to make sure netlists and simulation results with and without parasitics are managed efficiently.

7. (Optional) Enter any command-line options for your mixed-signal simulation.

Note:

Many options are automatically added by the PrimeWave Design Environment based on setup and entries in other dialog boxes.

8. (Optional) Enable **Use SSL2**.

This option is on by default.

9. (Optional) Enable **Case Sensitive**.

For a multi-view cell, the Verilog module name's case and the SPICE subcircuit name's case must be identical. The cases for port names must be identical between SPICE and Verilog views. Also, be aware that every name in HSPICE netlists is treated as lowercase (by default). Because Verilog is case-sensitive, and the VCS tool processes everything as-is (by default), you must be aware of any possible case discrepancies between the two views.

10. Click **OK** to save your settings.

Specifying Model Files

Model files typically contain model definitions for a process library.

This section contains the following topics:

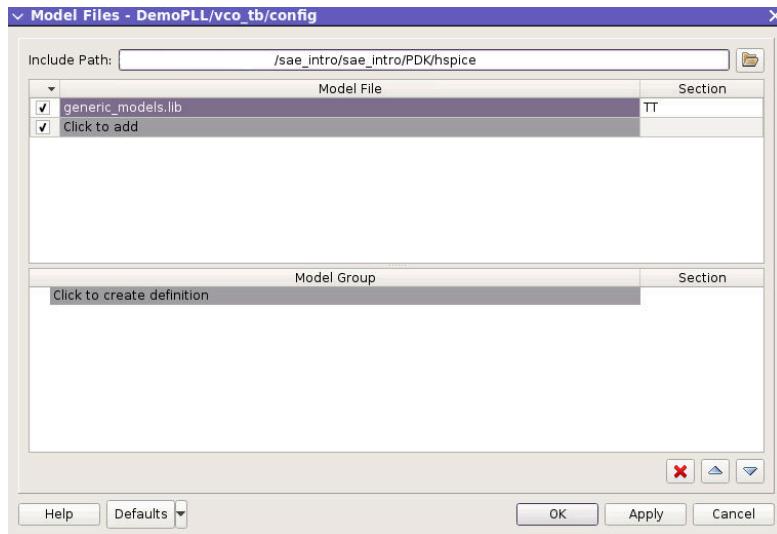
- [Setting Up Model Files](#)
- [Editing Model Files](#)
- [Reordering Model Files](#)
- [Removing Model Files](#)
- [Copying and Pasting Model Files](#)

Setting Up Model Files

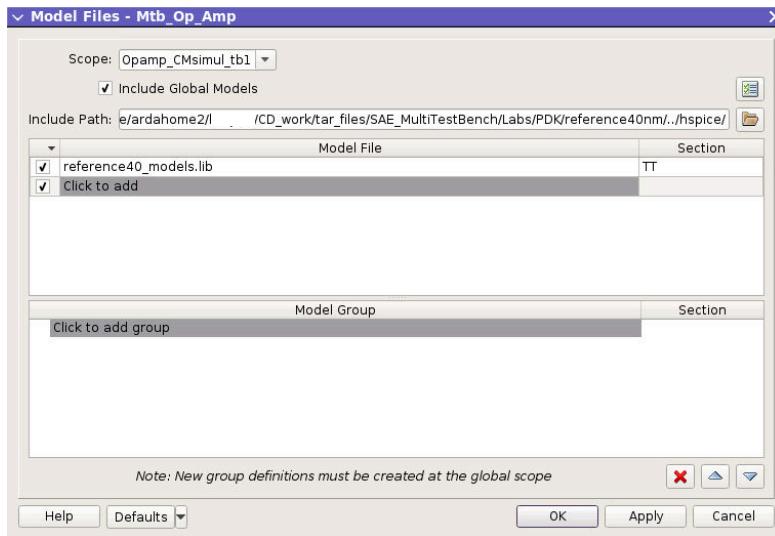
To set up model files:

1. Choose **Setup > Model Files** from the PrimeWave Design Environment main menu bar.

The **Model Files** dialog box opens.



2. Specify **Scope** if working in MTB mode. If you are working in a test suite with multiple testbenches, choose either **Global** or the name of a testbench. Model files are either included in all testbenches (globally) or are included in just the testbench you choose, respectively. Enable **Include Global Models** if you choose a single testbench but still want to include any global model file.



3. Enter an **Include Path** or click the folder icon to browse to a directory where your model files are located. Any files that you include with relative paths are assumed to be relative to the path that you specify.

Note:

Relative model file paths specified in the **Model Files** dialog box override model file Include Paths specified in the **Environment Options** dialog box (**Setup > Environment Options**).

4. In the **Model File** table, click the cell labeled "Click to add" and either enter the path of a model file or click the folder icon to browse to the location of a desired model file.

If the model file you select contains separate sections, such as .LIB sections in HSPICE, the sections are listed in a menu in the **Section** column for that row.

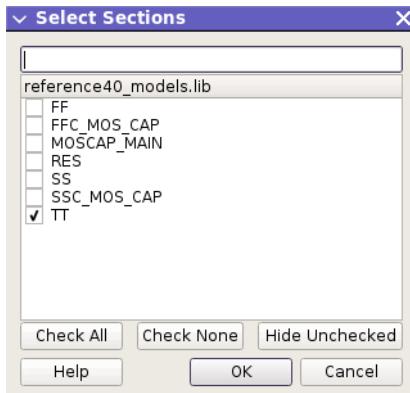
If the model file is not readable (the file does not exist or does not have read permission), a red highlight appears around the model file field. You cannot apply these changes until you fix the erroneous entry.

To edit a row, right-click in the row, then choose **Edit** from the menu to access a text editor for the selected model file.

5. If the model file you choose contains sections, ensure the section you want to use appears in the **Section** column. Once you start typing in a cell in the **Section** column, a menu shows matching section names available to select.

If multiple sections exist that need to be included in a single file, you have to include the file multiple times.

The **Section** column also contains the **Edit section name(s)** button, which opens the **Select Sections** dialog box. You can choose multiple sections. Clicking **OK** in this dialog box inserts a row in the Model File table for each chosen section name.

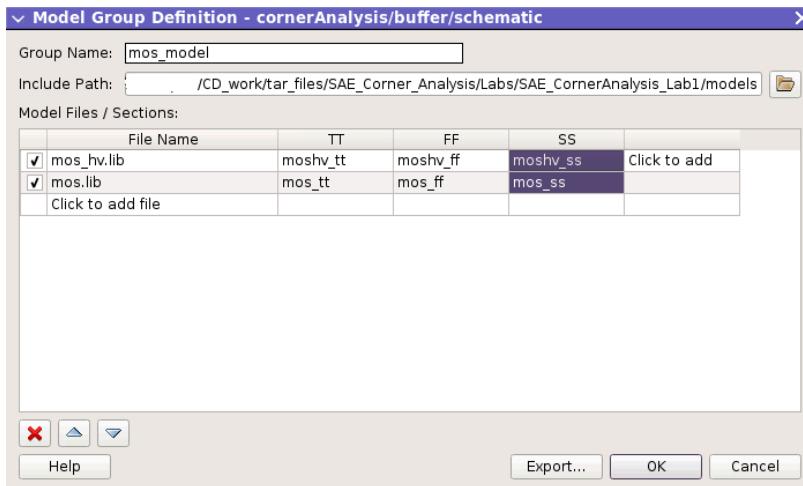


Note:

Only top-level .LIB statements are included in the list. Recursive .LIB statements (those found in included files) are not processed.

6. (Optional) Click the check box to the left of the model name to enable or disable the model. Models are enabled by default.
If you disable the model results, the model statement is not netlisted.
7. In the **Model Group** table, you can group multiple model files so that they can be changed all at once. For example, you might have model files for MOS, BJT, resistors, and capacitors each containing sections TT, FF, and SS. You can create a model group that moves each of these models files to the correct section with only one entry. Model files are often set up by the CAD or Library teams to simplify the setup for the PrimeWave Design Environment user. Once a model group is created, its behavior from the **Model Files** dialog box is virtually the same as a single model file.

Click the cell labeled "Click to add group" to create a global model group definition. In the **Model Group Definition** dialog box that opens, set the group name, the optional include path, and the model files to be used, then click OK. You can rename model group section columns by right-clicking the column header and choosing **Rename Section**.



Note:

Include paths from the **Model Files** dialog box and **Environment Options** dialog box are available.

Enter the sections to use together for each model file. Right-click on the default header name to open a menu, from which you can update the **Section** name if desired. Once a section is created, a new column is added to create a new section for this group.

If the model group definition is not readable (the group does not exist or does not have read permission), a red highlight appears around the model group field. You cannot apply these changes until you fix the erroneous entry.

To edit a row, right-click in the row, then choose **Edit Global Definition** from the menu to access a text editor for the selected model group.

8. Click **Apply** or **OK** to add the model file to your testbench.

Once files are entered in the table, you can edit them by right-clicking in a row that contains the file you want to edit, then choosing **Edit** from the menu that opens.

Editing Model Files

To edit model files that are already specified in the model file table (accessible from **Setup > Model Files**), right-click anywhere in a row that contains the file you want to edit, and choose **Edit** from the menu. The file opens in the text editor that is already specified.

If the file does not open, you might not have your text editor specified. See [Specifying a File Text Editor](#).

You can also use the **Edit** button in the Model Files dialog box to edit the model files.

Reordering Model Files

To reorder model files in the model file table (accessible from **Setup > Model Files**), click the name of the model file, then click the up or down arrows to change the position of the model file.

Removing Model Files

To remove model files from the model file table (accessible from **Setup > Model Files**), click the name of a model file, and then click the **X** button.

You can also right-click the name of a model file, and choose **Delete** from the menu.

Copying and Pasting Model Files

To copy model files that are already specified in the model file table (accessible from **Setup > Model Files**), right-click anywhere in a row that contains the file you want to copy, and choose **Copy** from the menu.

If you now right-click and choose **Paste** from the menu, the copied model file gets added to the bottom of the currently selected **Model Files** table.

Specifying Parasitic Back-Annotation Flow

You can provide the required parasitic netlist file(s) (SPF/DSPF or SPEF), the active nets file for selective net extraction and back-annotation, and also set up the required simulator options for post-layout analysis simulation jobs.

This section contains information on the following topics:

- [Specifying Parasitic Back-Annotation Flow for PrimeSim HSPICE](#)
- [Specifying Parasitic Back-Annotation Flow for FineSim](#)

Specifying Parasitic Back-Annotation Flow for PrimeSim HSPICE

PrimeSim HSPICE back annotation supports Full Back-Annotation and Selective Net Back-Annotation for the SPF and SPEF formats.

In PrimeSim HSPICE, the post-layout simulation is similar to pre-layout simulation. You can do the post-layout simulation with DSPF only if it is a fully extracted netlist with instances, and you can include the DSPF file in the pre-layout netlist.

To set up the parasitic back-annotation flow for PrimeSim HSPICE:

1. Choose **Setup > Back Annotation** from the PrimeWave Design Environment main menu bar.

The **Parasitic Back Annotate** dialog box opens. The top portion of the dialog box has a table, while the **Active Nets** section is in the lower portion. The table displays the list of parasitic files/cellviews to back-annotate, and the **Active Nets** section provides the user interface to set up the active nets file.

2. Specify the values for the fields in the top table as shown:

Field Name	Description
Enabled	Select all rows on which parasitic back-annotation must be performed.
Type	The supported types are Cellview , File , and GPD . Cellview is a SPICE Text view containing a parasitic-extracted SPICE Text file. Type File lets you pick the parasitic file to be back-annotated that is saved in the disk. Type GPD supports the <code>ba_gpd</code> option in the HSPICE netlist and the <code>finesim_gpd</code> option in the FineSim netlist.
Back Annotate	Defines the path for the spf, dspf, or spef file. You can use an available file. For File Type, you get an icon next to this field. You can click the icon to view the SPF/DSPF files in a text viewer.

3. Click **Options**. The **Global Simulation Options** dialog box opens and the **Parasitic Back Annotation** category is visible. You can now set the options. They are applied to all the parasitic files. For more information on the simulation options, see the *PrimeSim HSPICE Reference Manual: Commands and Control Options*.
4. (Optional) The user interface has the following buttons:
 - Use the **Delete**, **Up**, or **Down** buttons to remove or move table rows.
 - Click the **Copy** button to duplicate the selected row in the table. You cannot duplicate more than one row at a time.
5. Enable **Active Nets** to enable the selective net back-annotation flow.
6. Choose **Mode**.

Choose **write** mode for generating the active nets files, or choose **read** mode to use the active nets file in the selective net back-annotation.

- For **write** mode:

Specify the values for the fields as shown in the table below to set up the active nets analysis.

Field Name	Description
Threshold	Defines the threshold of voltage variation. A node is considered to be active when the voltage change, compared to the initial value of the node is larger than <code>val</code> (default of <code>val</code> is 0.1 V).
Level	Defines the number of hierarchical depth levels when checking node activity (default is 0, which means all levels).
Start Time	Defines the start time of the activity check.
Stop Time	Defines the stop time of the activity check.
Include Nets	Defines the nets to be included in the activity check.
Exclude Nets	Defines the nets to be excluded from the activity check.

In the **Analysis Output** section, specify the values for the fields as shown in the table below to set up the active nets analysis output.

Field Name	Description
Output Format	Defines the output format of the active nets file.
File Name	The name of the active nets file. In HSPICE this is a read only field, therefore you cannot change the file name. Click the icon next to the file name to open the active nets file in a text viewer. This way, you do not need to look at the active nets file in the netlist directory always.
Save In	You can choose Netlist Directory to generate the active nets file in the netlist directory, or choose Specify Directory and specify a directory of your choice.
Save Directory	Defines the output directory for the active nets file. It is enabled only when Save In is set to Specify Directory .

- For **read** mode:

Specify the values for the fields as shown in the table below for reading the active nets files.

Field Name	Description
Enabled	Click the "Click to Add" text to add a new row in the table.
Active Nets File	The active nets file name is automatically determined based on the output format and the directory in the write mode.

7. Click **Apply** or **OK**.

The parasitic back-annotation flow options are now set up.

When working with PrimeSim HSPICE simulations for SPF_net devices, you can simulate using only net parasitics for the placed SPF_net devices. **Ignore Estimated Parasitic Devices** must be off for this flow. You can add a new element to the parasitic library called "SPF_net" and attach that element to specific nets to reference an SPF file. Then, when you run a back-annotation flow (**Setup > Back Annotation**), the SPF file is included and uses the parasitic information specified in the SPF_net element.

Specifying Parasitic Back-Annotation Flow for FineSim

To set up the parasitic back-annotation flow for FineSim:

1. Choose **Setup > Back Annotation** from the PrimeWave Design Environment main menu bar.

The **Parasitic Back Annotate** dialog box opens. The top portion of the dialog box has a table, while the **Active Nets** section is in the lower portion. The table displays the list of parasitic files/cellviews to back-annotate, and the **Active Nets** section provides the user interface to set up the active nets file.

2. Specify the values for the fields in the top table as shown:

Field Name	Description
Enabled	Select all rows on which parasitic back-annotation must be performed.
Type	The two supported types are Cellview and File . Cellview is a SPICE Text view containing a parasitic-extracted SPICE Text file. File lets you pick the parasitic file to be back-annotated that is saved in the disk.
Back Annotate	Defines the path for the SPF, DSPF, or SPEF file. You can use an available file. For File Type, you get an icon next to this field. You can click the icon to view the SPF/DSPF files in a text viewer.
Skip Nets	Defines the nets that should not be back-annotated.

Field Name	Description
Min R	Defines the minimum resistor value to keep. (Default is 0.01 ohm.)
Min FC	Defines the minimum floating capacitor value to keep (default is 0).
Max R	Defines the upper threshold of resistors (default is 1e12).
RC Nets	These nets are used for full RC back-annotation.
C Nets	These nets use lumped capacitance back-annotation.
CC Nets	These nets use Cg+Cc back-annotation.
Instance Section	Defines whether to back-annotate the DPF section (default is 1).
Other Options	You can add extra simulator options that are not in the table.
Note:	
The syntax used should match exactly with what the simulator expects (including using net, subcircuit, and instance names in the netlist namespace), as the PrimeWave Design Environment does not check for errors in the content of this field.	

Note:

By default, **Skip Nets** and **Min R** columns are shown in the table. Right-click in the column header of the table and choose the **Show/Hide Columns** menu option to select the columns to display.

For FineSim, each file can have a different set of options. The options after **Back Annotate** can be specified for an individual file. The table below shows the mapping of column names with the FineSim options. Tooltips are also available.

Column Name	FineSim Option
Skip Nets	finesim_spfnet
Min R	finesim_spfrmin
Min FC	finesim_spffcmin
Max R	finesim_spfrmax
RC Nets	finesim_spfcnet
C Nets	finesim_spfcnet
CC Nets	finesim_spffcnet

Column Name	FineSim Option
Instance Section	finesim_spfinst

3. Click **Options**. The **FineSim Global Simulation Options** dialog box opens and the **Parasitic Back Annotation** category is visible. You can now set the options. The options are applied to all the parasitic files. For more information on the simulation options, see the *FineSim User Guide: Pro and SPICE Reference*.
4. (Optional) The user interface has the following buttons:
 - Use the **Delete**, **Up**, or **Down** buttons to remove or move table rows.
 - Click the **Copy** button to duplicate the selected row in the table. You cannot duplicate more than one row at a time.
5. Enable **Active Nets** to enable the selective net back-annotation flow.
6. Choose **Mode**.

Choose **write** mode for generating the active nets files, choose **auto** mode for generating the active nets file and then running the post-layout simulation using the active nets file in the same run, or choose **read** mode to use the active nets file in the selective net back-annotation.

- For **write** mode:

The table below shows how to set up the active nets analysis.

Field Name	Description
Threshold	Defines the threshold of voltage variation. A node is considered to be active when the voltage change, compared to the initial value of the node is larger than <code>val</code> (default of <code>val</code> is 0.1 V).
Reuse Active Nets File	Reuse the existing active nets file.
Start Time	Defines the start time of the activity check.
Stop Time	Defines the stop time of the activity check.
Exclude Nets	Defines the nets to be excluded from the activity check.

In the Analysis Output group box, set up the active nets analysis output as shown in the table below.

Field Name	Description
File Name	Defines the name of the active nets file.
Save In	You can choose Netlist Directory to generate the active nets file in the netlist directory, or choose Specify Directory and specify a directory of your choice.
Save Directory	Defines the output directory for the active nets file. It is enabled only when Save In is set to Specify Directory .

- For **auto** mode:

The table below shows how to set up the active nets analysis.

Field Name	Description
Threshold	Defines the threshold of voltage variation. A node is considered to be active when the voltage change, compared to the initial value of the node is larger than <code>val</code> (default of <code>val</code> is 0.1 V).
Reuse Active Nets File	Reuse the existing active nets file. Disabling regenerates the active nets file, and is equivalent to "force" mode in FineSim.
Start Time	Defines the start time of the activity check.
Stop Time	Defines the stop time of the activity check.
Exclude Nets	Defines the nets to be excluded from the activity check.

In the **Analysis Output** group box, set up the active nets analysis output as shown in the table below.

Field Name	Description
File Name	Defines the name of the active nets file.
Save In	You can choose Netlist Directory to generate the active nets file in the netlist directory, or choose Specify Directory and specify a directory of your choice.
Save Directory	Defines the output directory for the active nets file. It is enabled only when Save In is set to Specify Directory .

- For **read** mode:

Specify the values for the fields as shown in the table below for reading the active nets files.

Field Name	Description
Enabled	Click the "Click to Add" text to add a new row in the table.
Active Nets File	The active nets file name is automatically determined based on the output format and the directory in the write mode.

7. Click **Apply** or **OK**.

The parasitic back-annotation flow options for FineSim are now set up.

Sequential Testbenches

In MTB mode, you can run testbenches in a sequential order of your choice to get a calculated measurement from the first testbench and then use that calculated value to set a design variable in subsequent simulations.

The PrimeWave Design Environment matches the parameter values in the destination testbench with parameter values in the source testbench to find the value of referenced output from source testbench.

To run sequential testbenches:

1. In MTB mode, set up two or more testbenches. One of these testbenches must use the `getVal()` function to retrieve a calculated measurement from the first testbench. (See [getVal\(\) Syntax](#).)

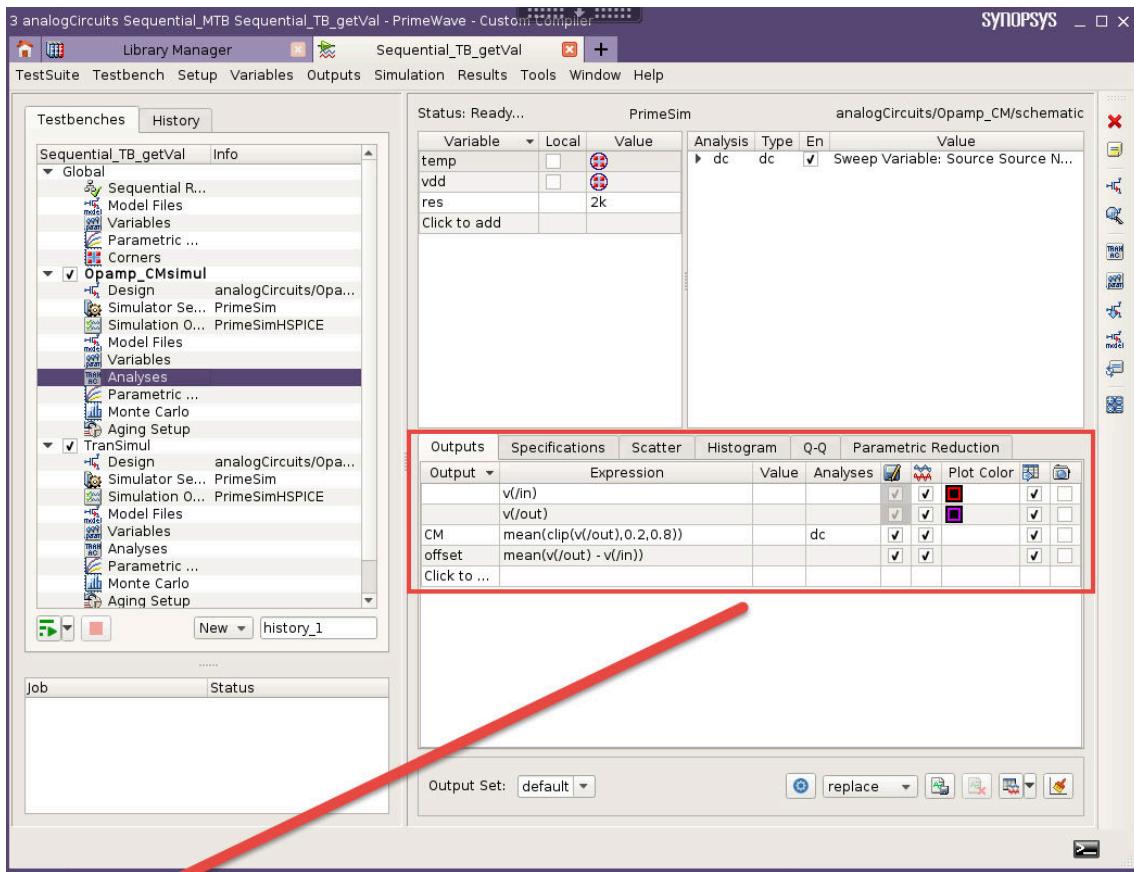
In the example shown below, the first (source) testbench, opamp_CMSimul, includes output expressions in the **Outputs** table for CM and offset.

Note:

Some of the features shown below are limited availability. For information about these features, refer to SolvNetPlus article #000036534 "[How to Enable the PrimeWave Design Environment Flow-Based Interface](#)" or consult your Synopsys representative.

Chapter 2: Setting Up Testbenches

Sequential Testbenches

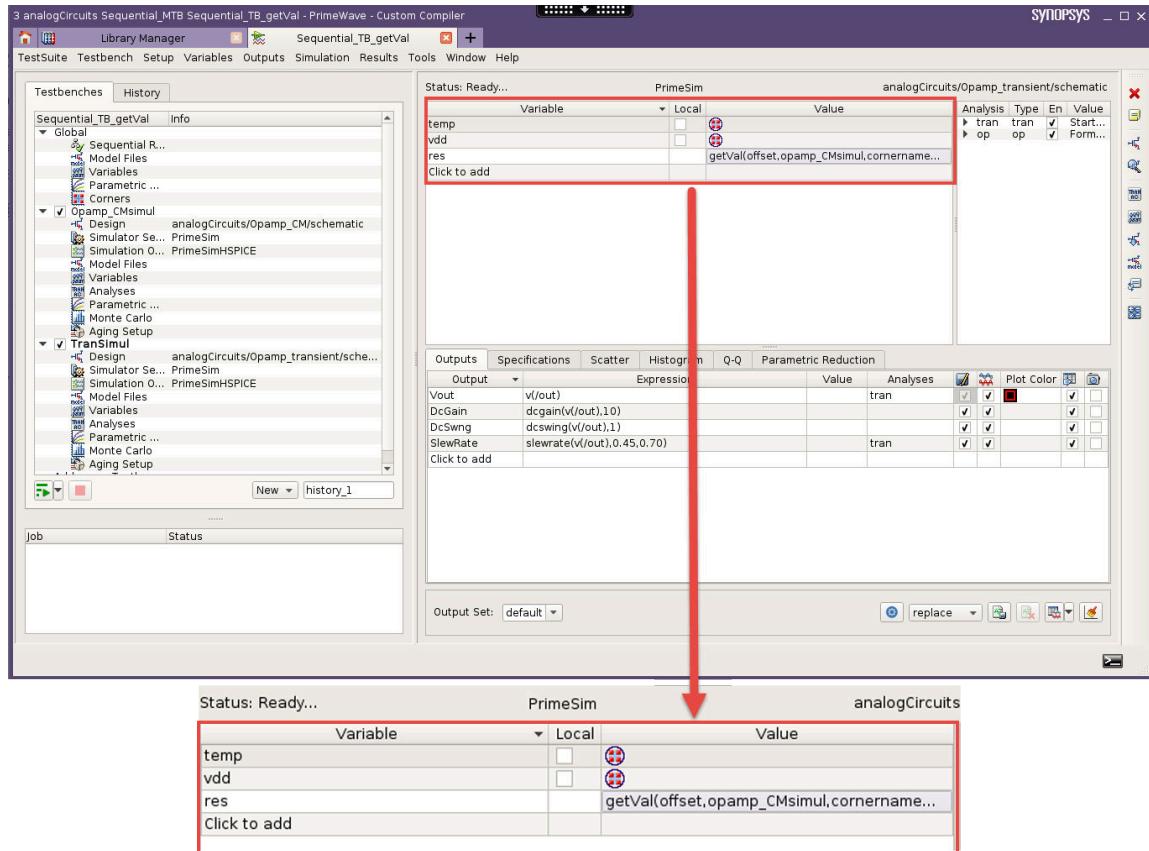


Outputs	Specifications	Scatter	Histogram	Q-Q	Parametric Reduction
Output	Expression	Value	Analyses	Plot Color	
v(/in)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
v(/out)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
CM	mean(clip(v(/out), 0.2, 0.8))	dc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
offset	mean(v(/out) - v(/in))		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Click to ...					

The second (destination) testbench, TransSimul, uses the `getVal()` function to calculate the value of the parameter `res` based on the values of `offset` from source testbench: `getVal(offset, opamp_CMsimul)`.

Chapter 2: Setting Up Testbenches

Sequential Testbenches

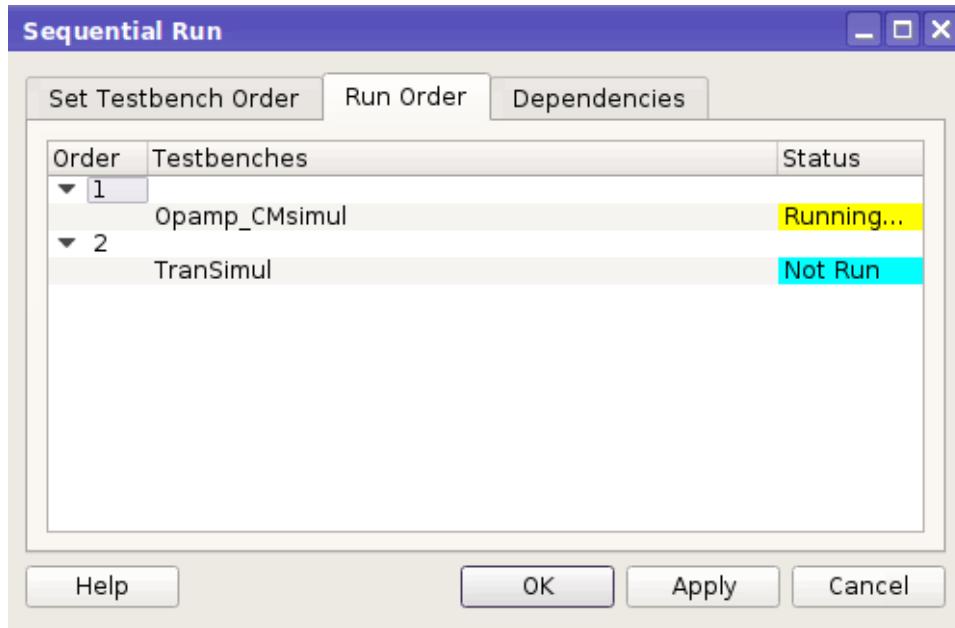


2. Double-click **Sequential Testbenches** in the testbenches tree under **Global** to review the sequence before running the simulation.
3. Choose **Simulation > Netlist and Run** to run the simulation. The Job Monitor opens, where you can see the testbenches running in the order you set.

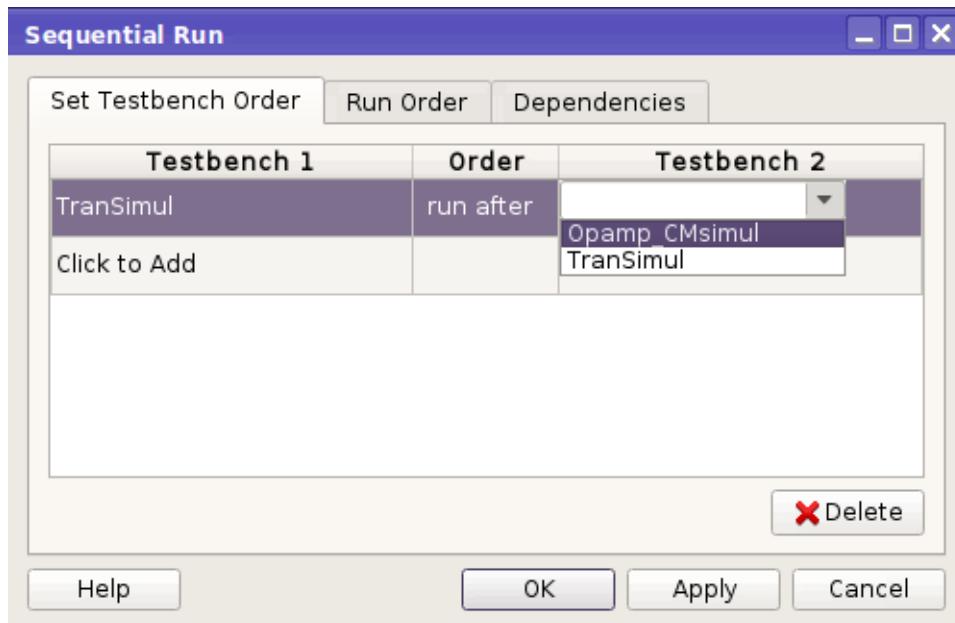
Because you are running a simulation with `getVal()` references, the **Sequential Run** dialog box automatically opens during the simulation, with the **Run Order** tab forward.

Alternatively, open the **Sequential Run** dialog box before running the simulation by selecting the Global testbench and choosing **Setup > Sequential Run**.

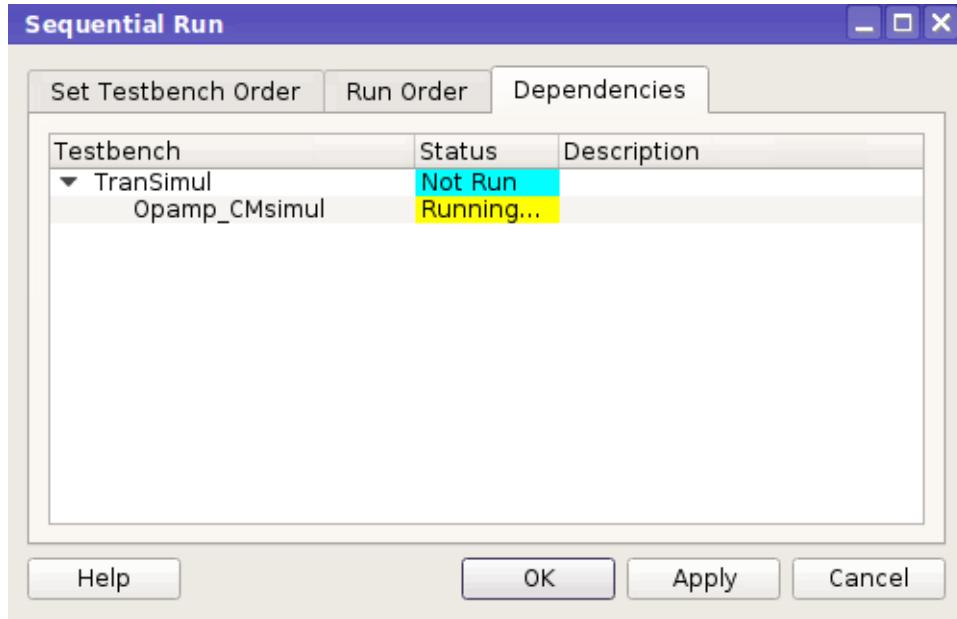
On the **Run Order** tab, you can view the **Status** of each testbench.



4. Click the **Set Testbench Order** tab. To change the order of testbenches, click in a table cell in the **Testbench1** or **Testbench2** column to select a new testbench order, as shown below.



5. Click the **Dependencies** tab to view the dependencies between testbenches. The testbench at the root of the dependency tree depends on the testbenches under its tree.



6. When the simulation is complete, open the ResultsView.

Chapter 2: Setting Up Testbenches

Sequential Testbenches

In the ResultsView, the Measurements table for the destination testbench (TranSimul) shows the calculated value for `res` based on the value of `offset` from the source testbench (Opamp_CMsimul).

Corner	offset dc	v(/in) dc	v(/out) dc	CM dc
Nominal	7.02654m	500.485m		
Corner1	-40.7269m	502.679m		
Corner2	1.59511m	500.117m		
Corner3	-33.6031m	502.037m		
Corner4	17.2606m	502.265m		

Corner	res dc	DcGain tran	DcSwing tran	SlewRate tran
Nominal	7.02654m	-241.013k	6.36321u	11.2858M
Corner1	-40.7269m	58.4035M	4.72912m	123.155k
Corner2	1.59511m	24.7832M	423.04m	164.249M
Corner3	-33.6031m	96.1113M	8.4419m	355.063k
Corner4	17.2606m	18.3827M	770.524u	20.5553M

getVal() Syntax

The `getVal()` function provides a way to access a value from one testbench and pass it to another testbench. The `getVal()` function can also pass a value to an output expression in a dependent testbench. This function is useful in simulations where you need to pass a calibrated value from one block in the design to another. In MTB mode, you can pass the values of an output/expression from one testbench to another, in a sequential manner, by controlling the dependencies.

Syntax:

```
getVal(output_name, testbench_name [, historyPoint=historyPoint_name,
cornerName=corner_name])
```

`output_name`

The name of the testbench output expression from which to get the value.

`testbench_name`

The name of the testbench from which to get the value.

`historyPoint` (Optional)

The name of the history point from which to get the value.

`cornerName` (Optional)

The name of the corner from which to get the value.

Saving and Plotting Terminal Node Voltages

To save and plot terminal node voltages in the parasitic back-annotation flow during post-layout simulation, you can do the following in the PrimeWave Design Environment Results Analyzer for the PrimeSim XA integration:

1. Build voltage measurements using the terminal names in the Results Analyzer.
2. Write probe statements for saving the terminal node voltages in the final netlist.
3. Plot MOSFET terminal node voltages from the Results Analyzer.

Saving Terminal Node Voltages

To save terminal node voltages in PrimeSim XA simulations:

1. Choose **Outputs > Save Options** from the PrimeWave Design Environment main menu bar.

The **Save Options** dialog box opens.

2. Choose **Terminal Node Voltages**.

You can choose the following options:

- **all** - All terminal node voltages of the MOSFET devices are saved.
- **level** - Specify the value for **Level**. All terminal node voltages up to the specified level are saved.
- **selected** - Only the selected terminal node voltages which were built from the Results Analyzer and added to the outputs table are saved.

3. Click **OK**.
-

Plotting Terminal Node Voltages

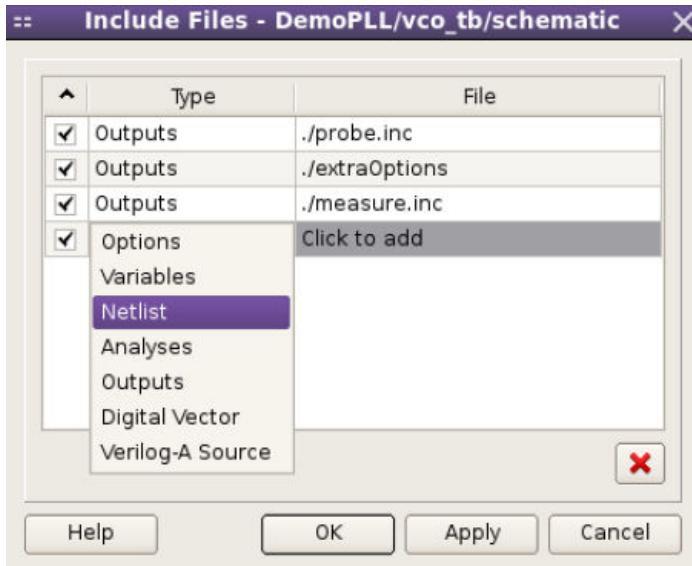
You can plot the saved terminal node voltages from the Results Analyzer. For more information on Results Analyzer, see [Using the Results Analyzer](#). Alternatively, right-click the pane under the Outputs tab in the PrimeWave Design Environment window to invoke the menu. You can plot the terminal node voltages by choosing the Plot menu item, if the expression represents terminal node voltage in the outputs table.

Specifying Include Files

To specify an include file:

1. Choose **Setup > Include Files** from the PrimeWave Design Environment main menu bar.

The **Include Files** dialog box opens.



- Choose the type of include file you want to add from the menu in the **Type** column.

Specifying the type of file helps to determine its place in the final netlist. File types include: **Options**, **Variables**, **Netlist**, **Analyses**, **Outputs**, **Libraries**, **Digital Vector**, and **Verilog-A Source**. See [Example 1](#) for a sample netlist showing include file locations.

- Click the "Click to add" text, and either enter the path of an include file or click the folder icon to browse to the location of a desired include file.

You can also copy or paste an include file. Right-click the include file name, and choose **Copy** or **Paste** from the menu that opens.

If the include file is not readable (the file does not exist or does not have read permission), the include file field is flagged as invalid. You cannot apply your changes until you fix the erroneous entry.

Note:

You can specify include files that contain .MEASURE statements.

Measurements included in this way are automatically added to the outputs section at the end of a simulation.

- (Optional) Click the check box to the left of the include file name to disable the file. Include files are enabled by default.

If you disable the included file, it is not netlisted.

- Click **OK** to save your settings.

The following example indicates the location of the different types of include files (indicated in purple) in the final netlist.

Example 1 Location of Include Files in Final Netlist

```

1 * Generated for: HSPICE
2 * Design library name: op_back_annotation
3 * Design cell name: bias2_tb
4 * Design view name: schematic
5
6
7 .include 'variables.inc'
8
9 .option PARHIER = LOCAL
10
11 .include 'options.inc'
12
13 .option ARTIST=2 PSF=2
14 .temp 25
15 .include 'mac_mod.inc'
16 .include 'reference_models.inc'
17
18 .inc 'libraries.inc'
19 .vec 'digitalVector..inc'
20 .hdl 'verilogA.inc'
21
22
23 .GLOBAL gnd! vdd! vss!
24
25 ** Testbench **
26 v9 gnd! vss! dc=0
27 vhilo net15 vss! dc=0
28 vibias net13 vss! dc=1.2
29 v1 vdd! vss! dc=1.2
30 xi0 net15 net13 bias2
31
32
33 .include 'netlist.inc'
34
35
36 .op A11 0
37 .tran 0.5u lu start=0
38 .include 'analyses.inc'
39 .option opfile=1 split_dp=1
40 .probe tran v(*)
41 .probe tran v(xi0.net186) v(net13) v(net15)
42
43
44 .include 'outputs.inc'
45
46
47

```

```
48
49 .end
```

Editing Include Files

To edit include files that are already specified in the include file table (accessible from **Setup > Include Files**), right-click anywhere in a row that contains the file you want to edit, and choose **Edit** from the menu. The file opens in the text editor that is already specified.

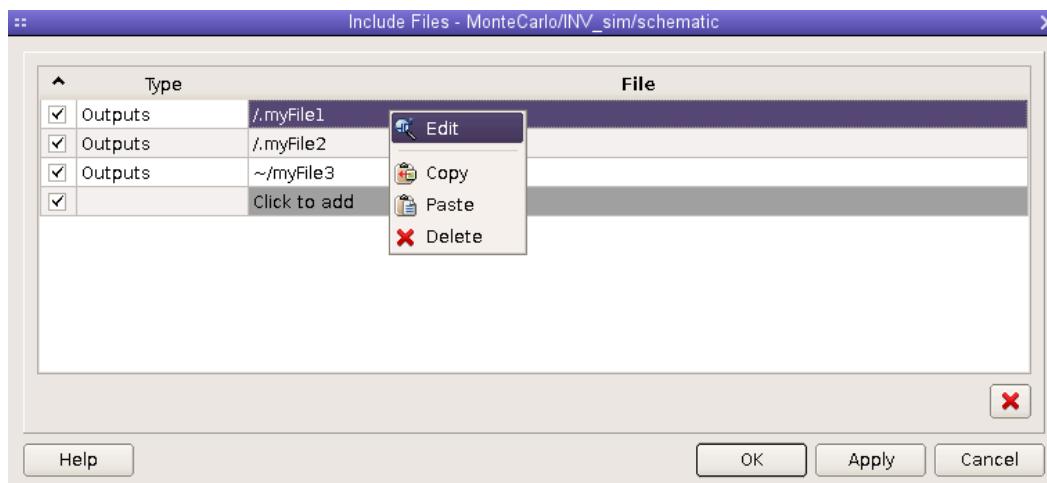
Note:

If the file does not open, you might not have your text editor specified. See [Specifying a File Text Editor](#).

Creating Include Files from the Edit Menu

To create an include file using the edit include file functionality:

1. Choose **Setup > Include Files** from the PrimeWave Design Environment main menu bar.
2. The **Include Files** dialog box opens.
3. Click in an empty **File** row and type in a desired file name, such as `./myFile1`. (See the following figure.)
4. Right-click your file name and choose **Edit** from the menu. The file opens in the text editor that is already specified.



Note:

If the file does not open, you might not have your text editor specified. See [Specifying a File Text Editor](#).

5. Edit and save the include file in the external editor as desired.



The screenshot shows a terminal window titled 'vi' containing the following text:

```
*This is a test for myFile1
.probe v(1) v(2)
```

The status bar at the bottom left of the window displays the text "— INSERT —".

6. Click **OK** in the **Include Files** dialog box to include the new file(s). Notice in the Text Viewer window that the files have been included.

Removing Include Files

To remove include files from the include file table (accessible from **Setup > Include Files**), click the name of an include file, and then click the **X** button.

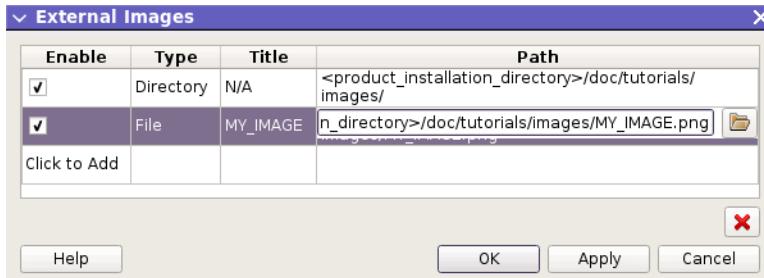
You can also right-click the name of an include file, and choose **Delete** from the menu.

Specifying External Images

You can add external images (of file type .png, .jpg, or .bmp) to simulation results for later viewing in the ResultsView. If you choose to export results to an HTML file, these external images can be included in the HTML report.

To add external images to results:

1. Choose **Setup > External Images**. The **External Images** dialog box opens.



2. Click **Click to Add** to add an external image. The image is enabled by default.
3. Select a **Type**. Select **File** for a single image file or **Directory** for a directory of image files.
4. Specify a **Title** for single image files.
5. Browse to the **Path** of the image file or directory.
6. Click **OK** to add the external images to the output results.

All external images appear in the ResultsView in the **Misc. Images** section of the view tree.

Specifying Digital Timing

Note:

Digital timing can only be specified for the VCS AMS and FineSim VCS integration.

To specify the digital timing options, which are used to control timing in a digital or mixed-signal simulation:

1. Choose **Setup > Digital Timing** from the Custom Compiler PrimeWave Design Environment menu.
- The **Digital Timing** dialog box opens.
2. Click the "Click to Add" text, and enter the path to a Standard Delay Format (SDF) file.
3. Click in the **Timing Condition** column, and choose either **min**, **typical**, or **max**.

4. Click in the **Instance** column, and enter an instance name or click the Select Instance button to select an instance in the schematic.
5. Click **OK** to save your changes.

Specifying Convergence Aids

To specify convergence aids, such as nodesets and initial conditions:

1. Choose **Setup > Convergence** from the PrimeWave Design Environment main menu bar, or press the **C** key.

The **Convergence Aids** dialog box opens.

2. Enter a value for the **Default Voltage**.

This voltage value is used for your nodeset or initial condition unless you override the value.

3. (Optional) Uncheck the **Initial Condition** or **Node Set** options to filter out that type of node from the list of node **Types** that are displayed.

These filters aid in the display and management of different types of convergence aids. Hidden convergence aids are still sent to the netlist. All convergence aids are displayed by default.

If you do not see a convergence aid in the table that you expect to see, ensure the **Filter** options are selected so that all types are displayed.

4. Choose the **Type** of convergence aid you are adding.

Initial Condition and **Node Set** are available.

5. Specify one or more nets.

To select nets from the Schematic Editor, click in the **Node** column of the convergence aid you are adding. Click **Select in Design**, which appears at the end of the **Value** column entry. Next, click one or more wires in the schematic canvas. As you click each net, the name of the net is added to the table of convergence aids. Once you select all needed nets, press Esc to close the Schematic Editor.

You can also add a net manually by clicking the "Click to add" text and entering the full hierarchical name of the net.

6. Enter a value for the voltage if you want to use a value that is different from the default.

7. (Optional) Click the check box to the left of the convergence aid **Type** column to enable or disable the convergence aid. Convergence aids are enabled by default. Disabled convergence aids are not netlisted.
 8. Click **OK** once all convergence aids are specified.
-

Adding Stimulus

To add stimulus to your testbench, choose **Setup > Stimulus Generator**. The **Stimulus Generator** dialog box opens.

Using the stimulus generator, you can specify instances for your testbench that are included in the netlist, but not be added to the designs as components.

This section contains information on the following topics:

- [Adding Instances](#)
 - [Editing Instances](#)
 - [Copying Instances](#)
 - [Deleting Instances](#)
 - [Creating, Renaming, and Deleting Stimulus Sets](#)
 - [Previewing Netlists](#)
-

Adding Instances

Note:

You must click **Enable** before you can add instances.

To add an instance:

1. Click **Add** in the **Stimulus Generator** dialog box.

The **Add Instance** dialog box opens.

2. Choose the desired **Library**, **Cell**, and **View**.
3. (Optional) Change the name of the instance.

By default, instances are named sequentially using the `I__<number>` format.

4. Enter parameter values as necessary.

If you do not see the parameters table, click the arrow next to the **Parameters** title.

5. Click **OK**.

The instance is added to the last row of the instances table.

Editing Instances

To edit an instance:

1. Click the name of an instance you want to edit in the **Stimulus Generator** dialog box.

The instance is highlighted in the instance table.

2. Click the **Edit** button in the **Stimulus Generator** dialog box.

The **Edit Instance** dialog box opens.

3. Make changes to the parameters as necessary.

You can also edit the name or any of the terms directly in the instances table by clicking in a table cell and changing values.

4. Click **OK** to save your changes.
-

Copying Instances

To copy an instance, click in the row of the instance you want copy, then click the **Copy** button. The instance is copied to a new row in the table after the last instance.

Deleting Instances

To delete an instance, click in the row of the instance you want to delete, and click the **Delete** button. The instance is removed from the table.

Creating, Renaming, and Deleting Stimulus Sets

By default, any instance you create are added to the "default" stimulus set. To save your instances to your own stimulus set, click the **New Stimulus Set** button, and enter a name for the stimulus set.

To rename a stimulus set, select it and click the **Rename Stimulus Set** button. A dialog box opens and asks you to enter a new name. Click **OK** to save the new name.

To delete a stimulus set, select it and click the **Delete Stimulus Set** button. A dialog box opens and asks you to confirm deletion. You cannot delete the default stimulus set; you have the option to clear it instead.

Previewing Netlists

To preview a netlist that is created from the instances you create, click the **Preview** button in the **Stimulus Generator** dialog box. The Text Editor opens with your netlist. When you netlist your design, this file is included in your final netlist using the "include" language of your simulator.

Specifying Temperature

The temp variable is displayed in the first row of the Variables pane in the PrimeWave Design Environment main window. To specify the temperature for the design, click in the cell to the right of **temp** and enter a value. The unit of measure can be chosen by right-clicking in the **temp** row and choosing **Temperature Units** from the menu that opens. Celsius is the default unit of measure.

Note:

The temperature is not a design variable object, and therefore cannot be deleted.

Expressions can also be entered as the temperature value. Evaluated expressions are always assumed to be in Celsius units.

Note:

Some simulators support multiple values and ranges for the temperature, but the PrimeWave Design Environment results postprocessing currently does not.

Working with History Points

In both single-testbench (STB) and multiple-testbench (MTB) modes, the history mechanism allows you to restore the in-memory configuration for the entire testbench back to any previous history point. In addition, this mechanism provides the individual testbenches' histories.

Note:

In MTB mode, the History pane is enabled by default.

The History pane displays the information in a tabular format with three columns: the **Name** of each history point, the **Index**, and the approximate **Size** of the results directory.

Each row in the table represents an on-disc history point of the STB or MTB. The **Name** column specifies the names of the history points. The **Index** column specifies an unique index number for the history point, which keeps increasing as the number of history points increases.

If you hold the pointer over the index field, a tooltip displays the index number from which the history point was derived. This is usually the previous run, but if you load an old history point and run another simulation, the index number is based off that index point.

Name	Lock	Size
history_5		32.94
history_4		34.09
history_4.2		0.82 M
history_4.1		...
history_3		6.39 M
history_2		33.47
history_2.2		0.81 M
history_2.1		...
history_1		6.96 M

Caution:

The history pane does not provide any mechanism to automatically undo design edits. Design edits that are performed in the design editor window are not bound to the testbench or test suite and are not captured in the history mechanism.

See also [Saving Multiple Testbench History](#).

This section contains information on the following topics:

- [Saving History Points](#)
- [Loading History Points](#)
- [Viewing Results of a History Point](#)
- [Renaming History Points](#)
- [Locking History Points](#)
- [Unlocking History Points](#)
- [Deleting History Points](#)
- [Deleting Results of a History Point](#)

Saving History Points

You can save a history point by clicking the **Netlist and Run** button on the PrimeWave Design Environment window. This creates a new history point, which is visible in the History pane under the name **Run Simulation**.

In MTB mode, you can also save a history point either by running a simulation using **Simulation > Netlist and Run** or by clicking the save icon to perform an explicit MTB save.

Loading History Points

To load a history point:

1. Select a row in the History pane.
2. Right-click and choose **Load History Point** option from the menu.

Note:

You will be prompted to select the simulation if there are multiple testbenches or simulations found in the history point.

The corresponding snapshot of the testbench is loaded. You can click the **Testbenches** tab and see the loaded testbench listed.

In the History pane, the tooltip shows the base history number for the loaded history point row when selected. The base history is the original (starting point) snapshot from which a particular history point is created.

Viewing Results of a History Point

You can choose **Results > Results Viewer** to view the simulation result of the current active history point in the **Results View** window. The name and the index of the current active history point is displayed in bold in the test suite history tree.

Alternatively, to view results of a history point:

1. Select a row in the History pane.
2. Right-click and choose **Open Results Viewer** from the menu.

Launching a Terminal from a History Point

The results for each history point are stored in different directories. To open a terminal for a selected history point that points to its results directory:

1. Select a row in the History pane.
2. Right-click and choose **Launch Terminal** from the menu.

A terminal is opened pointing to the path of the selected history point. You can now browse the simulation output files in that history point's directory.

Renaming History Points

To rename a history point:

1. Select a row in the History pane.
2. Right-click and choose **Rename** from the menu.

You can now rename the history point.

When the preference `saShowHistoryDirectoryInHistoryManager` is true, the renaming feature renames the history directory name. If the same history directory is shared in other single-testbench session testbenches, the new name will be updated in those testbenches as well.

Locking History Points

When you lock a history point, the snapshot cannot be deleted by the auto purge mechanism. You can unlock a locked history point.

To lock a history point:

1. Select a row in the History pane.
2. Right-click and choose **Lock** from the menu.

A lock icon is visible on the left of the index, indicating that the specific history point is locked.

Unlocking History Points

To unlock a history point:

1. Select a locked row (identified by the lock icon on the left of the index) in the History pane.
2. Right-click and choose **Unlock** from the menu.

The lock icon is removed, indicating that the history point is unlocked.

Deleting History Points

To delete a history point:

1. Select one or more rows in the History pane.
2. Right-click and choose **Delete History Points** from the menu. Alternatively, you can also delete the selected history points by clicking the delete icon.

The selected history points are deleted.

Deleting Results of a History Point

To delete results of a history point:

1. Select a row in the History pane.
2. Right-click and choose **Delete Results** from the menu.

The simulation results of the selected history point are deleted.

Saving States

A PrimeWave Design Environment state is a collection of files that captures all settings in a testbench. Once states are named, a directory is also created with that name, in which the various files and settings are stored. Individual state files are saved in XML format and are named so that you can identify the type of information included in each file.

For example, the `analyses.xml` file includes all information for analyses that are defined within a testbench. The `models.xml` file includes all information for models that are part of the testbench.

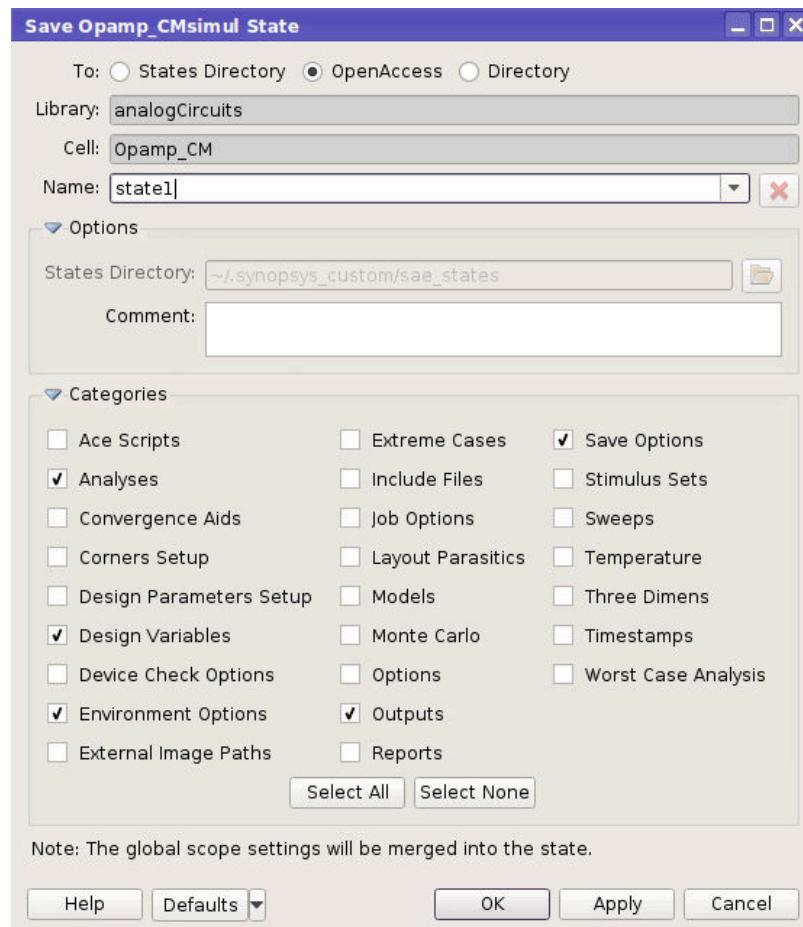
You can save states in any directory or with your design in the corresponding cell directory. The latter option gives you the ability to co-manage the state with your design data.

By default, you are prompted to save the state before you exit the PrimeWave Design Environment. To change this, choose **TestSuite > Options** and uncheck the **Prompt for Save State** box in the **Session Options** dialog box.

To save a state:

1. Choose **Testbench > Save State** from the PrimeWave Design Environment main menu bar.

The **Save State** dialog box opens.



2. Choose the location where you want to save the state:

- **States Directory**

When you choose this option, your state is saved in the following location:

```
<top_level_states_dir>/<Library_Name>/<Cell_Name>/<simulator_name>/  
<state_name>
```

The individual XML files for each major category of the testbench are saved in the <state_name> directory.

You can set the <top_level_states_dir> in the **States Directory** text box. If you do not see this text box, click the arrow next to the **Options** section heading to expand the section.

The **Library** and **Cell** names correspond to the design that is currently under test, and the simulator name corresponds to the active simulator. Any existing states for the current design or simulator combination are listed in the **State** menu. If you choose an existing state, you are asked to confirm that you want to overwrite the state.

- **OpenAccess**

When you choose this option (the default), the state is stored in the cell directory of your current design.

If you open the Library Manager after saving a state in this manner, the states appear with your cellviews in the **View** column. You can double-click a state name in the Library Manager to start a PrimeWave Design Environment session with that state loaded. You must have write privileges for these cell directories in order to save states.

- **Directory**

When you choose this option, you can specify any directory in the **States Directory** text box. Any existing states are listed in the **State** menu. If you choose an existing state, you are asked to confirm that you want to overwrite the state.

You can also add comments to the state that you save, regardless of which method you choose to store the state. These comments then are visible in the **Load State** and **Save State** dialog boxes.

3. In the **Categories** section, select what information in the state file you want to save in the state.

For example, you can save analysis information, but not the outputs. In this case, the `outputs.xml` file is not created, but the `analyses.xml` file is created.

You can use this selective saving feature to incrementally build states from various sessions. All categories are saved by default.

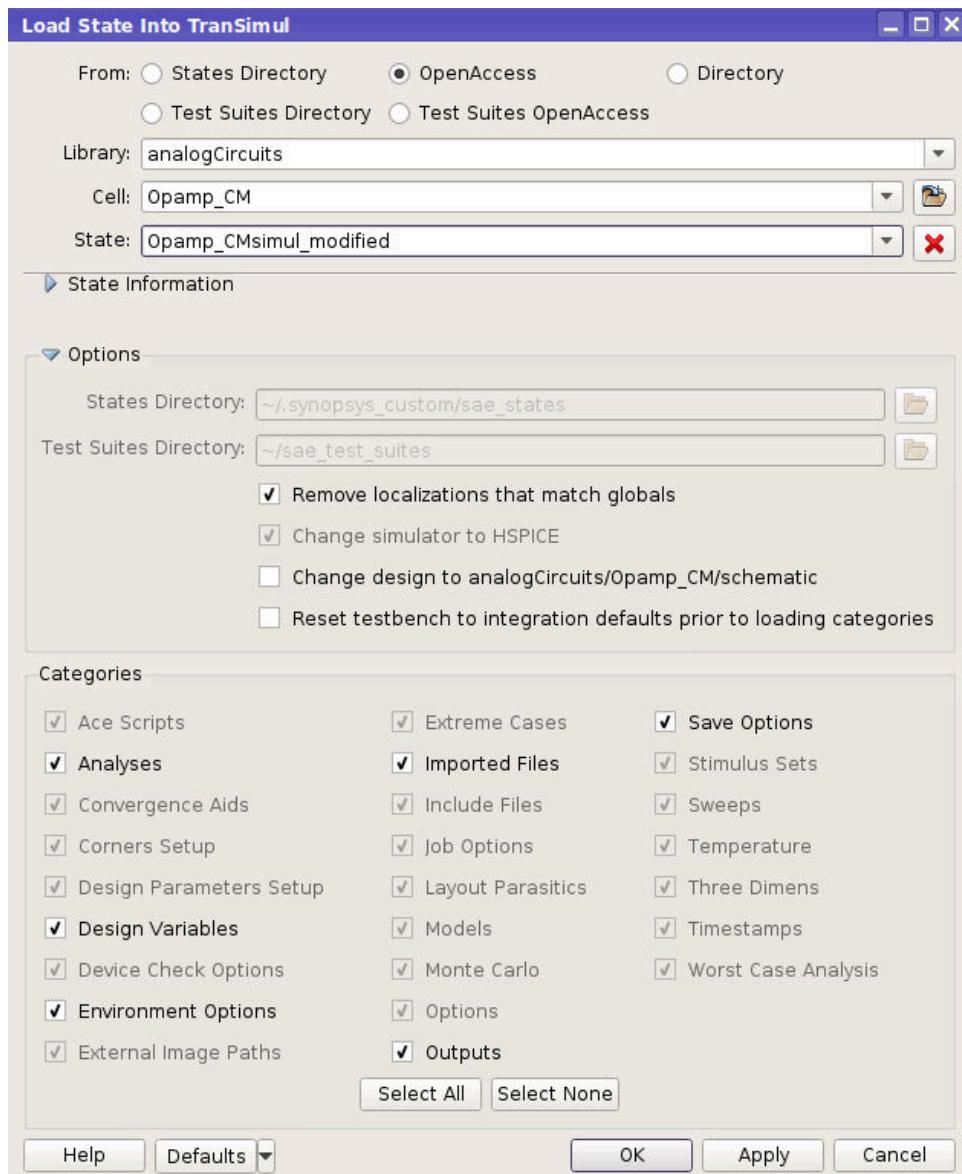
4. Click **OK** to save the state.

Loading States

To load a state into the PrimeWave Design Environment:

1. Choose **Testbench > Load State** from the PrimeWave Design Environment main menu bar.

The **Load State** dialog box opens.



2. Choose the location of the saved state that you want to load:

- **States Directory**

When you choose this option, you also need to specify a **Library**, **Cell**, and **Simulator** that is associated with the saved state.

State information is located in the following directory hierarchy:

```
<top_level_states_dir>/<Library_Name>/<Cell_Name> <simulator_name>/  
<state_name>
```

You can set the `<top_level_states_dir>` in the **States Directory** text box. If you do not see this text box, click the arrow next to the **Options** section heading to expand the section.

Once you specify the **Library**, **Cell**, and **Simulator**, you can choose from a list of available states in the **State** menu.

- **OpenAccess**

When you choose this option (the default), you can load states that are stored in the cell directories of your design libraries.

Once you specify the **Library** and **Cell**, you can choose from a list of available states in the **State** menu.

Note:

When the **OpenAccess** option is enabled, you do not need to specify a directory since the state directory resides in your design data.

- **Directory**

When you choose this option, you can load states from any specified directory.

Enter the directory path into the **States Directory** text box or browse to the directory that contains the desired state information. If you do not see this text box, click the arrow next to the **Options** section heading to expand the section.

Once you specify a states directory in the **Options** text box, any state directories that are located in that directory are listed in the **State** menu.

If a difference is detected between the current active simulator or design and the active simulator or design corresponds to the saved state, you can click the **Change Simulator to <simulator>** or **Change Circuit to <circuit>** options to restore the simulator or design to the previously saved state. Selecting either of these options reinitializes the session with the selected State loaded.

Note:

The on-disk representation of the state is identical regardless of the option that you select. Only the directory structure above the actual state directory differs with the options.

You can find information about the state in the **State Information** section. The state information includes the time and date of the last change, as well as any comments.

3. In the **Options** section, you can change the simulator, the design, the testbench, or the results directory. You can also reset the testbench to integration defaults.

You can change the name of the testbench so that simulations will be written to another results directory and not overwrite previous results.

You can incrementally add setup options to the active state. When you do this, the PrimeWave Design Environment merges the setup from active state into the new state. To load different states and incrementally build the final state, be sure to uncheck **Change testbench name** so the testbench name will not be changed after loading another state.

4. In the **Categories** section, select what information in the state you want to load.

All information is loaded by default.

Note:

If you include **Outputs** in your state when loading it, those outputs from the loaded state are added to any outputs you have already set up—any outputs that already exist are not overwritten.

5. Click **OK** to load the state into the PrimeWave Design Environment.

Saving Tcl Scripts

To save a Tcl script that contains all the settings of the current simulation setup, choose **TestSuite > Save Script** from the PrimeWave Design Environment main menu bar.

You can include the following parts of your test suite in your Tcl script:

- **Run Simulation**
- **Plot Waveforms**
- **Generate Text Output**

Click the **Edit** button to specify what kind of report(s) you want, as well as what types of information you want to include in the report(s). See [Specifying Report Options](#).

You can use this Tcl script in the non-graphical PrimeWave Design Environment equivalent, pw_shell. In the pw_shell, you can use a collection of high-level Tcl commands to perform tasks such as specifying the testbench settings, netlisting, simulating, and postprocessing results.

See [Creating PrimeWave Design Environment Scripts](#) for more information on pw_shell scripting.

3

Working With Design Variables

This chapter contains information on how to add, edit, and delete design variables.

Design variables are used to parameterize design information such as the widths of MOSFETs, voltage values of testbench sources, and model parameters. You can use design variables to create an impact on the design (via the testbench setup) without changing the design itself.

The first row in the design variables pane is "temperature", which is not a design variable object, and therefore cannot be deleted. The context-sensitive menu for "temperature" has different options.

Design variables consist of name and value pairs, but the variable values can also be references to expressions or to other design variables. The simulator evaluates design variables, and the values that are specified in the PrimeWave Design Environment main window are netlisted as is.

Design variables can also be grouped, which allows you to switch between collections of variables and associated values.

Note:

Design variables can be used in many of the text fields throughout the Primewave Design Environment. For example, design variables can be used in text fields within any kind of analysis.

This chapter contains the following topics:

- [Adding and Editing Design Variables](#)
- [Editing Design Variables in a Text Editor](#)
- [Adding Cell View Variables](#)
- [Adding String Variables](#)
- [Deleting Design Variables](#)
- [Globalizing Local Design Variables](#)

- [Localizing Global Design Variables](#)
- [Pushing Design Variables to All Testbenches](#)
- [Adding Design Variable Sets](#)
- [Copying Design Variable Sets](#)
- [Choosing Design Variable Sets](#)
- [Deleting Design Variable Sets](#)
- [Copying Design Variables from a Design](#)
- [Copying Design Variables to a Design](#)
- [Searching for Design Variables](#)
- [Parameterizing Designs](#)
- [Parameterizing Files](#)

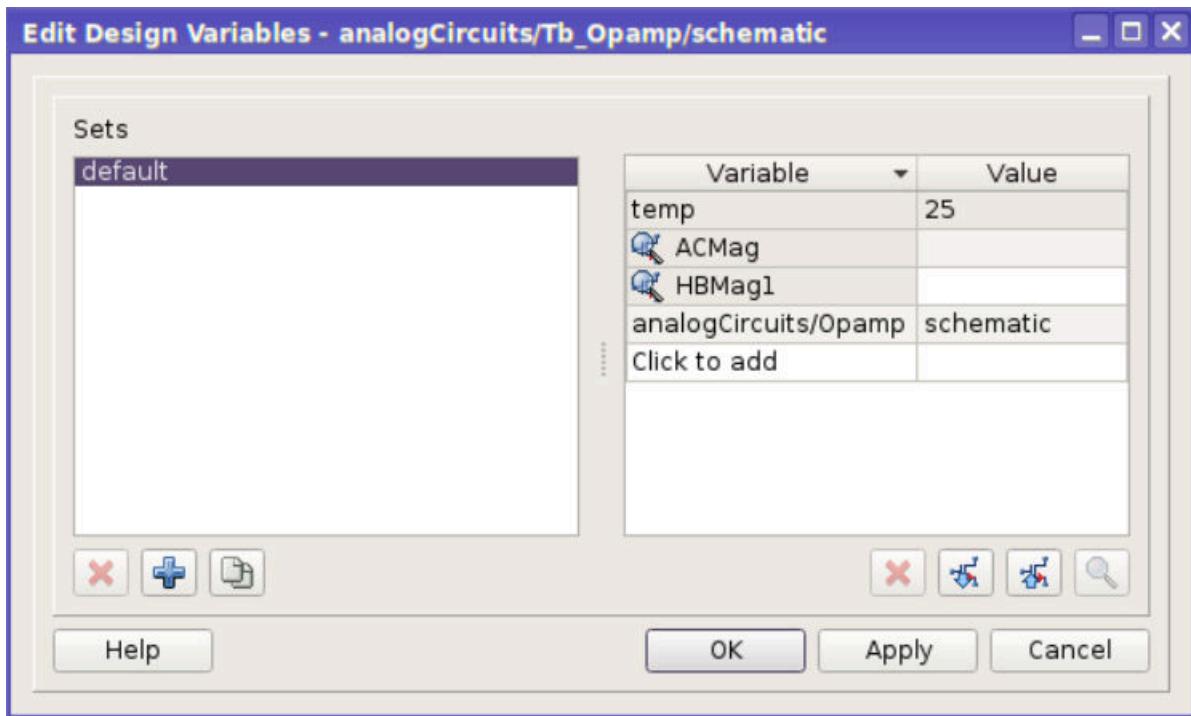
Adding and Editing Design Variables

Note:

New variables are added to the "default" variable set. If you want to add more design variable sets, see [Adding Design Variable Sets](#).

You can add design variables in any one of following ways:

- Enter variables directly into the variables table of the PrimeWave Design Environment main window.
- Choose **Variables > Edit** from the PrimeWave Design Environment main menu bar (or press the V key) to open the **Edit Design Variables** dialog box.



This method provides more options such as creating additional sets of design variables.

Note:

If you are working in a test suite with multiple testbenches, choose **Global** or a testbench name from the **Scope** menu to add variables globally or to a specific testbench, respectively.

- Choose **Variables > Copy from design** to retrieve any variables that are included in the design hierarchy.

Any design variables that are specified in your testbench are netlisted according to the target simulator parameter syntax. For example, in HSPICE, the design variables are represented as `.param` statements in the netlist. Design variables are normally single values, but you can quickly create them by entering sweeps as the values directly.

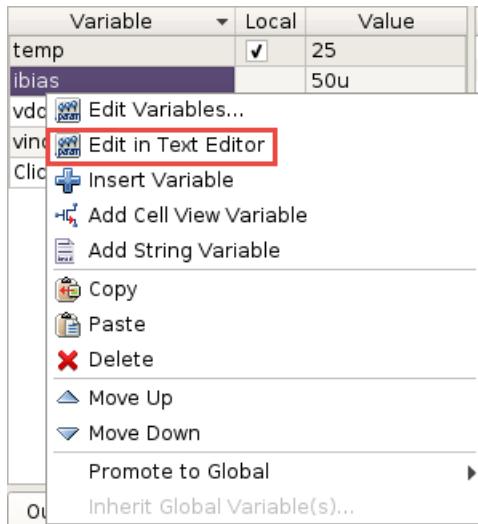
The **Points of Interest** and **Start:Step:Stop** values are supported. For example:

```
TEMP = -55, 25, 55
VDD = 1.5 : 0.1 : 2.1
```

Entering sweeps directly automatically creates sweep analyses, which you can edit if necessary.

Editing Design Variables in a Text Editor

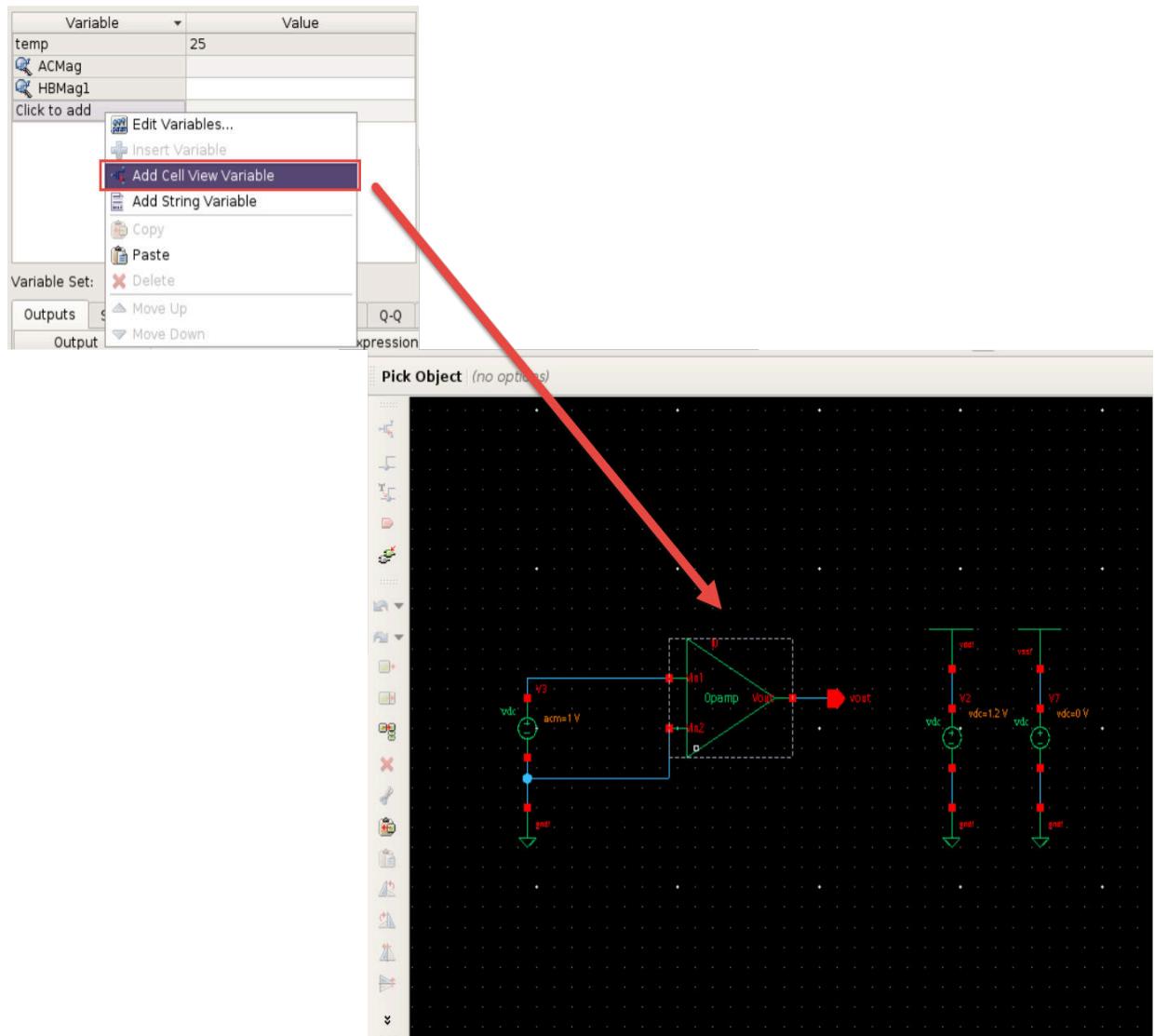
You can edit a design variable in the text editor by right-clicking in the **Variable** table and selecting **Edit in Text Editor**.



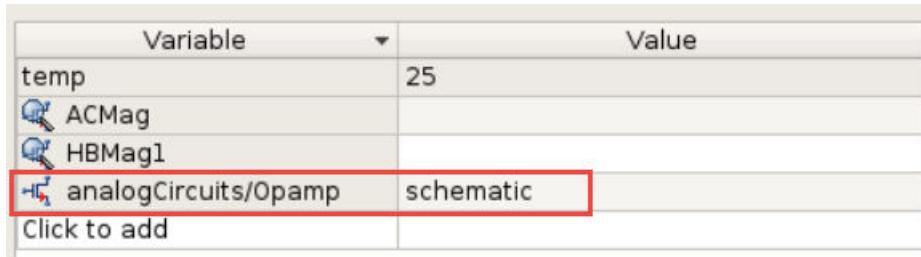
Adding Cell View Variables

You can add cell view variables to the design variables pane:

- Choose **Variables > Add Cell View Variable** from the PrimeWave Design Environment main menu bar to open the schematic design. Alternatively, right-click a cell in the variables table to open the menu and choose **Add Cell View Variable**.



- Select the schematic object for which you wish to create a cell view variable. The cell view variable is added to the variables table as a nominal value.



Variable	Value
temp	25
ACMag	
HBMag1	
analogCircuits/Opamp	schematic
Click to add	

Adding String Variables

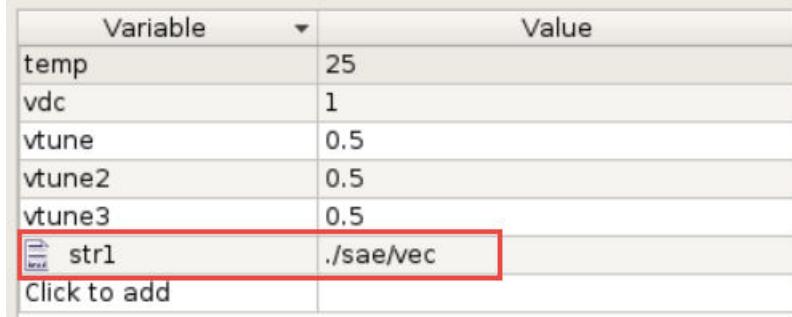
You can add string variables to the design variables pane:

- Right-click a cell in the variables table to open the menu and choose **Add String Variable**.

The **Add String Variable** dialog box opens.



- Enter the **Name** of your string variable.
- Enter the **Nominal Value** of your string variable.
- Click **OK** to add the string variable to the variables table.



Variable	Value
temp	25
vdc	1
vtune	0.5
vtune2	0.5
vtune3	0.5
str1	./sae/vec
Click to add	

The value of the string variable is written in the final PrimeSim SPICE netlist using the `str()` function

For example, for the string variable `str1` with the value `./sae/vec` (shown in the above screen capture), the PrimeSim SPICE netlist is as follows:

```
.param f0=1G R1=50 str1=str('./sae/vec')
.temp 25
```

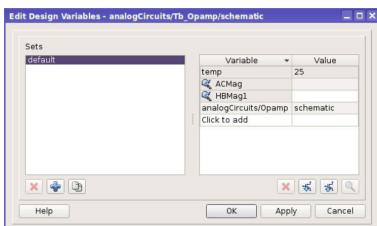
Third-party format netlists use double quotation marks:

```
parameters f0=1G R1=50 str1=".sae/vec"
```

Deleting Design Variables

To delete a design variable, do one of the following:

1. Choose **Variables > Edit** from the PrimeWave Design Environment main menu bar. The **Edit Design Variables** dialog box opens.



2. Place your cursor in either the **Variable** or **Value** field, and click the **Delete Selected Variable(s)** button in the lower right corner of the **Edit Design Variables** dialog box.
3. Right-click the name of a design variable in the **Design Variables** table of the PrimeWave Design Environment main window, and choose **Delete** from the menu that opens.
4. Select the contents of a row in the **Design Variables** table, and press the **Delete** key.

Note:

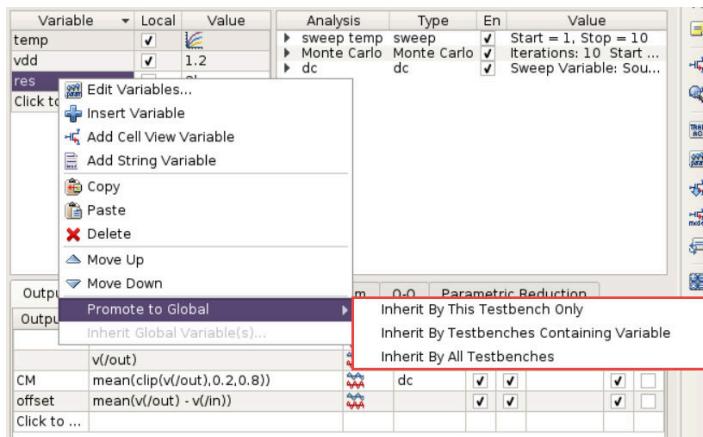
The first row of the design variables pane (including the **Edit Design Variables** dialog box) is **temperature** and cannot be deleted.

Globalizing Local Design Variables

A local design variable is a variable that applies only to a particular testbench. When you globalize a local variable, that variable is included in all testbenches in a test suite.

To make a local design variable global when working in a test suite with multiple testbenches:

1. In the PrimeWave Design Environment main window, click the name of a testbench in the test suite pane that contains the local variable you want to globalize.
2. In the Design Variables pane of the PrimeWave Design Environment main window, right-click a local design variable, and choose **Promote to Global** from the menu that opens.
3. Choose how the selected variable will be inherited: **Inherit By This Testbench Only**, **Inherit By Testbenches Containing Variable**, or **Inherit By All Testbenches**.



The selected variable is now global to the testbenches you chose.

Localizing Global Design Variables

A global design variable is a variable that applies to all testbenches in a test suite. When you localize a global variable, that variable is included in a single testbench you specify.

To make a global design variable local when working in a test suite with multiple testbenches:

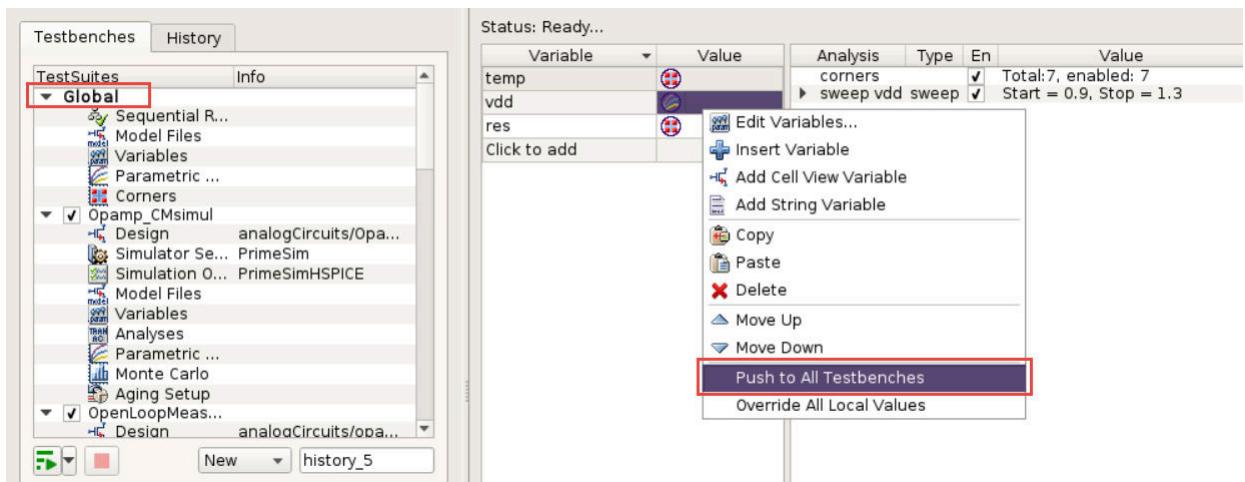
1. In the PrimeWave Design Environment main window, click the name of a testbench in the test suite pane that contains a global variable you want to localize to that testbench.
2. In the Design Variables pane of the PrimeWave Design Environment main window, right-click a global design variable, and choose **Localize Global Variable(s)** from the menu that opens.

The selected variable is now local to the selected testbench.

Pushing Design Variables to All Testbenches

To push a design variable to all testbenches when working in a test suite with multiple testbenches:

1. Click the **Global** heading in the test suite pane.
2. Right-click a design variable name from the **Variable** table in the PrimeWave Design Environment main window.



3. Choose one of the following menu options:

- **Push to All Testbenches:**

The design variable is applied to all testbenches in the test suite.

- **Override All Local Values:**

This option de-localizes all existing local variables with the selected name. This therefore has the effect of overriding all local variables that already exist. It does not create any new local variable.

Adding Design Variable Sets

To add a new design variables set, type a name directly into the **Variable Set** selector in the PrimeWave Design Environment main window.

A set corresponding to the name that you enter is added, which includes the design variables and associated values that are currently displayed.

You can also add design variable sets using the **Edit Design Variables** dialog box:

1. Choose **Variables > Edit** from the PrimeWave Design Environment main menu bar.

The **Edit Design Variables** dialog box opens.

2. Click **Add New Set**  just below the **Variable Sets** text box.

A new variable set name named `group<number>` appears in the **Variable Sets** text box, which you can edit by clicking and typing a new name.

Note:

New design variable sets created using the **Edit Design Variables** dialog box are empty.

Copying Design Variable Sets

To copy a set of design variables:

1. Choose **Variables > Edit** from the PrimeWave Design Environment main menu bar.

The **Edit Design Variables** dialog box opens.

2. Click the name of the variable set you want to copy from the list of available design variable sets.

3. Click the **Copy Selected Set** button to the right of the plus button that is just below the list of variable sets.

A new design variable set named `group<number>`, which contains all of the variables of the copied set, appears in the list of design variable sets. To edit the design variable set name, click the name and enter a new name.

You can also copy a design variable set by directly editing the contents of the **Variable Set** menu in the PrimeWave Design Environment main window. To edit the contents, choose the name of the set from the **Variable Set** menu, and when you focus away from the field, the variable set will be copied to the new set.

Choosing Design Variable Sets

You can choose a variable set from **Variable Set** selector, which is located just below the **Variables** table in the PrimeWave Design Environment main window.

Unless you create additional design variable sets, only the "default" variable set is available. You can create additional design variable sets; see [Adding Design Variable Sets](#).

Deleting Design Variable Sets

To delete a set of design variables:

1. Choose **Variables > Edit** from the PrimeWave Design Environment main menu bar.
The **Edit Design Variables** dialog box opens.
2. Click the name of the variable set you want to delete.

Note:

- The default variable set cannot be deleted.
3. Click the **X** button just below the **Variable Sets** text box.
The variable set disappears from the list.

Copying Design Variables from a Design

To copy design variables from a design, choose **Variables > Copy from Design** from the PrimeWave Design Environment main menu bar, or press **Shift+V**.

You can also copy design variables from a design by clicking the **Copy from Design** button in the **Edit Design Variables** dialog box (**Variables > Edit**).

Copying Design Variables to a Design

To copy a set of design variables back to your design, choose **Variables > Save to Design** from the PrimeWave Design Environment main menu bar.

You can also copy a set of design variables from the **Edit Design Variables** dialog box (**Variables > Edit**). Click the button in the lower-right section of the dialog box.

Design variables are stored in the cell-level DM Data as an oaAppProp with the name desVarList.

Searching for Design Variables

To search a design for components that use the design variables you create:

1. Choose **Variables > Edit** from the PrimeWave Design Environment main menu bar.

The **Edit Design Variables** dialog box opens.

2. Click the name(s) of one or more variables you want to find in your design.

Ctrl+click each variable name to select more than one variable.

3. Click the **Find variable(s) in design** button.

The Schematic Editor opens. Probes are created in the design, which lead to the components that reference the variable.

Each located parameter appears as a different color, and the probe assistant displays the instance and design variable name for each probe.

To see the details of how the variable is used, select the component and invoke the Property Editor. If you do not see the design variable referenced in any of the values, you might be looking at a hierarchical block that contains your component. Descend into the block to see if any present additional probes lead elsewhere.

Parameterizing Designs

Parameterization allows you to vary device properties on read-only designs. You can save the settings from the **Design Parameterization** dialog box as part of a PrimeWave Design Environment state.

This feature is supported for nominal and corner analyses. You can iterate through the variable values from the Corners setup.

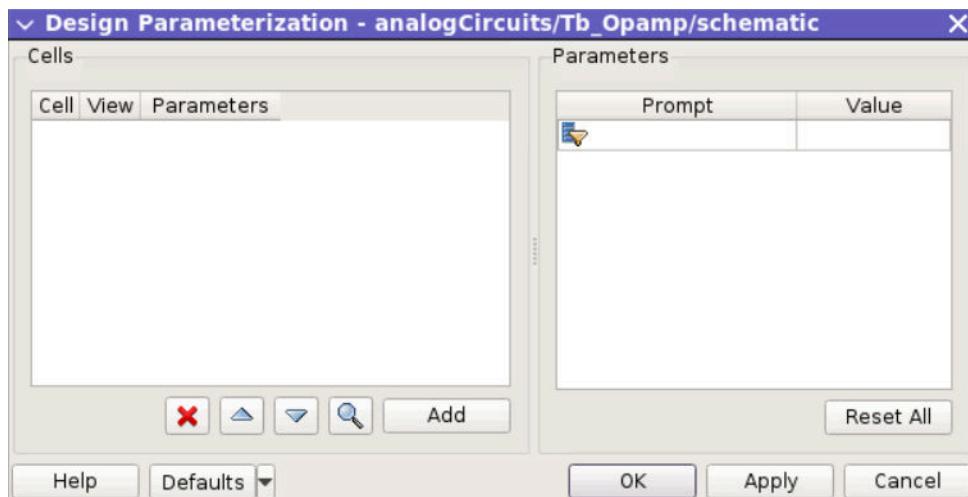
This section contains the following topics:

- [Creating Design Variables from Existing Design Parameters](#)
- [Netlisting Parameterization Setup](#)

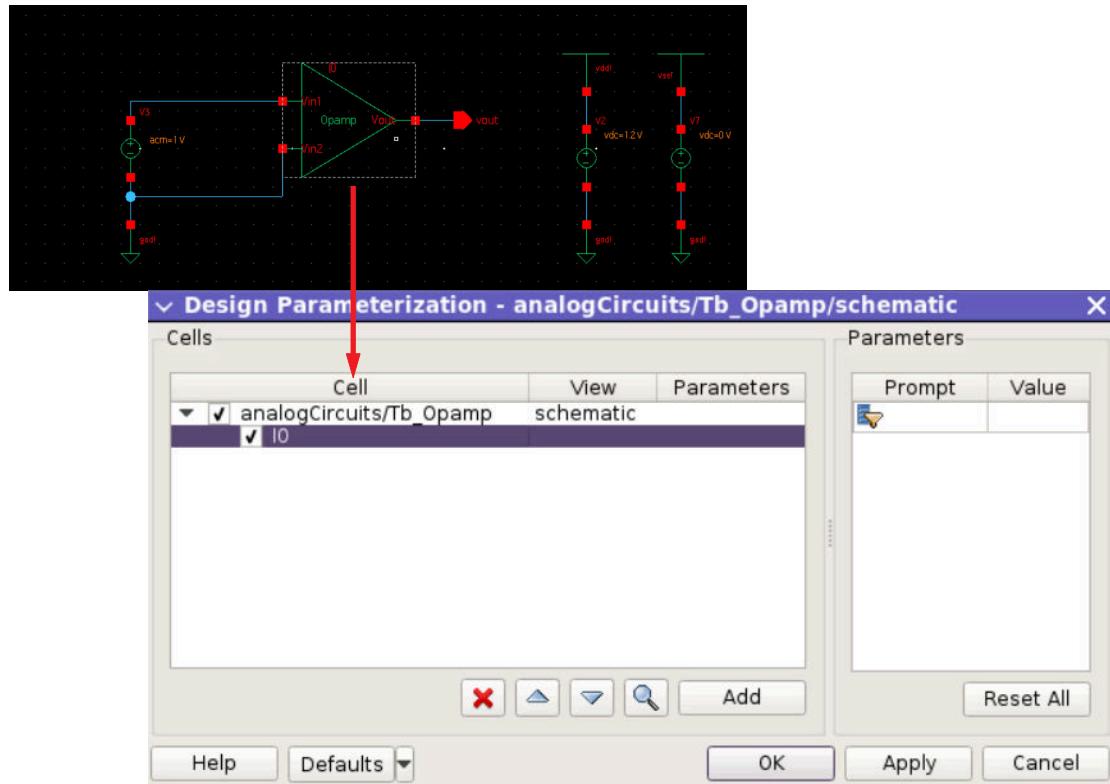
Creating Design Variables from Existing Design Parameters

To create design variables from existing design parameters:

1. Choose **Variables > Design Parameterization** from the main PrimeWave Design Environment menu. The **Design Parameterization** dialog box opens.

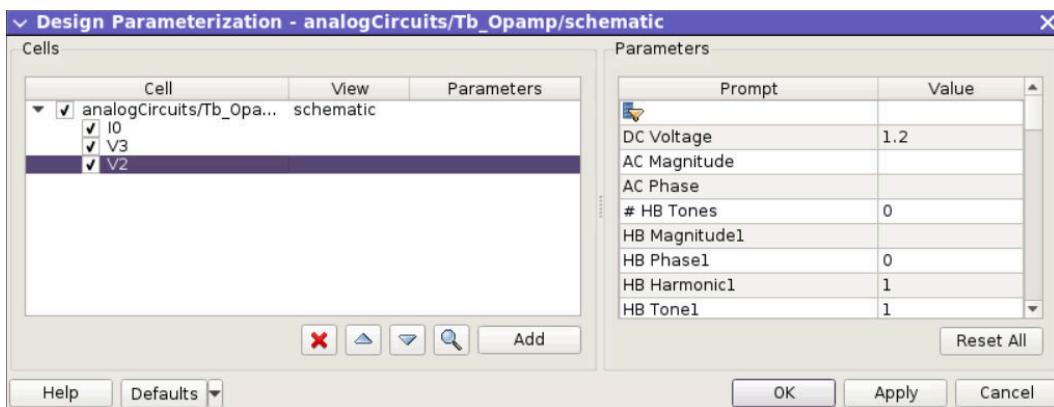


2. Click **Add** to open the schematic design.
3. Click an instance in the schematic design to add it to the **Design Parameterization** dialog box.



The parent information of the instance, including the Library/**Cell** name and the and bounded **View** name, populates a new row in the **Cells** hierarchy table.

4. (Optional) Continue to add instances from the schematic design using the **Add** button and picking from the schematic.



The **Parameters** table is populated with the parameters of the selected device in the hierarchy table.

Note:

The parameters appear in the same order that they would appear in the Property Editor, but the **Parameters** table is not an extension of the Property Editor. CDF callback procedures are not triggered when values are entered or changed in the **Parameters** table.

5. Edit parameter values in the **Parameters** table.

You can parameterize any parameter type. For example, parameters of type cyclic, string, and so forth are supported.

The new parameter values can be either a variable name or an expression (such as NF/2) with a variable name or a literal value.

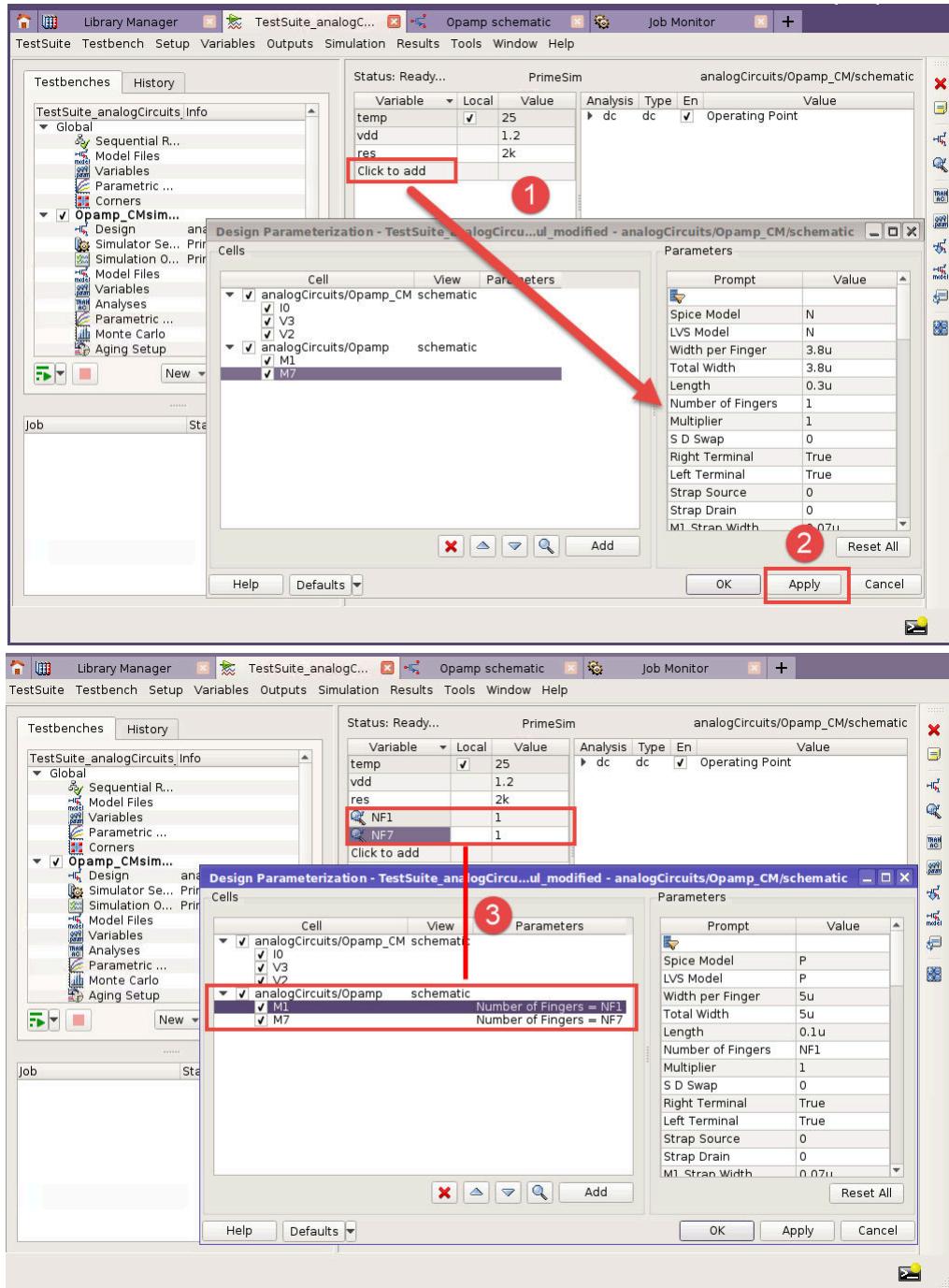
The new parameter values you add are displayed in the **Parameters** column of the **Cells** hierarchy table.

You can filter the parameters to be displayed using the **Filter** box.

6. Click **OK** or **Apply** to create design variables.

Chapter 3: Working With Design Variables

Parameterizing Designs



If the parameter value is not an existing design variable, a new design variable is created. In the example above, there are no design variables for Number of Fingers

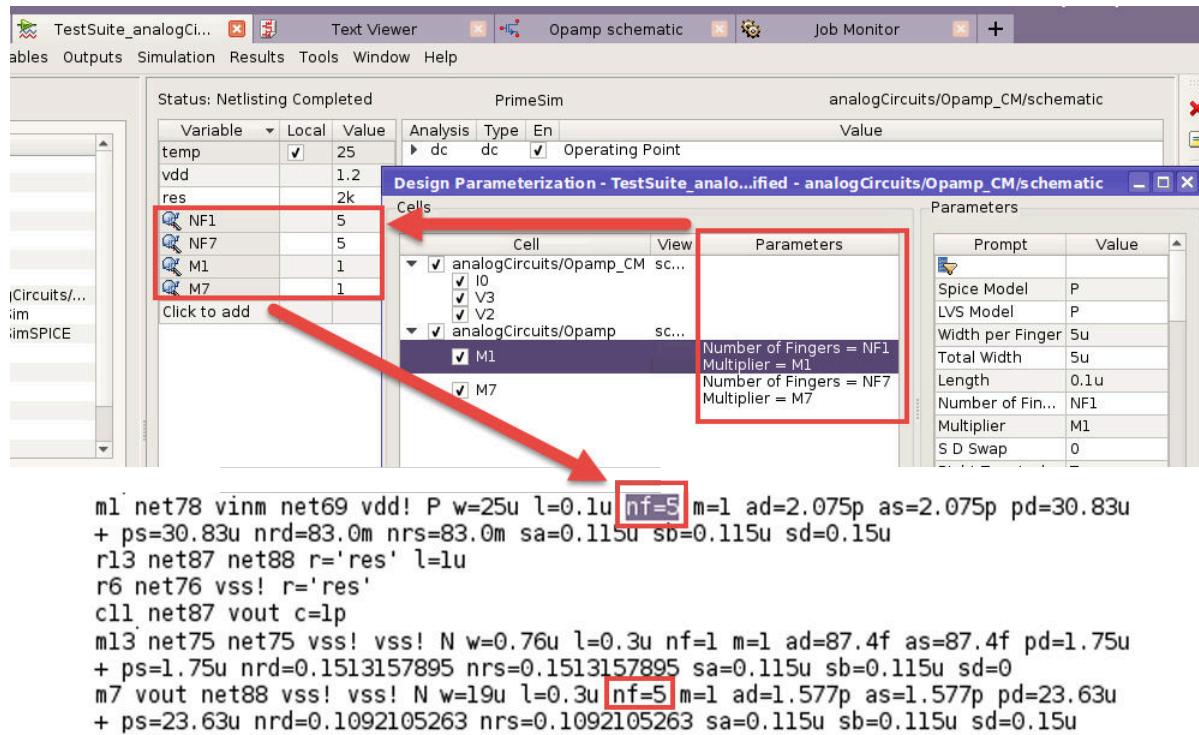
in the Variables pane (1). After pressing **Apply** (2) in the **Design Parameterization** dialog box, the new design variables NF1 and NF7 appear in the Variables pane (3).

When a new design variable is created, its default value is set to the current value of the parameter. The variables that are created using this Design Parameterization and not originally present in the design are denoted by the  icon.

7. Choose **Simulation > Netlist and Run** to generate simulation results.
8. Choose **Testbench > Save State** to save the new design variables.

Netlisting Parameterization Setup

The tool updates the design with parameterized values before running the netlister. The figure below illustrates the effect on the netlist of parameterizing the Number of Fingers properties, of the Instances M7 and M1, with the nf variable values.



Parameterizing Files

You can parameterize file names for the purpose of using that file name in a corner analysis.

Possible applications for using parameterized files include:

- to sweep through different initial condition files
- to sweep through different S-Param blocks in an S-model
- to sweep through different pattern files in a vpat or vpwl source

[VAR\(\) Expressions](#) can be used to aid in file parameterization. Supported file types and how to parameterize them are described in the following topics:

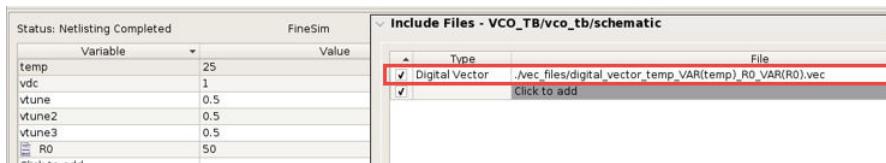
Type of File	Description	See ...
Include Files	Any include file Type (digital vector, veriloga, node set, and so forth) from the Include Files dialog box	Parameterizing Include Files
SPF/DSPF	SPF/DSPF files (extracted for different conditions but for same block) from the Parasitic Back Annotation dialog box	Parameterizing SPF/DSPF Files
Stimulus	The stimulus file specified in the instance parameter of the source elements (for example, PWL sources) in the schematic design	Parameterizing Stimulus Files

Parameterizing Include Files

Digital vector files can be parameterized based on the file name pattern.

To parameterize include files:

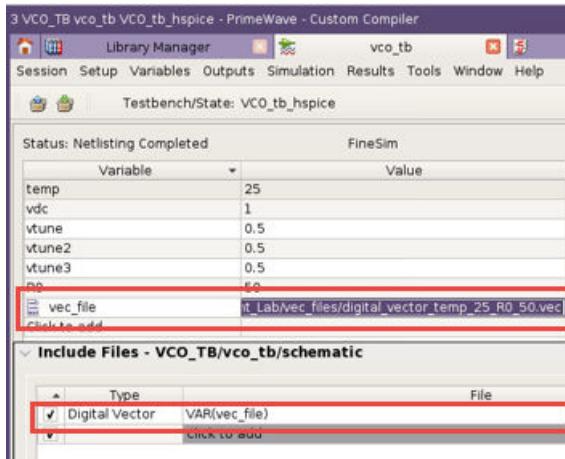
1. Use a `VAR()` expression to parameterize any include file type in the **Include Files** dialog box. Multiple `VAR()` expressions are also allowed. (See [VAR\(\) Expressions](#).)



2. Add a new string variable in the Variables table. (See [Adding String Variables](#).)

Chapter 3: Working With Design Variables

Parameterizing Files



3. Specify a nominal value for the variable in the Value column of the Variables table.
4. To promote the string variable created, right click to open the menu and choose **Promote to Global > Inherit by Testbenches** containing this variable. Now, this variable can be used like any corner parameter.
5. Use the **Manage Corner Parameters** dialog box to add the variable as a corner parameter. (See [Specifying Design Variables for Corners](#).)
6. Specify the file name to be used for each corner in the **Corners Setup** dialog box. (See [Renaming Corners](#).)
7. Netlist and run the simulation.

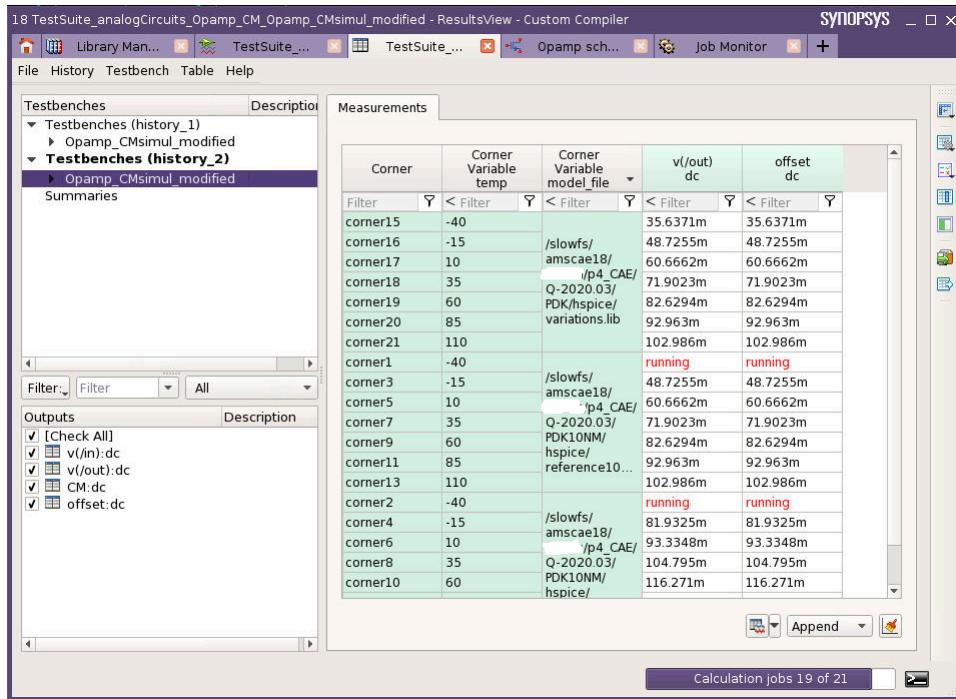
While creating the final netlist for each corner/nominal, the PrimeWave Design Environment evaluates the `VAR()` expressions found in the file name, replaces `VAR()` in the file name string with the evaluated value, and writes the file name string in the netlist.

```
primesim.spi
*
* Generated for: PrimeSim SPICE
* Design library name: analogCircuits
* Design cell name: Opamp_CM
* Design view name: schematic
.lib 'reference40_models.lib' TT
.include '/slowfs/amscae18/PDK/hspice/variations.lib'
```

The WaveView tool, the Results Viewer, and the Results Analyzer display the parameterized variable like any other corner parameter.

Chapter 3: Working With Design Variables

Parameterizing Files



You can use an existing design variable in the `VAR()` expression if the include files are named with variable values.

Parameterizing SPF/DSPF Files

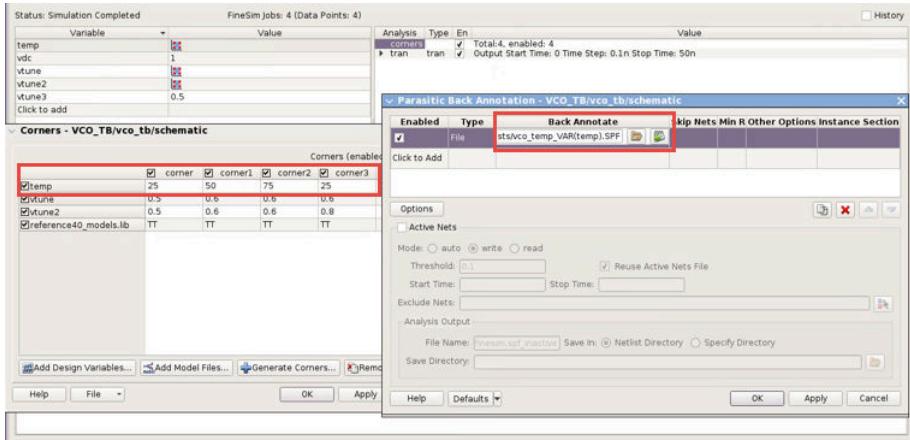
When running simulations using multiple SPF files that come from post-extraction runs, these file names typically follow a naming pattern. You can set up corner simulations to run such multiple SPF files at the same time using [VAR\(\) Expressions](#).

The procedure to parameterize SPF/DSPF files, cell views, or GPD files in the **Parasitic Back Annotation** dialog box is very similar to [Parameterizing Include Files](#). Either a string variable or an existing design variable can be used in the `VAR()` expression in this method.

In the example below, `temp` is a variable for the corner simulation.

Chapter 3: Working With Design Variables

Parameterizing Files

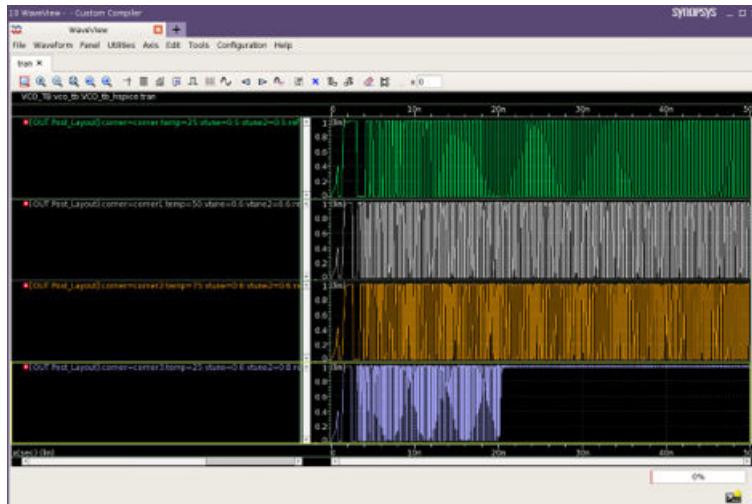


While creating the final netlist for each corner/nominal, the PrimeWave Design Environment evaluates the `VAR()` expressions found in the file name, replaces `VAR()` in the file name string with the evaluated value, and writes the file name string in the netlist.

The results from the multiple SPF files used in the above corner simulations are:

- <path to simulation>/VCO_TB,vco_tb,schematic/history_1/simulation/VCO_tb_hspice/FineSimPro/corners/netlist
 - corner: .option finesim_spf=<path to extracted_netlists>/vco_temp_25.SPF"
 - corner1: .option finesim_spf=<path to extracted_netlists>/vco_temp_50.SPF"
 - corner2: .option finesim_spf=<path to extracted_netlists>/vco_temp_75.SPF"
 - corner3: .option finesim_spf=<path to extracted_netlists>/vco_temp_25.SPF"

The waveform results from all post-layout simulations appear in the WaveView tool:



Parameterizing Stimulus Files

To parameterize stimulus files in the design:

1. In the schematic design, open the Property Editor.
2. Select an object in the design.
3. Check the **File name is variable** option and add a variable name in the file parameter.
4. In the Variables table of the main PrimeWave Design Environment window, right-click and choose **Copy from Design** to scan the design variables from the design. (See [Copying Design Variables from a Design](#).)

The PrimeWave Design Environment adds a new string variable for the stimulus files found in the design.

5. Specify the nominal value for the stimulus file in the Value column of the Variables table.
6. Use the **Manage Corner Parameters** dialog box to add the variable as a corner parameter. (See [Specifying Design Variables for Corners](#).)
7. Specify the file name to be used for each corner in the **Corners Setup** dialog box. (See [Renaming Corners](#).)
8. Netlist and run the simulation.

VAR() Expressions

The `VAR()` expression provides a way to parameterize file names. The `VAR()` expression can be used to specify the entire file name string or part of the file name string. Multiple `VAR()` expressions can be used in the file name string.

Syntax

`VAR(variable_name)`

Arguments

variable_name

The user-defined name of the variable. The variable name can be the design variable from the design, or temp, or a string variable. (See [Adding String Variables](#).)

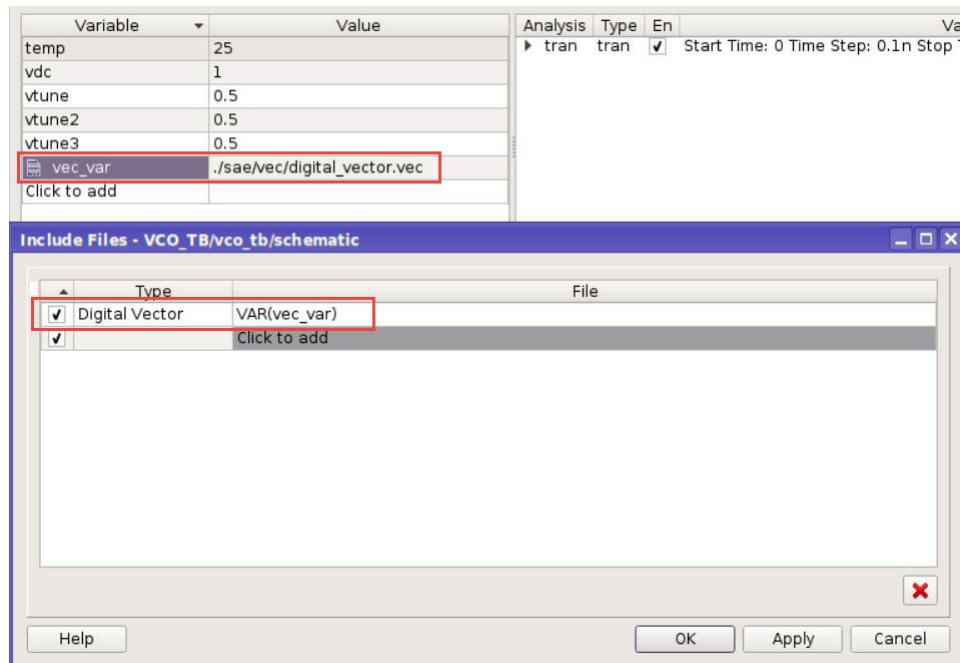
Returns

The value of the variable *variable_name*

Examples

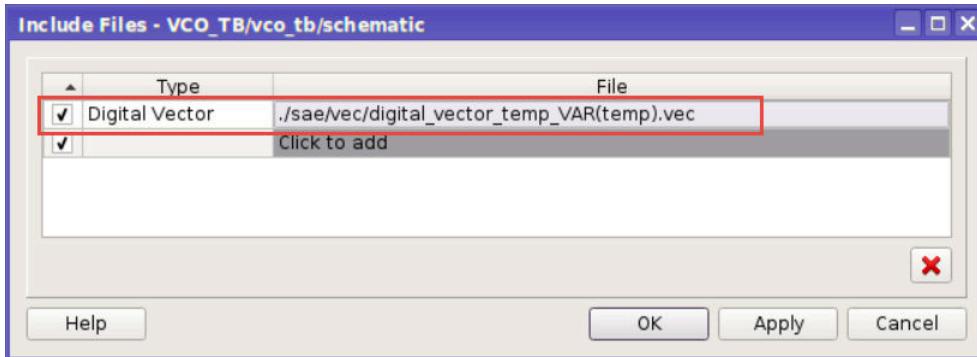
Example 1

The following example shows a `VAR()` expression with the string variable `vec_var`. The string variable `vec_var` represents a digital vector.



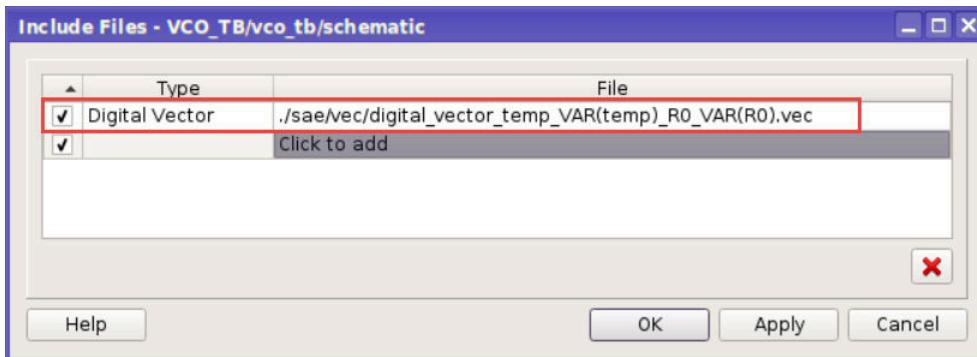
Example 2

The following example shows a `VAR()` expression as part of the file name. The digital vector file uses the `VAR()` expression with the temp variable. When `temp=25`, it represents the file `./sae/vec/digital_vector_temp_25.vec`.



Example 3

The following example shows multiple `VAR()` expressions in the file name. The digital vector file uses the `VAR()` expression with the variables `temp` and `R0`. When `temp=25` and `R0=50`, it represents the file `./sae/vec/digital_vector_temp_25_R0_50.vec`.



4

Working with Analyses

This chapter contains information on how to set up and enable analyses.

Note:

You can use design variables in any analysis.

This chapter contains the following sections:

- [Setting Up Analyses](#)
 - [Enabling Save and Restore Times](#)
 - [Enabling and Disabling Analyses](#)
 - [Editing Analysis Values](#)
 - [Deleting Analyses](#)
-

Setting Up Analyses

To create or edit an analysis, choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

Enabling Save and Restore Times

You can specify save and restore time points for VCS PrimeSim AMS transient analyses as well as for the PrimeSim and FineSim transient analyses. This allows you to run another simulation beginning at a certain saved point.

This section contains information on the following topics:

- [Enabling Save Times](#)
- [Enabling Restore Times](#)

Enabling Save Times

Save times are specified time points during your simulation when you want to save your simulation before it is finished. To enable save times for your transient analysis:

1. Ensure the **Edit/Create Analysis** dialog box is open (**Setup > Analyses** from the PrimeWave Design Environment tab page menu bar).
2. Set up your analysis.
3. Expand the **Save/Restore** section of the analysis dialog box.
4. Enter one or more ordered points at which you want to save the simulation during simulation in the **Save Times** text box.
Multiple time points are separated by a space.
5. Click **OK**.

Your next simulation is saved at the specified time points.

Enabling Restore Times

Restore times are specified time points during your simulation when a simulation is saved. To enable restore times for your transient analysis:

1. Ensure you run a simulation that includes already specified save times.
See [Enabling Save Times](#).
2. Ensure the **Edit/Create Analysis** dialog box is open (**Setup > Analyses** from the PrimeWave Design Environment tab page menu bar).
If you use **Digital Run Control**, you can change the **Stop Time** to be any value beyond the **Restore** time you select in the next steps.
3. Click the check box next to the **Restore Time** menu.
4. Choose a saved time point from the **Restore Time** menu.
The next simulation you run begins at this restore time point.

5. (Optional) Enter any UCLI commands to execute prior to restarting the simulation into the text field below the **Restore Time** menu.

You might consider the `Release` and `Force` commands, which allow your predefined testbench to switch to a different test vector.

6. (Optional) Unless you want to specify the same or additional save time points during your next simulation, uncheck **Save Time Points** to disable saving any more time points.

7. Click **OK**.

Your next simulation begins at the specified time point.

In MTB mode you have the option of running your restored simulation from the same history point, which continues the simulation for that history point. You can also run a simulation from the restored point in a new history point, which keeps each simulation run in a separate results directory.

Enabling and Disabling Analyses

To enable an analysis as you create or edit it, click **Enable**, which is located near the bottom of each analysis form in the **Edit/Create Analyses** dialog box (**Setup > Analyses**).

If an analysis is already set up and displayed in the Analyses pane of the PrimeWave Design Environment main window, you can click (check) the check box in the En column to enable the analysis.

To disable an analysis, click (uncheck) the check box in the En column of the Analyses pane.

You can also enable or disable all analyses at once by right-clicking an analysis name and choosing **Enable All Analyses** or **Disable All Analyses** from the menu that opens.

Editing Analysis Values

To edit the values of an analysis that already exists, choose one of the following options:

- Directly edit values in the **Analyses** pane of the PrimeWave Design Environment main window.

To edit a subset of the analysis parameters for each analysis type, click the plus icon, which is located to the upper-left of an analysis name. Double-click a value, and enter a new value.

- Edit values in the **Edit/Create Analyses** dialog box.

Choose **Setup > Analyses** to open the **Analyses** window, or double-click an analysis in the Analysis pane on the main PrimeWave Design Environment window.

Change values as necessary, and click **OK** to save the changes.

Deleting Analyses

To delete an analysis, select the analysis you want to delete in the PrimeWave Design Environment main window, and click the delete button , which is located to the right of the **History** text box.

5

Setting Up PrimeSim HSPICE Analyses

This chapter contains information on how to set up and enable PrimeSim HSPICE General and RF analyses.

Note:

For more information on a particular PrimeSim analysis, enter `primesim -webhelp` on the PrimeSim command line. Once the PrimeSim help opens, search for the name of an analysis to find information regarding usage, syntax, and options.

Note:

If you want to use an analysis card for any applicable PrimeSim analyses, click the **Use <analysis> card** option at the bottom of the analysis setup page, and enter any needed analysis commands in the text box. When the **Use <analysis> card** option is enabled, any analysis settings you set up on the analysis setup page are disabled and have no effect on netlisting.

Any analysis statements entered in the analysis card text box are netlisted as-is with no error checking. If you add any additional supported sweeps (parameter, .data, or Monte Carlo, for example), the simulation runs, but all internal sweeps are blocked.

To create or edit an analysis, choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The following analyses are available in the PrimeSim HSPICE integration:

- [PrimeSim HSPICE Transient Analysis](#)
- [PrimeSim HSPICE Operating Point Analysis](#)
- [PrimeSim HSPICE DC Analysis](#)
- [PrimeSim HSPICE AC Analysis](#)
- [PrimeSim HSPICE Noise Analysis](#)
- [PrimeSim HSPICE FFT Analysis](#)

- PrimeSim HSPICE Linear Network Parameter Analysis
- PrimeSim HSPICE AC Match Analysis
- PrimeSim HSPICE DC Match Analysis
- PrimeSim HSPICE Transient Noise Analysis
- PrimeSim HSPICE Statistical Eye Analysis
- PrimeSim HSPICE Loop Stability Analysis (LSTB)
- PrimeSim HSPICE RF Harmonic Balance Analysis
- PrimeSim HSPICE RF Harmonic Balance Oscillator Analysis
- PrimeSim HSPICE RF Harmonic Balance AC Analysis
- PrimeSim HSPICE RF Harmonic Balance Transfer Function Analysis
- PrimeSim HSPICE RF Harmonic Balance Noise Analysis
- PrimeSim HSPICE RF Shooting Newton Analysis
- PrimeSim HSPICE RF Shooting Newton Oscillator Analysis
- PrimeSim HSPICE RF Shooting Newton AC Analysis
- PrimeSim HSPICE RF Shooting Newton Transfer Function Analysis
- PrimeSim HSPICE RF Shooting Newton Noise Analysis
- PrimeSim HSPICE RF Shooting Newton with Fourier Transform Analysis
- PrimeSim HSPICE RF Phase Noise Analysis
- PrimeSim HSPICE RF Periodic Time-Dependent Noise Analysis
- PrimeSim HSPICE RF Envelope Analysis
- PrimeSim HSPICE RF Envelope Oscillator Analysis
- PrimeSim HSPICE RF Envelope Fast Fourier Transform Analysis
- PrimeSim HSPICE Bias Check Analyses

PrimeSim HSPICE Transient Analysis

A transient analysis calculates the circuit solution as a function of time and over a specified time range.

To create a transient analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **tran** radio button.
3. (Optional) Specify a value for the **Start Time**.
4. (Optional) Select the number of transient intervals (**Number of Intervals**) to include in the analysis. With multiple intervals you can specify different step times along a transient analysis.

A pair of **Time Step** and **Stop Time** text boxes appear or disappear for each additional interval you add or delete.

5. Specify **Time Step** and **Stop Time** values for each interval.
6. (Optional) Click **UIC** to bypass the initial DC operating point.

When this option is enabled, PrimeSim does not calculate the initial DC operating point but directly enters transient analysis. Transient analysis uses the .IC initialization values as part of the solution for the initial timepoint.

7. (Optional) Choose a **Method** in the Advanced Settings group box. The following options are available:

- **Temp Sweep**

Temp Sweep defines the temperature profile for the transient analysis. You can specify the **Temp Values** and **Temp Step**. If you click the icon next to the **Temp Values** field, the **Edit Temp Settings** dialog box opens, and you can set the temperature values at different time points. You can also specify the value directly in the **Temp Values** field using the syntax of timevec. **Temp Step** defines the time interval for temperature update.

- **Run Level**

You can specify **Run Level Values**. You can set different run level values at the different time points. If the RUNLVL is not specified anywhere in the netlist, the default value is 3. The .option RUNLVL<=value> overrides the default RUNLVL=3.

RUNLVL values defined for a specific transient period in a .TRAN command, overrides the RUNLVL value set by the .option RUNLVL or .option ACCURATE.

To learn the syntax for the temperature sweep, run level, and other options, see the *PrimeSim HSPICE Reference Manual: Commands and Control Options*.

8. (Optional) Expand the **Simulator Check** section to enable the **Check Safe Operating Area** check.

9. (Optional) Expand the **Save/Restore** section to save and restore time points.

See [Enabling Save and Restore Times](#).

10. Ensure **Enable** is checked so you can include the transient analysis as part of your testbench.

11. Click **OK** or **Apply**.

Your transient analysis is now set up.

See Also

- [Transient Analysis](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

PrimeSim HSPICE Operating Point Analysis

Note:

If you want to be able to annotate or print device operating points for any time point other than time=0, set up a transient analysis as well. See [PrimeSim HSPICE Transient Analysis](#).

To create an operating point analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **op** radio button.
3. Select the **Number of Formats** to include in the analysis.

The **Format** parameter controls the type of information that is generated by PrimeSim at a particular time point. A **Format** and **Times** pair appears depending on the number you specify.

Note:

This option only impacts the appearance of the output in the output log file and does not have any effect on the PrimeWave Design Environment postprocessing of operating points.

4. Choose a **Format** type for each format and corresponding time point values for each operating point information that is calculated.
5. (Optional) Check **Interpolation** to specify that operating points can be determined by interpolation.
6. Click **Enable** to enable this analysis as part of your testbench.
7. Click **OK** or **Apply**.

Your operating point analysis is now set up.

See Also

- [Initializing DC-Operating Point Analysis](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

PrimeSim HSPICE DC Analysis

To create a DC analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **dc** radio button.
3. Choose the sweep variable type using **Sweep Variable: Source, Design Variable, Temperature, or Data Driven**.

To specify a **Source** variable type, click **Source** and continue to [Step 4](#).

To specify a **Design Variable** type, click **Design Variable** and skip to [Step 5](#).

To specify a **Temperature** variable type, click **Temperature** and skip to [Step 6](#).

To specify a **Data Driven** variable type, click **Data Driven** and skip to [Step 7](#).

4. Enter the name of the source in the **Source Name** text box, or click the **Select Source Name** button  and click a source in the Schematic Editor.
Return to the **Edit/Create Analyses** dialog box, and skip to [Step 6](#).
5. Specify a design variable name by choosing a **Variable Name** from the drop-down menu.
6. Choose a **Sweep Type**:
 - **Linear Steps**: Specify the **Start** and **Stop** values, as well as the **Step Size**.
 - **Linear Points**: Specify the **Start** and **Stop** values, as well as the **Number of Points**.
 - **Octave**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
 - **Decade**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
 - **Points of Interest**: Specify the **Points** values separated by commas.
7. Click **Manage Parameters** to open the **Manage Data Driven Sweep Parameters** dialog box. Enable your desired parameters in the dialog box or **Click to Add** others and click **OK** to populate the table. Enter the necessary values for each parameter into the table.
8. (Optional) Enable **Hysteresis Sweep** to perform a hysteresis sweep. When set, the hysteresis sweep option is added after the Monte Carlo sweep options.
9. Click **Enable** to enable this analysis as part of your testbench.
10. Click **OK** or **Apply**.

Your DC analysis is now set up.

See Also

- [Initializing DC-Operating Point Analysis](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

PrimeSim HSPICE AC Analysis

An AC analysis calculates the AC output variables as a function of frequency.

To create an AC analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **ac** radio button.

3. Choose a **Sweep Type**:

- **Linear**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
- **Octave**: Specify the **Start** and **Stop** values, as well as the **No. of Points** per octave.
- **Decade**: Specify the **Start** and **Stop** values, as well as the **No. of Points** per decade.
- **Points of Interest**: Specify the **Points** values separated by commas.

4. Click **Enable** to enable this analysis as part of your testbench.

5. Click **OK** or **Apply**.

Your AC analysis is now set up.

See Also

- [AC Small-Signal and Noise Analysis in the PrimeSim HSPICE User Guide: Basic Simulation and Analysis](#)

PrimeSim HSPICE Noise Analysis

Note:

Before running a Noise analysis in PrimeSim, you must first set up and enable an AC analysis. You can set up a Noise analysis at any time, but you cannot enable the Noise analysis unless an AC analysis is enabled. See [PrimeSim HSPICE AC Analysis](#).

To create a noise analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **noise** radio button.

3. Choose the **Output Type**, which determines the node or branch at which PrimeSim sums the noise:

- To specify a specific voltage, choose **Voltage** as the **Output Type**.

Enter the name of a node in the **Positive Node** and **Negative Node** text boxes, or click the **Select Output Node** button  and click a wire instance in the Schematic Editor.

- To specify a branch current as the noise output, choose **Branch Current** as the **Output Type**.

Enter the name of an instance in the **Output Instance** text box, or click the **Select Output Instance** button  and click an instance in the Schematic Editor.

4. (Optional) Enter the **Source Name** or click the **Select Source** button  to pick from the schematic design.

5. Specify a value for the **Frequency Interval**.

This is the interval at which PrimeSim prints a noise analysis summary.

6. Choose whether or not to include **Port Noise** in the noise analysis.

7. Use the **List Frequencies** option to select how many frequencies to list in the noise summary report: **all**, **none**, or specific **frequencies** in a comma-separated list.

8. Choose whether or not to **List Sources** (on by default). This prints the element noise value to a **.lis** file when the element has multiple noise sources. When this option is enabled, you can enter values for **List Count** and **List Floor**.

9. Choose whether or not to **List Subcircuits**.

10. Choose whether or not to **List Noise Type**.

11. Click **OK** or **Apply**.

Your noise analysis is now set up.

See Also

- [AC Small-Signal and Noise Analysis](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

PrimeSim HSPICE FFT Analysis

Note:

Before enabling and running an FFT analysis, you must first set up and enable a Transient analysis. See [PrimeSim HSPICE Transient Analysis](#).

To create an FFT analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **fft** radio button.
3. Choose a name for the analysis.

The analysis is named "fft" by default. Click the buttons below the **Analysis Name** menu to add a new FFT analysis , make a copy of the currently selected FFT analysis , change the currently selected name , or delete the selected analysis .

4. Choose an **Output Variable**:

- To specify a voltage as the output, click **Voltage** and specify the voltage to use as the output.

The voltage can be a single node or the voltage between two nodes. Enter the node names directly into the **Positive Node** and **Negative Node** text boxes, or click the  **Select in Design** button and select a wire in the Schematic Editor. Specifying a **Negative Node** is optional.

- To specify a current as the output, click **Current** and select an **Output Instance**.

Enter the instance name directly into the **Output Instance** text box, or click the **Select in Design** button  and select an instance in the Schematic Editor.

- To specify a power value as the output type, click **Power** and select an **Output Instance** with the desired power dissipation.

5. Enter values for the **Start Time**, **Stop Time**, and **No. of Points**.

To calculate the FFT, you must include the specified transient analysis time points, as well as the number of points used in the FFT calculation.

6. Choose an output **Format**.

NORM is normalized magnitude (default). **UNORM** is unnormalized magnitude.

7. Choose a **Window Type**.

The following window types are available. If you choose the **GAUSS** or **KAISER** window types, move on to the next step; otherwise, skip to [Step 9](#).

- **RECT**: Simple rectangular truncation window (default).
- **BART**: Bartlett window.
- **HANN**: Hanning window.
- **HAMM**: Hamming window.
- **BLACK**: Blackmann window.
- **HARRIS**: Blackmann-Harris window.
- **GAUSS**: Gaussian window.
- **KAISER**: Kaiser-Bessel window.

8. For the **GAUSS** and **KAISER** windows, enter a value for the **Alpha**, which is used to control highest side-lobe level and bandwidth.

Valid values are 1.0 to 20.0, inclusive. The default is 3.0.

9. (Optional) Specify values for the **Analysis**, **Minimum**, and **Maximum Frequency**.
10. Click **Enable** to enable this analysis as part of your testbench.
11. Click **OK** or **Apply**.

Your FFT analysis is now set up.

See Also

- [.FFT Analysis](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

PrimeSim HSPICE Linear Network Parameter Analysis

A .LIN analysis calculates S (scattering) and noise parameters, as well as additional measurements.

Note:

Before running a .LIN analysis, you must first set up and enable an AC analysis.

See [PrimeSim HSPICE AC Analysis](#).

To create a linear network parameter (lin) analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **lin** radio button.

3. Enter a **Model Name**.

4. Choose an output file **Format**.

- **selem**: S element format
- **citi**: CITI file format

Note:

CITI format is not supported for postprocessing. Use only for exporting to external tools.

- **touchstone**: Touchstone v1.0 file format
- **touchstone2**: Touchstone v2.0 file format

You can also choose to extract the S parameters from your design by clicking **Extract S Parameters**, and include the group delay information by clicking **Group Delay**.

5. Select a **Mixed Mode** value.

6. (Optional) Check **Data Format Use Simulator Default** to use the same data format output that is set in PrimeSim.

7. (Optional) Set the **Frequency Precision**, which is the numerical precision (number of digits) for frequency output in the output files.
8. (Optional) Set the **S Parameter Precision**, which is the numerical precision (number of digits) for S Parameter output in the output files.
9. (Optional) If you also want to perform a noise analysis, click **Perform Noise Analysis** and choose a **Noise Level**.
10. When **Perform Noise Analysis** is enabled, you can use the **List Frequencies** option to select how many frequencies to list in the noise summary report: **all**, **none**, or specific **frequencies** in a comma-separated list.
11. Choose whether or not to **List Sources** (off by default). This prints the element noise value to a **.lis** file when the element has multiple noise sources. When this option is enabled, you can enter values for **List Count** and **List Floor**.
12. Click **Enable** to enable this analysis as part of your testbench.
13. Click **OK** or **Apply**.

Your linear network parameter analysis is now set up.

See Also

- [.LIN Analysis](#) in the *PrimeSim HSPICE User Guide: Signal Integrity Modeling and Analysis*

PrimeSim HSPICE AC Match Analysis

An AC Match analysis determines the combined effects of device variations on the frequency response of a circuit.

Note:

To create an AC match analysis, you must first create an AC analysis, as well as include a model library in your testbench with a Variation block section specified. See [PrimeSim HSPICE AC Analysis](#) for more information on creating AC analyses and [Specifying Model Files](#) for more information on specifying model files.

To create an AC match analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **acmatch** radio button.
3. Choose an output type: **Voltage** or **Current**.
4. Choose a modifier:
 - **Magnitude**
 - **Phase**
 - **Real**
 - **Imaginary**
5. If you chose **Current** as the output type, skip to [Step 7](#). Otherwise, continue to the next step.
6. Choose **Positive Node** and **Negative Node**.

You can enter the name of each node, or click the  button to choose a node in a schematic. Skip to [Step 9](#).

7. Choose an **Output Instance**.

You can enter the name of an instance or click the  button to choose an instance in your schematic.

8. Click **Add** to add the node or instance to the analysis.
9. Specify **Threshold** and **Interval** values.
10. (Optional) To add **Virtual Sensitivity** to the analysis, click the arrow next to the **Virtual Sensitivity** text to expand the section.

Virtual Sensitivity calculates virtual parasitic capacitor sensitivity, which can be used for high-precision (differential) analog circuits and switched capacitor filters. These analog circuits and switched capacitor files are sensitive to layout parasitics, but their values are not known at the pre-layout stage.

Otherwise, skip to step [Step 13](#).

11. Specify a value for the **Sensitivity Threshold**.

12. Choose **Sensitivity Nodes**.

You can enter the name or click the  button to choose a node in your schematic.

13. Click **Enable** to enable this analysis as part of your testbench.

14. Click **OK** or **Apply**.

Your AC match analysis is now set up.

See Also

- [ACMatch Analysis](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

PrimeSim HSPICE DC Match Analysis

A DC Match analysis examines the combined effects of variations of all devices on a specified node voltage or branch current. Groups of matched devices are also identified.

To create a DC match analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **dcmatch** radio button.

3. Choose an output type: **Voltage** or **Current**.

4. If you chose **Current** as the output type, skip to [Step 6](#). Otherwise, continue to the next step.

5. Choose **Positive Node** and **Negative Node**.

You can enter the name of each node, or click the  button to choose a node in a schematic. Skip to [Step 8](#).

6. Choose an **Output Instance**.

You can enter the name of an instance or click the  button to choose an instance in your schematic.

7. Click **Add** to add the node or instance to the analysis.

You can also change the information for a node or instance by selecting the row in the table, changing types, modifiers, or nodes/instances, and clicking **Modify**.

8. Specify **Threshold** and **Interval** values.
9. Click **Enable** to enable this analysis as part of your testbench.
10. Click **OK** or **Apply**.

Your DC match analysis is now set up.

See Also

- [DCMatch Analysis](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

PrimeSim HSPICE Transient Noise Analysis

A transient noise analysis calculates noise statistics and their variation over time for circuits that are driven with non-periodic waveforms. This analysis is very similar to a transient analysis, except that all random noise sources are activated in the signal.

Note:

To create a transient noise analysis, you must first create a transient analysis. See [PrimeSim HSPICE Transient Analysis](#) for more information on creating transient noise analyses.

If Transient Noise and Monte Carlo analyses are set up at the same time, only Monte Carlo results are produced.

To create a transient noise analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
- The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **trannoise** radio button.
3. Choose a **Method** to model the noise contributions in the noise analysis: **Monte Carlo** or **SDE**. You can also choose an empty **Method** (the default), which uses Single Sample Monte Carlo (**Seed** = 2, **Samples** = 1).
4. Choose an **Output Type**: **Node** or **Instance**.
5. If you chose **Instance** as the **Output Type**, skip to [Step 7](#). Otherwise, continue to the next step.
6. Choose **Positive Node** and **Negative Node**.

You can enter the name of each node, or click the  button to choose a node in a schematic. Skip to [Step 8](#).

7. Choose an **Output Instance**.

You can enter a name of an instance or click the  button to choose an instance in your schematic.

8. If you chose the **Monte Carlo** method for performing your transient noise analysis, skip to [Step 10](#). Otherwise, continue to the next step.
9. For **SDE** method, choose **All** or **Value** for **Time Points**.
 If you choose **Value**, enter a value in the **Time Point Value** field. Skip to [Step 13](#).
10. For **Monte Carlo** method, in the **Sampling** section, choose a **Sampling Type**: **Value** or **List**.
 If you choose **Value**, continue to the next step. If you choose **List**, skip to [Step 12](#).
11. Enter values for **Seed** and **Samples** (the number of Monte Carlo samples). The default values are 2 and 1, respectively.
12. Enter values for a **Seed List**. Valid formats include a comma-separated list of values, or a comma-separated list of ranges.
13. (Optional) Enter values for the **Min Frequency**, **Max Frequency**, **Start Time**, and **Scale Factor**.

These parameters calculate the base and maximum frequencies for noise sources, as well as an optional scale factor that can be used to inflate the contributions. The default values are 1/tstop, 1/tstep, and 1, respectively.

- Check **Auto** to allow the simulator to select the **Max Frequency** value,
14. (Optional) Select a value for **Auto Correlation**: **Disabled**, **Enabled**, or **Normalize**. **Normalize** applies normalization over the simulation interval for the calculation.
 15. (Optional) Check **Power Spectral Density** to enable power spectral density calculations.
 16. (Optional) Select noise calculation values for **Phase Noise**. Options are **Disabled**, **Delay-Line based**, and **Phase Detector based**. If you choose an option other than **Disabled**, select a **Noise Reference** by entering the name or clicking the  button to pick from the schematic.
 17. (Optional) Select noise calculation values for **Jitter**. Options are **Disabled**, **Phase Noise based**, and **Phase Detector** based. If you choose an option other than **Disabled**, select a **Noise Reference** by entering the name or clicking the  button to pick from the schematic.
 18. Click **Enable** to enable this analysis as part of your testbench.
 19. Click **OK** or **Apply**.

Your transient noise analysis is now set up.

See Also

- [Transient Noise Analysis](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE Statistical Eye Analysis

A statistical eye analysis creates eye diagrams and measures bit error rates for circuits such as high-speed serial interfaces.

To create a statistical eye analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **stateye** radio button.

3. Click the **Ports** tab of the statistical eye analysis.

See [Ports](#) for information on how to set up this part.

4. (Optional) Set up the **Eye Diagrams** and **Measurements** parts of your statistical eye analysis.

See [Eye Diagrams](#) and [Measurements](#) for information on how to set up these parts.

5. Once you are done setting up all needed statistical eye setup parts, click **Enable** to enable this analysis as part of your testbench.

6. Click **OK** or **Apply**.

Your statistical eye analysis is now set up.

Ports

To add ports to your statistical eye analysis:

1. Ensure the **Ports** tab is selected.

From this tab, you can set up characteristics for the input (incident) and output (probe) ports that are in your design. See [PrimeSim HSPICE Statistical Eye Analysis](#).

2. Choose a port **Type: Probe or Incident**.

Note:

If you want to include eye diagrams or measurements as part of your statistical eye setup, you need to add **Probe** ports. **Incident** ports are not used in the eye diagram or measurement setup.

3. Choose a **Port Instance**.

You can enter a port instance name or click the  button to choose an instance in your schematic.

4. (Optional) Enter a value for **Random jitter**.

This value is an array of real number that specify the standard deviation of the Gaussian random jitter. The array must be in order of the port element index. The default random jitter value is 0.

5. Click **Add** to add the instance to your analysis.

You can also change the information for an instance by selecting the row in the table, changing types, instances, or random jitter values and clicking **Modify**.

6. Enter values or expressions for the following:

- **Eye Width**

The single-bit width of a the incident signal. The default value is 0.

- **Tran Init**

The number of intervals used by the transient analysis to determine the system response. The default value is 60.

- **Max LFSR/PAT Size**

The limit for the number of bits to be examined.

7. Choose the **Number of Edges** to analyze the rising and falling edges individually: **1** or **2**.

If you choose **1**, skip to step [Step 10](#). Otherwise, move on to the next step.

8. Choose an **Edge Type: Symmetrical** or **Asymmetrical**.

9. If you chose **Asymmetrical**, enter values greater than zero for the **Rise Time** and the **Fall Time**, and skip to [Step 12](#). Otherwise, move on to the next step.

10. Enter values for the **Rise/Fall Time**, **Low Voltage Level**, and **High Voltage Level**.

The **Rise/Fall Time** can be a numeric value or a valid expression. The **Low Voltage Level** and **High Voltage Level** can be numeric values.

11. (Optional) Enter values for the **Incident Time Delay** and **Probe Time Delay**.

12. If you do not need to enter information for the **Eye Diagrams** or **Measurements** sections of the statistical eye analysis setup, click **Enable** to enable this analysis as part of your testbench, then click **OK** or **Apply**. Your statistical eye analysis is now set up.

Otherwise, you can continue to the [Eye Diagrams](#) or [Measurements](#) sections to set up the outputs for your statistical eye analysis.

Eye Diagrams

Caution:

You must use WDF output to successfully plot and measure your statistical eye results.

Unlike other analyses in the PrimeWave Design Environment, the statistical eye analysis requires that you specify outputs in conjunction with this analysis. The simulator performs the eye diagram calculations and measurement and not on the raw data as a postprocess as in other analyses.

Note:

The **Ports** setup of your statistical eye analysis is required before setting up any **Eye Diagrams**. Only ports of type "probe" are used in the **Eye Diagrams** setup. See [Ports](#) for information on how to set up ports.

To generate eye diagrams as part of your statistical eye analysis:

1. Ensure the **Eye Diagrams** tab is selected.
See [PrimeSim HSPICE Statistical Eye Analysis](#).
2. Choose one of the following types of eyes to add:
 - **eyeT**
 - **eyeV**
 - **eye**
 - **BER**
 - **bathtubV**
 - **bathtubT**
3. Choose a port for the chosen eye type from the **Port** menu.
4. If you chose **eye** or **BER** as the eye type, skip to step [Step 6](#). Otherwise, move on to the next step.
5. For **eyeT** or **bathtubT** eyes, specify a time value or expression in the **Time** field. Otherwise, for **eyeV** or **bathtubV** eyes, specify a voltage value in the **Voltage** field.
6. Click **Add** to add the eye to your statistical eye analysis setup.

You can also change the information for an eye by selecting the row in the table, changing types, ports, or time/voltage values (if applicable) and clicking **Modify**.

7. (Optional) Enter values for the **T Resolution** and **V Resolution**.
The **T Resolution** specifies the image resolution of the time axis and has a default value of 200. The **V Resolution** specifies the image resolution of the voltage axis and has a default value of 200.
8. (Optional) Enter a value for the **Voltage Display Range**.
The **Voltage Display Range** specifies the voltage display range. If not specified, the simulator will automatically determine the value for this option.
9. (Optional) Click **Add to Outputs** to add the specified eyes to the outputs in the PrimeWave Design Environment.

The **Add to Outputs** option is enabled by default.

10. If you do not need to enter information for the **Measurements** section of the statistical eye analysis setup, click **Enable** to enable this analysis as part of your testbench, then click **OK** or **Apply**. Your statistical eye analysis is now set up.

Otherwise, you can continue to the **Measurements** section specify measurements for your statistical eye analysis.

Measurements

Note:

The **Ports** setup of your statistical eye analysis is required before setting up any **Measurements**. Only ports of type "probe" are used in the **Measurements** setup. See [Ports](#) for information on how to set up **Ports**.

To add measurements to your statistical eye analysis:

1. Ensure the **Measurements** tab is selected.

See [PrimeSim HSPICE Statistical Eye Analysis](#).

2. Enter a name for the measurement you want to add in the **Name** field.
3. Choose one of the following measurement types to add:

- **Veye**

Measures the vertical eye opening.

- **Heye**

Measures the horizontal eye opening.

- **WorstBits**

Measures the worst eye opening at a given time and state of the bit pattern. When **Time** is not specified, the worst bit pattern at the maximum vertical eye opening point is measured.

- **Eye**

Calculates the probability at a specific time.

- **BER**

Measures the bit error rate.

4. Choose a port for the chosen measurement type from the **Port** menu.

5. If you chose the **Eye** or **BER** measurement types, skip to [Step 7](#). Otherwise, continue to the next step.
 6. Enter values for the following measurement types as necessary:
 - **Veye:** (Optional) Enter the **Time** and **Tolerance** values.
 These values can be numeric values or valid expressions.
 - **Heye:** (Optional) Enter the **Voltage** and **Tolerance** values.
 These values can be numeric values or valid expressions.
 - **WorstBits:** (Optional) Enter a value for **Time**, and click **State** to select the logic high state. This specifies the port, time, and state of the bit pattern, which produces the worst eye opening.
- Skip to [Step 7.a](#).
7. (Optional) Enter values for the **Time** and **Voltage**.
 These values can be numeric values or valid expressions.
 - a. Click **Add** to add the measurement to your statistical eye analysis setup.
 You can also change the information for a measurement by selecting the row in the table, changing types, ports, or time/voltage/tolerance values (if applicable) and clicking **Modify**.
 8. (Optional) Click **Add to Outputs** to add the specified measurements to the outputs in the PrimeWave Design Environment. The **Add to Outputs** option is enabled by default.
 9. Click **Enable** to enable this analysis as part of your testbench, then click **OK** or **Apply**. Your statistical eye analysis is now set up.

See Also

- [Statistical Eye Analysis](#) in the *PrimeSim HSPICE User Guide: Signal Integrity Modeling and Analysis*

PrimeSim HSPICE Loop Stability Analysis (LSTB)

To create an LSTB analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
 The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **Istb** radio button.
3. Select an **Analysis Name** from the drop-down menu. The PrimeSim integration with the PrimeWave Design Environment supports multiple LSTB analyses. You can **Add**, **Copy**, **Rename**, or **Delete** the selected analysis. Unique LSTB probes and measurements are generated for each LSTB analysis.
4. Specify a voltage **Mode** for the stability analysis: **Single**, **Differential**, or **Common**.
5. If you choose the **Single** mode, enter a voltage source in the **Voltage Source 1** text box, or click the  button to choose a voltage source in the schematic.
 If you choose a **Differential** or **Common** modes, enter voltage sources for **Voltage Source 1** and **Voltage Source 2**, or click the  buttons to choose voltage sources in the schematic.
6. Click **Enable** to enable this analysis as part of your testbench.
7. Click **OK** or **Apply**.

Your LSTB analysis is now set up.

See Also

- [.LSTB Analysis](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

PrimeSim HSPICE RF Harmonic Balance Analysis

To create a harmonic balance analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
 The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hb** radio button.
3. Choose or enter the number of **Fundamental Tones**.

Tone and **No.** of Harmonic fields are added or subtracted from the harmonic balance analysis setup form depending on how many **Fundamental Tones** you choose. You can include up to 7 fundamental tones, and the default number of fundamental tones is 2.

4. For each **Fundamental Tone** you create, enter a value for the **Tone** and corresponding number of harmonics, unless you enter a value for **Max Intermodulation**.
5. Choose or enter a value for the **Max Intermodulation**, which is a product order that you can specify in the analysis spectrum.
You can specify up to 99 for the **Max Intermodulation** value. The default value is 5.
6. (Optional) If you want to include a sweep of your harmonic balance analysis, click the arrow next to the **Sweep** text to expand the **Sweep** parameters section and continue to the next step. Otherwise, skip to [Step 12](#).
7. Choose a **Variable Name** to include in the sweep.

Only previously specified variables are listed in the **Variable Name** menu. See [Adding and Editing Design Variables](#) for information on creating variables.

8. Choose a unit of measure for the **Variable Units**: "", **dBm**, or **W**.
The default is **dBm** units.
9. Choose one of the following for **Sweep Type**:
 - **Linear**
 - **Octave**
 - **Decade**
 - **Points of Interest**
10. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 12](#). Otherwise, continue to the next step.
11. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
12. Click **Enable** to enable this analysis as part of your testbench.
13. Click **OK** or **Apply**.

Your harmonic balance analysis is now set up.

See Also

- [Harmonic Balance Analysis](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Harmonic Balance Oscillator Analysis

To create a harmonic balance oscillator analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
- The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hbosc** radio button.
3. Enter values in the **Oscillator Freq** and **No. of Harmonics** fields.

The oscillator frequency is in units of Hertz (Hz).

4. Choose a **Positive Node** and **Negative Node**.

You can enter the name of each node, or click the  button to choose a node in a schematic.

5. Enter a value for the **Initial Probe Voltage**.

You might want to try a value that is one-half of the supply voltage.

6. Enter a value for **Subharmonics**.

7. Enter a value for the **Initial Transient** (default is 0). Check **Use Initial Conditions** if necessary.

8. (Optional) If you want to **Add Frequency Search Points**, click the arrow to expand that parameter section and continue to the next step. Otherwise, skip to [Step 11](#).

9. Choose a **Number** of search points and a **Min (Hz)** and **Max (Hz)**.

10. Select a **Stability** value. Choices include **Single point**; **Single point, no estimates**; **Single point, no nonlinear analysis**; **Multipoint**; and **Multipoint, no nonlinear analysis**.

11. (Optional) If you want to include a sweep of your harmonic balance analysis, click the arrow next to the **Sweep** text to expand the **Sweep** parameters section and continue to the next step. Otherwise, skip to [Step 17](#).
12. Choose a **Variable Name** to include in the sweep.
Only previously specified variables are listed in the **Variable Name** menu. See [Adding and Editing Design Variables](#) for information on creating variables.
13. Choose a unit of measure for the **Variable Units**: "", **dBm**, or **W**.
The default is **dBm** units.
14. Choose one of the following for **Sweep Type**:
 - **Linear**
 - **Octave**
 - **Decade**
 - **Points of Interest**
15. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 17](#). Otherwise, continue to the next step.
16. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
17. Click **Enable** to enable this analysis as part of your testbench.
18. Click **OK** or **Apply**.

Your harmonic balance oscillator analysis is now set up.

See Also

- [Oscillator Analysis \(.HBOSC and .SNOSC\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Harmonic Balance AC Analysis

Note:

To create a harmonic balance AC analysis, you must first run a harmonic balance or harmonic balance oscillator analysis. See [PrimeSim HSPICE RF Harmonic Balance Analysis](#) or [PrimeSim HSPICE RF Harmonic Balance Oscillator Analysis](#) for more information on creating those analyses.

To create a harmonic balance AC analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hbac** radio button.
3. Choose one of the following for **Sweep Type**:
 - **Linear**
 - **Octave**
 - **Decade**
 - **Points of Interest**
4. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 6](#). Otherwise, continue to the next step.
5. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
6. Click **Enable** to enable this analysis as part of your testbench.
7. Click **OK** or **Apply**.

Your harmonic balance AC analysis is now set up.

See Also

- [Multitone Harmonic Balance AC Analysis \(.HBAC\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Harmonic Balance Transfer Function Analysis

Note:

To create a harmonic balance transfer function analysis, you must first run a harmonic balance or harmonic balance oscillator analysis. See [PrimeSim HSPICE RF Harmonic Balance Analysis](#) for more information on creating those analyses.

To create a harmonic balance transfer function analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible and click the **hbxf** radio button.
3. Choose an output type: **Voltage** or **Branch Current**.
4. If you chose **Branch Current** as the output type, skip to [Step 6](#). Otherwise, continue to the next step.
5. Choose a **Positive Node** and **Negative Node**.

Note:

Specifying a **Negative Node** is optional. If not specified, PrimeSim assumes the second node is ground.

You can enter the name of each node, or click  to choose a node in a schematic. Skip to [Step 7](#).

6. Choose an **Output Instance**.

You can enter a name of an instance or click  to choose an instance in your schematic.

7. Choose one of the following for **Sweep Type**:

- **Linear**
- **Octave**
- **Decade**
- **Points of Interest**

8. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 10](#). Otherwise, continue to the next step.
9. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
10. Click **Enable** to enable this analysis as part of your testbench.

All sources are saved by the simulator when this analysis is enabled.

11. Click **OK** or **Apply**.

Your harmonic balance transfer function analysis is now set up.

See Also

- [Multitone Harmonic Balance Transfer Function Analysis \(.HBXF\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Harmonic Balance Noise Analysis

Note:

A harmonic balance analysis must be run before running this analysis. See the [PrimeSim HSPICE RF Harmonic Balance Analysis](#) section for more information on creating a harmonic balance analysis.

To create a harmonic balance noise analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hbnoise** radio button.
3. Choose an output type: **Voltage** or **2-terminal Instance**.
4. If you chose **Voltage** as the output type, choose a **Positive Node** and **Negative Node**. Otherwise, move on to the next step.

Note:

Specifying a **Negative Node** is optional. If not specified, PrimeSim assumes the second node is ground.

You can enter the name of each node, or click the  button to choose a node in a schematic.

5. If you chose **2-terminal Instance** as the output type, choose an **Output Instance**. Otherwise, move on to the next step.
6. Enter the name of the **Input Source**, or click the  button to choose an input source in your schematic.

Note:

Along with the .HBNOISE statement in your netlist, the onoise, ssnf, dsnf, inoise and nf are also probed automatically if this analysis is enabled with an **Input Source** specified.

7. Choose one of the following for **Sweep Type**:
 - **Linear**
 - **Octave**
 - **Decade**
 - **Points of Interest**
8. If you choose **Points of Interest**, enter one or more values separated by commas for the number of points, and skip to [Step 12](#). Otherwise, continue to the next step.
9. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
10. Use the **List Frequencies** option to select how many frequencies to list in the noise summary report: **all**, **none**, or specific **frequencies** in a comma-separated list.
11. Choose whether or not to **List Sources** (off by default). This prints the element noise value to a **.lis** file when the element has multiple noise sources. When this option is enabled, you can enter values for **List Count** and **List Floor**.
12. (Optional) If you want to define the output frequency band of your harmonic balance noise analysis, click the arrow next to the **Define Output Frequency Band** text to expand that parameters section and continue to the next step. Otherwise, skip to [Step 14](#).

Note:

If the number of tones can be obtained from the harmonic balance analysis, these parameters are not displayed.

13. Choose or enter values for the following parameters, which are index terms that define the output frequency band:
 - **No. Tones**
 - **Tone Coefficients**
 - **IFB Coefficient**
14. Click **Enable** to enable this analysis as part of your testbench.
15. Click **OK** or **Apply**.

Your harmonic balance noise analysis is now set up.

Note:

A validation check is performed after the harmonic balance noise analysis is set up to ensure that the number of tones for the OFB index matches the number of tones in the harmonic balance analysis.

See Also

- [Multitone Harmonic Balance Noise \(.HBNOISE\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Shooting Newton Analysis

To create a shooting Newton analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **sn** radio button.
 3. Choose the domain in which you want to create the shooting Newton analysis: **Frequency** or **Time**.
 4. If you choose **Frequency**, enter values for the **Fundamental Freq** and **No. of Harmonics** and skip to step [Step 6](#). Otherwise, move on to the next step.
 5. If you choose the **Time** domain, enter values for the **Period** and **Time Resolution**.
 6. Enter values for the **Initial Transient** and **Maximum Initial Cycles**.
 7. (Optional) If you want to include a sweep of your shooting Newton analysis, click the arrow next to the **Sweep** text to expand the **Sweep** parameters section and continue to the next step. Otherwise, skip to [Step 13](#).
 8. Choose a **Variable Name** to include in the sweep.
- Only previously specified variables are listed in the **Variable Name** menu. See [Adding and Editing Design Variables](#) for information on creating variables.
9. Choose a unit of measure for the **Variable Units**: **dBm** or **W**.

The default is **dBm** units.

10. Choose one of the following for **Sweep Type**:
 - **Linear**
 - **Octave**
 - **Decade**
 - **Points of Interest**
11. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 12](#). Otherwise, continue to the next step.
12. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
13. Click **Enable** to enable this analysis as part of your testbench.
14. Click **OK** or **Apply**.

Your shooting Newton analysis is now set up.

See Also

- [Shooting Newton Steady-State Time Domain Analysis \(.SN\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Shooting Newton Oscillator Analysis

To create a shooting Newton oscillator analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snosc** radio button.
3. Choose the domain in which you want to create the shooting Newton oscillator analysis: **Frequency** or **Time**.
4. If you choose **Frequency**, enter values for the **Oscillator Freq** and **No. of Harms** and skip to step [Step 6](#). Otherwise, move on to the next step.

5. If you choose the **Time** domain, enter values for the **Period** and **Time Resolution**.

6. Choose an **Oscillator Node**.

You can enter the name a node, or click the  button to choose a node in your schematic.

7. Enter values for the **Initial Transient** and **Maximum Initial Cycles**.

8. (Optional) If you want to include a sweep of your shooting Newton oscillator analysis, click the arrow next to the **Sweep** text to expand the **Sweep** parameters section and continue to the next step. Otherwise, skip to [Step 14](#).

9. Choose a **Variable Name** to include in the sweep.

Only previously specified variables are listed in the **Variable Name** menu. See [Adding and Editing Design Variables](#) for information on creating variables.

10. Choose a unit of measure for the **Variable Units**: **dBm** or **W**.

The default is **dBm** units.

11. Choose one of the following for **Sweep Type**:

- **Linear**
- **Octave**
- **Decade**
- **Points of Interest**

12. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 14](#). Otherwise, continue to the next step.

13. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.

14. Click **Enable** to enable this analysis as part of your testbench.

15. Click **OK** or **Apply**.

Your shooting Newton oscillator analysis is now set up.

See Also

- [Oscillator Analysis \(.HBOSC and .SNOSC\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Shooting Newton AC Analysis

Note:

To create a shooting Newton AC analysis, you must first run a shooting Newton or shooting Newton oscillator analysis. See [PrimeSim HSPICE RF Shooting Newton Analysis](#) or [PrimeSim HSPICE RF Shooting Newton Oscillator Analysis](#) for more information on creating those analyses.

To create a shooting Newton AC analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snac** radio button.
3. Choose one of the following for **Sweep Type**:
 - **Linear**
 - **Octave**
 - **Decade**
 - **Points of Interest**
4. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 6](#). Otherwise, continue to the next step.
5. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
6. Click **Enable** to enable this analysis as part of your testbench.
7. Click **OK** or **Apply**.

Your shooting Newton AC analysis is now set up.

See Also

- [Shooting Newton AC Analysis \(.SNAC\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Shooting Newton Transfer Function Analysis

Note:

To create a shooting Newton transfer function analysis, you must first run a shooting Newton or shooting Newton oscillator analysis. See [PrimeSim HSPICE RF Shooting Newton Analysis](#) for more information on creating those analyses.

To create a shooting Newton transfer function analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snxf** radio button.
3. Choose an output type: **Voltage** or **Branch Current**.
4. If you choose **Branch Current** as the output type, skip to [Step 6](#). Otherwise, continue to the next step.
5. Choose a **Positive Node** and **Negative Node**.

You can enter the name of each node, or click the  button to choose a node in a schematic. Skip to [Step 7](#).

6. Choose an **Output Instance**.

You can enter a name of an instance or click the  button to choose an instance in your schematic.

7. Choose one of the following for **Sweep Type**:
 - **Linear**
 - **Octave**
 - **Decade**
 - **Points of Interest**
8. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 10](#). Otherwise, continue to the next step.

9. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
10. Click **Enable** to enable this analysis as part of your testbench.
 All sources are saved by the simulator when this analysis is enabled.
11. Click **OK** or **Apply**.
 Your shooting Newton transfer function analysis is now set up.

See Also

- [Shooting Newton Transfer Function Analysis \(.SNXF\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Shooting Newton Noise Analysis

Note:

A shooting Newton analysis must be run before running this analysis. See the [PrimeSim HSPICE RF Shooting Newton Analysis](#) section for more information on creating a shooting Newton analysis.

To create a shooting Newton noise analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snnoise** radio button.
3. Choose an output type: **Voltage** or **2-terminal Instance**.
4. If you chose **Voltage** as the output type, choose a **Positive Node** and **Negative Node**. Otherwise, move on to the next step.

Note:

Specifying a **Negative Node** is optional. If not specified, PrimeSim assumes the second node is ground.

You can enter the name of each node, or click the  button to choose a node in a schematic.

5. If you chose **2-terminal Instance** as the output type, choose an **Output Instance**. Otherwise, move on to the next step.
6. Enter the name of the **Input Source**, or click the  button to choose an input source in your schematic.

Note:

Along with the `.SNNOISE` statement in your netlist, the `onoise`, `ssnf`, `dsnf`, `inoise` and `nf` are also probed automatically if this analysis is enabled with an **Input Source** specified.

7. Choose one of the following for **Sweep Type**:
 - **Linear**
 - **Octave**
 - **Decade**
 - **Points of Interest**
8. If you choose **Points of Interest**, enter one or more values separated by commas for the number of **Points**, and skip to [Step 12](#). Otherwise, continue to the next step.
9. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
10. Use the **List Frequencies** option to select how many frequencies to list in the noise summary report: **all**, **none**, or specific **frequencies** in a comma-separated list.
11. Choose whether or not to **List Sources** (off by default). This prints the element noise value to a `.lis` file when the element has multiple noise sources. When this option is enabled, you can enter values for **List Count** and **List Floor**.
12. (Optional) If you want to define the output frequency band of your shooting Newton noise analysis, click the arrow next to the **Define Output Frequency Band** text to expand that parameters section and continue to the next step. Otherwise, skip to [Step 14](#).

Note:

If the number of tones can be obtained from the shooting Newton analysis, these parameters are not displayed.

13. Choose or enter values for the **Harmonic Multiplier** and **IFB Coefficients**, which are index terms that define the output frequency band.
- The **Harmonic Multiplier** must be a number between 1 and the number of tones in the shooting Newton analysis.

14. Click **Enable** to enable this analysis as part of your testbench.

15. Click **OK** or **Apply**.

Your shooting Newton noise analysis is now set up.

Note:

A validation check is performed after the shooting Newton noise analysis is set up to ensure that the number of tones for the OFB index matches the number of harmonics in the shooting Newton noise analysis.

See Also

- [Shooting Newton Noise Analysis \(.SNNOISE\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Shooting Newton with Fourier Transform Analysis

Note:

To create a shooting Newton with Fourier transform analysis, you must first run a shooting Newton or shooting Newton oscillator analysis. See [PrimeSim HSPICE RF Shooting Newton Analysis](#) or [PrimeSim HSPICE RF Shooting Newton Oscillator Analysis](#) for more information on creating those analyses.

To create a shooting Newton fourier transform (SNFT) analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snft** radio button.
3. Choose an **Analysis Name**.

The analysis is named **snft** by default. Click the buttons below the **Analysis Name** menu to add a new SNFT analysis  , make a copy of the currently selected SNFT analysis  , change the currently selected name  , or delete the selected analysis  .

Multiple SNFT analyses are allowed at once.

4. Choose an **Output Variable**:

- To specify a voltage as the output, click **Voltage** and specify the voltage to use as the output.

The voltage can be a single node or the voltage between two nodes. Enter the node names directly into the **Positive Node** and **Negative Node** text boxes, or click the



Select in Design button , and select a wire in the Schematic Editor. Specifying a **Negative Node** is optional.

- To specify a current as the output, click **Current**, and select an **Output Instance**.

Enter the instance name directly into the **Output Instance** text box or click the



Select in Design button , and select an instance in the Schematic Editor.

- To specify a power value as the output type, click **Power** and select an **Output Instance** with the desired power dissipation.

Enter the instance name directly into the **Output Instance** text box or click the

Select in Design button , and select an instance in the Schematic Editor.

5. Enter values for the **Start Time**, **Stop Time**, and **No. of Points**.

To calculate the SNFT, you must include the specified transient analysis time points, as well as the number of points used in the SNFT calculation.

6. Choose an output **Format**.

NORM is normalized magnitude (default). **UNORM** is unnormalized magnitude.

7. Choose a **Window Type**.

The following window types are available. If you choose the **GAUSS** or **KAIser** window types, move on to the next step; otherwise, skip to [Step 9](#).

- RECT**: Simple rectangular truncation window (default).
- BART**: Bartlett window.
- HANN**: Hanning window.
- HAMM**: Hamming window.
- BLACK**: Blackmann window.
- HARRIS**: Blackmann-Harris window.

PrimeWave Design Environment™ User Guide
 S-2021.09

175

- **GAUSS**: Gaussian window.
 - **KAISER**: Kaiser-Bessel window.
8. If necessary, enter a value for the **Alpha**, which is used to control highest side-lobe level and bandwidth for the GAUSS and KAISER windows.
 Valid values are 1.0 to 20.0, inclusive. The default is 3.0.
 9. (Optional) Specify values for the **Analysis Frequency**, **Minimum Frequency**, and **Maximum Frequency**.
 10. Click **Enable** to enable this analysis as part of your testbench.
 11. Click **OK** or **Apply**.

Your shooting Newton with Fourier transform analysis is now set up.

See Also

- [Shooting Newton with Fourier Transform Analysis \(.SNFT\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Phase Noise Analysis

Note:

To create a phase noise analysis, you must first run a harmonic balance oscillator analysis or shooting Newton oscillator analysis. See [PrimeSim HSPICE RF Harmonic Balance Oscillator Analysis](#) or [PrimeSim HSPICE RF Shooting Newton Oscillator Analysis](#) for more information on creating those analyses.

To create a phase noise analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **phasenoise** radio button.
3. Choose an output type: **Voltage** or **Branch Current**.

4. If you choose **Branch Current** as the output type, skip to [Step 6](#). Otherwise, continue to the next step.

5. Choose a **Positive Node** and **Negative Node**.

You can enter the name of each node, or click the  button to choose a node in a schematic. Skip to [Step 7](#).

6. Choose an **Output Instance**.

You can enter a name of an instance or click the  button to choose an instance in your schematic.

7. Choose one of the following for **Sweep Type**:

- **Linear**
- **Octave**
- **Decade**
- **Points of Interest**

8. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 10](#). Otherwise, continue to the next step.

9. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.

10. Choose one of the following algorithm methods from the **AlgoMethod** menu:

- **Nonlinear Perturbation**
- **Periodic AC**
- **Broadband Phase Noise**

11. Choose or enter a value for the **Carrier Index**.

12. Use the **List Frequencies** option to select how many frequencies to list in the noise summary report: **all**, **none**, or specific **frequencies** in a comma-separated list.

13. Choose whether or not to **List Sources** (on by default). This prints the element noise value to a **.lis** file when the element has multiple noise sources. When this option is enabled, you can enter values for **List Count** and **List Floor**.

14. (Optional) Check **Spurious Analysis** to included a spurious analysis with your phase noise analysis.

15. Check **Enable** to enable this analysis as part of your testbench.

Note:

When the analysis is enabled, the `listfreq=all` `listsources=yes` analysis parameters are always used in the netlist.

16. Click **OK** or **Apply**.

Your phase noise analysis is now set up.

See Also

- [Introduction to Phase Noise Analysis \(.PHASENOISE\)](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Periodic Time-Dependent Noise Analysis

Note:

A shooting Newton analysis must be run before running this analysis. See the [PrimeSim HSPICE RF Shooting Newton Analysis](#) section for more information on creating a shooting Newton analysis.

To create a periodic time-dependent noise analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
- The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **ptdnoise** radio button.
3. Choose an **Output Type**: **Voltage** or **2-terminal Instance**.
4. If you chose **Voltage** as the output type, choose a **Positive Node** and **Negative Node**. Otherwise, move on to the next step.

Note:

Specifying a **Negative Node** is optional. If not specified, PrimeSim assumes the second node is ground.

You can enter the name of each node, or click the  button to choose a node in a schematic.

5. If you chose **2-terminal Instance** as the output type, choose an **Output Instance**. Otherwise, move on to the next step.

6. In the **Time Value** section choose a **Type: Time, Measurement, Threshold, or Sweep**.

If you chose **Time**, continue on to the next step.

If you chose **Measurement**, skip to [Step 8](#).

If you chose **Threshold**, skip to [Step 9](#).

If you chose **Sweep**, skip to [Step 10](#).

7. For **Time**, enter a **Value**. Skip to [Step 11](#).

8. For **Measurement**, enter a **Name**. Skip to [Step 11](#).

9. For **Threshold**, enter the following, then skip to [Step 11](#):

- a. **Signal**. You can enter the name of the signal, or click the  button to choose a signal in a schematic.
- b. **Rise** or **Fall**. Triggers on the rising edge or falling edge of the signal.
- c. **Voltage**.

When you choose the threshold option and enter a signal and threshold, the PrimeWave Design Environment autogenerated the `.measure` statements for jitter, integ, and slewrate calculations. You must enable **Plot Waveforms/Evaluate** for these measurements in the Output table in the main PrimeWave Design Environment window.

10. For **Sweep**, choose a **Time Sweep Type (Linear, Octave, or Decade)**, then enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the time sweep.

11. Enter a value for **Delta**. This is a time value used to determine the slew rate of the time-domain output signal.

12. Choose a **Frequency Sweep Type**:

- **Linear**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
- **Octave**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
- **Decade**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
- **Points of Interest**: Specify the **Points** values separated by commas.

13. (Optional) Enter a value for **Side Band** to control the number of sidebands used for noise source processing. If left unspecified, PrimeSim defaults to 80 or nharms, whichever is less. In some instances when noise contribution is from very high frequencies, a larger value might be required.
14. Use the **List Frequencies** option to select how many frequencies to list in the noise summary report: **all**, **none**, or specific **frequencies** in a comma-separated list.
15. Choose whether or not to **List Sources** (off by default). This prints the element noise value to a `.lis` file when the element has multiple noise sources. When this option is enabled, you can enter values for **List Count** and **List Floor**.
16. Select whether to **Add Measure** (on by default) to write a `.measure` statement to the final netlist. Enter **From** and **To** values.

For example: `.measure ptdnoise strobejit strobejitter onoise from=1e4 to=1e10`

...where `1e4` and `1e10` are the user-defined values in the **From** and **To** fields.

17. Click **Enable** to enable this analysis as part of your testbench.

Note:

When the analysis is enabled, the `listfreq=all` `listsources=yes` analysis parameters are always used in the netlist.

18. Click **OK** or **Apply**.

Your periodic time-dependent noise analysis is now set up.

See Also

- [Introduction to Periodic Time-Dependent Noise Analysis](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Envelope Analysis

To create an envelope analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **env** radio button.
 3. Choose or enter the number of **Fundamental Tones**.
- Tone** and **No.** of Harmonic fields are added or subtracted from the harmonic balance analysis setup form depending on how many **Fundamental Tones** you choose. You can include up to 7 fundamental tones, and the default number of fundamental tones is 2.
4. For each **Fundamental Tone** you create, enter a value for the **Tone** and corresponding number of harmonics.
 5. Enter values for the **Time Step** and **Stop Time** fields.
 6. Click **Enable** to enable this analysis as part of your testbench.
 7. Click **OK** or **Apply**.

Your envelope analysis is now set up.

See Also

- [Envelope Analysis](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Envelope Oscillator Analysis

To create an envelope oscillator analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **envosc** radio button.
 3. Enter values in the **Carrier Frequency** and **No. of Harmonics** fields.
- The oscillator frequency is in units of Hertz (Hz).
4. Choose a **Positive Node** and **Negative Node**.

You can enter the name of each node, or click the  button to choose a node in a schematic.

5. Enter a value for the **Initial Probe Voltage**.

You might want to try a value that is one-half of the supply voltage.

6. Enter values for **Time Step** and **Stop Time**.

7. (Optional) If you want to **Add Frequency Search Points**, click the arrow to expand that parameter section and continue to the next step. Otherwise, skip to [Step 9](#).

8. Choose a **Number** of search points and a **Min (Hz)** and **Max (Hz)**.

9. Click **Enable** to enable this analysis as part of your testbench.

10. Click **OK** or **Apply**.

Your envelope oscillator analysis is now set up.

See Also

- [Envelope Analysis](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE RF Envelope Fast Fourier Transform Analysis

Note:

To create an envelope Fast Fourier transform analysis, you must first run an envelope or envelope oscillator analysis. See [PrimeSim HSPICE RF Envelope Analysis](#) or [PrimeSim HSPICE RF Envelope Oscillator Analysis](#) for more information on creating those analyses.

To create an envelope fast Fourier transform analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **envfft** radio button.

3. Choose an **Analysis Name**.

The analysis is named **envfft** by default. Click the buttons below the **Analysis Name** menu to add a new analysis , make a copy of the currently selected analysis , change the currently selected name , or delete the selected analysis .

Multiple analyses are allowed at once.

4. Choose an **Output Variable**:

- To specify a voltage as the output, click **Voltage** and specify the voltage to use as the output.

The voltage can be a single node or the voltage between two nodes. Enter the node names directly into the **Positive Node** and **Negative Node** text boxes, or click the  **Select in Design** button Negative Node is optional.

- To specify a current as the output, click **Current**, and select an **Output Instance**.

Enter the instance name directly into the **Output Instance** text box or click the  **Select in Design** button 

- To specify a power value as the output type, click **Power** and select an **Output Instance** with the desired power dissipation.

Enter the instance name directly into the **Output Instance** text box or click the  **Select in Design** button 

5. Enter a value for the **No. of Points**.

To calculate the envelope fast Fourier transform analysis, you must include the number of points used in the envelope fast Fourier transform calculation.

6. Choose an output **Format**.

NORM is normalized magnitude (default). **UNORM** is unnormalized magnitude.

7. Choose a **Window Type**.

The following window types are available. If you choose the **GAUSS** or **KAISER** window types, move on to the next step; otherwise, skip to [Step 9](#).

- RECT**: Simple rectangular truncation window (default).
- BART**: Bartlett window.
- HANN**: Hanning window.

- **HAMM**: Hamming window.
 - **BLACK**: Blackmann window.
 - **HARRIS**: Blackmann-Harris window.
 - **GAUSS**: Gaussian window.
 - **KAISER**: Kaiser-Bessel window.
8. If necessary, enter a value for the **Alpha**, which is used to control highest side-lobe level and bandwidth for the GAUSS and KAISER windows.
Valid values are 1.0 to 20.0, inclusive. The default is 3.0.
9. Click **Enable** to enable this analysis as part of your testbench.
10. Click **OK** or **Apply**.

Your envelope fast Fourier transform analysis is now set up.

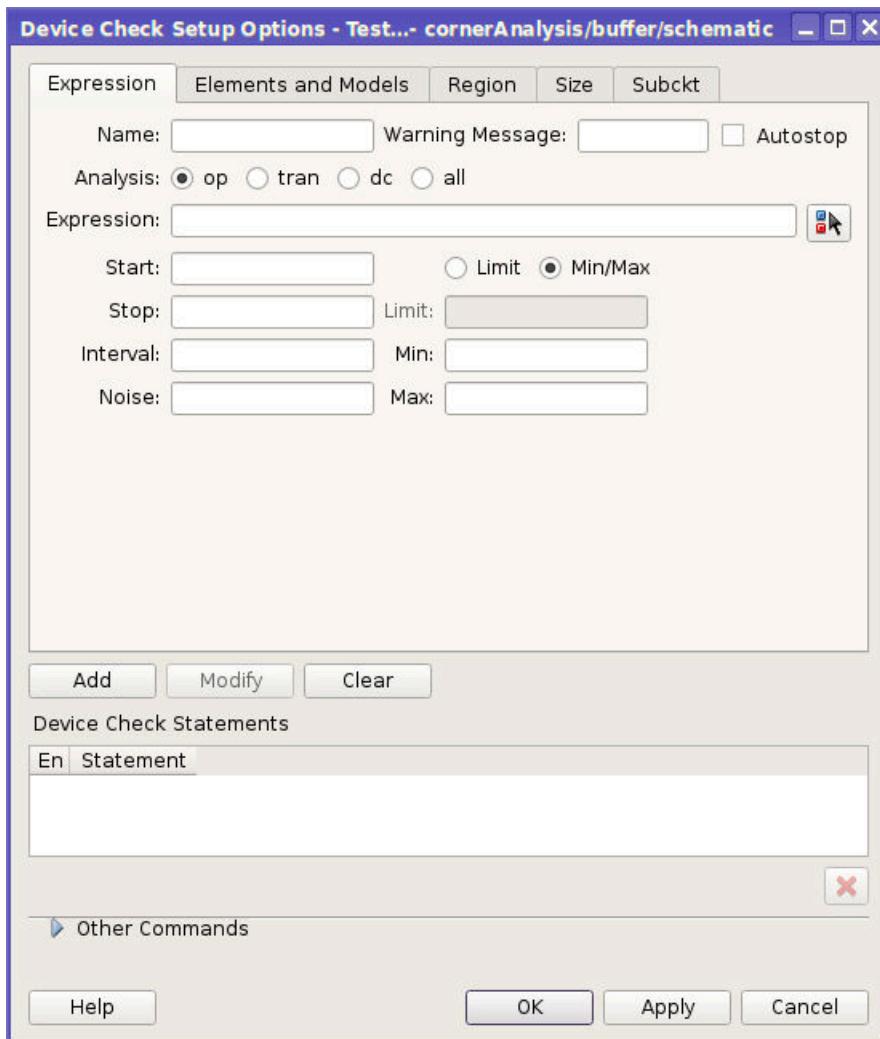
See Also

- [Envelope Analysis](#) in the *PrimeSim HSPICE User Guide: Advanced Analog Simulation and Analysis*

PrimeSim HSPICE Bias Check Analyses

To create a bias check analysis:

1. Choose **Setup > Device Checks** from the PrimeWave Design Environment main menu bar.
The **Device Check Setup Options** dialog box opens.



2. Set up the bias check statements in the tabs of the **Device Check Setup Options** dialog box: **Expression**, **Elements and Models**, **Region**, **Size**, and **Subckt**.
3. Click **OK** or **Apply**.

Your Bias Check analysis is now set up.

Results from the bias check analysis can be accessed using **Results > Print > Simulation Check Viewer**. See [Using the Simulation Check Viewer](#) for information on viewing the results of this check.

See Also

- [Dynamic Check Using the .BIASCHK Statement](#) in the *PrimeSim HSPICE User Guide: Basic Simulation and Analysis*

6

Setting Up PrimeSim SPICE Analyses

This chapter contains information on how to set up and enable PrimeSim SPICE General and RF analyses.

Note:

For more information on a particular PrimeSim analysis, enter `primesim -webhelp` on the PrimeSim command line. Once the PrimeSim help opens, search for the name of an analysis to find information regarding usage, syntax, and options.

Note:

If you want to use an analysis card for any applicable PrimeSim analyses, click the **Use <analysis> card** option at the bottom of the analysis setup page, and enter any needed analysis commands in the text box. When the **Use <analysis> card** option is enabled, any analysis settings you set up on the analysis setup page are disabled and have no effect on netlisting.

Any analysis statements entered in the analysis card text box are netlisted as-is with no error checking. If you add any additional supported sweeps (parameter, .data, or Monte Carlo, for example), the simulation runs, but all internal sweeps are blocked.

To create or edit an analysis, choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The following analyses are available in the PrimeSim SPICE integration:

- [PrimeSim SPICE Transient Analysis](#)
- [PrimeSim SPICE Operating Point Analysis](#)
- [PrimeSim SPICE AC Analysis](#)
- [PrimeSim SPICE DC Analysis](#)
- [PrimeSim SPICE Noise Analysis](#)
- [PrimeSim SPICE Linear Network Parameter Analysis](#)

- [PrimeSim SPICE Loop Stability Analysis \(LSTB\)](#)
- [PrimeSim SPICE AC Match Analysis](#)
- [PrimeSim SPICE DC Match Analysis](#)
- [PrimeSim SPICE PZ Analysis](#)
- [PrimeSim SPICE XF Analysis](#)
- [PrimeSim SPICE Sensitivity Analysis](#)
- [PrimeSim SPICE Include Analysis](#)
- [PrimeSim SPICE RF Shooting Newton Analysis](#)
- [PrimeSim SPICE RF Shooting Newton AC Analysis](#)
- [PrimeSim SPICE RF Shooting Newton Transfer Function Analysis](#)
- [PrimeSim SPICE RF Shooting Newton Noise Analysis](#)
- [PrimeSim SPICE RF Shooting Newton Linear Analysis](#)
- [PrimeSim SPICE RF Shooting Newton Stability Analysis](#)
- [PrimeSim SPICE RF Harmonic Balance Analysis](#)
- [PrimeSim SPICE RF Harmonic Balance AC Analysis](#)
- [PrimeSim SPICE RF Harmonic Balance XF Analysis](#)
- [PrimeSim SPICE RF Harmonic Balance Noise Analysis](#)
- [PrimeSim SPICE RF Harmonic Balance Linear Analysis](#)
- [PrimeSim SPICE RF Harmonic Balance Stability Analysis](#)
- [PrimeSim SPICE RF Envelope Following Analysis](#)
- [PrimeSim SPICE Circuit Checks](#)

PrimeSim SPICE Transient Analysis

A transient analysis calculates the circuit solution as a function of time and over a specified time range.

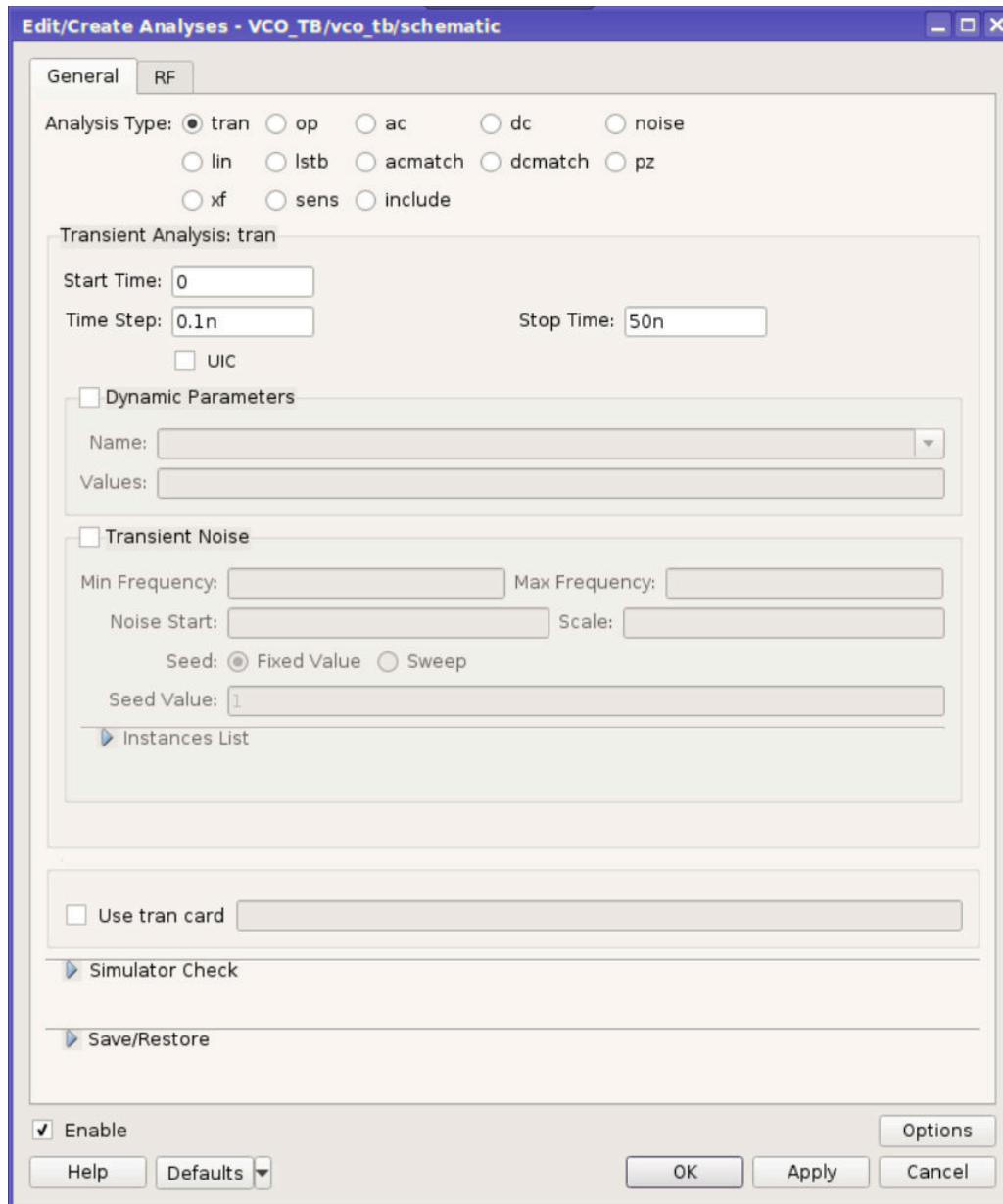
To create a transient analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **tran** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.

4. Click **OK** or **Apply**.

Your transient analysis is now set up.

PrimeSim SPICE Operating Point Analysis

Note:

If you want to be able to annotate or print device operating points for any time point other than time=0, set up a transient analysis as well. See [PrimeSim SPICE Transient Analysis](#).

To create an operating point analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

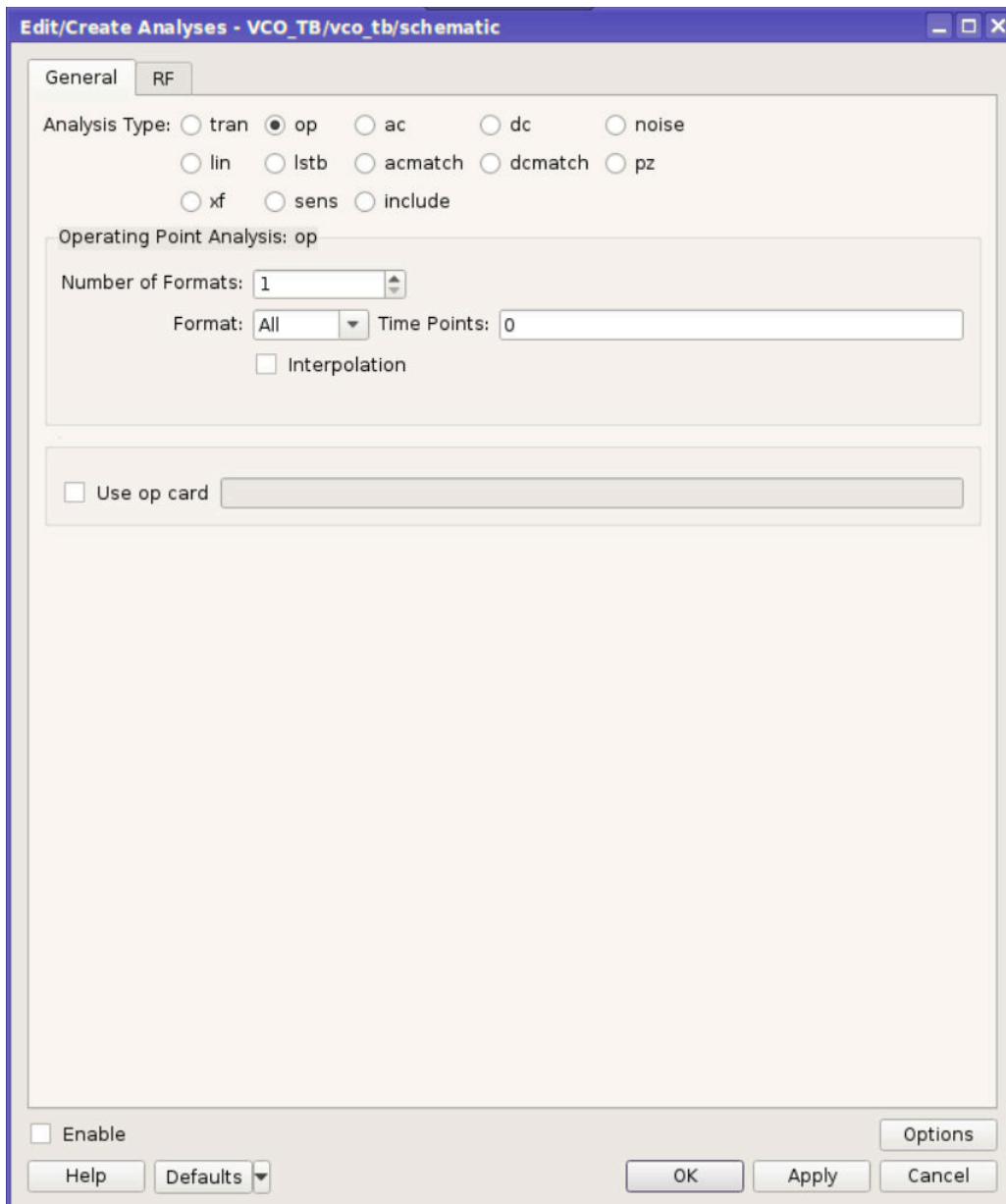
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **op** radio button.

Chapter 6: Setting Up PrimeSim SPICE Analyses
PrimeSim SPICE Operating Point Analysis



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your operating point analysis is now set up.

PrimeSim SPICE AC Analysis

An AC analysis calculates the AC output variables as a function of frequency.

To create an AC analysis:

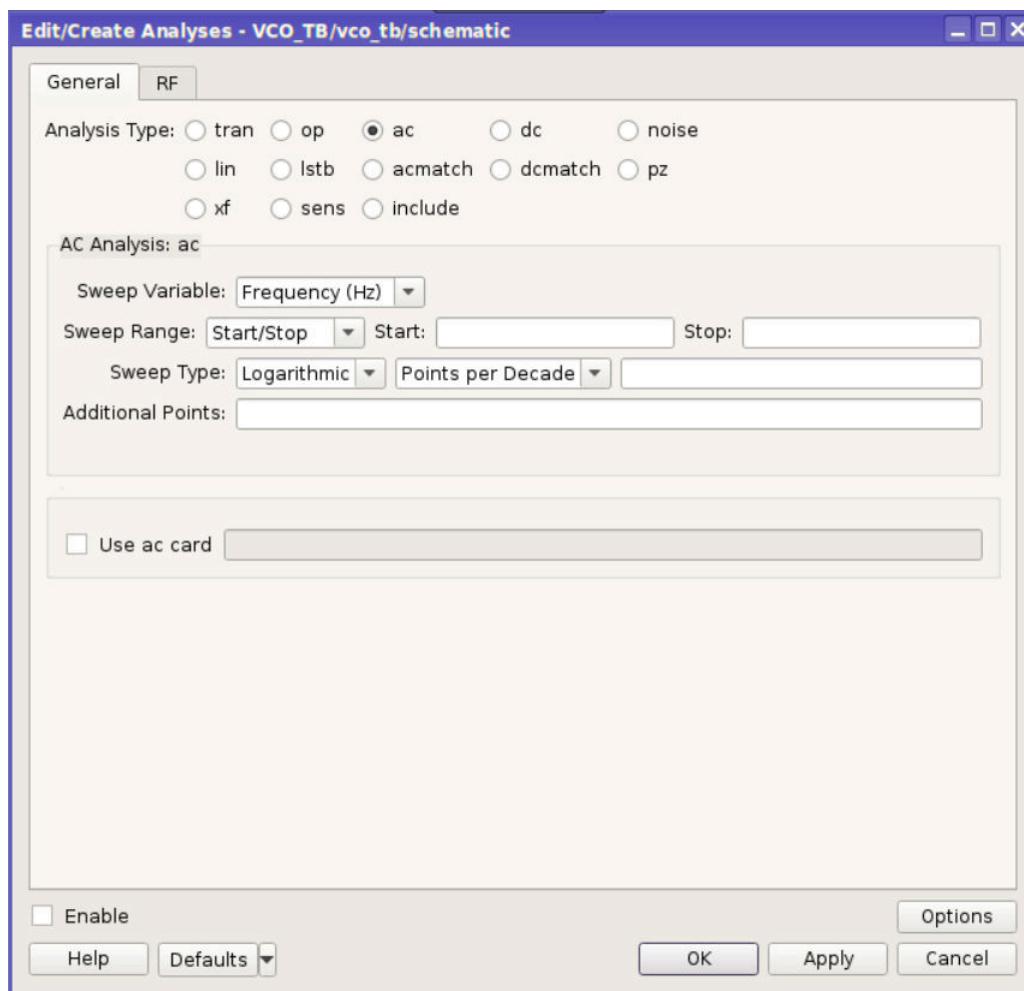
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **ac** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your AC analysis is now set up.

PrimeSim SPICE DC Analysis

To create a DC analysis:

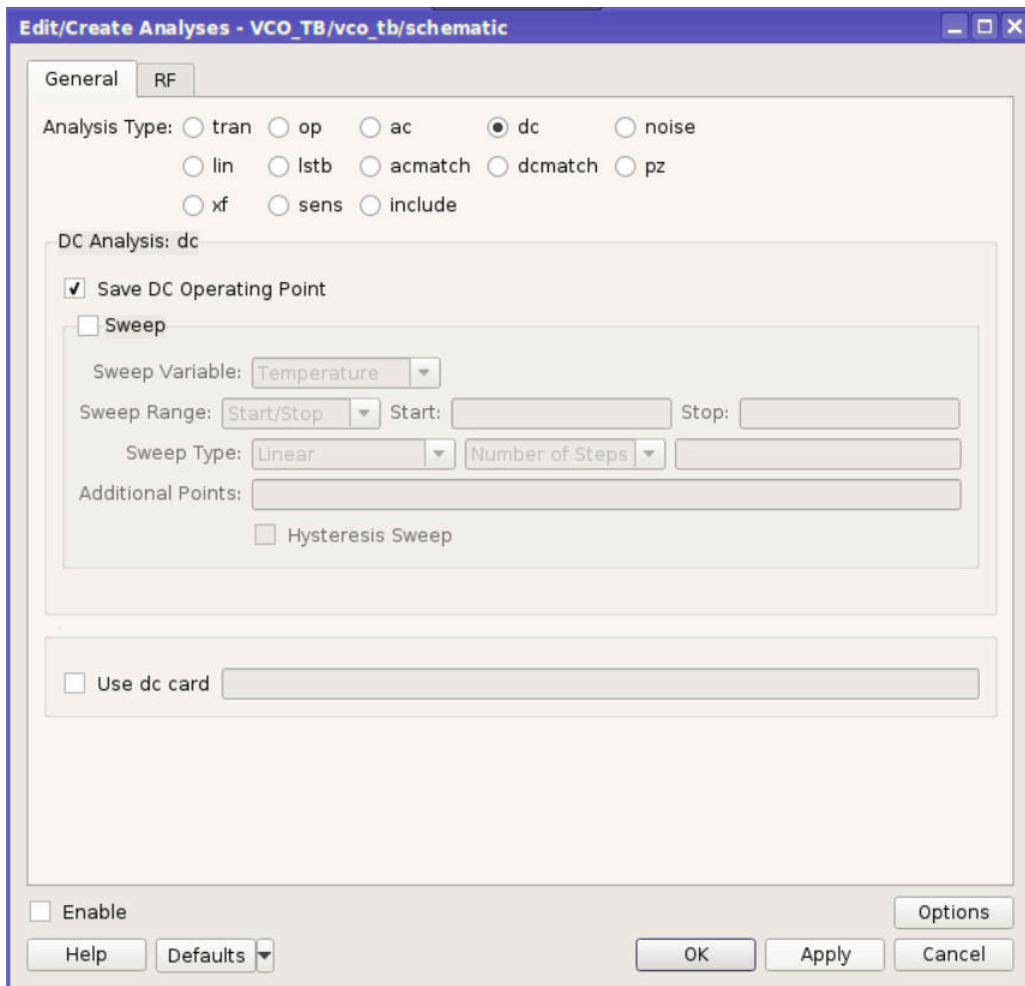
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **dc** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.

4. Click **OK** or **Apply**.

Your DC analysis is now set up.

PrimeSim SPICE Noise Analysis

Note:

Before running a noise analysis in PrimeSim, you must first set up and enable an AC analysis. You can set up a noise analysis at any time, but you cannot enable the noise analysis unless an AC analysis is enabled. See [PrimeSim SPICE AC Analysis](#).

To create a noise analysis:

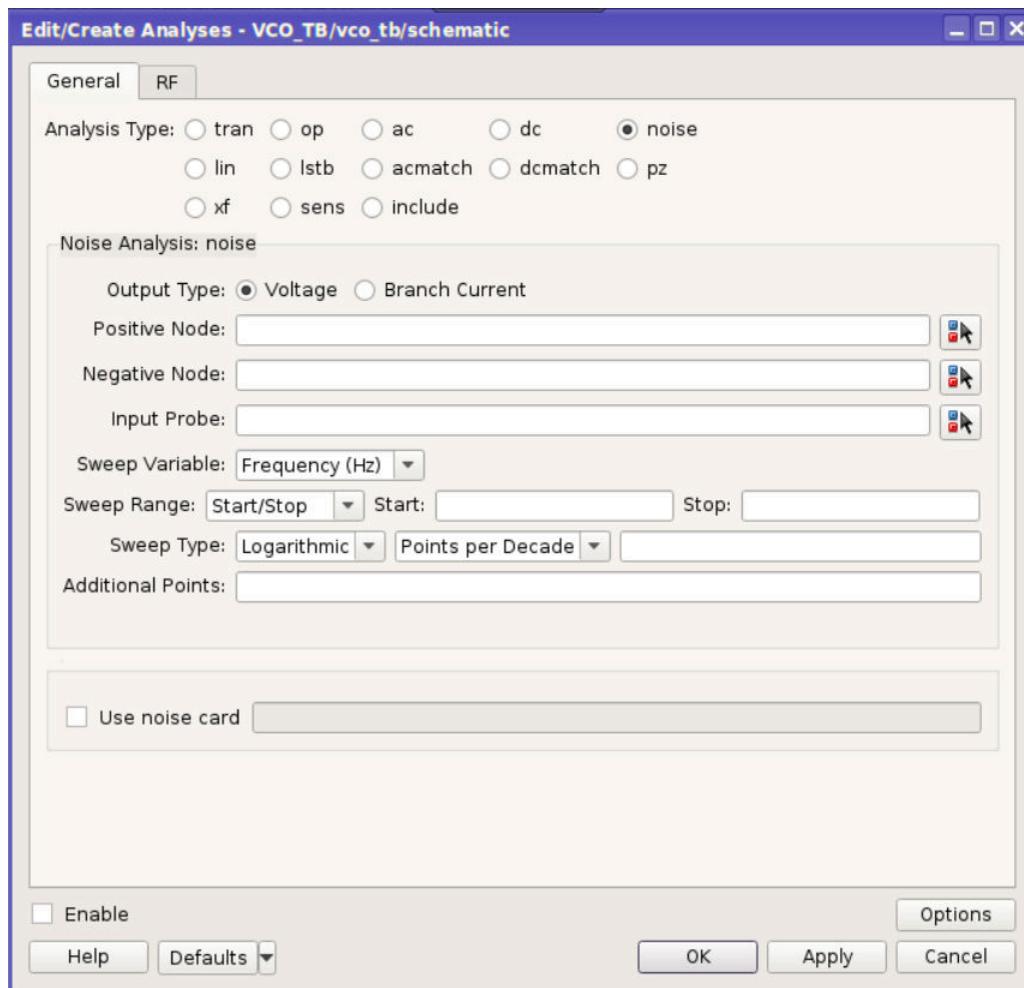
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **noise** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your noise analysis is now set up.

PrimeSim SPICE Linear Network Parameter Analysis

A .LIN analysis calculates S (scattering) and noise parameters, as well as additional measurements.

Note:

Before running a .LIN analysis, you must first set up and enable an AC analysis.
See [PrimeSim SPICE AC Analysis](#).

To create a linear network parameter (lin) analysis:

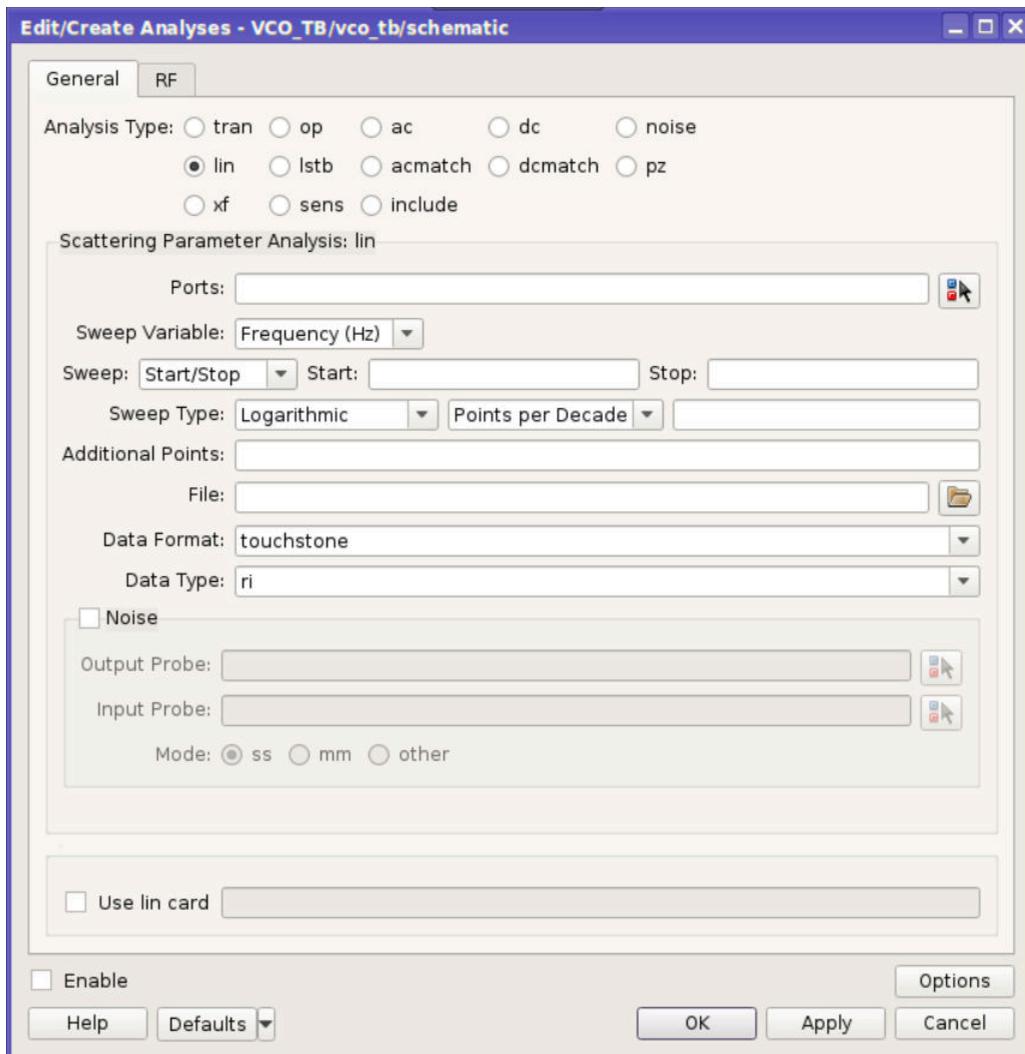
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **lin** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your linear network parameter (lin) analysis is now set up.

PrimeSim SPICE Loop Stability Analysis (LSTB)

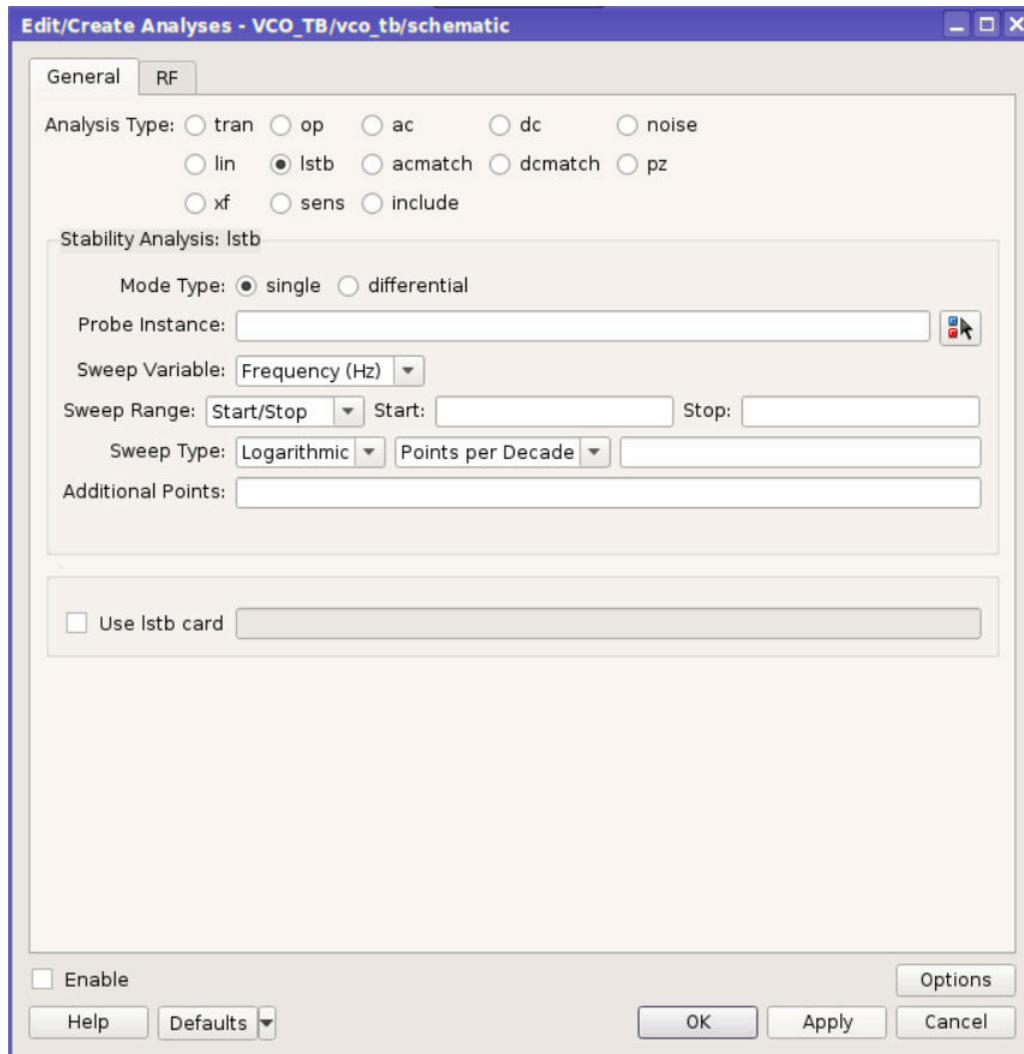
To create an LSTB analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
 The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **Istb** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your LSTB analysis is now set up.

PrimeSim SPICE AC Match Analysis

An AC Match analysis determines the combined effects of device variations on the frequency response of a circuit.

Note:

To create an AC match analysis, you must first create an AC analysis, as well as include a model library in your testbench with a Variation block section specified. See [PrimeSim SPICE AC Analysis](#) for more information on creating AC analyses and [Specifying Model Files](#) for more information on specifying model files.

To create an AC match analysis:

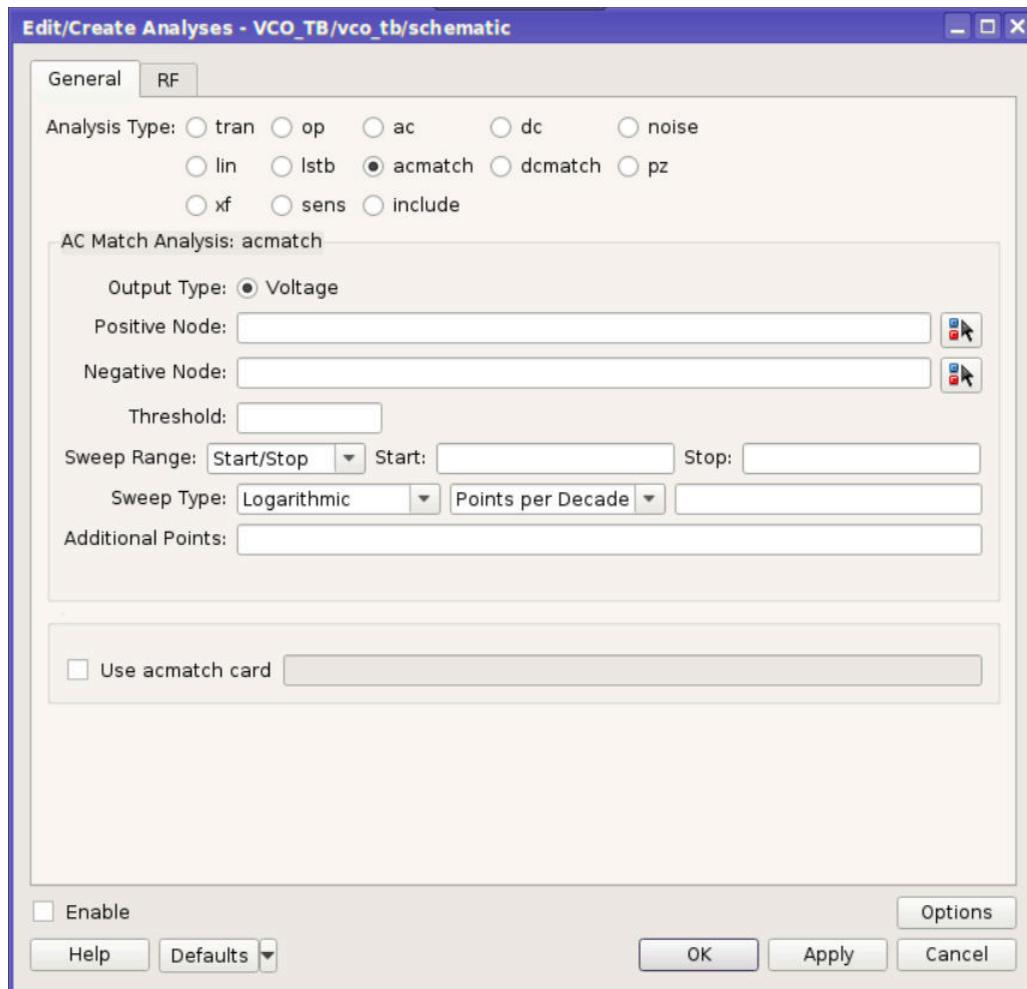
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **acmatch** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your AC match analysis is now set up.

PrimeSim SPICE DC Match Analysis

A DC Match analysis examines the combined effects of variations of all devices on a specified node voltage or branch current. Groups of matched devices are also identified.

To create a DC match analysis:

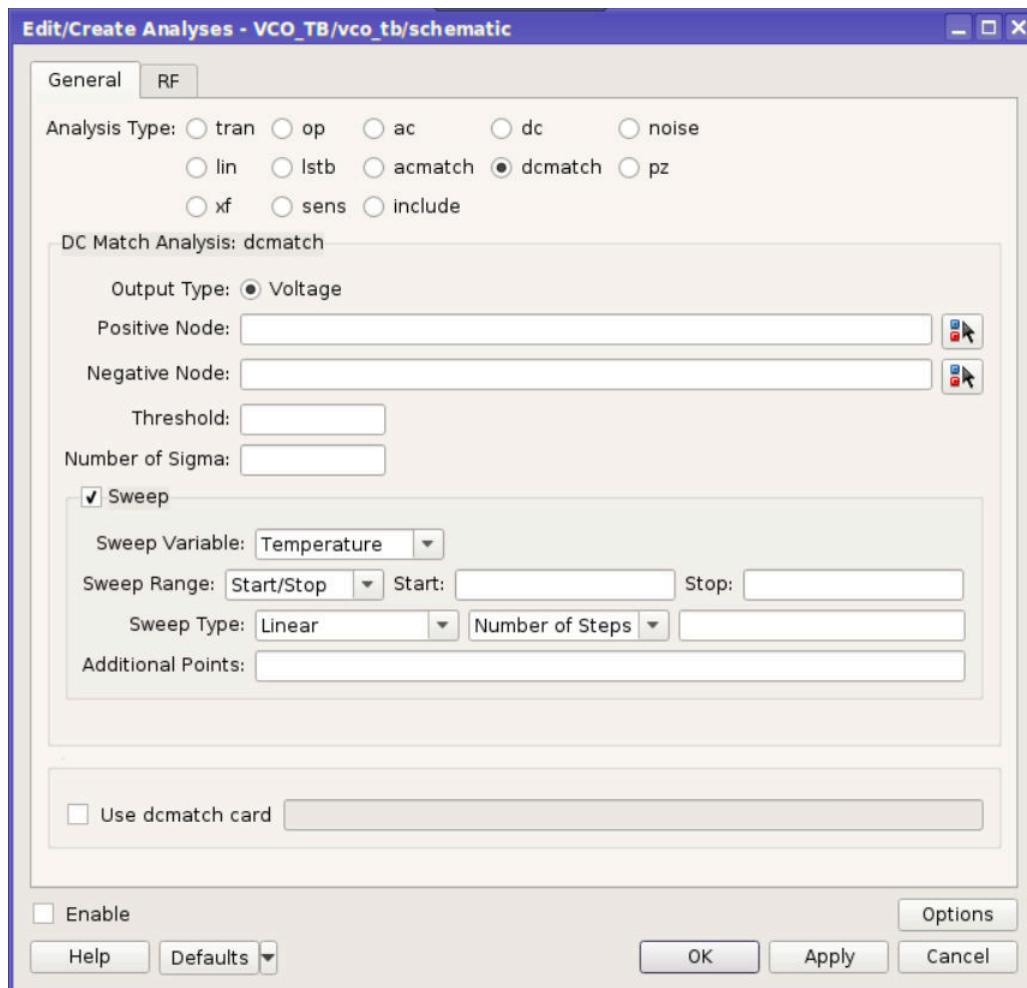
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **dcmatch** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.

4. Click **OK** or **Apply**.

Your DC match analysis is now set up.

PrimeSim SPICE PZ Analysis

To create a PZ analysis:

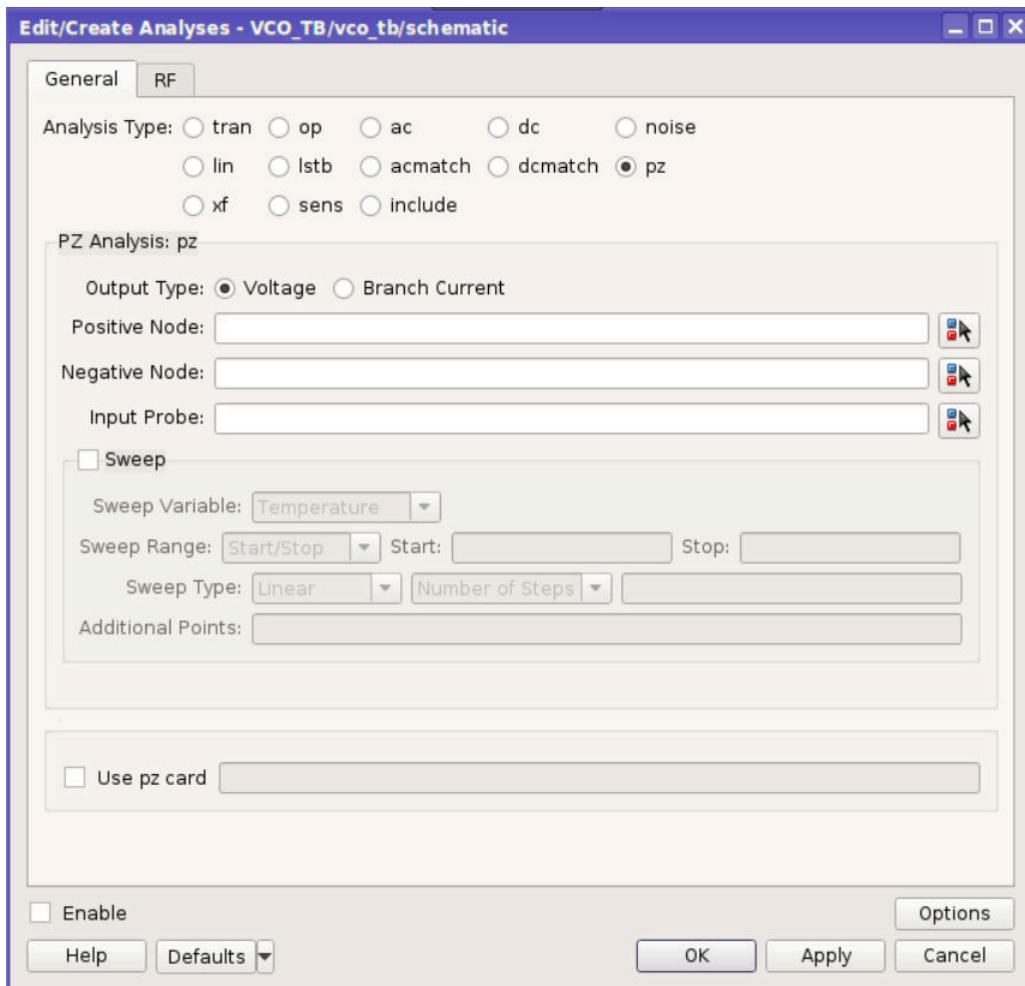
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **pz** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your PZ analysis is now set up.

PrimeSim SPICE XF Analysis

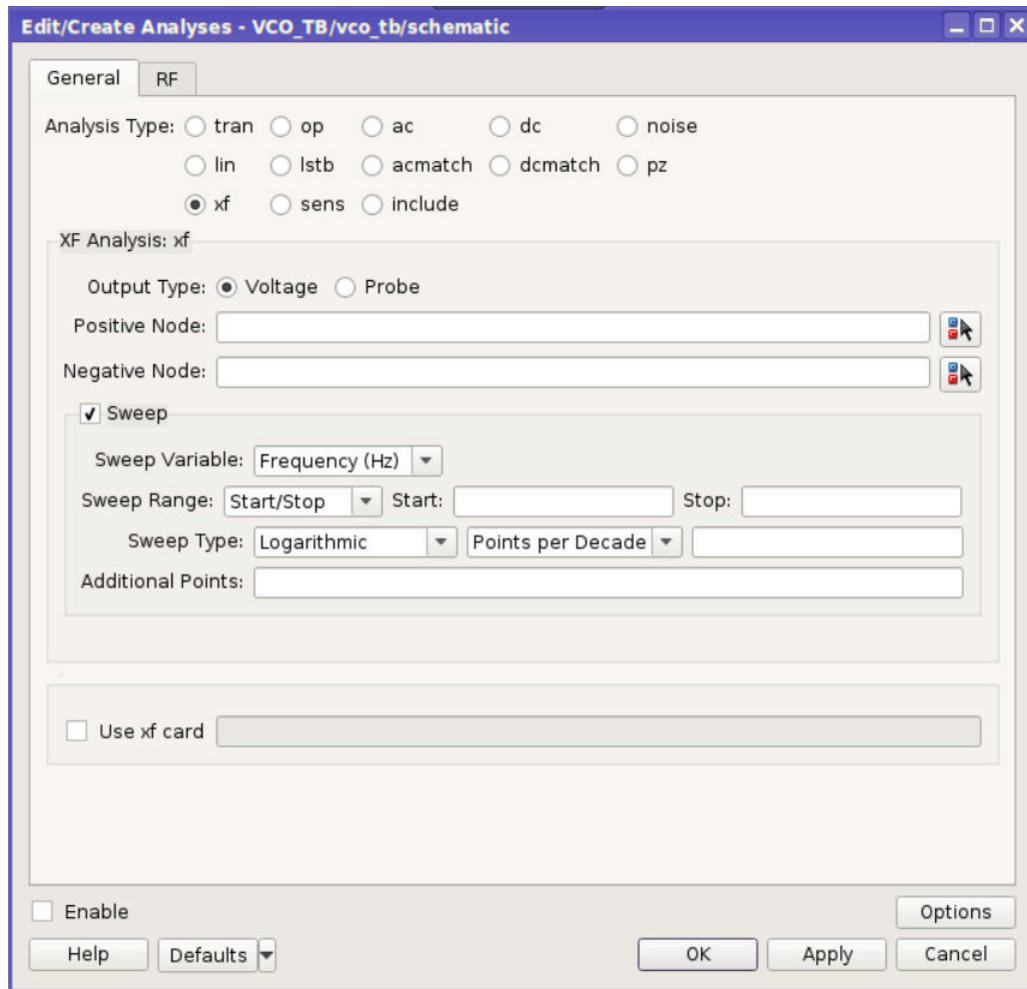
To create an XF match analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
 The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **xf** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your XF analysis is now set up.

PrimeSim SPICE Sensitivity Analysis

To create a sensitivity analysis:

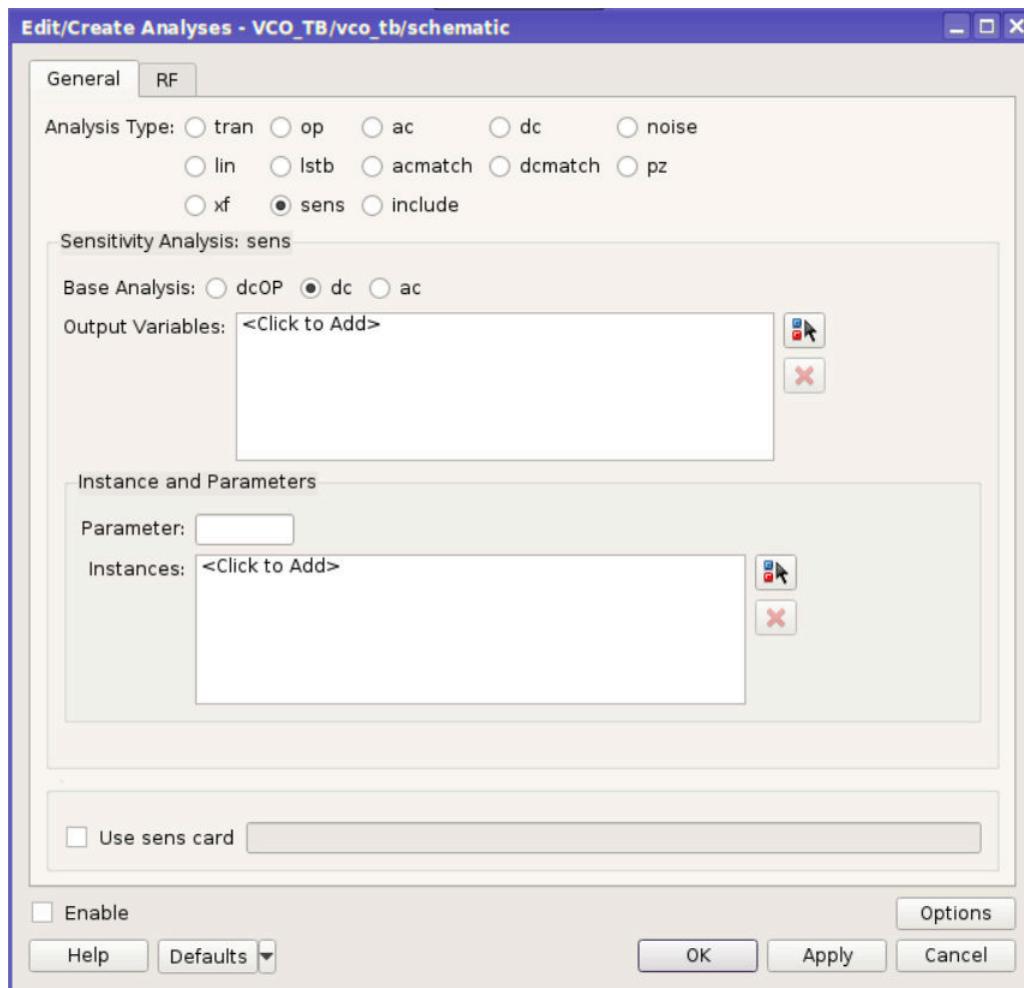
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **sens** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.

4. Click **OK** or **Apply**.

Your sensitivity analysis is now set up.

PrimeSim SPICE Include Analysis

To create an include analysis:

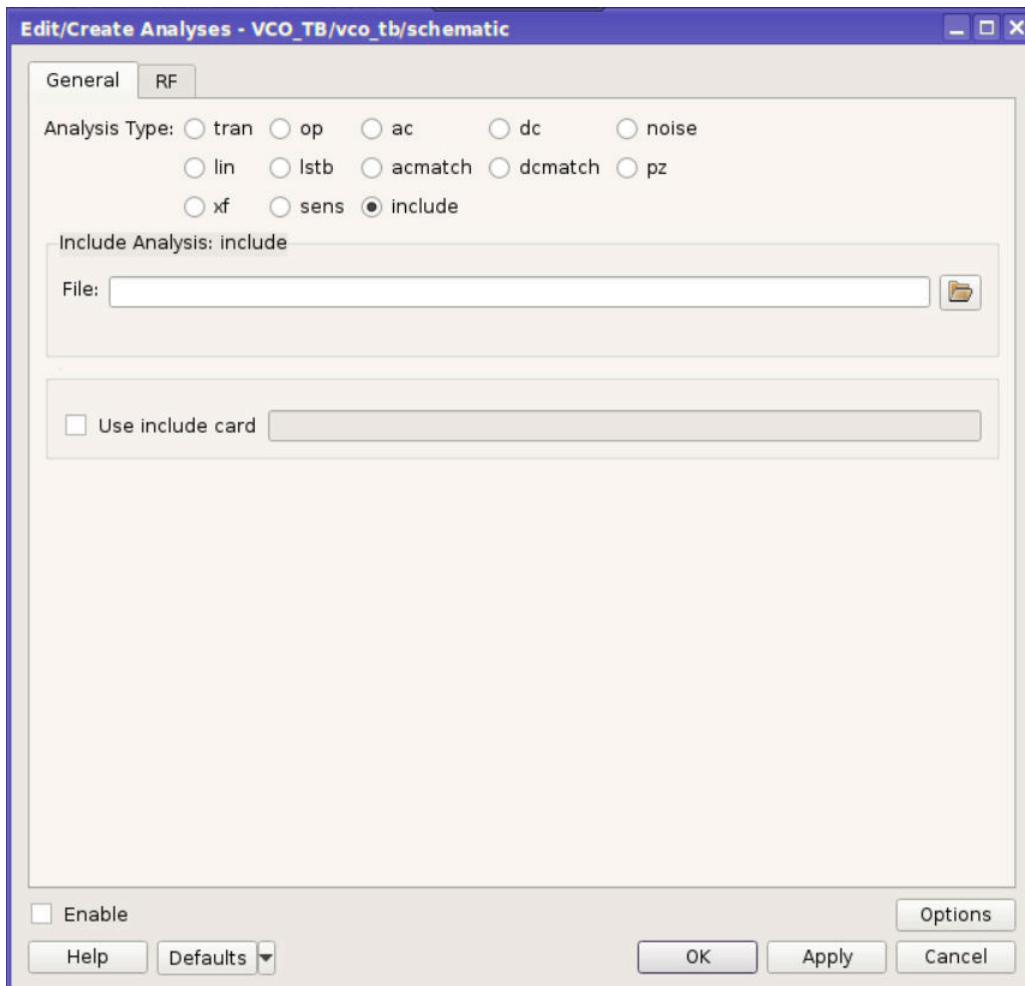
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and click the **include** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your include analysis is now set up.

PrimeSim SPICE RF Shooting Newton Analysis

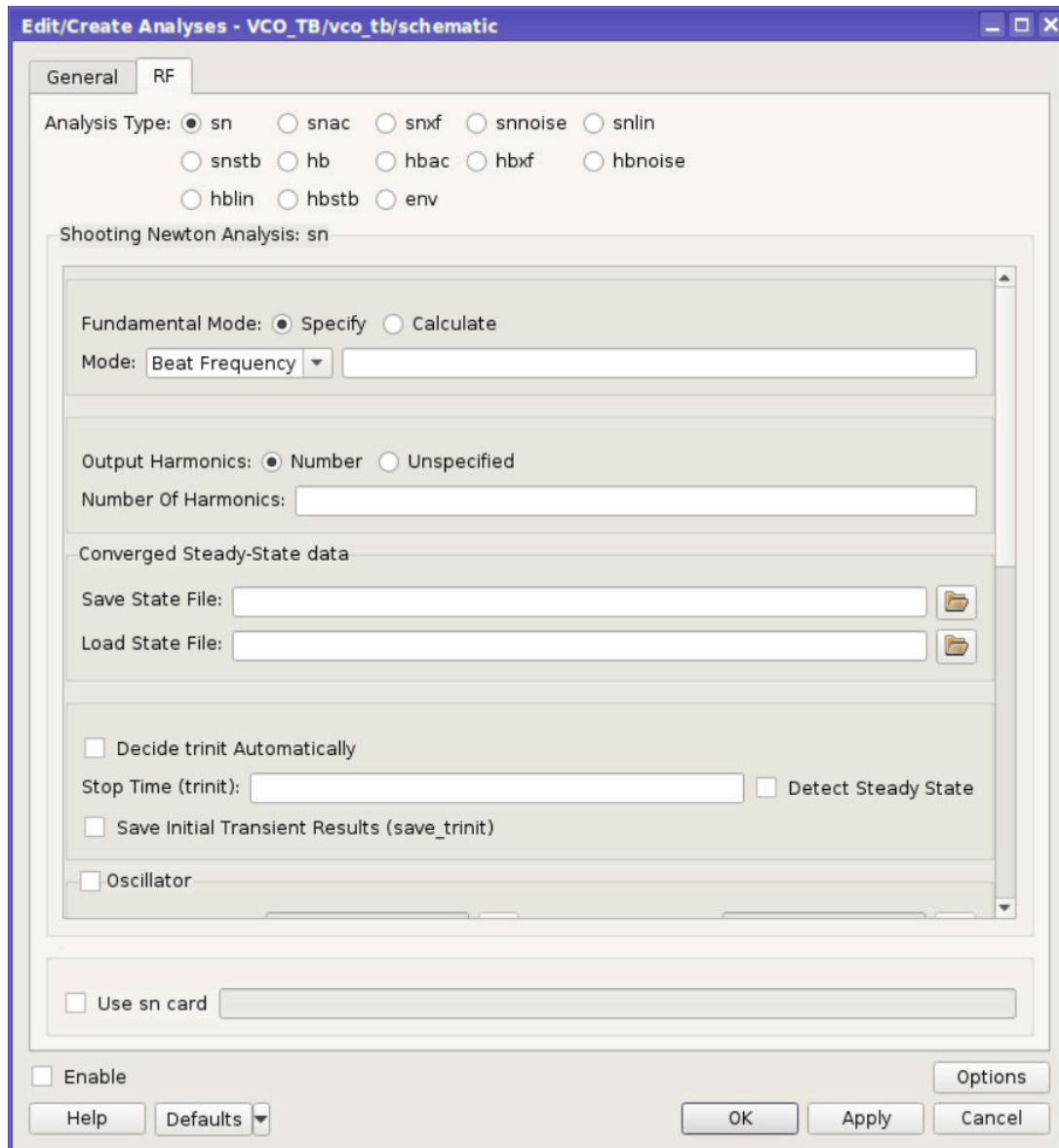
To create a shooting Newton analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **sn** radio button.



3. Choose the domain in which you want to create the shooting Newton analysis: **Frequency or Time**.

4. If you choose **Frequency**, enter values for the **Fundamental Freq** and **No. of Harmonics** and skip to step [Step 6](#). Otherwise, move on to the next step.
5. If you choose the **Time** domain, enter values for the **Period** and **Time Resolution**.
6. Enter values for the **Initial Transient** and **Maximum Initial Cycles**.
7. (Optional) If you want to include a sweep of your shooting Newton analysis, click the arrow next to the **Sweep** text to expand the **Sweep** parameters section and continue to the next step. Otherwise, skip to [Step 13](#).
8. Choose a **Variable Name** to include in the sweep.

Only previously specified variables are listed in the **Variable Name** menu. See [Adding and Editing Design Variables](#) for information on creating variables.

9. Choose a unit of measure for the **Variable Units**: **dBm** or **W**.

The default is **dBm** units.

10. Choose one of the following for **Sweep Type**:

- **Linear**
- **Octave**
- **Decade**
- **Points of Interest**

11. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 12](#). Otherwise, continue to the next step.
12. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
13. Click **Enable** to enable this analysis as part of your testbench.
14. Click **OK** or **Apply**.

Your shooting Newton analysis is now set up.

PrimeSim SPICE RF Shooting Newton AC Analysis

Note:

To create a shooting Newton AC analysis, you must first run a shooting Newton analysis. See [PrimeSim SPICE RF Shooting Newton Analysis](#) for more information on creating those analyses.

To create a shooting Newton AC analysis:

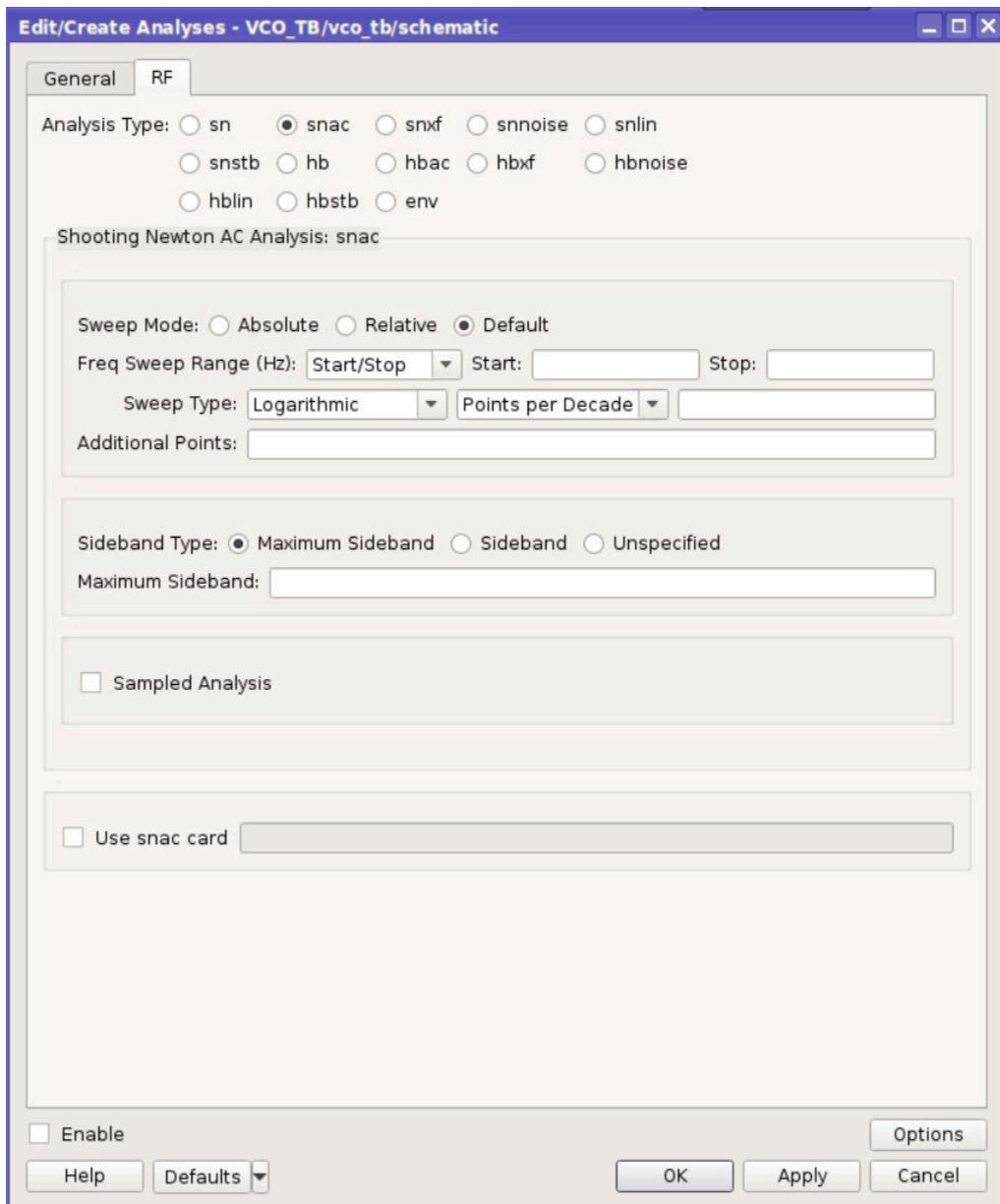
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snac** radio button.



3. Choose one of the following for **Sweep Type**:

- **Linear**
- **Octave**
- **Decade**
- **Points of Interest**

4. If you choose **Points of Interest**, enter one or more values separated by commas for **Points**, and skip to [Step 6](#). Otherwise, continue to the next step.
5. Enter values for the **Start** and **Stop** points, as well as the **No. of Points** to include in the sweep.
6. Click **Enable** to enable this analysis as part of your testbench.
7. Click **OK** or **Apply**.

Your shooting Newton AC analysis is now set up.

PrimeSim SPICE RF Shooting Newton Transfer Function Analysis

Note:

To create a shooting Newton transfer function analysis, you must first run a shooting Newton or shooting Newton oscillator analysis. See [PrimeSim HSPICE RF Shooting Newton Analysis](#) for more information on creating those analyses.

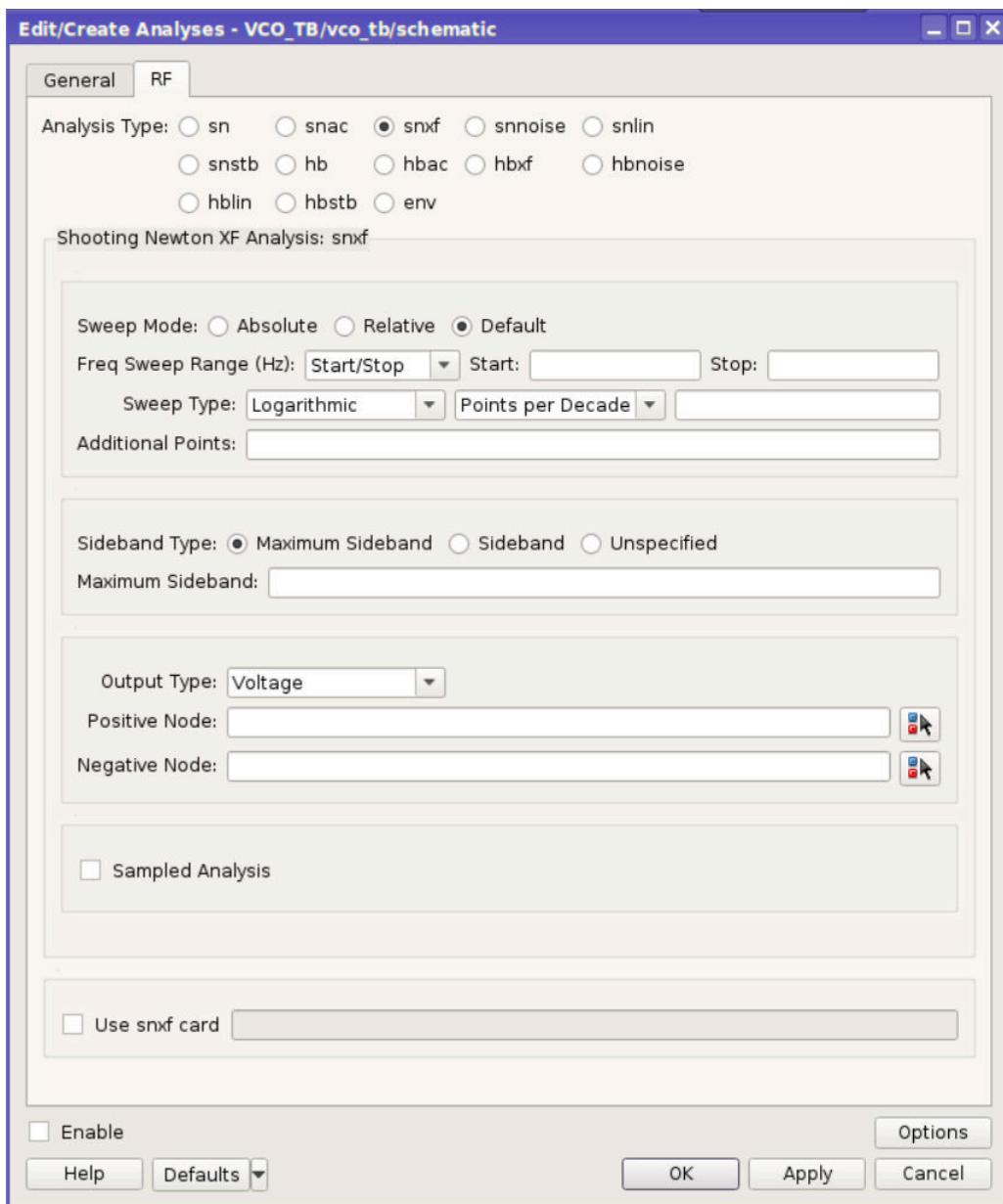
To create a shooting Newton transfer function analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snxf** radio button.



- Click **Enable** to enable this analysis as part of your testbench.

All sources are saved by the simulator when this analysis is enabled.

- Click **OK** or **Apply**.

Your shooting Newton transfer function analysis is now set up.

PrimeSim SPICE RF Shooting Newton Noise Analysis

Note:

A shooting Newton analysis must be run before running this analysis. See the [PrimeSim SPICE RF Shooting Newton Analysis](#) section for more information on creating a shooting Newton analysis.

To create a shooting Newton noise analysis:

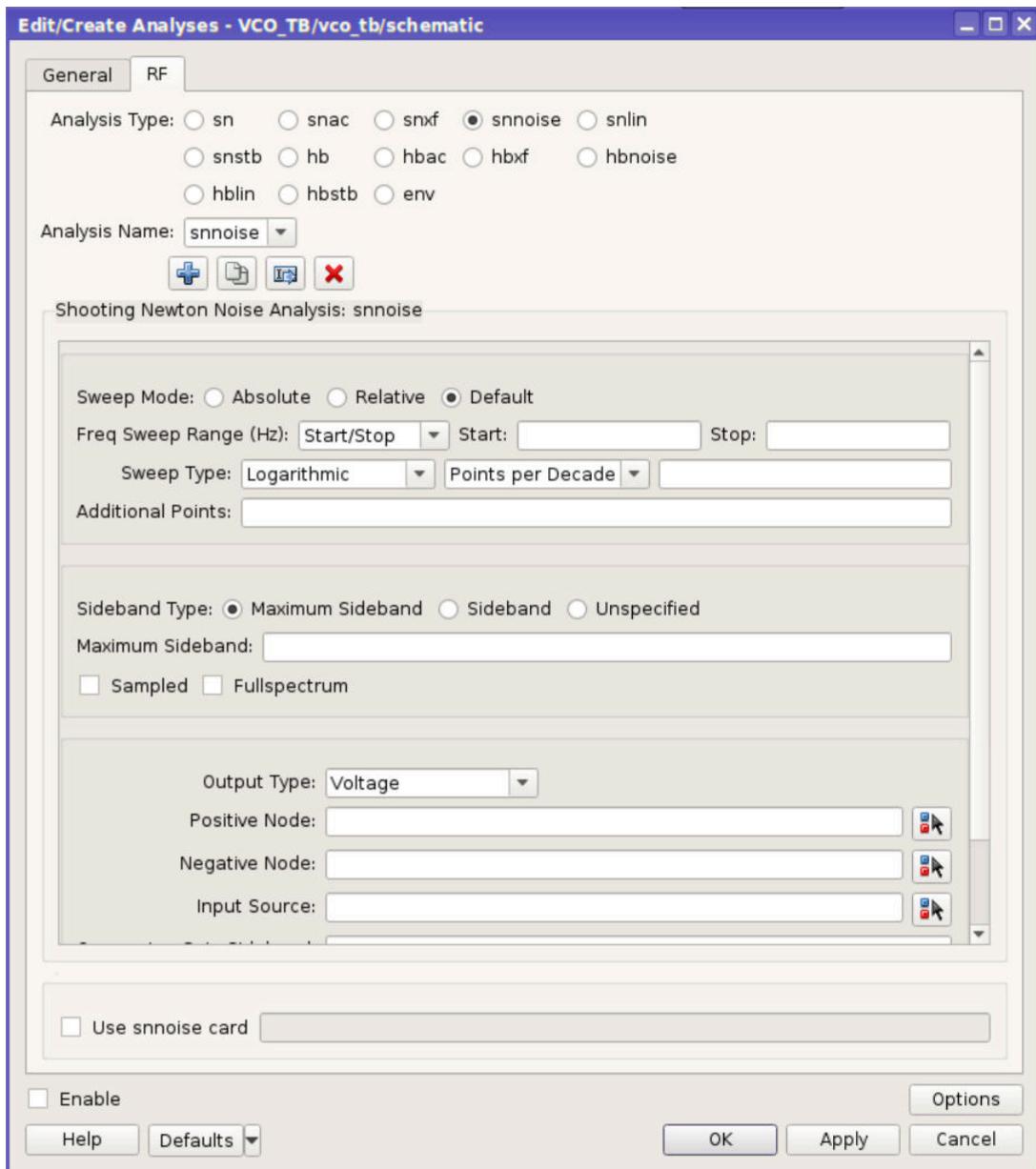
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snoise** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your shooting Newton noise analysis is now set up.

Note:

A validation check is performed after the shooting Newton noise analysis is set up to ensure that the number of tones for the OFB index matches the number of harmonics in the shooting Newton noise analysis.

PrimeSim SPICE RF Shooting Newton Linear Analysis

Note:

A shooting Newton analysis must be run before running this analysis. See the [PrimeSim SPICE RF Shooting Newton Analysis](#) section for more information on creating a shooting Newton analysis.

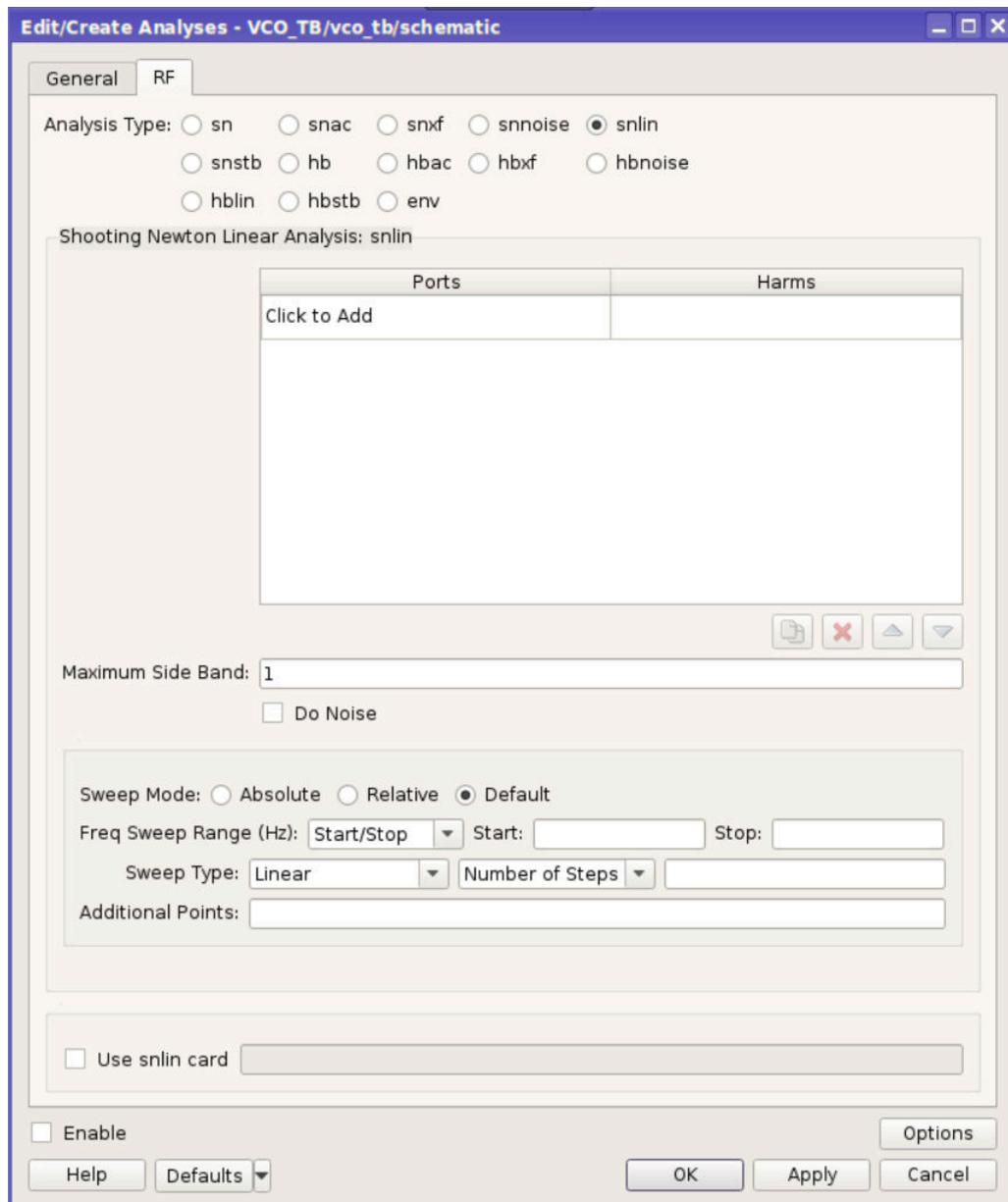
To create a shooting Newton linear analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **snlin** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your shooting Newton linear analysis is now set up.

PrimeSim SPICE RF Shooting Newton Stability Analysis

Note:

A shooting Newton analysis must be run before running this analysis. See the [PrimeSim SPICE RF Shooting Newton Analysis](#) section for more information on creating a shooting Newton analysis.

To create a shooting Newton stability analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

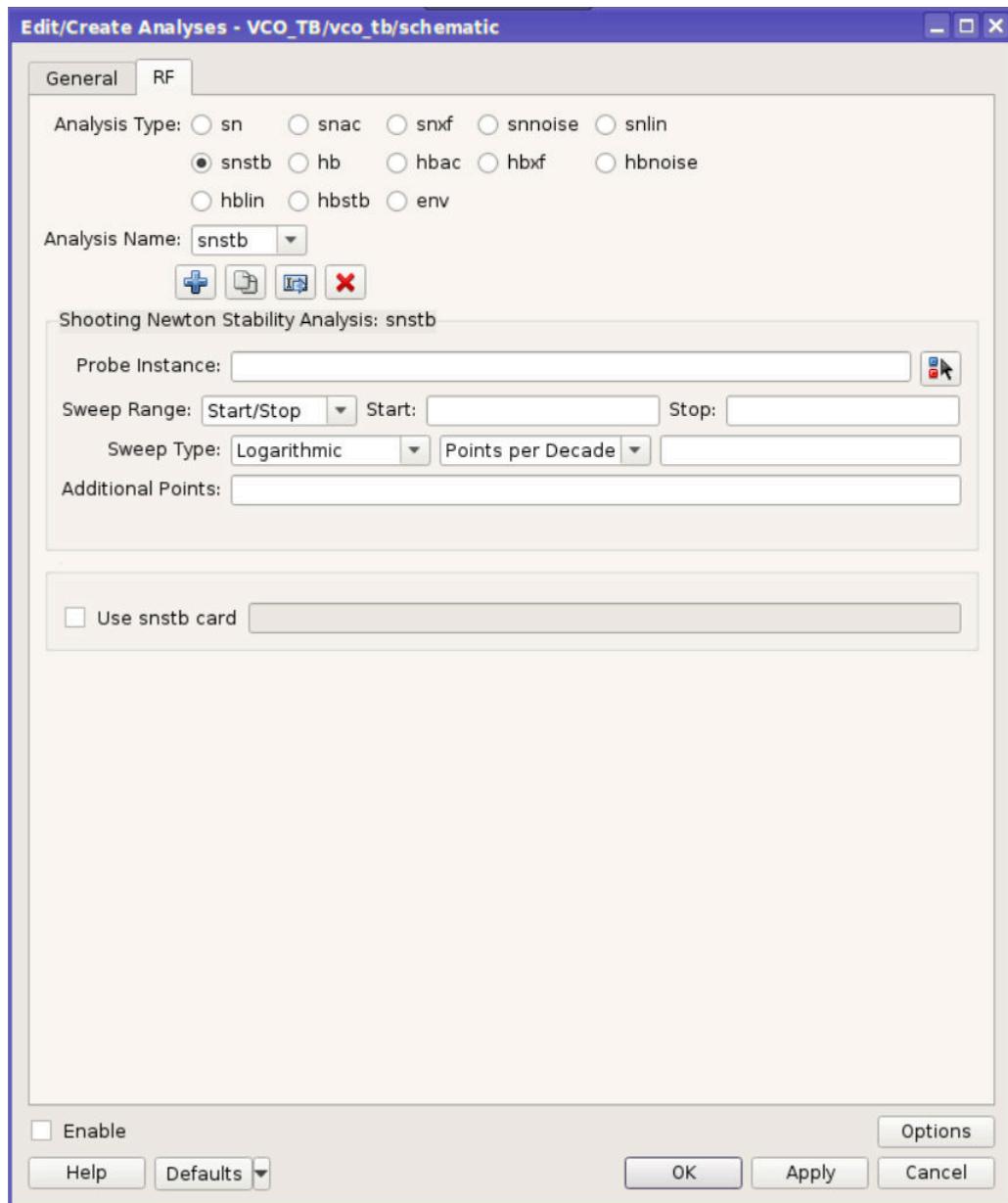
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **sNSTB** radio button.

Chapter 6: Setting Up PrimeSim SPICE Analyses
PrimeSim SPICE RF Shooting Newton Stability Analysis



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your shooting Newton stability analysis is now set up.

PrimeSim SPICE RF Harmonic Balance Analysis

To create a harmonic balance analysis:

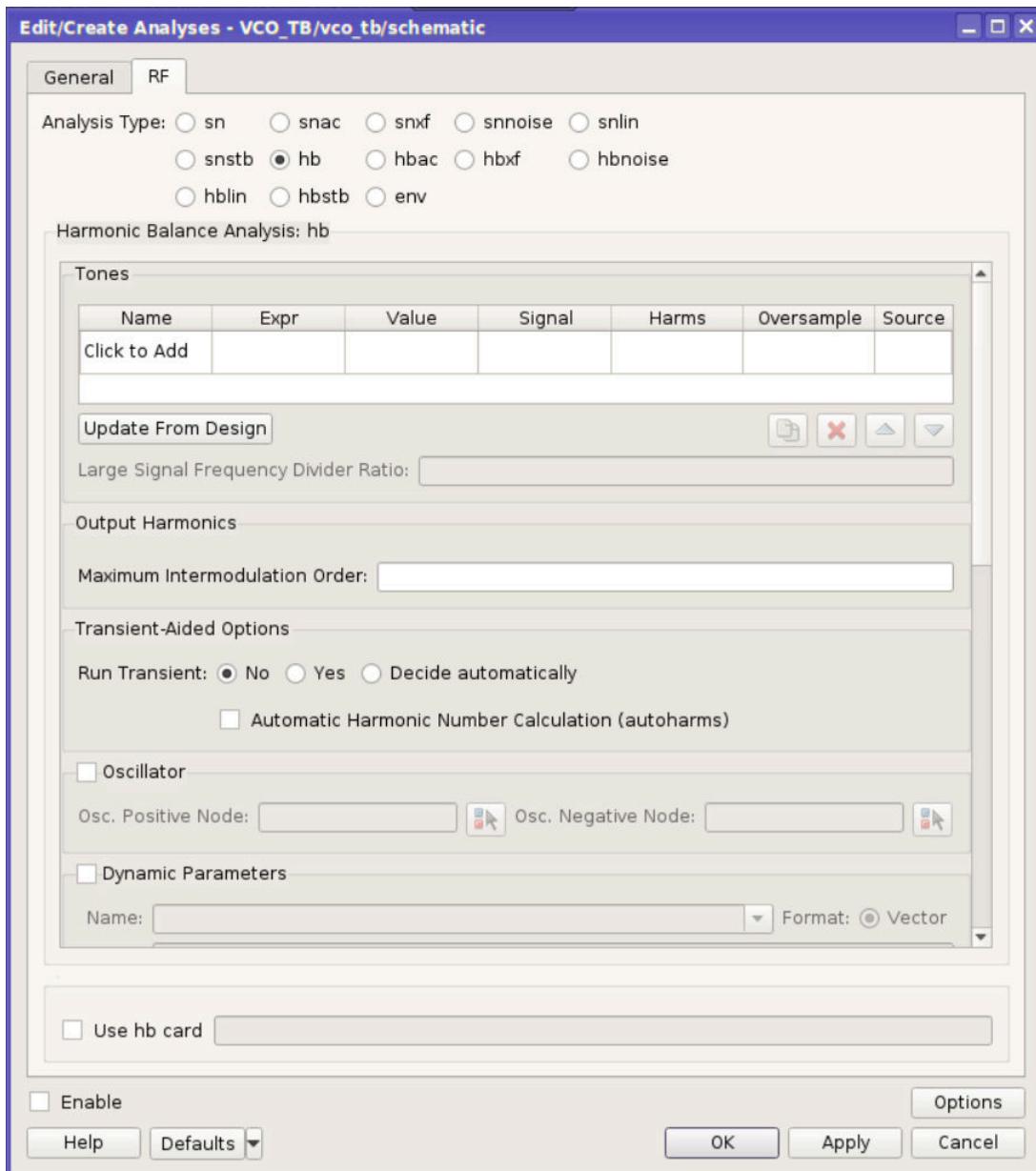
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hb** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your harmonic balance analysis is now set up.

PrimeSim SPICE RF Harmonic Balance AC Analysis

To create a harmonic balance AC analysis:

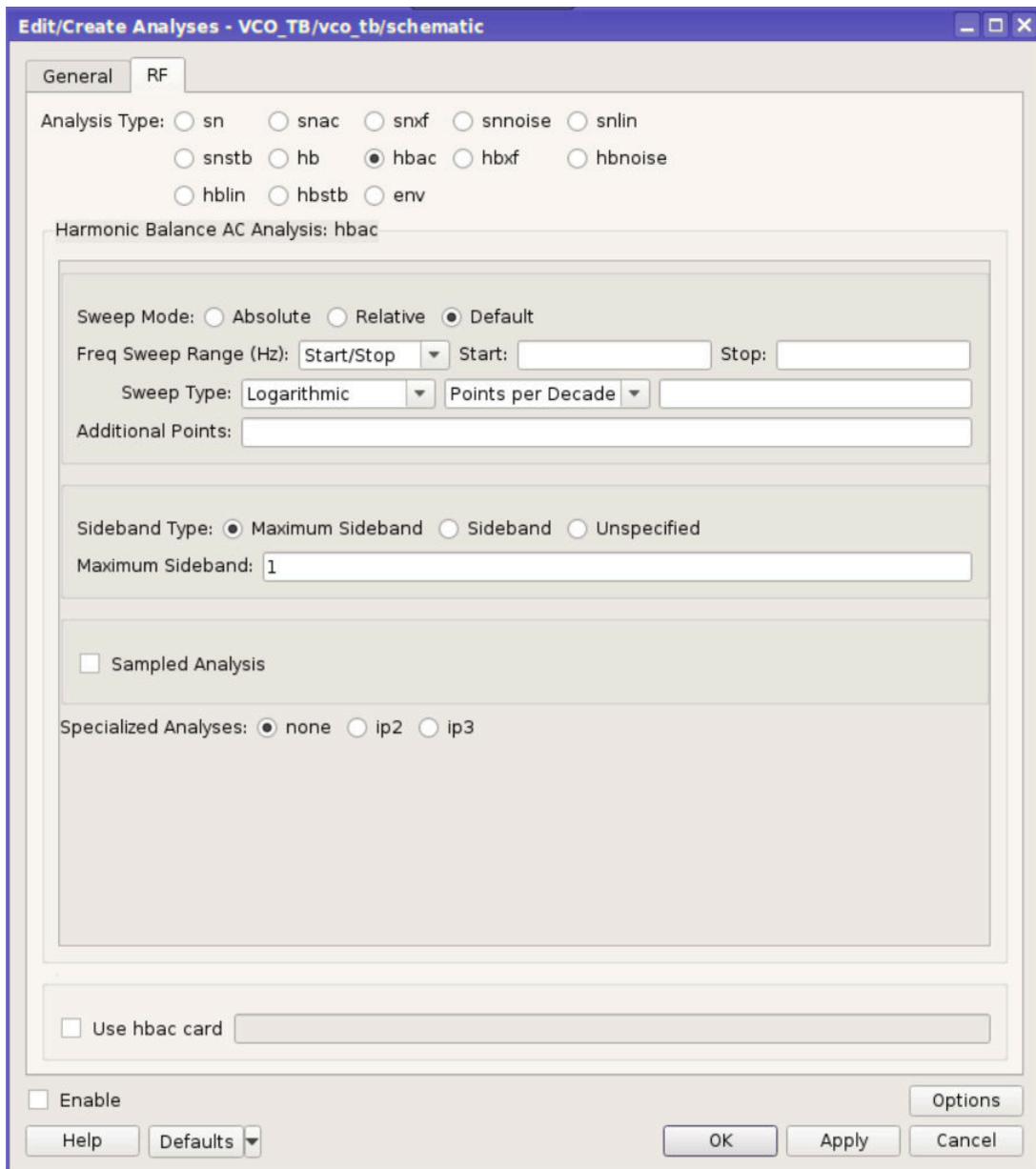
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

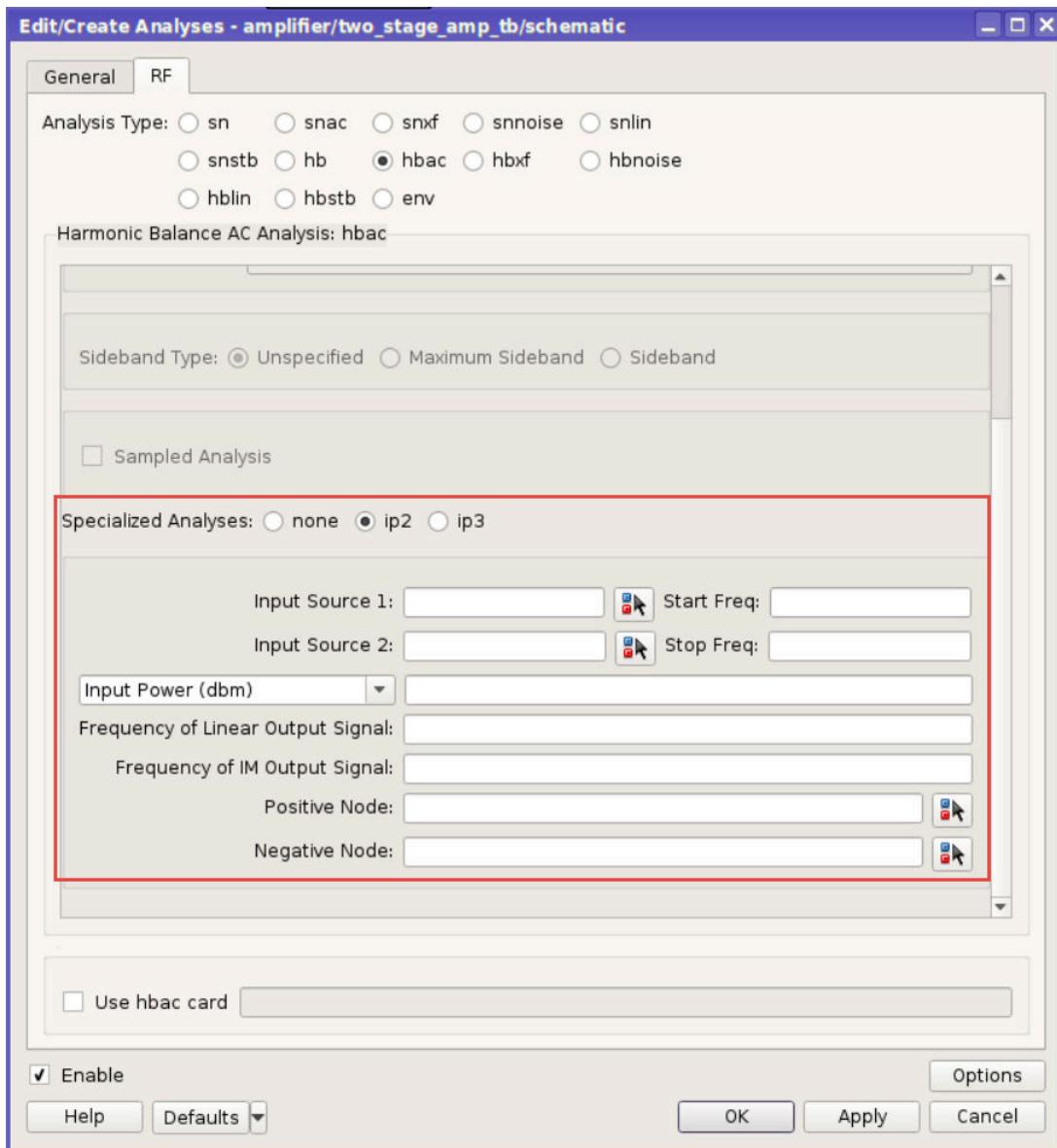
You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hbac** radio button.

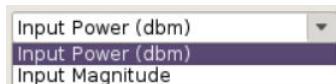


3. (Optional) Select one of the **Specialized Analyses** to view other options.

Chapter 6: Setting Up PrimeSim SPICE Analyses
 PrimeSim SPICE RF Harmonic Balance AC Analysis



4. (Optional) When using specialized analyses **ip2** and **ip3**, you have the option of setting **Input Power (dbm)** or **Input Magnitude**.



5. Click **Enable** to enable this analysis as part of your testbench.

6. Click **OK** or **Apply**.

Your harmonic balance AC analysis is now set up.

PrimeSim SPICE RF Harmonic Balance XF Analysis

To create a harmonic balance XF analysis:

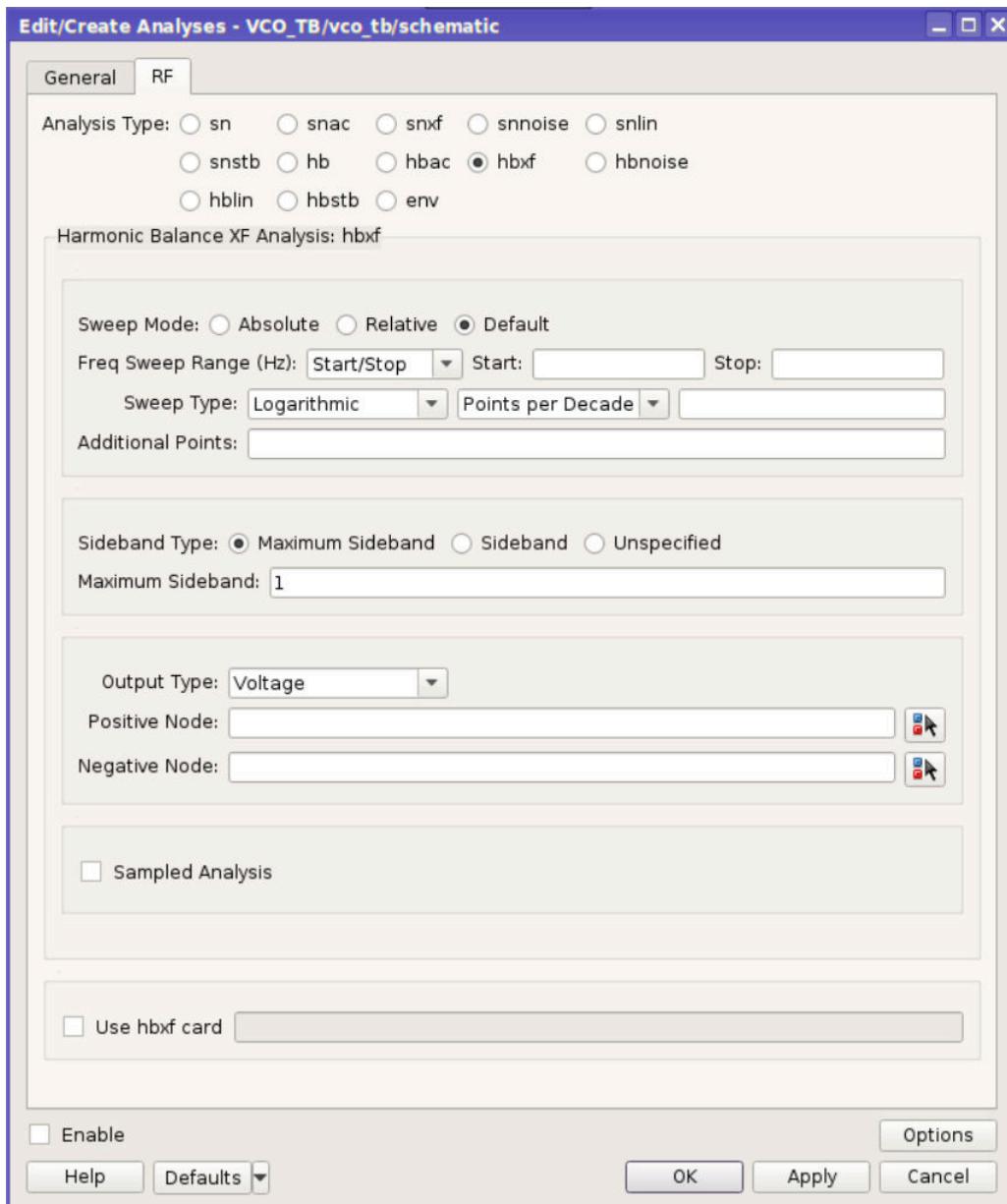
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hbxf** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your harmonic balance XF analysis is now set up.

PrimeSim SPICE RF Harmonic Balance Noise Analysis

To create a harmonic balance noise analysis:

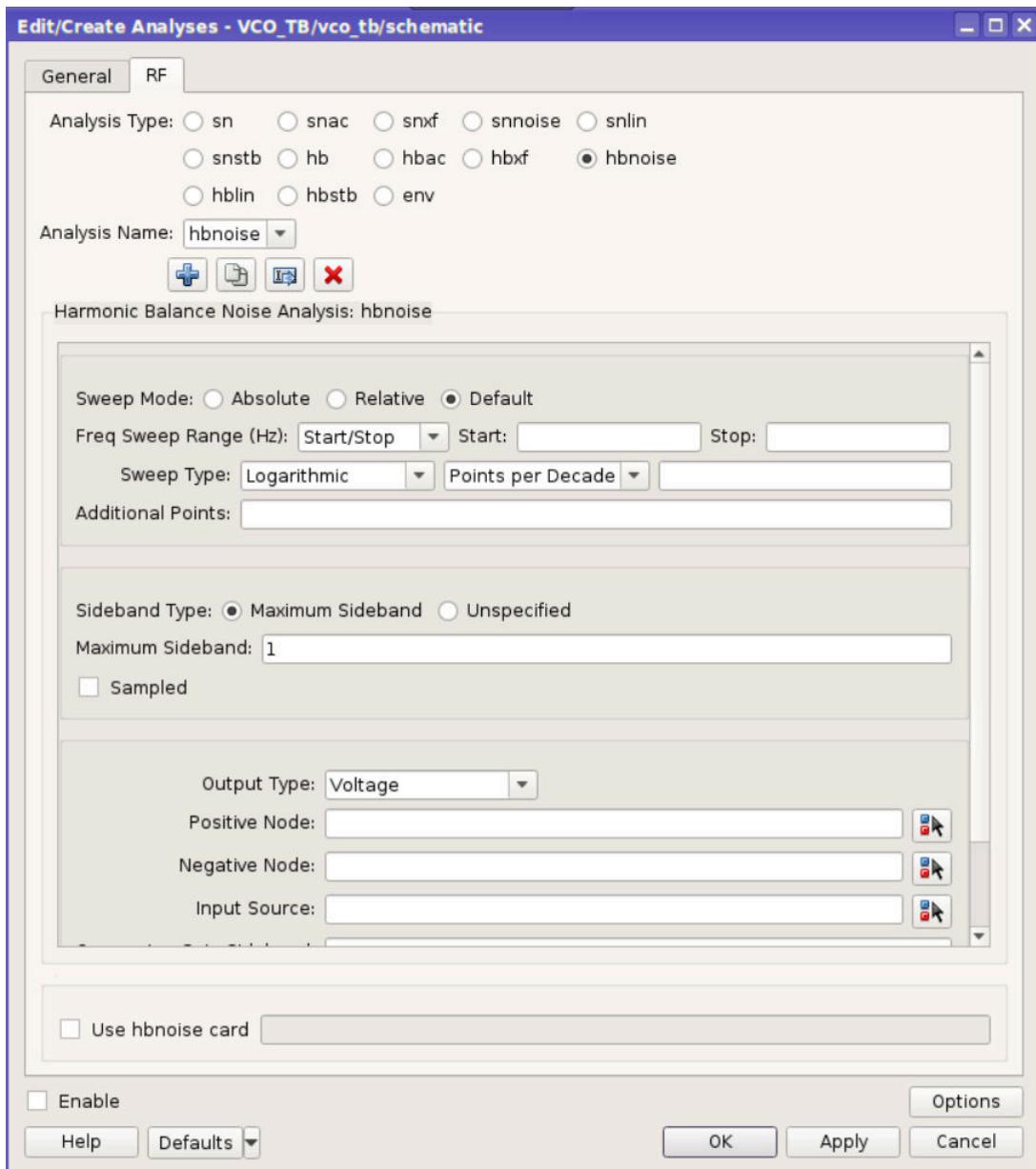
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hbnoise** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your harmonic balance noise analysis is now set up.

PrimeSim SPICE RF Harmonic Balance Linear Analysis

To create a harmonic balance linear analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

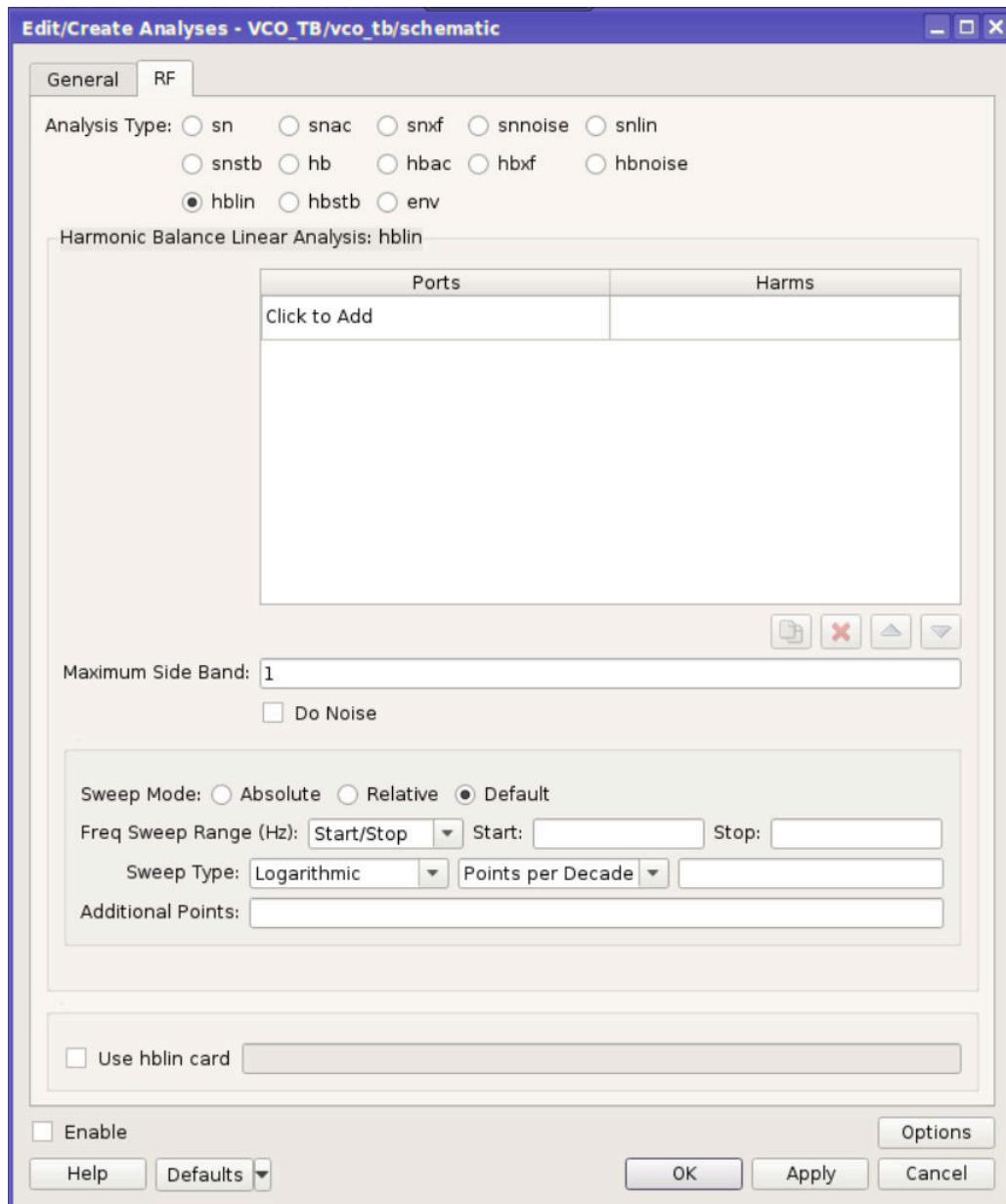
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hblin** radio button.

Chapter 6: Setting Up PrimeSim SPICE Analyses
PrimeSim SPICE RF Harmonic Balance Linear Analysis



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your harmonic balance linear analysis is now set up.

PrimeSim SPICE RF Harmonic Balance Stability Analysis

To create a harmonic balance stability analysis:

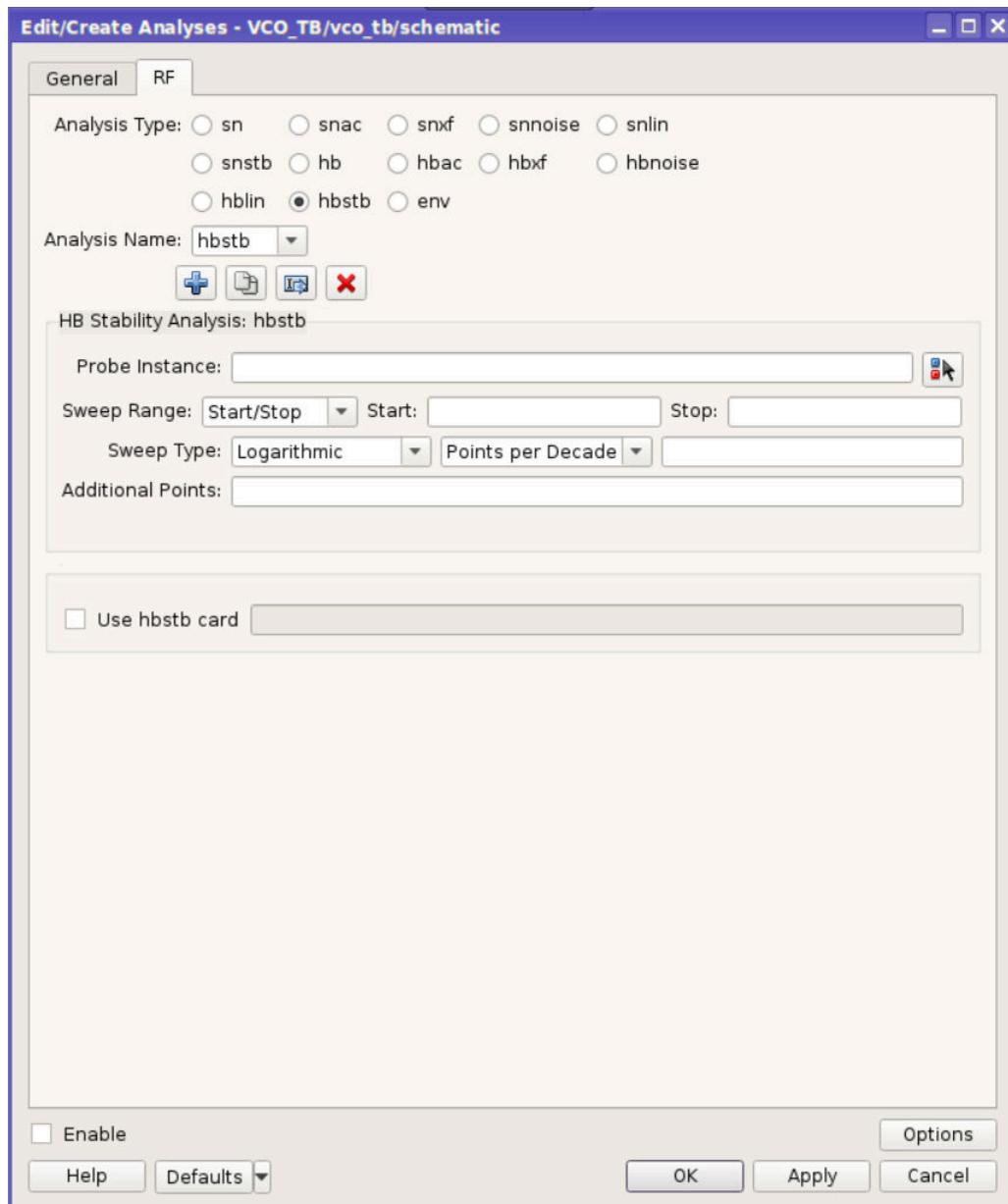
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **hbstb** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your harmonic balance stability analysis is now set up.

PrimeSim SPICE RF Envelope Following Analysis

To create an envelope following analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

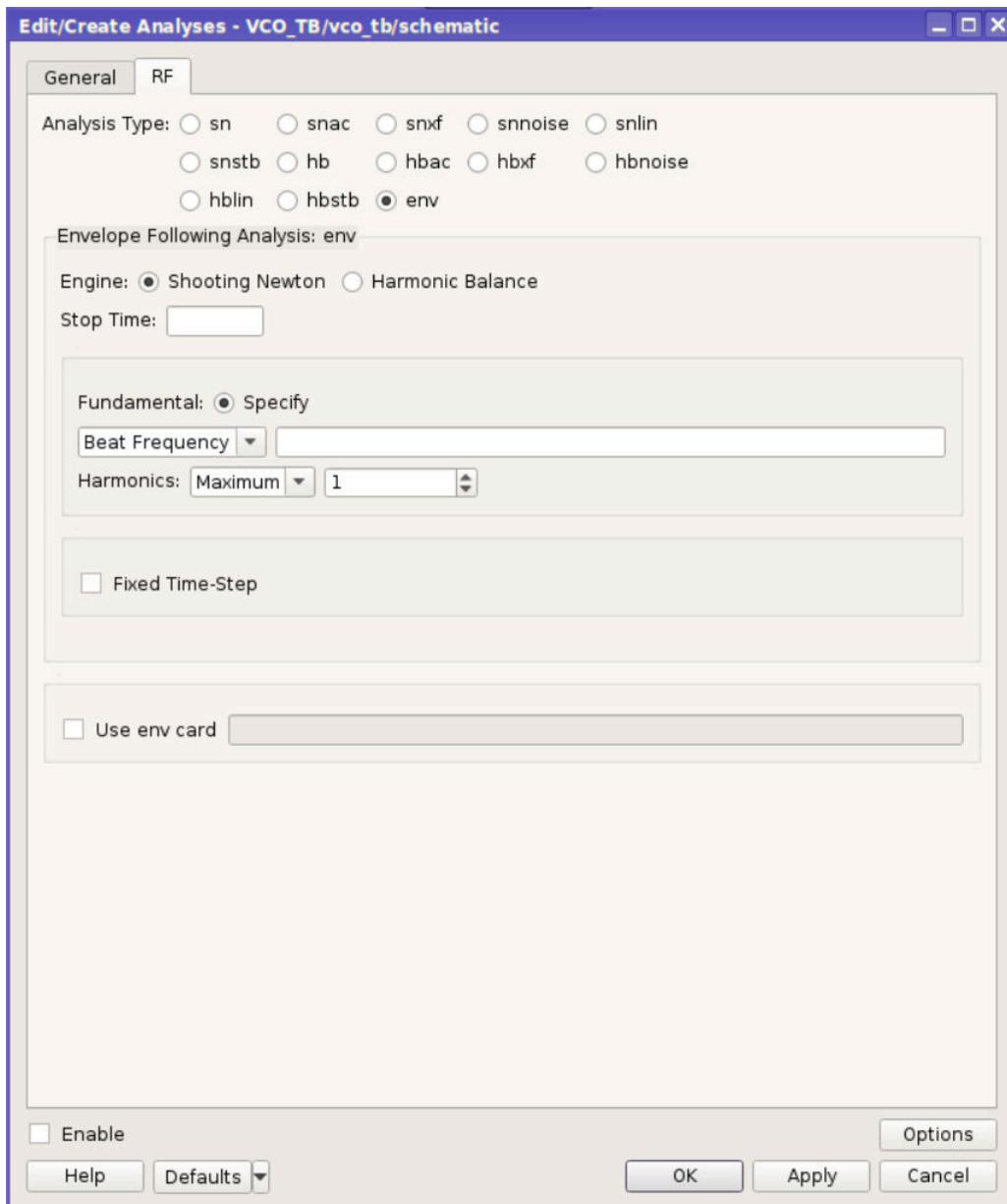
The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **RF** tab are visible, and click the **env** radio button.

Chapter 6: Setting Up PrimeSim SPICE Analyses
PrimeSim SPICE RF Envelope Following Analysis



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your envelope following analysis is now set up.

PrimeSim SPICE Circuit Checks

The following circuit checks for PrimeSim simulations are available:

- DC Path
- Device Operating Point
- High Impedance State Node
- Expression
- Rise/Fall Transition Time
- Timing Setup/Hold/Delay/Width
- Active/Inactive Nodes
- Device Current
- Check and Report Toggle Count

Note:

For more information on the PrimeSim Circuit Check analyses, type `primesim -webhelp` on the PrimeSim command line. When the PrimeSim help opens, search for `circuit checks` to find information about usage, syntax, and options.

To create a Circuit Check analysis:

1. Choose **Setup > Circuit Check** from the PrimeWave Design Environment main menu bar.

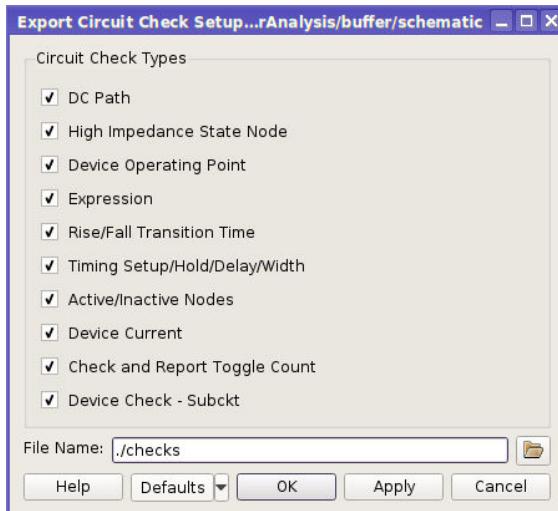
The **Circuit Check Setup Options** dialog box opens.

2. Select the **Circuit Check Type**.
3. Enter check-specific parameters. These generally correspond to PrimeSim circuit check options. Consult the PrimeSim documentation for information by typing `primesim -webhelp` on the PrimeSim command line.



4. Click  to navigate to the schematic view and pick subcircuits, elements, and nodes to include in the check.
5. Click **Add** to add the check to the Circuit Check table. Click **Clear** to clear the entries in the dialog box. Click **Modify** to edit an existing check, selected from the Circuit Check table.

6. (Optional) Right-click a column header in the Circuit Check table and choose **Hide** to hide that column. To undo the **Hide** action, right-click in the Circuit Check table to open the menu and select **Show/Hide Columns** to show or hide columns in the table.
7. (Optional) Right-click a column header in the Circuit Check table and choose **Freeze** to keep that data visible on the left side of the table as you scroll. Right-click the column header and choose **Unfreeze** to unfreeze that column.
8. (Optional) To save the settings of the **PrimeSim Circuit Check Setup Options** dialog box to a specified file, click the **Export** button  to open the **Export Circuit Check Setup** dialog box and choose a **File Name**. Later, if you want to use the same settings, you can use the **Import** button  to import that file.



Click **OK** or **Apply** to close the **Export Circuit Check Setup** dialog box.

9. (Optional) Under the Circuit Check table, click  to enable all of the checks,  to disable all of the checks, or  to delete the selected check.
10. Click **OK** or **Apply**.

Your Circuit Check analysis is now set up.

Results can be accessed using **Results > Print > Simulation Check Viewer**. See [Using the Simulation Check Viewer](#) for information on viewing the results of this check.

7

Setting Up PrimeSim Pro Analyses

This chapter contains information on how to set up and enable PrimeSim Pro analyses.

Note:

For more information on a particular PrimeSim analysis, enter `primesim -webhelp` on the PrimeSim command line. Once the PrimeSim help opens, search for the name of an analysis to find information regarding usage, syntax, and options.

Note:

If you want to use an analysis card for any applicable PrimeSim analyses, click the **Use <analysis> card** option at the bottom of the analysis setup page, and enter any needed analysis commands in the text box. When the **Use <analysis> card** option is enabled, any analysis settings you set up on the analysis setup page are disabled and have no effect on netlisting.

Any analysis statements entered in the analysis card text box are netlisted as-is with no error checking. If you add any additional supported sweeps (parameter, .data, or Monte Carlo, for example), the simulation runs, but all internal sweeps are blocked.

To create or edit an analysis, choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The following analyses are available in the PrimeSim Pro integration:

- [PrimeSim Pro Transient Analysis](#)
- [PrimeSim Pro DC Analysis](#)
- [PrimeSim Pro AC Analysis](#)
- [PrimeSim Pro Operating Point Analysis](#)
- [PrimeSim Pro Circuit Checks](#)

PrimeSim Pro Transient Analysis

A transient analysis calculates the circuit solution as a function of time and over a specified time range.

To create a transient analysis:

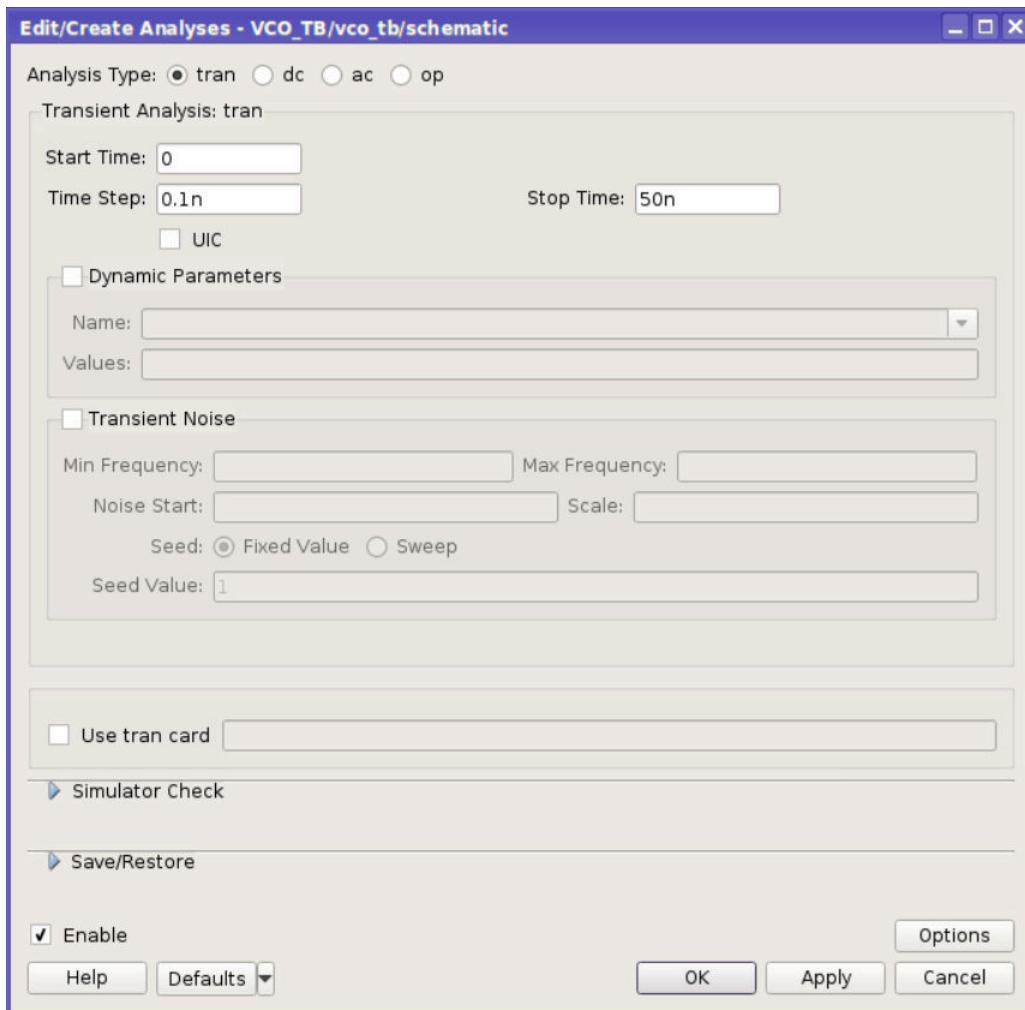
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **tran** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.

4. Click **OK** or **Apply**.

Your transient analysis is now set up.

See Also

- [.TRAN \(Transient Analysis\)](#) in the *PrimeSim™ User Guide: Pro and SPICE*

PrimeSim Pro DC Analysis

To create a DC analysis:

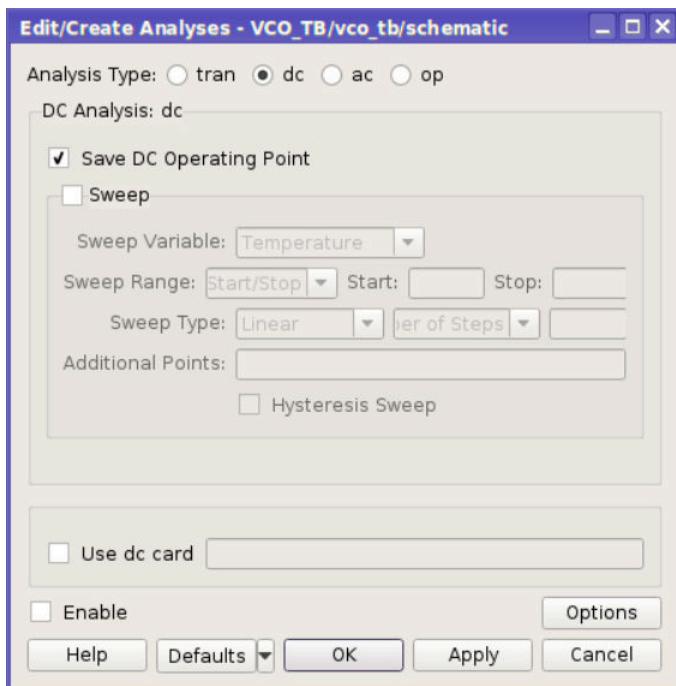
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **dc** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your DC analysis is now set up.

See Also

- [.DC \(DC Analysis\)](#) in the *PrimeSim™ User Guide: Pro and SPICE*

PrimeSim Pro AC Analysis

An AC analysis calculates the AC output variables as a function of frequency.

To create an AC analysis:

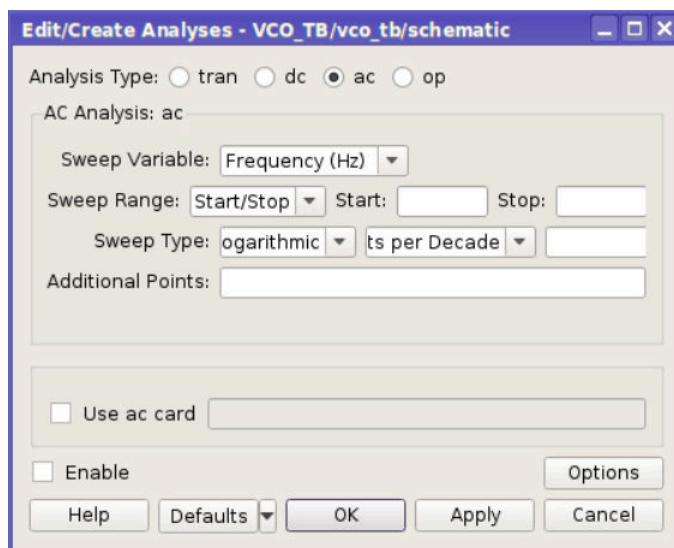
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **ac** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your AC analysis is now set up.

See Also

- [.AC \(AC Analysis\)](#) in the *PrimeSim™ User Guide: Pro and SPICE*

PrimeSim Pro Operating Point Analysis

Note:

If you want to be able to annotate or print device operating points for any time point other than time=0, set up a transient analysis as well. See [PrimeSim Pro Transient Analysis](#).

To create an operating point analysis:

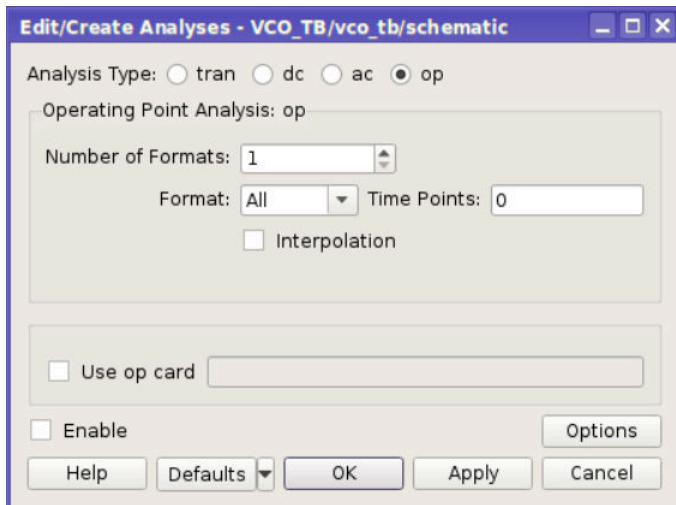
1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **op** radio button.



3. Click **Enable** to enable this analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your operating point analysis is now set up.

See Also

- [.OP in the PrimeSim™ User Guide: Pro and SPICE](#)

PrimeSim Pro Circuit Checks

The following circuit checks for PrimeSim simulations are available:

- DC Path
- Device Operating Point
- High Impedance State Node
- Expression
- Rise/Fall Transition Time
- Timing Setup/Hold/Delay/Width
- Active/Inactive Nodes
- Device Current
- Check and Report Toggle Count

Note:

For more information on the PrimeSim Circuit Check analyses, type `primesim -webhelp` on the PrimeSim command line. When the PrimeSim help opens, search for `circuit checks` to find information about usage, syntax, and options.

To create a Circuit Check analysis:

1. Choose **Setup > Circuit Check** from the PrimeWave Design Environment main menu bar.

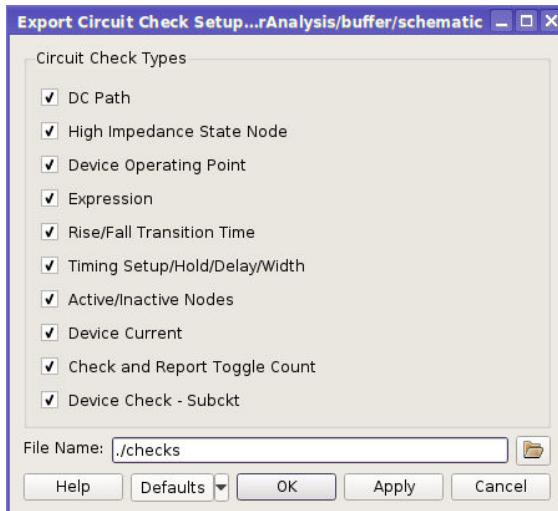
The **Circuit Check Setup Options** dialog box opens.

2. Select the **Circuit Check Type**.
3. Enter check-specific parameters. These generally correspond to PrimeSim circuit check options. Consult the PrimeSim documentation for information by typing `primesim -webhelp` on the PrimeSim command line.



4. Click  to navigate to the schematic view and pick subcircuits, elements, and nodes to include in the check.
5. Click **Add** to add the check to the Circuit Check table. Click **Clear** to clear the entries in the dialog box. Click **Modify** to edit an existing check, selected from the Circuit Check table.

6. (Optional) Right-click a column header in the Circuit Check table and choose **Hide** to hide that column. To undo the **Hide** action, right-click in the Circuit Check table to open the menu and select **Show/Hide Columns** to show or hide columns in the table.
7. (Optional) Right-click a column header in the Circuit Check table and choose **Freeze** to keep that data visible on the left side of the table as you scroll. Right-click the column header and choose **Unfreeze** to unfreeze that column.
8. (Optional) To save the settings of the **PrimeSim Circuit Check Setup Options** dialog box to a specified file, click the **Export** button  to open the **Export Circuit Check Setup** dialog box and choose a **File Name**. Later, if you want to use the same settings, you can use the **Import** button  to import that file.



Click **OK** or **Apply** to close the **Export Circuit Check Setup** dialog box.

9. (Optional) Under the Circuit Check table, click  to enable all of the checks,  to disable all of the checks, or  to delete the selected check.
10. Click **OK** or **Apply**.

Your Circuit Check analysis is now set up.

Results can be accessed using **Results > Print > Simulation Check Viewer**. See [Using the Simulation Check Viewer](#) for information on viewing the results of this check.

8

Setting Up PrimeSim XA Analyses

This chapter contains information on how to set up and enable PrimeSim XA analyses.

To create or edit an analysis, choose **Setup > Analyses** from the main menu bar.

Note:

For more information on a particular PrimeSim XA analysis, enter `primesim -webhelp` on the command line within PrimeSim XA. Once the PrimeSim XA help opens, search for the name of an analysis to find information regarding usage, syntax, and options.

Note:

If you want to use an analysis card for any applicable PrimeSim XA analyses, click **Use <analysis> card** at the bottom of the analysis setup page, and enter any needed analysis commands in the text box. When **Use <analysis> card** is enabled, any analysis settings you set up on the analysis setup page are disabled and have no effect on netlisting.

Any analysis statements entered in the analysis card text box are netlisted as-is with no error checking. If you add any additional supported sweeps (parameter, .data, or Monte Carlo, for example), the simulation runs, but all internal sweeps are blocked.

This chapter contains the following sections:

- [PrimeSim XA Analyses](#)
- [PrimeSim XA \(Eldo\) Analyses](#)

PrimeSim XA Analyses

- [PrimeSim XA Transient Analysis](#)
- [PrimeSim XA Operating Point Analysis](#)

For further information on these analyses, see the *PrimeSim XA User Guide*.

PrimeSim XA Transient Analysis

To create a transient analysis:

1. Choose **Setup > Analyses** from the main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the analysis pane.

2. Click the **tran** radio button.
3. Specify **Time Step** and **Stop Time** values.
4. (Optional) Click **UIC** to bypass the initial DC operating point.

When this option is enabled, PrimeSim XA does not calculate the initial DC operating point but directly enters transient analysis. Transient analysis uses the .IC initialization values as part of the solution for the initial timepoint.

5. (Optional) Expand the **Simulator Check** section to enable the **Check Safe Operating Area** check.
6. (Optional) Expand the **Save/Restore** section to save and restore time points.
See [Enabling Save and Restore Times](#).
7. Click **Enable** to enable this analysis as part of your testbench.
8. Click **OK** or **Apply**.

Your transient analysis is now set up.

PrimeSim XA Operating Point Analysis

Note:

To create an operating point analysis, you must first run a transient analysis.
See [PrimeSim XA Transient Analysis](#).

To create an operating point analysis:

1. Choose **Setup > Analyses** from the main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the analysis pane.

2. Click the **op** radio button.
3. Enter a value for **Times**.
4. Click **Enable** to enable this analysis as part of your testbench.
5. Click **OK** or **Apply**.

Your operating point analysis is now set up.

PrimeSim XA (Eldo) Analyses

Note:

For more information on a particular PrimeSim XA analysis, enter `primesim -webhelp` on the command line within PrimeSim XA. Once the PrimeSim XA help opens, search for the name of an analysis to find information regarding usage, syntax, and options.

Note:

If you want to use an analysis card for any applicable PrimeSim XA analyses, click **Use <analysis> card** at the bottom of the analysis setup page, and enter any needed analysis commands in the text box. When the **Use <analysis> card** option is enabled, any analysis settings you set up on the analysis setup page are disabled and have no effect on netlisting.

Any analysis statements entered in the analysis card text box are netlisted as-is with no error checking. If you add any additional supported sweeps (parameter, .data, or Monte Carlo, for example), the simulation runs, but all internal sweeps are blocked.

- [PrimeSim XA \(ElDO\) Transient Analysis](#)
 - [PrimeSim XA \(ElDO\) Operating Point Analysis](#)
-

PrimeSim XA (ElDO) Transient Analysis

To create a transient analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the analysis pane.

2. Click the **tran** radio button.
3. Specify **Time Step** and **Stop Time** values.
4. (Optional) Click **UIC** to bypass the initial DC operating point.

When this option is enabled, PrimeSim XA does not calculate the initial DC operating point but directly enters transient analysis. Transient analysis uses the .IC initialization values as part of the solution for the initial timepoint.

5. Click **Enable** to enable this analysis as part of your testbench.
6. Click **OK** or **Apply**.

Your transient analysis is now set up.

PrimeSim XA (ElDO) Operating Point Analysis

Note:

To create an operating point analysis, you must first run a transient analysis.
See [PrimeSim XA \(ElDO\) Transient Analysis](#).

To create an operating point analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the analysis pane.

2. Click the **op** radio button.
3. Enter a value for **Times**.
4. Click **Enable** to enable this analysis as part of your testbench.
5. Click **OK** or **Apply**.

Your operating point analysis is now set up.

9

Setting Up VCS PrimeSim AMS Analyses

This chapter contains information on how to set up and enable VCS PrimeSim AMS analyses.

To create or edit an analysis, choose **Setup > Analyses** from the main menu bar.

The following analyses are available in the VCS PrimeSim AMS integration:

- [VCS PrimeSim AMS Transient Analysis](#)
- [VCS PrimeSim AMS AC Analysis](#)
- [VCS PrimeSim AMS Operating Point Analysis](#)

For further information on these analyses, see the *VCS PrimeSim AMS User Guide*.

VCS PrimeSim AMS Transient Analysis

A transient analysis calculates the circuit solution as a function of time and over a specified time range.

To create a transient analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **tran** radio button.
3. Specify **Time Step** and **Stop Time** values for the interval.
4. (Optional) Click **UIC** to bypass the initial DC operating point.

When this option is enabled, VCS PrimeSim AMS does not calculate the initial DC operating point but directly enters transient analysis. Transient analysis uses the .IC initialization values as part of the solution for the initial timepoint.

5. Click **Enable** to enable this analysis as part of your testbench.

6. Click **OK** or **Apply**.

Your transient analysis is now set up.

VCS PrimeSim AMS AC Analysis

An AC analyses calculates the AC output variables as a function of frequency.

To create an AC analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **ac** radio button.

3. Choose the sweep variable type using **Sweep Variable: Frequency (Hz), Design Variable, Temperature, or None**.

To specify a **Frequency (Hz)** variable type, click **Frequency (Hz)** and continue to [Step 4](#).

To specify a **Design Variable** type, click **Design Variable** and select a **Variable** from the drop-down menu. Continue to [Step 4](#)

To specify a **Temperature** variable type, click **Temperature** and continue to [Step 4](#).

4. Choose a **Sweep Range**:

- **Start/Stop:** Specify the **Start** and **Stop** values.
- **Center/Span:** Specify the **Center** and **Span** values.

5. Choose a **Sweep Type**:

- **Linear:** Specify the **Number of Steps** or **Step Size** values.
- **Logarithmic:** Specify the **Points per Decade** and **Number of Steps** values.
- **Automatic**

6. (Optional) Specify any **Additional Points**.

7. Click **Enable** to enable this analysis as part of your testbench.

8. Click **OK** or **Apply**.

Your AC analysis is now set up.

VCS PrimeSim AMS Operating Point Analysis

Note:

If you want to be able to annotate or print device operating points for any time point other than time=0, set up a transient analysis as well. See [VCS PrimeSim AMS Transient Analysis](#).

To create an operating point analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **op** radio button.
3. Enter the **Times** you want to include in the operating point analysis.
4. Click **Enable** to enable this analysis as part of your testbench.
5. Click **OK** or **Apply**.

Your operating point analysis is now set up.

10

Setting Up FineSim Analyses

This chapter contains information on how to set up and enable FineSim analyses.

To create or edit an analysis, choose **Setup > Analyses** from the main menu bar.

Note:

For more information about a particular FineSim analysis, enter `finesim -webhelp` on the FineSim command line. Once the FineSim help opens, search for the name of an analysis to find information regarding usage, syntax, and options.

Note:

If you want to use an analysis card for any applicable FineSim analyses, click **Use <analysis> card** at the bottom of the analysis setup page, and enter any needed analysis commands in the text box. When the **Use <analysis> card** option is enabled, any analysis settings you set up on the analysis setup page are disabled and have no effect on netlisting.

Any analysis statements entered in the analysis card text box are netlisted as-is with no error checking. If you add any additional supported sweeps (parameter, .data, or Monte Carlo, for example), the simulation runs, but all internal sweeps are blocked.

This chapter contains the following sections:

- [FineSim Analyses](#)
- [FineSim VCS Analyses](#)
- [Setting Performance Options](#)

FineSim Analyses

FineSim supports the following analyses in HSPICE mode:

- [FineSim Transient Analysis](#)
- [FineSim Operating Point Analysis](#)
- [FineSim DC Analysis](#)
- [FineSim AC Analysis](#)
- [FineSim Noise Analysis](#)
- [FineSim FFT Analysis](#)
- [FineSim Loop Stability Analysis](#)
- [FineSim Circuit Check Analysis](#)

Note:

Of these HSPICE-based analyses, only AC, Noise and Loop Stability strictly require SPICE mode (because they operate in the frequency domain). The FineSim tool reports an error if one of those three analyses are run in Pro mode.

See Also

- [Transient Analysis](#) in the *PrimeSim™ HSPICE® User Guide: Basic Simulation and Analysis*
-

FineSim Transient Analysis

To create a transient analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Choose **tran** for the **Analysis Type** field.

Note:

The setup for **tran** analysis is exactly the same as HSPICE, except that there is no separate tran noise and tran statement for FineSim. Transient noise is part of the tran statement. If the **Transient Noise** Group is enabled, it is netlisted.

3. (Optional) Specify a value for the **Simulation Start Time**.
4. (Optional) Specify a value for the **Output Start Time**.
5. (Optional) Select the number of transient intervals (**Number of Intervals**) to include in the analysis. With multiple intervals you can specify different step times along a transient analysis.

A pair of **Time Step** and **Stop Time** text boxes appear or disappear for each additional interval you add or delete.

6. Specify **Time Step** and **Stop Time** values for each interval.
7. (Optional) Enable **UIC** to bypass the initial DC operating point.
8. (Optional) Specify a **Strobe Delay** (default is 0) and a **Strobe Period**.
9. (Optional) Expand the **Advanced Settings** section to set **Temp Sweep** options.
10. Enable the **Transient Noise** group and specify the value for **Min Frequency**, **Max Frequency**, **Seed**, and **Scale Factor**.

If **Transient Noise** is not required, skip to [Step 11](#).

11. (Optional) Expand the **Simulator Check** section to enable the **Check SOA** (Safe Operating Area) check.

Note:

This is only available with simulator version N-2017.12 or higher.

12. (Optional) Expand the **Save/Restore** section to save and restore time points.

See [Enabling Save and Restore Times](#).

13. Check **Enable** to allow transient analysis as part of your testbench.
14. Click **OK** or **Apply**.

Your transient analysis is now set up.

FineSim Operating Point Analysis

To create an operating point analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Choose **op** for the **Analysis Type** field.

The operating point analysis supports list input.

3. Specify multiple time points in the **Times** field.

4. Check **Enable** to allow operating point analysis as part of your testbench.

5. Click **OK** or **Apply**.

Your operating point analysis is now set up.

FineSim DC Analysis

To create a DC analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Choose **dc** for the **Analysis Type** field.

3. Choose the sweep variable type using **Sweep Variable: Source, Design Variable, Temperature, or Data Driven**.

To specify a **Source** variable type, choose **Source** and continue to [Step 4](#).

To specify a **Design Variable** type, choose **Design Variable** and skip to [Step 5](#).

To specify a **Temperature** variable type, choose **Temperature** and skip to [Step 6](#).

To specify a **Data Driven** variable type, click **Data Driven** and skip to [Step 7](#).

4. Enter the name of the source in the **Source Name** text box, or click the **Select Source Name** button  and click on a source in the Schematic Editor.
 Return to the **Edit/Create Analyses** dialog box, and skip to [Step 6](#).
5. Specify a **Variable Name**.
6. Choose a **Sweep Type**:
 - **Linear Steps**: Specify the **Start** and **Stop** values, as well as the **Step Size**.
 - **Linear Points**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
 - **Octave**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
 - **Decade**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
 - **Points of Interest**: Specify the **Points** values separated by commas.
7. Click **Manage Parameters** to open the **Manage Data Driven Sweep Parameters** dialog box. Enter the parameters in the dialog box and click **OK**. Enter the data into the table.
8. (Optional) Enable **Hysteresis Sweep** to perform a hysteresis sweep.
9. Check **Enable** to allow DC analysis as part of your testbench.
10. Click **OK** or **Apply**.

Your DC analysis is now set up.

FineSim AC Analysis

To create an AC analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Choose **ac** for the **Analysis Type** field.

3. Choose a **Sweep Type**:

- **Linear**: Specify the **Start** and **Stop** values, as well as the **No. of Points**.
- **Octave**: Specify the **Start** and **Stop** values, as well as the **No. of Points** per octave.
- **Decade**: Specify the **Start** and **Stop** values, as well as the **No. of Points** per decade.
- **Points of Interest**: Specify the **Points** values separated by commas.

4. Check **Enable** to allow AC analysis as part of your testbench.

5. Click **OK** or **Apply**.

Your AC analysis is now set up.

FineSim Noise Analysis

Note:

FineSim does not support **Branch Current** Output Type, though HSPICE supports this feature.

To create a noise analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Ensure the analyses on the **General** tab are visible, and choose **noise** for the **Analysis Type** field.

3. Enter the **Positive Node** field either manually or by clicking on the **Select** button .

You are allowed to specify the net for this field.

4. Enter the **Negative Node** field either manually or by clicking on the **Select** button .

You are allowed to specify the net for this field.

5. Enter the **Source Name** field either manually or by clicking on the **Select Source** button .

6. Specify a value for the **Frequency Interval** field.
7. Use the **List Frequencies** option to select how many frequencies to list in the noise summary report: **all**, **none**, or specific **frequencies** in a comma-separated list.
8. Choose whether or not to **List Sources**. This prints the element noise value to a `.lis` file when the element has multiple noise sources. When this option is enabled, you can enter values for **List Share**, **List Count**, and **List Floor**.
9. Check **Enable** to allow noise analysis as part of your testbench.
10. Click **OK** or **Apply**.

Your noise analysis is now set up.

FineSim FFT Analysis

To create an FFT analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Choose **fft** for the **Analysis Type** field.
3. Choose a name for the analysis.

The analysis is named "fft" by default. Click the buttons below the **Analysis Name** menu to add a new FFT analysis , make a copy of the currently selected FFT analysis , change the currently selected name  or delete the currently selected name .

4. Choose an **Output Variable**:

- To specify a voltage as the output, click **Voltage** and specify the voltage to be used as the output.

The voltage can be a single node or the voltage between two nodes. Enter the node names directly into the **Positive Node** and **Negative Node** text boxes, or click the  **Select in Design** button  and select a wire in the Schematic Editor. Specifying a **Negative Node** is optional.

- To specify a current as the output, click **Current** and select an instance to use as the output.

Enter the instance name directly into the **Output Instance** text box, or click the



Select in Design button and select an instance in the Schematic Editor.

- To specify a power value as the output type, click **Power** and select an **Output Instance** with the desired power dissipation.

5. Enter values for the **Start Time**, **Stop Time**, and **No. of Points**.

To calculate the FFT, you must include the specified transient analysis time points, as well as the number of points used in the FFT calculation.

6. Choose an output **Format**.

NORM is normalized magnitude (default). **UNORM** is unnormalized magnitude.

7. Choose a **Window Type**.

The following window types are available. If you choose the **GAUSS** or **KAISER** window types, move on to the next step; otherwise, skip to [Step 9](#).

- **RECT**: Simple rectangular truncation window (default).
- **BART**: Bartlett window.
- **HANN**: Hanning window.
- **HAMM**: Hamming window.
- **BLACK**: Blackmann window.
- **HARRIS**: Blackmann-Harris window.
- **GAUSS**: Gaussian window.
- **KAISER**: Kaiser-Bessel window.

8. Enter a value for the **Alpha**, which is used to control highest side-lobe level and bandwidth for the **GAUSS** and **KAISER** windows.

Valid values are 1.0 to 20.0, inclusive. The default is 3.0.

9. (Optional) Specify values for the **Analysis**, **Minimum Frequency**, and **Maximum Frequency**.

10. Check **Enable** to allow FFT analysis as part of your testbench.

11. Click **OK** or **Apply**.

Your FFT analysis is now set up.

FineSim Loop Stability Analysis

To create a loop stability analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Choose **Istb** for the **Analysis Type** field.
3. Specify a **Mode**: **Single**, **Differential**, or **Common**.
4. Choose a **Voltage Source 1**. (For **Differential** and **Common** modes, also choose **Voltage Source 2**.)

You can enter the name of the source, or click the  button to choose a node in a schematic.

5. Check **Enable** to allow loop stability analysis as part of your testbench.
6. Click **OK** or **Apply**.

Your loop stability analysis is now set up.

FineSim Circuit Check Analysis

The following circuit checks for FineSim simulations are available:

- DC Path
- Device Operating Point
- High Impedance State Node
- Expression
- Rise/Fall Transition Time
- Timing Setup/Hold/Delay/Width

- Active/Inactive Nodes
- Device Current
- Check and Report Toggle Count

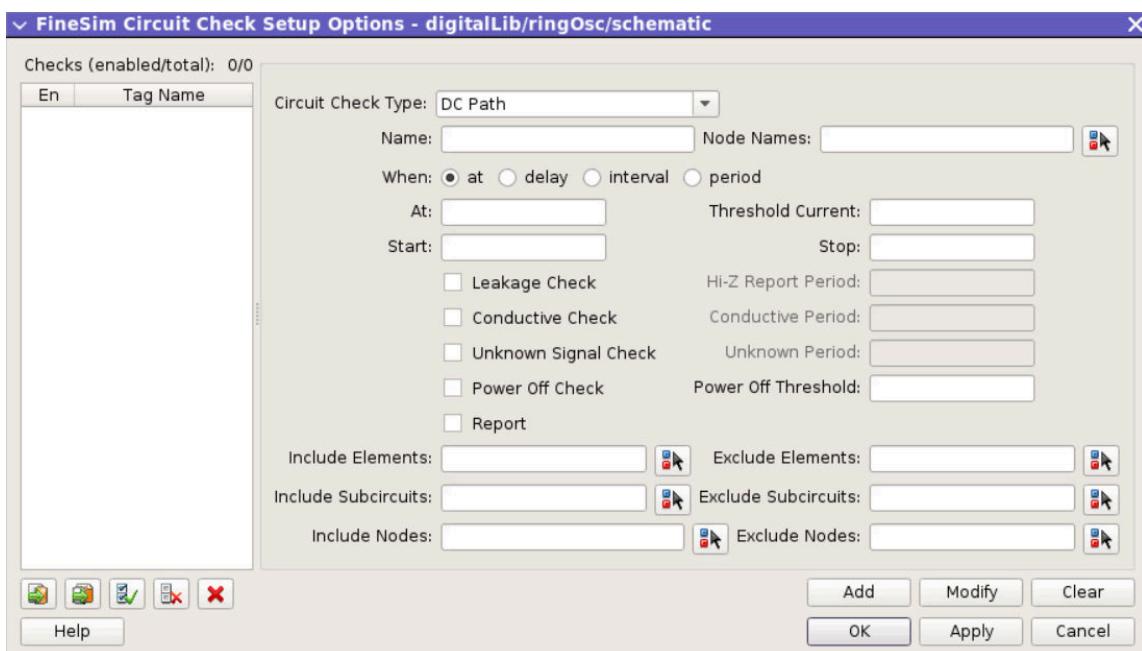
Note:

For more information on the FineSim Circuit Check Analysis, type `finesim -webhelp` on the FineSim command line. When the FineSim help opens, search for `circuit checks` to find information about usage, syntax, and options.

To create a Circuit Check analysis:

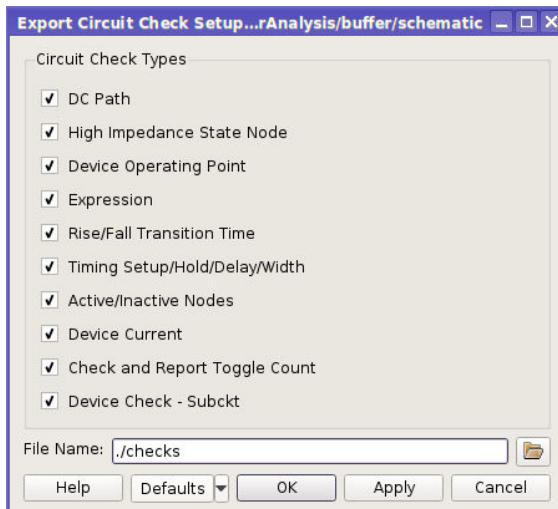
1. Choose **Setup > Circuit Check** from the PrimeWave Design Environment main menu bar.

The **Circuit Check Setup Options** dialog box opens.



2. Select the **Circuit Check Type**. The figure in [Step 1](#) shows the **Circuit Check Setup Options** dialog box for a DC Path circuit check (`.CHKDCPATH`).
3. Enter check-specific parameters. These generally correspond to FineSim circuit check options. Consult the FineSim documentation for information by typing `finesim -webhelp` on the FineSim command line.

4. Click  to navigate to the schematic view and pick subcircuits, elements, and nodes to include in the check.
5. Click **Add** to add the check to the Circuit Check table. Click **Clear** to clear the entries in the dialog box. Click **Modify** to edit an existing check, selected from the Circuit Check table.
6. (Optional) Right-click a column header in the Circuit Check table and choose **Hide** to hide that column. To undo the **Hide** action, right-click in the Circuit Check table to open the menu and select **Show/Hide Columns** to show or hide columns in the table.
7. (Optional) Right-click a column header in the Circuit Check table and choose **Freeze** to keep that data visible on the left side of the table as you scroll. Right-click the column header and choose **Unfreeze** to unfreeze that column.
8. (Optional) To save the settings of the **FineSim Circuit Check Setup Options** dialog box to a specified file, click the **Export** button  to open the **Export Circuit Check Setup** dialog box and choose a **File Name**. Later, if you want to use the same settings, you can use the **Import** button  to import that file.



Click **OK** or **Apply** to close the **Export Circuit Check Setup** dialog box.

9. (Optional) Under the Circuit Check table, click  to enable all of the checks,  to disable all of the checks, or  to delete the selected check.
10. Click **OK** or **Apply**.

Your Circuit Check analysis is now set up.

Results can be accessed using **Results > Print > Simulation Check Viewer**. See [Using the Simulation Check Viewer](#) for information on viewing the results of this check.

FineSim VCS Analyses

The following analyses are available in the FineSim VCS integration:

- [FineSim VCS Transient Analysis](#)
- [FineSim VCS Operating Point Analysis](#)

For further information on these analyses, see the *FineSim User Guide: Pro and SPICE Reference*.

FineSim VCS Transient Analysis

A transient analysis calculates the circuit solution as a function of time and over a specified time range.

To create a transient analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **tran** radio button.
3. Specify **Time Step** and **Stop Time** values for the interval.
4. (Optional) Click **UIC** to bypass the initial DC operating point.

When this option is enabled, FineSim does not calculate the initial DC operating point but directly enters transient analysis. Transient analysis uses the .IC initialization values as part of the solution for the initial timepoint.

5. Click **Enable** to enable this analysis as part of your testbench.
6. Click **OK** or **Apply**.

Your transient analysis is now set up.

FineSim VCS Operating Point Analysis

Note:

If you want to be able to annotate or print device operating points for any time point other than time=0, set up a transient analysis as well. See [FineSim VCS Transient Analysis](#).

To create an operating point analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **op** radio button.
3. Enter the Times you want to include in the operating point analysis.
4. Click **Enable** to enable this analysis as part of your testbench.
5. Click **OK** or **Apply**.

Your operating point analysis is now set up.

Setting Performance Options

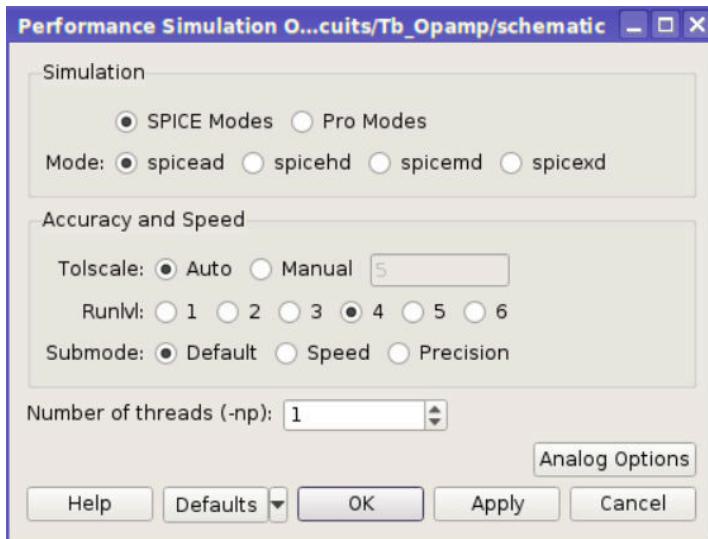
Note:

Performance options are available with FineSim analyses only.

Performance options are used to fine-tune the FineSim simulator performance in different modes of simulation to attain the desired level of results accuracy.

To specify performance options:

1. Choose **Setup > Performance Options**. The **Performance Simulation Options** dialog box opens.



2. Select **SPICE Modes** or **Pro Modes**.
3. For SPICE modes, select a **Mode**: spicead, spicehd, spicemd, or spicexd. For **Pro Modes**, skip to [Step 5](#).
4. Select SPICE mode options as described in [Table 5](#). For information on these mode options, see [finesim_mode](#) in the *FineSim User Guide: Pro and SPICE Reference*.

Table 5 *Performance Simulation SPICE Mode Options*

Options	Description
spicead	
Accuracy and Speed	Set the following accuracy and speed options: Tolscale : Select Auto or Manual . If you choose Tolscale: Manual , set the Tolerance Scale. Runlvl : Choose from 1 through 6. The default is 2. Submode : Select Default , Speed , or Precision .
Number of threads (-np)	The default is 0.
Analog Options	Click to open the Analog Options dialog box. See Specifying FineSim Analog Options for option descriptions.

Options	Description
spicehd	
Accuracy and Speed	Set the following accuracy and speed options: Runlvl: Choose from 1 through 6 . The default is 2 . Model: Select 2, 3, 4 (the default), 4.5 , or 5 . Speed: Select 0, 0.5 (the default), 1, 2, 3, 4 , or 5 .
Number of threads (-np)	The default is 0.
Analog Options	Click to open the Analog Options dialog box. See Specifying FineSim Analog Options for option descriptions.
spicemd	
Accuracy and Speed	Set the following accuracy and speed options: Runlvl: Choose from 1 through 6 . The default is 2 . Model: Select 2, 3, 4 (the default), 4.5 , or 5 . Speed: Select 0, 0.5, 1 (the default), 2, 3, 4 , or 5 .
Number of threads (-np)	The default is 0.
Analog Options	Click to open the Analog Options dialog box. See Specifying FineSim Analog Options for option descriptions.
spicexd	
Accuracy and Speed	Set the following accuracy and speed options: Runlvl: Choose from 1 through 6 . The default is 2 . Model: Select 2, 3 (the default), 4, 4.5 , or 5 . Speed: Select 0, 0.5, 1 (the default), 2, 3, 4 , or 5 .
Number of threads (-np)	The default is 0.
Analog Options	Click to open the Analog Options dialog box. See Specifying FineSim Analog Options for option descriptions.

Note:

When switching modes from spicehd/spicemd/spicexd to spicead, the **Reset speed and model options to simulator defaults** is added to the dialog box. If you enable the **Reset** option and click **OK/Apply**, then close the **Performance Simulation Options** dialog box, finesim_model and finesim_speed options set in spicehd will be forgotten. Additionally, if you enable **Reset**, click **Apply** and leave the **Performance Simulation Options** dialog box open, you can still see the value you set for spicehd in the dialog

box. However, unless you press **OK/Apply** for spicehd again, you will lose these values when you close the dialog box.

5. For Pro modes, select a **Mode**: prohd, promd, or proxd.
6. Select Pro mode options as described in [Table 6](#). For information on these mode options, see [finesim_mode](#) in the *FineSim User Guide: Pro and SPICE Reference*.

Table 6 Performance Simulation Pro Mode Options

Options	Description
prohd	
Accuracy and Speed	Set the following accuracy and speed options: Runlvl : Choose from 1 through 6 . The default is 2 . Model : Select 2 , 3 (the default), 4 , 4.5 , or 5 . Speed : Select 0 , 0.5 , 1 (the default), 2 , 3 , 4 , or 5 . Power Block : Select 0 or 1 (the default).
Number of threads (-np)	The default is 0.
Analog Options	Click to open the Analog Options dialog box. See Specifying FineSim Analog Options for option descriptions.
promd	
Accuracy and Speed	Set the following accuracy and speed options: Runlvl : Choose from 1 through 6 . The default is 2 . Model : Select 2 (the default), 3 , 4 , 4.5 , or 5 . Speed : Select 0 , 0.5 , 1 , 2 (the default), 3 , 4 , or 5 . Power Block : Select 0 or 1 (the default).
Number of threads (-np)	The default is 0.
Analog Options	Click to open the Analog Options dialog box. See Specifying FineSim Analog Options for option descriptions.
proxd	
Accuracy and Speed	Set the following accuracy and speed options: Runlvl : Choose from 1 through 6 . The default is 2 . Model : Select 2 (the default), 3 , 4 , 4.5 , or 5 . Speed : Select 0 , 0.5 , 1 , 2 , 3 (the default), 4 , or 5 . Power Block : Select 0 or 1 (the default).
Number of threads (-np)	The default is 0.
Analog Options	Click to open the Analog Options dialog box. See Specifying FineSim Analog Options for option descriptions.

7. Click **OK** or **Apply**.

Your performance options are now set up.

Specifying FineSim Analog Options

Analog options are used to improve the performance of the simulator.

To specify analog options:

1. Click **Analog Options** in the **Performance Simulation Options** dialog box to open the **Analog Options** dialog box.
2. Select analog common options from the **Common Options** tab, as described in [Table 7](#).

Table 7 FineSim Analog Common Options

Options	Description
Accuracy	Model: (For spicead mode only.) Select 2, 3, 4, 4.5, or 5 . Integration Method: Select AUTO, TRAP, BE, or GEAR . Max Time Step: Sets the maximum time step. Time Unit: (For SPICE modes only.)
DC Convergence	DC Algorithm: Select 0, 1, 2, 3, or 4 . DC Effort: Select 0, 1, 2, or 3 .
Post Simulation	Set the following post simulation options: Rcred: Select 0, 1, 2, or 3 . Posti: Select 0, 1, 2, or 3 . Reduction: Select 0, 1, 2, or 3 . Min Resistance: The default is 0.001.

3. When you have selected the necessary options, click **Add** to add the `.option` statement. Use the up and down arrows to arrange the order of the `.option` statements. Use the **Delete** button to delete an `.option` statement.
4. Switch to the **Misc** tab.
5. Select miscellaneous analog options from the **Misc** tab, as described in [Table 8](#).

Table 8 *FineSim Analog Misc Options*

Options	Description
Output Control	<p>Set the following output controls:</p> <p>vprbtol: Sets the voltage tolerance for the specified node.</p> <p>iprbtol: Sets the current tolerance for the specified node.</p> <p>Warn limit: The number of warning to be displayed in the simulator log file.</p> <p>tflush: Flushes output files at the specified time interval of analysis.</p> <p>For details on these options, see the FineSim Pro Command Reference in the <i>FineSim User Guide: Pro and SPICE Reference</i>.</p>
Block Level Setting	Allows you to set block-level FineSim options. For details on these options, see the FineSim Pro Command Reference in the <i>FineSim User Guide: Pro and SPICE Reference</i> .
Other Options	Allows you to set other FineSim options. For details on these options, see the FineSim Pro Command Reference in the <i>FineSim User Guide: Pro and SPICE Reference</i> .

6. (Optional) Click **Performance Options** to open the **Performance Options** dialog box. See [Setting Performance Options](#) for option descriptions.
7. Click **OK** or **Apply**.

Your analog options are now set up.

11

Setting Up FineSim VCS AMS Analyses

This chapter contains information on how to set up and enable FineSim VCS AMS analyses.

To create or edit an analysis, choose **Setup > Analyses** from the main menu bar.

The following analyses are available in the FineSim VCS integration:

- [FineSim VCS Transient Analysis](#)
 - [FineSim VCS Operating Point Analysis](#)
-

FineSim VCS Transient Analysis

A transient analysis calculates the circuit solution as a function of time and over a specified time range.

To create a transient analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **tran** radio button.
3. Specify **Time Step** and **Stop Time** values for the interval.
4. (Optional) Click **UIC** to bypass the initial DC operating point.

When this option is enabled, FineSim does not calculate the initial DC operating point but directly enters transient analysis. Transient analysis uses the .IC initialization values as part of the solution for the initial timepoint.

5. Click **Enable** to enable this analysis as part of your testbench.

6. Click **OK** or **Apply**.

Your transient analysis is now set up.

FineSim VCS Operating Point Analysis

Note:

If you want to be able to annotate or print device operating points for any time point other than time=0, set up a transient analysis as well. See [FineSim VCS Transient Analysis](#).

To create an operating point analysis:

1. Choose **Setup > Analyses** from the PrimeWave Design Environment main menu bar.

The **Edit/Create Analyses** dialog box opens.

Note:

You can also open the **Edit/Create Analyses** dialog box by double-clicking anywhere in the PrimeWave Design Environment analysis pane.

2. Click the **op** radio button.
3. Enter the **Times** you want to include in the operating point analysis.
4. Click **Enable** to enable this analysis as part of your testbench.
5. Click **OK** or **Apply**.

Your operating point analysis is now set up.

12

Specifying Outputs and Output Options

This chapter contains information on how to set up outputs and output options.

This chapter contains the following topics:

- [Setting Up Outputs](#)
- [Adding Outputs from Designs](#)
- [Adding Outputs from Designs without Automatically Plotting Waveforms](#)
- [Saving, Loading, and Editing Outputs as Text](#)
- [Reordering Outputs](#)
- [Reordering Output Columns](#)
- [Controlling Output Column Visibility](#)
- [Editing Outputs in a Text Editor](#)
- [Editing Output Image Titles and Captions](#)
- [Deleting Outputs](#)
- [Showing Outputs in Designs](#)
- [Adding Output Sets](#)
- [Copying Output Sets](#)
- [Choosing Output Sets](#)
- [Deleting Output Sets](#)
- [Specifying ACE Scripts](#)
- [Specifying MATLAB Scripts](#)
- [Specifying Output Options](#)

Setting Up Outputs

You can set up outputs such as specifications, parametric reduction, and plots to be viewed in the ResultsView. In the PrimeWave Design Environment main window, there are several tabs for setting up output results:

Tab Name	Description
Outputs	<p>Set up the output results.</p> <p>Only named outputs can have specifications. If you want to set goals for a given output, you must specify a name for that output. The Specifications tab is available for only those outputs that are named. The specifications are saved in the "Outputs" state category. If you load a state with specifications in the measurement summary category, those specs are loaded into the outputs table.</p>
Specifications	Specify the Goal value and the Marginal (Poor) value for each output that is to be viewed in the result.
Scatter	Set up scatter plots and select which plots will generate images.
Histogram	Select input parameters for histogram plot axes and select which plots will generate images.
Q-Q	Specify measurements for Q-Q plots and select which plots will generate images.
Parametric Reduction	Specify corner, parameter, and Monte Carlo iterations for chart creation.

This section contains information on the following topics:

- [Specifying Outputs](#)
- [Setting Specification Goals](#)
- [Setting Up Scatter Charts](#)
- [Setting Up Histograms](#)
- [Setting Up Q-Q Plots](#)
- [Setting Up Parametric Reduction](#)
- [Adding External Images to Output](#)
- [Specifying Output Logic Radix](#)
- [Specifying Output Plot Type](#)

Specifying Outputs

To set up output:

1. Add output in one of the following ways:

- Type an output expression into the **Expression** column of the **Outputs** table in the PrimeWave Design Environment main window or from the **Output Setup** window (accessible from **Outputs > Edit**).

Note:

You can also enter `.MEASURE` statements as part of your expression directly in the **Outputs** table (and then clicking the **Add To Outputs** button), or via the **Measures Assist** tab in the Results Analyzer.

Outputs		Specifications	Scatter	Histogram	Q-Q	Parametric Reduction				
Output	▼	Expression			Value	Analyses				
CMRR	v(/out)						<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PM	phase_m(v(/out),-180)				ac		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
UnGain	gain1 f(v(/out))				ac		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
BW	bw(v(/out),3)				ac		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Click to add										

Output Set: default Import Measures Plotting Mode: replace Plot Clear

Note that when an expression evaluation errors out in the Outputs table for any reason, an error message is shown in red in the Outputs table, and you can hover over the error to see a tooltip describing it.

- Choose **Outputs > Select in Design**, and select an object from the Schematic Editor. This method uses cross-selection with the design to specify a voltage or current signal.

Selecting wires adds voltages to the outputs list, and selecting instance terminals adds current signals.

- Use the Results Analyzer to build an expression.

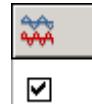
You can open the Results Analyzer directly from the Outputs table by clicking in the **Expression** column and then clicking the **Results Analyzer** button . The contents of the **Expression** field are placed in the Results Analyzer expression buffer.

2. (Optional) Click in the **Output** column, and enter a name for your output.

This name is used when displaying results, as well as building expressions that depend on the results of other expressions.

3. Click in the **Analyses** column, and choose one or more analyses from the menu.

The expression you create is evaluated against the analyses you choose.



4. (Optional) Click **Plot Waveforms/Evaluate** to add the expression to the signals that waveform viewer automatically plots when the simulation is completed. You can select more than one expression, right-click, and choose **Check Selected** or **Uncheck Selected** from the menu to select or deselect these options as a group.

This option is enabled by default for signals added directly by selecting in the design.



5. (Optional) Click **Save Output** to direct the simulator to save the output. You can select more than one expression, right-click, and choose **Check Selected** or **Uncheck Selected** from the menu to select or deselect these options as a group.

Note:

Expressions you choose to autoplot are automatically saved, so this option is disabled when you choose to autoplot the signal. If you use other save options, such as those in the **Outputs > Save Options** dialog box, selections in this column might be redundant and are displayed as a disabled option since selecting or deselecting this option has no effect.



6. (Optional) Check **Show in Results Viewer** to display the results for this output in the ResultsView. You can select more than one expression, right-click, and choose **Check Selected** or **Uncheck Selected** from the menu to select or deselect these options as a group.

This option is enabled by default. If disabled, the results are still saved in the ResultsView, but they are not visible. You can re-enable this option from the ResultsView.



7. (Optional) Click **Save Waveform Images** to save a full-scale image of the plotted waveform, which can help save disk space and be used for long-term archiving or documentation. You can select more than one expression, right-click, and choose

Check Selected or Uncheck Selected from the menu to select or deselect these options as a group.

When this option is enabled, waveform images are saved to your results directory, and they are also included with results in the ResultsView for further reference.

Note:

If you are running simulations for multiple testbenches and saving the resulting waveform images, processing might be slow. To help increase processing speed, you can run measurement jobs in the background and use a display wrapper script.

See Step [and Step 15](#) in the [Choosing a Simulator](#) section for more information.

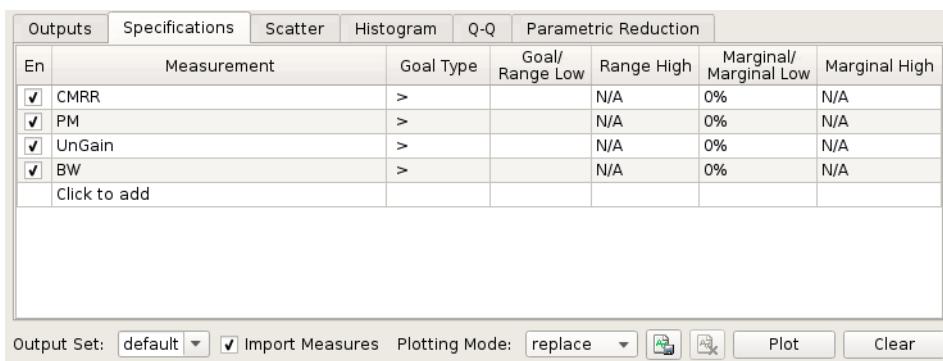
Setting Specification Goals

The page under the **Specifications** tab displays a table that has several columns (**En** (enable), **Measurement**, **Goal Type**, **Goal/Range Low**, **Range High**, **Marginal/Marginal Low**, and **Marginal High**) and multiple rows.

A goal determines at which point a measurement is a violation.

 [Show me](#) how to set specification goals.

Marginal values specify the tolerance around a specified goal that is either close to failing or close to passing. These will be highlighted in the ResultsView with transition colors: Green/Yellow for those in the marginal range but still passing; and Yellow/Red for those in the marginal range but still failing.



The screenshot shows the 'Specifications' tab in the PrimeWave Design Environment. The table has the following columns: En, Measurement, Goal Type, Goal/Range Low, Range High, Marginal/Marginal Low, and Marginal High. There are four rows of data, each with a checked 'En' column. The 'Measurement' column lists CMRR, PM, UnGain, and BW. The 'Goal Type' column contains '>'. The 'Goal/Range Low' and 'Range High' columns both show 'N/A'. The 'Marginal/Marginal Low' and 'Marginal High' columns both show '0%'. A button 'Click to add' is located at the bottom left of the table area. At the bottom of the window, there are buttons for 'Output Set' (default), 'Import Measures', 'Plotting Mode' (replace), and 'Plot/Clear'.

Outputs	Specifications	Scatter	Histogram	Q-Q	Parametric Reduction	
En	Measurement	Goal Type	Goal/Range Low	Range High	Marginal/Marginal Low	Marginal High
<input checked="" type="checkbox"/>	CMRR	>		N/A	0%	N/A
<input checked="" type="checkbox"/>	PM	>		N/A	0%	N/A
<input checked="" type="checkbox"/>	UnGain	>		N/A	0%	N/A
<input checked="" type="checkbox"/>	BW	>		N/A	0%	N/A
	Click to add					

Output Set: default ▾ Import Measures Plotting Mode: replace   Plot Clear

Each row in the table represents a specified output that you name. You can specify the various **Goal** and the **Marginal** values in the respective columns. If you disable a specification, then measurements are not checked against specifications.

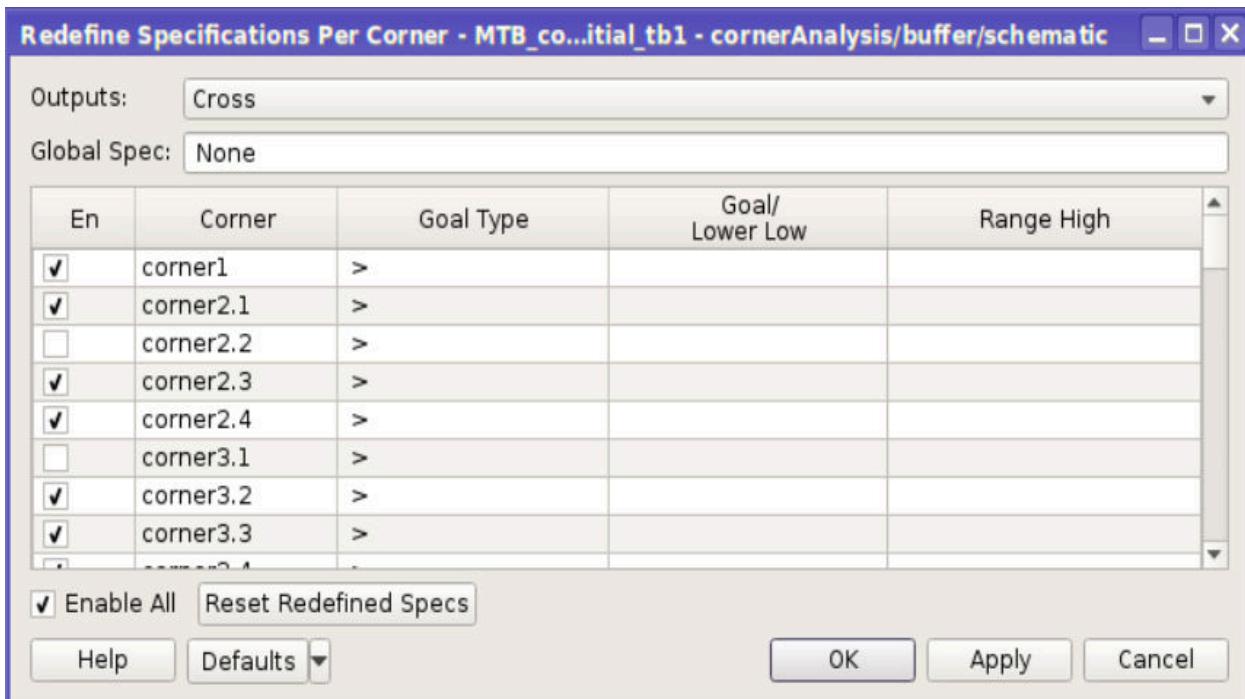
Note:

The measurement summary category is not available while saving states.

You can use the context-sensitive menu in the **Specifications** table to **Copy**, **Paste**, and **Clear** specifications.

Outputs		Specifications	Scatter	Histogram	Q-Q	Parametric Reduction	
En	Measurement	Goal Type	Goal/ Range Low	Range High	Marginal/ Marginal Low	Marginal High	
<input checked="" type="checkbox"/>	Risetime	<	85p	N/A	0%	N/A	
<input checked="" type="checkbox"/>	Falltime	<	0.25n	N/A	0%	N/A	
<input checked="" type="checkbox"/>	Cross	>		N/A	0%	N/A	
Click to add		<input type="button" value="Copy"/> <input type="button" value="Paste"/> <input type="button" value="Clear"/> <input type="button" value="Redefine Specifications Per Corner"/> <hr/> <input type="button" value="Evaluate Specifications"/>					

You can also select **Redefine Specifications Per Corner** to open the **Redefine Specifications Per Corner** dialog box and enable or disable multiple corner specifications.



To specify a goal or marginal value:

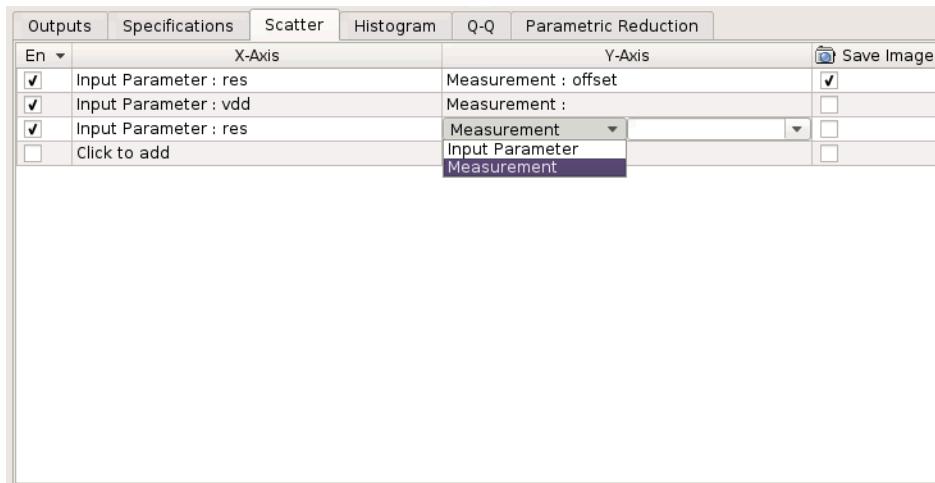
1. Click the **Specification** tab on the PrimeWave Design Environment main window.
2. Specify one of the following for **Goal Type**:
 - < (less than)
 - > (greater than)
 - = (exact)
 - **Range** (between two values)
 - <= (less than or equal to)
 - >= (greater than or equal to)
 - **Incl Range** (equal to or between two values)
3. Enter values for the chosen threshold or range in the remaining columns (**Goal/Range Low**, **Range High**, **Marginal/Marginal Low**, **Marginal High**). You can tab from column to column to enter these values.
4. Press the **Enter** key, or click anywhere outside the **Goal Type** cell to update the goal value.

To see the impact of the goal values specified, choose **Results > Results Viewer**. The ResultsView window opens.

Setting Up Scatter Charts

To set up scatter charts to be output to the results:

1. Click the **Scatter** tab on the PrimeWave Design Environment main window.



2. Click **Click to add** in the **X-Axis** and **Y-Axis** columns and set up the input parameters and measurements to plot.
3. Click **En** to enable the chart output.
4. To delete a chart output, right-click it and select **Delete**.
5. Click **Save Image** to output an image of the chart for viewing in the ResultsView.

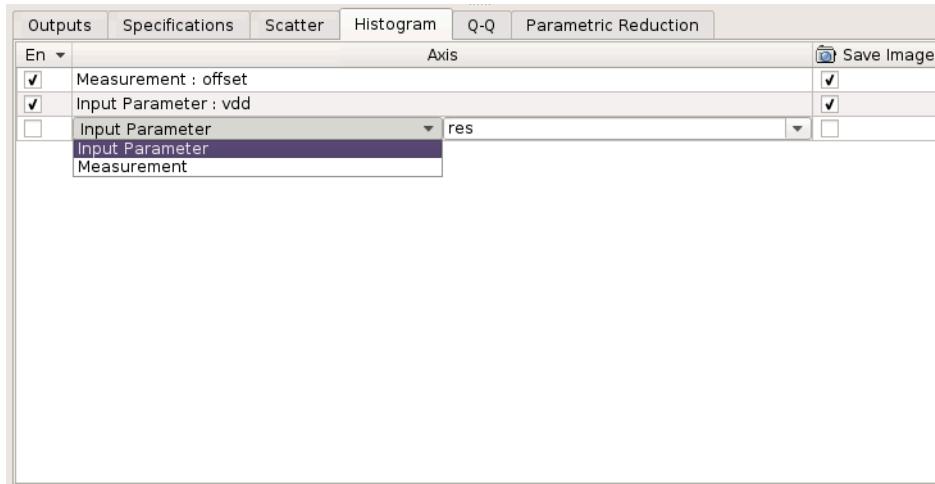
Note:

You can also set up scatter charts from the ResultsView using the **Plot Scalars** dialog box. See [Plotting Results From the ResultsView](#).

Setting Up Histograms

To set up histogram charts to be output to the results:

1. Click the **Histograms** tab on the PrimeWave Design Environment main window.



2. Click **Click to add** in the **Axis** column and set up the input parameters and measurements to plot.
3. Click **En** to enable the chart output.
4. To delete a chart output, right-click it and select **Delete**.
5. Click **Save Image** to output an image of the chart for viewing in the ResultsView.

Note:

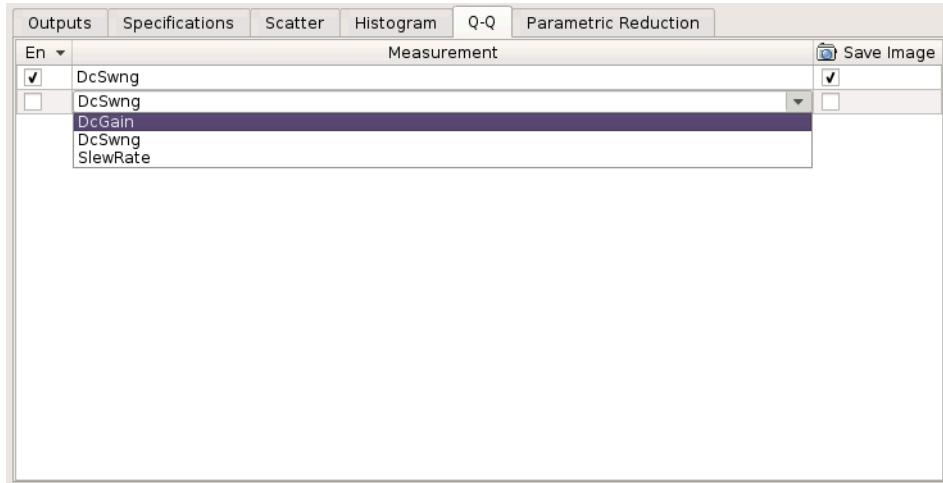
You can also set up histograms from the ResultsView using the **Plot Scalars** dialog box. See [Plotting Results From the ResultsView](#).

Setting Up Q-Q Plots

Q-Q plots are only generated if a Monte Carlo analysis was run, and they show if the measured results are Gaussian.

To set up Q-Q plots to be output to the results:

1. Click the **Q-Q** tab on the PrimeWave Design Environment main window.



2. Click **Click to add** in the **Measurement** column and set up the input measurement to plot.
3. Click **En** to enable the chart output.
4. To delete a chart output, right-click it and select **Delete**.
5. Click **Save Image** to output an image of the chart for viewing in the ResultsView.

Note:

You can also set up Q-Q plots from the ResultsView using the **Plot Scalars** dialog box. See [Plotting Results From the ResultsView](#).

Setting Up Parametric Reduction

The **Parametric Reduction** tab provides a global method to do parametric reduction on the output tabs. For example, if you ran 16 corners, you might only want to plot your outputs for certain corners. You can do this from this tab with one global setting.

You can also set the initial value of parametric reduction for charts, histograms, and Q-Q Plots. You can later edit these settings in the Charts window, but the initial settings are defined in the **Parametric Reduction** tab.

Note:

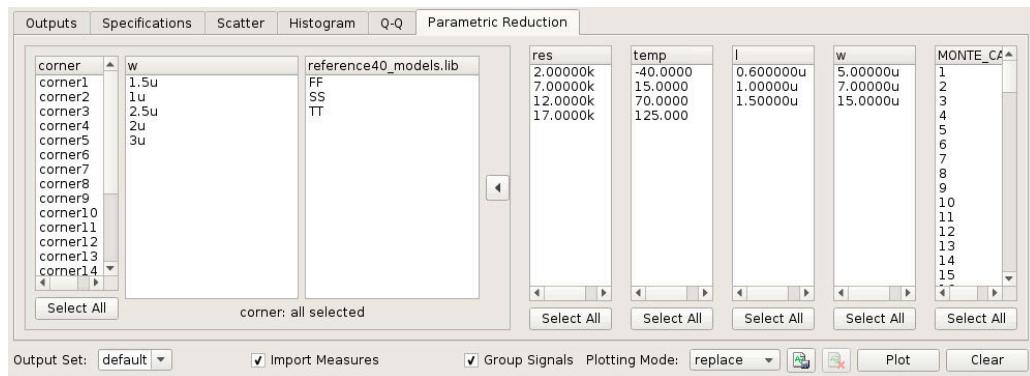
If you do not have any parametric reduction set, this means all parameters are taken into account (as opposed to none). In other words, no parametric reduction is equivalent to no filtering.

You can specify corner, parameter, and Monte Carlo iterations. To set up parametric reduction to be output to the results:

1. Click the **Parametric Reduction** tab on the PrimeWave Design Environment main window.

Note:

You can use the arrow in the middle of the tab to show and hide various parameters in the tab.



2. Select a corner or click **Select All** to select all corners.
3. Select other input variables and, if available, Monte Carlo iterations.

Adding External Images to Output

You can add external images (of file type .bmp, .gif, .jpeg, .jpg, .png, or .tiff) to simulation results for later viewing in the ResultsView. If you choose to export results to an HTML file, these external images can be included in the HTML report.

You can specify individual external image files and set their titles, or specify a folder with image files. File formats can be different. All image files are added to the **Misc. Images** section of the ResultsView.

See [Specifying External Images](#) for instructions.

Specifying Output Logic Radix

To specify the logic radix for an output, right-click the name of an output in the Outputs table, and choose **Logic Radix > <option>** from the menu that opens.

The following options are available:

- **Ascii**
 - **Binary**
 - **Decimal**
 - **Default**
 - **Hexadecimal**
 - **Octal**
 - **Signed**
-

Specifying Output Plot Type

To specify what type of plotted signal you want for an output, right-click the name of an output and choose **Plot Type > <option>** from the menu that opens.

The following options are available:

- **Auto**
Automatically determines the type of plot (default)
 - **Eyediag**
 - **Histogram**
 - **Linear**
 - **Polar**
 - **Scatter**
 - **Smith**
 - **Spectrum**
 - **Stateye**
-

Adding Outputs from Designs

To add an output from a design, choose **Outputs > Add from Design** from the PrimeWave Design Environment menu bar. The Schematic Editor opens. Outputs you choose from your design are added to the Outputs table in the PrimeWave Design Environment window.

Adding Outputs from Designs without Automatically Plotting Waveforms

To add an output from a design without automatically plotting waveforms after simulation, choose **Outputs > Add from Design (not plot)** from the PrimeWave Design Environment menu bar. The Schematic Editor opens. Outputs you choose from your design are added to the Outputs table in the PrimeWave Design Environment window, but the **Plot Waveforms/Evaluate** option is not checked.

Saving, Loading, and Editing Outputs as Text

You can save outputs to an ASCII file, which you can edit and reload into a future PrimeWave Design Environment session.

- To edit an output file in an external text editor, choose **Outputs > Text > Edit in Text Editor**.
 - To edit all MTB testbench expressions at once, choose **Outputs > Text > Edit All Testbench Outputs**. All the outputs for all testbenches are shown in external text editor for editing.

- To load an output file, choose **Outputs > Text > Load** from the main menu bar. Browse to an output file and click **Open**. Loaded outputs are appended to existing entries in your setup.
 - To save an output file, choose **Outputs > Text > Save** from the main menu bar. Enter a name for the output file and click **Save**.

Reordering Outputs

To reorder the outputs in the Outputs table, right-click an output and choose **Move Up** or **Move Down** from the menu that opens.

Reordering Output Columns

To reorder output columns, click a column header, then drag the column to the desired location.

Controlling Output Column Visibility

To control the visibility of each column in the outputs, right-click a column and choose an output column from the list that opens. Columns without check marks are not visible.

Editing Outputs in a Text Editor

To edit an output in a text editor, right-click an output in the Outputs table of the PrimeWave Design Environment tab page and choose **Edit in Text Editor** from the menu that opens.

The header lines display the syntax of the file to help show the required syntax. All changes you make when you exit the editor (additions, deletions, or modifications) are made directly in the output panel.

See [Specifying a File Text Editor](#) for information on setting up your text editor.

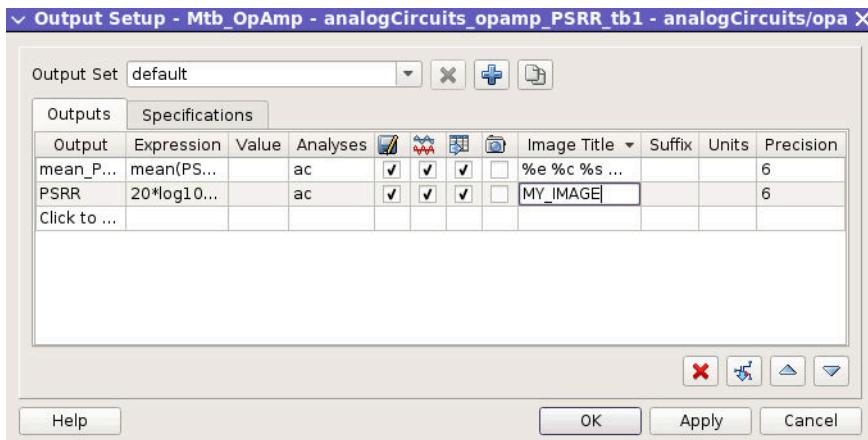
Editing Output Image Titles and Captions

Output images are automatically titled according to the following naming conventions:

- %c (corner name) (for example, FF, SS, TT)
- %e (expr/signal name) (for example, Vout or v(/out))
- %m (Monte Carlo iteration) (for example, 1, 2,17, 18, 1000)
- %s (sweep condition) (for example, param1=1.2,param2=0.7)
- %t (testbench name) (for example, Mytestbench, primesim_default, ...)

To edit image titles and captions:

1. Choose **Outputs > Edit** from the PrimeWave Design Environment main menu bar. The **Output Setup** dialog box opens.



2. Click an **Image Title** cell and type the new title of your choice. When typing the new title, the % codes listed above can be used. They can be interspersed with straight text and are substituted for at the appropriate time when the final titles are rendered.
3. Alternatively, right-click the selected image in ResultsView **Details** tab. Select **Update Title and Captioning**. The **Update Title & Caption** dialog box opens. Type the title and caption of your choice in the dialog box. Select **Apply title to all images in column** to update titles for all plots in a column.

Deleting Outputs

Outputs can be deleted directly from the PrimeWave Design Environment main window, or from in the **Output Setup** dialog box.

To delete outputs from the PrimeWave Design Environment main window, click a row that contains the output to be deleted, and click the **X** button in the vertical toolbar on the left side of the PrimeWave Design Environment main window.

To delete outputs from the **Output Setup** dialog box:

1. Choose **Outputs > Edit** from the PrimeWave Design Environment main menu bar, or press Shift+O.

The **Output Setup** dialog box opens.

2. Click the name of the output you want to delete.

If you do not see the output you want to delete, you might have the incorrect output set selected. Click the names of other output sets to locate the desired output.

3. Click the **X** button, which is just to the left of the **Select in Design** button.

The output disappears from the list of outputs.

Showing Outputs in Designs

To show an output in the schematic of your design, click an output in the Output table, and choose **Outputs > Show in Design** from the PrimeWave Design Environment tab page menu bar. The Schematic Editor opens and displays the selected output.

Adding Output Sets

To add a set of outputs:

1. Choose **Outputs > Edit** from the PrimeWave Design Environment main menu bar, or press Shift+O.

The **Output Setup** dialog box opens.

2. Click the blue + button to the right of the **Output Set** text box.

A new output set named `set<number>` appears in the Output Set text box as highlighted text.

3. (Optional) Click the `set<number>` text and rename the output set.
-

Copying Output Sets

To copy a set of outputs:

1. Choose **Outputs > Edit** from the PrimeWave Design Environment main menu bar.

The **Output Setup** dialog box opens.

2. Click the name of the output set you want to copy in the list of available output sets.

3. Click the **Copy Selected Output Set** button to the right of the blue plus button.

A new output set named `set<number>` appears in the list of output sets, which is an exact copy of the selected set.

4. (Optional) Change the name of the copied output set by clicking the `set<number>` text, and typing in a new output set name.

You can also copy an output set by entering a new name in the Output Set menu in the lower-left corner of the PrimeWave Design Environment main window.

Choosing Output Sets

You can choose an output set from the **Output Set** menu, which is located in the lower-left corner of the PrimeWave Design Environment main window.

Unless you create additional output sets, only the "default" output set is available. You can create additional output sets; see [Adding Output Sets](#).

Deleting Output Sets

To delete a set of outputs:

1. Choose **Outputs > Edit** from the PrimeWave Design Environment main menu bar.
The **Output Setup** dialog box opens.
2. Click the name of the output set you want to delete.

Note:

- The default output set cannot be deleted.
3. Click the **X** button just to the right of **Output Set** text box.
The output set disappears from the list.

Specifying ACE Scripts

Some advanced data processing and waveform viewer features are not directly available from the PrimeWave Design Environment (such as certain plot types or measurement calculations). To access these features, a wrapper utility exists that leverages the ACE scripting language in waveform viewer to read simulation data and export measurements and signals into the PrimeWave Design Environment outputs, which later can be used in the PrimeWave Design Environment ResultsView.

You can specify ACE scripts to be run in waveform viewer after the simulation is complete in the PrimeWave Design Environment. The results, including measurement values and screen captures, are captured and returned to the PrimeWave Design Environment for

use in postprocessing, including in the ResultsView. In order to return any results to the PrimeWave Design Environment, your ACE scripts need to be wrapped in custom Tcl helper functions provided by the PrimeWave Design Environment (see [PrimeWave Design Environment Provided ACE Utilities](#)).

To select ACE scripts to run after simulation is complete:

1. Choose **Outputs > ACE Scripts** from the PrimeWave Design Environment main menu bar.

The **Specify ACE Scripts** dialog box opens.



2. Click in the **Script Path** field and browse to the desired ACE script.
3. (Optional) Click **Use GUI** to run the script using the waveform viewer GUI (use this option if you want to save screen captures from waveform viewer to the PrimeWave Design Environment results database).
4. (Optional) **Click to Add** another ACE script.
5. Select or deselect **Enable** to choose the desired scripts. Delete unwanted ACE scripts using the **Delete** button. Enabled ACE scripts are run in descending order after simulation, each in a separate waveform viewer window.
6. Click **Apply and Evaluate** to run the ACE script(s) immediately on the existing simulation data.

Alternatively, click **OK** to save the script information and run it at the end of the simulation.

PrimeWave Design Environment Provided ACE Utilities

ACE is a powerful scripting language provided as part of waveform viewer ADV. To support ACE within the PrimeWave Design Environment, several ACE helper functions are provided with the PrimeWave Design Environment.

Utility	Description
<code>foreach_sim varName script</code>	<p>Run the script over multiple simulation results (ex, corners, sweeps). It can also be used for nominal results. Required user-defined arguments are:</p> <ul style="list-style-type: none"> • <i>varName</i>: The variable name to store the location of the results directory path for the simulation result currently being processed. • <i>script</i>: The Tcl script to run for each simulation results directory.
<code>get_analysis_filename analysis</code>	<p>Get simulator result filename given the analysis type (ex "tran", "ac"). This makes it possible to reuse the same ACE script for different simulators supported by the PrimeWave Design Environment. Required user-defined arguments are:</p> <ul style="list-style-type: none"> • <i>analysis</i>: The analysis name.
<code>get_var_value varName</code>	<p>Get the value of a variable that was defined by the PrimeWave Design Environment for the simulation. It allows variable values to be used in the calculations. Required user-defined arguments are:</p> <ul style="list-style-type: none"> • <i>varName</i>: The variable name.
<code>write_measures names values [filename] [resultType]</code>	<p>Write scalar measurement values calculated by ACE in the directory to a file that the PrimeWave Design Environment will import into the ResultsView database. Required user-defined arguments are:</p> <ul style="list-style-type: none"> • <i>names</i>: The list of measurement names. • <i>values</i>: The list of measurement values (must be the same length as the names). <p>Optional user-defined arguments are:</p> <ul style="list-style-type: none"> • <i>filename</i>: The filename to write measurement values (default: measures). • <i>resultType</i>: Allows you to specify the PrimeWave Design Environment <i>resultType</i> instead of <i>filename</i>.

Utility	Description
<code>write_signals signals [resultType] [analysisName] [time] [dataType]</code>	<p>Write signals calculated by ACE in the directory to a file that the PrimeWave Design Environment will import into the Output Table.</p> <p>Required user-defined arguments are:</p> <ul style="list-style-type: none"> • <i>signals</i>: The list of signals. <p>Optional user-defined arguments are:</p> <ul style="list-style-type: none"> • <i>resultType</i>: Use if there are multiple signals of the same name for different result types. • <i>analysisName</i>: Used in addition to result type if there are multiple analyses of the same result type. • <i>time</i>: Used to distinguish results of the same name for different time points. • <i>dataType</i>: Used to distinguish analog and digital results of the same name.

Sample ACE Script

The following sample ACE script creates an eye diagram and demonstrates utilities that:

- Pass scalar values back to the PrimeWave Design Environment for displaying in the ResultsView.
- Save images to be displayed in the ResultsView.
- Use the PrimeWave Design Environment design variables in ACE measurements.
- Generate ACE signals you can later plot in the PrimeWave Design Environment Outputs table.

PrimeWave Design Environment provided utilities are highlighted in purple.

```
# Utilities for compatibility with PrimeWave Design Environment results
# directories and database import

source $::env(SYNOPSYS_CUSTOM_INSTALL)/waveform/waveform viewerUtils.tcl

namespace eval saeUtils {

    foreach_sim resultsDir {
        # Open "tran" analysis file
        set filename [get_analysis_filename "tran"]
        sx_open_sim_file_read $filename

        # Create an eye diagram
        set signal [sx_signal "v(rx)"]
        set eye_obj [sx_create_eye $signal "ui=400p"]
        sx_display_eye $eye_obj
    }
}
```

```

# Take a screenshot
set activewv [sx_get_active_wv]
sx_dump_png $activewv ./eye.png

# Get zterm variable value
set zterm [get_var_value "zterm"]
puts "zterm=$zterm"

# Perform measurements on the eye
sx_measure_eye $eye_obj "type=opening" "vref=0.0"\n
    "vdc=($zterm*1e-3)" "vac=(2*$zterm*1e-3)"
set names {wh wl mh ml oh ol fuss aper dvdt}
set values {}
foreach name $names {
    lappend values [sx_query_eye $eye_obj $name]
}

# Write measurements to a file on disk that the PrimeWave Design
# Environment can read
write_measures $names $values

# Write signals to a file on disk that the PrimeWave Design
# Environment can read and load into the Output Table
set sig1 [sx_equation sig1='0|v(rx)**2']
set sig2 [sx_equation sig2='0|v(rx)**3]
set sig3 [sx_equation sig3='0|v(rx)**5]
set signals [list $sig1 $sig2 $sig3]
write_signals $signals

# Clear old plots for next run
} sx_clear_wv
}

exit

```

Specifying MATLAB Scripts

You can specify MATLAB scripts for postprocessing PrimeWave Design Environment results in a waveform viewer.

Note:

For more information on MATLAB, see the [MATLAB documentation](#) from MathWorks®.

Before you start, review the information in [Setting Up the MATLAB Interface](#) in the *waveform viewer™ User Guide*.

MATLAB scripts can be placed anywhere, but the full absolute path to the script must be given as `scriptPath` argument. No default directory is assumed. For example:

```
matlabScalarScript(scriptPath="/path/to/matlab/script.m")
```

Shell environment variables can be embedded in the path, for example:

```
scriptPath="$USER/path/to/$PROJECT/script.m"
```

`matlabScalarScript` scalar measurements are treated just like regular PrimeWave Design Environment measurements and calculator functions. If a measurement fails, the PrimeWave Design Environment typically tries to evaluate it one more time (standard PrimeWave Design Environment behavior).

The MATLAB script runs at the end of each individual (leaf-level) simulation. If sweeping a variable, the script runs at the end of each sweep iteration. If running corner analysis, the script runs at the end of each corner. If running Monte Carlo analysis, the script runs at the end of each Monte Carlo simulation run (after the final iteration). MATLAB scripts can open a graph/plot window.

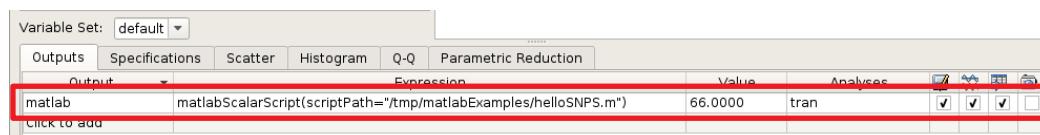
To specify MATLAB scripts:

1. Make sure MATLAB and a recent version of waveform viewer are included in your path.
2. Add an output expression to the **Outputs** table in the form

```
matlabScalarScript(scriptPath="/path/to/matlab/script.m")
```

The example below shows an output named `matlab` with the expression

```
matlabScalarScript(scriptPath="/tmp/matlabExamples/helloSNPS.m").
```



Output	Expression	Value	Analyses
matlab	matlabScalarScript(scriptPath="/tmp/matlabExamples/helloSNPS.m")	66.0000	tran
Click to add			

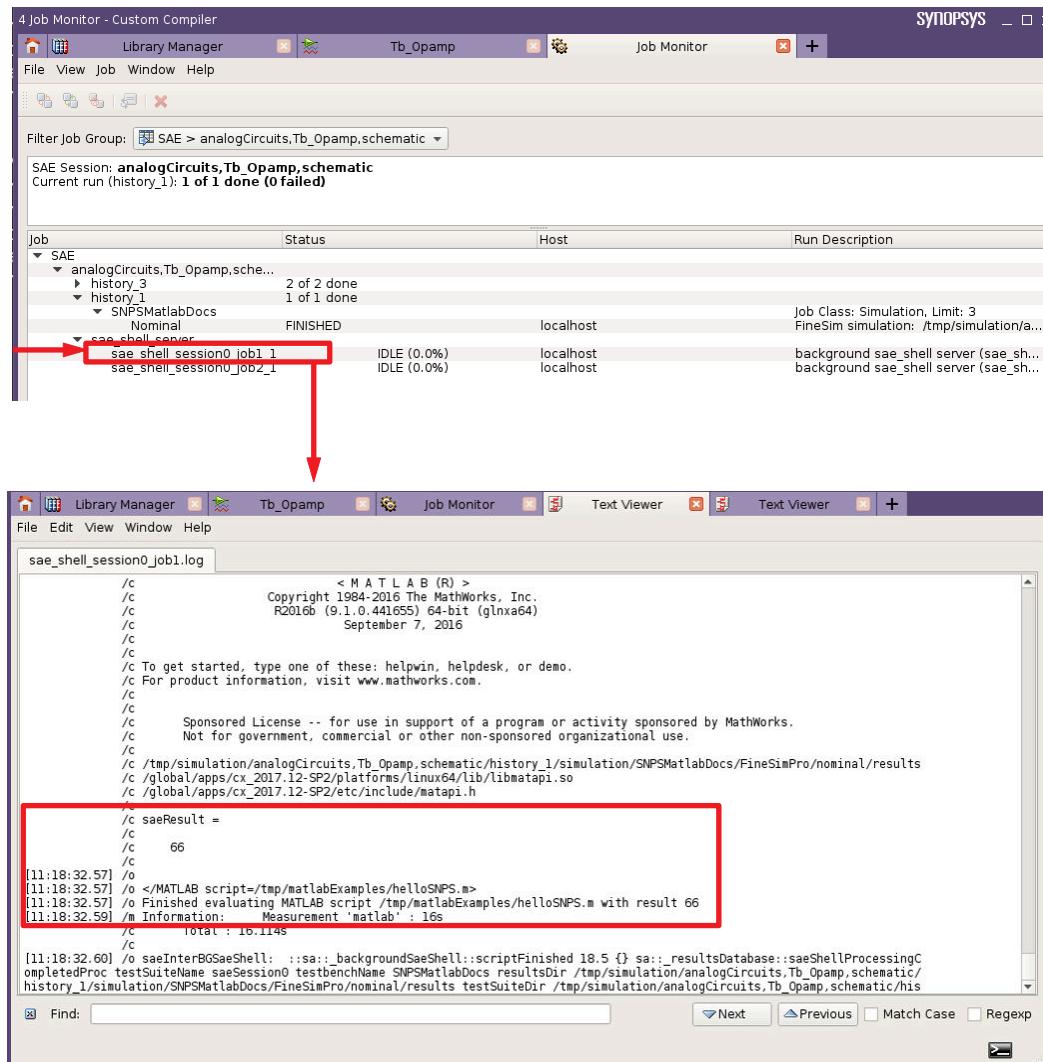
For details on the contents of a MATLAB script, see [Sample MATLAB Script](#).

3. Choose **Simulation > Netlist and Run** to run the simulation.

Caution:

MATLAB scripts can be slow due to MATLAB license checkouts, and so forth. To optimize performance, run MATLAB scripts in the background to keep the PrimeWave Design Environment responsive. In the **Simulator** setup dialog box, in the **Measurement Jobs** section, select the **Background Mode** option. When debugging script and environment setup issues, you might choose to change to **Foreground Mode**.

- In the Job Monitor, double-click the PrimeWave Design Environment shell session to open the log file in the Text Viewer. If the MATLAB script is successful, the log file indicates success and the value of the `matlab` output expression is added to the **Outputs** table in the main PrimeWave Design Environment window. If the log file indicates an error, see [Debugging MATLAB Script Issues](#).



- Review the results directory and notice that the following files are created in each leaf-level PrimeWave Design Environment results directory:

- `matlab.log.<scriptName>.m`

Contains stdout from MATLAB process

This is the main output that is parsed looking for `saeResult = <value>`

- *matlab.stderr.<scriptName>.m*

Contains stderr from MATLAB process

```
$ pwd
/tmp/simulation/analogCircuits.Tb_Opamp,schematic/history_1/simulation/SNPSMatla
bDocs/FineSimPro/nominal/results
$ ls -l
charts.xml
designVariables.wdf
finesim.elog
finesim.ic
finesim.log
finesim.wdf
matlab.log.helloSNPS.m
matlab.stderr.helloSNPS.m
resultsMap.xml
results.xml
run.log
$
```

Sample MATLAB Script

The MATLAB script should:

- Reference the MATLAB variable `saeCurrentResultsPath`
 This contains the path to the results directory for a leaf-level simulation.
- Return a scalar value via the assignment `saeResult = <value>` (the PrimeWave Design Environment monitors the MATLAB session transcript looking for this variable to be written)

The following sample script provides a full round trip with the PrimeWave Design Environment. It displays some variables set by the PrimeWave Design Environment, computes the results, and returns a scalar value.

```
% load libmatapi library

addpath (pathToWVReaderSharedObject)
loadlibrary('libmatapi', pathToWVReaderHeader)

% open file

pfile = calllib('libmatapi','wv_open_file',filepath);

% read some data from testcase.ac0 in current results path

filepath = sprintf('%s/testcase.ac0', saeCurrentResultsPath);
pfile = calllib('libmatapi','wv_open_file',filepath);

% verify the MATLAB variable saeCurrentResultsPath
% point at a PrimeWave Design Environment results directory

disp(saeCurrentResultsPath);
```

```
% verify that the PrimeWave Design Environment has set some variables
% containing the paths to the waveform viewer database Reader API shared
% object

disp(pathToWVReaderSharedObject);
disp(pathToWVReaderHeader);

% send a scalar result back to the PrimeWave Design Environment

saeResult = 99.9
```

Debugging MATLAB Script Issues

MATLAB script issues might include a script trying to load a file that does not exist or problems with your environment setup. The PrimeWave Design Environment detects any such problems and stop waiting for the MATLAB script, reporting the problems to the log file.

Use the following environment variables to help debug scripts and environment setup issues.

Environment Variable	Description
SAE_MATLAB_DEBUG	<p>Verbosely logs all calls to and all outputs from a MATLAB matlabScalarScript.</p> <p>Location for the logged information depends on the PrimeWave Design Environment measurement evaluation mode:</p> <ul style="list-style-type: none"> • Foreground: goes to Custom Compiler console and Custom Compiler log file • Background: goes to the PrimeWave Design Environment shell log file only
SAE_MATLAB_WAIT_INTERVAL	<p>Instructs the PrimeWave Design Environment to wait no longer than this interval(s) for slow/non-responsive MATLAB scripts.</p> <p>Must be a positive integer between 0 and 3600. By default, the PrimeWave Design Environment will wait up to 3600 seconds (1 hour) before giving up on a slow MATLAB script.</p> <p>Set to a low number (30-60 seconds) when debugging.</p>

When running measurement evaluations in **Background Mode**, the PrimeWave Design Environment writes errors in MATLAB script handling to the PrimeWave Design Environment shell log file, as shown in the following figure.

Chapter 12: Specifying Outputs and Output Options

Specifying MATLAB Scripts

```

sae_shell_session0_job1.log
[11:21:04.30] /m Information: Saving the images completed for testbench:SNPSMatlabDocs in Os
[11:21:04.30] /m Information: Generating the thumbnails completed for testbench:SNPSMatlabDocs in Os
[11:21:04.31] /m Information: Finished reading Simulation Check Results for testbench SNPSMatlabDocs
[11:21:04.32] /m Information: Creating the setup tables completed for testbench:SNPSMatlabDocs in Os
[11:21:04.32] /m Information: Start incremental database creation...
[11:21:04.34] /i Invoking script /tmp/matlabExamples/helloSNPSNotGood.m in resultsDir /tmp/simulation/analogCircuits.Tb_Opamp.sch
ematic/history_1/simulation/SNPSMatlabDocs/FineSimPro/nominal/results
[11:21:04.34] /o Begin evaluating MATLAB script /tmp/matlabExamples/helloSNPSNotGood.m
[11:22:06.52] /m Information: Giving up on MATLAB process MATLAB 6 (oa:0x192edcc0) after a total wait time of 62 s
[11:22:06.52] /o <MATLAB script=/tmp/matlabExamples/helloSNPSNotGood.m>
[11:22:06.52] /c
[11:22:06.52] /c < M A T L A B (R) >
[11:22:06.52] /c Copyright 1984-2016 The MathWorks, Inc.
[11:22:06.52] /c R2016b (9.1.0.441655) 64-bit (glnxa64)
[11:22:06.52] /c September 7, 2016
[11:22:06.52] /c
[11:22:06.52] /c To get started, type one of these: helpwin, helpdesk, or demo.
[11:22:06.52] /c For product information, visit www.mathworks.com.
[11:22:06.52] /c
[11:22:06.52] /c Sponsored License -- for use in support of a program or activity sponsored by MathWorks.
[11:22:06.52] /c Not for government, commercial or other non-sponsored organizational use.
[11:22:06.52] /c
[11:22:06.52] /c /tmp/simulation/analogCircuits.Tb_Opamp.schematic/history_1/simulation/SNPSMatlabDocs/FineSimPro/nominal/results
[11:22:06.52] /c
[11:22:06.52] /c foo =
[11:22:06.52] /c
[11:22:06.52] /c 99
[11:22:06.52] /c
[11:22:06.52] /c
[11:22:06.52] /o </MATLAB script=/tmp/matlabExamples/helloSNPSNotGood.m>
[11:22:06.52] /e Error: Sentinel value saeResult not found, or error encountered in script /tmp/matlabExamples/helloSNPSNotGood.m
[11:22:06.52] /o output
[11:22:06.52] /o Finished evaluating MATLAB script /tmp/matlabExamples/helloSNPSNotGood.m with result <ERROR>

```

To debug errors in MATLAB script handling while in **Background Mode**:

1. Set the environment variable `SAE_MATLAB_DEBUG`:

```
setenv SAE_MATLAB_DEBUG 1
```

2. Invoke Custom Compiler and start the PrimeWave Design Environment.
3. Evaluate measurements in the **Outputs** table by right-clicking in the table and choosing **Evaluate Measurements**.
4. After running the simulation, examine the MATLAB log files in the results directory.
5. In the PrimeWave Design Environment window, choose **Tools > Show Simulation Jobs**. The Job Monitor opens.
6. In the Job Monitor, double-click the PrimeWave Design Environment shell job (usually the first one in the list) and examine the transcript/debug output in the file that opens in the Text Editor. Alternatively, you can find the `pw_shell` log file using a terminal window.

When running measurement evaluations in **Foreground Mode**, the PrimeWave Design Environment writes errors to the Console and the Custom Compiler log file, in addition to the PrimeWave Design Environment shell log file.

To debug errors in MATLAB script handling while in **Foreground Mode**:

1. Set the environment variable `sae_MATLAB_DEBUG`:

```
setenv SAE_MATLAB_DEBUG 1
```

2. Invoke Custom Compiler and start the PrimeWave Design Environment.
3. In the **Simulator** setup dialog box, in the **Measurement Jobs** section, select the **Foreground Mode** option.
4. Evaluate measurements in the **Outputs** table by right-clicking in the table and choosing **Evaluate Measurements**.
5. After running the simulation, examine the transcript and debug output in the `Console`. Alternatively, examine the output in Custom Compiler's primary log file.

Specifying Output Options

You can specify output options for the following integrated simulators:

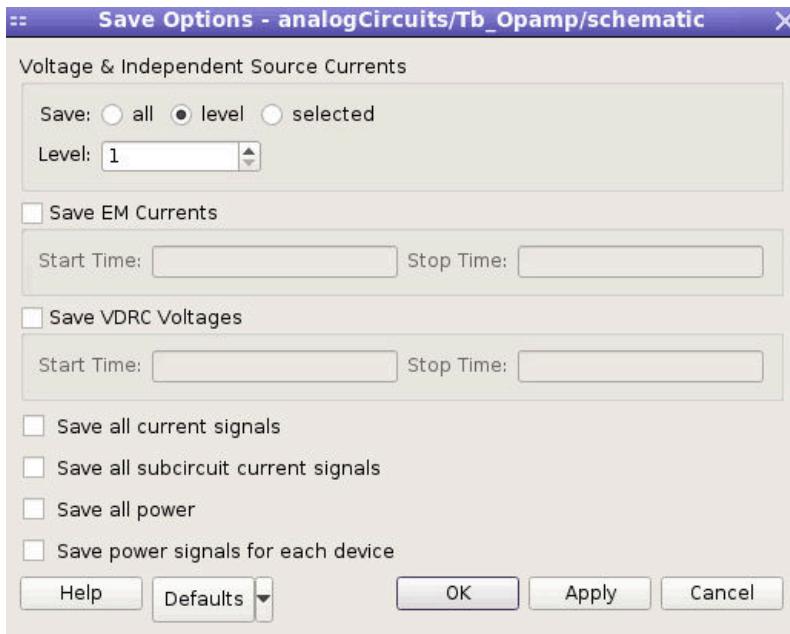
- [PrimeSim HSPICE Output Options](#)
- [FineSim Output Options](#)
- [PrimeSim XA Output Options](#)
- [FineSim VCS Output Options](#)
- [VCS PrimeSim AMS Output Options](#)

PrimeSim HSPICE Output Options

To specify output options for the PrimeSim HSPICE integration:

1. Choose **Outputs > Save Options** from the PrimeWave Design Environment main menu bar.

The **Save Options** dialog box opens.



2. Select which signals you want to output:

- **all**

This saves all node voltages, as well as currents through independent sources.

- **level**

This controls the design hierarchy levels for the saved outputs you specify. If you select the **level** option, enter a value for **Level**.

In addition to saving all voltages for X levels, this option also saves all selected signals from the outputs.

- **selected**

This limits the data written to your results file to the voltages and currents that are in the **Outputs** list.

3. (Optional) Choose whether or not to **Save EM Currents**. For more information, see *Measuring Terminal Current Data Using the PrimeWave Design Environment* in the *Custom Compiler Electrical Checking and Reporting User Guide*.
4. (Optional) Choose whether or not to **Save VDRC Voltages**. For more information, see *Measuring Voltage Data Using the PrimeWave Design Environment* in the *Custom Compiler Electrical Checking and Reporting User Guide*.

5. (Optional) Select one or more of the following options:

- **Save all current signals**
- **Save all subcircuit current signals**
- **Save all power**
- **Save power signals for each device**

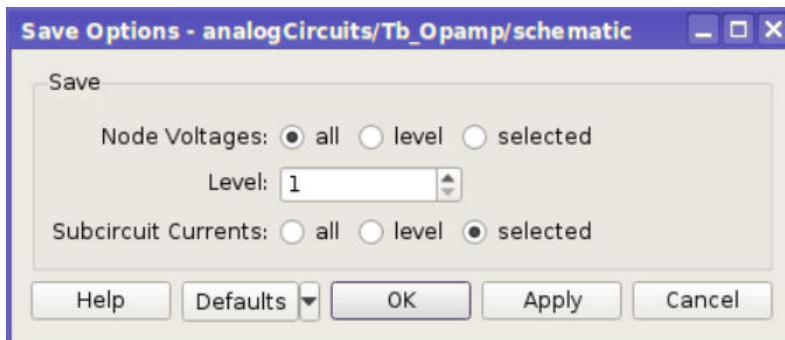
6. Once all output options are set, click **OK**.

FineSim Output Options

To specify output options for the FineSim integration:

1. Choose **Outputs > Save Options** from the PrimeWave Design Environment main menu bar.

The **Save Options** dialog box opens.



2. Choose which **Node Voltages** and **Subcircuit Currents** you want to include.

If you choose **level**, choose a level value to save all node voltages or subcircuit currents up to the specified level.

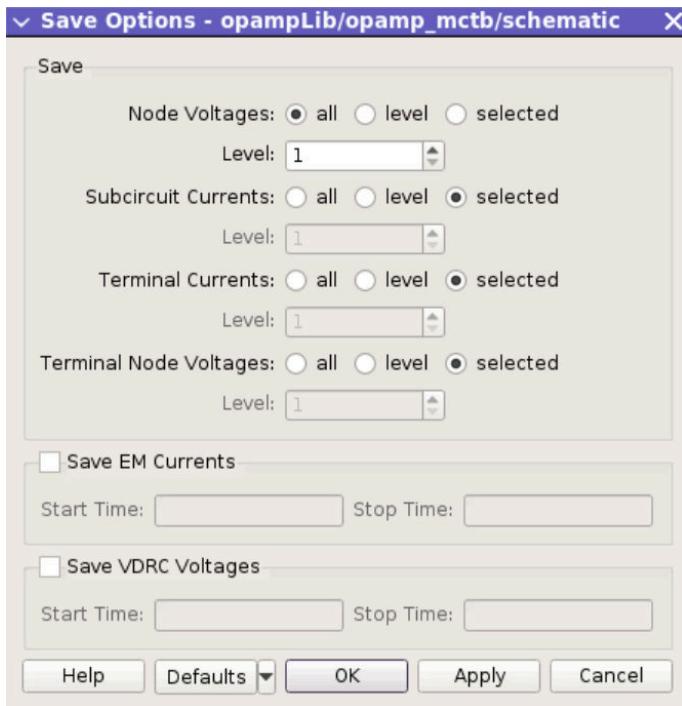
3. Once all output options are set, click **OK**.

PrimeSim XA Output Options

To specify output options for the PrimeSim XA integration:

1. Choose **Outputs > Save Options** from the main menu bar.

The **Save Options** dialog box opens.



2. Choose which **Node Voltages**, **Subcircuit Currents**, **Terminal Currents**, and **Terminal Node Voltages** you want to include.

If you choose **Level** for the **Node Voltages**, **Subcircuit Currents**, **Terminal Currents**, or **Terminal Node Voltages**, choose a level value to save all voltages or currents up to the specified level.

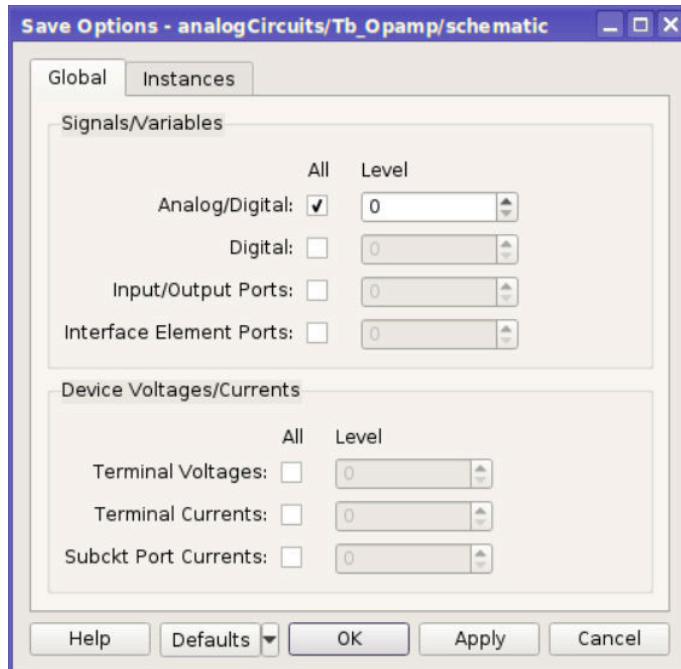
3. (Optional) Choose whether or not to **Save EM Currents**. For more information, see Measuring Terminal Current Data Using the PrimeWave Design Environment in the *Custom Compiler Electrical Checking and Reporting User Guide*.
4. (Optional) Choose whether or not to **Save VDRC Voltages**. For more information, see Measuring Voltage Data Using the PrimeWave Design Environment in the *Custom Compiler Electrical Checking and Reporting User Guide*.
5. Click **OK** to save your changes.

FineSim VCS Output Options

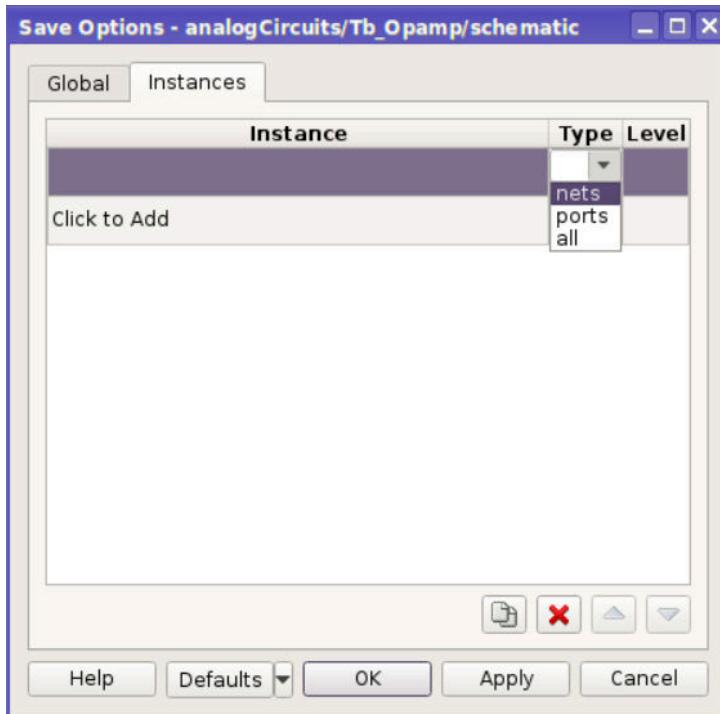
To specify output options for the FineSim VCS integration:

1. Choose **Outputs > Save Options** from the PrimeWave Design Environment main menu bar.

The **Save Options** dialog box opens with the **Global** tab forward.



2. In the **Signals/Variables** section, choose which signals and variables you want to include and at which **Level** value.
3. In the **Device Voltages/Currents** section, choose which voltages and currents you want to include and at which **Level** value.
4. Click the **Instances** tab to display the instances output options.



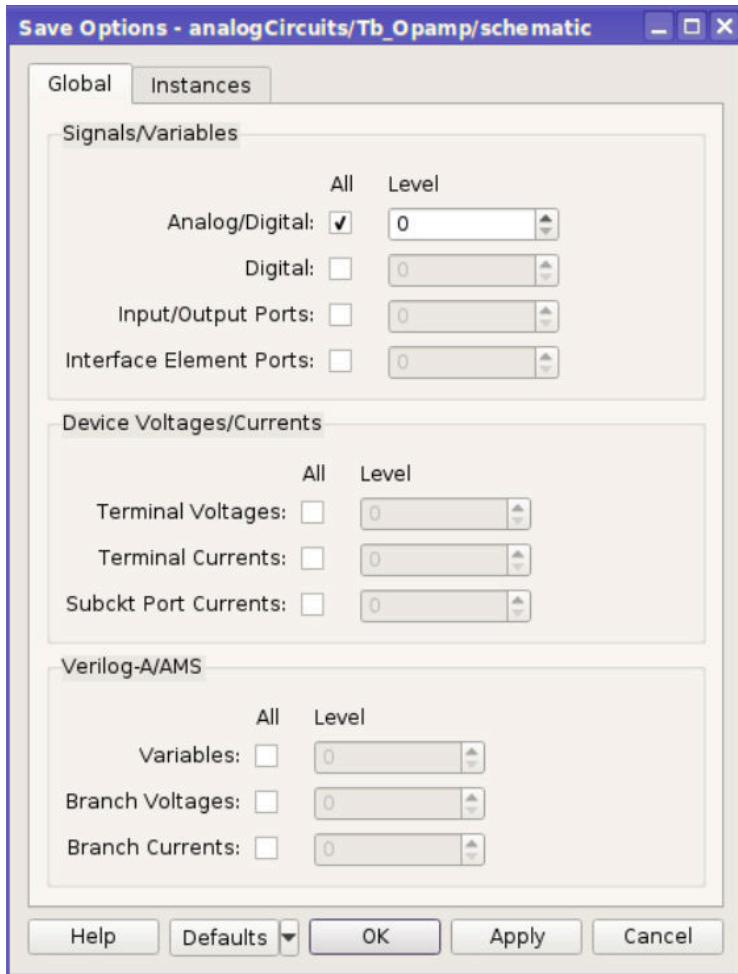
5. Click the **Click to Add** text, and enter the name of an instance or click the **Select Instance** button to select an instance from a schematic.
6. In the **Type** column, choose **nets**, **ports**, or **all** to save the net voltages, port currents, or both, respectively.
A save statement is added to the `simulate.cmd` mixed-signal command file with your choice.
7. In the **Level** column, enter the level for the instance.
8. Once all output options are set, click **OK**.

VCS PrimeSim AMS Output Options

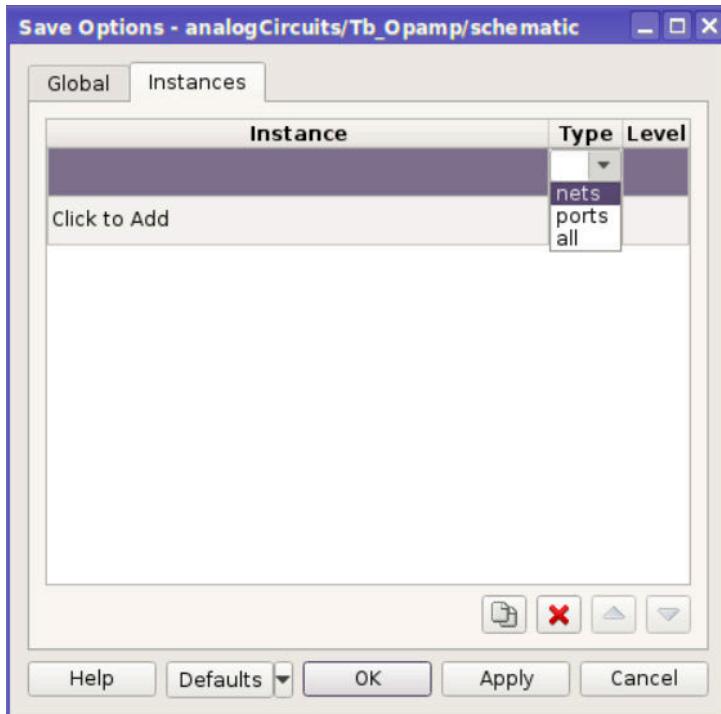
To specify output options for the VCS PrimeSim AMS integration:

1. Choose **Outputs > Save Options** from the PrimeWave Design Environment main menu bar.

The **Save Options** dialog box opens with the **Global** tab forward.



2. In the **Signals/Variables** section, choose which signals and variables you want to include and at which **Level** value.
3. In the **Device Voltages/Currents** section, choose which voltages and currents you want to include and at which **Level** value.
4. In the **Verilog-A/AMS** section, choose Verilog-A/AMS variables, branch voltages, and branch currents you want to include and at which **Level** value.
5. Click the **Instances** tab to display the instances output options.



6. Click the **Click to Add** text, and enter the name of an instance or click the **Select Instance** button to select an instance from a schematic.
7. In the **Type** column, choose **nets**, **ports**, or **all** to save the net voltages, port currents, or both, respectively.
A save statement is added to the `simulate.cmd` mixed-signal command file with your choice.
8. In the **Level** column, enter the level for the instance.
9. Once all output options are set, click **OK**.

13

Creating Netlists and Running Simulations

This chapter contains information on how to create netlists and run simulations in the PrimeWave Design Environment.

This chapter contains the following topics:

- [Creating Netlists](#)
- [Viewing and Editing Netlists](#)
- [Starting and Stopping Simulations](#)
- [Running Simulations in Append Mode](#)
- [Setting VCS PrimeSim AMS Run Mode](#)
- [Setting Mixed-Signal Simulator Options](#)
- [Setting Block-Level Options](#)
- [Displaying the Simulation Log](#)
- [Saving Simulation History](#)
- [Displaying Simulation Jobs](#)

Creating Netlists

In the Primewave Design Environment, you can create just a netlist, or you can combine netlisting with running a simulation.

The following sections are available:

- [Netlisting a Design](#)
- [Netlisting and Running Simulations](#)

Note:

Any .MEAS statements you have in your testbench are netlisted.

Netlisting a Design

If you want to create a netlist without immediately running a simulation, choose **Simulation > Netlist > Create**, or press the N key. You are prompted to run a check and save on any designs in your session that have unsaved changes.

Note:

When multiple testbenches are present, only those that are enabled (the check box is selected next to the name of the testbench) are netlisted when you choose **Simulation > Netlist > Create**. All testbenches are enabled by default.

Netlisting and Running Simulations

To recreate the structural netlist and run a simulation all in one action, choose **Simulation > Netlist and Run** from the PrimeWave Design Environment main menu bar, click **Netlist and Run** in the toolbar, or press Ctrl+S. This applies only to a single testbench.

Note:

When multiple testbenches are present, only those that are enabled (the check box is selected next to the name of the testbench) are netlisted and run when you choose **Simulation > Netlist and Run**. All testbenches are enabled by default.

If you want to netlist and run all testbenches, click the multiple testbench netlist and run button, which is above the **Testbenches** tab.

If you have unsaved changes in any of the designs in your Custom Compiler session, you are prompted to run a check and save on these designs before netlisting. All designs within a session are considered, because the hierarchy in which you are working might contain other designs.

If changes have been made to the final netlist, the **Overwrite final netlist** dialog box appears, prompting you to select the testbenches to overwrite and keep.



When you combine netlisting with running a simulation, a structural netlist is created, which includes just the circuit topology. This netlist is located under the name **NETLIST** in the following directory:

```
<top_level_results_dir>/<Library_Name>/<Cell_Name>/<View_Name>/
<simulator_name>/nominal/netlist
```

The netlist that is sent to the simulator, referred to as the final netlist, is located under the name **INPUT.SPI** in the following directory:

```
<top_level_results_dir>/<Library_Name>/<Cell_Name>/<View_Name>/
<simulator_name>/nominal/netlist
```

Note:

These file names reflect the PrimeSim HSPICE integration; other integrations might use alternative file names for netlists.

Viewing and Editing Netlists

Choose **Simulation > Netlist > Display** from the PrimeWave Design Environment main menu bar to view the final netlist that is currently stored on disk.

You can edit the final netlist by choosing **Simulation > Netlist > Edit**. You can also choose **Simulation > Terminal**, which launches an xterm window in the following directory:

```
<top_level_results_dir>/<Library_Name>/<Cell_Name>/<View_Name>/
<simulator_name>/nominal/netlist
```

You can enter Linux commands to view or edit existing netlists from within this directory.

Starting and Stopping Simulations

To start a simulation using the structural netlist that is already stored on disk, choose **Simulation > Run** from the PrimeWave Design Environment main menu bar, or click the **Run** button in the toolbar located on the right side of the PrimeWave Design Environment main window. This applies only to a single testbench.

Note:

When multiple testbenches are present, only those that are enabled (the check box is selected next to the name of the testbench) are run when you choose **Simulation > Run**. This also applies when you choose **Simulation > Stop**. All testbenches are enabled by default.

You can use the multiple testbench start and stop buttons to start and stop multiple testbench simulations, which are just above the Testbenches tab.

All changes you make to your design since the last saved version are ignored. The final netlist is recreated whether or not any changes occur in the testbench.

To stop a simulation that is in progress, choose **Simulation > Stop** from the PrimeWave Design Environment main menu bar, or click the **Stop** button in the button toolbar, which is located in the right side of the PrimeWave Design Environment main window.

Running Simulations in Append Mode

You can run incremental simulations using Append mode. Append mode allows you to have an early look at simulation results before committing to more, or, in the case of Monte Carlo analyses, to determine whether more runs are required. Append mode lets you run just those simulations that would be appended to the rest of the results without rerunning earlier iterations.

Append mode is valid for all simulator integrations and in both single- and multiple-testbench (MTB) modes. Append mode supports corners, parametric sweep, and Monte Carlo simulations.

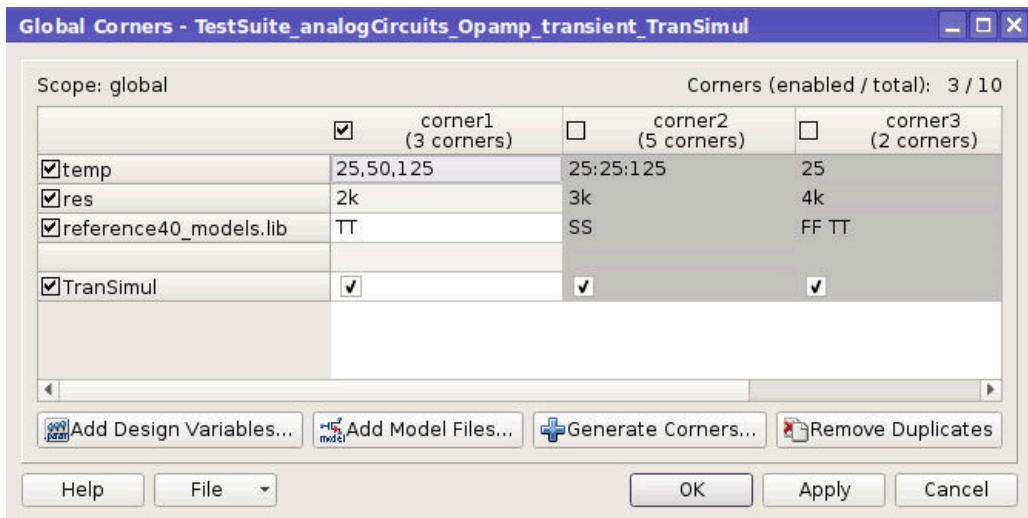
To run simulations in Append mode:

1. Run a simulation as usual to establish an initial set of results and baseline history point, but do not include all possible iterations. A baseline history point is needed for all the simulation types you plan to add later.

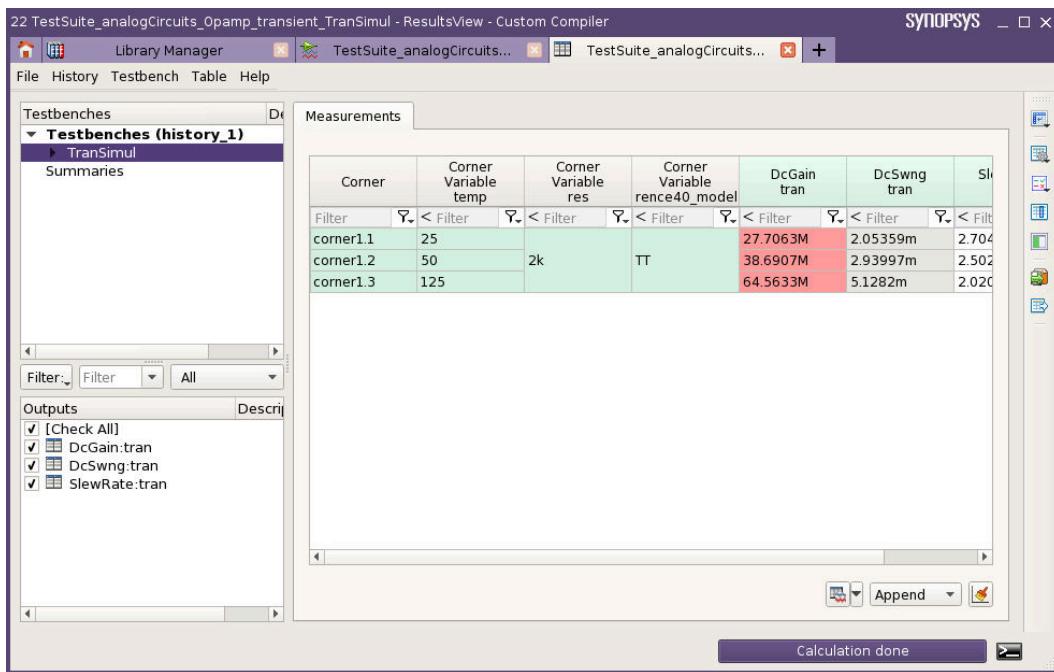
For example, if you are performing a global corner analysis, this first run could enable only corner1, not all corners in the testbench (shown in the following figure). (See [Setting Up Corner Analyses](#) for information on corner analyses.)

Chapter 13: Creating Netlists and Running Simulations

Running Simulations in Append Mode

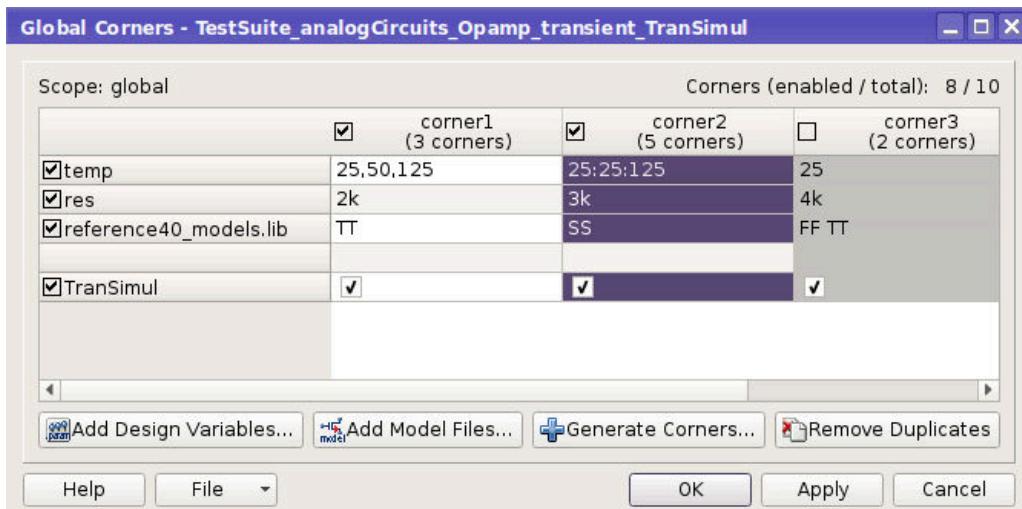


2. In the ResultsView window, notice the results of your first simulation.



3. Add more simulations to the run.

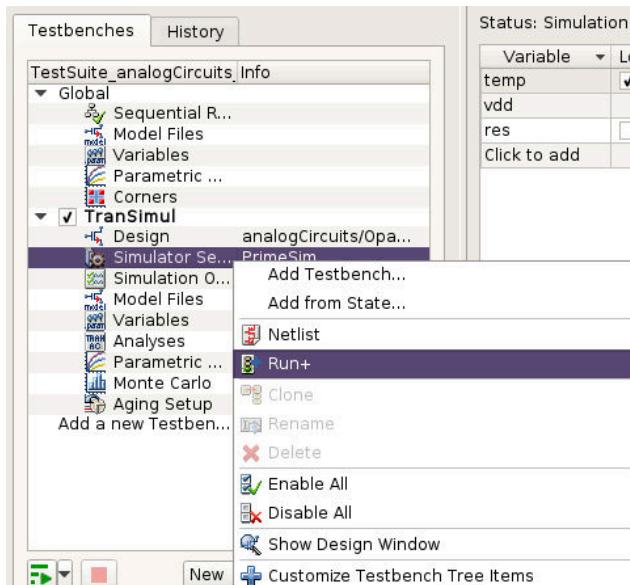
For example, if you are performing a global corner analysis, add corner2 to the testbench (shown in the following figure).



- In the main PrimeWave Design Environment window, choose the **Run+** button  to run an incremental simulation using Append mode. Alternatively, choose **Run+** from the right-click menu for the testbench.

Note:

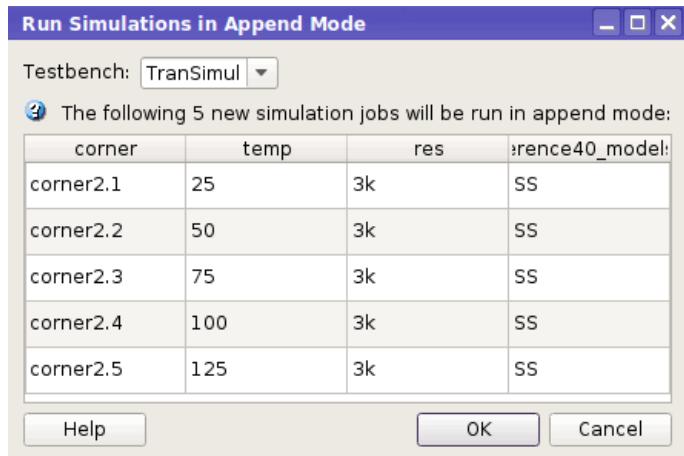
Some of the features shown below are limited availability. For information about these features, refer to SolvNetPlus article #000036534 "[How to Enable the PrimeWave Design Environment Flow-Based Interface](#)" or consult your Synopsys representative.



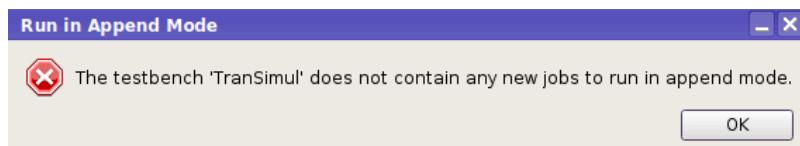
Note:

Append mode always runs from the current history point.

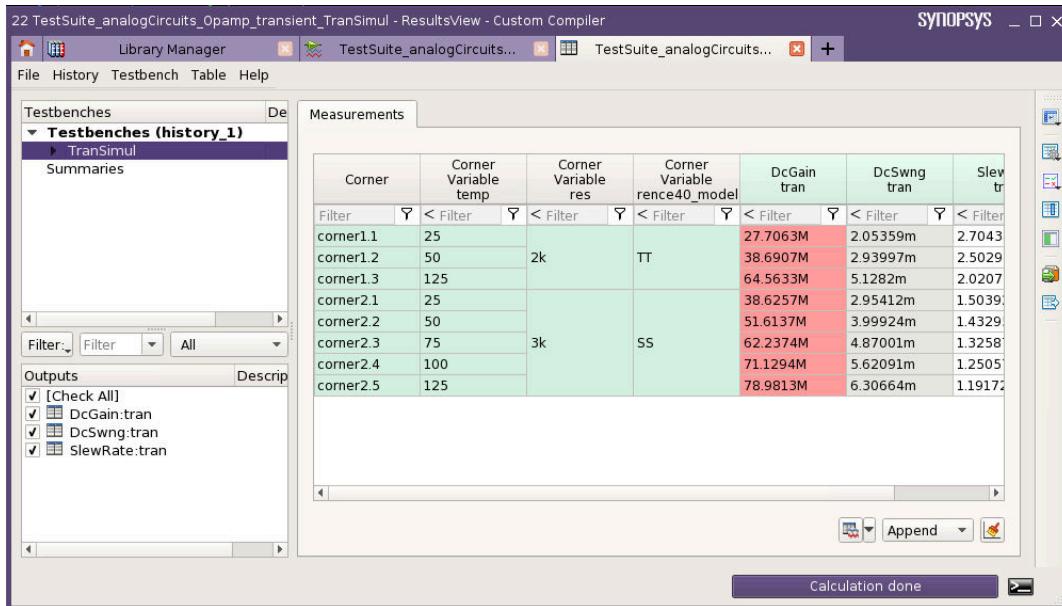
5. The **Run Simulations in Append Mode** dialog box appears, indicating which new simulation jobs will be run in append mode. Click **OK** to run the simulation.



If no new simulations have been added, an error message appears.



6. In the ResultsView window, notice the results of your incremental simulation.



7. Repeat [Step 4](#) through [Step 6](#) to run in append mode as many times as necessary.

Setting VCS PrimeSim AMS Run Mode

To set the run mode for both the VCS PrimeWave AMS integration and the FineSim VCS integration, choose **Simulation > Mixed Signal Options** from the PrimeWave Design Environment menu bar. You can see the **Batch Run Mode** and the **GUI Run Mode** options on the **VCS Options** tab of the **Mixed Signal Options** dialog box.

You can see the following run modes on the **Simulation** menu:

- [Batch Run Mode](#)
- [GUI Run Mode](#)

Batch Run Mode

The **Batch Run Mode** is the default run mode. In the **Batch Run Mode**, the mixed-signal simulator is invoked as a batch job. The simulation runs to completion (success or failure) and you can see its progress in the log file text viewer.

GUI Run Mode

The **GUI Run Mode** launches the simulation as an interactive job in one of the available digital debuggers for the VCS-based mixed-mode simulators.

Note:

In the **Environment Options** dialog box (**Setup > Environment Options**), set your **Output Data Format** to **fsdb** with the **Merged Output Data File** option on for best results.

You can select from the following digital debuggers in the **Environment Options** dialog box:

- Synopsys Verdi® (the default)
- Discovery Visual Environment (DVE)

You can start, control, and debug the simulation within the digital debugger. Exiting the digital debugger ends the simulation. Once you exit the digital debugger, control is returned to the PrimeWave Design Environment.

After-simulation plotting is not done in the GUI mode but the user can click the **Plot** button during or after simulation to plot the outputs given in the Outputs pane to the default waveform tool (waveform viewer).

Note:

Iterative simulation (corners and sweeps) is not supported in the GUI mode. Any enabled iterative simulation is disabled (with a warning message sent to the Custom Compiler console) when the GUI mode is selected.

To run the simulation in the **GUI Run Mode**:

1. Choose **Simulation > GUI Run Mode** from the PrimeWave Design Environment menu.
A tick mark appears next to the **GUI Run Mode** option.
2. Choose **Simulation > Run** from the PrimeWave Design Environment menu.

Running the simulation in the **GUI Run Mode** launches the digital debugger and invokes the debugger user interface. The digital debugger helps in cross-probing to verify the designs.

Note:

Another method to invoke the debugger is to choose **Tools > Digital Debugger** from the PrimeWave Design Environment menu.

In the Verdi debugger, the right-click menu (on an instance or net) shows the following options (double-click in Verdi's hierarchy pane to activate the selected instance/module):

- **Crossprobe in Custom Compiler:** You can right-click an instance or a net and choose the **Probe in Custom Compiler** option. The **Probe in Custom Compiler** option probes and shows the schematic with the appropriate block highlighted. As part of cross probing, if you select an object in the Custom Compiler Schematic Editor, the same object gets selected in the debugger.
- **Plot in waveform viewer:** You can right-click a net and choose the **Plot in waveform viewer** option to see the signal plotted in waveform viewer. You can also select multiple objects at the same time and then plot the selected signals in waveform viewer.

For more information about Verdi, see the *Synopsys Verdi® User Guide and Tutorial*.

Setting Mixed-Signal Simulator Options

After [Choosing a Simulator](#) and [Choosing a PrimeSim Simulation Engine](#) (if necessary), you can set options for mixed-signal simulation flows.

Choose which flow you are using:

- If you selected **System-Verilog** as the mixed-signal flow in the Simulator Options dialog box, continue to the [Specifying System-Verilog Flow Mixed-Signal Simulator Options](#) section.
- **Verilog-AMS**

Continue to the [Specifying Verilog-AMS Flow Simulator Options](#) section.

See Also

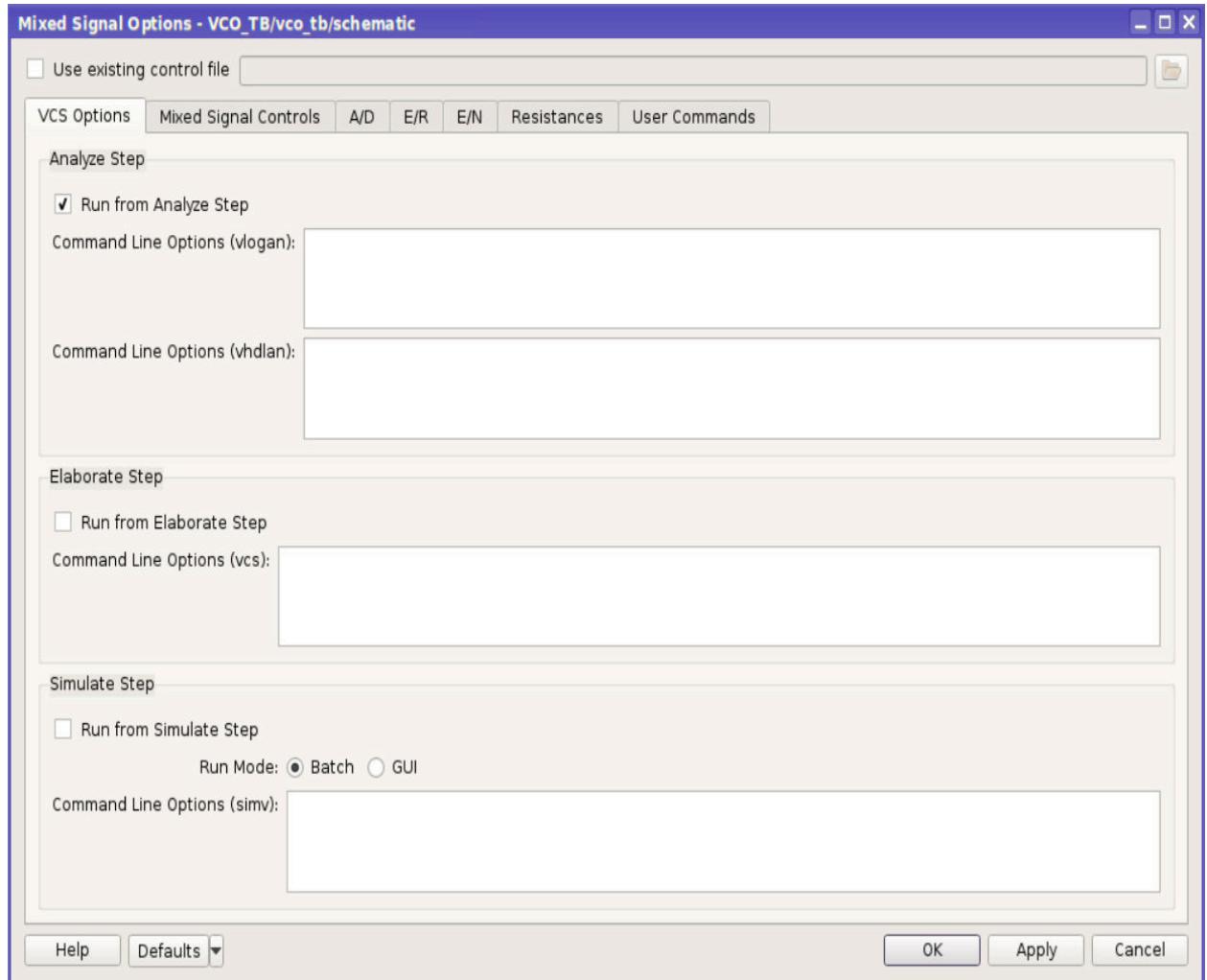
- [Mixed-Signal Control Commands](#) in the *VCS PrimeSim AMS User Guide*

Specifying System-Verilog Flow Mixed-Signal Simulator Options

After [Choosing a Simulator](#) and [Choosing a PrimeSim Simulation Engine](#) (if necessary), you can set options for mixed-signal simulation flows.

To set mixed-signal simulator options for the VCS PrimeSim AMS and the FineSim VCS integrations in the System-Verilog flow:

1. Choose **Simulation > Mixed-Signal Options** from the PrimeWave Design Environment menu bar. The **Mixed Signal Options** dialog box opens.



2. (Optional) If you have a control file, click the **Use Existing Control File** option, and enter the path for or browse to a control file.
3. Specify any **VCS Options**, **Mixed Signal Controls**, **A/D**, **E/R**, **E/N**, **Resistances**, or **User Commands**.

See the following sections for more information:

- [Specifying VCS Options](#)
- [Specifying Mixed Signal Controls](#)
- [Specifying A/D Options](#)

- Specifying E/R Options
 - Specifying E/N Options
 - Specifying Resistance Options
 - Specifying User Commands
4. Click **OK** to save your settings.

Specifying VCS Options

To specify VCS options in the System-Verilog flow:

1. Click the **VCS Options** tab.
2. Set **Analyze Step** options.
3. Set **Elaborate Step** options.
4. Set **Simulate Step** options.
5. Click **Apply** to save your changes.

Specifying Mixed Signal Controls

To specify Mixed Signal Controls in the System-Verilog flow:

1. Click the **Mixed Signal Controls** tab.
2. Enter the **SPICE Top Name**.
3. Select a **Bus format(s)** option: **[%d]** (the default), **<%d>**, or **_%d**.
4. Choose whether or not to **Resolve SPICE Inst x Prefix** (off by default).
5. Choose whether or not to **Enable IE Activity Report** (off by default).
6. Click **Apply** to save your changes.

Specifying A/D Options

To specify A/D options in the System-Verilog flow:

1. Click the **A/D** tab to display the **Analog-to-Digital Interface Elements** and **Digital-to-Analog Interface Elements** tables.
2. Click the "Click to Add" text in the **Analog-to-Digital Interface Elements** table to add elements of that type.

The row is automatically populated with the following value types:

Field	Description
Low Threshold*	Low logic level Threshold.
High Threshold*	High logic level Threshold.
Unit	Unit for threshold, valid values are v and %. Default is v.
High Impedance	If true, HiZ states are passed to the digital if the analog engine identifies them. Default is <i>false</i> .
Type*	Default is <i>net</i> but can also be <i>instTerm</i> or <i>term</i> .
Circuit Element*	Circuit element(s) to which this interface element applies. You can use a wildcard such as an asterisk (*) to match a range containing any character.
Vdd Net	The VDD net.
Min. Vdd	The minimum VDD value.
Min. Vdd Logic	The minimum VDD logic value.
Cap. Load	The capacitance load.
MidV Time	Sets the time (seconds) when the a2d command generates an X or Z-state.
MidV Logic	Specifies the output logic value.
X-Band	

Items marked with an asterisk are mandatory.

3. Click any of the values to change them as necessary.
4. Click the "Click to Add" text in the **Digital-to-Analog Interface Elements** table to add elements of that type.

The row is automatically populated with the following value types:

Field	Description
Power Net	When true, identifies the d2a node as an ideal voltage source without the resistance map resistor. Default is <i>false</i> .
Rise Time*	The analog rise time.
Fall Time*	The analog fall time.
Delay	The time delay before the analog transition. Default is 0.
Logic-X	Sets the rule on how a logic X must be translated to a voltage level on the analog side. Default is <i>midPoint</i> .

Field	Description
High Voltage*	Specifies the logic 1 voltage value.
Low Voltage*	Specifies the logic 0 voltage value.
Level Units*	The units of measure for the level.
Type*	Default is <i>net</i> but can also be <i>instTerm</i> or <i>term</i> .
Circuit Element*	Circuit element(s) to which this interface element applies. You can use a wildcard such as an asterisk (*) to match a range containing any character.
Vdd Net	The VDD net.

Items marked with an asterisk are mandatory.

5. Click any of the values to change them as necessary.
6. Click in the **Circuit Element** column to enter the name of an element, or click the **Select Object** button  to pick an element in your schematic.
7. (Optional) Click the **Force Nets** category name to display the **Force Nets** table, then add the following values as necessary:

Field	Description
DC Voltage	DC voltage to connect at the given net to replace any digital-to-analog interface element. Default is <i>0.0</i> .
Net*	Net(s) for which the override applies. You can use a wildcard such as an asterisk (*) to match a range containing any character.

Items marked with an asterisk are mandatory.

8. Click **Apply** to save your changes.

Specifying E/R Options

To specify E/R options in the System-Verilog flow:

1. Click the **E/R** tab to display the **Electrical-to-Real Interface Elements** and **Real-to-Electrical Interface Elements** tables.
2. Click the "Click to Add" text in the **Electrical-to-Real Interface Elements** table to add elements of that type.

The following value types are available:

Field	Description
Current	The current type. Click the check box to enable this element.

Field	Description
Min. Delta	Specifies the absolute current threshold value for e2r events.
Max. Delta	Sets the voltage difference between two e2r events to be less than or equal to the specified value, which must be greater than 0. This option disables the min_delta option.
Gain	The gain value.
Resistance	The unit of resistance.
Resistance Node	Specifies a path to a SPICE node name to connect the resistor.
Type	Default is <i>net</i> but can also be <i>instTerm</i> or <i>term</i> .
Circuit Element*	Circuit element(s) to which this interface element applies. You can use a wildcard such as an asterisk (*) to match a range containing any character.

3. Click in the **Resistance Node** column to enter the name of a resistance node, or click the **Select Object** button  to pick a node from your schematic.
4. Click in the **Circuit Element** column to enter the name of an element, or click the **Select Object** button  to pick an element in your schematic.
5. Click the "Click to Add" text in the **Real-to-Electrical Interface Elements** table to add elements of that type.

The following value types are available:

Field	Description
Current Type	The current type. Click the check box to enable this element.
Min. Delta	Specifies the absolute voltage threshold value for r2e events.
Gain	The gain value.
R/F Rate	Specifies how fast the voltage changes from the present value to the target value.
Type	Default is <i>net</i> but can also be <i>instTerm</i> or <i>term</i> .
Circuit Element*	Circuit element(s) to which this interface element applies. You can use a wildcard such as an asterisk (*) to match a range containing any character.

6. Click in the **Circuit Element** column to enter the name of an element, or click the **Select Object** button  to pick an element in your schematic.
7. Click **Apply** to save your settings.

Specifying E/N Options

To specify E/N options in the System-Verilog flow:

1. Click the **E/N** tab to display the **Electrical-to-Net Type Interface Elements** and **Net Type-to-Electrical Interface Elements** tables.
2. Click the "Click to Add" text in the **Electrical-to-Net Interface Elements** table to add elements of that type.

The following value types are available:

Field	Description
Net Type	The net type.
Net	A specified net.
Fidelity	The fidelity value.
Gain	The gain value.
Res	Specifies series resistance. The default is 1000 Ohms.
HiZ On	Turns on e2n drive strength calculation. Default is 1, which means calculation is on.

3. Click the "Click to Add" text in the **Net Type-to-Electrical Interface Elements** table to add elements of that type.

The following value types are available:

Field	Description
Net Type	The net type.
Net	A specified net.
Fidelity	The fidelity value.
R/F Rate	Specifies a unit of second/volt, with the default set to 10ps/V. This value indicates how fast the voltage changes from the current value to the target value.
Gain	The gain value.
Res	Specifies parallel resistance. The default is 1100G Ohms.
R Off	Series resistance value. Default is 10 G Ohm.
x2v	Voltage source value.

Field	Description
HIZ On	Turns on n2e drive strength calculation. Default is 1, which means calculation is on.

4. Click **Apply** to save your settings.

Specifying Resistance Options

To specify resistance options in the System-Verilog flow:

1. Click the **Resistances** tab to display the **Resistance Map Files** table.
2. (Optional) Enter the name or browse to a **Default Resistance Map** file.
3. Click the "Click to Add" text, and enter the **FileName** for a map file.
4. Choose a **Category**, and add an **Element** or **File** as necessary.
5. (Optional) Click the **Node Resistance Overrides** category name to display the **Resistance** table, and add any **Resistance** and **Net** values as necessary.
6. Click **Apply** to save your changes.

Specifying User Commands

To specify user commands in the System-Verilog flow:

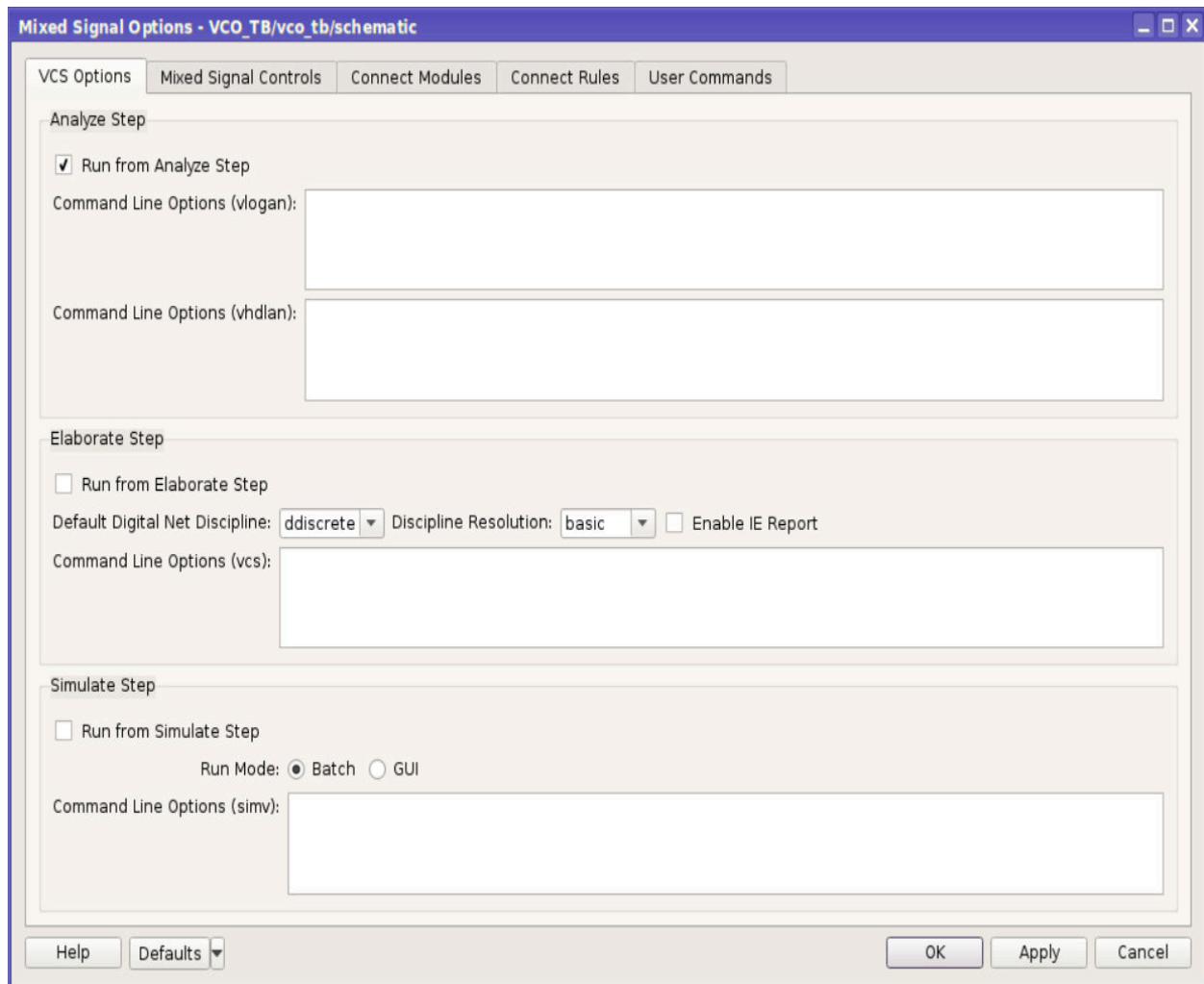
1. Click the **User Commands** tab to display the text boxes for entering any other VCS control file or UCLI file options you want to include.
2. Enter any needed VCS control file or UCLI file options in the respective text boxes.
3. Click **Apply** to save your changes.

Specifying Verilog-AMS Flow Simulator Options

After [Choosing a Simulator](#) and [Choosing a PrimeSim Simulation Engine](#) (if necessary), you can set options for mixed-signal simulation flows.

To set mixed-signal simulator options for the VCS PrimeSim AMS and the FineSim VCS integrations in the Verilog-AMS flow:

1. Choose **Simulation > Mixed-Signal Options** from the PrimeWave Design Environment menu bar. The **Mixed Signal Options** dialog box opens.



2. (Optional) If you have a control file, click the **Use Existing Control File** check box, and enter the path for or browse to a control file.
3. Specify any **Connections, E/R, Other, or Extra Options**.

See the following sections for more information:

- [Specifying VCS Options](#)
- [Specifying Mixed Signal Controls](#)
- [Specifying Connect Modules](#)

- Specifying Connect Rules
 - Specifying User Commands
4. Click **OK** to save your settings.

Specifying VCS Options

To specify VCS options in the Verilog-AMS flow:

1. Click the **VCS Options** tab.
2. Set **Analyze Step** options.
3. Set **Elaborate Step** options.
4. Set **Simulate Step** options.
5. Click **Apply** to save your changes.

Specifying Mixed Signal Controls

To specify Mixed Signal Controls in the Verilog-AMS flow:

1. Click the **Mixed Signal Controls** tab.
2. Enter the **SPICE Top Name**.
3. Select a **Bus format(s)** option: **[%d]** (the default), **<%d>**, or **_%d**.
4. Choose whether or not to **Resolve SPICE Inst x Prefix** (off by default).
5. Choose whether or not to **Enable IE Activity Report** (off by default).
6. Click **Apply** to save your changes.

Specifying Connect Modules

To specify connect modules in the Verilog-AMS flow:

1. Click the **Connect Modules** tab to display the **Connect Rule File** and **Connect Module File** tables.
2. Set options for **Install Connect Rule and Module Locations** and **User Connect Rule and Module Locations**.
3. Click **Apply** to save your settings.

Specifying Connect Rules

To specify connect rules in the Verilog-AMS flow:

1. Click the **Connect Rules** tab to display the **Connect Rule Name** and **File Path**.
2. Click **Click to add** to add new connect rules.
3. Modify parameters in the **Connect Modules** table by clicking a parameter value to select it. Use the  button to reset a parameter value.
4. Click **Apply** to save your settings.

Specifying User Commands

To specify user commands in the Verilog-AMS flow:

1. Click the **User Commands** tab to display the text boxes for entering any other VCS control file or UCLI file options you want to include.
2. Enter any needed VCS control file or UCLI file options in the respective text boxes.
3. Click **Apply** to save your changes.

Setting Block-Level Options

The following block-level options are available:

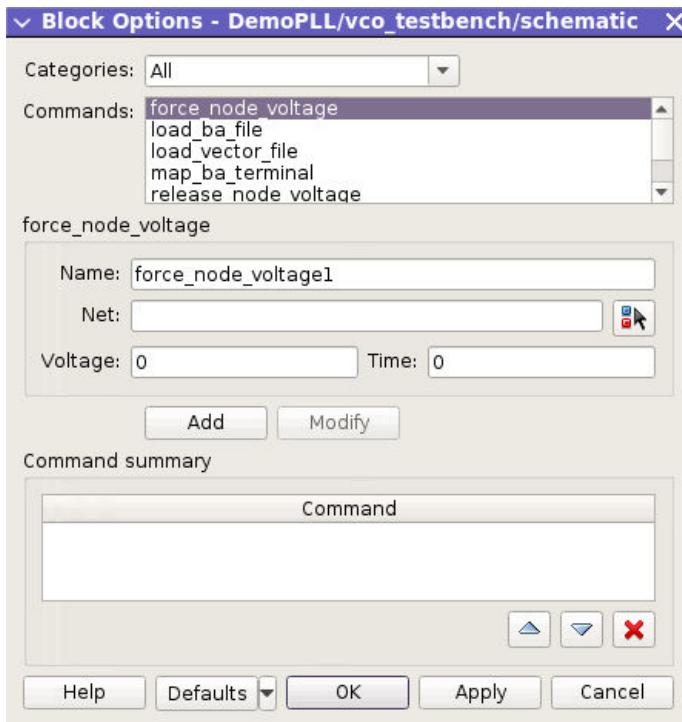
- [Setting Block-Level VCS PrimeSim AMS Options](#)
- [Setting Block-Level FineSim VCS Options](#)
- [Setting Block-Level FineSim Options](#)

Setting Block-Level VCS PrimeSim AMS Options

To set block-level simulator options for the VCS PrimeSim AMS integration:

1. Choose **Simulation > Analog Block Options** from the PrimeWave Design Environment main menu bar.

The **Block Options** dialog box opens.



2. Choose an option and specify any needed arguments.

See the *VCS PrimeSim AMS User Guide* for information on each option.

3. If available for the option you are setting, choose a scope for each option:

- **Global**

When available, this option applies to all instances.

- **Instance**

The option is applied only for the particular instance that you choose. You can type in the instance name or use the cross selection to pick the instance from a design.

- **Block**

This option is applied to all instances that use the same master. You are prompted to select an instance from your design. The master of that instance is used to scope the option. The master field next to the instance name is used for reporting purposes only and is not editable.

4. Click **Add** to add the specified option to the list.

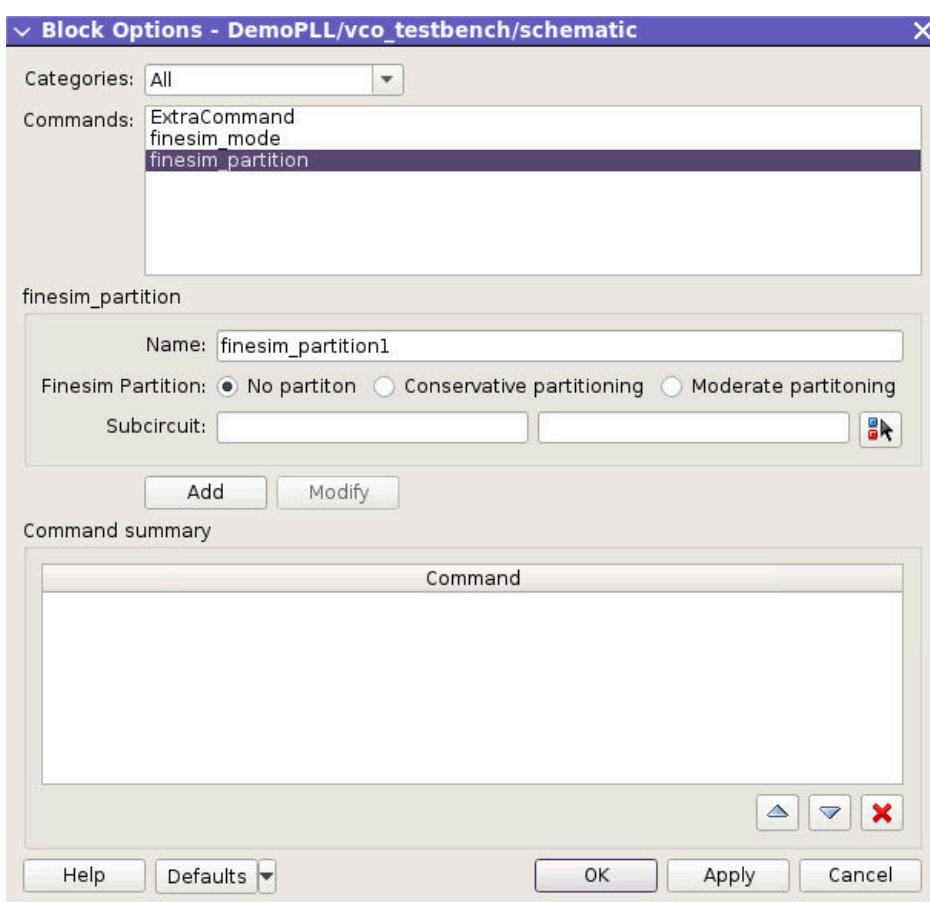
5. (Optional) If you need to modify an option, click the option name in the options table, change values as necessary, and click **Modify**. The table is updated with the value(s) you changed.
6. Click **OK** to save your changes.

Setting Block-Level FineSim VCS Options

To set block-level simulator options for the FineSim VCS integration:

1. Choose **Simulation > Analog Block Options** from the PrimeWave Design Environment main menu bar.

The **Block Options** dialog box opens.



2. Choose an option and specify any needed arguments.

See the [FineSim Pro Command Reference](#) in the *FineSim User Guide: Pro and SPICE Reference* for information on each option.

3. If available for the option you are setting, choose a scope for each option:
 - **Global**
When available, this option applies to all instances.
 - **Instance**
The option is applied only for the particular instance that you choose. You can type in the instance name or use the cross selection to pick the instance from a design.
 - **Block**
This option is applied to all instances that use the same master. You are prompted to select an instance from your design. The master of that instance is used to scope the option. The master field next to the instance name is used for reporting purposes only and is not editable.
4. Click **Add** to add the specified option to the list.
5. (Optional) If you need to modify an option, click the option name in the options table, change values as necessary, and click **Modify**. The table is updated with the value(s) you changed.
6. Click **OK** to save your changes.

Setting Block-Level FineSim Options

To set block-level simulator options for the FineSim integration:

1. Choose **Simulation > Block Options** (or **Simulation > Analog Block Options** when working in a test suite with mixed-signal simulation) from the PrimeWave Design Environment main menu bar.

The **Block Options** dialog box opens.
2. Choose an option and specify any needed arguments.

See the [FineSim Pro Command Reference](#) in the *FineSim User Guide: Pro and SPICE Reference* for information on available commands and their options.
3. If available for the option you are setting, choose a scope for each option:
 - **Global**
When available, this option applies to all instances.
 - **Instance**
The option is applied only for the particular instance that you choose. You can type in the instance name or use the cross selection to pick the instance from a design.

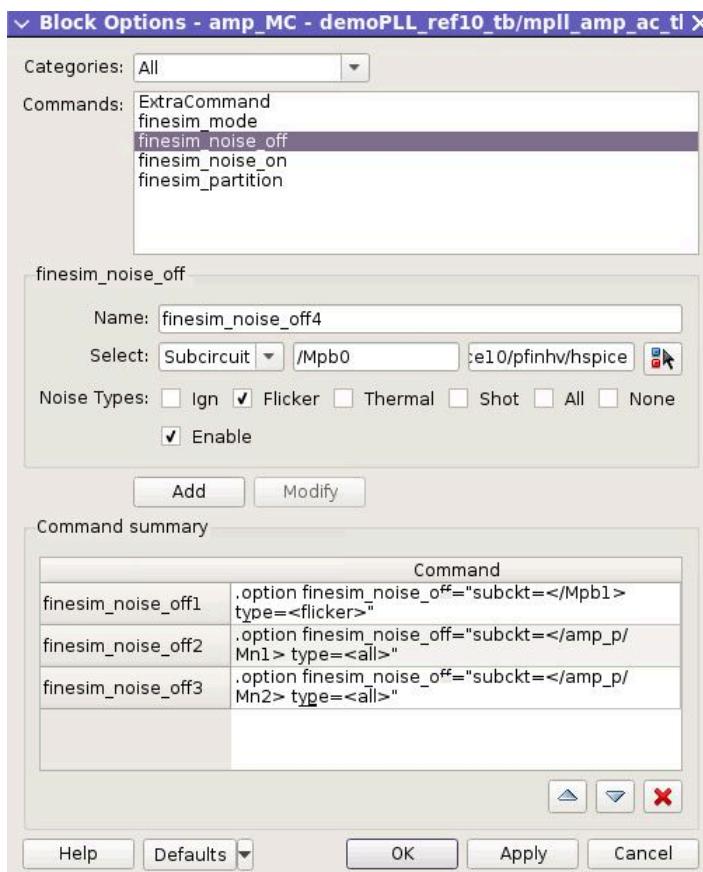
- **Block**

This option is applied to all instances that use the same master. You are prompted to select an instance from your design. The master of that instance is used to scope the option. The master field next to the instance name is used for reporting purposes only and is not editable.

4. Click **Add** to add the specified option to the list.
5. (Optional) If you need to modify an option, click the option name in the options table, change values as necessary, and click **Modify**. The table is updated with the value(s) you changed.
6. Click **OK** to save your changes.

Example

The following example shows the `finesim_noise_off` command, which is set up similarly to the `finesim_noise_on` command.



Select either **Instance** or **Subcircuit**. If you select **Instance**, when you **Pick** from the schematic design the text field is populated with the instance name. If **Subcircuit** is selected, when you **Pick** from the schematic design the text field is populated with the instance name and gets its subcircuit type automatically.

The `finesim_noise_off` and `finesim_noise_on` commands are mutually exclusive. You can set multiple commands using the same instance, but the simulator only takes as valid the last one in the netlist. In order to disable a command, you must uncheck **Enable** and then **Add** the command to the **Command** summary or modify it from there. If mutually exclusive commands are enabled at the same time, the netlisting process fails and an error message is issued.

Noise Types available to be injected include: **Ign**, **Flicker**, **Thermal**, **Shot**, **All**, and **None**. When **All** or **None** are selected, they disable the first four noise types, and they are mutually exclusive. You can select the **Ign**, **Flicker**, **Thermal**, **Shot** noise types together if you wish to specify more than one noise type.

Displaying the Simulation Log

To display the simulation log, choose **Simulation > Display Log** from the PrimeWave Design Environment main menu bar.

You can also choose **TestSuite > Options** and check the **Display Simulation Log** box in the **Session Options** dialog box.

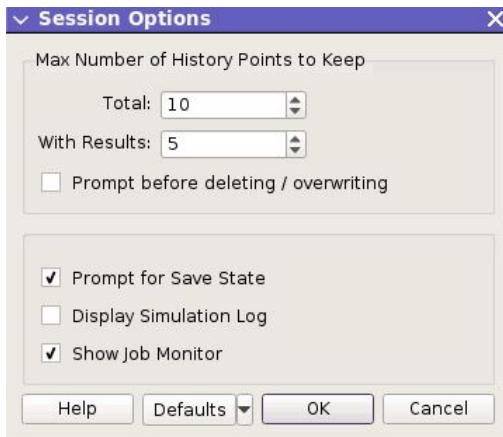
If the **Display Log** menu item is grayed out, then you have not run a simulation yet. See [Starting and Stopping Simulations](#).

Saving Simulation History

For single testbenches, you can save a backup of the results for each simulation you perform.

To save a history of simulation results:

1. Click **History**, which is located in the upper-right part of the PrimeWave Design Environment main window.
2. Specify the maximum number of backups you want to save. Choose **TestSuite > Options**. In the **Session Options** dialog box that opens, enter the number of history points to keep. To save space, you can save the results for a smaller number of history points.



Note:

When a results directory is backed up, the "nominal" directory is copied to history_1, and then history_2, and so on, until the history_<history_limit> directory is created. At that point, the oldest directory is deleted when the next simulation is run.

The number you enter specifies the number of additional results directories that are maintained on disk in addition to the current directory. By default, the tool saves your current results in a directory called "nominal", which can you can find in the following location on disk:

```
<top_level_results_dir>/<Library_Name>/<Cell_Name>/
<View_Name>/<simulator_name>
```

The <top_level_results_dir> directory is specified in the **Setup > Simulator** dialog box. See [Choosing a Simulator](#).

Note:

Any postprocessing references the selected data until you manually switch back to the current run or until another simulation is run.

Note:

History points are built into MTB mode.

Displaying Simulation Jobs

To show a list of PrimeWave Design Environment simulation jobs that are running in your current session, choose **Tools > Show Simulation Jobs** from the PrimeWave Design Environment menu bar. In the PrimeWave Design Environment, a PrimeWave-specific

version of the Job Monitor window is launched with a list of simulations that are currently running and the current status of the run. This filters out non-PrimeWave jobs.

In the PrimeWave-specific version of the Job Monitor window, information is initially displayed under four column headings **Job**, **Status**, **Host**, and **Run Description**. You can configure these columns using the following Tcl preferences: `xtJobMonitorJobColumn`, `xtJobMonitorStatusColumn`, `xtJobMonitorHostColumn`, and `xtJobMonitorRunDescriptionColumn`.

See Job Monitor Tool for more information.

This section contains information on the following topics:

- [Configuring Concurrent Job Limit and Queuing Engine Settings](#)
- [Selecting Columns for Display](#)
- [Grouping Jobs](#)
- [Operating on Selected Jobs](#)
- [Filtering Job Groups](#)

Configuring Concurrent Job Limit and Queuing Engine Settings

The **Job Management Options** dialog box is used to configure concurrent job limit and queue engine settings used for launching jobs.

To configure concurrent job limit and queue engine settings:

1. Choose **File > Options** from the Job Monitor window.

The **Job Management Options** dialog box opens.

2. Specify the **Batch Job Limits** setting.

In the **Batch Job Limits** group box you can specify the following:

- Choose **Job Class**.

Job Class is used to select a job limit.

- Choose **Concurrent Jobs**.

The number of concurrent jobs for the selected class can be adjusted using **Concurrent Jobs**. You can enable **Limit Jobs** such that, there are no limits set at all, in which case the **Job Class** will be empty, and **Concurrent Jobs** will be disabled.

3. Specify the **Queue Engine** setting.

In the **Queue Engine** section, you can specify the following:

- Choose the queue **Engine**.

You can select the default queue engine. The selected queue engine can be configured by clicking the **Configure** button. The Engine contains an entry labeled <None>. Selecting <None> launches all subsequent batch jobs without using a queue engine.

- Choose **Status Interval**.

Status Interval defines the wait period (in seconds) between calls, to query queue engines for job status.

4. Click **OK** to save the setting.

Using Remote Host Without Load-Queueing Systems

You can use remote hosts even in the absence of load-queueing systems such as Sun Grid Engine (SGE) and Load Sharing Facility (LSF).

In the **Job Management Options** dialog box, set the **Queue Engine** to **None**. Click **Configure** and enter the host name of the remote host in the dialog box.

In order to launch jobs on a specified remote host:

- Your `.rhosts` file must be set up so that you can log into the host using `rsh(1)` without requiring a password.
- The simulation directory must be on a shared drive with a mount point by the same name as the local host.
- Your `.bashrc`, `.cshrc`, or similar shell login script must set up your path to contain the simulator executable.

Selecting Columns for Display

You can pick the columns to display in the Job Monitor using the **View** menu option in the **Job Monitor** dialog box.

The following table details the columns in the **Job Monitor** dialog box.

Column Name	Description
Job	Displays a tree of job groups and jobs, with leaf nodes for each running application.
Status	Displays the status of the process, such as QUEUED, RUNNING, or KILLED. Progress feedback for the job is displayed as a bar graph along with the percentage completed. In addition, the status text is color-coded as follows: <ul style="list-style-type: none"> • FAILED - bright red • KILLED - dark red • Others - black (default text color)
Host	Specifies the host on which the process is running. The name of the Host is "localhost", if the process is on the same machine from where the Custom Compiler tool is invoked.
Run Description	Displays the description associated with the job.
Type	Specifies the job type. The possible job types are: <ul style="list-style-type: none"> • aim (AIM scripts) • batch (normal jobs) • interactive (interactive jobs) • group (grouping of jobs or groups)
Creation Time	Specifies the job creation time.
Start Time	Specifies the job start time.
End Time	Specifies the job end time.
Process ID	Specifies the process ID on the specified host.
Queue ID	The grid identifier associated with the job (only applies if the job was launched via SGE or LSF).
Output Files	Displays the names of the expected output files for the job.
%CPU	CPU usage of the job.
Peak RSS	Peak Resident Set Size, which is used to show how much memory is allocated to that process and is in RAM.
Job Types	The cascade menu Job Types contains the following two checkable options: <ul style="list-style-type: none"> - Batch - toggles the display of batch jobs in the Job Monitor. - Interactive - toggles the display of interactive jobs in the Job Monitor.

Grouping Jobs

Jobs are grouped hierarchically in the Job Monitor for convenience of use. Each job group is a node in the tree displayed in the **Job** column of the Job Monitor. Jobs that are not contained in any group are placed at the top level in the Job Monitor.

The job group is automatically deleted when the last contained job is deleted (this includes jobs contained in subgroups). A group containing a running job cannot be destroyed, the incomplete jobs must be killed first.

The job group status consists of the following details:

- The number of runs completed.
- The total number of runs.
- The number of unsuccessful runs.

Operating on Selected Jobs

To operate on selected jobs in the Job Monitor, you can select a row and right-click to activate the menu or select the **Job** menu option in the **Job Monitor** dialog box. The following table provides details of the menu options.

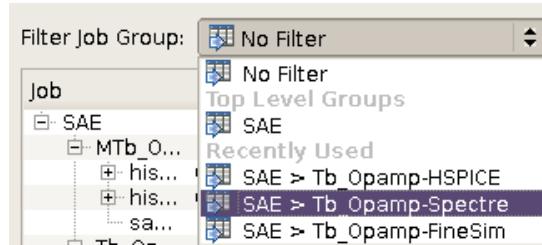
Menu Option	Description
Suspend	Suspends job.
Resume	Resumes job.
Kill	Kills job.
View Output	Displays job output.
Delete	Deletes job.
Delete All Completed	Deletes any and all jobs that are completed in the entire table.
Filter to this job group	Changes the current filter to the selected job group. This updates the Filter Job Group menu and history.

Filtering Job Groups

You can filter jobs that are visible in the Job Monitor in order to, for example, change from a specific PrimeWave Design Environment session to a more general one. The Job

Monitor also remembers the previous filters so you can go back and forth between filtered views.

Select the job group you wish to filter from the **Filter Job Group** menu.



14

Analyzing Simulation Data Using the Results Analyzer and Results Compare

This chapter contains information on how to analyze and compare simulation results.

This chapter contains the following topics:

- [Using the Results Analyzer](#)
 - [Using Results Compare](#)
-

Using the Results Analyzer

Caution:

Before using the Results Analyzer, ensure you have the desired simulation results data source selected. See [Choosing Simulation Result Data Sources](#).

The Results Analyzer encompasses the following postprocessing applications that work together to streamline the process of analyzing outputs and building expressions:

- Plotting Assistant

With this utility, you can plot outputs that you commonly need such as a differential transient signal, the group delay of an AC result, and the voltage gain of a DC analysis. As you plot these items, the plotting assistant captures the corresponding expression, which you can later use to customize the outputs.

See [Using the Plotting Assistant](#).

- Calculator

The calculator provides a collection of functions for operating on signal data.

See [Accessing the Calculator](#).

- Statistical Analysis

When Monte Carlo data is present, you can create histograms and scatter plots from this utility.

See [Analyzing Statistical Data](#).

- Parametric Reduction

When a testbench setup includes corners, parametric analyses, or Monte Carlo analyses, you can choose which of these to plot, as well as specify datasets used in the Calculator.

See [Using Parametric Reduction](#).

- Measures Assist

You can add PrimeSim HSPICE `.MEASURE` statements to your testbench here.

See [Creating .MEASURE Statements](#).

Note:

When multiple testbenches are analyzed, data in the Results Analyzer is updated after calculations are completed for each testbench. If you keep the Results Analyzer open, the data is automatically refreshed after each testbench is completed.

This section contains the following topics:

- [Opening the Results Analyzer](#)
- [Choosing Simulation Result Data Sources](#)
- [Using the Plotting Assistant](#)
- [Creating DC Operating Point Expressions for an Instance Included in a DSPF Netlist File](#)
- [Accessing the Calculator](#)
- [Creating .MEASURE Statements](#)
- [Analyzing Statistical Data](#)

Opening the Results Analyzer

To open the Results Analyzer, choose **Results > Analyzer** from the PrimeWave Design Environment main menu bar.

Choosing Simulation Result Data Sources

At the top of the Results Analyzer, click the **Testbench** button to use the testbench directory data, or click the **Reference** button to browse to and choose another directory that contains the simulator results you want.

If you click the **Both** button, expressions are evaluated and plotted twice: once from the testbench data, and once from the specified simulation results data directory. If an expression is invalid for one of the directories, an error message is issued to the console for that evaluation. The other directory result is still plotted if evaluated successfully. Ensure that the reference directory netlist and results are similar enough to the current setup that the expression is valid for its results, too.

Using the Plotting Assistant

The Plotting Assistant provides a sequentially coordinated flow for creating the waveform and measurement expressions you most commonly use. To access the Plotting Assistant, click the **Plot Assist** tab in the Results Analyzer.

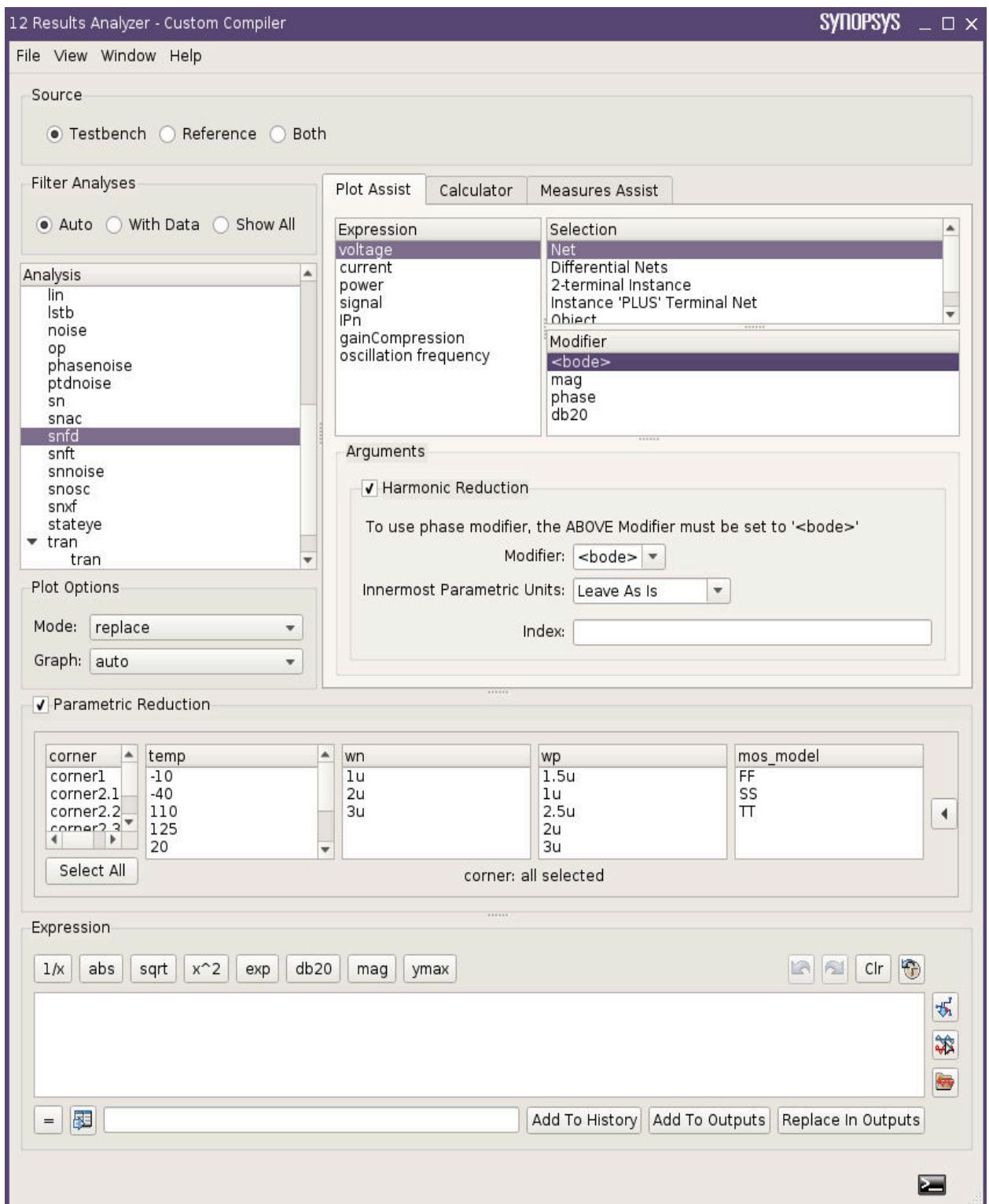
You can use the **<bode>** modifier in the Results Analyzer for Plotting AC signals. Choosing this modifier plots dB and phase values of any selected signal.

Plotting Signals With the Plotting Assistant

To plot a signal using the Plotting Assistant:

1. Choose **Results > Analyzer** from the PrimeWave Design Environment main menu bar.
The Results Analyzer window opens.
2. Click the **Plot Assist** tab in the Results Analyzer window.

Chapter 14: Analyzing Simulation Data Using the Results Analyzer and Results Compare
Using the Results Analyzer



3. (Optional) Choose your simulation result data source if you have not already done so. See [Choosing Simulation Result Data Sources](#) for instructions. Only analysis results located in the source results directory are listed in the **Analysis** pane.
4. (Optional) Choose how to **Filter Analyses**:
 - **Auto** selects between **With Data** and **Show All** depending on whether there is any data. If there is no data, the behavior of **Auto** matches **Show All**. If there is data, the behavior of **Auto** matches **With Data**.
 - **With Data** displays only analyses for which you have results.
 - **Show All** shows all the analyses supported by the simulator integration.
5. Choose an analysis that has associated output you want to access.

Analyses appearing in italics indicate that the analysis has corresponding expressions that can be used in the Plotting Assistant. Analyses that are not listed in italics contain no expressions to plot, but you can use them in conjunction with the Calculator.

Note:

If you do not choose an analysis, expressions are evaluated against all analyses that are in the run directory.

6. Choose an item from the **Expression** pane.

A message appears in the lower-left corner of the Results Analyzer window asking you to select an object to include in the design canvas.

Based on the expression you select, a **Selection** list appears. These selections control the type of object(s) to be selected in the design canvas.

In some cases, such as when using AC data, a **Modifier** list also appears.

7. (Optional) Enable **Harmonic Reduction** to access arguments for plotting harmonics. Depending on the expression being constructed, there might be additional optional or mandatory fields to configure. For mandatory field values, follow the instructions shown in the lower-left corner of the Results Analyzer.
8. Once all necessary selections are made, the corresponding signal is plotted when you click either **Evaluate/PLOT**  or **Evaluate/Tabulate** .

If the expression evaluates to a single scalar or a waveform, the **Evaluate/PLOT** button must be used to see the scalar or waveform. If the expression evaluates to a group of scalars, then the **Evaluate/Tabulate** button must be used to open a new table window showing the scalar values.

See [Example: Plotting a Transient Differential Signal](#) for an example of how to plot a signal with the Plotting Assistant.

Example: Plotting a Transient Differential Signal

To plot a transient differential signal:

1. Ensure the Results Analyzer is open (**Results > Analyzer**).
2. In the **Analysis** pane, select the **tran** analysis.
3. In the **Expression** pane, select **voltage**.
4. In the **Selection** pane, select **Differential Nets**.
5. In the design canvas, select one net, and then a second.

After the second selection, the corresponding signal appears in the waveform viewer.

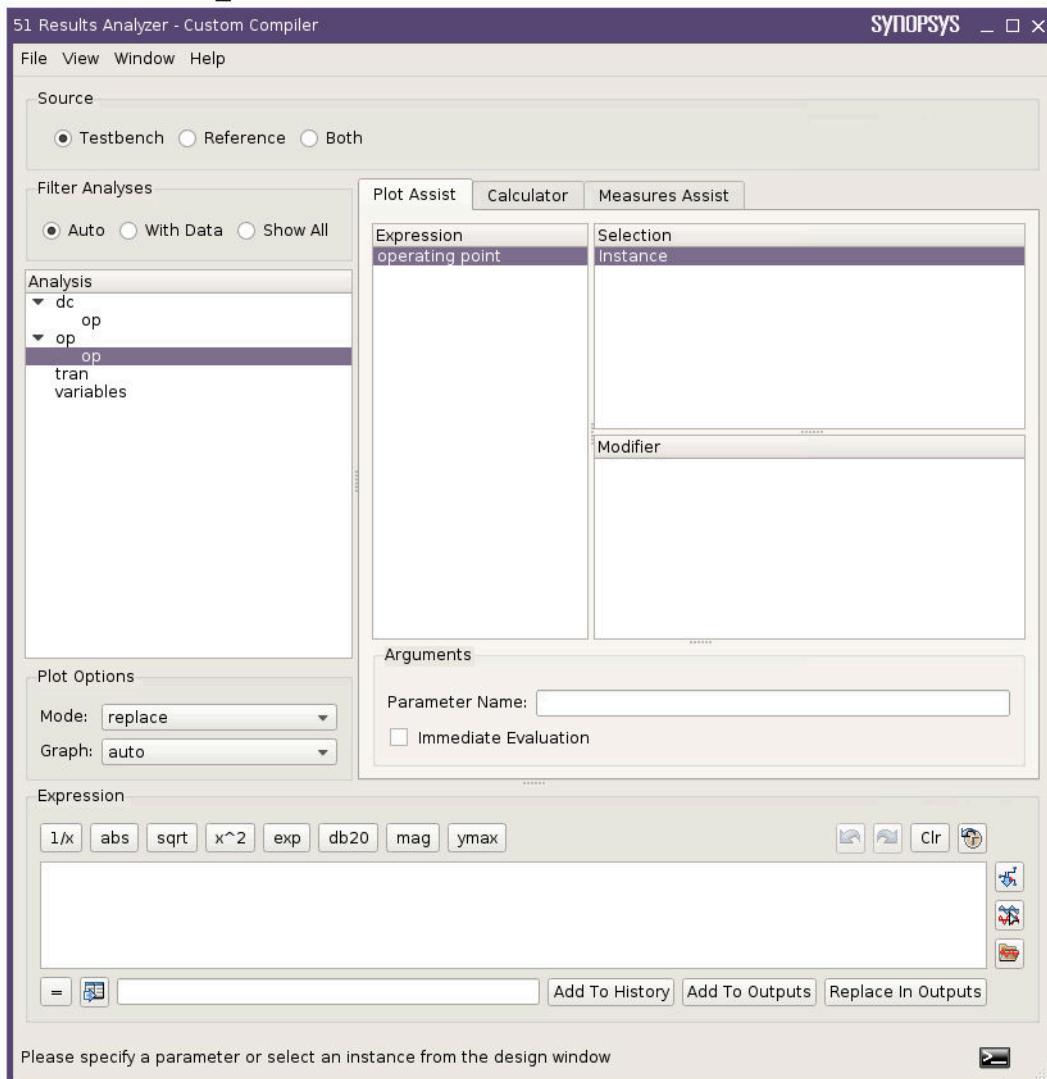
Creating DC Operating Point Expressions for an Instance Included in a DSPF Netlist File

For post-layout simulations with a DSPF netlist, you can create the DC operating point (OP) expression for instances in the DSPF file using the Results Analyzer. This is supported for PrimeSim HSPICE and FineSim.

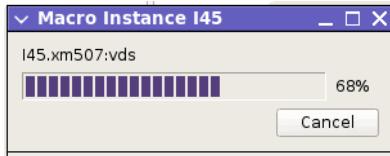
To do this:

1. Netlist and run a simulation that includes tran, dc, and op analyses.
2. Open the Results Analyzer. In the Results Analyzer window, select:
 - Analysis = op
 - Expression = operating point
 - Selection = Instance

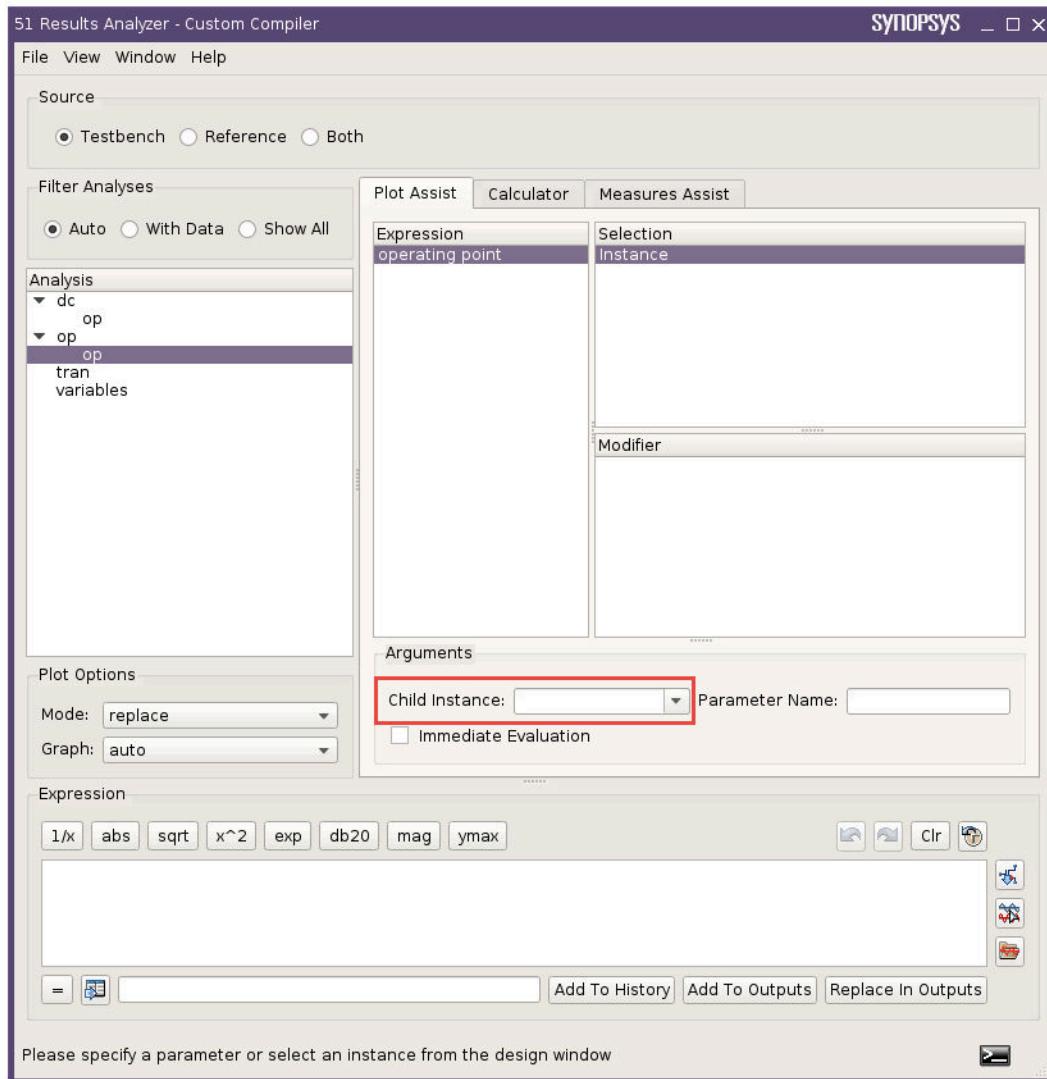
Chapter 14: Analyzing Simulation Data Using the Results Analyzer and Results Compare Using the Results Analyzer



3. In the Schematic Editor, select the symbol view of the subcircuit for which you have included the DSPF file. This opens the **Macro Instance** dialog box, which shows:
 - The instance number, of the subcircuit.
 - The devices inside that subcircuit, along with the parameters that are defined in the CDF for those instances.
 - A progress bar showing the progress of data accumulated.



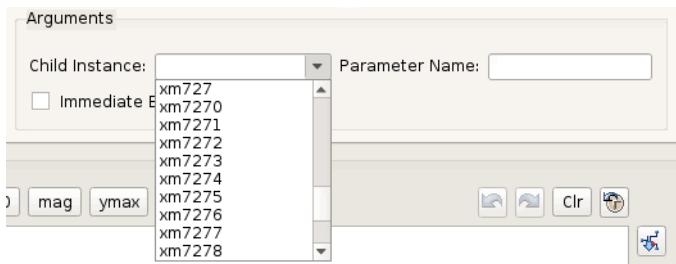
4. Once the data is compiled, the Results Analyzer **Arguments** section shows a new field, **Child Instance**.



Note:

To get the **Child Instance** name from the DSPF netlist initially, you need to first run an OP simulation.

5. Click in the drop-down menu to show the list of all instances and parasitic devices that were read from the DSPF file.

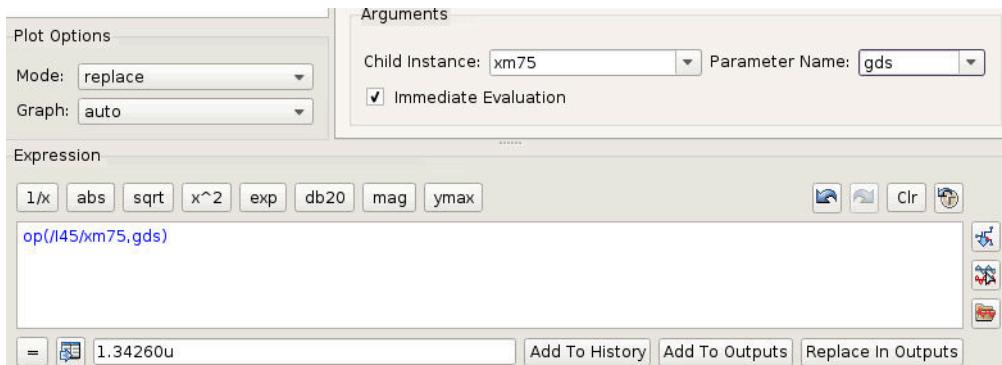


Once you select an instance, the **Parameter Name** field loads the list of all operating point parameters for that device.



6. Click **Immediate Evaluation** for quick evaluation of the expression.

Select a parameter for which you want to create an expression. As soon as you select the parameter, the expression is added to the **Expression** editor pane.



7. Add the expression to the PrimeWave Design Environment outputs table by clicking the **Add To Outputs** button. You can use a similar procedure to add expressions for other devices and add them to the PrimeWave Design Environment outputs table.

Outputs	Specifications	Scatter	Histogram	Q-Q	Parametric Reduction
Output	Expression				
	op(/l45/xm7401, gds, analysisName="op")				
	op(/l45/xm44, ids, analysisName="op")				
	op(/l45/xm75, gds, analysisName="op")				
Click to add					

- Run the simulation and notice that these expressions are evaluated automatically.

Outputs	Specifications	Scatter	Histogram	Q-Q	Parametric Reduction
Output	Expression	Value			
	op(/l45/xm7401, gds, analysisName="op")	2.04476p			
	op(/l45/xm44, ids, analysisName="op")	18.0599n			
	op(/l45/xm75, gds, analysisName="op")	1.34260u			
Click to add					

Accessing the Calculator

The Calculator provides a collection of functions used to analyze simulation data. To access the Calculator, click the **Calculator** tab in the Results Analyzer.

See Using the PrimeWave Design Environment Calculator in the *Synopsys PrimeWave Design Environment: Calculator Function Reference Manual* for complete information about calculator functions and their use.

Creating .MEASURE Statements

To create a `.MEASURE` statement, which you can add to your outputs or as part of an expression:

- Ensure the Results Analyzer is open (**Results > Analyzer**), and click the **Measures Assist** tab.
- The `.MEASURE` statement options are displayed.
- Enter a name for the measurement in the **Measure Name** text field.
- Choose an analysis from the **Analysis** menu.
- Choose the type of `.MEASURE` statement you want to create:
 - TRIG/TARG**
 - Find/When**
 - Functions**

- **Derivative**
 - **Equation**
5. Enter values for the type of .MEASURE statement you are creating.
See the *PrimeSim HSPICE Reference: Commands and Control Options* for information on the .MEASURE statement and options.
 6. Click **Show Preview** to see the corresponding .MEASURE statement created. Change the entries if needed.
 7. Click **Add to Outputs** to reflect this measure in **Outputs** pane of the PrimeWave Design Environment.
-

Analyzing Statistical Data

You can use the Results Analyzer to create histograms and scatter plots from Monte Carlo simulation results. To access the Statistical Analysis setup, click the **Statistical Analysis** tab, which is to the right of the **Calculator** tab in the Results Analyzer.

Note:

If you do not see the **Statistical Analysis** tab in the Results Analyzer, then you do not have any Monte Carlo data present in your testbench.

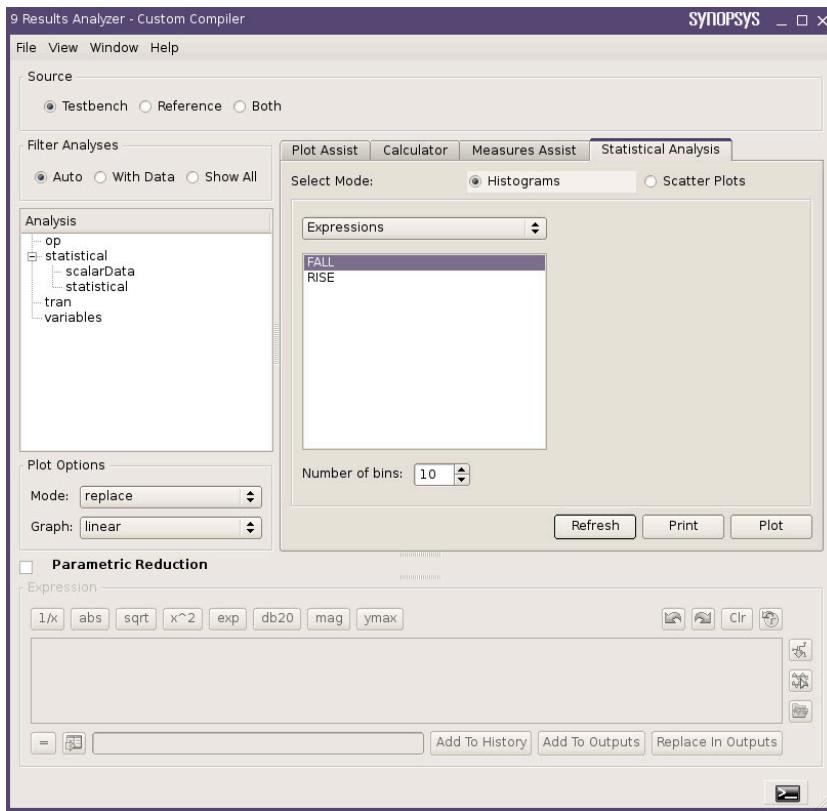
This section contains information on the following topics:

- [Creating Histograms](#)
- [Creating Scatter Plots](#)
- [Using Parametric Reduction](#)

Creating Histograms

To create a histogram:

1. Ensure the Results Analyzer is open (**Results > Analyzer**), and click the Statistical Analysis tab.
2. Choose **Histograms** as the **Select Mode**.



3. Choose **Expressions** or **Input Parameters** from the menu.

The **Expressions** correspond to the outputs defined in the testbench, which are evaluated as scalar values for each Monte Carlo iteration. The **Input Parameters** are statistically varying parameters as defined in the Monte Carlo Analysis.

See the PrimeSim HSPICE documentation for more information.

4. Select one or more expressions or parameters for which you want to build a histogram.
All expressions and parameters are combined into one histogram.
5. Enter the **Number of bins** you want to include in the histogram.
6. Click **Plot** to plot the histogram in the waveform viewer.

Note:

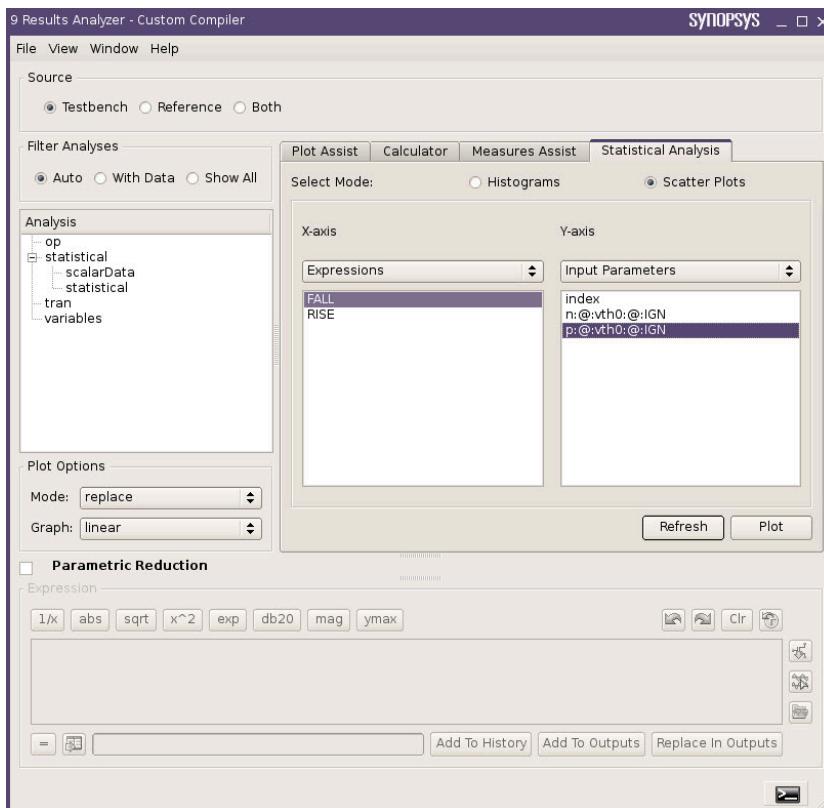
The mean and standard deviation are displayed in histograms.

You can also click **Print** to print a report of the iteration versus value for any expression or parameter. The report is a table displaying the value versus the iteration number.

Creating Scatter Plots

To create a scatter plot:

1. Ensure the Results Analyzer is open (**Results > Analyzer**), and click the Statistical Analysis tab.
2. Choose **Scatter Plots** as the **Select Mode**.



3. Choose one item (**Expressions** or **Input Parameters**) from each of the menus.

The item you choose from the menu on the left defines the values for the x-axis, and the item you choose from the menu on the right defines values for the y-axis.

4. Click **Plot** to plot the scatter plot in the PrimeWave Design Environment.

Using Parametric Reduction

Note:

Parametric reduction is not available for Monte Carlo results when the **Statistical Analysis** tab is open in the Results Analyzer.

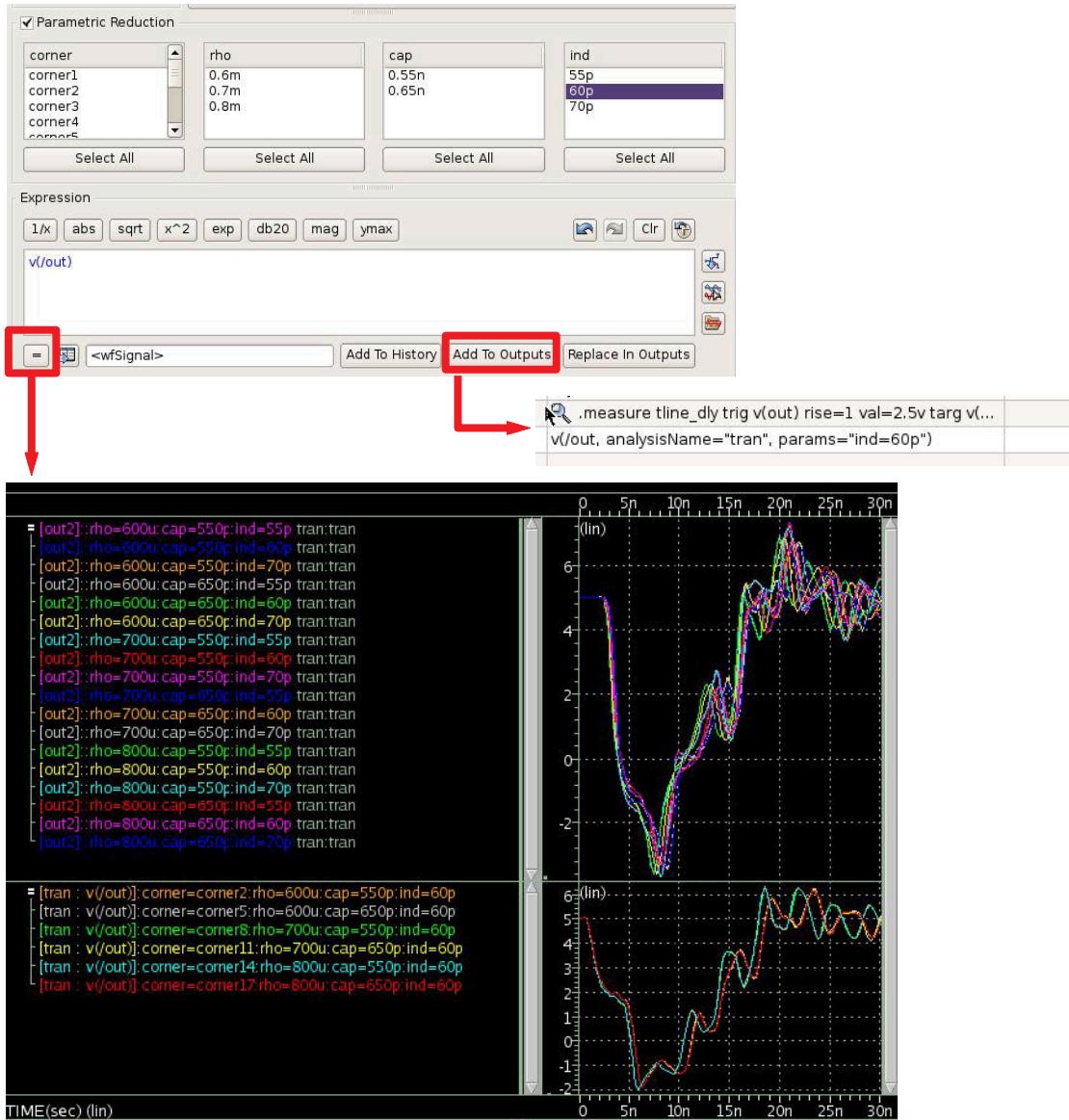
When a testbench setup includes corners, parametric analyses, or Monte Carlo analyses, the **Parametric Reduction** pane appears above the **Expression** pane in the Results Analyzer. The **Parametric Reduction** pane includes a list of all corners, parameters (with their swept values) and Monte Carlo iterations that are included in your testbench.

You can select which of the available corners, parameters, and Monte Carlo iterations you want to include in your postprocessing. These selections control the signals that are plotted from the Results Analyzer and specify which datasets are included when you use the Calculator.

Note:

Expressions in the expression buffer are evaluated only on the specified reduced set of traces. You do not need to change expressions.

You can perform parametric reduction for corners based on corner conditions or corner name. Corner parameters can be specified as part of the `params` argument using the same syntax as sweep parameters. This can help you filter waveforms in the waveform viewer by those parameters. Additionally, expressions that are added to the **Outputs** table from the Calculator include corner parameters.



Note:

The Primewave Design Environment's display of corner conditions is only supported in PrimeWave Design Environment X-Y plots. Other plot types (like 2-D and eye diagrams) do not support the display of corner conditions. If you do not see a **Sweep Attributes** menu choice in your menu, you are not in an X-Y plot.

Using Results Compare

Note:

Results can only be compared when simulations are run in multiple testbench mode.

The Results Compare tool compares multi-testbench simulation results between runs (History Points), which provides an additional level of data mining to assess the impact of design and testbench deltas.

You can use the Results Compare tool to determine if some of your measurements are failing to meet specifications after running a simulation on your initial design. After making a design change, you can rerun the testbenches and then compare the results to determine what impact those changes have on specifications.

The Results Compare tool uses the Results Database to extract and compare data. Visual indicators are displayed, which indicate if the changes improve or degrade the design behavior.

To compare simulation results between two runs, choose **Results > Results Compare** from the PrimeWave Design Environment menu bar. The Results Compare window opens.

The following types of comparisons are available, which you can choose from the **View Mode** menu in the upper-left corner of the Results Compare window:

- **summary**

A comparison of all measurements between two history points (the default). For each iteration of a measurement, the change is calculated and the min and max variations for each measurement is displayed.

See [Viewing Comparison Summary Results](#).

- **measurement**

A comparison of a single measurement over all iterations in two different history points. See [Comparing Measurement Results](#).

- **iteration**

A comparison of all measurements between two individual iterations in two different history points. See [Comparing Iteration Results](#).

To resort any of the columns alphabetically or numerically, click the title of a column in the table of results.

This section contains information on the following topics:

- [Viewing Comparison Summary Results](#)
- [Comparing Measurement Results](#)
- [Comparing Iteration Results](#)
- [Measurement and Iteration View Coloring Rules](#)

 [Show me](#) how to use Results Compare.

Viewing Comparison Summary Results

To view a summary of all measurements compared between two history points, ensure **summary** is selected from the View Mode menu on the Results Compare window (**Results > Compare** from the PrimeWave Design Environment tab page menu bar). A summary of the differences between two history points is displayed along with the number of new and eliminated violations.

Choose history points to compare from the **Reference** and **Current** menus. By default, the last (most recent) history point is set as Current, and the history point just before the **Current** one is set as the **Reference** history point.

Note:

You can double-click any row in the summary results to open the measurement results view in the Results Compare window.

The following information is available in the summary view:

Category	Description
Testbench	The name of a testbench.
Measurement	A measurement in a testbench.
Note:	
	If measurements do not match between the Reference and Current testbenches, a warning is issued in the console, and only the common measurements are displayed.
Specification	The specification value.
Reference Violations	The number of iterations where the specification is not met in the reference history point.
Current Violations	The number of iterations where the specification is not met in the current history point.

Category	Description
Violation Diff	The violation difference value. Cells are colored red when the violation difference value is greater than 0; cells are colored green when the violation difference value is less than 0.
Min Diff	The minimum difference value, which is calculated using the following equation that iterates across each iteration and measurement: $\min(\text{TBCurrent}(\text{Iteration}_I \# (\text{Measurement}_M)) - \text{TBRef}(\text{Iteration}_I \# (\text{Measuelement}_M)))$
Max Diff	The maximum difference value, which is calculated using the following equation that iterates across each iteration and measurement: $\max(\text{TBCurrent}(\text{Iteration}_I \# (\text{Measurement}_M)) - \text{TBRef}(\text{Iteration}_I \# (\text{Measuelement}_M)))$
Min Diff %	The minimum percentage difference. All percentage values are absolute values.
Max Diff %	The maximum percentage difference. All percentage values are absolute values.
Magnitude and Direction of the Max Difference (no heading text displayed)	<p>The contents of this column depends on the magnitude and direction of the maximum percentage difference. An equal sign (=), or a number of plus (+) or minus (-) signs are displayed to show this difference. If the maximum percentage difference for the Reference and history points stays the same (0% exactly), then an = is displayed. If the maximum percentage difference increases from the Reference History Point to the Current History Point:</p> <ul style="list-style-type: none"> • 0% < x <= 10%: + • 10% < x <= 25%: ++ • 25% < x <= 50%: +++ • > 50%: +++++ <p>If the maximum percentage difference decreases from the Reference History Point to the Current History Point:</p> <ul style="list-style-type: none"> • 0% < x <= 10%: - • 10% < x <= 25%: -- • 25% < x <= 50%: --- • > 50%: -----

Comparing Measurement Results

To compare measurement results:

1. Ensure **measurement** is selected from the **View Mode** menu on the Results Compare window (**Results > Compare** from the PrimeWave Design Environment tab page menu bar).

A table is displayed, which shows the comparisons for all iterations of selected measurements and testbenches. The results window is automatically updated when resimulate your design and update the simulation results.

2. Choose simulation results to compare from the **Reference** and **Current** menus.
3. Choose which testbench measurement results you want to view from the **Testbench** menu.
4. Choose a measurement from the **Measurement** menu.

The following information is available in the measurement view:

Category	Description
Corner/Monte Carlo	The corner name or Monte Carlo iteration.
Corner/Sweep Variable <temperature> or <variable_name>	One or more columns that contain the corner/sweep variable names.
Corner Model File <model_file_name>	One or more columns that contain the corner model variable values.
<measurement_name> Reference	The reference value for the measurement you choose from the Measurement menu. See Measurement and Iteration View Coloring Rules for information on how the cells are colored in relation to your specification.
<measurement_name> Current	The current value for the measurement you choose from the Measurement menu. See Measurement and Iteration View Coloring Rules for information on how the cells are colored in relation to your specification.
Difference	The difference between the reference and current measurement values.
Difference %	The percentage difference between the reference and measurement values.

Category	Description
Magnitude and Direction of the Max Difference (no heading text displayed)	<p>The contents of this column depends on the magnitude and direction of the maximum percentage difference. An equal sign (=), or a number of plus (+) or minus (-) signs are displayed to show this difference. If the maximum percentage difference for the Reference and history points stays the same (0% exactly), then an = is displayed. If the maximum percentage difference increases from the Reference History Point to the Current History Point:</p> <ul style="list-style-type: none"> • $0\% < x \leq 10\%$: + • $10\% < x \leq 25\%$: ++ • $25\% < x \leq 50\%$: +++ • $> 50\%$: ++++ <p>If the maximum percentage difference decreases from the Reference History Point to the Current History Point:</p> <ul style="list-style-type: none"> • $0\% < x \leq 10\%$: - • $10\% < x \leq 25\%$: -- • $25\% < x \leq 50\%$: --- • $> 50\%$: ----

Comparing Iteration Results

To compare iteration results across reference and current iterations:

1. Ensure **iteration** is selected from the **View Mode** menu on the Results Compare window (**Results > Compare** from the PrimeWave Design Environment tab page menu bar).

A table of iteration results is displayed. The results window is automatically updated when you resimulate your design and update the results.

2. Choose simulation results to compare from the **Reference** and **Current** menus.
3. Choose which testbench measurement results you want to view from the **Testbench** menu.
4. Choose a reference and current iteration from the **Reference Iteration** and **Current Iteration** menus, respectively.

If a testbench (either the current or the reference) has sweep parameters or Monte Carlo iterations specified or enabled, you can select values for those parameters.

The following information is available in the iteration view:

Category	Description
Testbench	The name of a testbench.

Category	Description
Measurement	<p>A measurement in a testbench.</p> <p>Note:</p> <p>If measurements do not match between the Reference and Current history points, a warning is issued in the Console, and only the common measurements are displayed.</p>
Specification	<p>The specification value.</p> <p>See Measurement and Iteration View Coloring Rules for information on how the cells are colored in relation to your specification.</p>
Reference	<p>The reference measurement value.</p> <p>See Measurement and Iteration View Coloring Rules for information on how the cells are colored in relation to your specification.</p>
Current	The current measurement value.
Difference	The difference value between the measurement values.
Difference %	The difference percentage between the measurement values. All percentage values are absolute values.
Magnitude and Direction of the Max Difference (no heading text displayed)	<p>The contents of this column depends on the magnitude and direction of the maximum percentage difference. An equal sign (=), or a number of plus (+) or minus (-) signs are displayed to show this difference. If the maximum percentage difference for the Reference and history points stays the same (0% exactly), then an = is displayed. If the maximum percentage difference increases from the Reference History Point to the Current History Point:</p> <ul style="list-style-type: none"> • $0\% < x \leq 10\%$: + • $10\% < x \leq 25\%$: ++ • $25\% < x \leq 50\%$: +++ • $> 50\%$: ++++ <p>If the maximum percentage difference decreases from the Reference History Point to the Current History Point:</p> <ul style="list-style-type: none"> • $0\% < x \leq 10\%$: - • $10\% < x \leq 25\%$: -- • $25\% < x \leq 50\%$: --- • $> 50\%$: ----

Measurement and Iteration View Coloring Rules

Colors are used to signify the change towards (green color) or away (red color) from a specification, which is the same as the coloring in the ResultsView.

When the Reference value equals the Current value, cells in the Reference and Current columns are colored green. Otherwise, cells are colored red or green depending on the specification you use.

The following rules apply when a Reference or Current value is assigned a green or red color:

- For greater-than goals:
 - If the difference between the current and reference measurement values is less than 0: red
 - If the difference between the current and reference measurement values is greater than or equal to 0: green
- For less-than goals:
 - If the difference between the current and reference measurement values is less than or equal to 0: green
 - If the difference between the current and reference measurement values is greater than 0: red
- For reference range goals within the target region:
 - Current measurement value in the target region: green
 - Current measurement value outside of the target region: red
- For reference goals outside the target region:
 - Current measurement value closer to the target region: green
 - Current measurement value farther from the target region: red
- For exact goals:
 - Current measurement value is closer than the reference measurement value to specification: green
 - Current measurement value is farther than reference measurement value from the specification: red

15

Using the ResultsView

This chapter contains information on how to view simulation results in the ResultsView.

The ResultsView provides a way to incrementally view simulation results for each iteration run, analyze the data against specifications, look for trends and causes of specification failures. The ResultsView consolidates all measurements across all corners/sweeps/agings as well as across all testbenches in one history points. Once one of the simulation jobs in a run is done, the ResultsView launches automatically. You can view the results for a single testbench, multiple testbenches, and multiple history points. You can export and share the results in CSV or HTML format.

All scalar data (both .MEASURE and PrimeWave Design Environment expressions) is stored in a special results database, which eliminates the need to constantly rerun measurements or archive waveform data. In addition, conditions for each iteration (corner conditions, sweep values, and Monte Carlo iterations) are also stored in the results database. Links to allow selective plotting of signals, as well as images of specified plots, is also included in the results database.

You can launch the ResultsView before, during, or after all the simulations are complete. The results update incrementally, so you can see the progress in real time.

This section contains information on the following topics:

- [Opening the ResultsView](#)
- [ResultsView User Interface](#)
- [ResultsView Menus](#)
- [ResultsView Toolbar Buttons](#)
- [Results Tables](#)
- [Outputs Tree View](#)
- [Viewing Incremental Results](#)
- [Plotting Results From the ResultsView](#)
- [Displaying Netlists](#)

- [Displaying Output Logs](#)
- [Displaying Images](#)
- [Launching the Terminal From a Result](#)
- [Rerunning Iterations](#)
- [Opening New Sessions](#)
- [Exporting ResultsView Results](#)
- [Selecting Items for Column Headers](#)
- [Filtering Column Results](#)
- [Showing and Hiding Columns in the Results Table](#)
- [Reordering Table Results](#)
- [Highlighting Whole Rows and Columns](#)
- [Adding Descriptions](#)

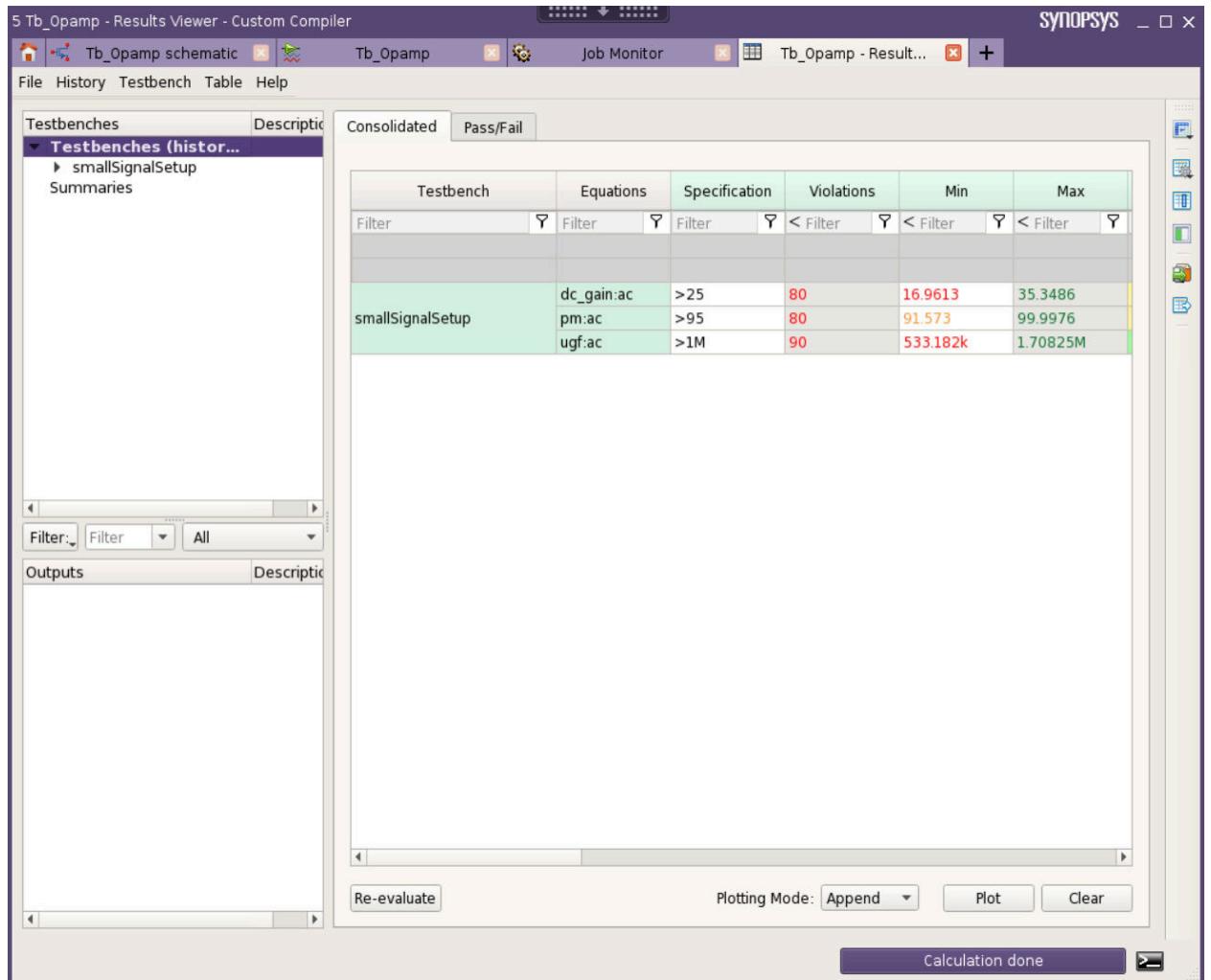
Opening the ResultsView

The ResultsView is automatically launched when a simulation job is complete.

Chapter 15: Using the ResultsView

Opening the ResultsView

- If the ResultsView is not already open, choose **Results > Viewer** from the PrimeWave Design Environment menu bar or right-click on a history point in the History tab and choose **Open ResultsView**.



ResultsView User Interface

The ResultsView consists of the following components:

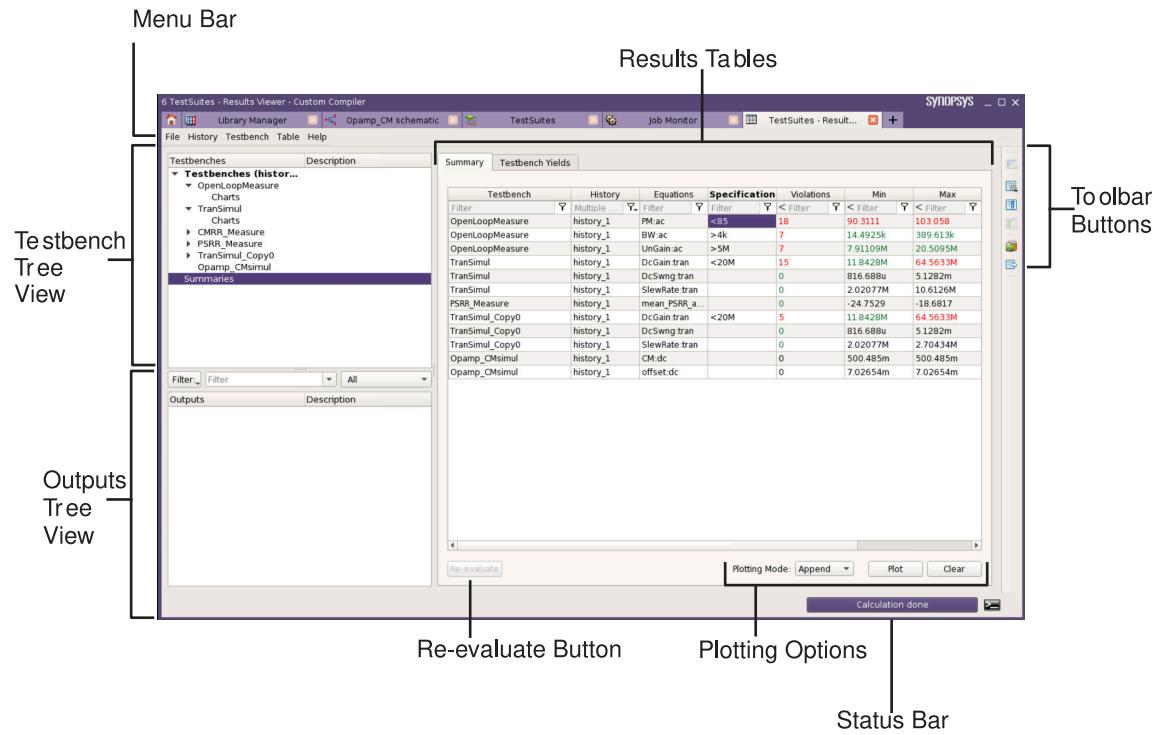


Table 9 describes the components in the ResultsView user interface.

Table 9 ResultsView Components

Component	Description
Menu Bar	Menus of commands that offer quick access to ResultsView operations. See ResultsView Menus .
Results Tables	<p>When you change the tree node selection, the results table changes to display the results corresponding to that node.</p> <ul style="list-style-type: none"> The Summaries node displays Summary and Testbench Yields tables. See Viewing Summary and Testbench Yield Results. Testbenches (<history name>) node displays Consolidated and Pass/Fail tables. See Viewing Consolidated and Pass/Fail Results. An individual testbench node selected in the tree view displays Measurements and Statistics tables. See Viewing Measurement and Statistical Results. Charts nodes display Histogram, Scatter, Q-Q, Multiple-Axis charts. See Using Charts to Visualize Simulation Results. <p>Results tables show incremental updates as new results become available with new simulation runs. See Viewing Incremental Results.</p>
Toolbar Buttons	Button commands and options. For more information, see ResultsView Toolbar Buttons .
Testbench Tree View	<p>The testbench tree view. The active history point is shown in bold.</p> <p>Upon opening, the Summaries node is selected by default.</p> <p>When you change the tree node selection, the results table changes to display the corresponding results.</p>
Outputs Tree View	<p>The tree view of the outputs corresponding to the selected node in the testbench tree view.</p> <p>When a testbench is selected on the testbench tree view, the outputs tree view shows all the outputs corresponding to that testbench.</p> <p>The default checked status of each output is the same as the Show in ResultsView status in the Outputs table on the main PrimeWave Design Environment page.</p>

Table 9 ResultsView Components (Continued)

Component	Description
Re-evaluate Button	Refreshes expressions, specifications, and/or images and updates the displayed data. You might consider re-evaluating results if you add or modify output expressions or if you modify specifications for one or more output expression. The button is enabled only when the selected history point is active and the simulation data is saved to disk. If the Save Simulation Data option is off in the Result Option dialog box (Results > Options), simulation data is not saved to disk and this option is disabled in the ResultsView table context-sensitive menu. See Specifying Results Options .
Plotting Options	Options for plotting signals. See Plotting Outputs .
Status Bar	Displays the progress of the simulation jobs.

ResultsView Menus

The following tables describe the ResultsView menus:

- [File Menu](#)
- [History Menu](#)
- [Testbench Menu](#)
- [Table Menu](#)
- [Help Menu](#)

Table 10 File Menu

Menu	Command	Description
File	Export Data to CSV	Exports the testbench results (the active view) to a .csv file. For more information, see Exporting Results to a CSV File .
	Create Datasheet	Opens the Create Datasheet dialog box. For more information, see Exporting Results to an HTML Datasheet .
	Export Options	Opens the Export Options dialog box. See Setting Export Options .

Table 10 File Menu (Continued)

Menu	Command	Description
	Rerun All Failed	Reruns all failed iterations. See Rerunning Iterations .
	Close	Closes the ResultsView.

Table 11 History Menu

Menu	Command	Description
History	 Close History	Closes the history point selected in the Testbenches tree view.

Table 12 Testbench Menu

Menu	Command	Description
Testbench	 Plot Outputs	Opens the waveform viewer and plots the outputs. See Plotting Results From the ResultsView .
	Launch Terminal	Opens a new Linux terminal in the current results directory. See Launching the Terminal From a Result .

Table 13 Table Menu

Menu	Command	Description
Table	Select Items for Column Header	Allows you to choose items to include in the results table column headers. Options include: <ul style="list-style-type: none"> • Corner Name (on by default) • Results

Table 13 Table Menu (Continued)

Menu	Command	Description
Table Configuration		<p>Provides options for configuring items that are displayed results table. Options here vary depending on the table type.</p> <ul style="list-style-type: none"> • Filters (on by default). Displays the Filter menu at the top of each column. See Filtering Column Results. • Min/Max (on by default). • Spec/Violations (on by default). • Scalar Equations • Wave Equations • Signals • Only Errors • Only Violations • Only Errors • Parameters
Show/Hide Table Columns		Opens the Show/Hide Table Column dialog box. See Showing and Hiding Columns in the Results Table .

Table 14 Help Menu

Menu	Command	Description
	ResultsView Help	Opens the ResultsView help in an external browser.
	Custom Compiler Help	Displays the Custom Compiler help in an external browser.
	About Custom Compiler	Opens the About Custom Compiler dialog box containing information about the Custom Compiler version number, copyright, legal notices, and Synopsys contact information.

ResultsView Toolbar Buttons

The ResultsView toolbar contains the following:

Table 15 ResultsView Toolbar Buttons

Button	Description
 Select Items for Column Header	Allows you to choose items to include in the results table column headers. Options here vary depending on the table type. See Selecting Items for Column Headers .

Table 15 ResultsView Toolbar Buttons (Continued)

Button	Description
	Provides options for configuring items that are displayed in the results table. Options here vary depending on the table type. <ul style="list-style-type: none">• Filters (on by default). Displays the Filter menu at the top of each column. See Filtering Column Results.• Min/Max (on by default).• Spec/Violations (on by default).• Scalar Equations• Wave Equations• Signals• Only Errors• Only Violations• Only Errors• Parameters
	Opens the Show/Hide Table Column dialog box. See Showing and Hiding Columns in the Results Table .
	Freezes the condition columns to keep that data visible on the left side of the table as you scroll through the results. Click the button again to unfreeze the columns. Available for condition columns only, such as Corner, Corner Parameters, Sweep Variables, and Monte Carlo.
	Exports the testbench results (the active view) to a .csv file. For more information, see Exporting Results to a CSV File .
	Opens the Create Datasheet dialog box. See Exporting Results to an HTML Datasheet .

Results Tables

When you change the tree node selection, the results table changes to display the results corresponding to that node.

Table 16 Results Table Types

Tree Node Selection	Results Table Types
Summaries	Summary Testbench Yields

Table 16 Results Table Types (Continued)

Tree Node Selection	Results Table Types
Testbenches (<history name>)	Consolidated Pass/Fail
Individual testbench	Measurements Statistics

The tables provided in the ResultsView are described in the following sections:

- [Viewing Summary and Testbench Yield Results](#)
- [Viewing Measurement and Statistical Results](#)
- [Viewing Consolidated and Pass/Fail Results](#)
- [Table Menu Options](#)

Note:

You can view simulation results that include .MEASURE statements in the ResultsView.

Viewing Summary and Testbench Yield Results

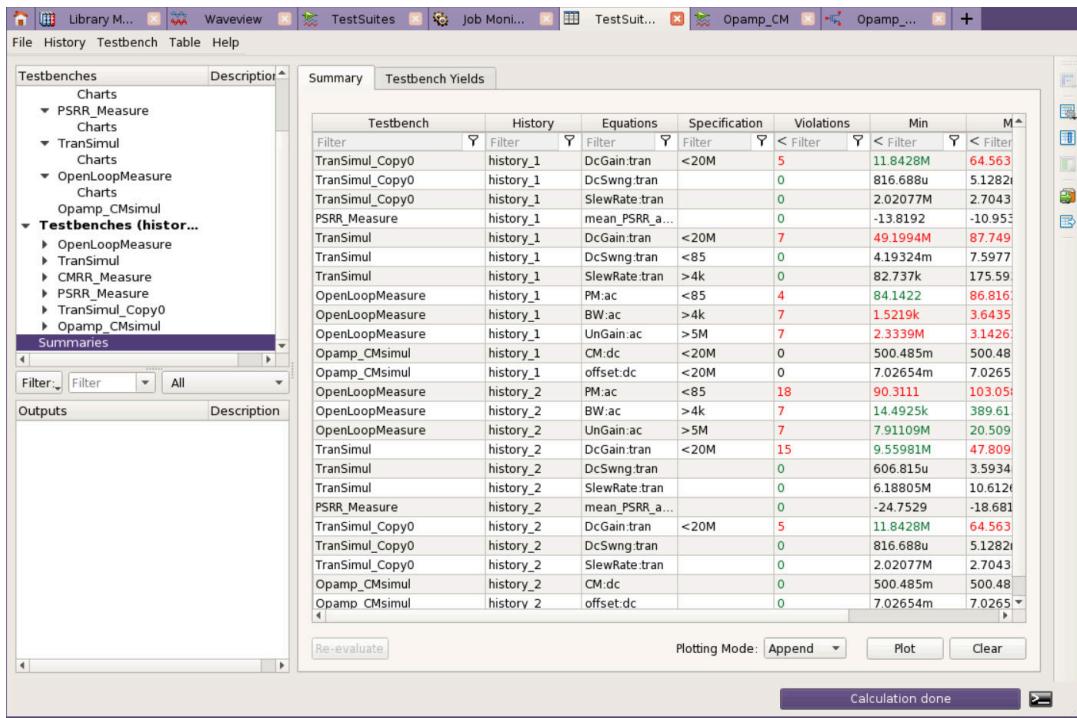
To view summary simulation results:

1. Open the ResultsView. See [Opening the ResultsView](#).

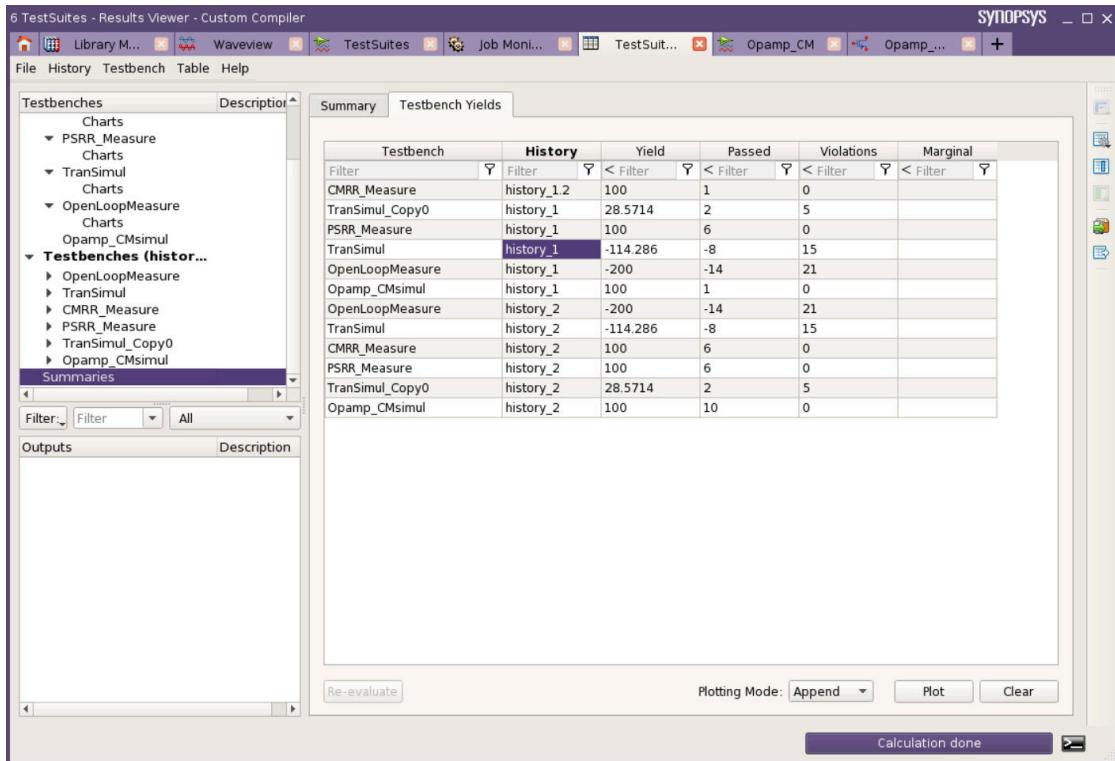
When **Summaries** is selected in the testbench tree view, the table on the right shows a **Summary** table detailing information about each testbench.

Chapter 15: Using the ResultsView

Results Tables



- Switch to the **Testbench Yields** table to view passed, failed, and marginal results.



The following table describes the testbench yields result categories.

Testbench Yields Result Category	Description
Yield	Displays all failed and passed testbench results.
Passed	Displays all passed testbench results. Passed results are colored green, and the marginally passed results are colored differently.
Violations	Displays all failed testbench results. Failed results are colored red, and marginally failed results are colored differently.
Marginal	Displays all marginally passed or failed testbench results.

Results can be exported to a .csv file or HTML report; see [Exporting ResultsView Results](#).

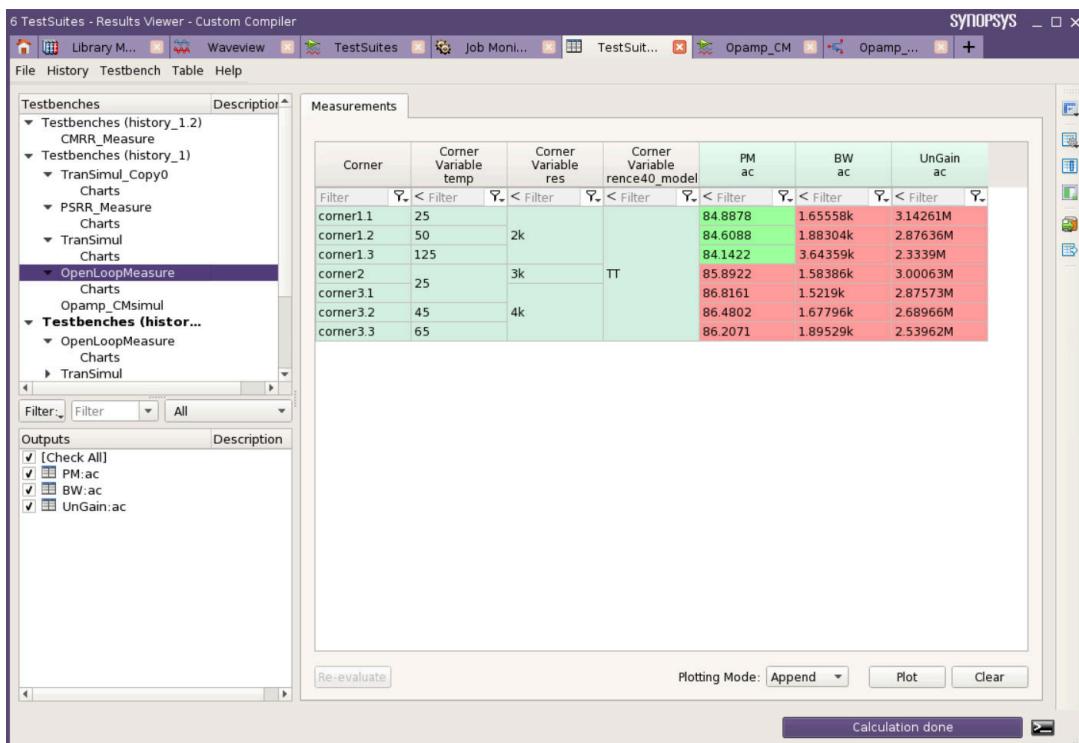
Viewing Measurement and Statistical Results

When a single testbench is selected in the testbenches view tree, the Measurements table is visible. The Measurements table details results of all measurements for each testbench.

If the testbench included Monte Carlo analysis, a Statistics table is also available. This can be particularly useful in visualizing and debugging Monte Carlo results, such as partial yields (per-measurement yields), skewness, excess kurtosis (numerical measures of Gaussian distribution), KS (Kolmogorov-Smirnov) statistic, mean +/- N sigma, and median +/- N sigma.

To view the Measurement results:

1. Open the ResultsView. See [Opening the ResultsView](#).
2. Select a single testbench in the testbenches view tree. The Measurements table appears.



3. (Optional) Click the **Filter** button to filter the testbench results based on specified filter values.

Corner	Corner Variable temp	Corner Variable res	Corner Variable vence40_mode	PS
Filter	< Filter	< Filter	< Filter	< Filter
corner1.1	25			
corner1.2	50			
corner1.3	125			
corner3.1	25			
corner3.2	45			
corner3.3	65			

Operator: <

- [Check All]
- 25
- 45
- 50
- 65
- 125

Clear Filter for "Corner Variable temp"
 Clear Filter for All Columns

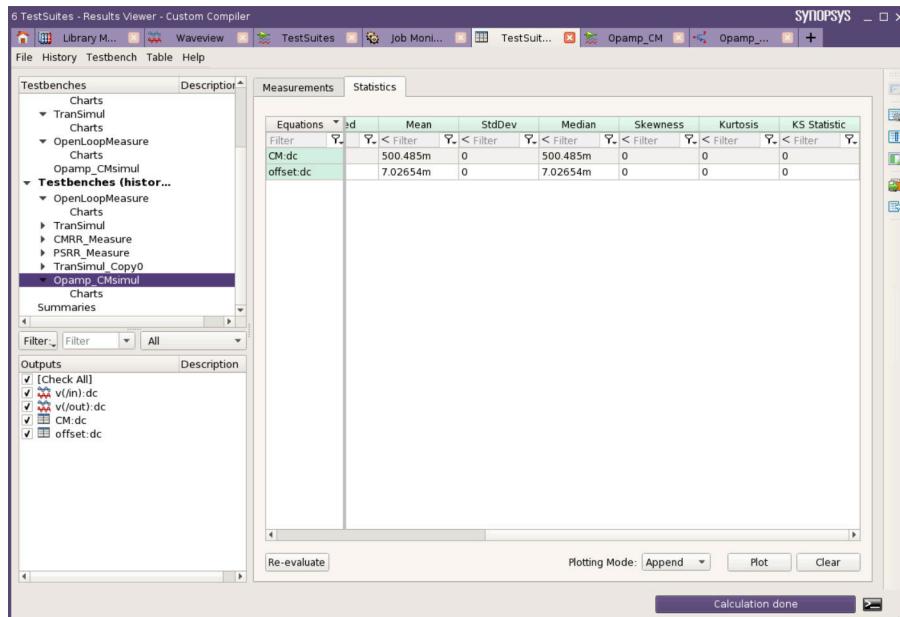
In the filter control dialog box that opens, specify the **Operator** and check the filter items. The filter is automatically applied.

For example, if a circuit is simulated under various temperatures (by setting the values for temp), then a filter setting such as temp > 50 can be defined to view only those results that satisfy this filter criteria.

Click **Clear Filter for <specific column>** or **Clear Filter for All Columns** to clear the filter.

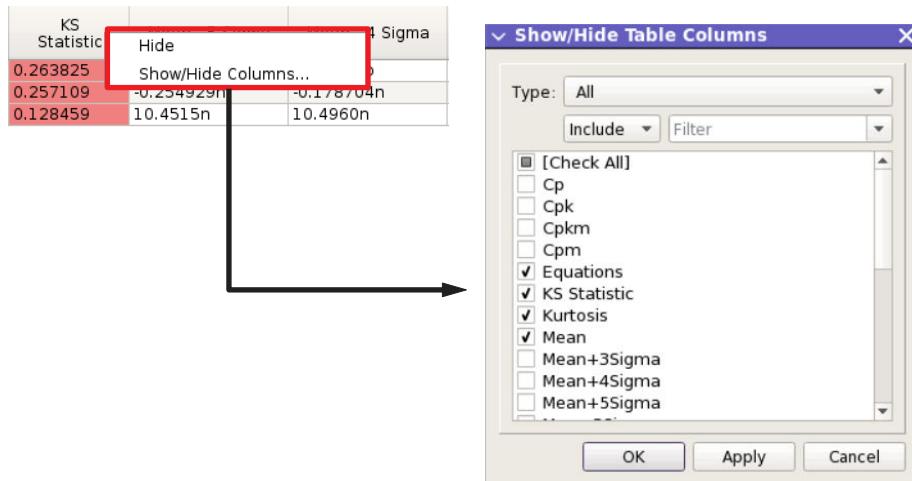
For Monte Carlo simulations, a statistical summary of results is available from the ResultsView. To view a statistical summary:

1. Select the **Statistics** tab in the results table.



Note that the condition column in the above screen capture is frozen using the **Freeze Header Columns** button, in order to keep it displayed while scrolling through the results.

2. (Optional) To show or hide columns in the Statistics table, right-click a column heading and choose **Show/Hide Columns** to open the **Show/Hide Columns** dialog box. Check the columns you wish to show and uncheck the columns you wish to hide.



3. (Optional) To hide an individual column in the Statistics table, right-click the column heading and choose **Hide**.

Viewing Consolidated and Pass/Fail Results

Consolidated tables allow you to view an individual measurement (sweep variables, corners, aging) in a more compact form than the Measurements table provides. In addition to simulations with multiple levels of advanced analysis, the consolidated table can also display the nominal case and single-level analyses, such as corner-only, Monte-Carlo-only, sweep-only, or aging-only analyses.

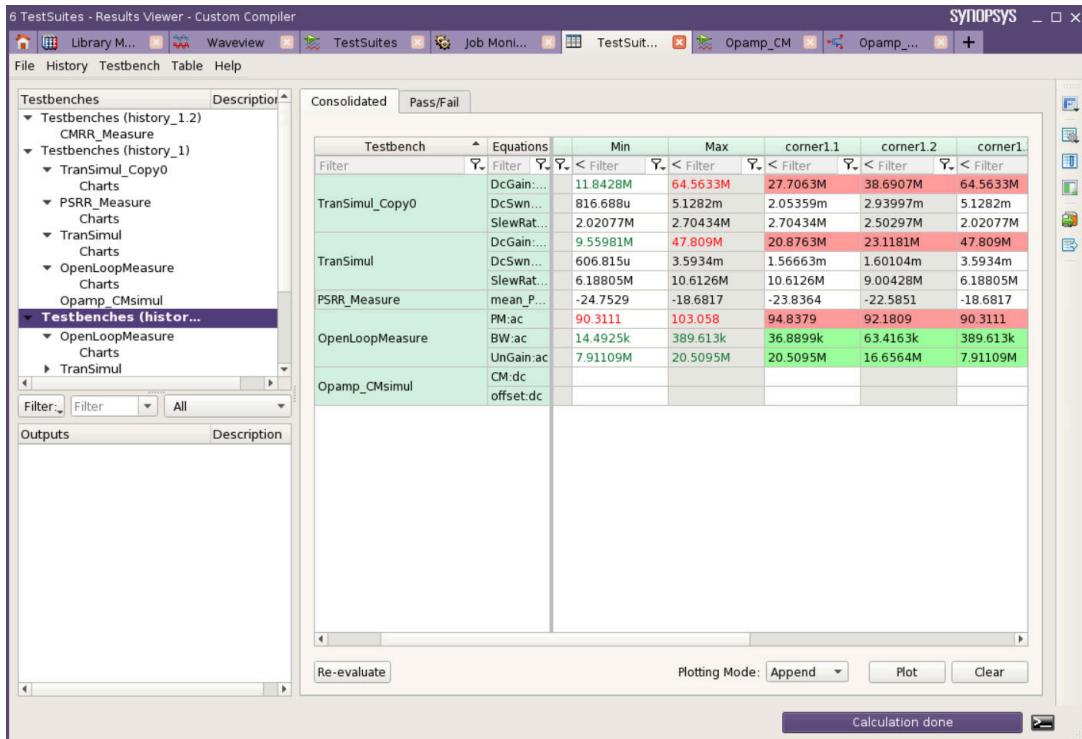
For data sheets, a consolidated table allows you to display the entire results for one measurement as a compact table as opposed to a long scrollable list.

To view results in a consolidated table:

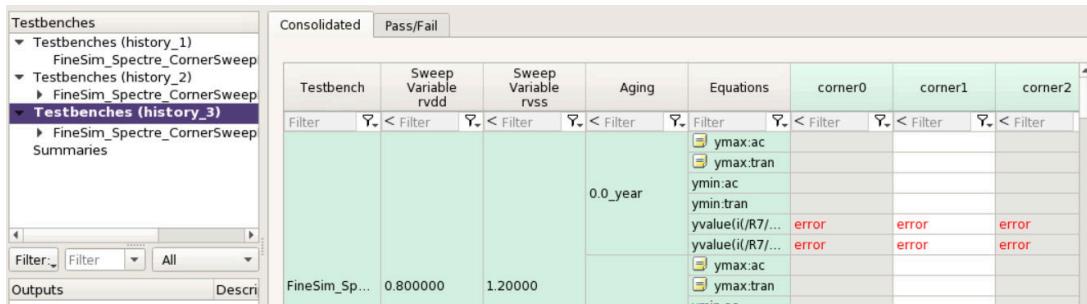
1. Open the ResultsView. See [Opening the ResultsView](#).
2. Select a history point in the testbench tree view of the ResultsView. The Consolidated table appears.

Chapter 15: Using the ResultsView

Results Tables

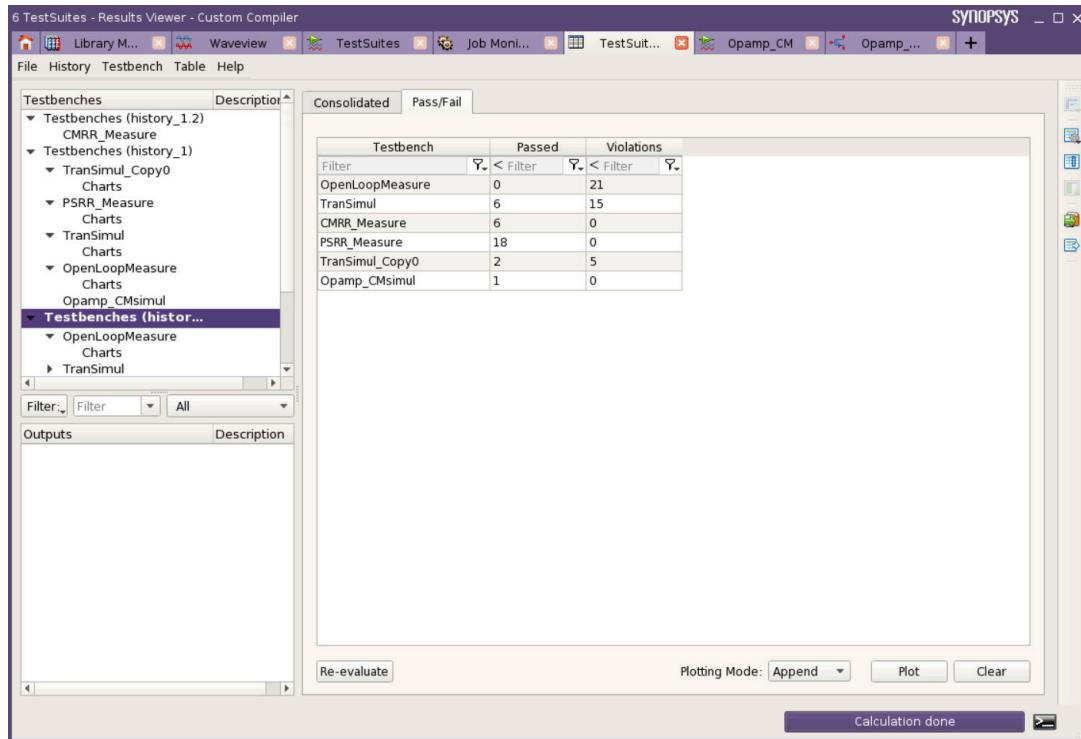


The following screen capture is an example of a consolidated table showing corner/sweep/aging results:



To view results in a Pass/Fail table:

- Select the **Pass/Fail** tab. The Pass/Fail table appears.



The following table describes the pass/fail result categories.

Pass/Fail Result Category	Description
Passed	Displays all passed testbench results.
Violations	Displays all testbench results in violation of spec.

Table Menu Options

The following table describes the context-sensitive menu options available from results tables. Note that some options are only available from certain tables.

Table 17 ResultsView Table Menu Options

Option	Description	Available in which tables?
 Plot All	Opens the waveform viewer and plots all the trace values for the selected measurements in the testbench. See Plotting Results From the ResultsView .	Consolidated, Measurements
 Plot	Opens the waveform viewer and plots the trace values under the selected conditions. See Plotting Results From the ResultsView .	Consolidated, Measurements
 Plot Across	Allows you to plot across Corner/Aging/MONTE_CARLO/Sweep Variables under the selected conditions. See Plotting Results From the ResultsView .	Consolidated, Measurements
 Plot Histogram	Plots a histogram for the selected equations under the selected condition, but across all Monte Carlo traces. See Plotting Results From the ResultsView .	Measurements (with Monte Carlo results)
Show Results Analyzer	Opens the Results Analyzer. See Using the Results Analyzer .	Consolidated, Measurements
Display Netlist	Opens the netlist in a text viewer. If the Save Netlists option is off in the Result Options dialog box (Results > Options), netlist data is not saved to disk and this option is disabled in the ResultsView table menu. See Displaying Netlists .	Measurements
Display Output Log	Opens the output log in a text viewer. See Displaying Output Logs .	Measurements
Launch Terminal	Opens a new Linux terminal in the current results directory. See Launching the Terminal From a Result .	Measurements
New Session	Opens a new PrimeWave Design Environment session in a new tab. See Opening New Sessions .	Measurements
Print	See Printing and Annotating Node Voltages and Operating Points .	Consolidated, Measurements
Plot Signal	Opens the design so that you can interactively select a signal to plot. (Available data types, depending on the analyses included in the simulation you ran, include Transient Signal, AC Bode, AC Magnitude, DC, Input Noise, and Output Noise. See Plotting Signals Interactively .)	Consolidated, Measurements
Annotate	See Printing and Annotating Node Voltages and Operating Points .	Consolidated, Measurements
Rerun	Reruns the simulation for the selected iteration. See Rerunning Iterations .	Consolidated, Measurements

Table 17 ResultsView Table Menu Options (Continued)

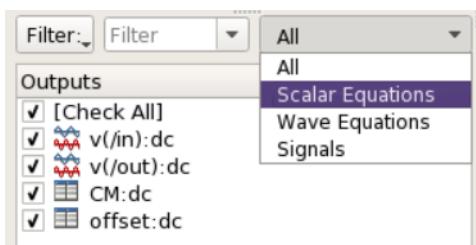
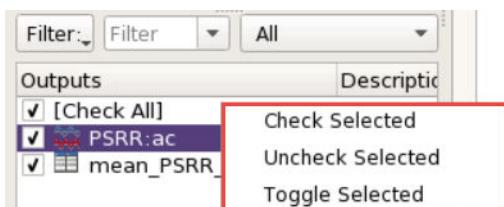
Option	Description	Available in which tables?
Default Row Order	Resets the table configuration to the default row order. See Reordering Table Results .	All
Default Column Order	Resets the table configuration to the default column order. See Reordering Table Results .	All
Highlight Selected Rows	Highlights the selected rows. See Highlighting Whole Rows and Columns .	All
Highlight Selected Columns	Highlights the selected columns. See Highlighting Whole Rows and Columns .	All

Outputs Tree View

The outputs tree view shows all the outputs corresponding to the selected node in the testbench tree view.

You can expand the outputs tree view to show the data (conditions, measurements, and so forth) that are part of that testbench, as well as control visibility of testbench-specific data.

The default enabled/disabled status of each output is the same as the **Show in ResultsView** status in the **Outputs** table on the main PrimeWave Design Environment page. You can enable or disable multiple measurements at once by selecting the measurement, right-clicking, and choosing **Check Selected**, **Uncheck Selected**, or **Toggle Selected** from the menu that appears.

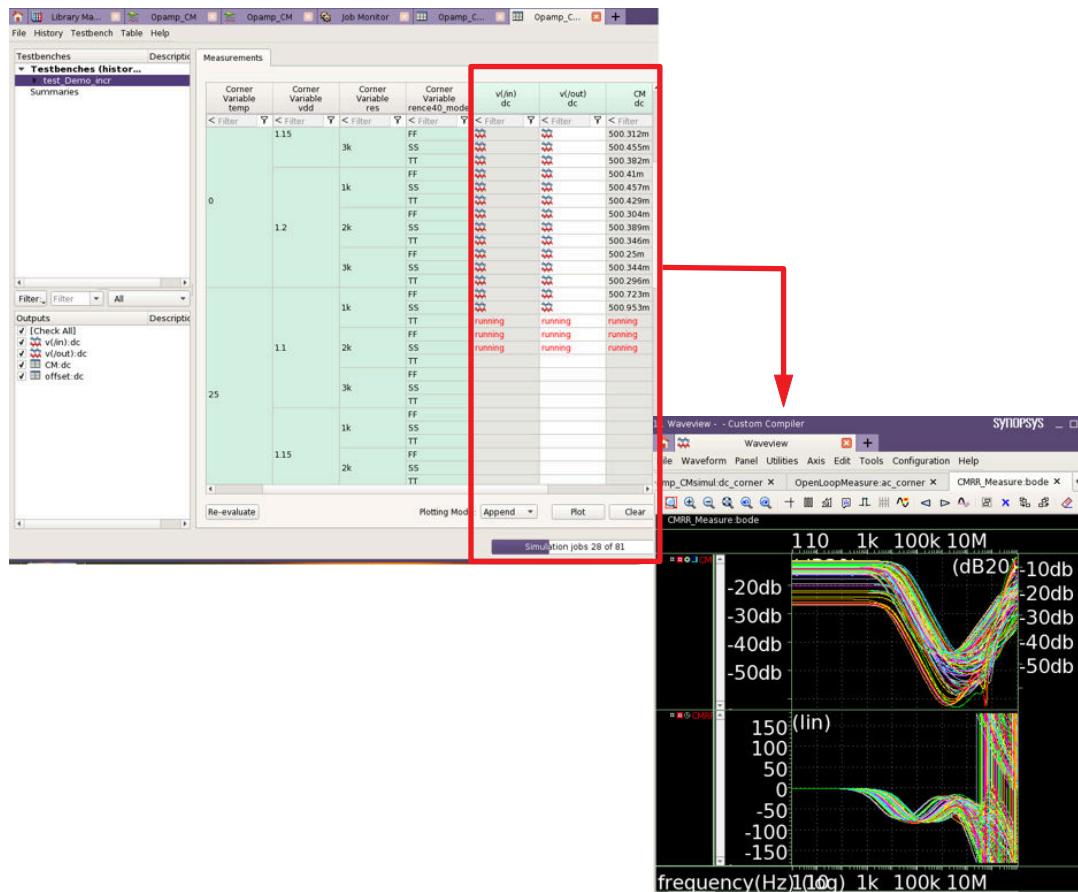


Viewing Incremental Results

The ResultsView incrementally displays the results for each iteration run.

To view incremental results:

1. Select a testbench to netlist and simulate from the testbenches tree view.
2. Notice the results table is updated in real time. Additionally, the status bar indicates the job status, and the waveforms are updated in the waveform viewer.



Plotting Results From the ResultsView

The following results types are plotted in the waveform viewer:

- waveforms

The waveform is plotted.

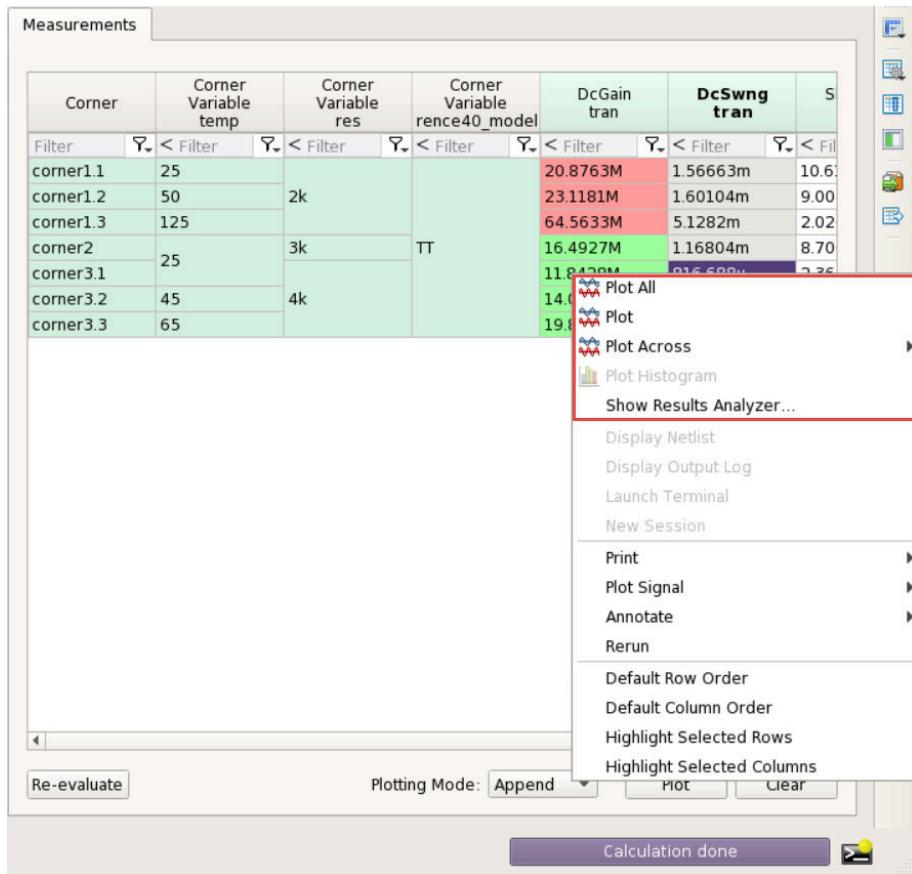
Chapter 15: Using the ResultsView

Plotting Results From the ResultsView

- scalar values

When scalar values include waveforms, the input waveforms are plotted. (Scalar measurements and their values run in corner analysis result in waveforms, whereas those run in Monte Carlo analysis result in histograms.)

To plot a result in the ResultsView table, right-click a result and choose **Plot All**, **Plot Across**, **Plot**, or **Plot Histogram** from the menu. Plotted data can come from a single testbench or from multiple testbenches. You can target specific waveforms from specific iterations if needed.



The following sections describe the various plotting options.

- [Plot All](#)
- [Plot](#)
- [Plot Across](#)
- [Plot Histogram](#)

Plot All

The **Plot All** option plots all the trace values for the selected measurements in the testbench.

The case shown in the following screen capture plots `v(VOUT): tran` and `ymin:tran`.

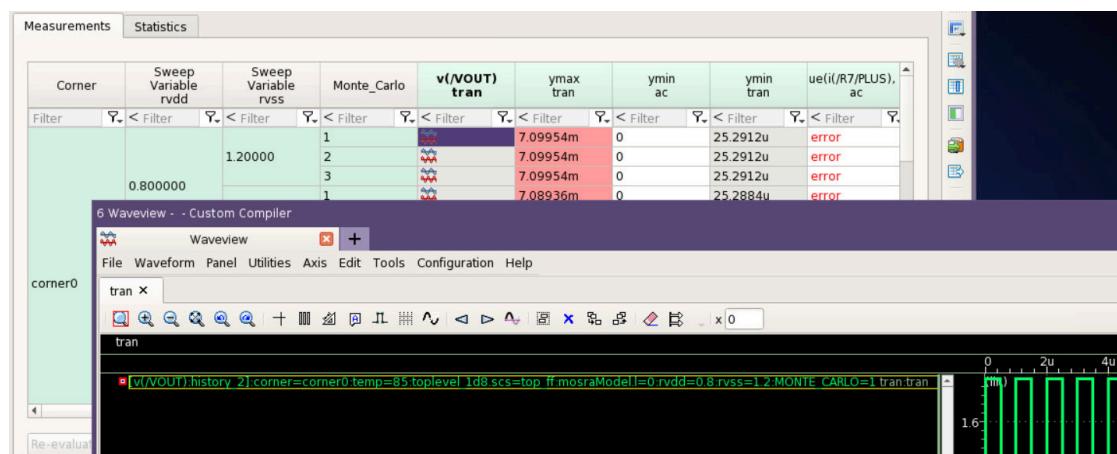


Plot

The **Plot** option plots the trace values under the selected conditions.

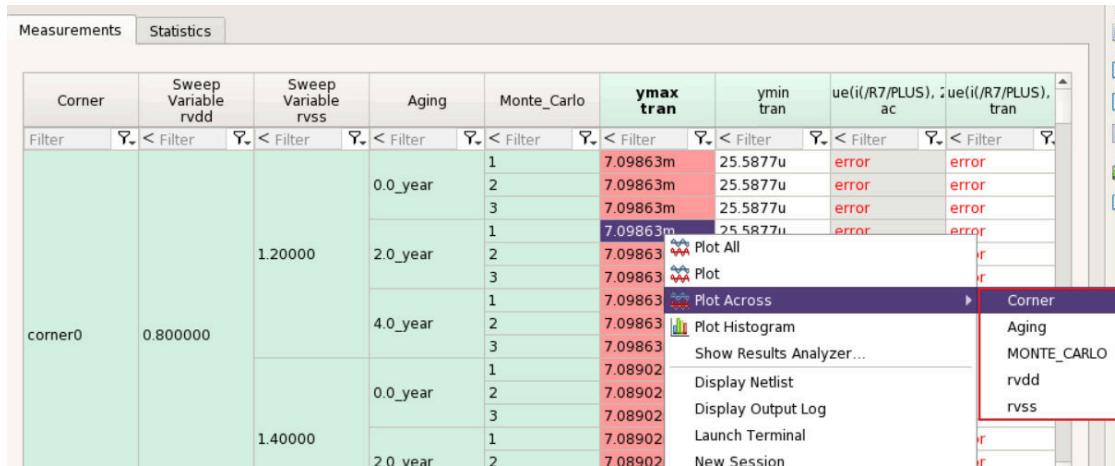
For the case shown in the following screen capture, the selected conditions are:

`corner=corner0, sweep rvdd=0.8, sweep rvss=1.2, monte_carlo=1.`



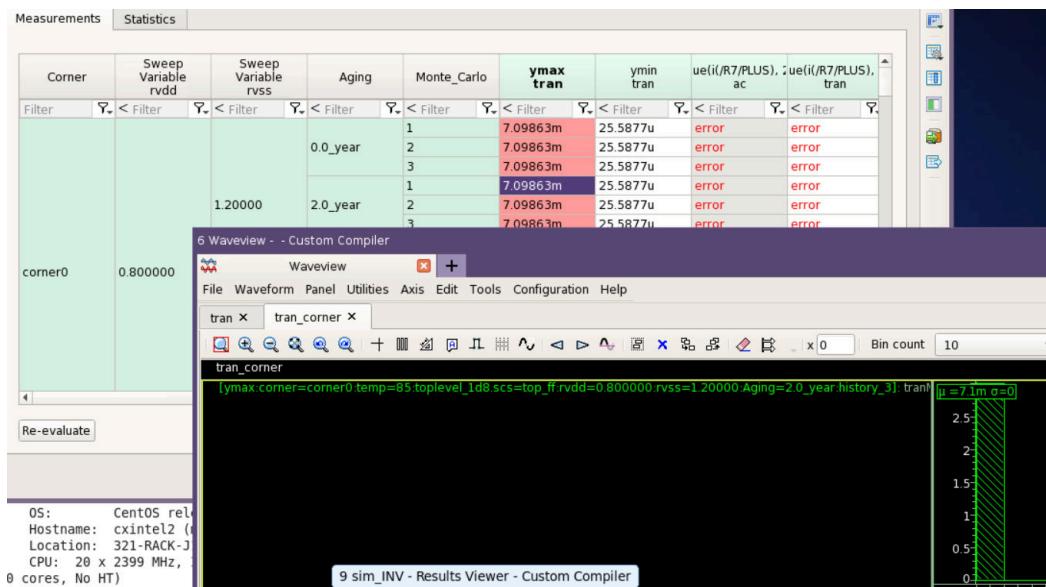
Plot Across

The **Plot Across** option plots across Corner/Aging/MONTE_CARLO/Sweep Variables under the selected conditions.



Plot Histogram

The **Plot Histogram** option plots a histogram for the selected equations under the selected condition, but across all Monte Carlo traces.



Displaying Netlists

To display the netlist for a result in the ResultsView, right-click a result and choose **Display Netlist** from the menu that opens. The netlist is displayed in the Text Viewer, which opens in a new tab.

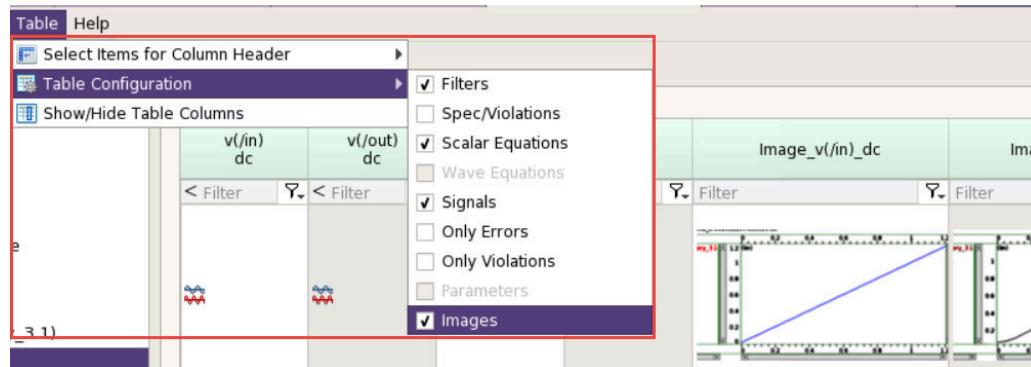
If the **Save Netlists** option is off in the **Result Option** dialog box (**Results > Options**), netlist data is not saved to disk and the **Display Netlist** option is disabled in the ResultsView table menu.

Displaying Output Logs

To display the output log for a result in the ResultsView, right-click a result and choose **Display Output Log** from the menu that opens. The output log is displayed in the Text Viewer, which opens in a new tab.

Displaying Images

To display images in the results tables for table types that support images, choose **Table > Table Configuration** and select the **Images** option.



Launching the Terminal From a Result

To launch a terminal from a result in the ResultsView, right-click a result and choose **Launch Terminal** from the menu that opens. The terminal is opened.

Each iteration or corner is run in a separate folder. When you choose **Launch Terminal**, a terminal opens in the folder that contains the iteration you initially right-clicked.

Rerunning Iterations

To rerun an iteration in the ResultsView, right-click a result and choose **Rerun** from the menu that opens. The simulation for the iteration you right-clicked is rerun.

To rerun all failed iterations, choose **File > Rerun All Failed** from the ResultsView menu bar.

Opening New Sessions

To open a new session from the ResultsView, which can be useful when you want to debug a specific iteration that failed in a separate session, right-click a result and choose **New Session** from the menu that opens. A new session is created.

New sessions are configured with the parameters of the iteration you right-clicked. (The parameters include corner setup, design variables, and outputs.)

Exporting ResultsView Results

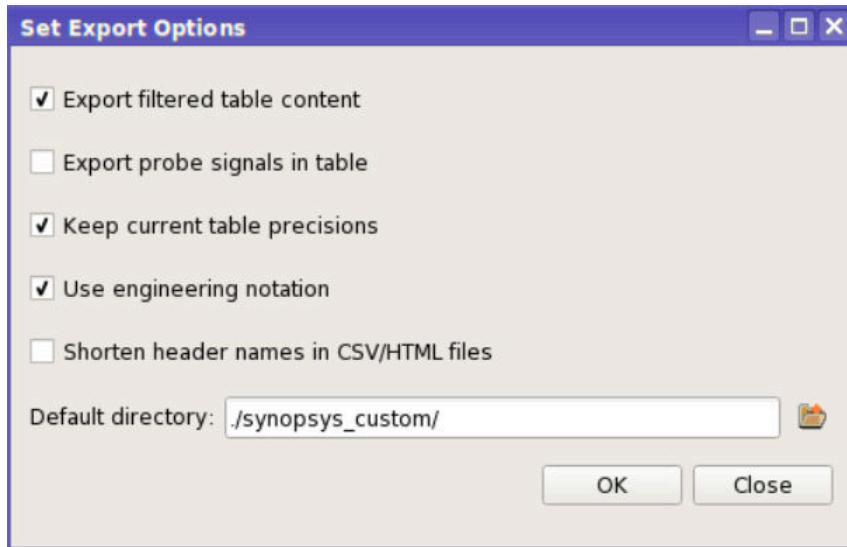
The following sections describe methods of exporting results and setting up export options:

- [Setting Export Options](#)
- [Exporting Results to a CSV File](#)
- [Exporting Results to an HTML Datasheet](#)

You can also copy the results to the clipboard and paste them directly into a file, such as an Excel spreadsheet.

Setting Export Options

Choose **File > Export Options** to adjust export options in the **Set Export Options** dialog box.

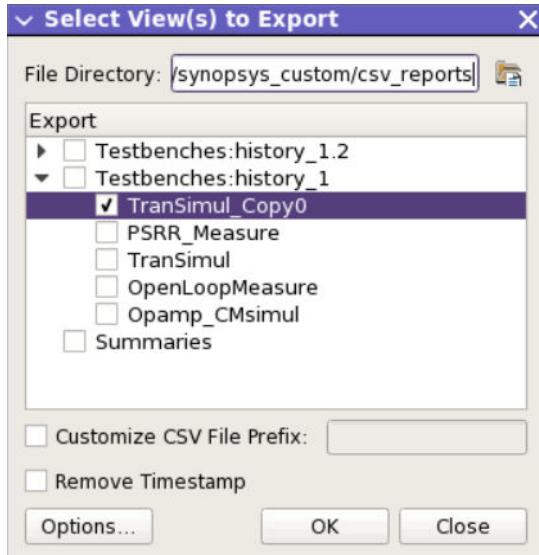


The following table describes the export options.

Export Option	Description
Export filtered table content	When enabled (the default), only exports the results from the visible table cells. When disabled, exports all the results from the table cells, including hidden columns and rows.
Export probe signals in table	When enabled, exports probe signals. Off by default.
Keep current table precisions	When enabled (the default), exports the values with precision settings you have selected. When disabled, exports the values with original complete precision setting.
Use engineering notation	When enabled (the default), exports scalar values with engineering notation. When disabled, exports scalar values with scientific notation.
Shorten headers name in CSV/HTML files	When enabled, uses shorter header names in exported files. Off by default.
Default directory	The default export directory.

Exporting Results to a CSV File

To export the testbench results (the active view) to a .csv file, choose **File > Export Data to CSV** from the ResultsView window menu. The **Select View(s) to Export** window opens.



Click the check box next to each view you want to export. Individual .csv files are exported for each view you select.

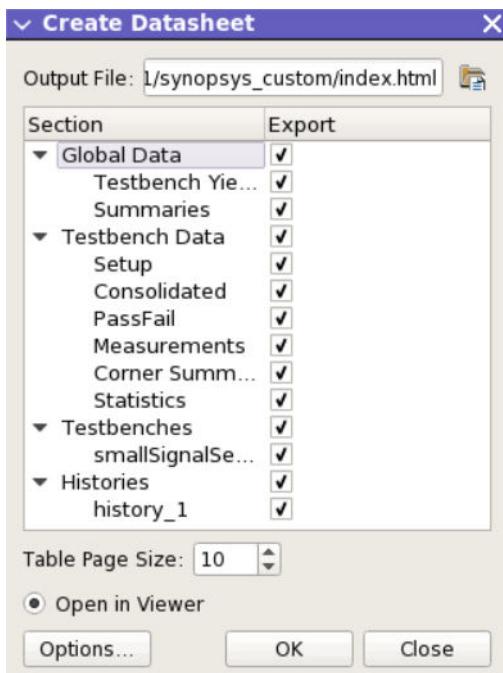
- The default directory name in the **Select View(s) to Export** dialog box is csv_reports. You can edit this directory name.
- The generated filenames are of the format `View_name_tree_Date_Timestamp.csv`. This format is not editable. Sample output names include:
 - Summaries_20151105_18:50:58.29.csv
 - Testbenches_locktest_20151105_18:50:58.29.csv
- To add a prefix to the filenames, select **Customize CSV File Prefix** and enter a filename prefix in the text box.
- Select **Remove Timestamp** to remove the timestamp.
- Click **Options** to open the **Set Export Options** dialog box. See [Setting Export Options](#).

Exporting Results to an HTML Datasheet

To export results to an HTML datasheet:

1. Choose **File > Create Datasheet** from the ResultsView window menu.

The **Create Datasheet** dialog box opens.



2. If you want a different location than the default location already selected, enter the path in the **Output File** text box where you want the HTML file exported.
3. Choose the sections you want to include in the datasheet by clicking the check box(es) in the **Export** column.

All testbench results that simulated successfully are included by default. You can expand the view tree by clicking on the icon, and then enable or disable sections as needed. Simulation results that did not simulate successfully are displayed in gray text and cannot be included in an HTML report.

If there are images in the results table (use **Table > Table Configuration > Images**), they are also exported to the HTML report.

4. Enter the number of records you want to view at a time in report tables by adjusting the **Table Page Size** option.

10 records are displayed in tables at a time by default. Multiple pages are created when more than this number of results are produced.

Chapter 15: Using the ResultsView Selecting Items for Column Headers

5. (Optional) Enable **Open in Viewer** to open the HTML datasheet in a Custom Compiler web viewer window.
 6. Click **Options** to open the **Set Export Options** dialog box. See [Setting Export Options](#).
 7. Click **OK** to export results to an HTML datasheet.

Several HTML files are exported along with other supporting files. Open the `index.html` file in your browser to view your HTML report.

Note:

If you have filters defined, the exported results also adhere to those filters.

Selecting Items for Column Headers

For Measurement and Consolidated tables, you can choose to rotate tables in order to display results differently. Items you can select as column headers vary depending on the result type.

By default, results tables are arranged according to Equation Name:

The following figure shows the results table rotated by Corner Name:

Chapter 15: Using the ResultsView Selecting Items for Column Headers

The following figure shows the results table rotated by Aging Time:

Corner	Sweep Variable rvdd	Sweep Variable rvss	Monte_Carlo	Equations	0.0_year	2.0_year	4.0_year
Filter	Y< Filter	Y< Filter	Y< Filter	Filter	Y< Filter	Y< Filter	Y< Filter
corner0	0.800000	1.200000		v(V/OUT).ac			
				v(V/OUT).tran			
				ymax.tran	7.09863m	7.09863m	7.09863m
				ymin.tran	25.5877u	25.5877u	25.5877u
				yvalue(i(R7...))	error	error	error
				v(VIN_N).ac			
				v(VIN_N).tran			
				vdifff.ac			
				vdifff.tran			
				v(V/OUT).ac			
				v(V/OUT).tran			
				ymin.tran	7.09863m	7.09863m	7.09863m
				yvalue(i(R7...))	error	error	error
				v(VIN_N).ac			
				v(VIN_N).tran			
				vdifff.ac			
				vdifff.tran			

The following figure shows the results table rotated by a sweep variable:

Corner	Sweep Variable rvss	Aging	Monte_Carlo	Equations	0.800000	1.00000	1.20000
Filter	Y< Filter	Y< Filter	Y< Filter	Filter	Y< Filter	Y< Filter	Y< Filter
corner0	1.200000	0.0_year		v(V/OUT).ac			
				v(V/OUT).tran			
				ymin.tran	7.09863m	7.09894m	7.0789m
				yvalue(i(R7...))	error	error	error
				v(VIN_N).ac			
				v(VIN_N).tran			
				vdifff.ac			
				vdifff.tran			

The following figure shows the results table rotated by Monte Carlo:

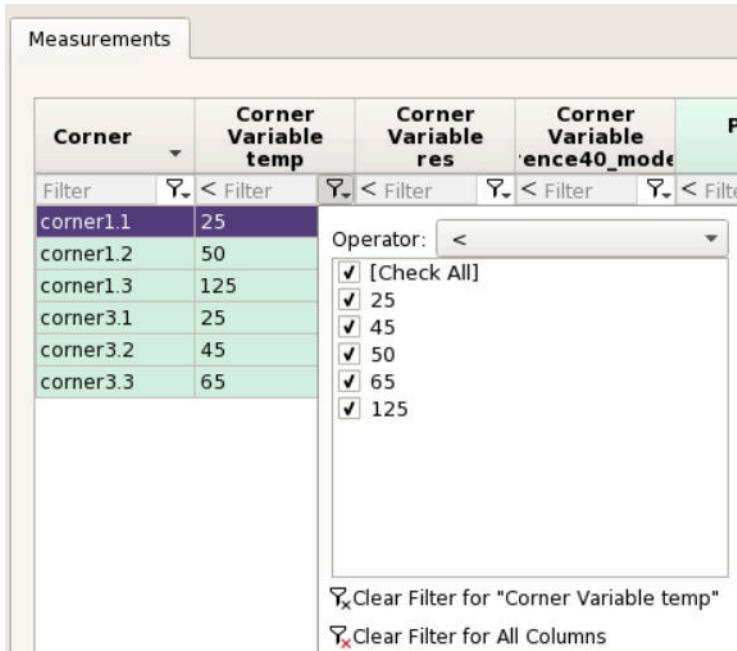
Corner	Corner Variable temp	Corner Variable oplevel 1d8 scs	Equations	1	2	3
Filter	Y< Filter	Y< Filter	Y< Filter	Y< Filter	Y< Filter	Y< Filter
corner0	85	top_ff	v(V/OUT).ac			
			v(V/OUT).tran			
			ymin.tran	7.09952m	7.09952m	7.09952m
			yvalue(i(R7...))	error	error	error
			yvalue(i(R7...))	error	error	error
			vdifff.ac			
			vdifff.tran			
corner1	90	top_fs	v(V/OUT).ac			
			v(V/OUT).tran			
			ymin.tran	6.14714m	6.14714m	6.14714m
			yvalue(i(R7...))	error	error	error
			yvalue(i(R7...))	error	error	error
			vdifff.ac			

The following figure shows the results table rotated by Results:

Corner	Sweep Variable rvdd	Sweep Variable rvss	Aging	Monte_Carlo	Equations	Results
Filter	Y< Filter	Y< Filter	Y< Filter	Y< Filter	Filter	Y< Filter
corner0	0.800000	1.200000	0.0_year		v(V/OUT).ac	
					v(V/OUT).tran	
					ymin.tran	7.09863m
					yvalue(i(R7...))	error
					v(VIN_N).ac	
					v(VIN_N).tran	
					vdifff.ac	

Filtering Column Results

You can filter column data by clicking the **Filter** button in the column header to filter the testbench results based on specified filter values.



In the filter control dialog box that opens, specify the **Operator** and select filtering parameters. The filter is automatically applied.

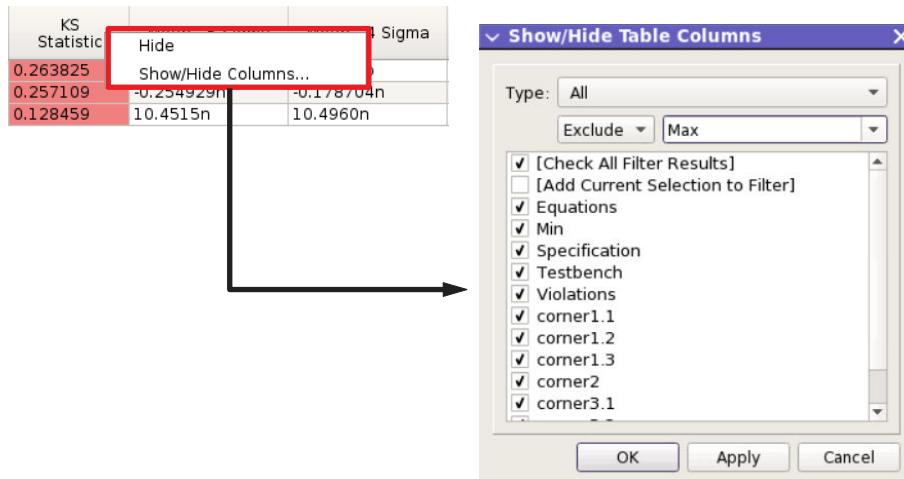
For example, if a circuit is simulated under various temperatures (by setting the values for temp), then a filter setting such as temp > 50 can be defined to view only those results that satisfy this filter criteria.

Click **Clear Filter for <specific column>** or **Clear Filter for All Columns** to clear the filter.

Showing and Hiding Columns in the Results Table

To show or hide columns in the results table, right-click a column heading and choose **Show/Hide Columns** to open the **Show/Hide Table Columns** dialog box. (Alternatively,

choose **Table > Show/Hide Table Columns.**) Check the columns you wish to show and uncheck the columns you wish to hide.



To hide an individual column in the **Statistics** tab, right-click the column heading and choose **Hide**.

You can control the type of columns that appear in the **Show/Hide Table Columns** checklist using the drop-down **Type** menu.

You can exclude or include specific columns by selecting **Exclude** or **Include** and typing the column name in the **Filter** textbox.

Reordering Table Results

To reorder the results displayed in the ResultsView results table by column, click and drag a column heading to the desired location.

Table columns with a sort arrow in the header can be sorted by row in ascending or descending order.

To return all the columns to the default order, right-click any result in the results table and choose **Default Column Order** from the menu that opens.

To return all the rows to the default order, right-click any result in the results table and choose **Default Row Order** from the menu that opens.

When you reorder columns and close the ResultsView, your column order is automatically saved and loaded the next time you open those results in the ResultsView.

Highlighting Whole Rows and Columns

To highlight a whole row or column, right-click a result in the ResultsView results table, and choose **Highlight Selected Rows** or **Highlight Selected Columns** from the menu that opens. The whole row or column where the result located is selected, respectively.

Adding Descriptions

Descriptions can be added to test suites, testbenches, history points, output expressions, or design variables in order to provide supplemental information about the various fields or short names.

Descriptions are visible in the ResultsView in the **Description** column of the testbench tree or as tooltips that appear when hovering over the object in the summary table.

To add descriptions:

1. In the main PrimeWave Design Environment window, select the object you wish to describe: test suite (top row of the tree or global scope), testbench, output expression, or variable.

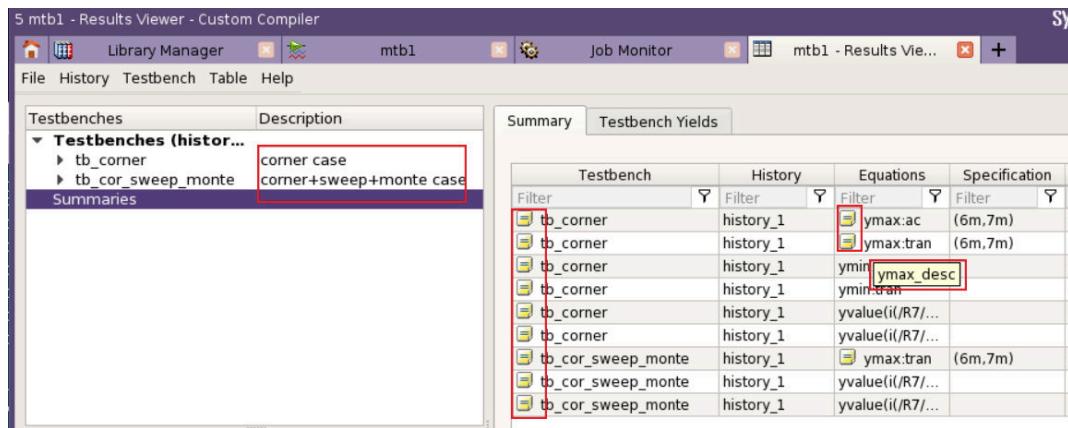
2. Click **Edit Description**  on the right edge of the window. The **Edit Description** dialog box appears.
3. Type your description in the **Edit Description** dialog box.



Note:

For test suites and testbenches, descriptions can also be added during the creation phase (in the **Comment** field of the **Create Test Suite** dialog box or the **Description** field of the **Create Testbench** dialog box).

4. Click **OK**. After running the simulation, the description you entered appears in the ResultsView (outlined in red below).



16

Using Charts to Visualize Simulation Results

This chapter contains information on how to use charts in the PrimeWave Design Environment's ResultsView to visualize simulation results.

The chart tools use the results database to compare advanced analyses (corners, sweeps, and Monte Carlo) graphically, providing an additional level of data mining that allows you to view all the data in one view. Charts allow you to analyze the statistical data representing circuit performance and the correlation of different parameters with each other.

The charts available from the ResultsView and some common chart usages are described in the following sections:

- [Chart Options](#)
- [Scatter Charts](#)
- [Multiple-Axis Charts](#)
- [Histogram Charts](#)
- [Q-Q Charts](#)
- [Zooming In and Out](#)
- [Editing Chart Properties](#)
- [Enabling Data Point Information Balloons](#)
- [Specifying Parametric Reduction From Charts](#)

Chart Options

The following sections describe common chart options.

- [Chart Toolbar Buttons](#)
- [Chart Menu Options](#)

Chart Toolbar Buttons

The following table describes the toolbar options available from results tables. Note that some options are only available from certain tables.

Table 18 Chart Toolbar Buttons and their Actions

Button	Tooltip	Action	Available in which chart?
	Parametric Reduction	Opens the Parametric Reduction dialog box. For details, see Specifying Parametric Reduction From Charts .	Histogram, Q-Q
	Tile Horizontal	Tiles the thumbnails in the chart gallery horizontally.	Histogram, Scatter, Q-Q
	Auto Tile to fit	Tiles the thumbnails in the chart gallery to fit the window.	Histogram, Scatter, Q-Q
	Tile 1x1	Tiles the thumbnails in the chart gallery in one column.	Histogram, Scatter, Q-Q
	Tile 2x1	Tiles the thumbnails in the chart gallery in two columns.	Histogram, Scatter, Q-Q
	Custom Tile	Opens the User Tile dialog box, in which you can choose the number of Columns and Rows by which to tile the thumbnails in the chart gallery.	Histogram, Scatter, Q-Q
	Center Data with the Median	Arranges the axis scaling so that the median value of each of the axes is aligned.	Multiple-axis
(None)	Bin count	Controls histogram binning by user-defined number of bins (size is calculated to fit the data).	Histogram

Chart Menu Options

The following table describes the context-sensitive menu options available from results tables. Note that some options are only available from certain tables.

Table 19 Chart Menu Options

Option	Action	Available in which chart?
Copy Image	Copies the chart on the active page to the clipboard. All	
Save Image	Exports the chart on the active page to an image file.	All
Chart Properties	Opens the Chart Properties dialog box. For details, see Editing Chart Properties .	All
Plot	Plots the selected chart in the waveform viewer.	Histogram, Scatter

Scatter Charts

Scatter charts can be helpful to view a graphic representation of all iterations of a specific measurement against a specification. Visualizing these results can help to identify specific trouble areas, review distributions, and view other statistical data.

To create a scatter chart:

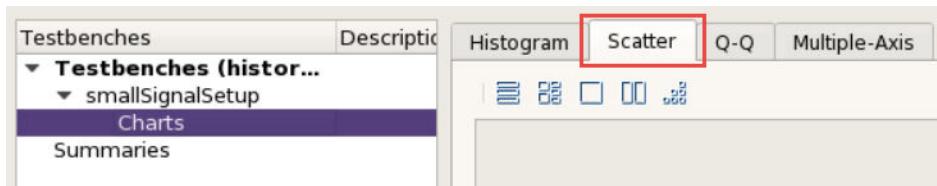
1. Choose **Results > Viewer** from the PrimeWave Design Environment main menu.

The ResultsView opens in a new tab.

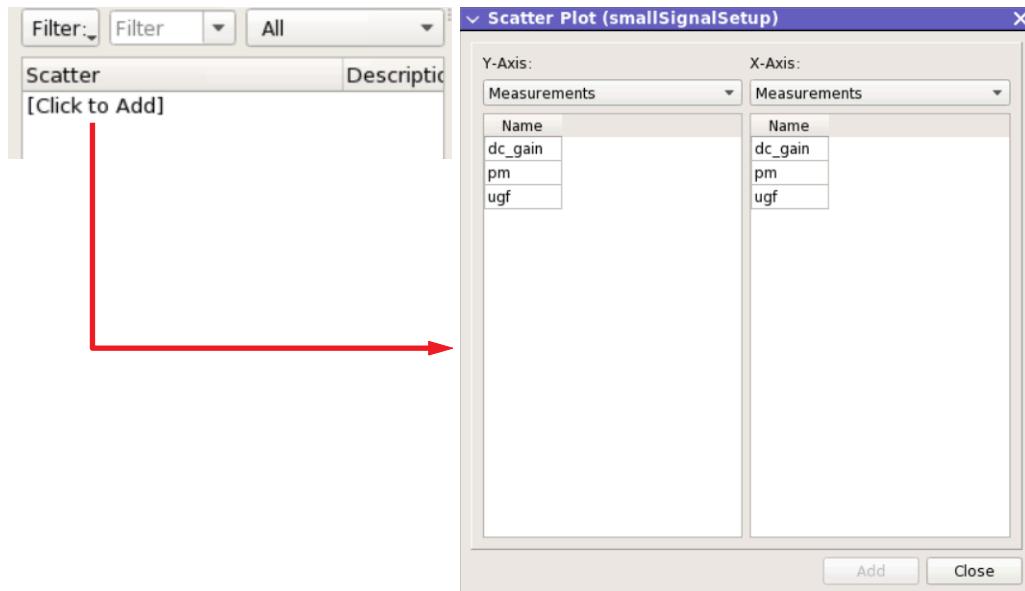
2. Select a testbench in the Testbenches tree view and expand the node to see the **Charts** option.



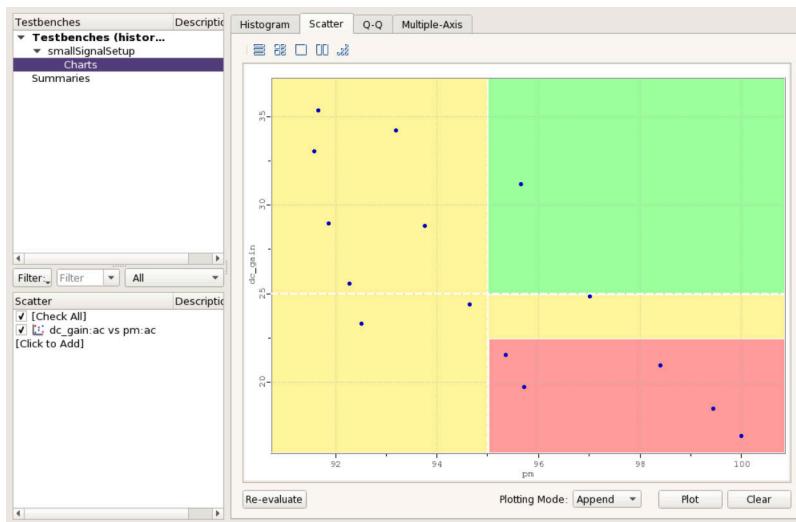
3. Select the **Charts** option and notice the results table view area now provides several tabs for various charts. Select the **Scatter** tab.



4. In the Outputs tree view, click [Click to Add]. The **Scatter Plot** setup dialog box for the selected testbench appears.



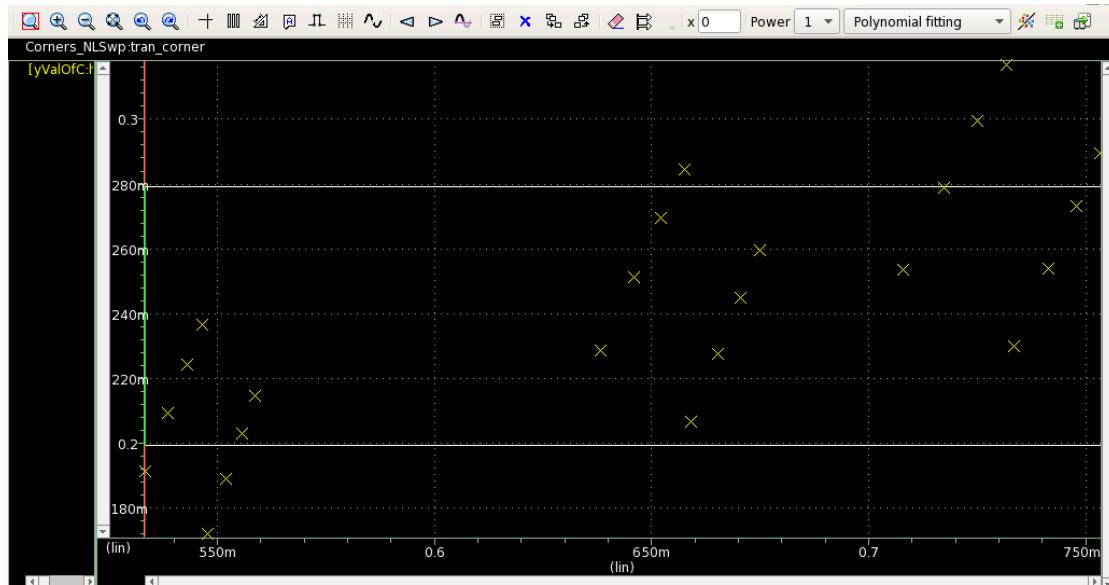
5. In the **Scatter Plot** dialog box, select the measurements you wish to plot. Click **Add** to add the measurements to the Outputs tree view and create the scatter plot. Click **Close** to close the **Scatter Plot** dialog box.



In the above figure, data points that are displayed within the green shaded area of a chart are within specification. Data points that are displayed within light green and orange shaded areas are the marginal areas. They are defined by the values in the

Marginal columns of the PrimeWave Design Environment Outputs/Specifications table. Data points that are displayed within red areas are outside of specification.

6. (Optional) Use the toolbar buttons to manipulate the look of the active chart displayed. See [Chart Toolbar Buttons](#).
7. (Optional) Plot the selected chart in WaveView by right-clicking and selecting **Plot**.



In WaveView, data points that are displayed within the green section of the y-axis are within specification. They are defined by the values in the Range columns of the PrimeWave Design Environment Output/Specifications table. Data points that are displayed within the red section are outside of specification.

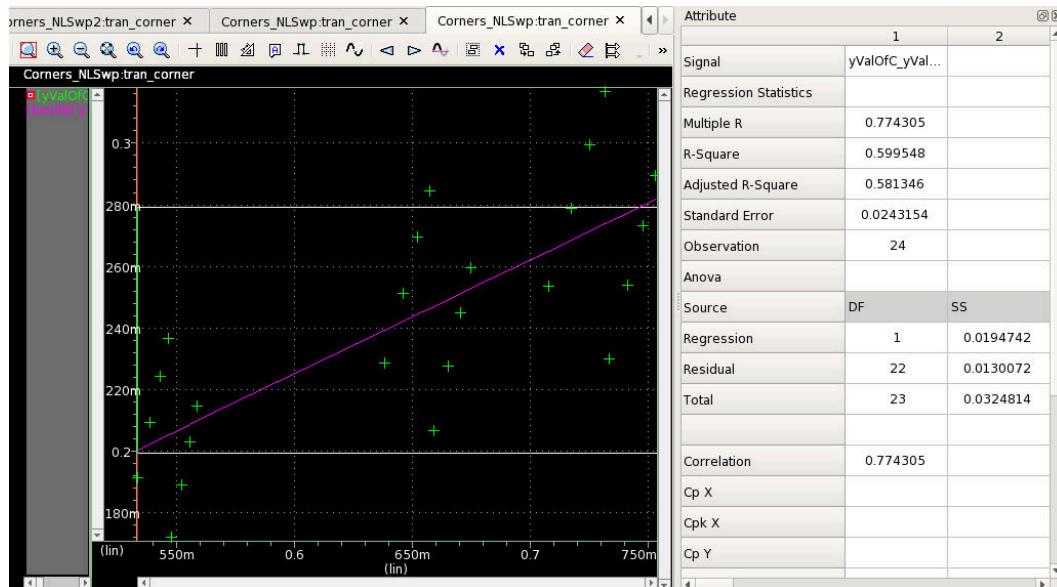
8. Use the toolbar controls in WaveView to configure the way the waveform results are displayed.

Table 20 WaveView Toolbar Buttons for Scatter Charts

Button	Option	Action
Power <input type="button" value="1"/>	Power	Applies to the regression fit curve. This is the order of the polynomial equation that can be used to fit the data. A power of 1 is a linear fit, power of 2 is a 2nd-order polynomial, and so forth. This setting is only applicable when the Fitting option is set to Polynomial.
Polynomial fitting	Line Fitting	Line fitting options include Polynomial, Exponential, Logarithmical, and Ratio polynom.
	Regression	Shows/hides the red regression fit line.

Button	Option	Action
	Show Statistics Info	Displays the statistics information table for the active scatter plot to the right of the WaveView window.
	Export Info Table	Exports the information table for the scatter chart in the active window to a text file.

9. (Optional) In WaveView, show the regression fit line by clicking the **Regression** button. The regression fit line appears in WaveView.
10. (Optional) In WaveView, show the statistics information table by clicking the **Show Statistics Info** button. The statistics information table appears to the right of the WaveView window.



11. (Optional) In the ResultsView, save the scatter chart image by right-clicking and selecting **Save Image**.

Multiple-Axis Charts

A multiple-axis chart can help you to identify and trace key parameters and then determine options to improve yield, center the design, or identify overly sensitive parameters. These charts can help provide key insight when making design trade-offs.

To create a multiple-axis chart:

1. Choose **Results > Viewer** from the PrimeWave Design Environment main menu.

The ResultsView opens in a new tab.

2. Select a testbench in the Testbenches tree view and expand the node to see the **Charts** option.



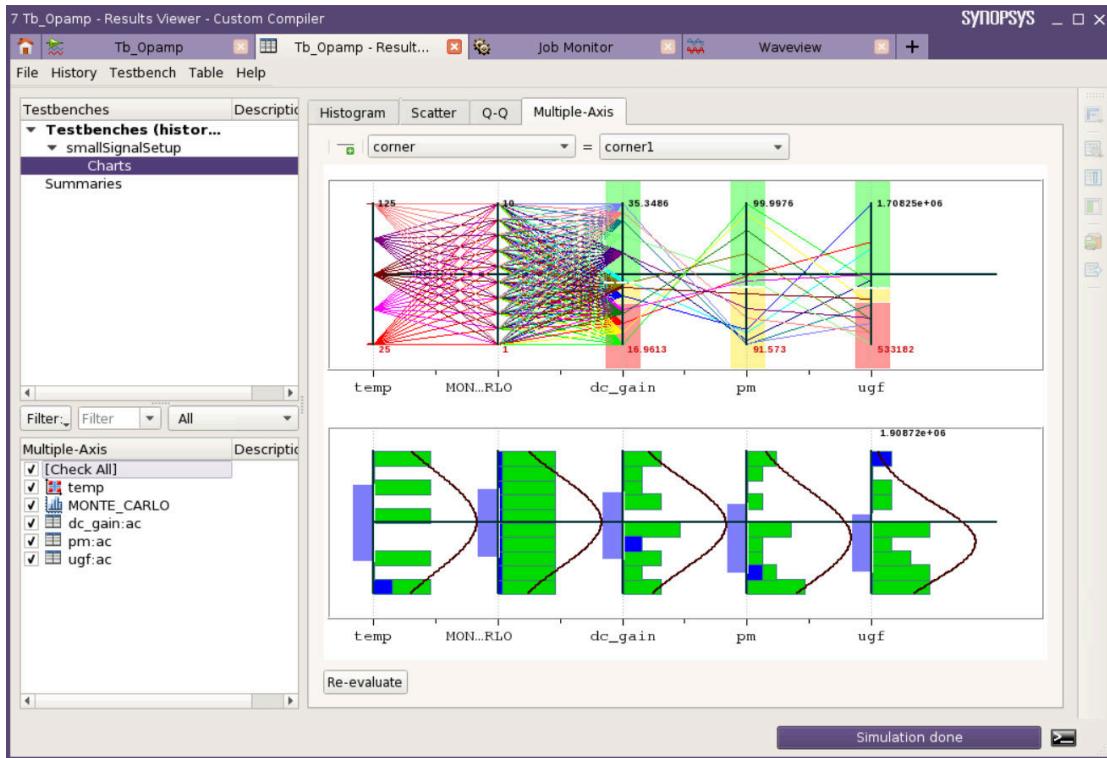
3. Select the **Charts** option and notice the results table view area now provides several tabs for various charts. Select the **Multiple-Axis** tab.



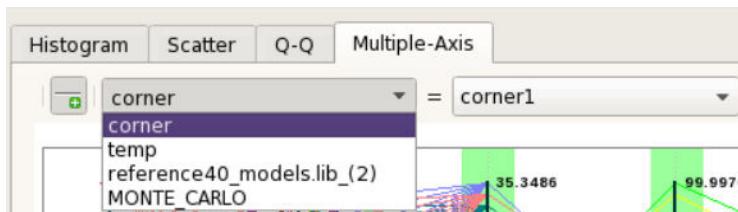
The multiple axis chart is displayed in the Chart window.

Chapter 16: Using Charts to Visualize Simulation Results

Multiple-Axis Charts



4. (Optional) Click the **Center data with the median** button to arrange the axis scaling so that the median value of each of the axes is aligned.
5. (Optional) Modify the primary design variable and its corresponding value using the **Group by variable** and **Select active segment** controls.



6. (Optional) Use the toolbar buttons to manipulate the look of the active chart displayed. See [Chart Toolbar Buttons](#).
7. (Optional) Save the multiple-axis chart image by right-clicking and selecting **Save Image**.

Histogram Charts

Histograms show the distribution of the data. They are useful for finding where the majority of that data set's values fall. Most often histograms are used with Monte Carlo simulation results to show whether the data matches a normal distribution. You can see how much of the data values lie in the acceptable range, and how many values fall outside of the acceptable range.

To create a histogram chart:

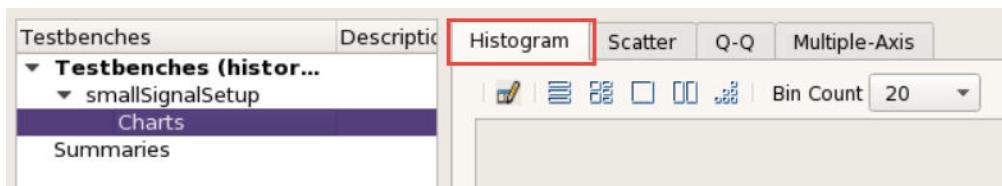
1. Choose **Results > Viewer** from the PrimeWave Design Environment main menu.

The ResultsView opens in a new tab.

2. Select a testbench in the Testbenches tree view and expand the node to see the **Charts** option.



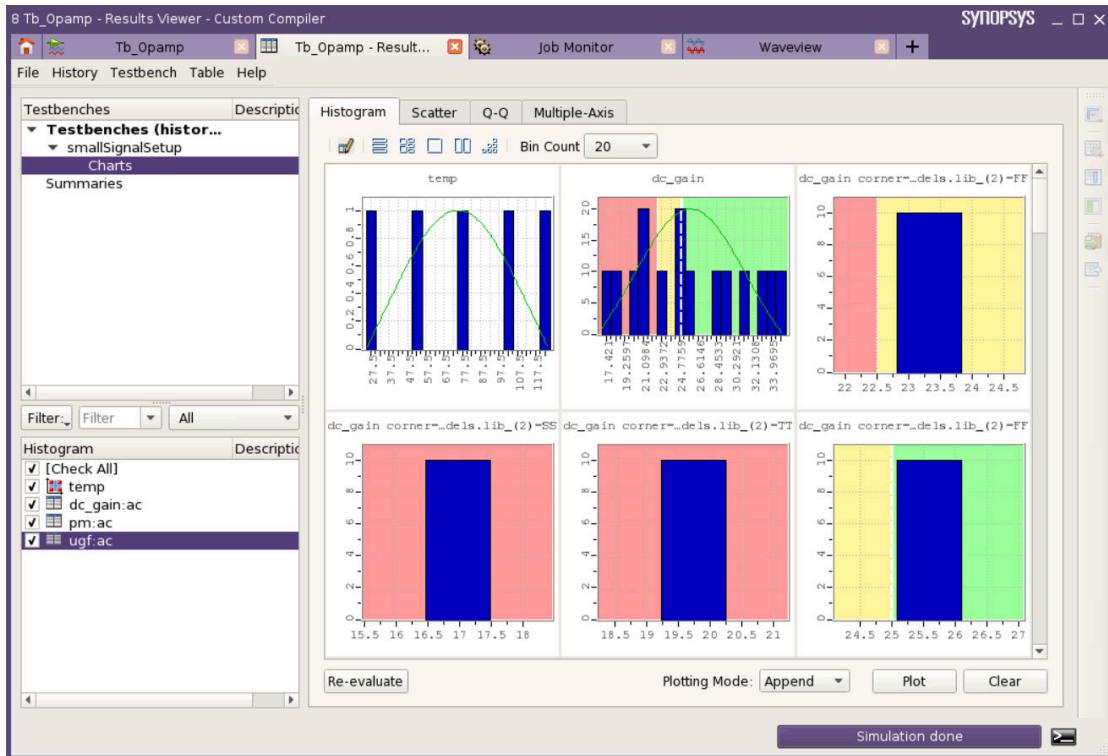
3. Select the **Charts** option and notice the results table view area now provides several tabs for various charts. Select the **Histogram** tab.



4. In the Outputs tree view, select the histogram parameters you wish to chart. The histogram chart is displayed in the Chart window.

Chapter 16: Using Charts to Visualize Simulation Results

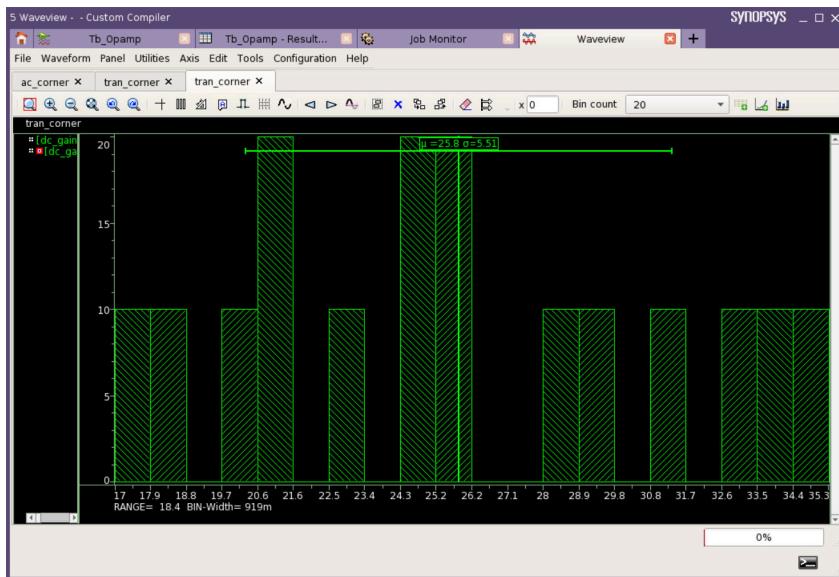
Histogram Charts



5. If you perform iterative analysis (such as sweeps or corners) in addition to the Monte Carlo analysis, then when using histograms you must first perform a parametric reduction. Click **Parametric Reduction** to open the **Parametric Reduction** dialog box (see [Specifying Parametric Reduction From Charts](#)) and select a subset of the available data for which to view histograms. Click **OK** to save the parametric reduction options.
6. (Optional) Use the toolbar buttons to manipulate the look of the active chart displayed. See [Chart Toolbar Buttons](#).
7. (Optional) Plot the selected chart in the waveform viewer by right-clicking and selecting **Plot**.

Chapter 16: Using Charts to Visualize Simulation Results

Histogram Charts

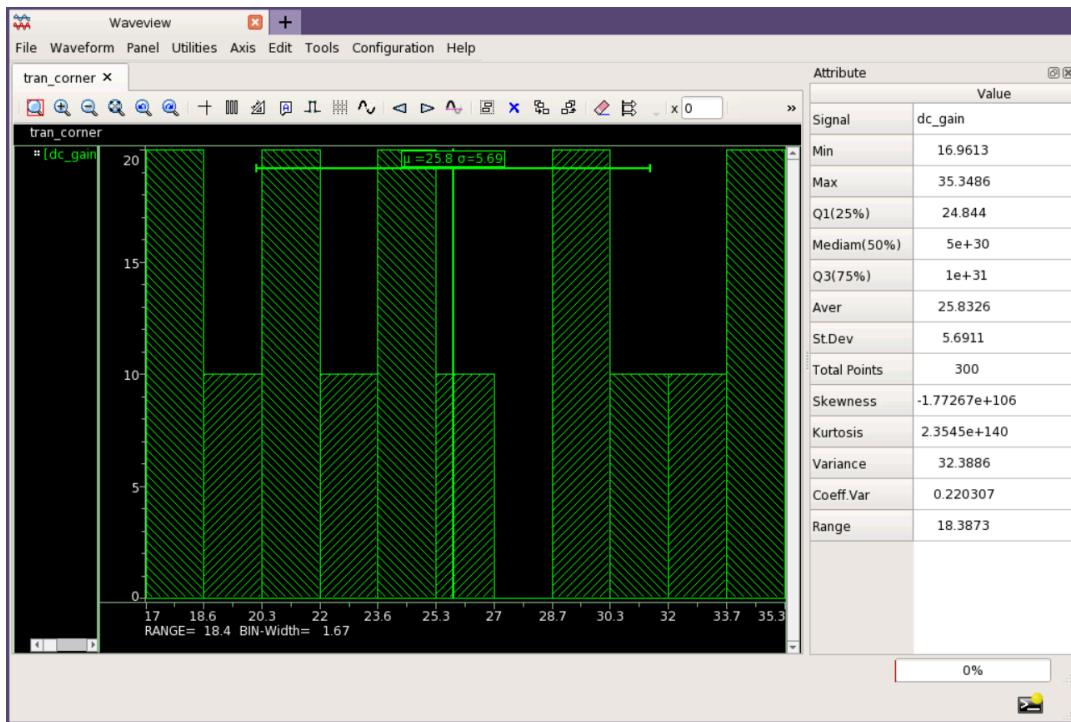


8. Use the toolbar controls in WaveView to configure the way the waveform results are displayed.

Table 21 WaveView Toolbar Buttons for Histogram Charts

Button	Option	Action
	Bin size/Bin count	Controls histogram binning either by user-defined bin size (number of bins is calculated to fit the data) or user-defined number of bins (size is calculated to fit the data).
	Show Statistics Info	Displays the statistics information table for the active histogram to the right of the chart window.
	Show Plot	Displays and hides the regression fit line. This line can be useful to determine how closely the data matches a normal distribution.
	Set tick center	Sets bin labels at the tick center.

9. (Optional) In WaveView, show the statistics information table by clicking the **Show Statistics Info** button. The statistics information table appears to the right of the WaveView window.



- (Optional) In the ResultsView, save the histogram chart image by right-clicking and selecting **Save Image**.

Q-Q Charts

A Q-Q (quantile-quantile) chart is a graphical technique for determining whether two data sets come from populations with a common distribution. It allows you to visualize any data anomalies, and is particularly useful for debugging Monte Carlo results. The y-axis plots the standard normal quantiles, and the x-axis plots the (sorted) measurement results. The chart shows a straight line relationship if measurements have a true Gaussian distribution.

To create a Q-Q chart:

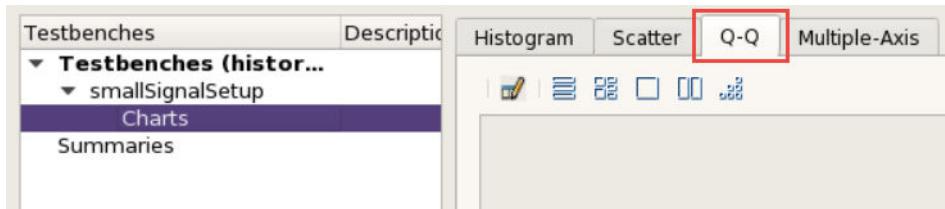
- Choose **Results > Viewer** from the PrimeWave Design Environment main menu.
The ResultsView opens in a new tab.
- Select a testbench in the Testbenches tree view and expand the node to see the **Charts** option.

Chapter 16: Using Charts to Visualize Simulation Results

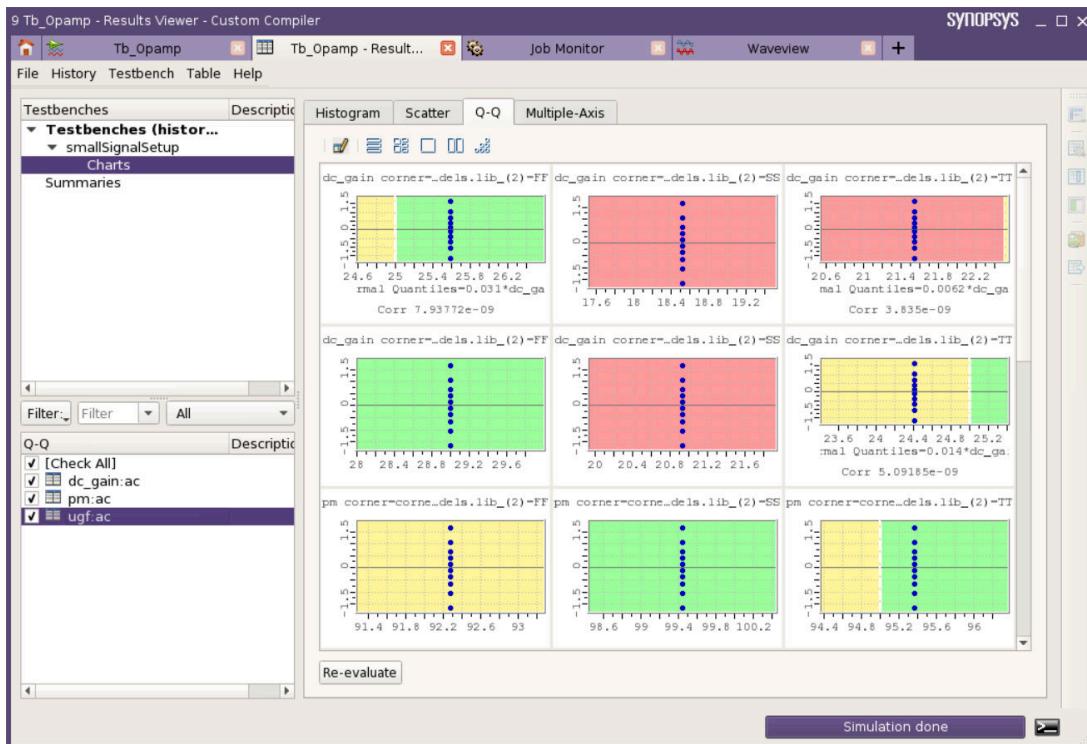
Q-Q Charts



3. Select the **Charts** option and notice the results table view area now provides several tabs for various charts. Select the **Q-Q** tab.



4. In the Outputs tree view, select the Q-Q parameters you wish to chart. The Q-Q chart is displayed in the Chart window.



5. If you perform iterative analysis (such as sweeps or corners) in addition to the Monte Carlo analysis, then when using Q-Q plots you must first perform a parametric

reduction. Click **Parametric Reduction**  to open the **Parametric Reduction** dialog box (see [Specifying Parametric Reduction From Charts](#)) and select a subset of the available data for which to view histograms. Click **OK** to save the parametric reduction options.

6. (Optional) Use the toolbar buttons to manipulate the look of the active chart displayed and select charted parameters. See [Chart Toolbar Buttons](#).
7. (Optional) In the ResultsView, save the Q-Q chart image by right-clicking and selecting **Save Image**.

Zooming In and Out

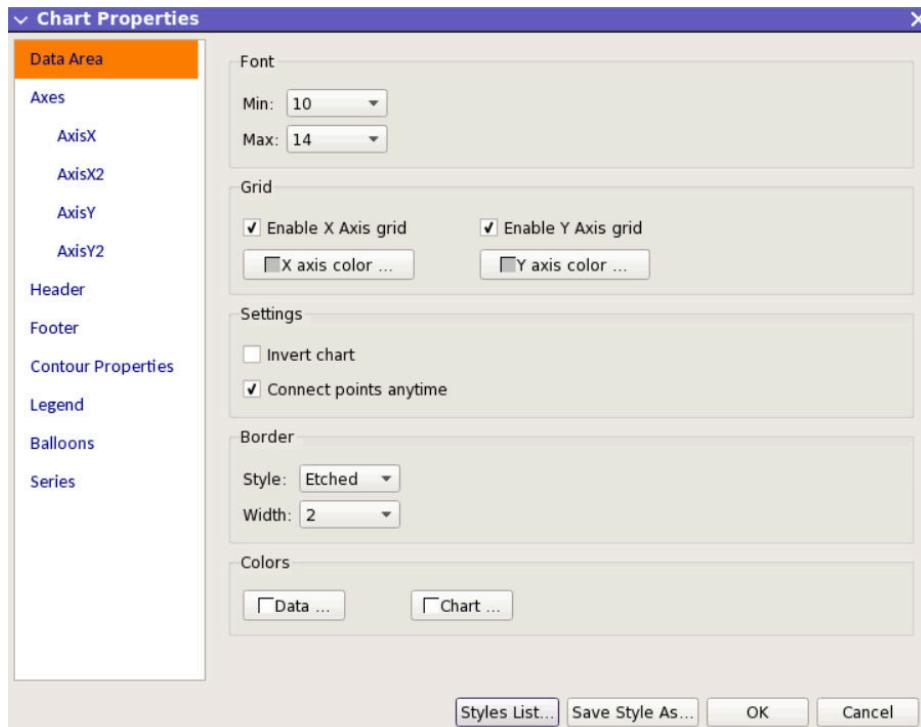
In a chart window, you can:

1. Zoom in on a chart by clicking the upper-left corner of the region you want to zoom in on, then drag your mouse to the lower-right corner of the region you want to zoom into, and release the mouse button.
2. Zoom out by clicking the lower-right corner of the region you want to zoom out of, then drag your mouse to the upper-left corner of the region you want to zoom out of, and release the mouse button.
3. Fit the chart to the current window size, by double-clicking the chart.

Editing Chart Properties

You can edit the properties of any chart using the **Chart Properties** dialog box. To open the **Chart Properties** dialog box:

1. Right-click on a chart, and choose **Chart Properties** from the menu that opens.

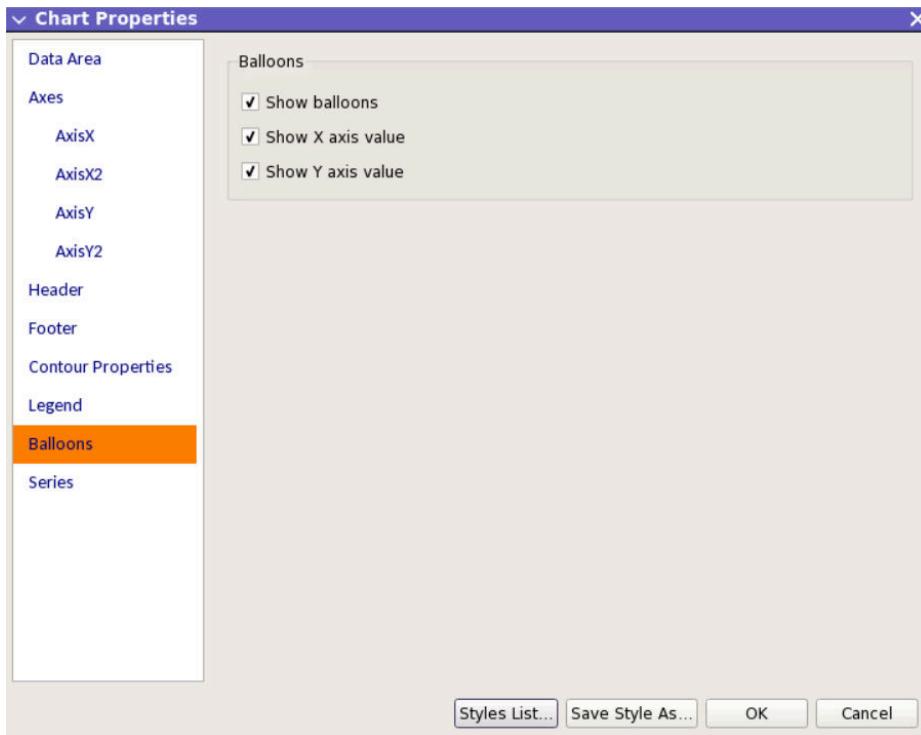


2. Select the menu of properties you wish you edit from the left side of the dialog box.
3. Adjust the properties as necessary and click **OK** to save your changes.

Enabling Data Point Information Balloons

You can enable data point information balloons to view associated value information.

1. Right-click on a chart, and choose **Chart Properties** from the menu that opens.
2. Click the **Balloons** property section in the list of properties displayed on the left-hand side of **Chart Properties** window. The balloons properties are displayed.



3. Click **Show Balloons**.
4. Click **OK** to save your changes. You can now hover over a data point to display value information.

Specifying Parametric Reduction From Charts

You can specify a particular corner, sweep, or Monte Carlo iteration for plotting histogram or Q-Q plots.

To specify parametric reduction from a histogram or Q-Q plot:

1. Click **Parametric Reduction** in the Chart window to open the **Parametric Reduction** dialog box.
2. Select your corner, sweep, or Monte Carlo iteration and parameters.

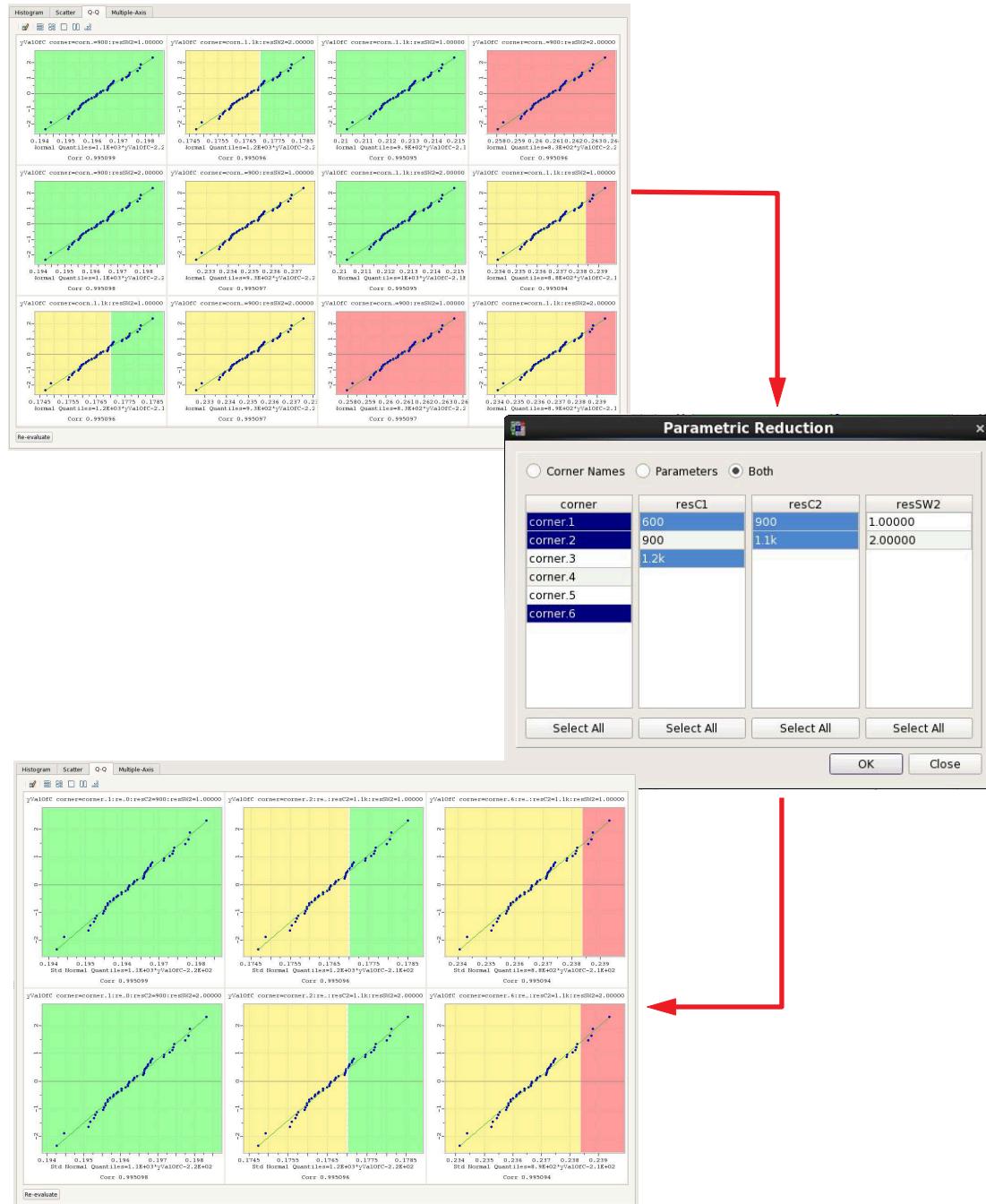
Note:

Multiple selections can be made for both histograms and Q-Q plots.

Chapter 16: Using Charts to Visualize Simulation Results

Specifying Parametric Reduction From Charts

- Click **OK** to plot the new chart with parametric reduction.



17

Plotting, Printing, and Reviewing Simulation Results

This chapter contains information on how to process simulation results.

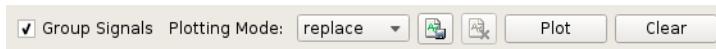
This chapter contains the following topics:

- [Plotting Outputs](#)
- [Creating Custom Plot Modes](#)
- [Plotting Output Sets](#)
- [Plotting Signals Interactively](#)
- [Printing and Annotating Node Voltages and Operating Points](#)
- [Using the Simulation Check Viewer](#)
- [Working with Monte Carlo Data Mining Results](#)
- [Debugging Using Extreme Case Results](#)
- [Saving Simulation Results](#)
- [Loading Simulation Results](#)
- [Specifying Results Options](#)
- [Using the Timestamps Dashboard](#)

Plotting Outputs

To plot outputs from simulations:

1. Ensure your simulation is complete with results you want to plot.
2. In the **Plotting Mode** section just below the Outputs table on the PrimeWave Design Environment main window, choose how you want to plot signals from the **Plotting Mode** menu:



Plotting Mode Description

append	<ul style="list-style-type: none">• Leaves existing signals as-is• If signals exist, plots with first occurrence of existing signal• If signals do not exist, plots in new panel in active tab
new	<ul style="list-style-type: none">• Leaves existing signals as-is• Plots all signals to a new tab
replace	<ul style="list-style-type: none">• If signals exist, replaces the first occurrence and removes all occurrences of the signal• If signals do not exist, plots in a new panel of the active tab• Leaves other existing signals as-is
newPanel	<ul style="list-style-type: none">• Leaves existing signals as-is• Plots all signals to a new panel of the active tab
(custom plot)	<ul style="list-style-type: none">• Clears all tabs for testbench• Plots signals to the custom setup• Plots new signals to last panel

For information on saving custom plot modes, see [Creating Custom Plot Modes](#).

Note:

If using History (single or multi-testbench), signal names will be annotated with the history point index.

3. (Optional) To create a custom plot mode, which saves all display and measurement settings you specify for plotted signals in the waveform viewer, click **Save waveform**

viewer setup as a custom plot mode



See [Creating Custom Plot Modes](#) for information on creating custom plot modes.

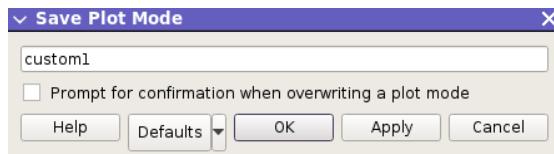
4. (Optional) Check **Group Signals** to plot new signals into one panel (grouped) rather than one signal per panel (stacked).
5. Click **Plot** to plot signals as specified.
 The waveform viewer opens with the specified signals.
6. Click **Clear** to clear waveform viewer of plotted signals and setup.

Creating Custom Plot Modes

You can save and reuse a WaveView session. To create a custom plot mode, which saves all display and measurement settings, as well as histograms for scalar output expressions you specify for plotted signals in the WaveView tool:

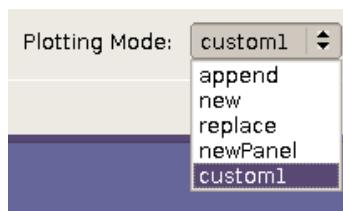
1. Ensure you complete your simulation, and plot all needed signals in the WaveView tool.
 Make sure you set up any display and measurement settings in the waveform viewer.
2. Click the **Save WaveView setup as a custom plot mode** button  in the **Plotting Mode** section of the PrimeWave Design Environment tab page.

The **Save Plot Mode** dialog box opens.



3. Enter a name for the custom plot mode, and click **OK**. (The default is **custom1**.)

The custom plot mode is saved. You can now choose this custom plot mode from the **Plotting Mode** menu on the PrimeWave Design Environment tab page.



Saved custom plot modes replace any saved setup; using custom plot modes does not preserve any existing plotted data.

Note:

In MTB mode, saving a custom plot mode in a testbench saves only the plots that were done for that particular testbench. When you plot again later, your plots will be restored correctly.

Contact your Synopsys representative for details about the limitations of this flow.

Plotting Output Sets

To plot the currently selected output set, choose **Results > Plot/Evaluate Outputs**. Any expression in the plot set that is enabled for auto-plotting is displayed in a waveform viewer graph window.

Expressions that resolve to scalars are re-evaluated, and their results are displayed in the Value column of the Outputs table.

Plotting Signals Interactively

When a simulation is complete, you can interactively plot any signal saved in the simulator results file by making selections on the canvas. For example, if you save all node voltages by clicking the **All** radio button in the **Outputs > Save Options** dialog box, you can interactively plot any voltage in your design without being required to set up an explicit expression for this voltage.

To plot a signal, choose **Results > Plot Signal** from the PrimeWave Design Environment main menu bar, and then choose one of the following data types:

- **Transient Signal**
- **AC Bode**
- **AC Magnitude**
- **DC**
- **Input Noise**
- **Output Noise**

Note:

Some data types might not be available depending on the type of analyses you included in the simulation you ran.

When you choose **Transient Signal**, **AC Bode**, **AC Magnitude**, or **DC**, you are prompted to select a wire or instance terminal in the design. When you select a wire, a voltage is plotted, and when you select an instance terminal, a current is plotted. Selecting an instance plots all currents associated with the terminals of that instance.

Note:

You must opt to save the information that you are attempting to plot.

When you choose **Input** or **Output Noise**, the simulator-generated value for this output is plotted.

Printing and Annotating Node Voltages and Operating Points

The items you can print or annotate differ depending on the integrated simulator you are using. This section includes the following topics:

- [Printing and Annotating Operating Points](#)
- [PrimeSim HSPICE Printing and Annotating](#)
- [FineSim Printing and Annotating](#)

Note:

The PrimeWave Design Environment issues a warning instead of an error if an operating point parameter listed in `opParamExprs` or `opParamExprList` is not found in the results during the Operating Point annotation. Annotation continues for the other parameters (`opParamExprs` or `opParamExprList`) listed next to the op parameter for which results were not found.

Printing and Annotating Operating Points

The first step toward debugging analog circuits is displaying operating point information for circuit components. Operating point information depends on the simulator as well as the device being used. For MOSFET devices, operating points include operation region, threshold voltage (v_{th}), saturation voltage (v_{dsat}), drain current (i_d), transconductance (g_m), and many other operating point parameters. For resistors, current, voltage drop, power, and effective resistance, values are available.

You can use the functionality provided for operating point back-annotation from simulation results. Each device master must include one or more `cdsParam` interpreted labels for the display of operating points. The number of labels in the placed master determine the number of available slots for annotation. The default operating point parameters to display are defined in the CDF property `opPointLabelSet`.

You can use the ability provided by the Device Label Editor to override the default `opPointLabelSet`. For devices that use primitive models (no subcircuits), this capability usually works without any changes to the default component description format information.

The subcircuit models contain a primary instance, which represents the component in the circuit. The process of back-annotation requires accessing the operating points of a device that is one or more levels down in the results hierarchy.

The following example of N-channel MOSFET "macromodel" device adds a 'scale' parameter and uses this parameter to calculate the width and length of the `nch_mac` device.

```
.subckt nch_mac n1 n2 n3 n4 l=length w=width multi='1' nf='1'
  scale='scale_mos'
main n1 n2 n3 n4 nch w='w*scale' l='l*scale' m=multi
.ends
.model nch NMOS level=54
```

After simulation, operating point information is available for the primitive device "main", but not for the `nch_mac` subcircuit. The information is stored as `XY.main:z`, where `XY` is the instance name that calls `nch_mac`, and `z` is the operating point parameter. For example, `X1.main:gm` will be the transconductance of the instance `X1` that uses `nch_mac` as its model.

To be able to back-annotate operating points for macromodel devices, you can use an operating point back-annotation expression language to retrieve these results.

The format is as follows:

```
( (Name "Expression_1" [viewName ... "Expression_n" ViewName_n]) ... )
```

where:

- `Name` is an operating point name.
- `Expression` is a PrimeWave Design Environment expression used to calculate the operating point value.
- `ViewName` is an optional entry used to indicate that the expression is specific to the current switch view of the device instance.

For "nch_mac" model defined above, proper `opParamExprs` expression to retrieve `id`, `vgs`, and `vds` operating point parameters is:

```
( (id "op(main,id)") (vgs "op(main,vgs)") (vds "op(main,vds)"))
```

`op()` is a PrimeWave Design Environment expression language function, which takes the name of the device, the operating point name, and returns operating point value for the named device.

The hierarchy of the current instance is implicitly calculated. The expression needs to be passed only through the hierarchy from the current instance downward. Using this technique, you can also define new operating point parameters.

For example:

```
( (gmid "op(main,gm)/op(main,id)") )
```

This expression defines a new operating point parameter, called "gmid", which is transconductance divided by drain current. "gmid" can be specified in the `opPointLabelSet` field, which adds it to the list of default parameters back-annotated for any devices that use the `nch_mac` model.

Tcl Preference to Control Operating Point Rendering

When you back-annotate operating points, you can control the delay in display of labels using the `deEvalTextRenderTimeout` preference. This preference defaults to 250 milliseconds. This means that, if drawing the labels during back-annotation takes more than 250 milliseconds, the label rendering is delayed and processed in the background as you continue to use the design window.

Refer the table given below for values of the `deEvalTextRenderTimeout` preference.

deEvalTextRenderTimeout Preference Value	Description
-1	Turns off the delayed label rendering capability.
0	Indicates that all labels should always be scheduled for delayed evaluation.
>0	Sets the cumulative time-out that triggers delayed evaluation for the remaining labels. Any value greater than 0 is valid, and it is calculated in milliseconds.

The system processes and displays the instance name labels first, before displaying other labels like operating point values. That way, as back-annotation is running, you can still find instances in the schematic window.

Note that the process to draw the labels itself does not speed up. But you can keep using the schematic tool while back-annotation is running, performing tasks like pan, zoom, or design edits.

Using OP Points Results Tables

Operating points are defined by the individual simulator based on the type of device. After running a simulation, you can view and analyze operating point data using OP Points Results tables. From an OP Points Results table, you can cross-probe to the schematic

design, back-annotate operating point information to overlay the design with a colorized heat map, as well as filter results according to particular parameters.

This section contains the following topics:

- [Creating OP Points Results Tables and Filtering Results](#)
- [Cross-Probing Operating Points With the Design](#)
- [Creating Heat Maps From Operating Point Expressions](#)
- [Exporting Operating Point Tables](#)
- [Filtering Expression Language in OP Points Results Tables](#)

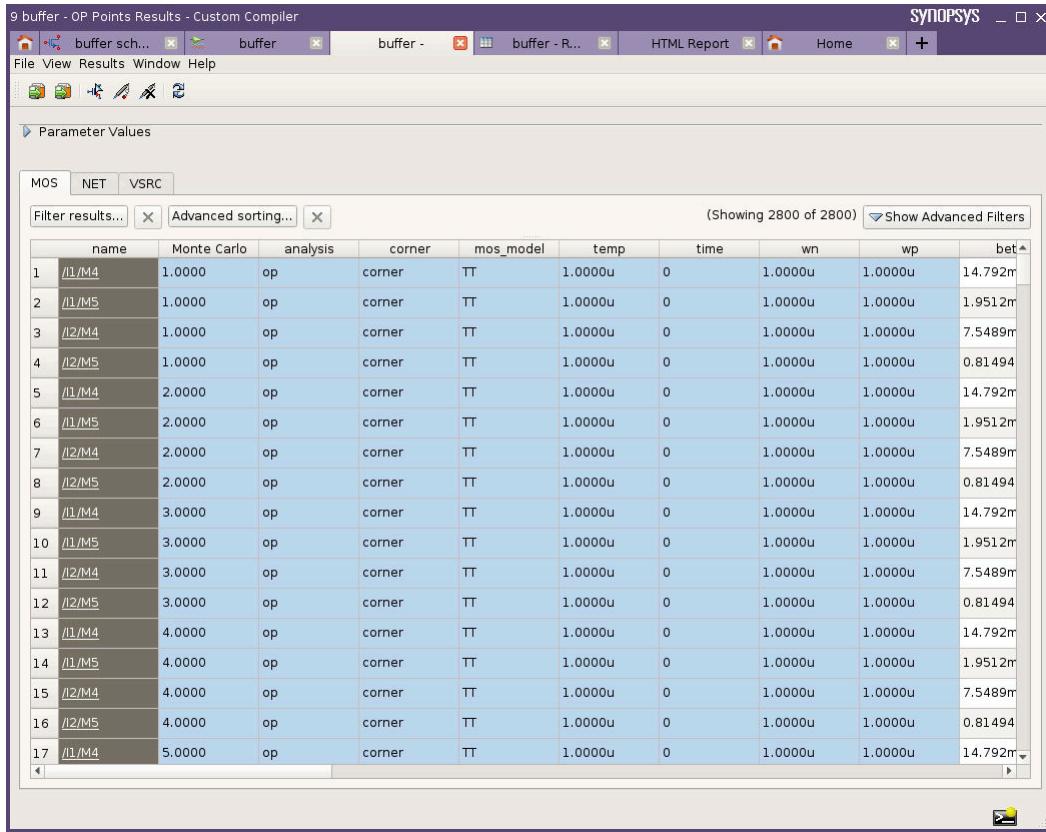
Creating OP Points Results Tables and Filtering Results

To create an OP Points Results table and filter the results:

1. In the main PrimeWave Design Environment window, choose **Results > Print > Operating Point Report**. The OP Points Results viewer opens in a new tab.

corner	temp	wn	wp	mos_model	MONTE_CARLO
corner	-10	1u	1.5u	FF	1
corner1	-40	2u	1u	SS	2
corner2.1	1u	3u	2.5u	TT	3
corner2.2	20		2u		4
corner2.3	25		3u		5
corner2.4	50				6
corner3.1	80				7
corner3.2	110				8
corner3.3	125				9
corner3.4					10
corner3.5					
corner3.6					
corner4					
corner5					
corner6					
corner7					
corner8					
corner9					
corner10					
corner11					
corner12					
corner13					
corner14					
corner15					
corner16					
corner17					
corner18					
corner19					
corner20					
corner21					
corner22					
corner23					
corner24					
corner25					
corner26					
corner27					
corner28					
corner29					
corner30					
corner31					
corner32					

- If you are running corners, sweeps, or Monte Carlo, you must first select the desired parameters from the **Parameter Values** and click **View Results**.

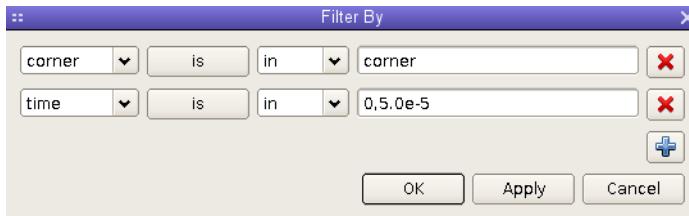


The screenshot shows the 'Parameter Values' table in the Synopsys Custom Compiler. The table has columns for name, Monte Carlo, analysis, corner, mos_model, temp, time, wn, wp, and bet. The data is filtered to show 2800 of 2800 results. The 'MOS' tab is selected.

name	Monte Carlo	analysis	corner	mos_model	temp	time	wn	wp	bet
/I1/M4	1.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
/I1/M5	1.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	1.9512m
/I2/M4	1.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	7.5489m
/I2/M5	1.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	0.81494
/I1/M4	2.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
/I1/M5	2.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	1.9512m
/I2/M4	2.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	7.5489m
/I2/M5	2.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	0.81494
/I1/M4	3.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
/I1/M5	3.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	1.9512m
/I2/M4	3.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	7.5489m
/I2/M5	3.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	0.81494
/I1/M4	4.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
/I1/M5	4.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	1.9512m
/I2/M4	4.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	7.5489m
/I2/M5	4.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	0.81494
/I1/M4	5.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m

The **OP Points Results** table has tabs for each type of device found in the results, such as voltage (VSRC), capacitor (CAP), resistor (RES), current (ISRC), MOS devices (MOS), and so on.

- Click **Filter results** to open the **Filter By** dialog box, in which you can filter results by particular parameters.



- Select from the available parameters using the drop-down menus and choose the options to set up the filter. Click to add a filter or to delete one.

5. Click **OK** to apply the filters and return to the **OP Points Results** table.
6. (Optional) Click the **X** next to the **Filter results** button to reset all previous filters.
7. Click **Show Advanced Filters** to display advanced filtering options (or choose **View > Show All Advanced Filters**).

The screenshot shows the 'Parameter Values' window in the Synopsys PrimeWave Design Environment. The 'Advanced Filter Expression' dialog is open, displaying two filter rows:

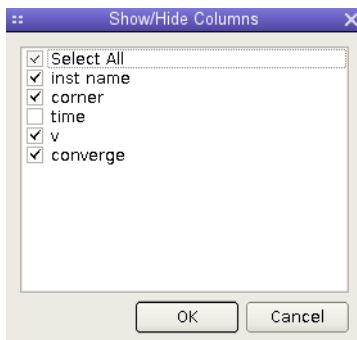
Enabled	Name	Advanced Filter Expression
<input checked="" type="checkbox"/>	Gmid	gm / id > 5
<input type="checkbox"/>		abs(vod) < vth && vgs > vth + vth

Below the dialog is a table listing simulation results:

	name	Monte Carlo	analysis	corner	mos_model	temp	time	wn	wp	bet
1	/I1/M4	1.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
2	/I1/M4	2.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
3	/I1/M4	3.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
4	/I1/M4	4.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
5	/I1/M4	5.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
6	/I1/M4	6.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
7	/I1/M4	7.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
8	/I1/M4	8.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
9	/I1/M4	9.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
10	/I1/M4	10.0000	op	corner	TT	1.0000u	0	1.0000u	1.0000u	14.792m
11	/I1/M4	1.0000	op	corner1	TT	25.0000	0	1.0000u	1.0000u	9.6982m
12	/I1/M4	2.0000	op	corner1	TT	25.0000	0	1.0000u	1.0000u	9.6982m
13	/I1/M4	3.0000	op	corner1	TT	25.0000	0	1.0000u	1.0000u	9.6982m
14	/I1/M4	4.0000	op	corner1	TT	25.0000	0	1.0000u	1.0000u	9.6982m

8. Click in a table cell to add an **Advanced Filter Expression**. You can use column names from the **OP Points Results** table in expressions using SQL syntax appropriate for a "where" clause. The table immediately updates when you enable the filter expression. See [Filtering Expression Language in OP Points Results Tables](#) for more information about allowable expressions.
9. (Optional) Click **Click to Add** to add a row in the **Advanced Filter Expression** table.
10. (Optional) Click **Advanced sorting** to open the **Sort By** dialog box, in which you can sort and hide columns in the table.
11. (Optional) Click the **X** next to the **Advanced sorting** button to reset all previous sorting rules.

12. (Optional) Click to add a sorting rule. Choose **ascending** or **descending** sort. Sorting rules are applied in the order in which they appear in the **Sort By** dialog box.
13. (Optional) Click to delete a sorting rule.
14. Click **OK** to apply the sorting rules and return to the **OP Points Results** table.
15. To hide a column in the table, right-click it and select **Hide** from the menu that appears.
16. To show or hide multiple columns, right-click a column header and choose **Show/Hide Columns** to open the **Show/Hide Columns** dialog box.



Select columns to show and click **OK** to apply the show/hide settings and return to the **OP Points Results** table.

All filtering and sorting preferences are automatically saved for each tab at the session level.

17. Custom Compiler automatically saves your display settings, including any advanced filtering, when you close the table. When you reopen the table, your settings and filters from the previous session are loaded. You can save and load specific display settings to an .xml file using **View > Save/Load Display Settings**.

To reload results and redraw the window, choose **Results > Refresh Results** or click .

Cross-Probing Operating Points With the Design

To cross-probe with the schematic design:

1. Double-click an instance name to highlight it in the schematic design window. To highlight all instances from the table in the schematic, click . To clear those highlights, click .
2. In the default mode, the OP Points Results viewer window displays all results when opened. To select an instance from the schematic to append to the table, click **Probe**

the selected instance in the schematic  and pick the instance in the schematic. When you select one or more instances in this way, the results are filtered and the viewer shows only results for the selected instances.

To change to append mode, set the preference `saOpPointsAppendMode` to true. In append mode, the viewer window displays no results when opened. When you select one or more instances with **Probe the selected instance in the schematic** , those instances are appended to the results.

Note:

If one or more instances are selected while pressing Ctrl, the instances are removed from the filter and the table.

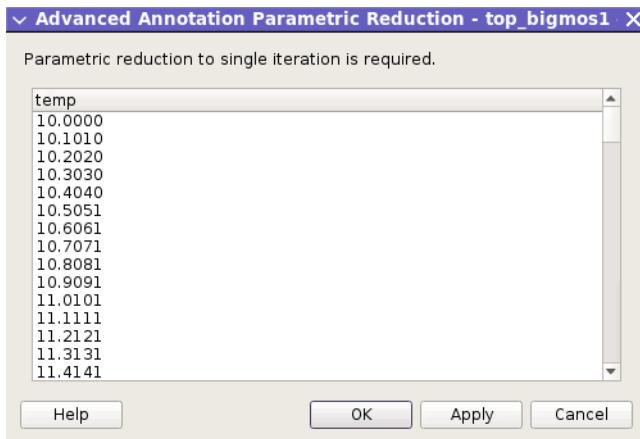
Creating Heat Maps From Operating Point Expressions

A *heat map* (or *thermal map*) is a graphical representation of data where the individual values contained in a matrix are represented as colors. The color-coded information is typically displayed over an image of the raw, data-bearing objects.

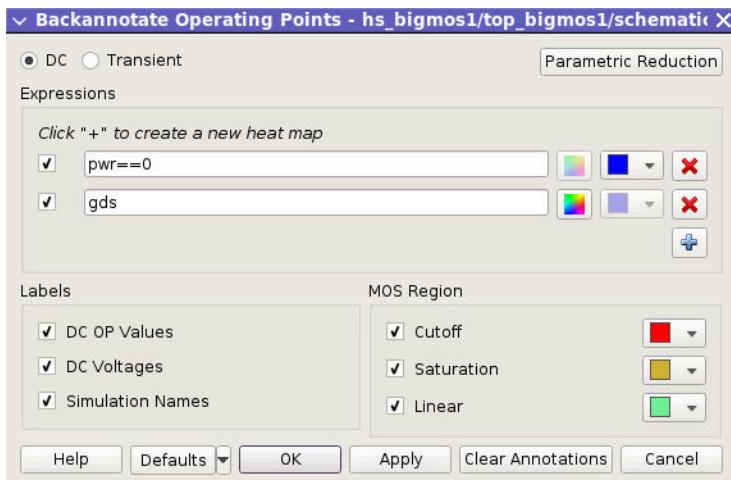
You can annotate schematic designs using operating point expressions to create thermal maps of the design. You can use expressions that yield Boolean or numeric values.

If the expression results in a Boolean value, a single color can be associated with it. This highlights all devices for which the expression is true with that color in the schematic canvas. If the expression results in a numeric value, a gradient can be associated with it. The gradient editor allows you to specify min/max filter limits and corresponding limit colors. Values that lie within the limits are colored onto the schematic, and appropriate colors are assigned based on the histogram/color bin value. To highlight devices in the schematic design based on operating point expressions, for the purpose of creating a heat map:

1. Choose **Results > Annotate > Advanced** from the PrimeWave Design Environment menu bar. You are prompted to select an iteration in the **Advanced Annotation Parametric Reduction** dialog box if the command is being invoked for the selected testbench for the first time. Click **OK** when you have made your selection.

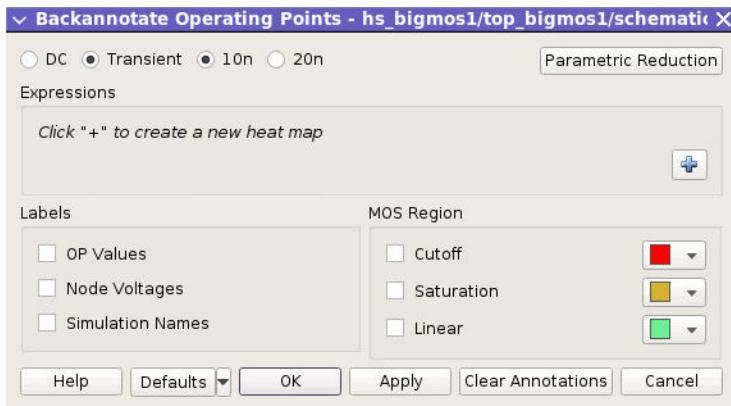


After setting up the parametric reduction, the **Backannotate Operating Points** dialog box opens.



2. Select from the operating point analyses available in the testbench, **DC** or **Transient**.

If you select **Transient**, time options appear allowing you to select the time point of interest.



3. Click to add an expression.

The **Expressions** section provides expressions to evaluate. Based on those evaluations, objects are highlighted in the schematic either by gradient or an exact color.

See [Filtering Expression Language in OP Points Results Tables](#) for more information about allowable expressions.

4. (Optional) Select **Labels**. The **Labels** section allows back-annotation of DC operating point values (**OP Values**), voltages (**Node Voltages**), and **Simulation Names**.

5. (Optional) Select a **MOS Region** to include in the analysis.

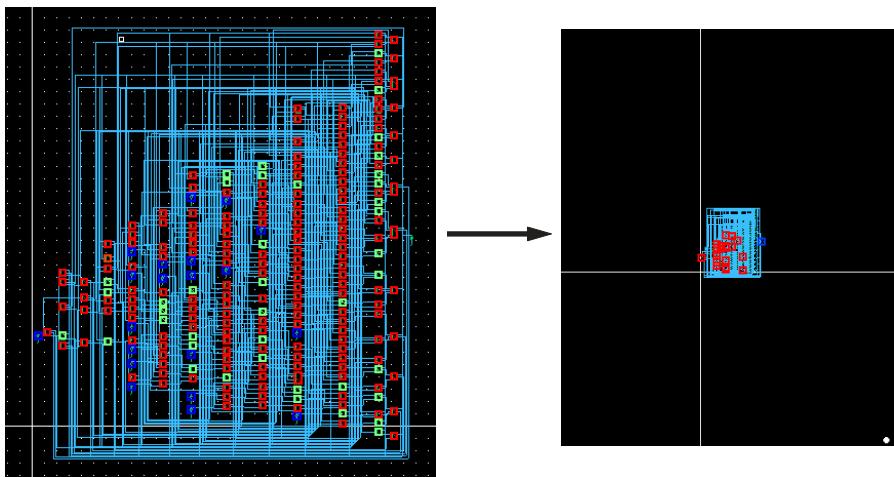
The **MOS Region** section lets you highlight MOS devices based on their working regions, **Cutoff**, **Saturation**, and **Linear**.

6. (Optional) Enable or disable expressions using the checkboxes next to each expression. Disabled expressions are not included in the evaluation.

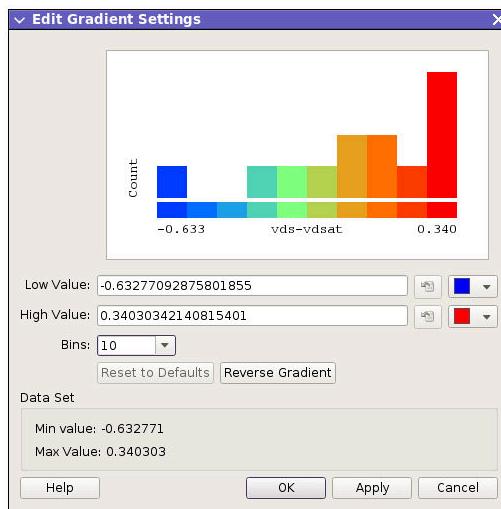
7. (Optional) Click to delete an expression.

8. (Optional) Click the solid color button to change the exact color associated with an expression or MOS region. Select from the color palette that appears.

9. Click **OK** or **Apply** to add color annotations to the schematic design. The Schematic Editor opens, showing the annotated design. Zooming out retains highlighting of instances requiring attention, as shown in the figure below.

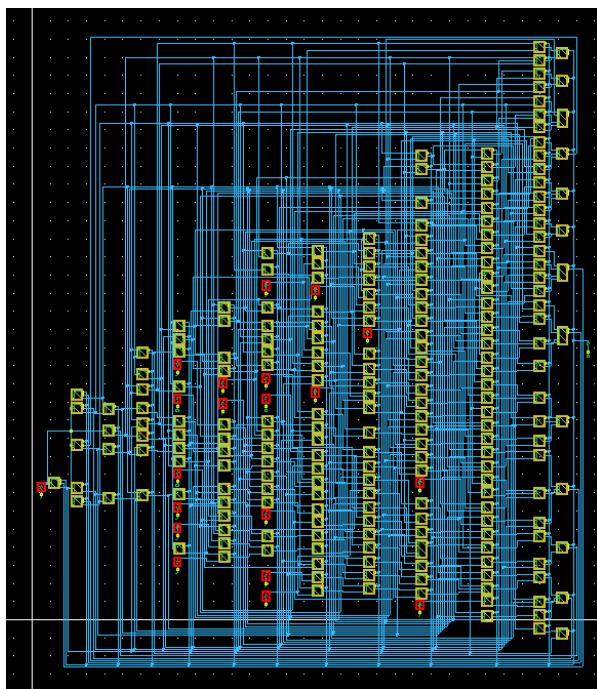


10. (Optional) Enable another expression in the **Expressions** section and click to adjust the gradient settings associated with the expression. The **Edit Gradient Settings** dialog box appears.



A histogram at the top of the dialog box indicates how many values for a given operating point or expression fall into each color bin. By default, colors closer to red are worst case, while those closer to blue are best case. In the example above, a large number of devices are yellow (best case), a smaller number of devices are in the middle light blue to green range, and about the same number are in the high, worst-case range (dark blue). When you hold the pointer over a color bin, a Tooltip appears listing the bin range values (Min, Max) and the number of operating points (Count) represented by that bin.

11. (Optional) Adjust the **Low Value** or **High Value** or change the colors associated with them using the color palette drop-down menus. By default, the entire data set is inspected for the most extreme values, which are used as the initial low/high filter limit values. The low/high value fields also operate as filters. Devices with operating point expression values that lie outside these limits are not annotated.
12. (Optional) Adjust the number of **Bins**.
13. (Optional) Click **Reset to Defaults** to reset the low/high values to the expression's Min/Max values from the database.
14. (Optional) Click **Reverse Gradient** to swap the colors representing the low and high values, making the opposite color the worst case.
15. Click **OK** to apply the gradient settings and close the dialog box.
16. In the **Backannotate Operating Points** dialog box, click **Clear Annotations** to clear all previous color and label annotations in the schematic design.
17. Click **Annotate** to add the updated color and label annotations to the schematic design. The Schematic Editor opens, showing the design with both the gradient and discrete color annotations.



You can navigate through the hierarchy as usual to inspect the color-coded objects. Notice that the user-defined worst-case color coding "bubbles up" through the

schematic hierarchy, all the way to the top level. This gives a quick visual indication of where the trouble spots in a hierarchical design are likely to be, allowing you to navigate as necessary to see more detail, all the way down to the leaf level.

Exporting Operating Point Tables

You can export operating point data to .csv or .html files.

1. To export the table to a .csv file, click **Export to CSV**  to open the **Export to CSV** dialog box. Choose your file name and **Save** the file.
2. To export the table to an HTML file, click **Export to HTML**  to open the **Export to HTML** dialog box. Choose your file name and **Save** the file.

Filtering Expression Language in OP Points Results Tables

When filtering data in an OP Points Results table, the following rules apply for the filtering expression language:

- Numbers: Both scientific and engineering notation can be used.
- Identifiers and Strings:
 - Column names should be enclosed in double quotes (for example, "name") but can often be left bare.
 - Strings must be enclosed in single quotes (for example, 'value').
- Supported numeric and logical operations:

Operators	+	Addition
	-	Subtraction
	*	Multiplication
	/	Division
	%	Modulo
	()	Grouping
Comparators	X BETWEEN y AND z	In range
	=	Equal to
	<	Less than
	<=	Less than or equal to
	>	Greater than

	$>=$	Greater than or equal to
Connectives	$\&\&$, AND	Logical and
	$\ $, OR	Logical or
	NOT	Logical not
Functions	$\text{abs}(X)$	Absolute value
	$\text{round}(X, \text{ digits})$	Round

- Supported string operations:

Comparators	$=$	Equal to
	$X \text{ in } (y, \dots)$	Equal to one of a list
	X GLOB pattern	"Glob" pattern matching
	$<$	Alphabetically less than
	$<=$	Alphabetically less than or equal to
	$>$	Alphabetically greater than
	$>=$	Alphabetically greater than or equal to
Functions	$\text{length}(X)$	Length
	$\text{lower}(X)$	Lowercase
	$\text{substr}(X, \text{ index}, \text{ length})$	Extract substring
	$\text{trim}(X)$	Trim leading and trailing space
	$\text{upper}(X)$	Uppercase

PrimeSim HSPICE Printing and Annotating

Once you complete certain PrimeSim HSPICE analyses, you can annotate information on your schematic or print the information to tables. In the absence of an operating point (.OP) analysis, the following items can be printed and annotated:

- DC node voltages

This is the voltage value from the DC analysis.

- Transient node voltages

This is the voltage value at a specified timepoint during the transient run.

If you run an operating point analysis in conjunction with the transient analysis, the device operating points at the specified timepoints are also available for printing or annotation.

The following HSPICE printing and annotation options are available:

- [Printing PrimeSim HSPICE Voltages and Operating Points](#)
- [Printing PrimeSim HSPICE AC and DC Mismatches](#)
- [Printing the PrimeSim HSPICE Noise Summary](#)
- [Annotating PrimeSim HSPICE Voltages and Operating Points](#)

Printing PrimeSim HSPICE Voltages and Operating Points

To print a node voltage or operating point, choose **Results > Print > DC Node Voltages**, **Transient Node Voltages**, or **Operating Point Report** from the PrimeWave Design Environment main menu bar.

If you choose **DC Node Voltages**, you are immediately prompted to select a wire from the schematic for voltage. Once you make a selection, the results appear in a table. You can pick additional wires to append to the table, and at any time you can click the **Append** button on a table and make further picks from the schematic.

If you choose **Transient Node Voltages**, you are prompted to select a time point for the printing or annotation and then a wire from the schematic for voltage and an instance for operating points.

If you choose **Operating Point Report**, the **OP Points Results** table opens in a new window. Operating points are defined by the individual simulator based on the type of device. See [Using OP Points Results Tables](#) for information about reading operating point reports.

Printing PrimeSim HSPICE AC and DC Mismatches

To print the AC or DC mismatch, choose **Results > Print > AC Mismatch** or **DC Mismatch** from the PrimeWave Design Environment main menu bar. Tabular data is generated, and that data is displayed in the Text Viewer. The data is also written as ASCII text to a *.am or *.dm file, respectively.

Printing the PrimeSim HSPICE Noise Summary

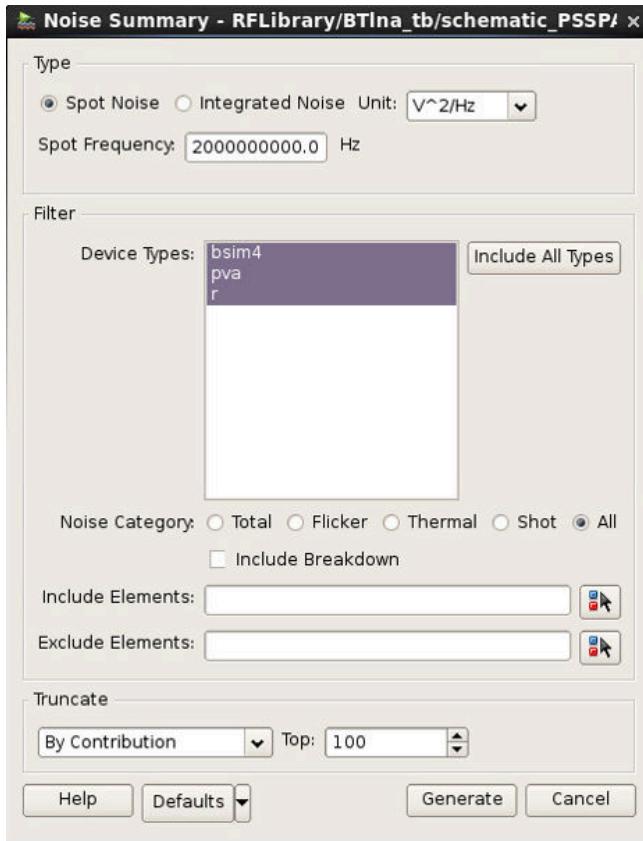
PrimeSim HSPICE calculates the normalized output and input noise levels relative to the square root of the noise bandwidth and writes the values to the noise output file. In addition, PrimeSim HSPICE calculates and writes the contribution of each noise generator in the circuit for each frequency point.

Noise summary reports are available for the following PrimeSim HSPICE analyses: Shooting Newton Noise, Harmonic Balance Noise, Phase Noise, AC Noise, and Periodic Time-Dependent Noise.

To print the noise summary:

1. Choose **Results > Print > Noise Summary** from the PrimeWave Design Environment main menu bar.

The **Noise Summary** dialog box opens.



2. Choose the range of noise you want to include in the summary: **Spot Noise** or **Integrated Noise**.
 The **Spot Noise** is the total noise at a specific frequency (the **Spot Frequency**), and the **Integrated Noise** is the noise over a specified frequency range (the **Start** and **End** frequencies).
3. Choose the **Units** used in the noise summary: **V²** or **V**.
4. If you choose **Spot Noise**, enter a value for the **Spot Frequency**. Otherwise, enter **Start** and **End** frequency values if you choose **Integrated Noise**.
5. (Optional) In the **Filter** section, if you want to display results only from specific device types, enter the name(s) of those devices in the **Device Types** text field. Or click **Include All Types** to automatically include all types.
6. Select the type of **Noise Category** to report: **Total**, **Flicker**, **Thermal**, **Shot**, or **All**.
7. (Optional) Click **Include Breakdown** to report all the noise parameter breakdowns.

8. (Optional) If you want to include or exclude certain elements in your noise summary, enter one or more instance names in the **Include Elements** and **Exclude Elements** text fields as necessary.

You can also click the **Select** buttons  to choose an instance from your schematic.

9. Choose one of the following **Truncate** methods, which sorts the summary data by a number or threshold:

- **By Contribution**

Displays the top X noise contributors (as opposed to only those contributing over X % or only those that contribute more than X).

- **By Relative Threshold**

An entry is interpreted as a percentage contribution. Any entries whose value in the Total% column falls below the specified value is omitted from the report.

- **By Absolute Threshold**

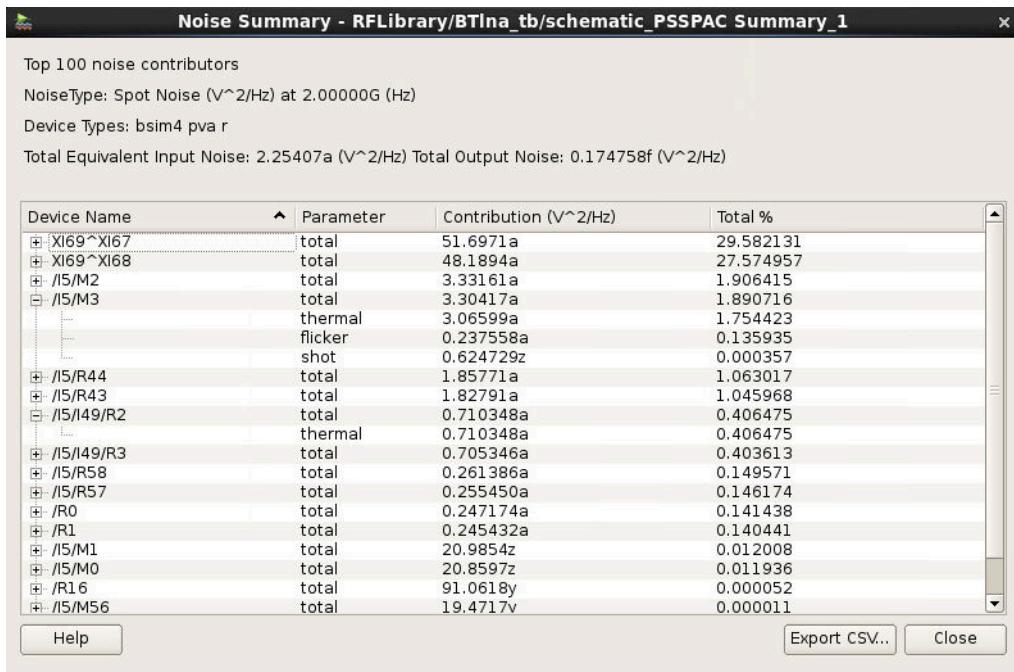
An entry is interpreted as the actual value of the noise. Any entries whose value in the Contribution column falls below the specified value is omitted from the report.

Note:

If you choose the relative or absolute threshold, only the total contribution of a device is filtered and not the individual device noise parameters. (These are the parameters that are included in the report if you enable the **Include device noise parameters** option.)

10. (Optional) Choose a **Top** number of noise contributors to include in the summary report. Default is 100.
11. Click **Generate** to save your settings and create the noise summary.

The noise summary report is displayed.

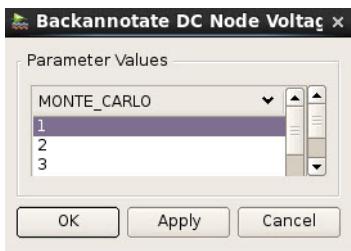


When a transistor is expanded, it gives the total, thermal, flicker, and shot noises for the selected device, as well as the percentage of the total noise. A resistor usually only contains thermal noise.

You have the option of exporting the report to a .csv file by clicking **Export CSV**.

Annotating PrimeSim HSPICE Voltages and Operating Points

To annotate a node voltage or operating point, choose **Results > Annotate > DC Node Voltages, Transient Node Voltages, DC Operating Point, or Transient Operating Point** from the PrimeWave Design Environment main menu bar. The **Backannotate** dialog box opens.



If you annotate voltages or operating points, the annotations immediately appear hierarchically through the schematic once you choose **Results > Print > DC Node**

Voltage or Operating Point Report. (See [Using OP Points Results Tables](#) for information about reading operating point reports.)

For **DC Node Voltage**, you can set the following preferences to control how these values are displayed:

- `saParamDisplayPrecision`

This preference controls the number of significant digits of the annotated values.

- `saParamDisplayEngNotation`

This preference controls whether the values are displayed in engineering notation (using suffixes) or exponential (using a format such as 1.256e-9).

The operating points that are displayed for each device are determined by the `opPointLabelSet` value in the cell Parameter Definition, which you can change using the Parameter Definition Editor. You can also use the Device Label Editor to control the display of these annotations; see the *Custom Compiler Environment User Guide* for more information.

Note:

When a simulation is rerun, values annotated to the schematic are automatically updated.

Component names that are used in the netlist can also be annotated to the schematic. Since netlisters name-map the identifiers so that they are legal in the simulator name space, often a mismatch occurs between the name of a component in a netlist and the name in the schematic. To see the netlist name on the schematic, chose **Results > Annotate > Simulation Names**.

When you back-annotate simulator names, the full hierarchical names are displayed in the language of the simulator you are using.

To clear the schematic of any simulator annotations, choose **Results > Annotate > Clear Annotations** from the PrimeWave Design Environment main menu bar.

FineSim Printing and Annotating

Once you complete certain FineSim analyses, you can annotate information on your schematic or print the information to tables. Without operating points specified, the following items can be printed and annotated:

- DC node voltages

This is the voltage value at time=0. (See "Important" note below.)

- DC operating points

This is the value of the various operating points of a device at time=0. (See "Important" note below.)

- Transient node voltages

This is the voltage value at a specified timepoint during the transient run.

If you run an operating point (.OP) analysis in conjunction with the transient analysis, the device operating points at the specified timepoints are also available for printing or annotation.

Caution:

You must explicitly add 0 (zero) to the op **Times** list in the Analysis pane in order for the **Annotate** menu to show **DC Node Voltages** and **DC Operating Point** after simulation.

Analysis	Type	En	Value
+ tran	tran	<input checked="" type="checkbox"/>	Start Time: 0 Time Step: 10p Stop Time:...
+ op	op	<input checked="" type="checkbox"/>	Times: 0 60n



To print a node voltage or operating point, choose **Results > Print > DC Node Voltages**, **Transient Node Voltage**, or **Operating Point Report** from the PrimeWave Design Environment main menu bar.

If you choose **Transient Node Voltage**, you are prompted to select a time point for the printing or annotation.

If you choose **DC Node Voltages**, you are prompted to select a wire from the schematic for voltage and an instance for operating points. Operating points are defined by the individual simulator based on the type of device. Once you make a selection, the results appear in tabulated form. You can pick additional wires or instances to append to the table, and at any time you can click the **Append** button on a table and make further picks from the schematic.

If you annotate voltages or operating points, the annotations immediately appear hierarchically through the schematic once you choose **Results > Print > DC Node Voltages** or **Operating Point Report**.

See [Using OP Points Results Tables](#) for information about reading operating point reports.

You can also print the noise summary by choosing **Results > Print > Noise Summary**.

For **DC Node Voltages**, you can set the following preferences to control how these values are displayed:

- `saParamDisplayPrecision`

This preference controls the number of significant digits of the annotated values.

- `saParamDisplayEngNotation`

This preference controls whether the values are displayed in engineering notation (using suffixes) or exponential (using a format such as 1.256e-9).

The operating points that are displayed for each device are determined by the `opPointLabelSet` value in the cell Parameter Definition, which you can change using the Parameter Definition Editor. You can also use the Device Label Editor to control the display of these annotations; see the *Custom Compiler Environment User Guide* for more information.

Note:

When a simulation is rerun, values annotated to the schematic are automatically updated.

Component names that are used in the netlist can also be annotated to the schematic. Since netlisters name-map the identifiers so that they are legal in the simulator name space, often a mismatch occurs between the name of a component in a netlist and the name in the schematic. To see the netlist name on the schematic, choose **Results > Annotate > Simulation Names**. If you have mixed-signal simulation results, you can also see the element names on a schematic by choosing **Results > Annotate > Element Names**

Using the Simulation Check Viewer

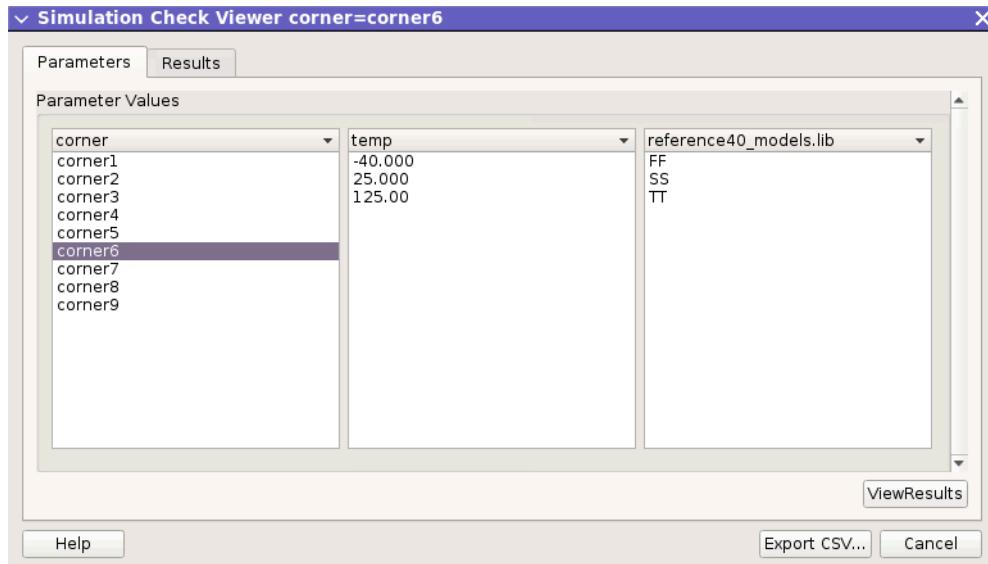
The Simulation Check Viewer processes simulation output files to let you quickly review results of checks you created in the PrimeWave Design Environment user interface, as well as checks included in the simulation via include files or PDKs. You can then cross-probe the results with the design in the Schematic Editor.

The Simulation Check Viewer shows the results of the following dynamic checks:

- PrimeSim HSPICE Bias checks
- PrimeSim HSPICE Safe Operating Area (SOA) checks
- FineSim Circuit checks
- FineSim SOA checks
- PrimeSim XA SOA checks

To view and analyze check results using the Simulation Check Viewer:

1. Choose **Results > Print > Simulation Check Viewer** from the PrimeWave Design Environment menu bar. The **Simulation Check Viewer** dialog box opens.

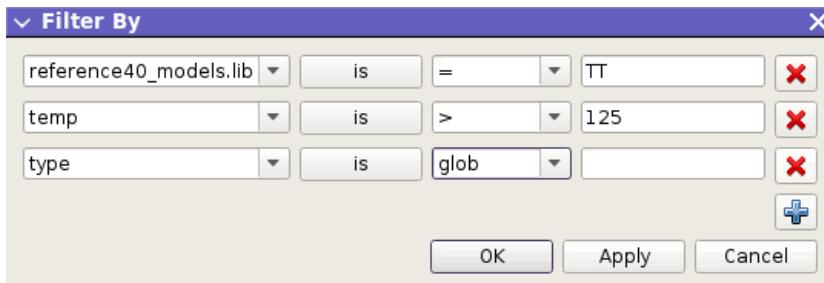


2. The **Parameters** tab opens if you are running advanced analyses (corners or sweeps, for example). Select one or more parameters and click **View Results** to see those results in the **Results** tab.
3. Double-click a device or a net in the results table to cross-probe to the design. Objects that can be probed have an underlined label.

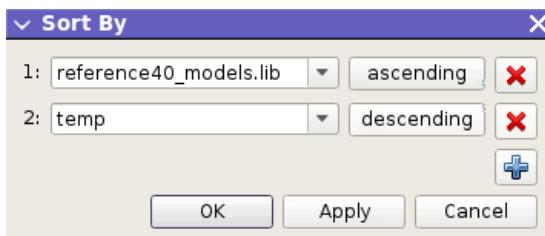
Caution:

Although you can double-click a device terminal, doing so does not cross-probe to the design.

4. (Optional) Click **Filter Results** to filter the results. This dialog box is populated with the parametric reduction selected earlier in the **Parameters** tab. The number of filters is not limited and can be used to reduce the visible data set to a manageable number to analyze.



5. (Optional) Click a column header to sort the results by that single column. For multi-column sorting, click **Advanced Sorting** to open the **Sort By** dialog box and select your sorting rules.



6. (Optional) Right-click a column header and choose **Freeze** to keep that data visible on the left side of the table as you scroll through the results. Right-click the column header and choose **Unfreeze** to unfreeze that column.

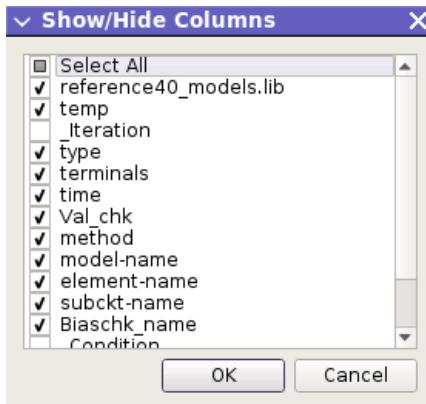
En	Tags	Error Count
<input checked="" type="checkbox"/>	ds_pmos_limit	58
<input checked="" type="checkbox"/>	min_1	234
<input checked="" type="checkbox"/>	ds_limit	4

Number of Errors: 618

7. (Optional) Adjust the columns displayed.

Hide an individual column by right-clicking the column header and choosing **Hide**.

Right-click a column header and choose **Show/Hide Columns** to select columns to display or hide in the **Show/Hide Columns** dialog box.



8. (Optional) Click **Export CSV** to export the results to a .csv file.

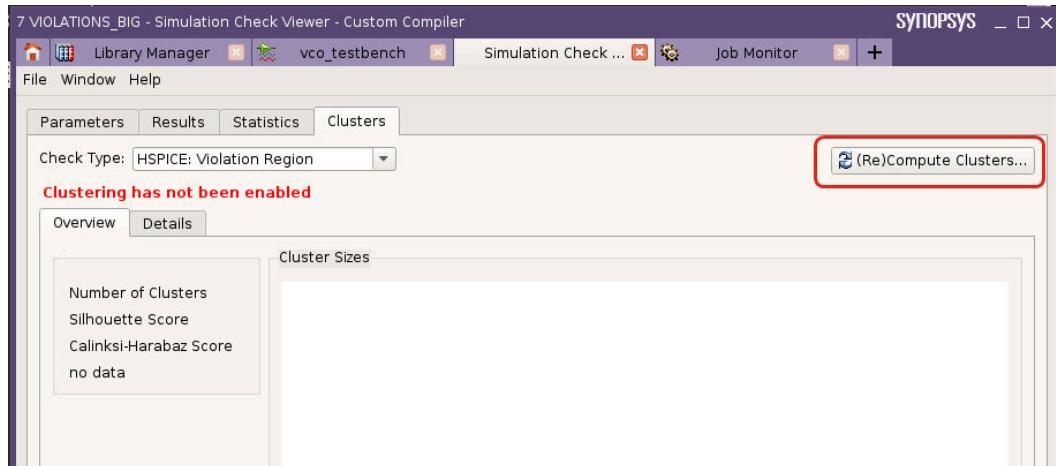
Cluster Analysis Using the Simulation Check Viewer

You can set up cluster analysis of circuit/device check violations in the **Setup > Simulator** dialog box before running a simulation, or in the Simulation Check Viewer after running a simulation.

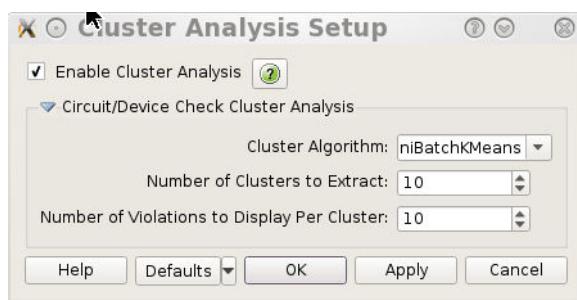
See [Choosing a Simulator](#) for information on how to set up cluster analysis before running a simulation.

To set up cluster analysis from the Simulation Check Viewer after running a simulation:

1. On the **Clusters** tab, click the **(Re)Compute Clusters** button.

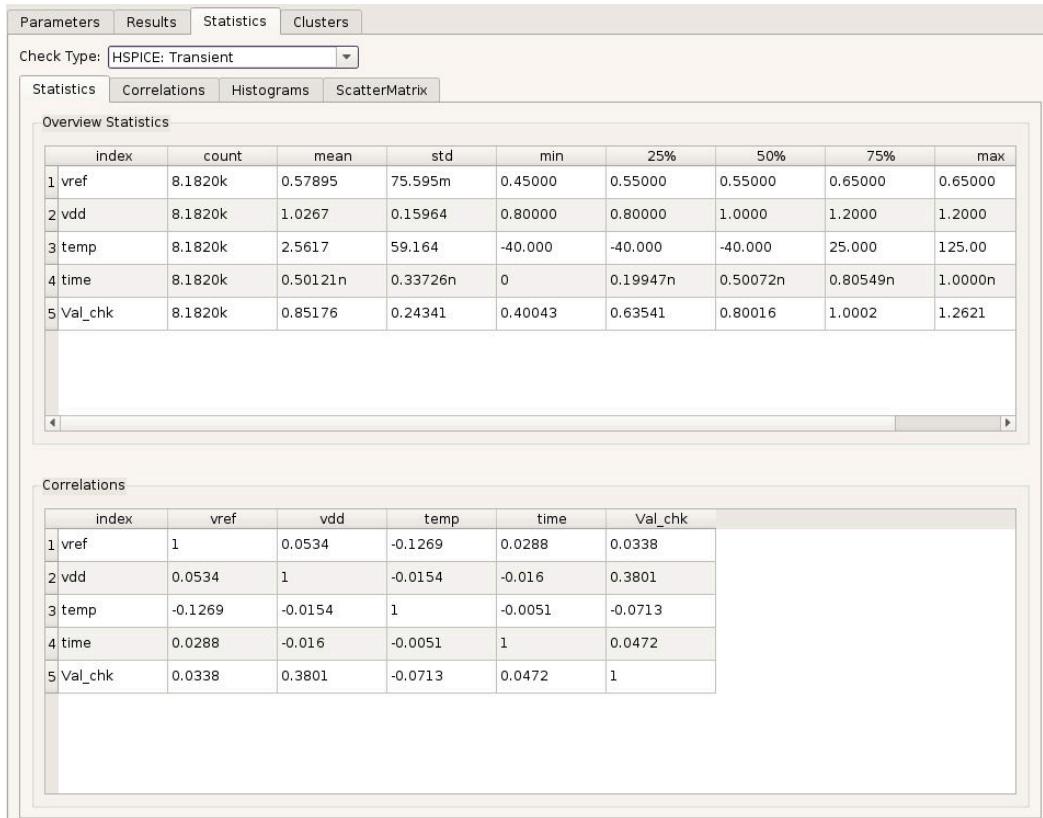


The **Cluster Analysis Setup** dialog box opens:



2. In the **Cluster Analysis Setup** dialog box, select **Enable Cluster Analysis** to perform a cluster analysis of circuit/device check violations.
3. Expand the **Circuit/Device Check Cluster Analysis** section and select your options for cluster analysis:
 - **Cluster Algorithm** is set by default to the **MiniBatchKMeans** algorithm, which is very similar to the commonly used K-means algorithm, but performs better with large data sets.
 - **Number of Clusters to Extract** (disabled when using the Affinity-Propagation Clustering algorithm)
 - **Number of Violations to Display per Cluster** sets the initial number of violations to display. Note that regardless of this setting, it is still possible to show all violation items for each cluster.

- Click **OK** to exit the dialog box and run the cluster analysis. The tables and graphs in the **Statistics** and **Clusters** tabs of the Simulation Check Viewer are updated with the results.



Working with Monte Carlo Data Mining Results

When a Monte Carlo simulation is run using PrimeSim HSPICE, PrimeSim XA, PrimeSim SPICE, PrimeSim Pro, or FineSim, the measurement results are postprocessed, which generates information about how devices affect the variation of each output. You can perform this analysis and review the results in the PrimeWave Design Environment.

This section contains information on the following topics:

- [Viewing Data Mining Results](#)
- [Probing Data Mining Results](#)
- [Viewing Data Mining Data Files](#)

Viewing Data Mining Results

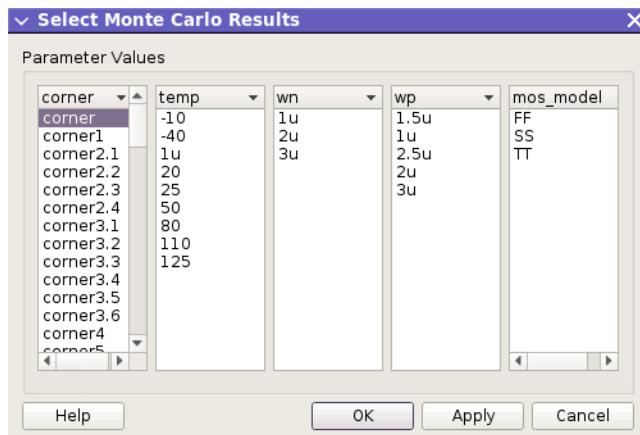
Note:

If your model file uses agauss expressions (as opposed to a variation block) to vary device parameters randomly, you must first set the SAMPLING_METHOD option to SRS. See [Setting Simulator Options](#).

To view and probe Monte Carlo data mining results:

1. If you have more than one Monte Carlo analysis set up, choose **Results > Data Mining > Analyze Results** from the PrimeWave Design Environment menu bar. Otherwise, skip to Step 3.

At this point, if there were iterative analyses (sweeps and/or corners) in addition to the Monte Carlo analysis, the **Select Monte Carlo Results** dialog box opens.

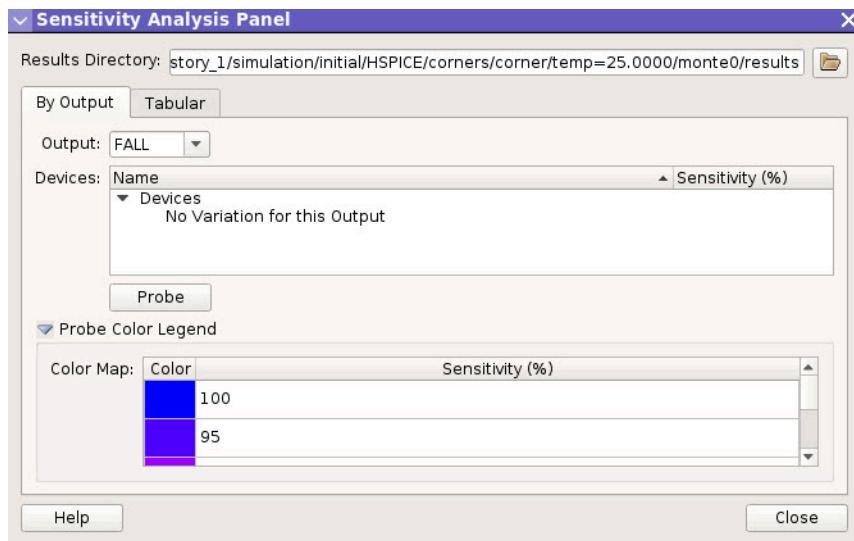


2. Select a Monte Carlo **Parameter Value** and click **OK**.

Data mining results are limited to this parameter value.

3. Choose **Results > Data Mining > Analyze Results** from the PrimeWave Design Environment menu bar.

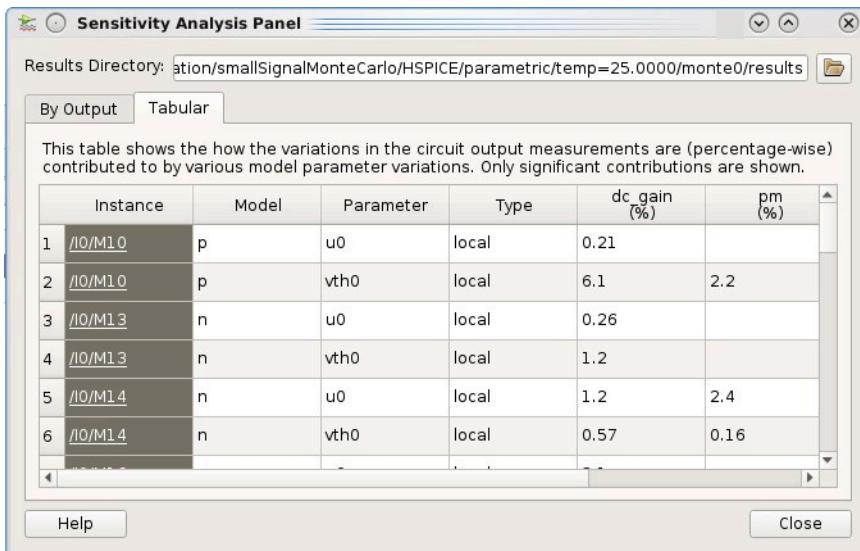
PrimeSim HSPICE performs Monte Carlo data mining postprocessing in the background, and the **Sensitivity Analysis Panel** opens with the results showing **By Output**.



Note:

If data is already present from a previous data mining analysis, you do not need to recompute the sensitivity analysis if you did not make changes to or add any measurements.

4. (Optional) Change the path to the **Results Directory** if needed.
5. Choose a statistical measurement from the **Output** menu.
The corresponding device information is displayed in the Devices table.
6. (Optional) Change to the **Tabular** tab to view multiple measurements at the same time.

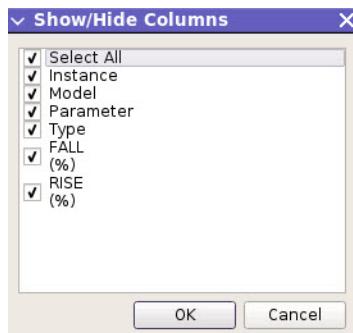


This table allows you to see the various sources that contribute to the variation in the output measurements. Some of these sources might be global, and some might be due to local mismatch variation for specified instances. Double-clicking those instances (where present) in the first column cross-probes to the schematic design or source text.

7. (Optional) Adjust the columns displayed.

Hide an individual column by right-clicking the column header and choosing **Hide**.

Right-click a column header and choose **Show/Hide Columns** to select columns to display or hide in the **Show/Hide Columns** dialog box.



8. (Optional) Sort the data based on a particular measurement by clicking that measurement's column header.
9. (Optional) Right-click a column header and choose **Freeze** to keep that data visible as you scroll through the results. Right-click the column header and choose **Unfreeze** to unfreeze that column.

10. (Optional) Double-click an **Instance** to cross-probe to the schematic design. You can change the probe mapping colors in the **Probe Color Legend** section of the **By Output** tab.

If you want to probe the data mining results, see [Probing Data Mining Results](#).

Probing Data Mining Results

To probe data mining results, click a device name or a device group in the **Sensitivity Analysis Panel** (**Results > Data Mining > Analyze Results**), and click **Probe**. The Schematic Editor opens, and the devices are highlighted in the schematic window.

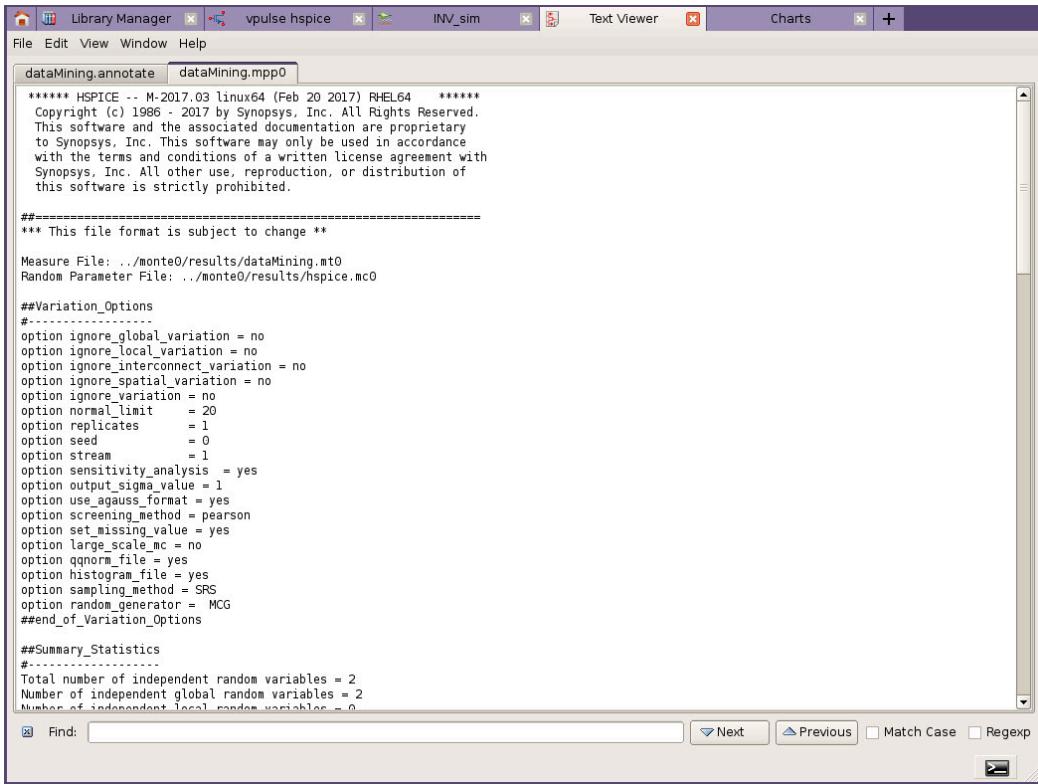
You can change the probe mapping colors in the **Probe Color Legend** section of the **Sensitivity Analysis Panel**.

Viewing Data Mining Data Files

When you run a Monte Carlo data mining analysis, the following data files are created in the specified results directory:

- `dataMining.annotate`
Contains Monte Carlo annotation data.
- `dataMining.mpp0`
Contains a summary statistics table for all the outputs and a variable screening section.

To view these data files, choose **Results > Data Mining > View Files** from the PrimeWave Design Environment menu bar. The data files open in a text viewer window.



```

Library Manager  vpulse.hspice  INV_sim  Text Viewer  Charts
File Edit View Window Help
dataMining.annotate  dataMining.mpp0
***** HSPICE -- M-2017.03 linux64 (Feb 20 2017) RHEL64 *****
Copyright (c) 1986 - 2017 by Synopsys, Inc. All Rights Reserved.
This software and the associated documentation are proprietary
to Synopsys, Inc. This software may only be used in accordance
with the terms and conditions of a written license agreement with
Synopsys, Inc. All other use, reproduction, or distribution of
this software is strictly prohibited.

=====
*** This file format is subject to change **

Measure File: .../monte0/results/dataMining.mt0
Random Parameter File: .../monte0/results/hspice.mc0

##Variation_Options
#-----
option ignore_global_variation = no
option ignore_local_variation = no
option ignore_interconnect_variation = no
option ignore_spatial_variation = no
option ignore_variation = no
option normal_limit = 20
option replicates = 1
option seed = 0
option stream = 1
option sensitivity_analysis = yes
option output_sigma_value = 1
option use_agauss_format = yes
option screening_method = pearson
option set_missing_value = yes
option large_scale_mc = no
option qnorm_file = yes
option histogram_file = yes
option sampling_method = SRS
option random_generator = MCG
##end_of_Variation_Options

##Summary_Statistics
#-----
Total number of independent random variables = 2
Number of independent global random variables = 2
Number of independent local random variables = 0

```

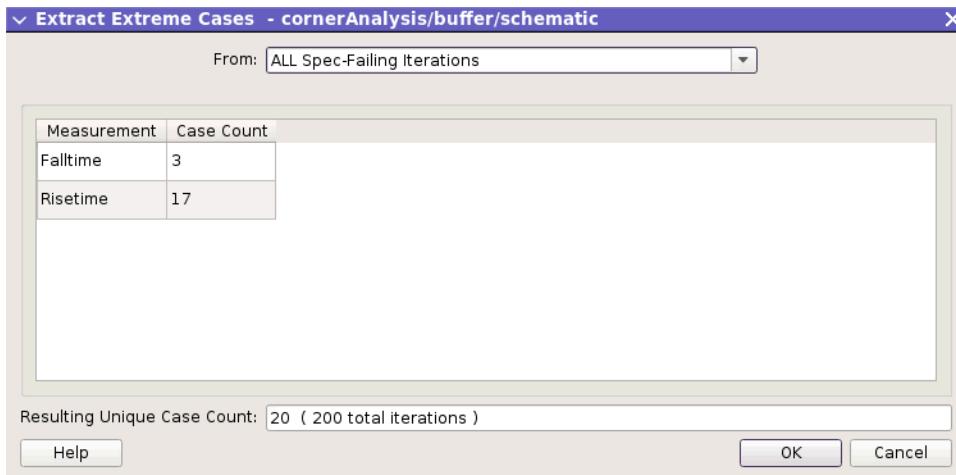
Debugging Using Extreme Case Results

You can designate failed corners or iterations from simulation results to postprocess as a new set of iterations for extreme case analysis. This allows you to continue improving the circuit without having to resimulate conditions that have already passed.

After you have run a simulation, you can set up an extreme case analysis from the results:

1. Netlist and run a simulation.
2. Choose **Results > Viewer** to view the results. Notice any failed measurements.
3. In the main PrimeWave Design Environment window, choose **Results > Debug Extreme Cases**.

The **Extract Extreme Cases** dialog box opens.



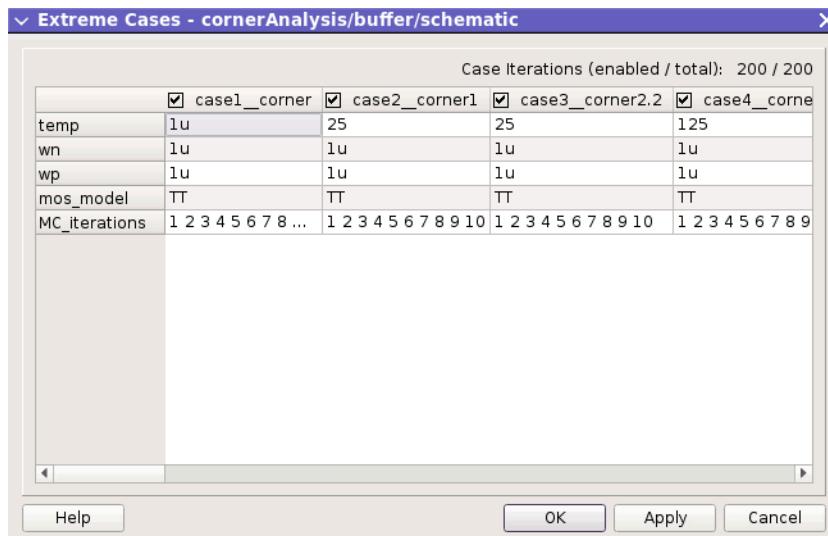
4. Select how to filter results from the Results Database. Choose from:
 - **ALL Spec-Failing Iterations:** Reports all the extreme cases and the total number of iterations that have failed.
 - **Spec-Failing Iterations, by Measurement:** Reports failing iterations for user-selected measurements.
 - **ALL Measurement Distribution Tails:** Creates a statistical curve of the results and reports the outliers (tails) on that curve.
 - **Measurement Distribution Tails, by Measurement:** Reports measurement distribution tails for user-selected measurements.
 - **ALL Measurement Extreme Value Iterations:** Creates a statistical curve of the results and reports the extreme high and low values.
 - **Measurement Extreme Value Iterations, by Measurement:** Reports extreme high and low values for user-selected measurements.
 - **Iterations Selected in Results Viewer:** Allows you to select iterations from the ResultsView.
5. (Optional) To push selected extreme cases to the **Corners** dialog box, right-click in the table to open the menu and select **Add to Corners**.
 - The new corner is created using the extreme case name.
 - If a corner already exists with the extreme case name, the corner is updated with the parameter values from the extreme case.

- The **Add to Corners** option does not create a new corner parameter while creating the corner. For example, if the extreme case has the parameter **tune**, the same parameter **tune** should exist in the corners setup.
 - This feature is only available for the corners run (without sweep or Monte Carlo).
6. Click **OK** to set up the extreme case analysis. In the PrimeWave Design Environment Analysis pane, the Debug Cases analysis appears (outlined in red below).

Analysis	Type	En	Value
Debug Cases		<input checked="" type="checkbox"/>	Total Iterations: 200, enabled: 200
corners		<input type="checkbox"/>	Total: 70, enabled: 70
▶ sweep vdd	sweep	<input type="checkbox"/>	Start = 1, Stop = 2
▶ Monte Carlo	Monte Carlo	<input type="checkbox"/>	Iterations: 20 Start Iteration: 1
▶ tran	tran	<input checked="" type="checkbox"/>	Start Time: 0 Time Step: 1n Stop Time: 20n
▶ op	op	<input checked="" type="checkbox"/>	Format: All

The grayed-out analyses will not be performed when you resimulate, although some of the settings (such as for Monte Carlo) are retained for extreme case analysis.

7. In the PrimeWave Design Environment Analysis pane, double-click the table cell **Debug Cases** to open the **Extreme Cases** dialog box.



For each measurement, the dialog box captures the settings at which that measurement failed.

Note:

When performing extreme cases with Monte Carlo, if you make a change to the circuit topology, MC_iterations are likely no longer valid. In this case, you need to rerun the full simulation.

8. Use the checkboxes to include or exclude cases in the analysis. At this point, you can fine-tune the analysis to pull tail points within specification. For example, you might want to change one of the variable's value.
9. Click **OK** save any changes you made to the extreme case analysis.
10. Choose **Simulation > Netlist and Run** to resimulate the extreme cases.
11. Choose **Results > Viewer** to view the results. Notice any cases where extreme case that failed previously are now passing. If the results are favorable, you can make changes to the actual circuit so that the measurements are now within specification.
12. Repeat the extreme case analysis as necessary until all extreme corners pass. At that point, you can delete the extreme case analysis in the Analysis pane.

Saving Simulation Results

To save the results directory of a simulation so that it can be loaded at a future time, choose **Results > Save** from the PrimeWave Design Environment main menu bar.

You are prompted to enter a directory name; the currently active directory that contains your testbench is copied to this new directory. Iterative results, which are created by parametric, corners, or Monte Carlo analyses, can be saved. Single run results can be saved as well.

Loading Simulation Results

To load previously saved results, choose **Results > Load** from the PrimeWave Design Environment main menu bar, and browse to the directory of the saved results.

Note:

The browser does not stop at the run directory location, so make sure you select the correct directory. The default name of a single run results directory is "nominal" or a variation of that name. Corners analyses are saved with the "corners" directory name, parametric analysis are saved with the "parametric" directory name, and Monte Carlo analyses are saved with the "monte_carlo" directory name.

Check the **Restore session to match loaded results** check box if you want to update the current session setup to match the settings that are specified when the results you are loading are created.

Specifying Results Options

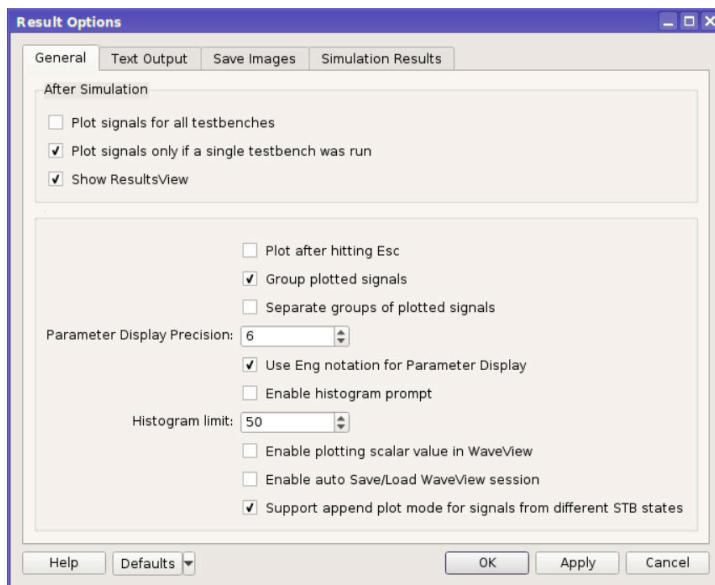
To specify results options, choose **Results > Options** from the PrimeWave Design Environment main menu bar. The **Result Options** dialog box opens.

See the following sections for information on specifying results options:

- [Specifying General Result Options](#)
- [Specifying Report Options](#)
- [Specifying Image Saving Options](#)

Specifying General Result Options

Specify general result options on the **General** tab of the **Result Options** dialog box (**Results > Options**).



The following table describes the results options available on the **General** tab of the **Results Options** dialog box:

Table 22 History Menu

Option	Description
After Simulation	Outputs are evaluated after a simulation is run, and signals are automatically plotted.

Table 22 History Menu (Continued)

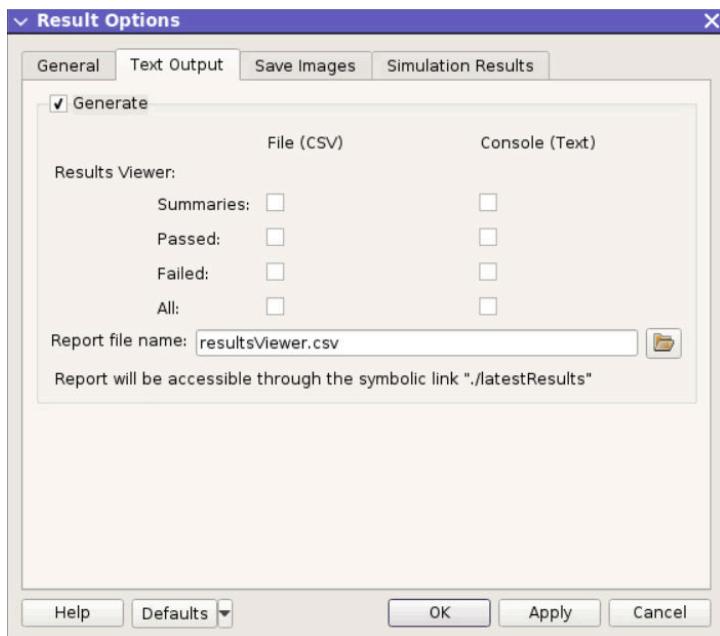
Option	Description
	<p>Plot signals for all testbenches Plots all enabled outputs for all testbenches upon simulation completion. For a large number of testbenches this can be excessive. By default this option is off.</p>
	<p>Plot signals only if a single testbench was run This option plots all enabled outputs if there is only one testbench or if only one of the multiple MTB testbenches was simulated. This is enabled by default.</p>
	<p>Show ResultsView This option automatically opens the ResultsView upon completion of the last testbench simulation. All measured results are displayed in the ResultsView across corners, sweeps, and Monte Carlo, providing quick access to that data. This is enabled by default.</p>
(Other options)	<p>Plot after hitting Esc This option controls when signals that are selected in the design are plotted directly from the schematic to the waveform viewer. If not enabled, on pressing Esc the PrimeWave Design Environment window is activated, but the signals are not plotted. If enabled, on pressing Esc the selected signals are directly plotted. The <code>saPlotAfterEsc</code> preference controls this option. This option is disabled by default; click the check box to enable.</p> <p>Group plotted signals Corresponds to the Group Signals option in the Outputs table. If enabled, all signals will be grouped in same panel of WaveView.</p> <p>Separate groups of plotted signals When plotting multiple times, if the Group Signals option is enabled, all new plots will be grouped with old plots; if Separate groups of plotted signals is also enabled, all new plots will be grouped but displayed in a separate panel.</p> <p>Parameter Display Precision This value specifies the number of significant digits to display when operating parameters are displayed.</p>

Table 22 History Menu (Continued)

Option	Description
Use Eng notation for Parameter Display	This option enables engineering notation (suffixes) for operating point parameters. If disabled, engineering notation is not used.
Enable histogram prompt	This option enables a prompt you must acknowledge before you can continue to plot more than the specified number of histograms from the Monte Carlo results. The <code>saEnableHistogramPrompt</code> preference controls this option.
Histogram limit	This value specifies the histogram limit. The <code>saHistogramLimit</code> preference controls this option.
Enable plotting scalar value in WaveView	This option allows you to plot a signal as a waveform for scalar values.
Enable auto Save/Load WaveView session	When enabled, you are prompted to save a custom plot session file when you close WaveView. The <code>saAutoSaveLoadWVSession</code> preference controls this option.
Support append plot mode for signals from different STB states	

Specifying Report Options

You can specify report options on the **Text Output** tab of the **Results Options** dialog box (**Results > Options**).



The following report options are available:

- **Generate**

When selected, this option generates reports with the attributes you choose.

- **File (CSV) and Console (Text)** check boxes.

Click the check box in the column of either of these two file output types to include the following ResultsView result types in a result file:

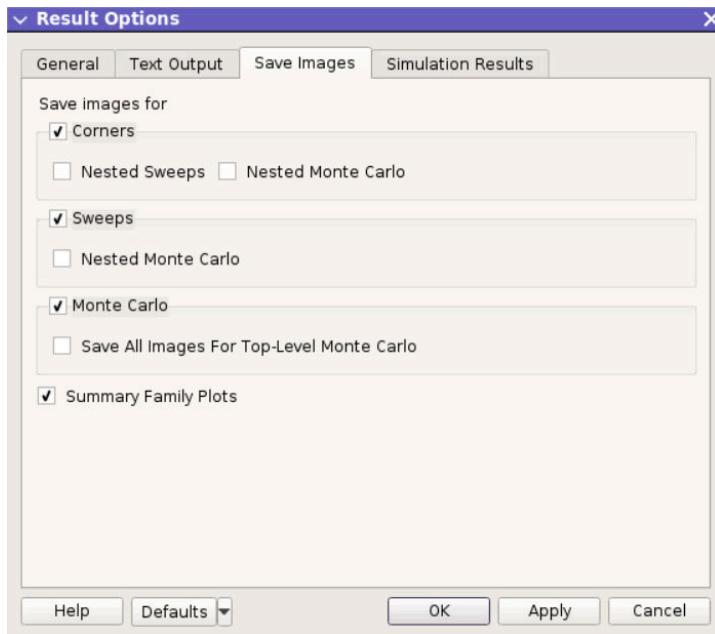
- **Summaries**
- **Passed**
- **Failed**
- **All**

- **Report file name**

By default, results are saved to `resultsViewer.csv`. You can browse to select a different report file. The report will be accessible through the symbolic link `./latestResults`.

Specifying Image Saving Options

You can save waveform images on the **Save Images** tab of the **Results Options** dialog box (**Results > Options**).



The following options are available:

- **Corners**

When selected, choose whether or not to save images for **Nested Sweeps** and/or **Nested Monte Carlo**.

- **Sweeps**

When selected, choose whether or not to save images for **Nested Monte Carlo**.

- **Monte Carlo**

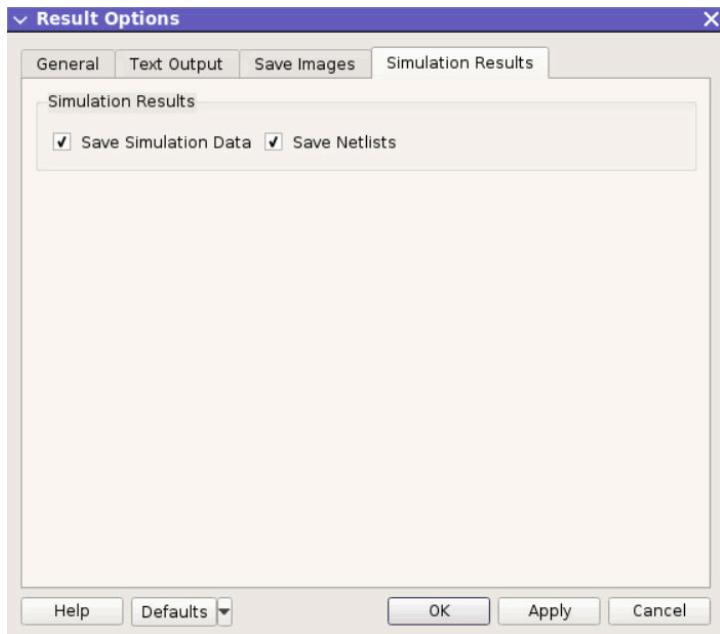
When selected, choose whether or not to **Save All Images for Top-Level Monte Carlo**.

- **Summary Family Plots**

Saves summary family plot images.

Specifying Simulation Result Options

You can specify which simulation results to save on the **Simulation Results** tab of the **Results Options** dialog box (**Results > Options**).



The following options are available:

- **Save Simulation Data**

(On by default.) When this option is off, the simulation data is not saved to disk after the simulation is done and, in the Results Viewer, the results table **Plot waveforms** button is greyed out, and the **Re-evaluate** button is disabled.

- **Save Netlists**

(On by default.) When this option is off, netlist data is not saved to disk and, in the ResultsView, the context-sensitive menu option **Display Netlist** is disabled in the results table.

Using the Timestamps Dashboard

The Timestamps Dashboard allows you to view all dependencies for creating the results from the active testbench. It is a dialog box that displays a tree of dependencies—for example, a netlist might depend on a given design, and a second netlist might depend on the first. With the Timestamps Dashboard, you can view the tree of dependencies, identify which nodes in the hierarchy tree are out of date, and execute actions to update nodes as needed.

The information displayed in the Timestamps Dashboard comes from a scan of the design hierarchy and comparison with the netlist nodes. This information is gathered during a full scan. Once a full scan has been performed, you can perform a quick scan that compares

the Results, Final Netlist, and Structured Netlist against the design data stored during the full scan.

This topic describes the following:

- [Opening the Timestamps Dashboard](#)
 - [Using Full Scan](#)
 - [Using Quick Scan](#)
 - [Managing Include Files](#)
 - [Filtering](#)
 - [Viewing the Full Design](#)
 - [Showing Only Out-of-Date Nodes](#)
 - [Executing Associated Actions](#)
-

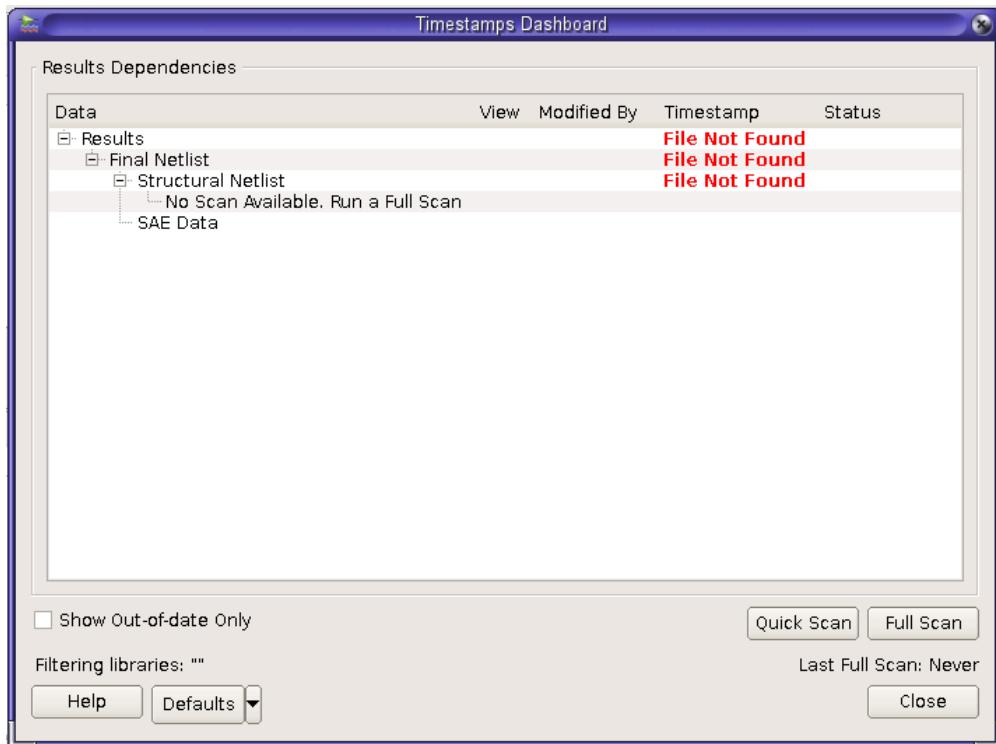
Opening the Timestamps Dashboard

To open the Timestamps Dashboard:

- ▶ Choose **Tools > Show Timestamps Dashboard** from the PrimeWave Design Environment main menu bar. The **Timestamps Dashboard** dialog box appears.

Note:

When running in MTB mode, the top of the dialog box has the option **Session: test | Testbench**, which allows you to switch testbenches in a drop-down menu without having to close the dialog box.



The Timestamps Dashboard shows the following columns:

- Data - The hierarchy of nodes that make up the results from the active testbench.
- View - The view; "schematic" is an example. In the image displayed, no scan has been run and thus no view is specified.
- Modified by - The last user to modify the node in that row. Again, no data displays because no scan has been run.
- Timestamp - Date of the most recent update. The timestamp is based on the date recorded when, for example, the tool asks if you want to save. When no scan has been run, this column displays File Not Found.
- Status - A string such as "Newer" that describes various conditions that can occur:
 - Newer
 - Has Newer Child
 - New Include

These conditions are explained in [Using Full Scan](#).

Using Full Scan

The full scan will traverse the whole design hierarchy and compare it against the netlist nodes.

After the full scan is done the following information will be stored in the testbench:

- Design data: relevant timestamps of the design hierarchy (mainly the newest nodes) and the first nodes to see which parts of the design hierarchy are out of date.
- Include File Data: all the include files used to create the final netlist are set as the compare point for future quick scans.
- Filter List: The filter preference set up at the moment of doing the Full Scan. Filter preferences are described in [Filtering](#).

A typical full scan that includes some newer design nodes resembles the following figure:

Timestamp Dashboard (TJ_LED_DESIGN/TJ_LED_TOP_TEST/schematic)				
Data	View	Modified By	Timestamp	Status
Results				
Final Netlist				
Structural Netlist				
TJ_LED_TOP_TEST	schematic	cdonoso	04/21/2016 17:56:49	
TBENCH	schematic	cdonoso	04/25/2016 14:45:55	
vpwl	hspice	cdmrg	04/25/2016 17:45:26	
TJ_LED_TOP	schematic	cdonoso	05/24/2012 15:31:37	Has Newer Child
TJ_LED_LOOP	schematic	cdonoso	04/21/2016 18:09:37	
TJ_LED_SUP	schematic	cdonoso	04/25/2016 17:43:15	Has Newer Child
nmos_5V_hp_sub_ver2	schematic	cdonoso	05/16/2012 19:44:13	
TJ_LED_TS	schematic	cdonoso	05/16/2012 19:44:13	Has Newer Child
TJ_LED_TS_CS	schematic	cdonoso	04/25/2016 17:45:31	Newer
TJ_LED_TS_COMP	schematic	cdonoso	05/16/2012 19:44:13	
TJ_LED_NAND	schematic	cdonoso	05/16/2012 19:44:11	
MT_PROBE_POINT	schematic	cdonoso	05/16/2012 19:44:13	
TJ_LED_JINV	schematic	cdonoso	05/16/2012 19:44:11	
rpmpoly3t	hspiceD	cdonoso	05/16/2012 06:26:41	
nmos_5V_hp_sub_ver2	schematic	cdonoso	05/16/2012 19:44:13	
TJ_LED_TS_SW	schematic	cdonoso	05/16/2012 19:44:13	
TJ_LED_OLP	schematic	cdonoso	05/16/2012 19:44:11	
TJ_LED_REG	schematic	cdonoso	05/16/2012 19:44:12	
TJ_BUCK_BGR_TOP	schematic	cdonoso	05/16/2012 19:44:11	

The display features the following:

- Nodes that are important within the hierarchy appear in bold.
- Nodes that are newer than a higher part of the hierarchy appear with a status of "Newer."
- Nodes that have a sub-node that is newer than a higher part of the hierarchy appear with a status of "Has Newer Child."

Note that a full scan can take some time for big designs, so the upper label around the table displays update information about the process.



Using Quick Scan

Quick scan is automatically performed when you open the Timestamps Dashboard, but is more useful after you have (at least once) performed a full scan. Much faster than full scan, a quick scan can tell you which part of the design hierarchy has newer nodes, as in the following figure:

Data	View	Modified By	Timestamp	Status
Results		cdonoso	04/21/2016 17:56:49	
Final Netlist		cdonoso	04/25/2016 14:45:59	
Structural Netlist		cdonoso	04/25/2016 17:45:26	
TJ_LED_TOP_TEST	schematic	cdonoso	04/25/2016 17:45:31	Newer
TBENCH	schematic	cdonoso	03/19/2012 13:00:41	
vpwl	hspice	cdmgrp	04/24/2016 20:39:57	
TJ_LED_TOP	schematic	cdonoso	04/25/2016 17:45:31	Newer
SAE Data				

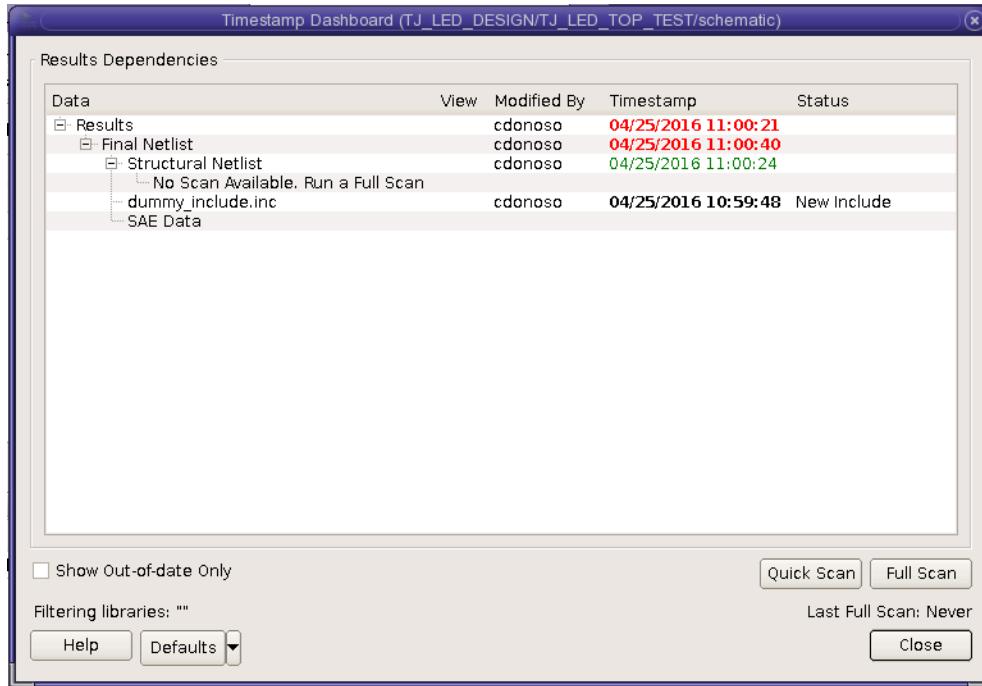
Below the table are buttons for 'Show Out-of-date Only', 'Quick Scan', 'Full Scan', 'Help', 'Defaults', and 'Close'. The status bar shows 'Last Full Scan: 04/25/2016 17:45:46'.

The nodes that have newer child nodes appear with a status of "Newer."

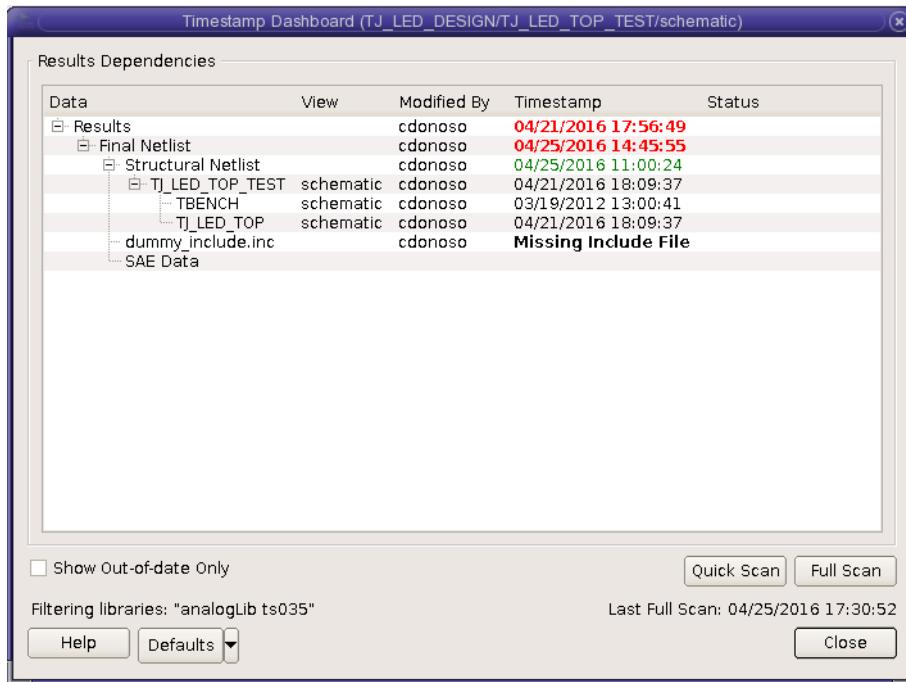
Managing Include Files

The Timestamps Dashboard will notify you when an included file is new since the creation of the final netlist, or has disappeared since then.

If a new include file appears, the file appears in the Timestamps Dashboard with a status of "New include."



If an include file has gone missing, the file appears in the Timestamps Dashboard with a status of "Missing Include File."



Filtering

It is possible to filter the nodes that are checked by a full scan, using the preference `saTimestampDashboardFilterList`. For example, to exclude nodes that match "analogLib" and "ts035", you would use the following command:

```
db::setPrefValue saTimestampDashboardFilterList -value "analogLib ts035"
```

The default filtering preference is an empty string (""). Once you have set a non-empty filtering preference, there will be a difference between the preference and what is stored in the testbench. A warning appears:

```
PrefFilter list differs from stored filter list
(Stored was ""; pref has "analogLib").
Please run Full Scan to update.
```

To reset the filter, set the preference back to an empty string:

```
db::setPrefValue saTimestampDashboardFilterList -value ""
```

Viewing the Full Design

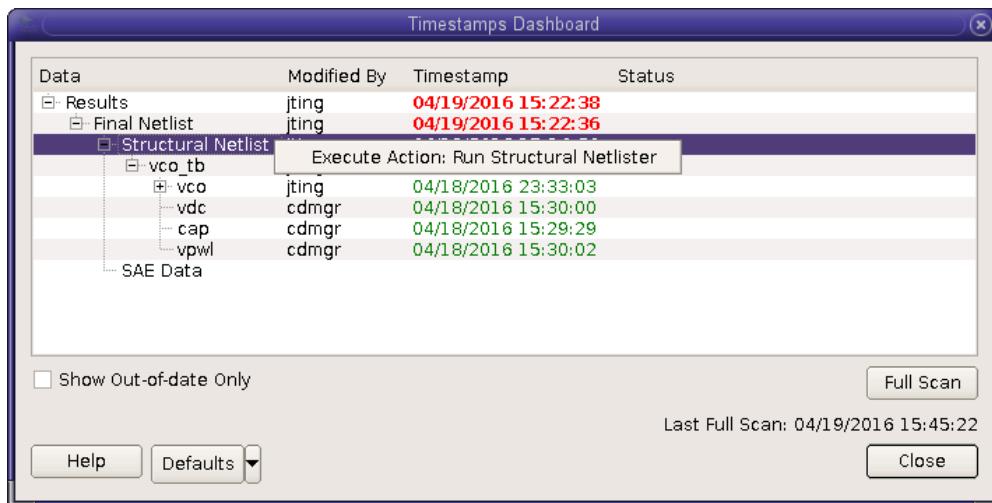
The initial view in the Timestamps Dashboard shows only a condensed view of the hierarchy. To view the entire design hierarchy, click the **Full Scan** button.

Showing Only Out-of-Date Nodes

In a given design, there can be many nodes in the hierarchy and you might want to view only the out-of-date nodes. To do this, check the box next to the label **Show Out-of-date Only**.

Executing Associated Actions

Each node in the Data column can have an associated action; this represents the action needed to reconstruct that node. For example, a Structural Netlist node might have the action **Run Structural Netlister**. The action appears in a context menu.



To view and execute the associated action for a given node, right-click the node. If no action is associated with the node, the menu shows **Execute Action**, but disabled.

18

Running Advanced Analyses

This chapter contains information on how to run advanced analyses.

This chapter contains the following major sections:

- [Setting Up Parametric Analyses](#)
 - [Setting Up Corner Analyses](#)
 - [Setting Up Aging Analyses](#)
 - [Setting Up Monte Carlo Analyses](#)
 - [Setting Up Monte Carlo Analyses for FineSim](#)
 - [Using Sigma Amplification in Monte Carlo Analysis](#)
 - [Setting Up Worst-Case Analysis](#)
 - [Setting Up Chained Testbenches](#)
 - [Computing Interface Elements](#)
 - [Preparing for Multi-Technology Simulation of 3-D Integrated Circuits \(3DICs\)](#)
-

Setting Up Parametric Analyses

The Parametric Analysis tool is an interactive analysis that measures performance by simulating a circuit under varying conditions. You can use Parametric Analyses to define one or more nested sweeps, and you can vary the value of a design variable for each sweep.

To set up a parametric analysis, choose **Tools > Parametric Analyses** from the PrimeWave Design Environment main menu bar.

Note:

You can use design variables in your Parametric analysis to specify a range or an individual value, which can be used to set up the multiple netlist runs.

PrimeWave Design Environment resolves design variables to numerical values. If the value is valid, a series of netlists are generated in the range of sweeps specified for simulation. Variables must be defined in your testbench and not via include files, because the PrimeWave Design Environment calculates the sweeps before sending the individual iterations to the simulator.

The following topics are included in this section:

- [Adding New Sweeps](#)
- [Editing Sweeps](#)
- [Deleting Sweeps](#)
- [Correcting Invalid Sweeps](#)
- [Performing Parametric Analyses](#)

 [Show me](#) how to run a parametric analysis.

Adding New Sweeps

You can add sweeps to your designs directly into the **Design Variables** table on the PrimeWave Design Environment tab page or via the **Parametric Analyses** dialog box:

- [Adding Sweeps to the Design Variables Table](#)
- [Adding Sweeps Using the Parametric Analyses Dialog Box](#)

Adding Sweeps to the Design Variables Table

To add a sweep directly into the **Design Variables** table on the PrimeWave Design Environment tab page:

1. Ensure the PrimeWave Design Environment window is forward.
2. Ensure any design variables for which you want to add sweeps are added to the **Design Variables** table.
3. Click in the **Value** column table cell for the design variable you want to add a sweep.
4. Enter a sweep range or sweep points of interest.

For a sweep range, enter values separated by colons (1:2:3, for example). The first value is the starting value, the second value is the step value, and the third value is the stopping value.

For points of interest, enter values separated by commas (1,2,3, for example).

You can also include values with scientific notation, values of varying units of measure, and design variables as part of your sweeps.

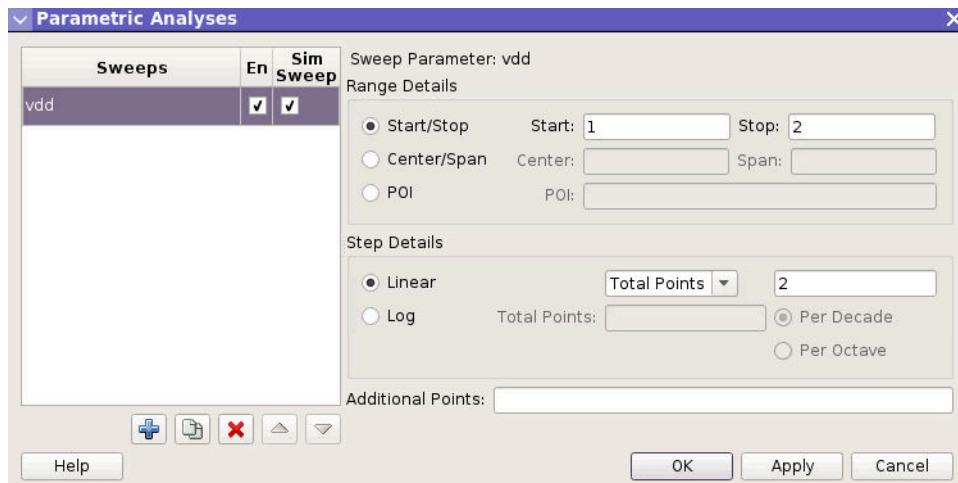
The  icon appears once you enter sweep values. All sweeps are automatically populated in the **Parametric Analyses** dialog box.

Adding Sweeps Using the Parametric Analyses Dialog Box

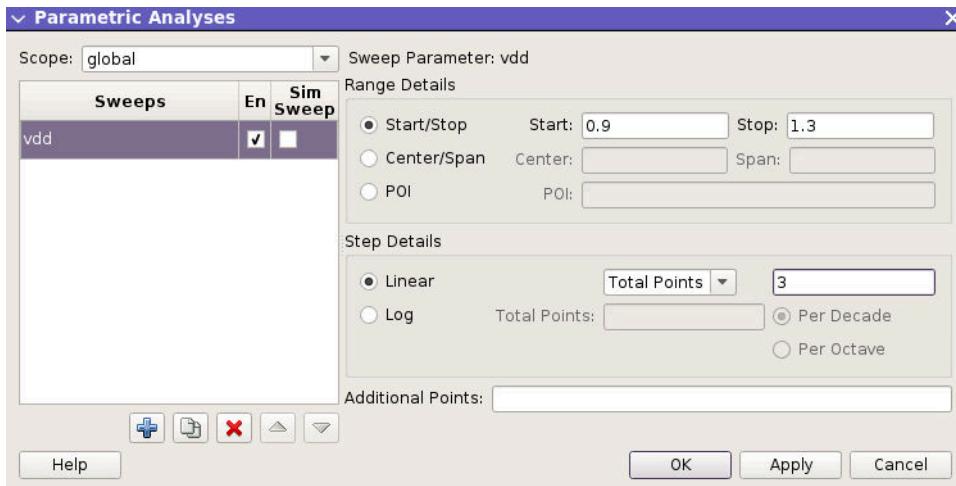
To add a new parametric sweep:

1. Choose **Tools > Parametric Analyses** from the PrimeWave Design Environment main menu bar.

The **Parametric Analyses** dialog box opens.

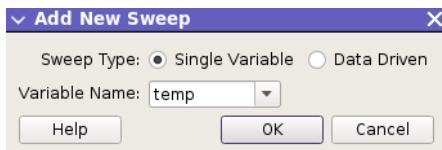


2. If you are working in a test suite with multiple testbenches, choose **global** or the name of a testbench from the **Scope** menu to add a sweep to all testbenches or a specific testbench, respectively.



- Click the **Add New Sweep** button to add a new sweep.

The **Add New Sweep** dialog box opens.



- Select a sweep type: **Single Variable** or **Data Driven**.

If you choose **Single Variable**, select a variable from the **Variable Name** menu that you already defined in your testbench, or enter the name of a new variable.

If you choose **Data Driven**, choose or enter the number of **Estimated Iterations**. Also, choose which variables you want to include in the sweep, or enter the name of a new variable.

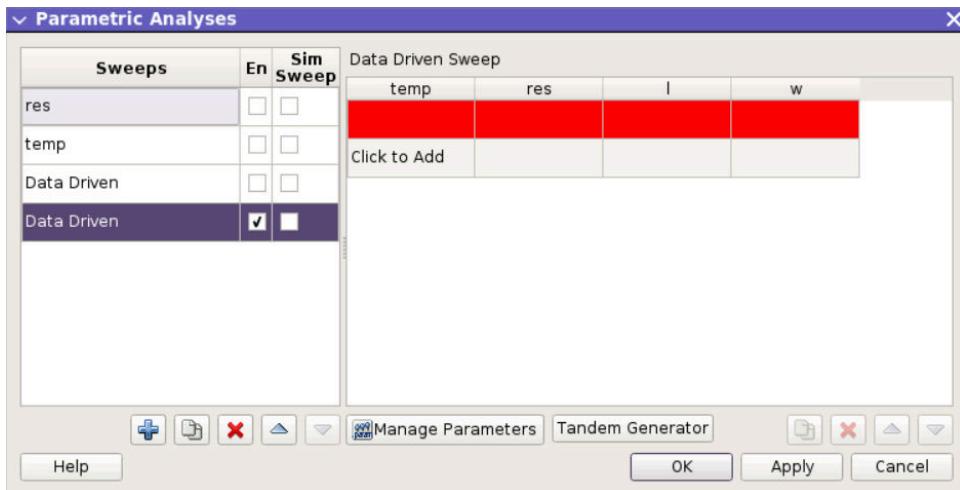
Note:

Added variables apply only to the sweep you are currently setting up.

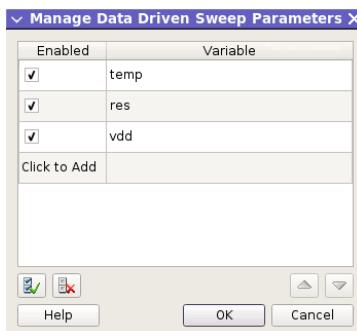
- Click **OK** to add the new sweep.

The new sweep is added to the **Sweeps** table in the **Parametric Analyses** dialog box.

Cells in the **Data Driven Sweep** table are colored red when no variable value is specified yet.



6. (Optional) For a data-driven sweep, you can click **Manage Parameters** to open the **Manage Data Driven Sweep Parameters** dialog box.



Enter the necessary information and click **OK**.

7. (Optional) For a data-driven sweep, you can click **Tandem Generator** to open the **Manage Tandem Parameters** dialog box. This allows you to specify parameter sets for use in multiple simulation runs. A parameter set consists of comma- or space-separated values for a sweep parameter. Each parameter sets must have the same number of input values.



Enter the necessary information and click **OK**. (For more information, see [Using the Tandem Parameter Generator](#).)

8. In the **Parametric Analyses** dialog box, click the check box in the **En** (enable) column of your created sweep if you want to include the sweep as part of your testbench.
9. (Optional) Click the check box in the **Sim Sweep** column for your sweep to create a single netlist that uses the internal simulator sweep syntax.

If you do not click the **Sim Sweep** check box, a separate netlist is created for each sweep point.

Note:

You might consider enabling this option if you have a fast simulation with small signals; otherwise, longer transient simulations might be better when run on the grid.

10. Select the **Range Details** and corresponding values for any **Single Variable** sweeps you create: **Start/Stop**, **Center/Span**, or **POI** (Point of Interest).

If you have no **Single Variable** sweeps in your test suite, continue to [Step 13](#).

- **Start/Stop**

This sweep is defined by specifying start and stop values.

- **Center/Span**

This sweep is defined by specifying the Center point for the sweep, and a Span on either side of the center point. For example, a sweep with a center point of 25 and a span of 1 covers 24 to 26.

- **POI**

This sweep encompasses discrete points.

11. If you select **POI**, skip to [Step 14](#). Otherwise, you must provide **Step Details** information when you select the **Start/Stop** or **Center/Span** sweep type.

You can specify steps as either **Linear** or **Log** (logarithmic). If you opt for a linear sweep, you can specify either the total number of points or the step size. If **Total Points** is specified, the sweep is performed in equal steps across the specified range. If the **Step Size** is specified, sweep points using that step size are generated. The last point is located on the upper range limit even if it falls under the extent of the last step.

12. (Optional) Enter any additional points you want to include in the **Additional Points** text box.
13. Enter values for each variable of any **Data Driven** sweeps you create.

Cells in the **Data Driven Sweep** table are colored red when no variable value is specified yet.

14. Click **Apply** to save your changes.

The sweeps are now included in the testbench, and any existing sweeps are updated. If you have any errors, see [Correcting Invalid Sweeps](#).

Editing Sweeps

To edit an existing sweep:

1. Choose **Tools > Parametric Analyses** from the PrimeWave Design Environment main menu bar.

The **Parametric Analyses** dialog box opens.

2. If you are working in a test suite with multiple testbenches, choose **global** or the name of a testbench from the **Scope** menu to edit a sweep for all testbenches or a specific testbench, respectively.

If you are not working in a test suite with multiple testbenches, continue to the next step.

3. Select the existing sweep that you want to change in the **All Sweeps** pane.
4. Adjust the values for the **Sweep Details** and **Step Details** as needed.
5. Click **Apply** to save your changes.

Deleting Sweeps

To delete a sweep:

1. Choose **Tools > Parametric Analyses** from the PrimeWave Design Environment main menu bar.

The **Parametric Analyses** dialog box opens.

2. If you are working in a test suite with multiple testbenches, choose **global** or the name of a testbench from the **Scope** menu to delete a sweep for all testbenches or a specific testbench, respectively.

If you are not working in a test suite with multiple testbenches, continue to the next step.

3. Select the sweep you want to delete in the **All Sweeps** pane.
 4. Click **X** just below the **Sweeps Pane** to delete the sweep.
-

Correcting Invalid Sweeps

If you see an exclamation point (!) displayed just before a sweep in the list of sweeps, then that sweep is currently invalid. Invalid sweep fields are also highlighted in red. See the error message that is displayed just below the list of sweeps for information on how to fix the problem, or hold the pointer over the sweep name or value(s) to display the problem.

The following cases are considered validation errors:

- The same variable is used in multiple enabled sweeps. For example, if you enable a temp sweep, as well as a data-driven sweep that includes a temp variable, the sweep is invalid.
- More than one data-driven sweep is enabled.

Note:

If you click the **Sim Sweep** check box for a sweep, all the inner sweeps are automatically checked, and then the simulator sweep is used.

If more than one sweep is invalid, an error message is displayed only for the first sweep in the list of sweeps that is invalid. Error messages for remaining invalid sweeps are displayed as the previous sweep errors are corrected. Sweeps with errors are not applied to testbenches.

Using the Tandem Parameter Generator

For a data-driven sweep, you can use the Tandem Parameter Generator to specify parameter sets for use in multiple simulation runs. Parameter sets help save time and also provide you the flexibility to run a specific set of variables.

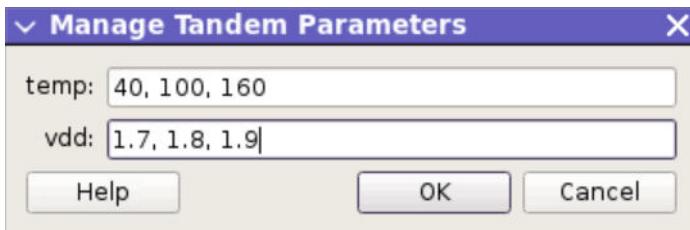
A parameter set consists of comma- or space-separated values for a sweep parameter. When you create a parameter set by combining two or more variables, only a selected set of sweep combinations are created by picking values from the same ordinal position for all the variables or parameters in the set. All parameter sets must therefore have the same number of input values.

In the **Parametric Analyses** dialog box, click **Tandem Generator** to open the **Manage Tandem Parameters** dialog box.

For example, specify the following values for the sweep variables vdd and temp:

temp = 40, 100, 160

$vdd = 1.7, 1.8, 1.9$



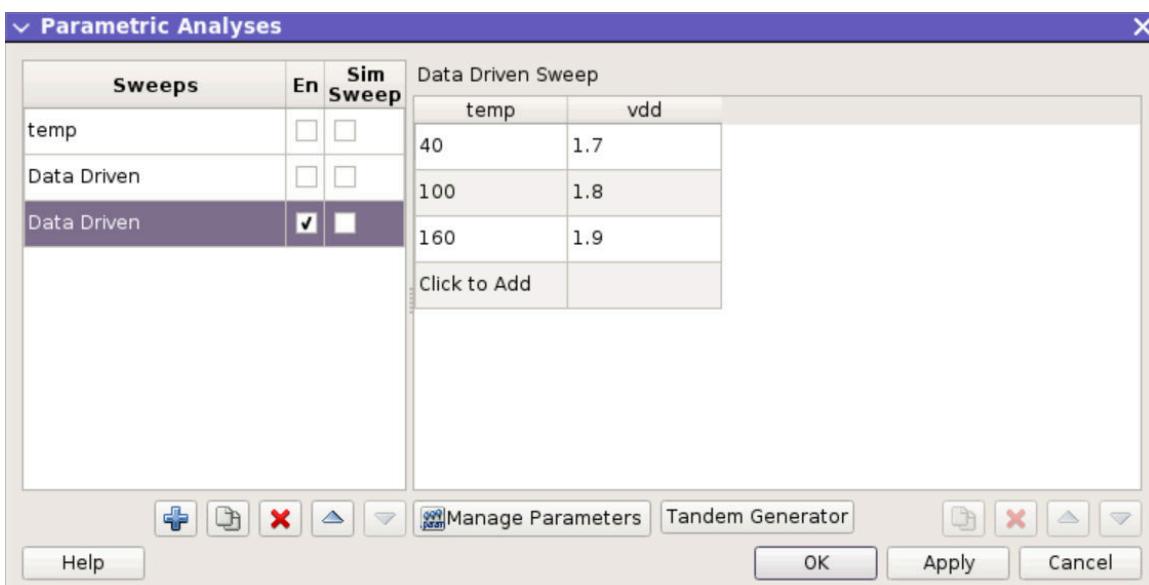
Notice that each parameter set has 3 input values.

Click **OK** in the **Manage Tandem Parameters** dialog box to populate the **Parametric Analyses** dialog box for the following simulations:

Simulation 1: $vdd=1.7$, $temp=40$

Simulation 2: $vdd=1.8$, $temp=100$

Simulation 3: $vdd=1.9$, $temp=160$



Click **OK** and run the simulation. The three simulation runs will use the parameter values you specified using the Tandem Parameter Generator.

Performing Parametric Analyses

Once a parametric analysis is specified, it is displayed in the Analysis pane of the PrimeWave Design Environment window along with other analyses. You can enable and disable these analyses by toggling the check box in the En column of the analysis tree.

When a parametric analysis is enabled, choosing **Simulation > Netlist and Run** or **Simulation > Run** runs a parametric analysis. Messages appear in the Console indicating that multiple netlists are being created, and multiple simulations are being launched. If simulation license availability is an issue, be sure to use the license queuing option of your simulator.

Printing and Annotating Parametric Results

When printing or annotating parametric data, choose a single iteration whose data you want to print or annotate, and use printing and annotation commands to produce iteration selectors. When printing and annotating transients, the selectors are added to the dialog boxes that are invoked for you to select the transient timepoint of interest.

Setting Up Corner Analyses

A corner analysis measures circuit performance by simulating a circuit with parameters that represent the most extreme variations in the manufacturing process.

To access the corners tool, choose **Tools > Corners** (or **Tools > Global Corners** if you are working with multiple testbenches) from the PrimeWave Design Environment menu.

Note:

You can use design variables in your corner analysis to specify a range or an individual value, which can be used to set up the multiple netlist runs. The PrimeWave Design Environment resolves design variables to numerical values. If the value is valid, a series of netlists are generated in the range of corners specified for simulation. Variables must be defined in your testbench and not via include files, because the PrimeWave Design Environment calculates the sweeps before sending the individual iterations to the simulator.

The following topics are included in this section:

- [Specifying Corner Parameters](#)
- [Defining Corners](#)
- [Specifying Corner Ranges](#)

- [Specifying Corner Variables in Tandem](#)
- [Setting Up a Corner Sweep Through Multiple Cellviews](#)
- [Removing Corner Parameters](#)
- [Enabling and Disabling Corner Parameters](#)
- [Deleting Corners](#)
- [Copying Corners](#)
- [Removing Duplicate Corners](#)
- [Enabling and Disabling Corners](#)
- [Changing Corner Values](#)
- [Renaming Corners](#)
- [Enabling or Disabling the Corner Analysis Setup](#)
- [Setting up Specifications for Each Corner Condition](#)
- [Sorting Corners](#)
- [Defining Corner Groups](#)
- [Importing and Exporting Corner Information to a File or State](#)

 [Show me](#) how to run corner simulation.

Specifying Corner Parameters

See the following sections for information on specifying design variables or model files:

- [Specifying Design Variables for Corners](#)
- [Specifying Model Files for Corners](#)

Specifying Design Variables for Corners

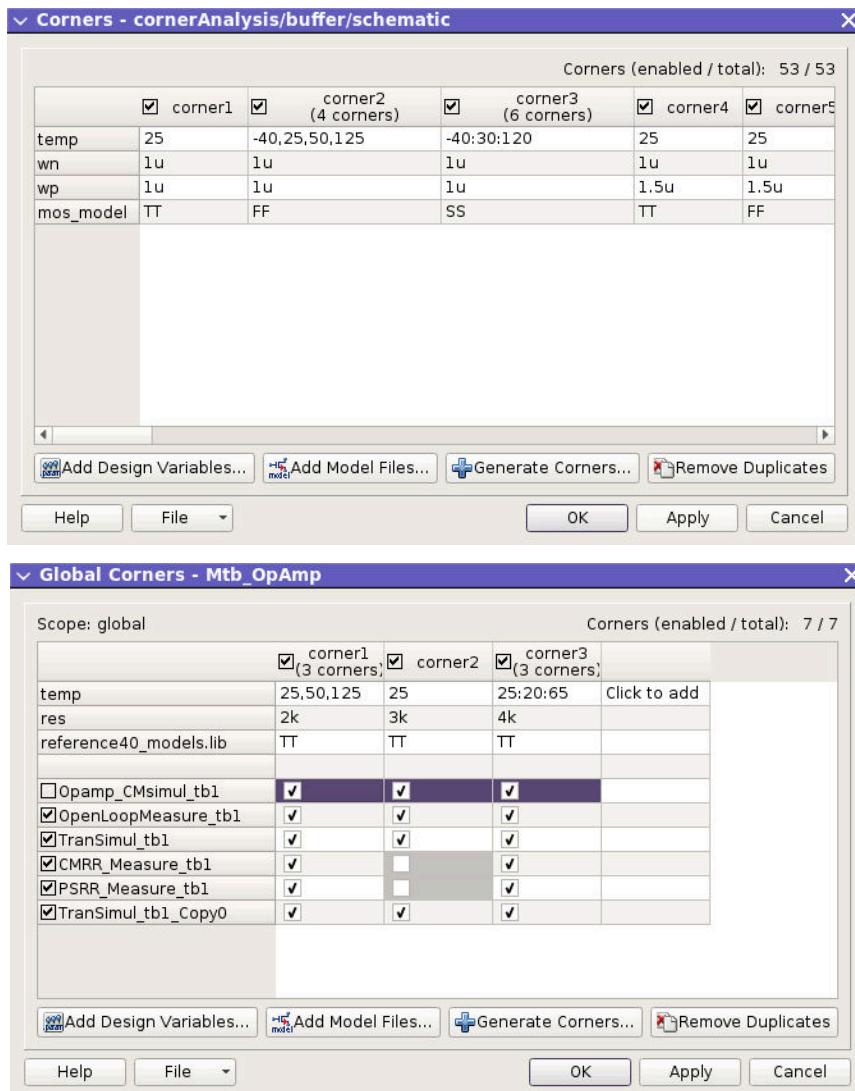
To specify one or more design variables to a group of existing corners:

1. Choose **Tools > Corners** (or **Tools > Global Corners** if you are working with multiple testbenches) from the PrimeWave Design Environment menu.

The **Corners** or **Global Corners** dialog box appears, respectively.

Chapter 18: Running Advanced Analyses

Setting Up Corner Analyses

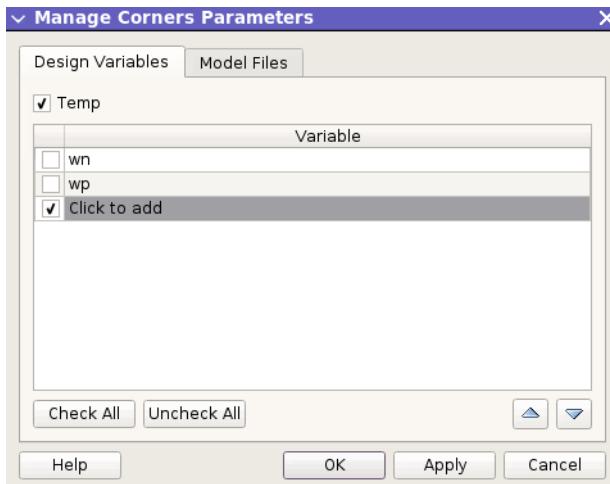


Note:

Corners setup supports expressions in this dialog box. Expressions are netlisted as-is for the simulator to evaluate. This allows them to be evaluated in the correct context. The ResultsView and HTML report show the expression, not the evaluated value, for the dependent parameter.

2. Click Add Design Variables.

The **Manage Corners Parameters** dialog box appears with the **Design Variables** tab in front.



Any design variables that are already set up in your testbench appear in the list of available variables.

3. Add any new design variables you want to include in corners definitions.

See [Adding and Editing Design Variables](#).

You can use design variables in the **Generate Corners** dialog box (see [Defining Corners](#)). For example, when defining wp as a point-of-interest (POI) value, you might want to use another design variable, such as wn, and define POI value for wp as $1.5*wn$, $1.95*wn$, $1.5*wn$, $1.95*wn$. This is an example of using another design variable when defining range, or POI for some other design variable.

4. Click the check box next to each design variable you want to include in your corners.
5. Click the **Temp** check box to include the previously specified temperature with your design variables.

The temperature is enabled by default. See [Specifying Temperature](#).

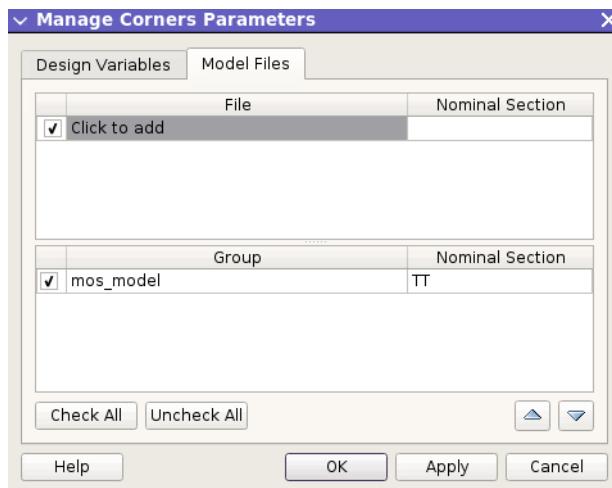
6. Click **OK** to save your changes.

Specifying Model Files for Corners

To specify one or more model files to a group of existing corners:

1. Choose **Tools > Corners** from the PrimeWave Design Environment menu.
The **Corners Setup** dialog box appears.
2. Click **Add Model Files**.

The **Manage Corners Parameters** dialog box appears with the **Model Files** tab in front.



Any model files that are already specified in your testbench appear in the list of available model files. (Model files are automatically added to corners by default if they are set up in the **Model Files** setup dialog box.)

Any model required in the netlist must be enabled in the **Corners Setup** dialog box, even if its section is not changing.

Note:

Model files added in the **Manage Corners Parameters** dialog are not reflected back in the PrimeWave Design Environment main **Model Files** setup dialog box.

3. Enable any model files that you want to include.

You can also add model files in this dialog box. See [Specifying Model Files](#).

4. Click **OK** to save your changes.

Defining Corners

You can define a single corner, or you can define multiple corners at once.

Corners can be added in the following ways:

- [Adding a Single Corner](#)
- [Adding Multiple Corners](#)

Adding a Single Corner

Choose **Tools > Corners** (or **Tools > Global Corners**) if you are working with multiple testbenches with a global scope) from the PrimeWave Design Environment menu. The **Corners** or **Global Corners** dialog box appears, respectively.

Click **Click to Add**, which is the last unused column on the right side of the corners table. A single corner is created with an automatically assigned name and blank values. You can edit the values as described in the [Changing Corner Values](#) section.

Adding Multiple Corners

When defining multiple corners at once, you can use one of the following methods:

- Configure and generate corners from the **Generate Corners** dialog box.
- Create corner groups to specify the range of variables, which is similar to the **Generate Corners** dialog box except that the range is deferred until netlisting. This method allows for easier range editing if needed.

To define multiple corners at once:

- Choose **Tools > Corners** (or **Tools > Global Corners**) if you are working with multiple testbenches with a global scope) from the PrimeWave Design Environment menu.

The **Corners** or **Global Corners** dialog box appears, respectively. Values, including range specifications used to calculate corners, are displayed in the table if you already created corners; otherwise, the table is empty.

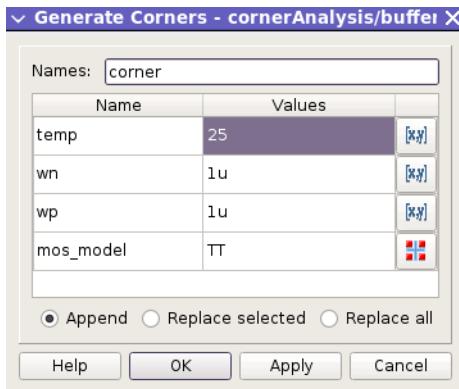
The default prefix for corner names are displayed in the **Names** field. This prefix is used as the base of each corner name in sequential order (corner_name, corner_name1, corner_name2, corner_name3, for example). You can also add specific space-separated names for your corners. The last entry is used as a prefix when you have more corners than entries.

Note:

Spaces are not allowed in corner names.

- Click the **Generate Corners** button.

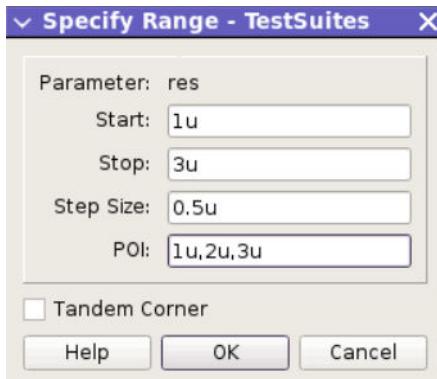
The **Generate Corners** dialog box appears.



3. If you are not including any design variables or the temperature, skip to step [Step 7](#).

Otherwise, click to specify ranges for the design variables or temperature. This range notation is used to calculate all possible combinations of values. If you have not yet specified design variables for any corners, see [Specifying Design Variables for Corners](#).

The **Specify Range** dialog box opens.



4. Enter values for the range starting, stopping, and step value. You can specify points of interest (POI) within or outside of the sweep.

Note:

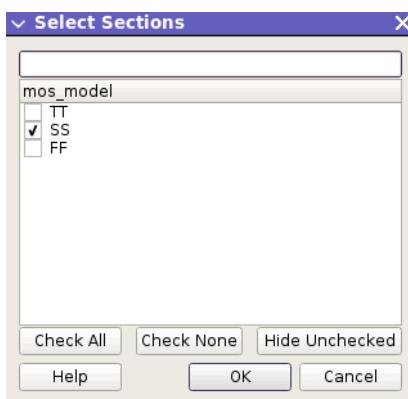
When you enter points of interest, separate values with commas (,).

You can also input these values directly into the **Values** column of a design value or the temperature in the **Generate Corners** dialog box using the following syntax:

```
<start>:<step>:<stop>, poi1, poi2, poi<n>
```

5. (Optional) Check **Tandem Corner** to specify corner variables in tandem. See [Specifying Corner Variables in Tandem](#).
 6. Click **OK** to save the corner range values.
- Repeat [Step 3](#) through [Step 6](#) for each additional design variable.
7. If you are not adding any corners for model files, continue to [Step 10](#). Otherwise, click  to select one or more sections in the model files to use for the corners you are adding.

The **Select Sections** dialog box appears.



If you have not yet specified model files for any corners, see [Specifying Model Files for Corners](#).

8. Choose one or more sections of model files to include.
9. Click **OK** to save your changes in the **Select Sections** dialog box.

The **Select Sections** dialog box closes.

10. Click **OK** in the **Generate Corners** dialog box to generate corners.

If the parameter ranges that you specify exceed 200 corners, you are asked to confirm the corner generation.

Specifying Corner Ranges

To specify the range for a corner:

1. Ensure the **Corners** dialog box is open (**Tools > Corners** or **Tools > Global Corners**).
A table is displayed, which contains the existing corners for the testbench(es).
2. Click in a row cell for the variable and corner for which you want to specify a range.

The **Specify Range** button appears.

3. Click the **Specify Range** button .

The **Specify Range** dialog box opens.

4. Enter values for the **Start** and **Stop** time, **Step** size, and **POI** (points of interest) as needed.

Note:

When you enter points of interest, separate values with commas (,).

5. (Optional) Check **Tandem Corner** to specify corner variables in tandem. See [Specifying Corner Variables in Tandem](#).

6. Click **OK** to save your changes.

The range values are saved for the corner.

Specifying Corner Variables in Tandem

When there are three corner variables each having three different values, the Custom Compiler tool expands that to 3^3 or 27 possible corners. If you would prefer to have only three corners, you can use the **Tandem Corner** option to expand corners with the selected variables in tandem.

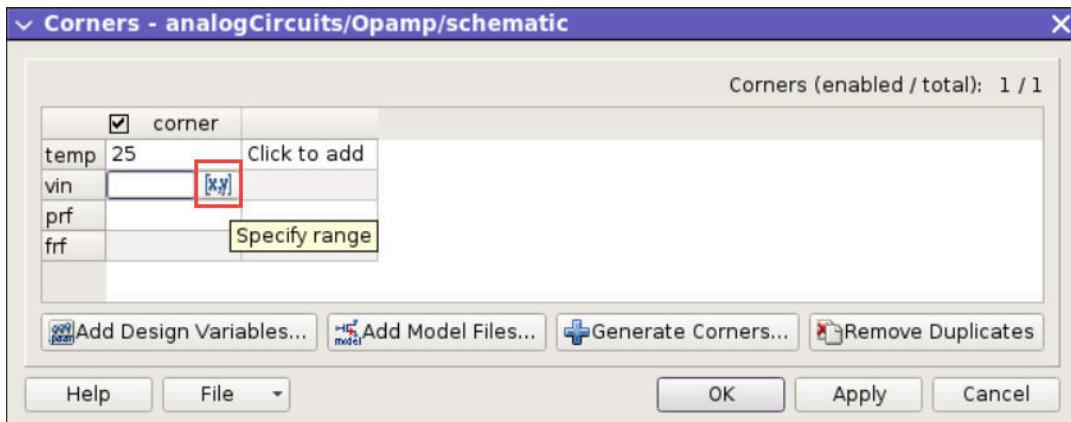
For example, assume there are three variables, as follows:

- var1, which can take values 1, 2, and 3
- var2, which can take values a, b, and c
- var3, which can take values x and y

If you select var1 and var2 to expand in tandem, the result will be 3 corners where the first value of var1 goes with the first value of var2, the second value of var1 goes with the second value of var2, and so on. However, if you select var1, var2, and var3 to expand in tandem, an error is reported because var3 only has 2 possible values.

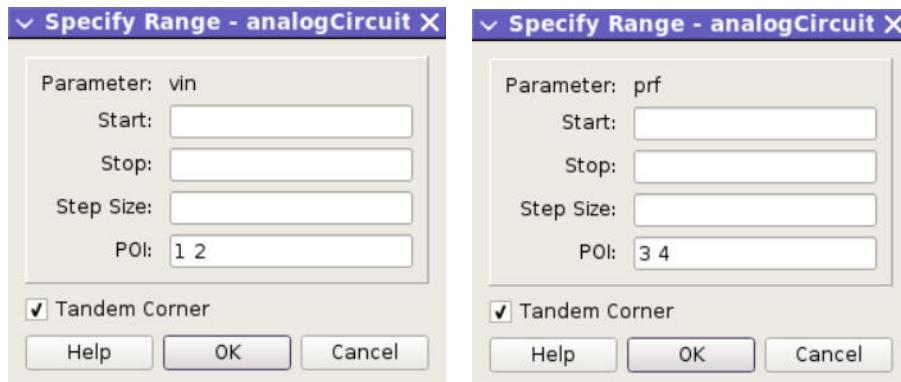
To specify corner variables in tandem:

1. Click  to open the **Specify Range** dialog box, either from the variable table cell in the **Corners** (or **Global Corners**) setup dialog box or from the **Generate Corners** dialog box.

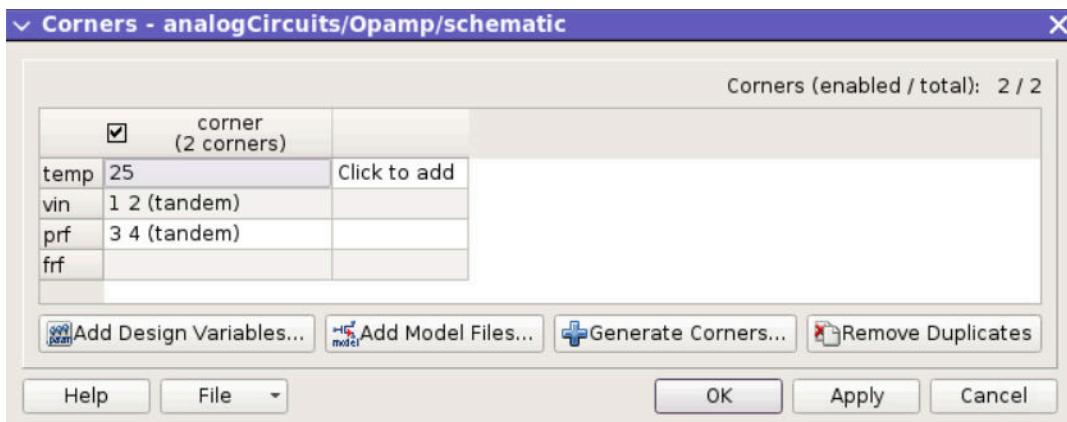


The **Specify Range** dialog box opens.

2. Specify several values for the corner variable and check the **Tandem Corner** option.



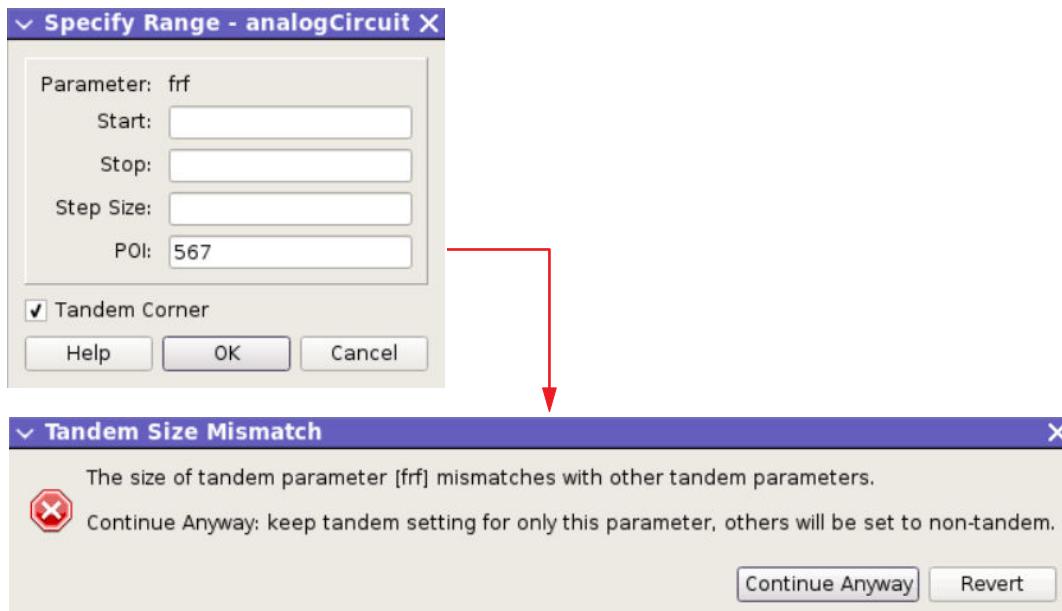
Click **OK** to close the **Specify Range** dialog box. Notice that the tandem corners are set up in the **Generate Corners** dialog box.



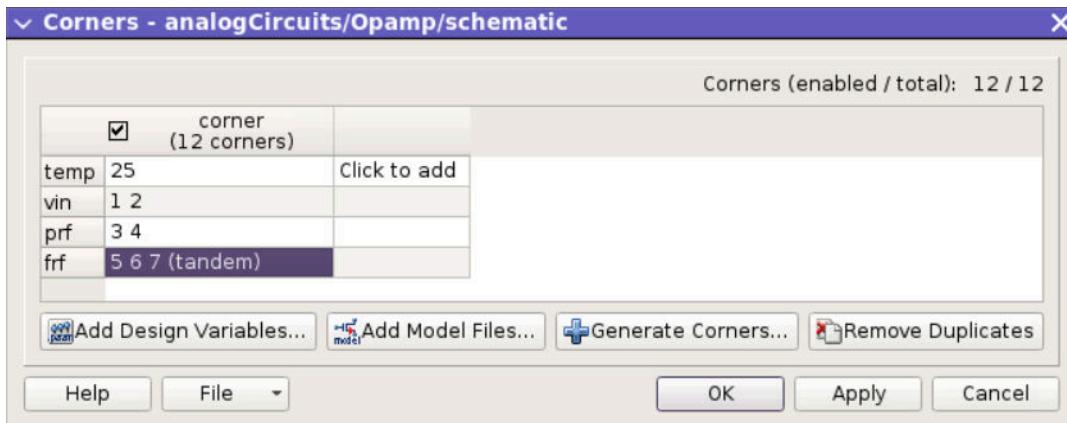
Note:

Alternatively, in the **Specify Range** dialog box, you could type a series of space-separated values and add "(tandem)" at the end.

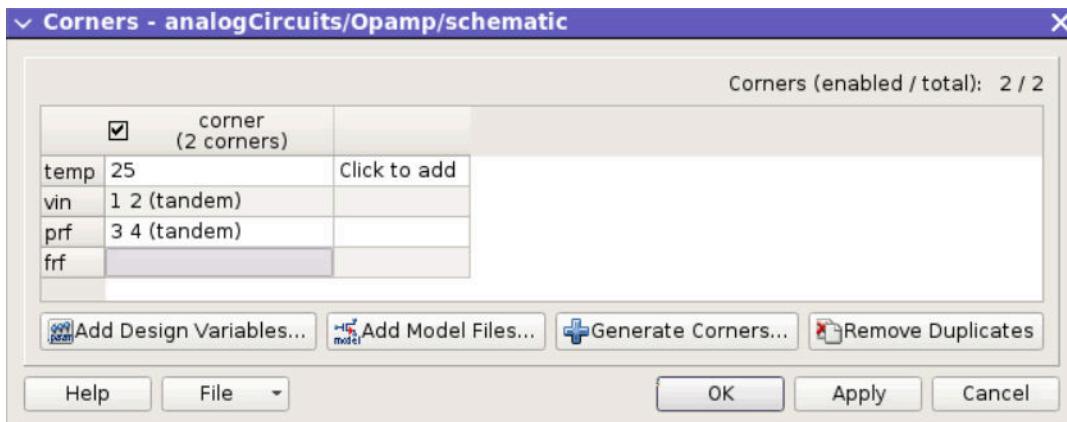
3. If you specify a number of values for the corner variable that does not match other tandem corners, a message appears warning you of a **Tandem Size Mismatch**.



If you click **Continue Anyway**, the current parameter is set for tandem corners and the other parameters are set to non-tandem.



If you click **Revert**, the tandem setting and the setting value revert to the previous setting.



If you set up the parameters from the **Generate Corners** dialog box, the **Corners** dialog box is updated when you click **OK** or **Apply**.

If there is a tandem size mismatch, you cannot set up the corners until you resolve the mismatch.

Setting Up a Corner Sweep Through Multiple Cellviews

You can select multiple cellviews or blocks on which to perform a corner sweep. You can specify a cell name and select from its available views to sweep. The default set of valid views for sweeping are schematic, text, config, and extracted views. You can control that set with the Tcl preference `saSweepCellViewTypes`.

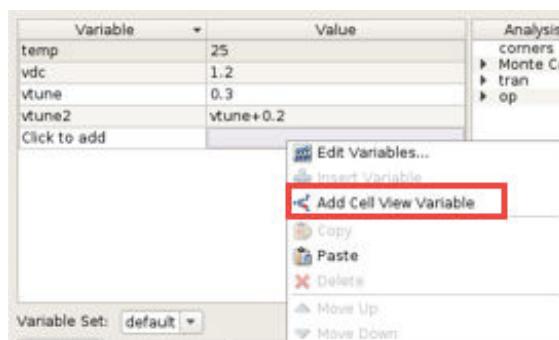
This feature is supported for corner sweeps.

For example, CellA has the following views, and the first five could be selected for the corner sweep:

- schematicA
- schematicB
- starrc_100C_fast
- starrc_0C_slow
- veriloga
- symbol
- sae_state

To select the desired cellviews:

1. Define the cellview as a variable in the **Variables** pane (in STB or MTB) by right-clicking and choosing **Add Cell View Variable** from the menu.



2. Select the cell from the schematic design or, if the **Add Cell View Variable** dialog box was invoked from the Global Variables pane in MTB, select the **Library**, **Cell**, and **View** in the **Add Cell View Variable** dialog box.

Chapter 18: Running Advanced Analyses

Setting Up Corner Analyses

Variable	Value
temp	25
vdc	1.2
vtune	0.3
vtune2	vtune+0.2
DemoVco/amp	schematic
Click to add	

Once the cellview is defined as a variable, it is available in the **Corners** dialog box to select views and **Generate Corners**.

	corner6	corner7	corner8	corner9	corner10
temp	25	25	25	25	25
vtune	0.3	0.3	0.3	0.3	0.3
DemoVco/amp	starrc_100C_fast	schematicB	schematicA	spf	spice

Removing Corner Parameters

To remove corner parameters:

1. Ensure the **Corners** dialog box is open (**Tools > Corners** or **Tools > Global Corners**).
A table is displayed that contains the existing corners for associated design variables, model files, or temperature.
2. Right-click anywhere in the row of the design variable, model file, or temperature parameters you want to delete.
3. Choose **Delete Parameters** from the menu that opens.

The selected design variable, model file, or temperature parameter is removed and is now not part of your corner analysis.

You can restore removed parameters by right-clicking any existing parameter and choosing **Manage Parameters** from the menu that opens. In the **Manage Corners Parameters** dialog box, click the check box next to any design variable or model file you want to restore, then click **Apply**. The re-enabled design variable or model file is once again part of your corner setup.

Enabling and Disabling Corner Parameters

To enable a corner parameter, right-click a parameter in the **Corners** dialog box (**Tools > Corners** or **Tools > Global Corners**), and choose **Manage Parameters** from the menu that opens. Click the check box next to any disabled design variables, model files, or the temperature you want to re-enable.

To disable a parameter, ensure the check box next to an enabled design variable, model file, or the temperature is unchecked.

Deleting Corners

To delete one or more corner:

1. Ensure the **Corners** dialog box is open (**Tools > Corners** or **Tools > Global Corners**).
A table is displayed that contains the existing corners for the testbench.
2. Highlight one or more corner table cells, and right-click anywhere on the highlighted corners.

Caution:

You cannot undo this operation. If you mistakenly delete a corner, click **Cancel** in the **Corners Setup** dialog box, and open the **Corners Setup** dialog box again.

3. Choose **Delete Corners** from the menu that opens.

All of the highlighted corners are deleted.

4. Click **Apply** to save your changes.

Copying Corners

To copy one or more corners:

1. Ensure the **Corners** dialog box is open (**Tools > Corners** or **Tools > Global Corners**).

A table is displayed that contains the existing corners for the testbench.

2. Click one or more corner names in the table, and right-click anywhere on the highlighted corners.

To select more than one corner, **Ctrl+click** the names of the corners.

3. Choose **Copy Corners** from the menu that opens.

All of the highlighted corners are copied and added to the column at the far right side of the table.

Removing Duplicate Corners

To remove all duplicate corners:

1. Ensure the **Corners Setup** dialog box is open (**Tools > Corners** or **Tools > Global Corners**).

A table is displayed that contains the existing corners for the testbench.

2. Click **Remove Duplicates**.

A dialog box confirming the removal of corners opens with a list of all the duplicates to be deleted.

Note:

All duplicates listed are deleted. Duplicates cannot be deleted individually in this dialog box. If you want to remove individual corner values, see [Deleting Corners](#).

3. Click **OK** to remove all duplicate corners.
-

Enabling and Disabling Corners

Note:

Only enabled corners are run as part of the corner analysis.

You can enable/disable a single corner, or you can enable/disable multiple corners.

Corners can be enabled or disabled in the following ways:

- [Enabling and Disabling Single Corner](#)
- [Enabling and Disabling Multiple Corners](#)

Enabling and Disabling Single Corner

To enable or disable a single corner, ensure that the **Corners** dialog box is open. Choose **Tools > Corners** or **Tools > Global Corners** to open the **Corners** dialog box.

A table is displayed that contains the existing corner for the testbench. There is a check box in each table header, to the left of the corner name. Enable or disable the check box to toggle the enabled/disabled state of the corner.

Enabling and Disabling Multiple Corners

To enable or disable multiple corners:

1. Ensure the **Corners** dialog box is open (**Tools > Corners** or **Tools > Global Corners**).

A table is displayed that contains the existing corners for the testbench.

2. Select one or more cells in the table and then right-click in the table cells of the corners you want to enable or disable.

All corners that include any of the selected cells will be enabled/disabled, as chosen in Step 3.

3. Choose **Enable Corner** to enable the corner or **Disable Corner** to disable the corner.

Disabled corners are highlighted with a gray background in the table; enabled corners have an uncolored or white background. The number of corners that are currently enabled is displayed in the upper right corner of the **Corners Setup** dialog box.

Changing Corner Values

If you want to change just one corner value, you can enter a new value directly in the table by clicking a cell and entering a new value.

To change multiple corner values at once:

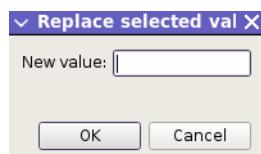
1. Ensure the **Corners** dialog box is open (**Tools > Corners** or **Tools > Global Corners**).

A table is displayed that contains the existing corners for the testbench.

2. Hold the **Ctrl** key while you click multiple values you want to change.

3. Right-click those selected values, and choose **Set Values** from the menu that opens.

The **Replace Selected Values** dialog box opens.



4. Enter a **New value** for all the selected corner values.
5. Alternatively, right-click those selected values, and choose **Fill with Series** from the menu that opens.

The **Replace Selected Values** dialog box opens.



6. Enter **Series Values** for all the selected corner values.

When you enter points of interest, separate values with commas (,). For example:

```
<start>:<step>:<stop>, poi1, poi2, poi<n>
```

The selected values must be on a single row (that is, for a single parameter).

If the number of values in the series specified in the dialog box is smaller than the number of selected values, the extra selected cells on the right are blank. If the number is larger, extra values in the series will be ignored.

7. Click **OK** to change the corner values.

The new values are updated in the corners table.

Renaming Corners

To rename a corner:

1. Ensure the **Corners** dialog box is open (**Tools > Corners** or **Tools > Global Corners**).

A table is displayed that contains the existing corners for the testbench.

2. Right-click anywhere in the column of the corner you want to rename, and choose **Rename Corner** from the menu that opens.

The **Rename Corner** dialog box opens.



3. Enter a new name and click **OK** to save your changes.

The following are examples of valid corner names:

- Cap>1p
- Cap<1p
- Cap"1p"
- corner 25

Note:

Special characters such as the following are supported (including space):

> < " + = ,

The new name is displayed in the corner values table.

Enabling or Disabling the Corner Analysis Setup

To enable all corner analyses' setups, click the check box in the **Enable** column of the **Analysis** window in the PrimeWave Design Environment main window. To disable the corner analysis setup, uncheck the check box.

Note:

When using multiple testbenches, you can disable testbenches on a per corner basis. Each testbench is added as a new row at the bottom of the **Corners** table. Click the check box to enable or disable specific testbenches for corners.

Setting up Specifications for Each Corner Condition

For information about how to set up specifications for individual corner conditions from the Specifications table, see [Setting Specification Goals](#).

Sorting Corners

To sort corners by ascending or descending values in the **Corners Setup** dialog box (**Tools > Corners** or **Tools > Global Corners**), right-click anywhere in a row or corner values and choose **Sort Ascending** or **Sort Descending** from the menu that opens.

The corners are sorted by ascending or descending value from left to right in the table.

Defining Corner Groups

A corner group is a single corner object that represents a set of subcorners. A single corner becomes a corner group implicitly by specifying multiple values for a parameter. The values can be specified using both ranges and individual Points of Interest (POI). The ranges can be specified using the same syntax as in the **Generate Corners** dialog box.

To define a corner group:

1. Choose **Tools > Corners** or **Tools > Global Corners** from the PrimeWave Design Environment main menu bar.

The **Corners** dialog box opens. First, create a single corner or edit an existing corner to create a corner group object. The range of values/POI must be entered explicitly by the user.

Note:

In the **Corners** dialog box, the number of subcorners in a group is displayed in the table header. The total number of corners (at the top right of the

Corners dialog box) indicates the actual number of runs and not the total number of corner objects.

2. Edit the values for the corner parameters by clicking in a table cell and directly typing in comma-separated values or clicking the **Edit Corner Value** button  to open the **Specify Range** dialog box.

Specifying multiple values for a parameter creates a corner group that represents a set of subcorners. You can repeat step 1 to step 2 to create multiple corner groups by specifying a varied range of values for the parameters.

3. Click **OK** or **Apply** in the **Corners** dialog box to add the corner analysis to the simulation.

To view the results, choose **Results > Results Viewer** from the PrimeWave Design Environment main menu bar. See [Using the ResultsView](#).

Note:

If you are importing testbenches that contain corners, the imported corner groups can be merged with the existing corner groups only if they have exactly the same values specified in exactly the same way.

The **Remove Duplicate** operation removes duplicate entries if two corner groups have exactly the same values specified in exactly the same way. On the other hand, the **Remove Duplicate** operation does not remove individual corners that are contained within a group, or groups that partially overlap.

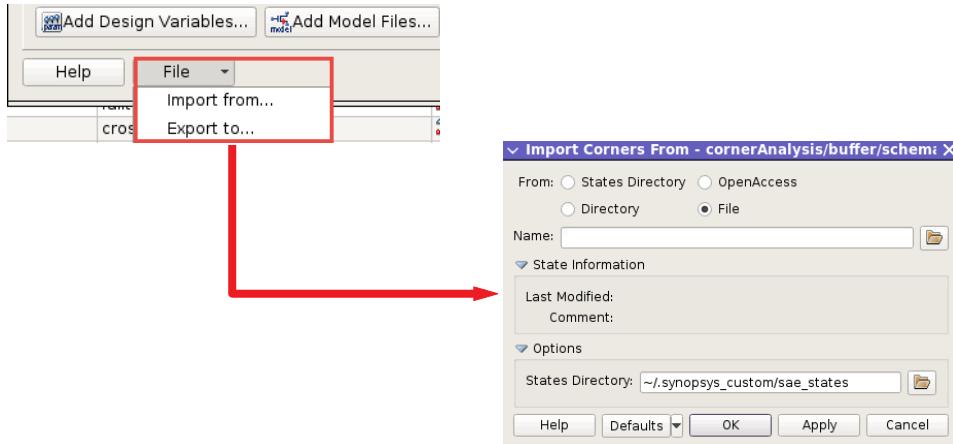
All menu operations that involve corner objects operate on the entire group. Enable/disable check boxes are applied to all subcorners; you cannot individually enable/disable a single subcorner within a corner group.

Importing and Exporting Corner Information to a File or State

You can import corner information from a file or state and export corner information to a file or a state. The corner data is imported/exported to an XML file employing the same format as is used for the `cornersSetup.xml` file within PrimeWave Design Environment states.

To import corner information from a file or state:

1. In the **Corners** setup dialog box, select **File > Import from**.



2. Select From where the corner information is to be imported, **States Directory**, **OpenAccess**, **Directory**, or **File** (the default).

Note:

The **States Directory**, **OpenAccess**, and **Directory** source types have the same meaning as in the **Load State** dialog box.

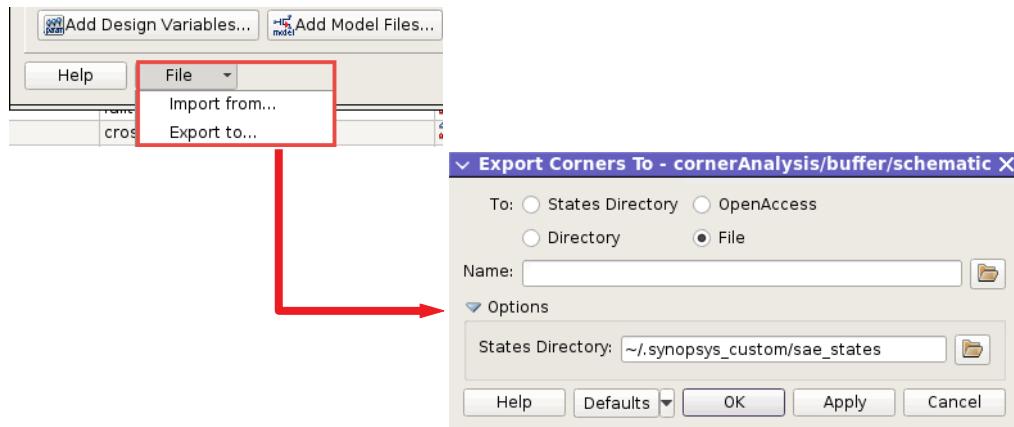
3. For **File**, browse to select the source file.
4. Click **Apply** or **OK** to import the corner information.

The corners are appended to the **Corners** setup dialog box.

5. Examine the imported corners and choose to cancel if you decide you do not want them.
6. Click **Apply** or **OK** in the **Corners** setup dialog box to save the corners in the PrimeWave Design Environment session or testbench.

To export corner information to a file or state:

1. In the **Corners** setup dialog box, select **File > Export to**.



2. Select where the corner information is to be exported To: **States Directory**, **OpenAccess**, **Directory**, or **File** (the default).

Note:

The **States Directory**, **OpenAccess**, and **Directory** destination types have the same meaning as in the **Save State** dialog box.

A warning is issued if an existing corners setup will be overwritten as a result of the export operation.

3. Click **Apply** or **OK** to export the corner information.

The corners are exported to your chosen destination.

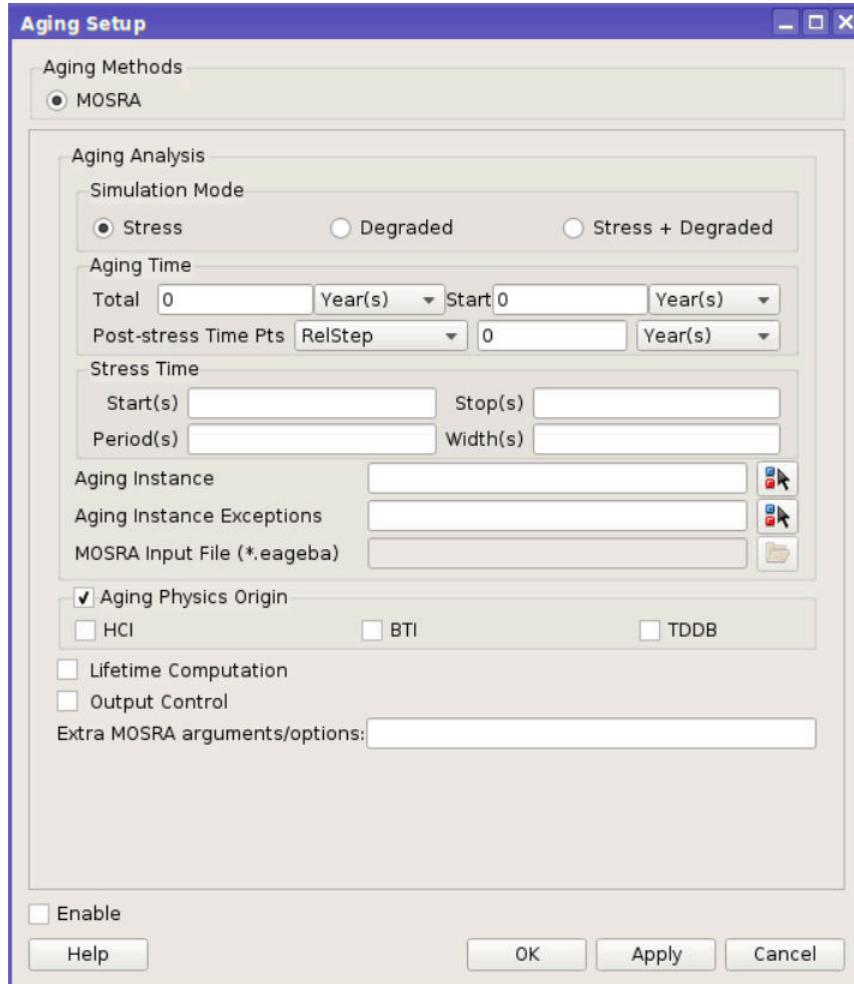
You can make edits to the corners in the **Corners** setup dialog box and export them multiple times over without affecting the PrimeWave Design Environment session.

Setting Up Aging Analyses

A detailed overview of MOSRA analysis, as well as a flow example, are provided in [MOSRA Integration with the PrimeWave Design Environment](#).

To set up aging analyses (such as MOSRA):

1. Choose **Tools > Aging Setup** to open the **Aging Setup** dialog box.



2. Set the necessary options in the **Aging Setup** dialog box.

The following options are available in the **Aging Setup** dialog box for the **MOSRA** aging method.

Option	Description
Simulation Mode	Choose from the following options: <ul style="list-style-type: none"> • Stress (see Stress Simulation) • Degraded (see Degraded Simulation) • Stress + Degraded (see Combined Stress + Degraded Simulation)

Option	Description
Aging Time	Reliability test time to use in post-stress simulation phase.
Stress Time	Optional time limits for stress effect calculation during transient simulation.
Aging Instance	Selects MOSFET devices to which the simulator applies HCI and/or BTI analysis. The default is all MOSFET devices with reliability model appended.
Aging Instance Exceptions	Deselects instances for MOSRA analysis.
MOSRA Input File (*.eageba)	MOSRA degradation information file.
Aging Physics Origin	When selected, choose from HCI , BTI , or TDDB . These are the control flags to select the HCI model, the BTI model, or the TDDB model defined in the API.
Lifetime Computation	When selected, specify Lifetime Item , Lifetime Criteria , Lifetime Criteria for NMOS , and Lifetime Criteria for PMOS .
Output Control	When selected, specify Age Threshold , Sorting Item , and MOSRA Output Format (YES, NO, or CSV) .
Extra MOSRA arguments/options	Add MOSRA arguments and options missing from the GUI in this field.

3. Ensure **Enable** is checked so you can include the MOSRA analysis as part of your testbench.
4. Click **OK** or **Apply**.

Your MOSRA analysis is now set up.

Setting Up Monte Carlo Analyses

Monte Carlo analyses are used to model the effects of process variations on circuit performance. The PrimeWave Design Environment supports an interface to the native Monte Carlo capability that your simulator supports.

You can use design variables in your Monte Carlo analysis to specify a range or an individual value, which can be used to set up the multiple netlist runs. The tool resolves design variables to numerical values. If the value is valid, a series of netlists are generated

in the range of sweeps specified for simulation. Before the simulation is run, a validity check is performed to ensure the design variable setup is valid.

PrimeSim noise summary reports are available when Monte Carlo analysis is run as part of SN Noise and HB Noise analyses. See [Printing the PrimeSim HSPICE Noise Summary](#).

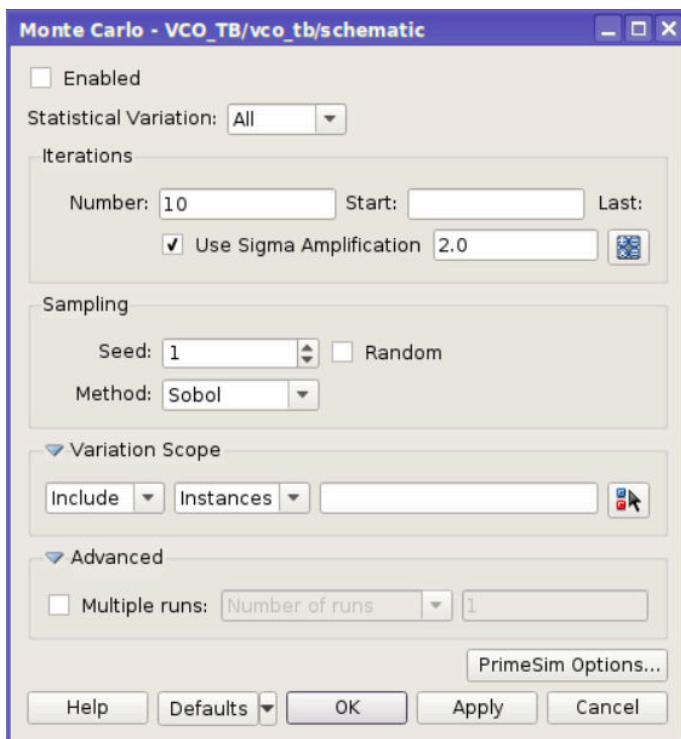
Note:

The PrimeWave Design Environment does not include an interface for defining the statistical variations of your device parameters. This information is assumed to be included in model or include files. The presence of statistical data is not verified before running a Monte Carlo analysis.

To set up a Monte Carlo analysis:

1. Choose **Tools > Monte Carlo** from the PrimeWave Design Environment main menu bar.

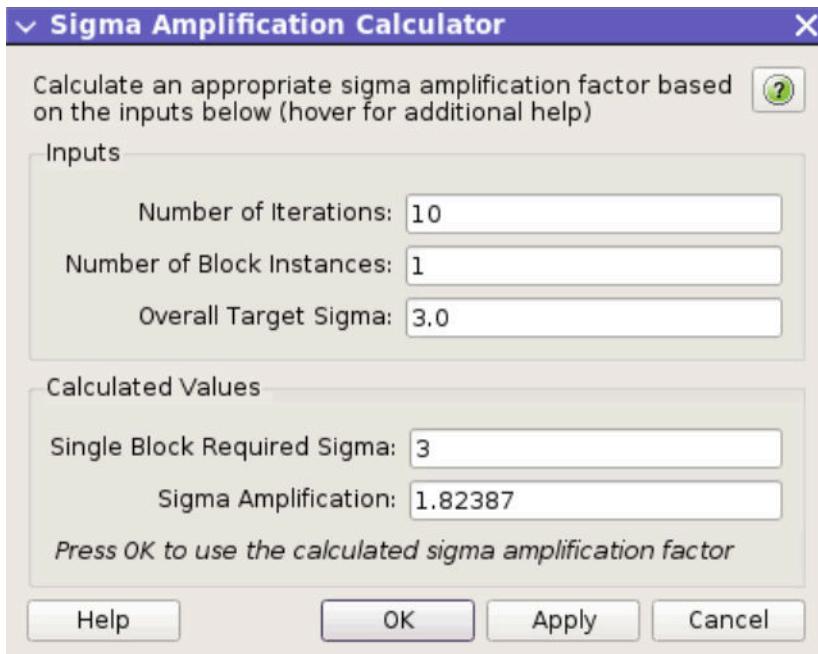
The **Monte Carlo** dialog box opens.



2. Check **Enabled** to include this analysis as part of your testbench.
3. Enter the **Number** of Monte Carlo iterations you want to run.

By default, the analysis **Start** iteration is 1. Specify an alternate value if you want to start with some other value for iteration.

4. (Optional) Enable **Use Sigma Amplification** to use sigma amplification.
5. (Optional) If **Use Sigma Amplification** is enabled, you can directly enter an amplification value, or click  to open the **Sigma Amplification Calculator** dialog box.



Note:

The first time you open the calculator, the **About Sigma Amplification** dialog box appears. Click **Close** to proceed to the **Sigma Amplification Calculator**. Click the **Information** button  to reopen the **About Sigma Amplification** dialog box.

Here you can use the number of iterations, block instances, and target sigma to auto-calculate the required sigma per block and the sigma amplification factor for process parameter samples. Click **OK** to close the **Sigma Amplification Calculator**.

6. (Optional) In the **Sampling** options section of the **Monte Carlo** dialog box, click the **Random** check box to remove the seed selector from the GUI. Alternatively, adjust the **Seed** value (default is 1). If you choose **Random**, the data will not be repeatable.

Note:

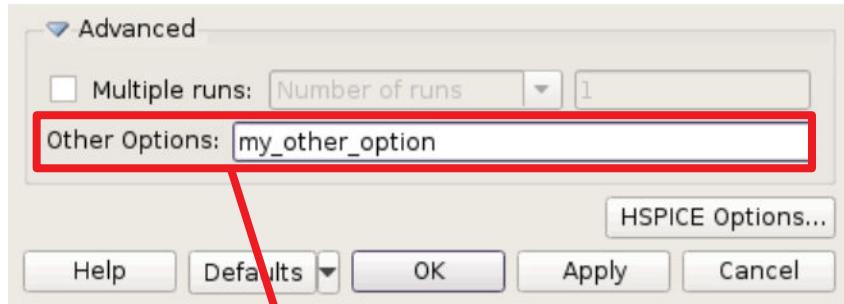
The **Random** option is not available with PrimeSim XA.

7. (Optional) Select a sampling **Method**. Options include **SRS**, **LHS**, **Factorial**, **OFAT**, **Sobol**, and **Neiderreiter**. If you choose **LHS**, select an LHS **Sample Size** and your desired number of **Replicates**.

Note:

The **Method** options supported with PrimeSim XA are **SRS** and **LHS**.

8. (Optional) Click the down arrow to expand the **Variation Scope** options section. These options allow you to **Include** or **Exclude** specific **Instances** or **Cells** where variation is applied (or not applied).
9. (Optional) Enter the instance or cell names directly into the text box, or click the **Select in Design** button  and select them in the Schematic Editor.
10. If you want to divide iterations among the generated netlists for concurrent or distributed processing, continue to [Step 11](#). By default, all iterations are placed in the same PrimeSim HSPICE netlist.
Otherwise, skip to step [Step 13](#).
11. (Optional) Click the down arrow to expand the **Advanced** options section.
12. (Optional) Click the **Multiple runs** check box to break the Monte Carlo analysis into multiple netlists.
Choose the method of separation (**Number of runs** or **Iterations per run**), and enter a value for the number of netlists or iterations per netlist.
13. (Optional) Type any necessary **Other Options** in the text field. The text you type is appended to the Monte Carlo definition line in the netlist, as shown in the figure below.



```

xn2 net08 avss avss nfet l=8n nfin=2 nf=1.0 cpp=54n p_la=0 ngcon=l fpitch=0.0
+ plorient=0 asej='((39n*5.0n)*2)' adej='((39n*5.0n)*2)' psej='((1*2)*((2*39n)+5.
+ pdej='((1*2)*((2*39n)+5.0n))' xpos=-.99 ypos=-.99 pre layout local=-1 l_shape=0
+ l_shape_s=0 lle_sa_sdb=0 lle_sa_ddb=0 lle_sb_sdb=0 lle_sb_ddb=0 nfin_a_sdb=0
+ nfin_a_ddb=0 nfin_b_sdb=0 nfin_b_dbb=0 nfin_a_sdb2=0 nfin_b_sdb2=0 lle_mgbn=0
+ lle_mgbs=0 lle_rxrxa_sdb=0 lle_rxrxa_ddb=0 lle_rxrxb_sdb=0 lle_rxrxb_ddb=0
+ lle_rxrxn=0 lle_rxrxs=0 lle_pcrxs=0.2405u lle_pcrxw=0.2405u lle_nwa=0
+ lle_nwb=0 lle_nwn=0 lle_nws=0 lle_ctne=0 lle_ctnw=0 lle_ctse=0 lle_ctsw=0
+ lle_ctzne=0 lle_ctznw=0 lle_ct2se=0 lle_ct2sw=0 lle_scne=0 lle_scfnw=0
+ lle_scse=0 lle_scsw=0 lle_vctse=0 lle_vctnw=0 lle_vctse=0
+ lle_vsctse=0 lle_rcnw=0 lle_rcsw=0 lle_rcne=0 lle_rcse=0 lle_rcctnw=0
+ lle_rccts=0 lle_rcctne=0 lle_rcctse=0 lle_rcrca=0 lle_rcrcb=0u lle_pcra=0
+ lle_pcrcb=0u lle_sa_ddb_min=50n lle_sb_ddb_min=50n nfin_a_ddb_min=2
+ nfin_b_ddb_min=2 lle_sa_ali1=0u lle_sb_ali1=0u sasb_ali1=0 mfin=1 m=1 par=1
+ pdevdops=1 pdevle=1 pdevwg=1 ptwell=0 plnest=1 pldist=1 p_vta=0 dtemp=0
+ psw_acv_sign=1 lrstd=19.5n

```

```

.ac DEC 100 1 10G sweep monte=1500 firstrun=1 my_other_option
.lstb mode=single vsource=v2
.probe ac lstb(db) lstb(m) lstb(p) lstb(r) lstb(i)
.measure lstb gain_margin gain_margin
.measure lstb phase_margin phase_margin
.measure lstb unity_gain freq unity_gain_freq
.measure lstb loop_gain_minifreq loop_gain_minifreq
.option opfile=1 split_dp=1
.option probe=1
.probe ac v(*) level=1

```

14. (Optional) Click the **PrimeSim Options** button to select any PrimeSim HSPICE Monte Carlo options you want to include in the analysis.

For more information on choosing PrimeSim options, see [Setting Simulator Options](#).

15. Click **OK** to save the analysis setup.

See [Analyzing Statistical Data](#) for information on creating histograms and scatter plots.

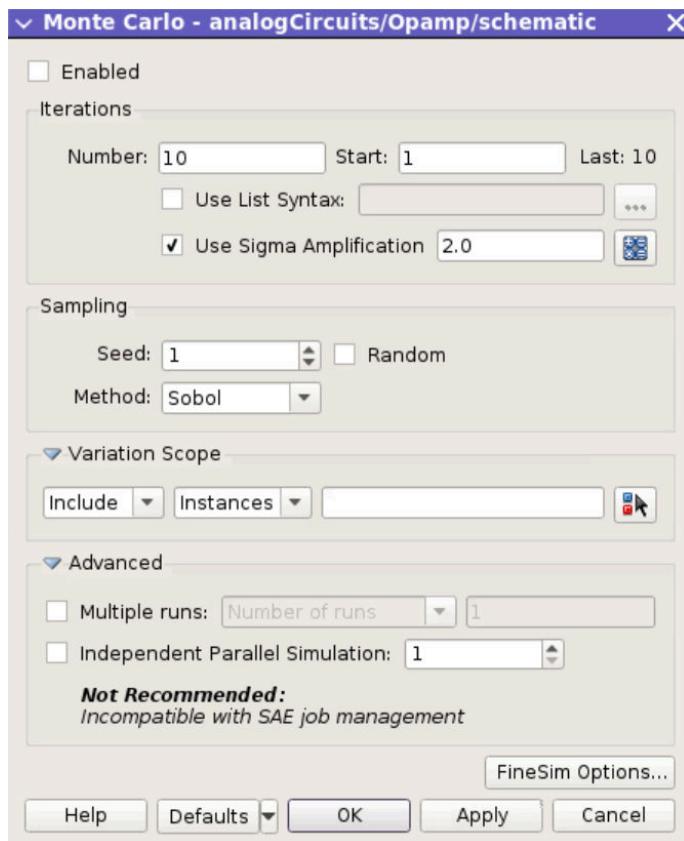
Setting Up Monte Carlo Analyses for FineSim

Monte Carlo analyses are used to model the effects of process variations on circuit performance. The PrimeWave Design Environment supports an interface to the native Monte Carlo capability that FineSim simulator supports.

To set up a Monte Carlo analysis for FineSim:

1. Choose **Tools > Monte Carlo Setup** from the PrimeWave Design Environment main menu bar.

The **Monte Carlo** dialog box opens.



2. Click **Enabled** to include this analysis as part of your testbench.
3. Enter the **Number** of Monte Carlo iterations you want to run (the default is 10).
4. By default, the value for **Start** is 1, as the analysis starts with iteration 1. Specify an alternate value if you want to start with some other value.
5. (Optional) Enable **Use List Syntax** to specify an iteration range.

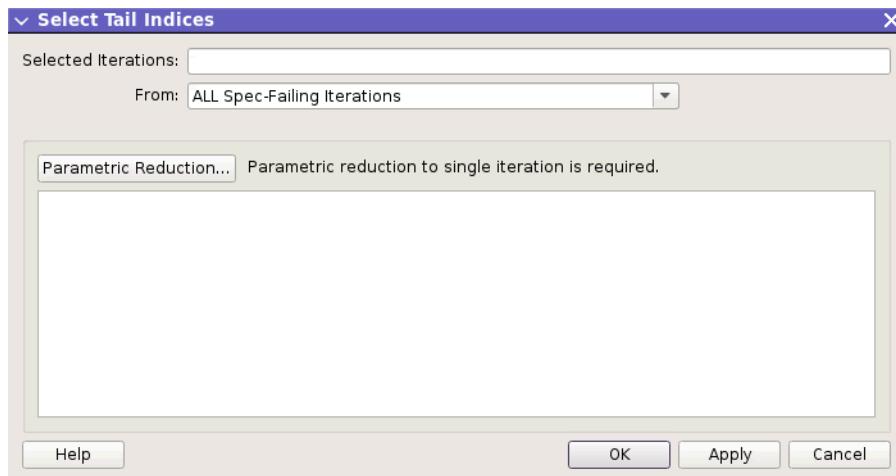
This feature supports a list syntax for specifying iterations. You can run individual iterations or subsets of an iteration range.

For example, you can specify a list of iterations:

5 : 7 10

This example includes iterations 5 through 7 and iteration 10.

6. (Optional) With **Use List Syntax** enabled, click  to open the **Select Tail Indices** dialog box.



Here you can choose subsets of Monte Carlo iterations to re-run when saving more waveforms. By default, all of the Monte Carlo iterations for which any of the specified measurements failed are prepopulated into the **Selected Iterations** field. The bottom table in the dialog box shows the breakdown of failed iterations on a per-measurement basis.

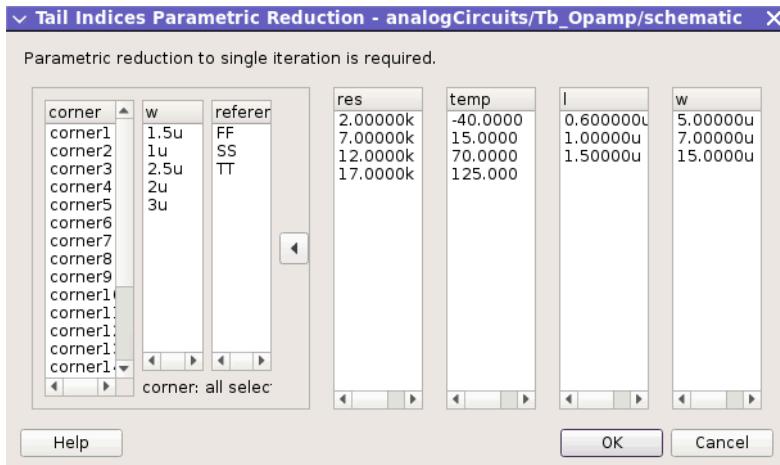
7. (Optional) Select iterations **From** the following options:

Select Iterations From ...	Description
All Spec Failing Iterations	Default. Includes the full superset of failing iterations.
Spec-Failing Iterations, by Measurement	Allows you to select failing iterations on a per-measurement basis, using checkboxes in the table to select iterations for specific measurements only.
ALL Measurement Distribution Tails	Includes all the iterations that correspond to a subset of the measurements that were furthest away from their mean values; that is, those in the tail regions of the corresponding measurement distributions.
Measurement Distribution Tails, by Measurement	Allows you to select measurements that were furthest away from their mean values (tails) on a per-measurement basis, using checkboxes in the table to select iterations for specific measurements only.

Select Iterations From ...	Description
ALL Measurement Extreme Value Iterations	Includes all extreme iterations; that is, only those iterations that correspond to the minimum and maximum values of circuit measurements. (This is essentially equivalent to a tail selection where the number of left and right tails is set to 1, for a total of 2 tails.)
Measurement Extreme Value Iterations, by Measurement	Allows you to select extreme iterations on a per-measurement basis, using checkboxes in the table to select iterations for specific measurements only.
From Iterations Selected in Results Viewer	Prompts you to Select in Results Viewer iterations (rows) to re-run. Multiple selections can be made, and the list can be cleared or added to at any point.

You can choose specification-failing iterations, iterations in measurement *tails*, or iterations for measurement extremes (min/max values), using an all or per-measurement level of granularity.

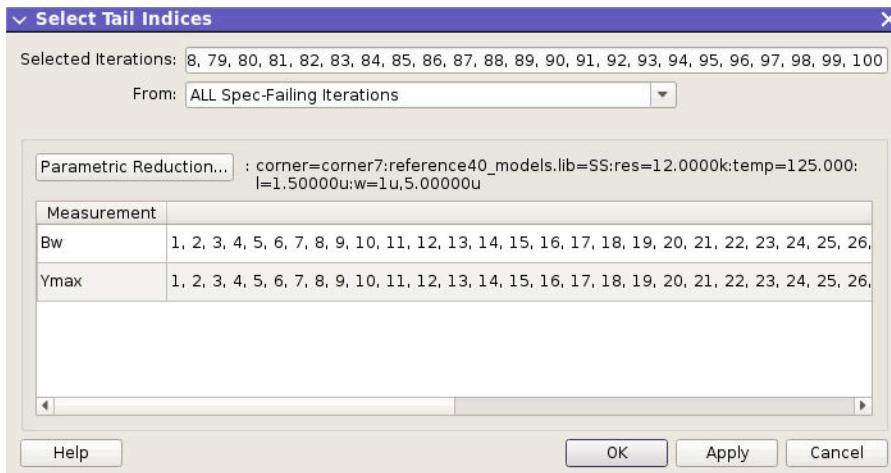
8. (Optional) In the **Select Tail Indices** dialog box, click **Parametric Reduction** to open the **Tail Indices Parametric Reduction** dialog box.



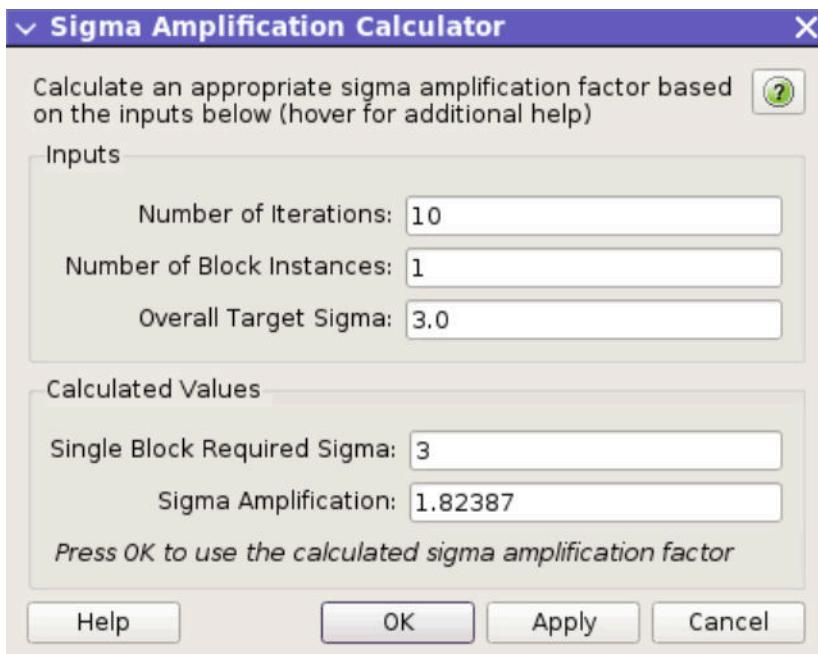
Note:

You can use the arrow in the middle of the dialog box to show and hide various parameters.

Specify a particular corner/sweep iteration for debugging and click **OK** to accept the changes and close the **Tail Indices Parametric Reduction** dialog box. The relevant measurements and spec-failing iterations are added to the **Select Tail Indices** dialog box.



9. Click **OK** to apply the **Select Tail Indices** changes and close the dialog box. At this point you might choose to run this set of iterations only, fine-tuning the analysis until the results are more favorable.
10. (Optional) Enable **Use Sigma Amplification** to use sigma amplification.
11. (Optional) If **Use Sigma Amplification** is enabled, you can directly enter an amplification value, or click to open the **Sigma Amplification Calculator** dialog box.



Note:

The first time you open the calculator, the **About Sigma Amplification** dialog box appears. Click **Close** to proceed to the **Sigma Amplification Calculator**. Click the **Information** button  to reopen the **About Sigma Amplification** dialog box.

Here you can use the number of iterations, block instances, and target sigma to auto-calculate the required sigma per block and the sigma amplification factor for process parameter samples. Click **OK** to close the **Sigma Amplification Calculator**.

12. (Optional) In the **Sampling** options section, click the **Random** check box to remove the seed selector from the GUI and set the `finesim_mcseed` option to -1. Alternatively, adjust the **Seed** value (default is 1). If you choose **Random**, the data will not be repeatable.
13. (Optional) Select a sampling **Method**. Options include **SRS**, **LHS**, **Factorial**, **OFAT**, **Sobol**, and **Neiderreiter**. If you choose **LHS**, select an **LHS Sample Size** and your desired number of **Replicates**.
14. (Optional) Click the down arrow to expand the **Variation Scope** options section. These options allow you to **Include** or **Exclude** specific **Instances** or **Cells** where variation is applied (or not applied).
15. (Optional) Enter the instance or cell names directly into the text box, or click the **Select in Design** button  and select them in the Schematic Editor.
16. (Optional) Click the down arrow to expand the **Advanced** options section.
17. (Optional) Click the **Multiple runs** check box to break the Monte Carlo analysis into multiple netlists.

Choose the method of separation (**Number of runs** or **Iterations per run**), and enter a value for the number of netlists or iterations per netlist.

18. (Optional) Enable **Independent Parallel Simulation** to include independent parallel simulation.

Caution:

The FineSim **Independent Parallel Simulation** is incompatible with the PrimeWave Design Environment job management and not recommended.

If the **Independent Parallel Simulation** is enabled, and the **Enable Parallel Simulation** group is enabled in the **Environment Options** dialog box, FineSim distributes all sweep simulations to different CPUs.

19. (Optional) Click the **FineSim Options** button to open the **FineSim Global Simulation Options** dialog box. Select any FineSim Monte Carlo option you want to include in the analysis.

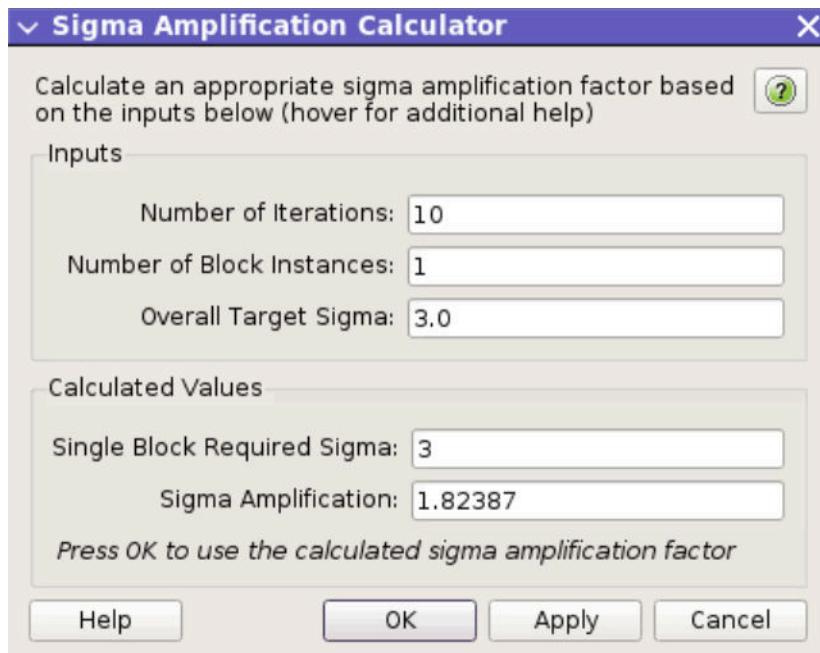
For more information on choosing FineSim options, see [Setting Simulator Options](#).

20. Click **OK** to save the analysis setup.

Using Sigma Amplification in Monte Carlo Analysis

To use sigma amplification, in the **Monte Carlo** dialog box:

1. Enable **Use Sigma Amplification**.
2. If **Use Sigma Amplification** is enabled, you can directly enter an amplification value, or click  to open the **Sigma Amplification Calculator** dialog box.



Note:

The first time you open the calculator, the **About Sigma Amplification** dialog box appears. Click **Close** to proceed to the **Sigma Amplification Calculator**. Click the **Information** button  to reopen the **About Sigma Amplification** dialog box.

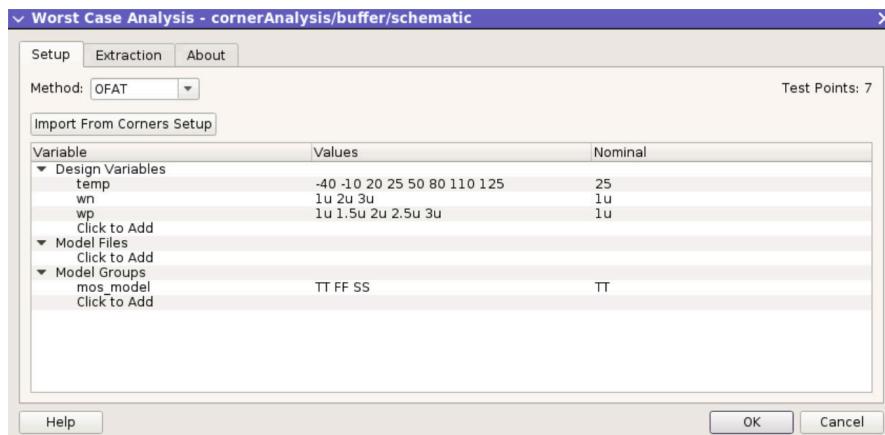
Here you can use the number of iterations, block instances, and target sigma to auto-calculate the required sigma per block and the sigma amplification factor for process parameter samples. Click **OK** to close the **Sigma Amplification Calculator** and return to the **Monte Carlo** dialog box.

Setting Up Worst-Case Analysis

Worst-case analysis is used to analyze corner setup and extract worst-case corners for each measurement that has a spec. Using worst-case analysis early in the design cycle provides a smaller set of corners to run initially for faster turnaround. Worst-case analysis is a conservative approach, as it typically considers only the maximum or minimum values of parameter variation in order to identify the parameter settings for each measurement spec that should result in the worst-case condition. Once the design stabilizes, you can run the full corners setup.

To set up worst-case analysis:

1. Choose **Tools > Worst Case Analysis** from the PrimeWave Design Environment main menu bar. The **Worst Case Analysis** dialog box appears.



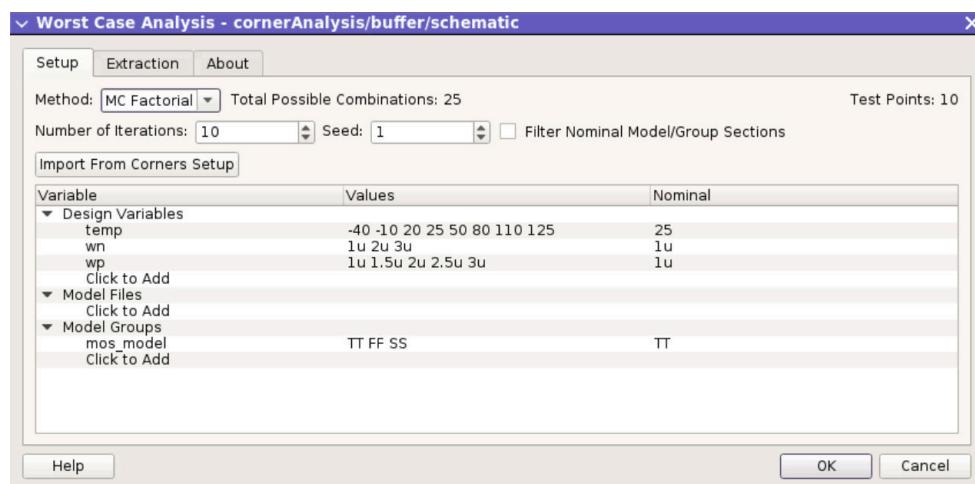
2. Click **Import From Corners Setup** to import parameters from the corner setup, or **Click to Add** variables, values, and nominal values to the table.

When adding model files manually, the full path must be specified if a model file path is not currently configured for the session.

3. Choose a simulation **Method**. The method you choose determines the number of **Test Points**, indicated in the upper right corner of the dialog box.

There are two methods of worst-case analysis:

- **OFAT** (One Factor At a Time) runs simulations for the nominal and the non-default extremes of each parameter being varied. For Model Sections, all but the typical parameter values are used. For other parameters, only the min/max values are used.
- **MC Factorial** (Monte Carlo Factorial) considers all combinations of the extremes and enables interdependency between parameters versus the OFAT, which only looks at one parameter at a time. Because the results of a factorial analysis could be larger than the total number of corners running, you must find a way to limit the number of runs.



The MC Factorial method allows you to specify the **Number of Iterations** to run and the **Seed** number to determine the worst-case corners.

You can choose to **Filter Nominal Model/Group Sections**, which ignores nominal section names from the Values column when generating test points (except for the nominal test point).

The Monte Carlo analysis is random only if the number of iterations requested is less than the factorial combinations. If the **Number of Iterations** you specify is greater than the maximum indicated by **Total Possible Combinations**, the maximum is used. The tool randomly selects the subset of iterations.

4. Click **OK** to close the **Worst Case Analysis** dialog box and add the worst-case analysis to the Analysis pane in the main PrimeWave Design Environment window.

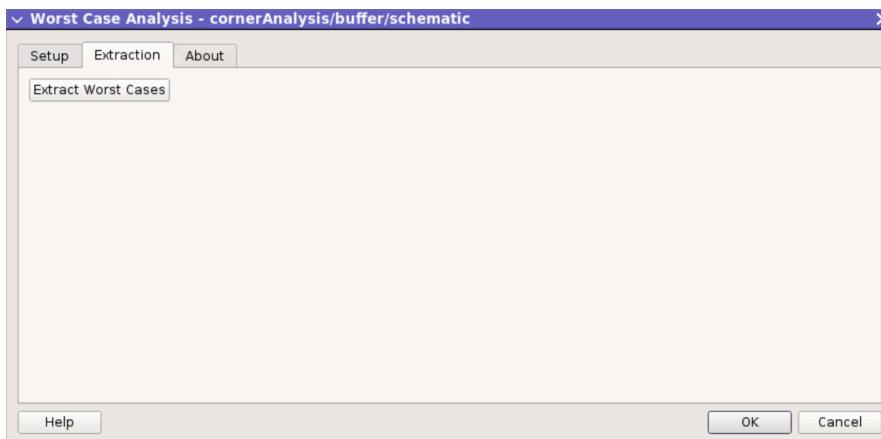
Any analyses that are not relevant to the worst-case analysis (such as corners analysis) are grayed out when the worst-case analysis is added.

Analysis	Type	En	Value
Worst Case Analysis			Test Points: 7
corners	<input checked="" type="checkbox"/>		Total:53, enabled: 53
▶ tran	tran	<input checked="" type="checkbox"/>	Start Time: 0 Time Step: 1n Stop Time: 20n
▶ op	op	<input checked="" type="checkbox"/>	Format: All

The worst-case analysis must be run to provide the simulation data required to extract worst-case corners. The worst-case corners are extracted based on the sensitivity results of the worst-case analysis along with measurements having specifications resulting in one worst-case corner per measurement specification.

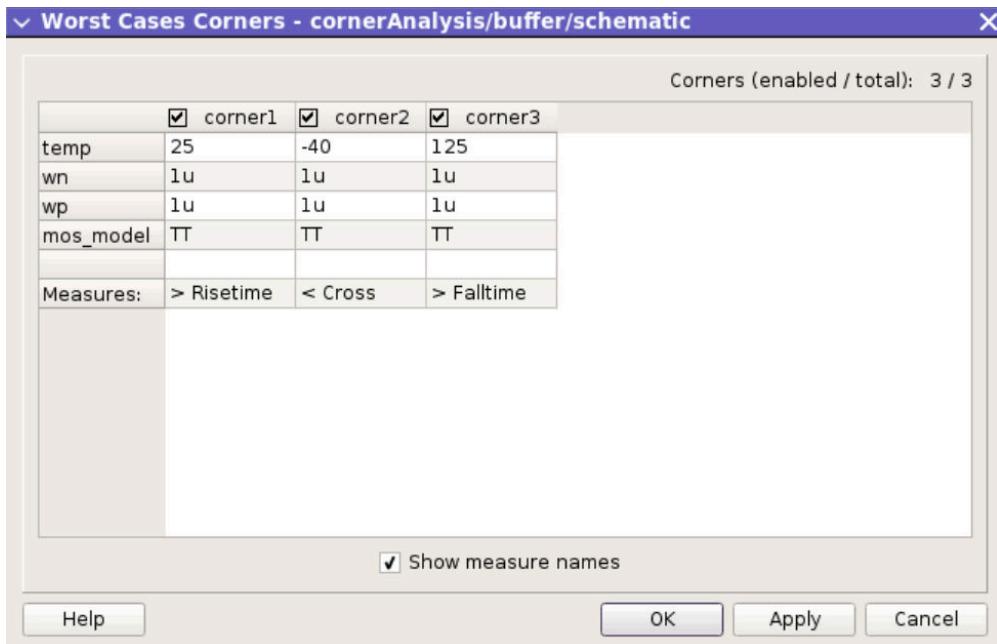
To extract worst-case corners:

1. Choose **Simulation > Netlist and Run** to run the worst-case analysis.
2. Reopen the **Worst Case Analysis** dialog box from the Analysis pane by double-clicking in the Value cell. Switch to the **Extraction** tab in the **Worst Case Analysis** dialog box (if there are simulation results, this tab is automatically on top).



3. Click **Extract Worst Cases** to extract the worst-case corners. The Worst Cases are added to the Analysis pane in the main PrimeWave Design Environment window and the **Worst Cases Corners** dialog box is opened. The enablements of these analyses in the Analysis pane change automatically when you click **Extract Worst Cases**.

Analysis	Type	En	Value
Worst Case Analysis			Test Points: 7
corners	<input checked="" type="checkbox"/>		Total:53, enabled: 53
Worst Cases	<input checked="" type="checkbox"/>		Total:3, enabled: 3
▶ tran	tran	<input checked="" type="checkbox"/>	Start Time: 0 Time Step: 1n Stop Time: 20n
▶ op	op	<input checked="" type="checkbox"/>	Format: All



The worst-case corners are shown in the table with the measurements that are applied to each corner.

The greater-than sign (>) indicates the measurement value is either greater than or closest to the maximum named measurement specification. The less-than sign (<) indicates the measurement value is either less than or closest to the minimum named measurement specification.

Range specifications have both minimum and maximum specifications.

Disable **Show measure names** to hide the measurement names.

You can disable any worst-case corners you do not wish to include in the analysis.

4. Click **OK** to close the **Worst Cases Corners** dialog box and apply the changes. You can invoke the **Worst Cases Corners** dialog box any time from the Analysis pane by double-clicking in the Value cell of the Worst Cases row.
5. Click **OK** to close the **Worst Case Analysis** dialog box.
6. In the Analysis pane of the main PrimeWave Design Environment window, make sure Worst Case Analysis and Corners are disabled and Worst Cases is enabled. (The enablements of these analyses change automatically when you click **Extract Worst Cases**, as in Step 3 above.)
7. Choose **Simulation > Netlist and Run** to run the worst-case corner analysis.

8. View the results in the ResultsView.
9. Repeat the worst-case corner analysis as necessary, changing any variables to experiment with the results. When you are satisfied with the design, you can delete Worst Case Analysis and Worst Cases in the Analysis pane and run the full Corners analysis.

Setting Up Chained Testbenches

Testbench chaining allows you to use measurement values from one testbench as input parameter values for another testbench. You select measurement values from the common database to populate input parameter values (such as design, sweep, or corner variables) for the current testbench.

This topic describes the following:

- [Opening the Chain Testbenches Tool](#)
- [Specifying a Source Testbench Using the Chain Testbenches Tool](#)
- [Specifying a Destination Testbench Using the Chain Testbenches Tool](#)
- [Specifying Destination Variables Using the Chain Testbenches Tool](#)

Opening the Chain Testbenches Tool

To chain testbenches using the Chain Testbenches tool:

1. Choose **Tools > Chaining** from the PrimeWave Design Environment main menu bar.
The **Chain Testbenches** dialog box appears.



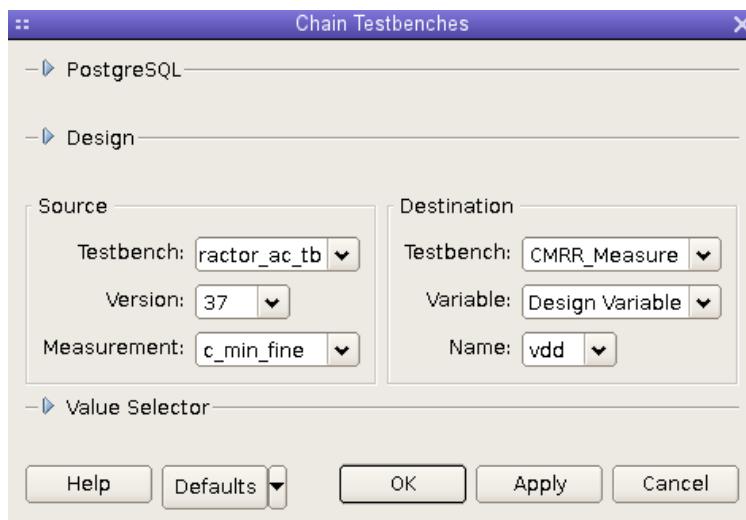
2. Expand the **PostgreSQL** and **Design** sections of the **Chain Testbenches** dialog box and notice that these options are prepopulated with your database and design information.

Specifying a Source Testbench Using the Chain Testbenches Tool

The PrimeWave Design Environment queries the database for a list of source testbenches and test suites, as well as the versions and measurements available for each.

To specify source information for testbench chaining:

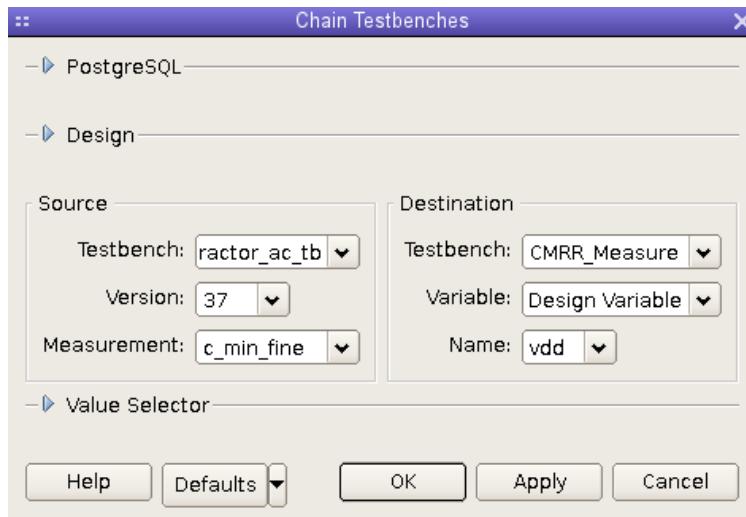
1. Begin by [Opening the Chain Testbenches Tool](#).
2. In the **Source** section of the **Chain Testbenches** dialog box, select the **Testbench**, **Version**, and **Measurement** values.



Specifying a Destination Testbench Using the Chain Testbenches Tool

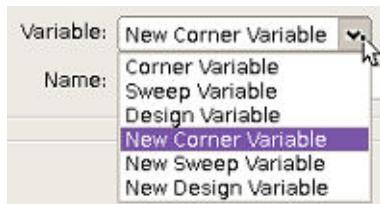
You can select a global scope, multiple testbench (MTB), or single testbench (STB) as the destination. To specify the destination testbench to chain:

1. Begin by [Specifying a Source Testbench Using the Chain Testbenches Tool](#).
2. In the **Destination** section of the **Chain Testbenches** dialog box, select the **Testbench**.



Specifying Destination Variables Using the Chain Testbenches Tool

Values from the source measurement are used to set the available values of the destination **Variable**.



Depending on the selected source and destination testbenches or test suites, you can select from the following types of variables to chain:

- Corner
- Design
- Sweep
- New Corner, New Sweep, or New Design

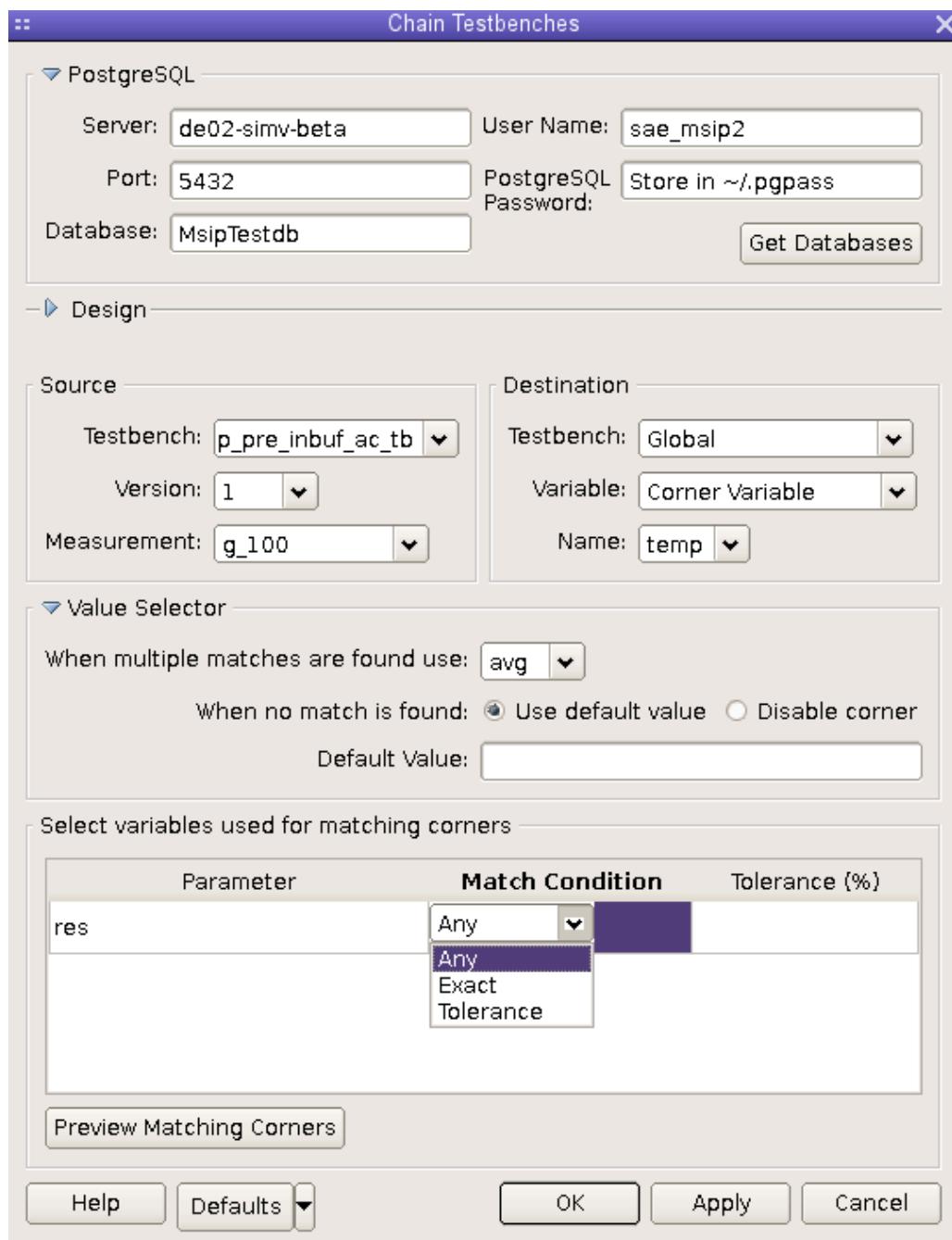
This topic describes the following:

- [Chaining Corner Variables](#)
- [Chaining Design Variables](#)
- [Chaining Sweep Variables](#)

Chaining Corner Variables

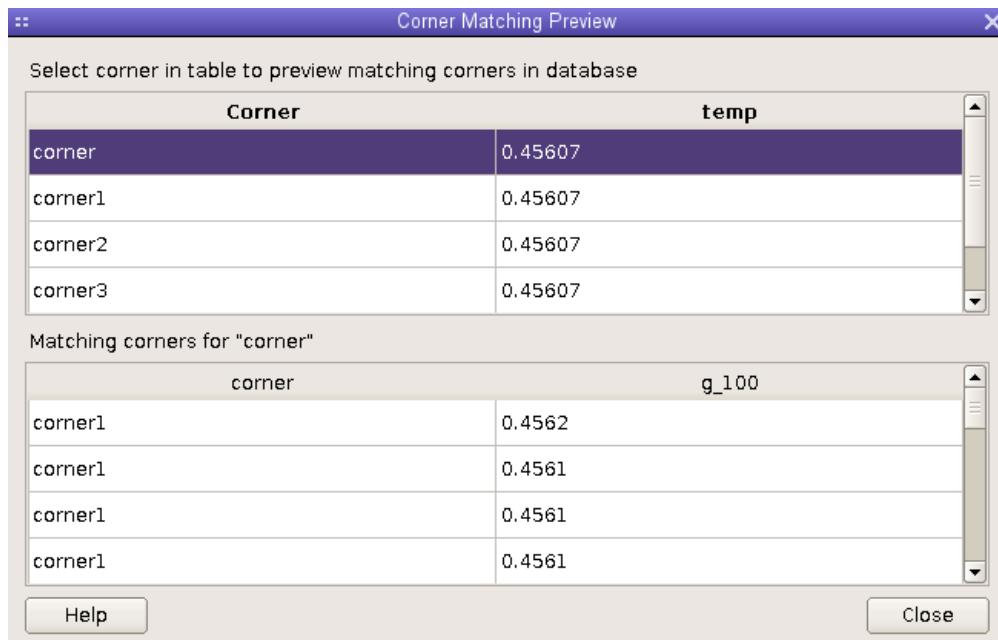
To chain corner variables:

1. Choose **Corner Variable** in the destination **Variable** field.
2. Select the **Name** of the destination corner variable.
3. Alternatively, choose **New Corner Variable** in the destination **Variable** field and designate a **Name** for that new variable. A new corner variable is added to corner setup of that testbench.
4. Expand the **Value Selector** section.



5. When many source corners match one destination corner, you can choose **avg**, **min**, **max**, **first**, or **last** from the **When multiple matches are found use** drop-down menu.

6. When no match is found for a corner, you can choose to **Use default value** or **Disable corner**. If you choose to use a default value, specify it in the **Default Value** field.
7. In the **Select variables used for matching corners** section, choose a **Match Condition** for each **Parameter**. Possible match conditions are: **Any**, **Exact**, and **Tolerance**. If you choose **Tolerance**, specify a **Tolerance (%)**.
8. Choose **Preview Matching Corners** to open the **Corner Matching Preview** dialog box. Observe the matching corners and close the dialog box.

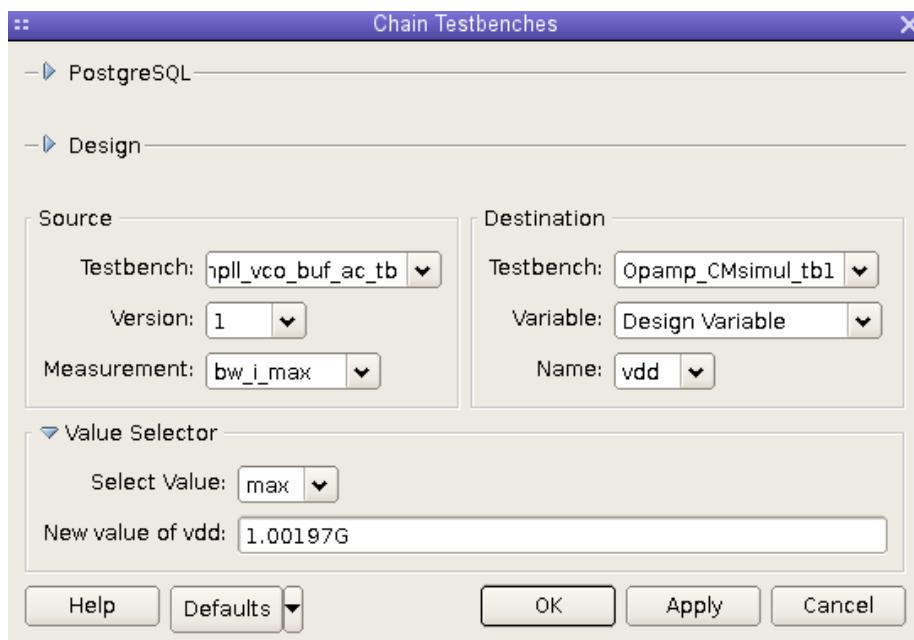


9. Click **Apply** and notice the updates to the analysis in the main PrimeWave Design Environment window.
10. Click **OK** to chain the testbenches using your selections. The **Chain Testbenches** dialog box closes.
11. In the main PrimeWave Design Environment window, choose **Simulation > Netlist and Run**. Review the results.

Chaining Design Variables

To chain design variables:

1. Choose **Design Variable** in the destination **Variable** field.
2. Alternatively, choose **New Design Variable** in the destination **Variable** field and designate a **Name** for that new variable. A new design variable will be added in the destination testbench's Design Variables table.
3. Expand the **Value Selector** section.



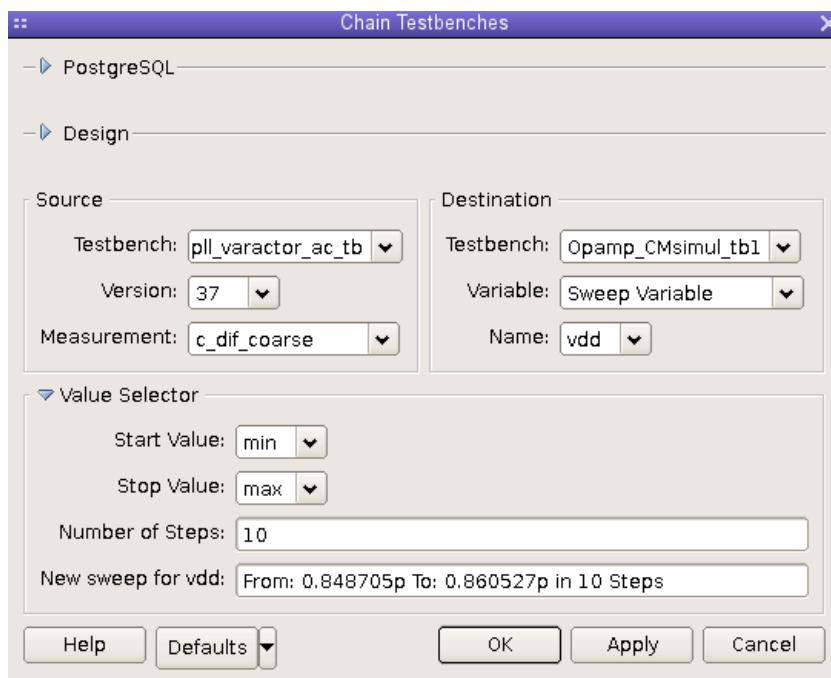
4. Depending on the type of simulation that was run for the source testbench, the database returns either a single value or a list of values for the **Select Value** option. If the query returns a single value, that value is used as the **New value of** the design variable.
If the query returns multiple values, choose **avg**, **min**, **max**, **first**, or **last** from the **Select Value** drop-down menu to set the **New value of** the design variable. (The **New value of** box is not editable.)
5. Click **Apply** and notice the updates to the variable in the main PrimeWave Design Environment window.

6. Click **OK** to chain the testbenches using your selections. The **Chain Testbenches** dialog box closes.
7. In the main PrimeWave Design Environment window, choose **Simulation > Netlist and Run**. Review the results.

Chaining Sweep Variables

To chain sweep variables:

1. Choose **Sweep Variable** in the destination **Variable** field.
2. Alternatively, choose **New Sweep Variable** in the destination **Variable** field and designate a **Name** for that new variable.
3. Expand the **Value Selector** section.



4. Choose the **Start Value** and **Stop Value** for the sweep. Available options are: **avg**, **min**, **max**, **first**, and **last**.
5. Specify a value for **Number of Steps**.
6. The **New sweep** for the sweep variable text box reflects the sweep options you chose. This is not editable.
7. Click **Apply** and notice the updates to the analysis in the main PrimeWave Design Environment window.

8. Click **OK** to chain the testbenches using your selections. The **Chain Testbenches** dialog box closes.
9. In the main PrimeWave Design Environment window, choose **Simulation > Netlist and Run**. Review the results.

Computing Interface Elements

It can be difficult to predict where the simulator will place interface elements (IEs) on the schematic, which could cause unexpected results. The PrimeWave Design Environment provides a quick way to compute IEs for the purpose of seeing where the simulator will place IEs using your current setup.

To compute interface elements:

1. Ensure your design is properly set up to run a simulation.
2. Select **Tools > Compute Interface Elements**.

The PrimeWave Design Environment automatically:

- Netlists the design (if needed).
- Runs the simulation with a stop time of 0ns.
- Shows the schematic(s).
- Calls the AMS Visualization assistant using the `interface_element.rpt` file.

For more information about AMS visualization, see **AMS Visualization Assistant** in the *Custom Compiler Schematic Editor User Guide*.

Preparing for Multi-Technology Simulation of 3-D Integrated Circuits (3DICs)

A three-dimensional integrated circuit (3DIC) is a single chip that integrates two or more layers of active electronic components both vertically and horizontally into a single circuit. All components on the layers communicate using on-chip signaling, whether vertically or horizontally. The PrimeWave Design Environment provides an intuitive way of setting up these modules. The generated netlist with 3DIC information is then available for use with the PrimeSim and FineSim simulators and with System-Verilog flows.

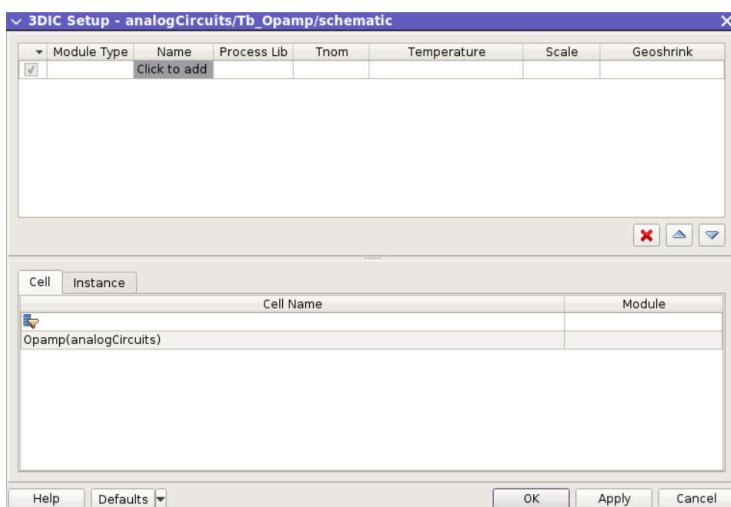
The 3DIC construct employs two commands, `.module` and `.endmodule`, to create a 3DIC-specific netlist block. These commands enable you to define the unique IC module entities without name labels or circuit properties and to avoid collision between different IC modules.

The `.modulevar` and `.endmodulevar` block enables you to define the unique IC module entities for each top-level instance instantiation.

To set up 3DIC:

1. Ensure your simulator is set to FineSim.
2. Select **Setup > 3DIC Setup**.

The **3DIC Setup** dialog box opens.

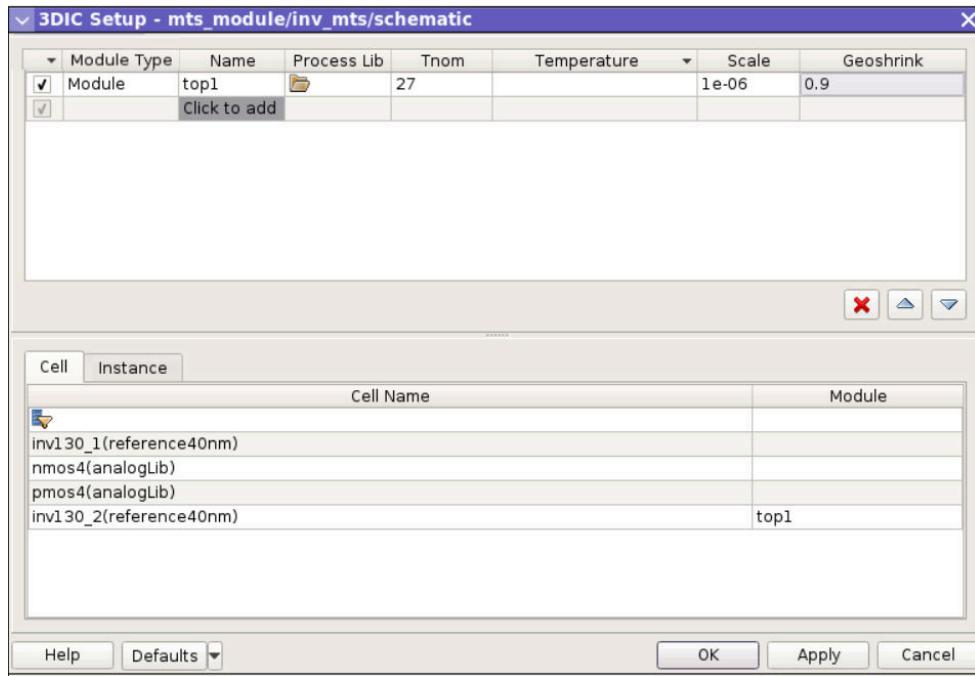


3. Fill in the options in the dialog box as described in [Table 23](#).

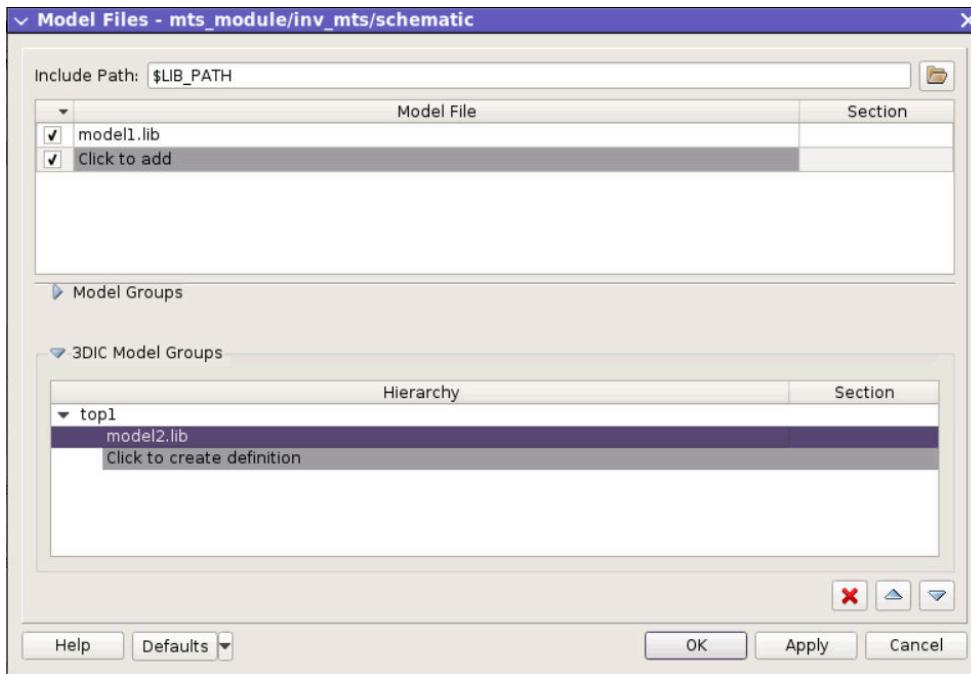
Table 23 3DIC Setup Dialog Box Options

Option	Description
Module Type	Options include Module (for cells) or ModuleVar (for instances).
Name	Select a name for the module or modulevar. Duplicate names are not permitted. If you select a duplicate name, the cell is outlined in red.
Process Lib	The IC module-specific model library. Click Process Lib to open the Model Files dialog box. See Step 6 .
Tnom	The reference temperature for the simulation.
Temperature	The circuit temperature for the simulation.
Scale	Sets the element scaling factor. If the Scale value is zero or negative, the cell is outlined in red.
Geoshrink	The element scaling factor used with <code>.option scale</code> .

4. Click in a cell in the **Name** column and define the name of your 3DIC module.
5. Select **Module** as the **Module Type**.



6. Click **Process Lib** to open the **Model Files** dialog box.

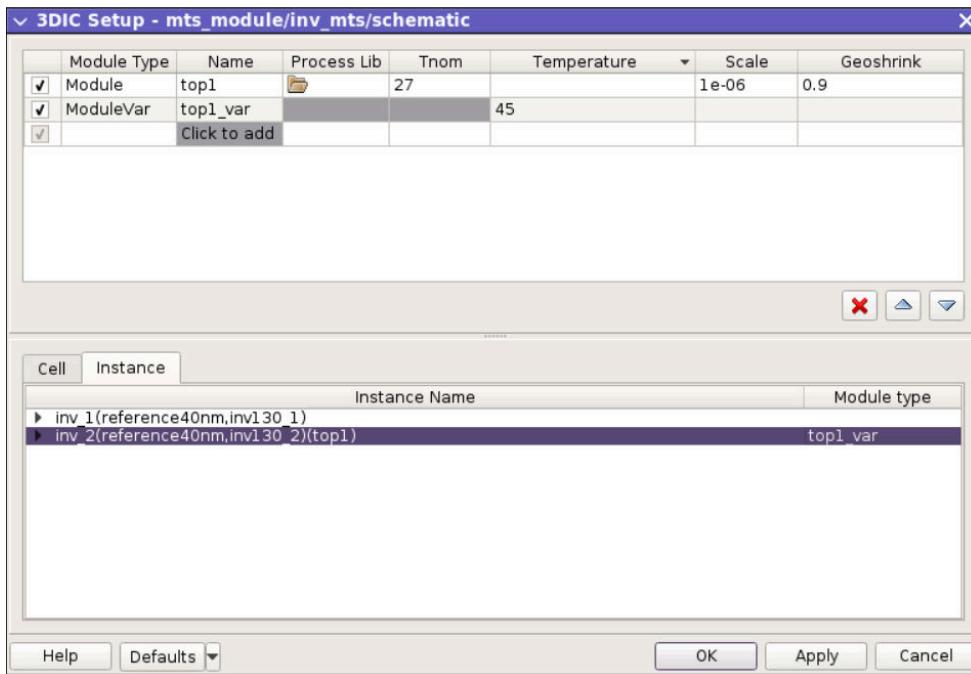


7. Browse to select the **Model File(s)** and **3DIC Model Group(s)** of your choice.
8. Click **OK** to apply the changes and close the **Model Files** dialog box.

The tool parses the design of this testbench, and fills in the **Cell** and **Instance** tables in the **3DIC Setup** dialog box.

Cell	Instance		
		Cell Name	Module
		inv130_1(reference40nm)	
		nmos4(analogLib)	
		pmos4(analogLib)	
		inv130_2(reference40nm)	top1

9. To set IC module instance-specific properties, click a cell in the **Name** column and define the name of your module variable block.
10. Set the **Module Type** to **ModuleVar**.
11. Set the variable values (**Temperature**, **Scale**, **Geoshrink**).
12. Assign this **ModuleVar** block to the specific IC module instance in the **Instance** tab.



13. Click **OK** to close the **3DIC Setup** dialog box.
14. Choose **Simulation > Netlist > Create** from the main menu bar.

As a result of setting up the **Module** in the **Cell** table and the **ModuleVar** in the **Instance** table of the **3DIC Setup** dialog box, the following netlist is generated:

```
* Generated for: FineSimPro
* Design library name: mts_module
* Design cell name: inv_mts
* Design view name: schematic
.option search='$LIB_PATH'
.option finesim_output=wdx
.temp 25
.include 'model1.lib'
*Custom Compiler Version Q-2020.03-SP2
*Tue Sep  8 18:42:27 2020
.global gnd gnd! vdd!
.option
  search='/slowfs/amscae18/flow_demos/SPICE/3DIC/simulation/mts_module,
inv_mts,schematic/history_1/simulation/FineSim_3DIC_1M/FineSimPro/nomi
nal/netlist'
.modulevar top1_var
.temp 45
.endmodulevar top1_var
* 3DIC wrappers added Tue Sep 08 18:42:28 IST 2020
***** 
* Library          : reference40nm
```

Chapter 18: Running Advanced Analyses
 Preparing for Multi-Technology Simulation of 3-D Integrated Circuits (3DICs)

```

* Cell           : inv130_1
* View          : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List : hspice hspiceD
*****subckt inv130_1 in out
mxmnl out in gnd gnd nch_lvt w=0.25u l=0.1u
mxmpl out in vdd vdd pch_lvt w=0.50u l=0.1u
.ends inv130_1
.module top1
.include top1.SAE.inc
*****Library      : reference40nm
* Cell           : inv130_2
* View          : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List : hspice hspiceD
*****subckt inv130_2 in out
mxmn2 out in gnd gnd nch_lvt w=0.25u l=0.1u
mxmp2 out in vdd vdd pch_lvt w=0.50u l=0.1u
.ends inv130_2
.endmodule top1
*****Library      : mts_module
* Cell           : inv_mts
* View          : schematic
* View Search List : hspice hspiceD schematic spice veriloga
* View Stop List : hspice hspiceD
*****xinv_1 in out1 inv130_1
v2 vdd! gnd! dc=1
v3 in gnd! dc=0 pulse ( 0 1 0 0.01n 0.01n 2n 4n )
xinv_2 in out2 top1::inv130_2 modulevar=top1_var
.tran 1p 32n start=0 simStart=0
.meas tran del2 trig v(in) val=0.5 rise=1 targ v(out2) val=0.5 fall=1
.meas tran dell1 trig v(in) val=0.5 rise=1 targ v(out1) val=0.5 fall=1
.probe tran v(*) level=1
.probe tran v(in) v(out1) v(out2) v(top1::*)
.option PARHIER = LOCAL
.option finesim_mode = spicead
.end

```

The module settings are written to the `top1.SAE.inc` file.

```

.option scale=1e-06
.option geoshrink=0.9
.option tnom=27
.include 'model2.lib'

```

The netlists are now available for 3DIC testing flows.

A

PrimeWave Design Environment Scripting

This appendix contains information on how to use PrimeWave Design Environment scripts to set up simulations and verify results outside of the PrimeWave Design Environment GUI.

The PrimeWave Design Environment's scripting feature provides a scripting layer for non-GUI simulation setup and verification. Scripting commands can perform all tasks that are possible in the GUI, including specifying design variables, analyses, outputs, and other testbench information. Commands are also provided for running analyses and postprocessing results.

This chapter contains the following major sections:

- [Invoking the PrimeWave Design Environment Shell](#)
 - [Creating PrimeWave Design Environment Scripts](#)
 - [Running PrimeWave Design Environment Scripts](#)
 - [Setting Shell Preferences](#)
-

Invoking the PrimeWave Design Environment Shell

The pw_shell can be invoked in one of two modes:

- Batch Mode

To invoke the PrimeWave Design Environment shell in batch mode, type `pw_shell <script_name>.tcl` on the command line. The specified Tcl script is run immediately and exits upon completion.

- Interactive Mode

To invoke the PrimeWave Design Environment shell in interactive mode, type `pw_shell` on the command line. In interactive mode, you are prompted for commands via a command line in a terminal window.

When invoked, the `pw_shell` reads the `.pw_shell.tcl` file from following sites:

- `SYNOPSYS_CUSTOM_INSTALL`
- `SYNOPSYS_CUSTOM_SITE`
- `SYNOPSYS_CUSTOM_PROJECT`
- `HOME`
- `SYNOPSYS_CUSTOM_LOCAL`

Note:

If you want to specify the location for a log file, use the following syntax when invoking `pw_shell`:

```
pw_shell -log <directory_path>
```

The last part of the part of the directory path is used as the prefix for log files.

Creating PrimeWave Design Environment Scripts

PrimeWave Design Environment scripts are made up of Tcl commands in the Simulation Scripting (or ss) namespace, which can be mixed in with the core PrimeWave Design Environment Tcl commands. The PrimeWave Design Environment scripting commands are not intended to duplicate the function of the core PrimeWave Design Environment Tcl commands, which are in the sa namespace. See the *Custom Compiler Tcl Reference Manual* for information on the ss and sa namespace commands.

In the PrimeWave Design Environment GUI, you can create a Tcl script of all the settings in your current testbench to use in `pw_shell`. See [Saving Tcl Scripts](#).

The following sample scripts are available:

- [Example 1: Initializing New PrimeWave Design Environment Sessions](#)
- [Example 2: Setting Up and Executing Simulations](#)
- [Example 3: Measuring a Bandgap Power Up](#)
- [Example 4: Measuring the Effect of Temperature on a Bandgap Output Versus Resistor Network Selected](#)

Example 1: Initializing New PrimeWave Design Environment Sessions

The following example shows how to initialize a PrimeWave Design Environment session :

```
namespace eval ::ss {
  newSession {
    resultsDir ~/simulation
    statesDir {}
  }
  setDesign DemoPLL/TB_inv_ss/schematic
  setSimulator HSPICE
  setTemperature 25C
  createAnalysis dc -options {
    designVar {}
    enabled 1
    name dc
    numPoints {}
    poi {}
    source /V1
    start 0
    stepSize 0.1
    stop 1.2
    sweep Source
    sweepType {Linear Steps}
  }
  setNode -ic {
    /in 0
  }
  setNode {
    /out 1
  }
  setEnvOptions {
    commandLineOptions {}
    enableMultiThreading 0
   .hpp 0
    includePath {}
    netlistLanguage HSPICE
    numThreads 1
    outputDataFormat psf
    stopViewList {hspice hspiceD veriloga}
    switchViewList {hspice hspiceD cmos.sch cmos_sch
      schematic veriloga}
    use64Bit 0
  }
  setIncludeFiles {
    [pwd] Options
  }
  setModelFiles {
    [pwd] TYP
```

```

}
}
```

Example 2: Setting Up and Executing Simulations

The following example shows how to set up and execute a simulation:

```

# evaluate scripts in the ss namespace to make it easier
# to call procs/commands.
namespace eval ::ss {
  setSimulator HSPICE
  setDesign PLL/test_vco/schematic
  setModelFiles "[pwd]/cln90g.l FF"
  # to visualize the setup
  setDesVars {
    vdc .7
    vtune 1
  }
  setTemperature -5F
  setSimOptions {
    ABSTOL 1e-6
    GMINDC 1e-8
  }
  createAnalysis tran -options {
    step .1p
    stop 1n
  }
  setIncludeFiles "[pwd]/myoptions.inc Options"
  # netlist
  # note that "run" is blocking.
  run

  plot "v(out)"
}
```

Example 3: Measuring a Bandgap Power Up

The following example shows how to measure the power up of a bandgap:

```

#####
### Bandgap simulation - powerup characterization
#
# Measure the time it takes to powerup a bandgap
#
# Criteria: The bandgap output must be between 1.1 and 1.19
#           Slope must be < 0.2
#
#####
set model_corners [list SS TT FF]
```

Appendix A: PrimeWave Design Environment Scripting

Creating PrimeWave Design Environment Scripts

```

set fileid [open "tb_scripts/sim_results_rt.txt" w]
set model_path /opt/technology/iPDK/models/hspice/toplevel.l
set w1 8
set w2 20
set w3 20
set threshold 5e-9

# Make a nice header (with separator) for the table first
set sep +-[string repeat - $w1]-+[string repeat - $w2]-+[string repeat
- $w3]
puts $fileid $sep
puts $fileid [format "| %-*s | %-*s | %-*s |" $w1 "Process" $w2 "Rise
Time" $w3 "Settling Time"]
puts $fileid $sep
close $fileid

namespace eval ::ss {
newSession {
    resultsDir ~/simulation
    statesDir {}
}
setDesign nr_lib/bg_ref_tb/schematic
setSimulator HSPICE
setTemperature 25C
createAnalysis op -options {
    enabled 1
    format All
    format2 All
    format3 All
    format4 All
    format5 All
    format6 All
    interpolation 0
    name op
    numFormats 1
    time 0
    time2 0
    time3 0
    time4 0
    time5 0
    time6 0
}
createAnalysis tran -options {
    enabled 1
    name tran
    numIntervals 1
    start 0
    step .1u
    step2 {}
    step3 {}
    step4 {}
    step5 {}
    stop 20u
}

```

Appendix A: PrimeWave Design Environment Scripting

Creating PrimeWave Design Environment Scripts

```

stop2 {}
stop3 {}
stop4 {}
stop5 {}
uic 0
}
setDesVars {
    vdd25 2.5
    vss25 0
    Wp 35u
    Wn 33.3u
    sw1 0
    sw2 2.5
}
setEnvOptions {
    commandLineOptions {}
    includePath {}
    netlistLanguage HSPICE
    outputDateFormat psf
    stopViewList {hspice hspiceD}
    switchViewList {hspice hspiceD cmos.sch cmos_sch schematic}
    use64Bit 0
}
setIncludeFiles {
    /home/technology/40nm/flat.inc Options
}
db::setAttr includePath -value / -of [db::getAttr models -of
    [getActiveTestbench]]
setModelFiles {
    /opt/technology/iPDK/models/hspice/toplevel.l TOP_FSG_LocalMC
}

setSaveOptions {
    allAnalogTC 0
    postLevel {}
    postTop {}
    power 0
    save all
    subcktCurrent 0
    totalPower 0
}

foreach corner $model_corners {
    setModelFiles [list ${model_path} TOP_${corner}G_LocalMC]
    netlist
    run

    set go_time [ss::expr cross(v(/out),1.1,1) -analysis tran]
    ss::expr ymax(v(/out)) -name yMax -analysis tran      set st_time
    [ss::expr cross(v(/out),1.19,1) -analysis tran]

    set slope_v 1;
}

```

```

set rt $go_time
while {[::expr {$st_time-$go_time}] > $threshold} {
    if {[ss::expr slope(v(/out),$go_time) -analysis tran] > 0.2} {
        set go_time [::expr {($go_time+$st_time)/2}]
        set slope [ss::expr slope(v(/out),$go_time) -analysis tran]
    }
    if {[ss::expr slope(v(/out),$st_time) -analysis tran] > 0.2} {
        set st_time [::expr {($go_time+$st_time)/2}]
        set slope [ss::expr slope(v(/out),$st_time) -analysis tran]
    }
    puts [format "go_time = %.19f" $go_time]
    puts [format "st_time = %.19f" $st_time]
    puts [format "slope = %.19f" $slope]
}
set fileid [open "tb_scripts/sim_results_rt.txt" a]
puts $fileid [format "| %-*s | %-*f | %-*f |" $w1 $corner $w2
$rt $w3 $go_time]
close $fileid
}
}

```

Example 4: Measuring the Effect of Temperature on a Bandgap Output Versus Resistor Network Selected

The following example shows how to measure the effect of temperature on the bandgap output versus resistor network selected:

```

#####
### Bandgap simulation
#
# Switch SW1 & SW2 and report results in a table vs TEMP
#
#####
set temperatures [list 20 30]
set dt 10
set switch_states [list 00 01 10 11]
set switch_high 2.5
set fileid [open "tb_scripts/sim_results.txt" w]
set w1 5
set w2 3
set w3 3
set w4 10

# Make a nice header (with separator) for the table first
set sep +-[string repeat - $w1]-+-[string repeat - $w2]-+-[string repeat
- $w3]-+-[string repeat - $w4]
puts $fileid $sep
puts $fileid [format "| %-*s | %-*s | %-*s | %-*s" $w1 "30C-20C" $w2
"Sw1" $w3 "SW2" $w4 "dV/dt"]

```

Appendix A: PrimeWave Design Environment Scripting

Creating PrimeWave Design Environment Scripts

```
puts $fileid $sep
close $fileid

namespace eval ::ss {
newSession {
    resultsDir ~/simulation
    statesDir {}
}
setDesign nr_lib/bg_ref_tb/schematic
setSimulator HSPICE
setTemperature 25C
createAnalysis op -options {
    enabled 1
    format All
    format2 All
    format3 All
    format4 All
    format5
    format6 All
    interpolation 0
    name op
    numFormats 1
    time 0
    time2 0
    time3 0
    time4 0
    time5 0
    time6 0
}
createAnalysis tran -options {
    enabled 1
    name tran
    numIntervals 1
    start 0
    step .1u
    step2 {}
    step3 {}
    step4 {}
    step5 {}
    stop 20u
    stop2 {}
    stop3 {}
    stop4 {}
    stop5 {}
    uic 0
}
setDesVars {
    vdd25 2.5
    vss25 0
    Wp 35u
    Wn 33.3u
    sw1 0
    sw2 2.5
}
```

Appendix A: PrimeWave Design Environment Scripting

Creating PrimeWave Design Environment Scripts

```

}
setEnvOptions {
    commandLineOptions {}
    includePath {}
    netlistLanguage HSPICE
    outputDateFormat psf
    stopViewList {hspice hspiceD}
    switchViewList {hspice hspiceD cmos.sch cmos_sch schematic}
    use64Bit 0
}
setIncludeFiles {
    /home/user/technology/40nm/flat.inc Options
}
db::setAttr includePath -value / -of [db::getAttr models -of
    [getActiveTestbench]]
setModelFiles {
    /opt/technology/iPDK/models/hspice/toplevel.l TOP_FSG_LocalMC
}
setSaveOptions {
    allAnalogTC 0
    postLevel {}
    postTop {}
    power 0
    save all
    subcktCurrent 0
    totalPower 0
}
set x 1
foreach state $switch_states {
    set fileid [open "tb_scripts/sim_results.txt" a]    set sw1V [::expr
    {$switch_high*[string index $state 0]}]
    set sw2V [::expr {$switch_high*[string index $state 1]}]
    setDesVars [list sw1 $sw1V sw2 $sw2V]
    set volt_list {}
    foreach temp $temperatures {
        puts "Run #$x"
        setTemperature ${temp}C
        netlist
        run
        lappend volt_list [ss:::expr ymax(v(/out)) -analysis tran]
        incr x
    }
    set volt1 [lindex $volt_list 0]
    set volt2 [lindex $volt_list 1]
    set dV [::expr {abs($volt1-$volt2)}]
    set tcoeff [::expr {$dV/$dT}]
    puts $fileid [format "| %-*1f | %-*2f | %-*2f | %-*9f" $w1 $dT $w2
    $sw1V $w3 $sw2V $w4 $tcoeff]
    close $fileid
}
}

```

Running PrimeWave Design Environment Scripts

To run a script in pw_shell batch mode, enter the following text on the command line:

```
pw_shell <script_name>.tcl
```

To run a script in pw_shell interactive mode, invoke pw_shell (type `pw_shell` on the command line), then source the desired Tcl script. Scripts can be run in Custom Compiler, as well.

Setting Shell Preferences

You can set preferences for your shell session, such as the maximum number of log files to create, in a file named `.pw_shell.tcl`.

See the *Custom Compiler Tcl Reference Manual* for more information on preferences.

B

PrimeWave Design Environment Shortcut Keys

This appendix provides the binding set available with the default installation.

The following sections describe the PrimeWave Design Environment shortcut keys.

- [PrimeWave Design Environment Function Shortcut Keys](#)
- [PrimeWave Design Environment Keypad Shortcut Keys](#)

PrimeWave Design Environment Function Shortcut Keys

The following table describes the available function shortcut keys when using the Custom Compiler (default) or Maestro binding sets. For more information about customizing the binding set, see the *Custom Compiler Environment User Guide*.

For the list of keypad shortcut keys, see [PrimeWave Design Environment Keypad Shortcut Keys](#).

Table 24 PrimeWave Design Environment Function Shortcut Keys

Shortcut Key	Description
F1	Help

PrimeWave Design Environment Keypad Shortcut Keys

The following table describes the available main keypad shortcut keys when using the Custom Compiler (default) or Maestro binding sets. For more information about customizing the binding set, see the *Custom Compiler Environment User Guide*.

For the list of function shortcut keys, see [PrimeWave Design Environment Function Shortcut Keys](#).

Table 25 PrimeWave Design Environment Keypad Shortcut Keys

Shortcut Key	Description
A	Opens the Create/Edit Analyses dialog box
B	Back-annotates DC node voltages in a schematic
C	Opens the Convergence Aids dialog box
D	Opens the Save Script dialog box
E	Opens the Environment Options dialog box
L	Opens the Load State dialog box
M	Opens the Model Files dialog box
N	Creates a single testbench netlist
O	Opens the Simulator Options dialog box
P	Opens the Results Analyzer dialog box
Q	Back-annotates DC operating point information in a schematic
R	Runs a single testbench
S	Opens the Save State dialog box
V	Opens the Edit Design Variables dialog box
Delete	Deletes selected objects
Ctrl+C	Opens a prompt to select outputs from design
Ctrl+D	Shows the design in the Schematic Editor
Ctrl+L	Displays the simulation log
Ctrl+N	Opens a new window
Ctrl+O	Opens a saved test suite.
Ctrl+O (MTB only)	Opens a multiple testbench (MTB) testsuite
Ctrl+P	Opens the Schematic Editor for adding output from a design
Ctrl+R	Netlists and runs a single testbench simulation
Ctrl+R (MTB only)	Runs a multiple testbench (MTB) simulation
Ctrl+S	Saves the current test suite

Table 25 PrimeWave Design Environment Keypad Shortcut Keys (Continued)

Shortcut Key	Description
Ctrl+S (MTB only)	Saves a multiple testbench (MTB) test suite
Ctrl+T	Opens a new tab
Ctrl+V	Saves design variables to a design
Ctrl+W	Closes the PrimeWave Design Environment session
Ctrl+Space	Switches to the Console window
Shift+A	Opens a prompt to select a net from a design and then plot it as an AC Bode diagram
Shift+B	Back-annotates transient node voltages in a schematic
Shift+C	Opens the Corners Setup dialog box
Shift+M	Opens the Monte Carlo Setup dialog box
Shift+N	Displays the design netlist
Shift+O	Open the Output Setup dialog box
Shift+P	Opens the Parametric Analysis dialog box
Shift+Q	Back-annotates transient operating point information in a schematic
Shift+S	Opens the Simulator Setup dialog box
Shift+T	Opens a prompt to select a net from a design and then plot it as a transient signal
Shift+V	Copies the design variables from a design
Ctrl+Shift+A	Opens the Results Analyzer dialog box
Ctrl+Shift+C	Opens the Results Compare tool
Ctrl+Shift+H	Opens the Charts dialog box
Ctrl+Shift+N	Creates a netlist
Ctrl+Shift+N (MTB only)	Creates a multiple testbench netlist
Ctrl+Shift+R (MTB only)	Netlists and runs a multiple testbench (MTB) simulation
Ctrl+Shift+V	Opens the ResultsView

C

Tools for Job Status and Monitoring

This chapter describes how to use monitoring tools with Custom Compiler to track and improve the tool performance.

This chapter contains the following topics:

- [Job Monitor Tool](#)
 - [Sentry Monitor Library Tool](#)
 - [Watchdog Tool](#)
-

Job Monitor Tool

Use the Job Monitor to track running job processes.

By default, the Job Monitor tracks batch processes. Interactive process tracking is also available. (See [Viewing Running Job Processes](#).)

In addition to using the Job Monitor to track jobs, you can use Tcl preferences to add *job classes* that combine multiple grid settings into one setting. You can also use Tcl preferences to add job process waiting to allow scripts to finish without interruption.

This topic describes the following:

- [Displaying the Job Monitor Tool](#)
- [Viewing Running Job Processes](#)
- [Applying Job Process Controls](#)
- [Managing Job Options](#)
- [Filtering Job Groups](#)
- [Displaying a Netlist](#)
- [Adding Job Classes With `xt::createJobClass` Tcl](#)
- [Adding Job Process Waiting With `xt::wait` Tcl](#)

Displaying the Job Monitor Tool

Open the Job Monitor to view current session's batch and interactive processes.

To open the Job Monitor:

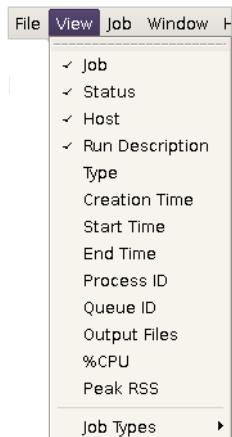
- In any design window, choose **Tools > Job Monitor**.

Viewing Running Job Processes

You can view job processes as they run with the Job Monitor.

To view a running job process:

1. Begin by opening the Job Monitor Tool. In any design window, choose **Tools > Job Monitor**.
2. In the Job Monitor, choose **View**.
3. In the **View** menu, toggle [Job Monitor View Options](#). The system adds and removes tracking columns as directed.



[Table 26](#) describes the Job Monitor view options.

Table 26 Job Monitor View Options

Option	Description
Job	Displays a tree of job groups and jobs, with leaf nodes for each running application.

Option	Description
Status	Displays the status of the process, such as FINISHED, QUEUED, RUNNING, or KILLED. Progress feedback for the job is displayed as a bar graph along with the percentage completed. In addition, the status text is color-coded as follows: <ul style="list-style-type: none">• FAILED - bright red• KILLED - dark red• Others - black (default text color) Double-click the status to open the Text Viewer and view associated output files.
Host	Name of host associated with this job, with local host noted if the job is running on this system.
Run Description	Displays the description associated with the job.
Type	The job type (group, batch, or interactive).
Creation Time	Specifies the job creation time.
Start Time	Time job was initiated.
End Time	Time job ends.
Process ID	ID as listed on the host.
Queue ID	The grid identifier associated with the job (only applies if the job was launched via SGE or LSF).
Output Files	Expected output files from this job as defined when the job was created.
%CPU	CPU usage of the job.
Peak RSS	Peak resident memory of the job.
Job Types	Toggles between the display of batch and interactive jobs.

4. You can choose to show interactive job processes in addition to batch job processes by choosing **View > Job Types**, and toggling the **Interactive** option. (Alternatively, use the `xtJobMonitorShowInteractiveJobs` preference.)
5. Choosing **File > Close** closes the Job Monitor.

Applying Job Process Controls

You can apply job process controls that suspend, resume, kill, and delete jobs.

To apply job process controls:

1. Begin by opening the Job Monitor Tool. In any design window, choose **Tools > Job Monitor**.
2. In the Job Monitor, choose **Job**.
3. Select the job by name.
(This step is not required when you are deleting all completed jobs.)
4. Choose one of the job process controls.

[Table 27](#) describes the job process controls.

Table 27 Job Process Controls

Option	Description
Suspend	Halts this process
Resume	Resumes the halted process
Kill Selected Jobs	Ends this process
Kill Selected Jobs and All Remaining Jobs	Ends this process and any remaining processes
View Output	Displays output in the Text Viewer
Delete	Removes this job
Delete All Completed	Removes records of all completed jobs

5. Choosing **File > Close** closes the Job Monitor.

Managing Job Options

You can manage job options to control batch processing limits, and to optionally set a queue engine. You can optionally specify a job class, which assigns queue engine resources based on a predefined job type.

To manage job options:

1. Begin by opening the Job Monitor Tool. In any design window, choose **Tools > Job Monitor**.
2. In the Job Monitor, choose **File > Options**.
3. Choose job management options.

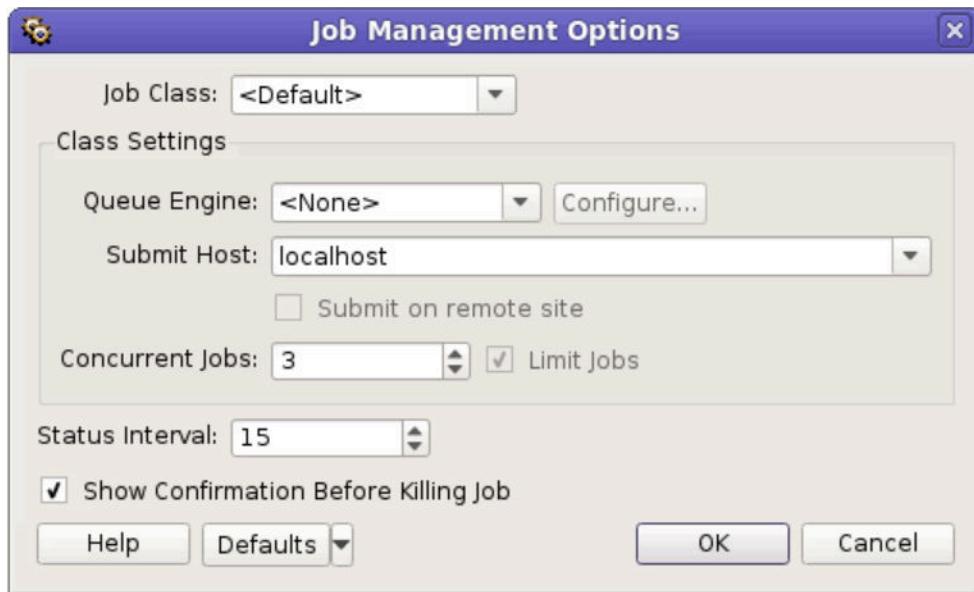


Table 28 describes the job management options.

Table 28 Job Management Options

Option	Description
Job Class	Specifies a setting that automatically includes an analysis process, a queue engine, whether concurrent jobs are allowed, and the number of allowable concurrent jobs. CAD Support teams can add custom job classes. For details, see Adding Job Classes With xt:createJobClass Tcl .
Class Settings	Queue Engine sets the distributed computing engine for batch processing to SGE (Sun Grid Engine) or LSF (Load Sharing Facility), or no queue engine. To set additional options for an engine, click Configure , and choose Configuration options. (For information about SGE option settings, refer to the Sun Grid Engine documentation. For information about LSF option settings, refer to the Load Sharing Facility documentation.) Submit Host is the host where the jobs will be launched. Choose Submit on remote site if this job class uses a remote (WAN) grid. Limit Jobs limits the concurrent jobs in this job class to the number you set in the Concurrent Jobs box.

Option	Description
	Concurrent Jobs sets the number of allowable jobs to process at one time, based on the job class you have set above. (You can create custom job classes. See Adding Job Classes With xt::createJobClass Tcl.)
Status Interval	Sets the number of seconds between each queue engine status poll.
Show Confirmation Before Killing Job	Allows you to control whether or not the Kill Jobs prompt appears before you end a job.

If you want to limit jobs to control license usage, and you do not have access to a queue engine for distributed processing, you can set the number of concurrent jobs to 1. This will effectively create a queue.

4. Click **OK**.
5. Choosing **File > Close** closes the Job Monitor.

Note:

See the Custom Compiler Application Note "Setting Up the Computer Network" for more details on configuring the **Job Management Options** dialog box.

Customizing the Job Name

You can customize the job name when launching simulations to the grid engine. The procedure `::sa::run::createJobArrayName` should be used to override the default implementation in which the format `testbenchName_simulatorName` is used for the job name.

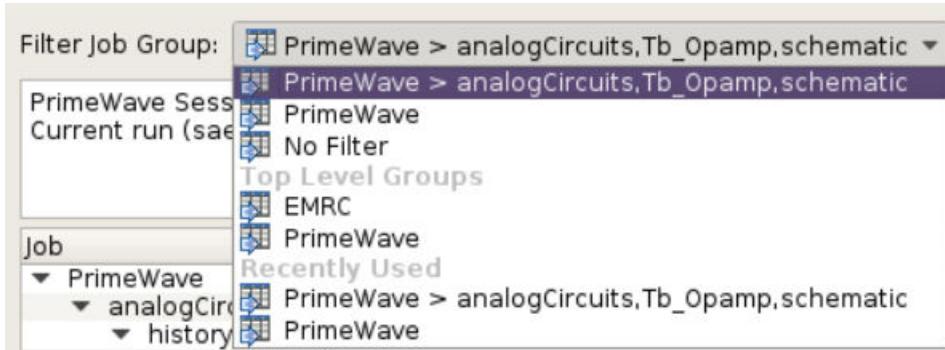
```
namespace eval ::sa::run {
proc createJobArrayName { tb }
{ set tbName [db::getAttr tb.name] set simName
[db::getAttr tb.simulator]
return "${tbName}_${simName}" }}
```

Filtering Job Groups

You can filter jobs that are visible in the Job Monitor in order to, for example, change from a specific PrimeWave Design Environment session to a more general one. The Job Monitor also remembers the previous filters so you can go back and forth between filtered views.

Select the job group you wish to filter from the **Filter Job Group** menu.

Figure 1 *Filter Job Group*



You can also change the current filter to a selected job group by right-clicking on a job group in the job tree and choosing **Filter to this job group** from the menu.

Displaying a Netlist

You can display a netlist from the Job Monitor. To display the netlist, select a specific iteration, right-click to display the menu, and choose **View Netlist**.

Adding Job Classes With `xt::createJobClass` Tcl

You can create *job classes* that combine settings for individual analysis tools and system environment grids into one setting option.

For example, if you want HSPICE tool simulation jobs to run on Solaris-64 machines via LSF, and to limit the number of concurrent jobs to 10, you can create a job class that contains these settings.

In use, designers can open the Job Monitor, and select this job class to apply the settings.

This documentation provides two script samples that create custom job classes. To use them, choose the one that most closely matches your requirements, adjust values to create an exact match, and implement the script. You can also write an original script using the supplied Tcl commands and preferences.

This section covers Tcl samples:

- [Sample Job Class for Design Rule Checking on Linux Systems](#)
- [Sample Job Class for HSPICE Simulation on Solaris Systems](#)

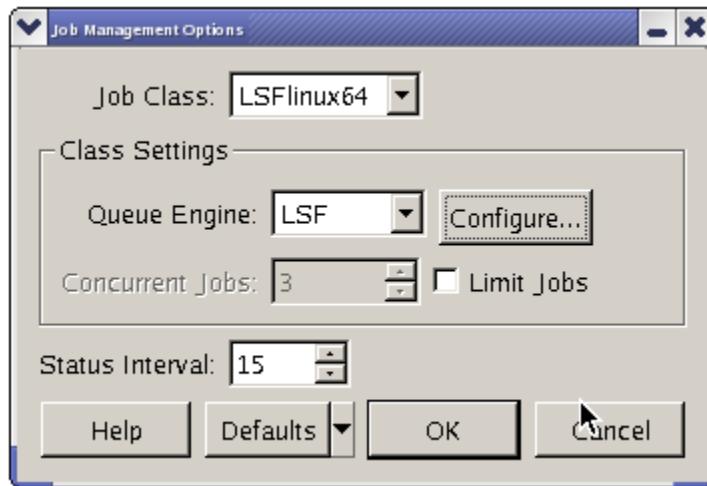
Sample Job Class for Design Rule Checking on Linux Systems

This Tcl sample configures XA and Hercules tool DRC jobs to run on Linux-64 machines via LSF, without limiting the number of concurrent jobs.

You can implement this job class using the sample code that follows.

In use, designers can specify this job class by opening the Job Monitor, choosing **File Options**, and selecting **LSFlinux64** as shown in [Figure 2](#).

Figure 2 Sample Job Class for Design Rule Checking on Linux Systems



```
# Job Class for Design Rule Checking on Linux Systems# For reference,
this sample lists available job class options.# General options:
#   JobQueueEngine
#   JobLimit
#   JobLimitEnabled
# LSF-specific options:
#   LSFQueues
#   LSFResources
#   LSFExtraOptions
# SGE-specific options:
#   SGEProjects
#   SGESelectedProject
#   SGEEExtraOptions
#
# Configure XA and Hercules DRC jobs to run on
# linux-64 machines via LSF with no limiting.
#
set lsflnx [xt:::createJobClass LSFlinux64]
db:::setPrefValue [xt:::getJobClassOptionPrefName $lsflnx \
    -option "JobQueueEngine"] -value "LSF"
db:::setPrefValue [xt:::getJobClassOptionPrefName $lsflnx \
    -option "JobLimitEnabled"] -value false
db:::setPrefValue [xt:::getJobClassOptionPrefName $lsflnx \
```

```
-option "LSFResources"] -value "SOL64 0 X86_64 1"
xt::addJobClassPrefix XA -jobClass $lsflnx
xt::addJobClassPrefix hercules_drc -jobClass $lsflnx
#
```

See Also

- [Sample Job Class for HSPICE Simulation on Solaris Systems](#)
- [Adding Job Classes With xt::createJobClass Tcl](#)

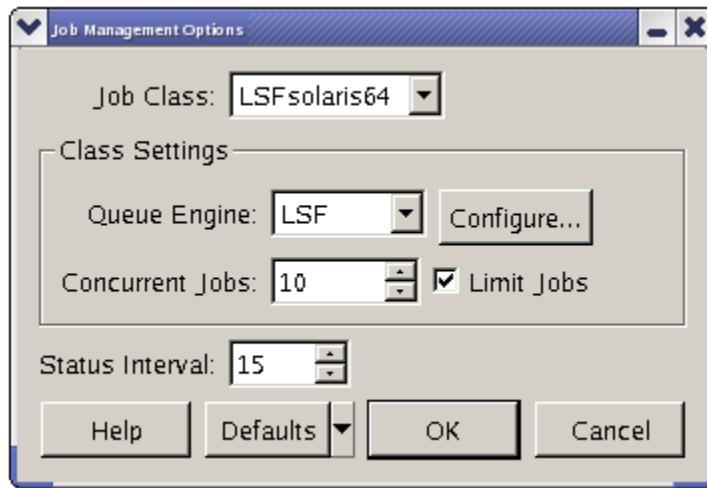
Sample Job Class for HSPICE Simulation on Solaris Systems

This Tcl sample configures HSPICE tool simulation jobs to run on Solaris-64 machines via LSF, and limits the number of concurrent jobs to 10.

You can implement this job class using the sample code that follows.

In use, designers can specify this job class by opening the Job Monitor, choosing **File Options**, and selecting **LSFsolaris64** (Figure 3).

Figure 3 Sample Job Class for HSPICE Simulation on Solaris Systems



```
# Job Class for HSPICE Simulation on Solaris Systems# For reference, this
sample lists available job class options.
#
# General options:
#   JobQueueEngine
#   JobLimit
#   JobLimitEnabled
# LSF-specific options:
#   LSFQueues
#   LSFResources
#   LSFEExtraOptions
```

```

# SGE-specific options:
#     SGEProjects
#     SGESelectedProject
#     SGEEExtraOptions
#
# Configure HSPICE jobs to run on solaris-64 machines
# via LSF, limited to 10 jobs at a time.
#
set lsfsol [xt::createJobClass LSFsolaris64]
db::setPrefValue [xt::getJobClassOptionPrefName $lsfsol \
    -option "JobQueueEngine"] -value "LSF"
db::setPrefValue [xt::getJobClassOptionPrefName $lsfsol \
    -option "JobLimit"] -value 10
db::setPrefValue [xt::getJobClassOptionPrefName $lsfsol \
    -option "JobLimitEnabled"] -value true
db::setPrefValue [xt::getJobClassOptionPrefName $lsfsol \
    -option "LSFResources"] -value "SOL64 1 X86_64 0"
xt::addJobClassPrefix HSPICE -jobClass $lsfsol
#

```

See Also

- [Sample Job Class for Design Rule Checking on Linux Systems](#)
- [Adding Job Classes With xt::createJobClass Tcl](#)

Adding Job Process Waiting With xt:::wait Tcl

In Tcl scripts, you can add job process waiting, which blocks user interaction until currently running jobs finish. Job process waiting is useful when your script needs to use the results of any currently running job as input for its next operation.

To add job process waiting to a Tcl script:

1. Identify the job.

- If you created the job using the `xt:::createJob` command, use the returned job object.
- If you want to add waiting to a standard system job process, use the job as listed in the Job Monitor **Job** column.

To find the job, run the process once using the graphic user interface. Then, in the Job Monitor **Job** column, take note of the process name.

For example, to add waiting to the IC Compiler export process, choose **Export > To ICC**, and run the process. In the Job Monitor **Job** column, note that the Job Monitor identifies the process as **dbExportICC**.

2. Using the identified job, add the `xt:::wait` command to the Tcl script at the line where you want the script to wait.

In the example that follows, `xt:::getJobs` gets all IC Compiler export jobs. It uses `xt:::wait` to cause the system to wait for these jobs to finish before executing the next Tcl command. While the system waits, it blocks user input. After the jobs finish, the script continues.

The system individually names each job process by appending the underscore character (_) and adding incremental numbering. For instance, the first dbExportICC job is `dbExportICC_1` and the fifteenth is `dbExportICC_15`. The example uses the wildcard character (*) to capture all `dbExportICC` jobs.

```
# wait for any dbExportICC jobs to finish
db::foreach xtJob [xt:::getJobs dbExportICC*] {
    xt:::wait $xtJob
}
```

3. Implement your Tcl script.

For details about running Tcl scripts on demand, see [Running Tcl Scripts](#).

For information about making Tcl scripts available from the installation tree, see [Defining Startup Settings](#).

Sentry Monitor Library Tool

The Sentry Monitor Library is a computer and network monitoring tool that reports error messages for computer and network-related performance issues while you are running a Synopsys software product.

See [Synopsys Sentry Monitor Library Error Messages](#).

The Sentry Monitor Library monitors and reports the following types of issues:

- High CPU use (host and processes)
- High memory use (physical memory, cache, and swap)
- Poor network conditions (packet transmission issues, host latency)

Watchdog Tool

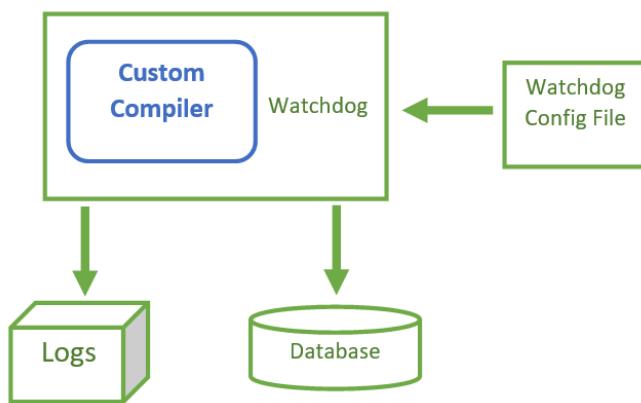
The Watchdog tool provides information to assist Design and CAD teams in analyzing tool usage and quality for their users. The main functionality of the Watchdog tool is to collect statistical information about Custom Compiler sessions. This version of the Watchdog tool can also copy Custom Compiler log files (only crashes or all logs) to a central location. By dividing the total session open time by the number of crashes, you can calculate the crash

rate of the Custom Compiler tool. This data can also be used to analyze common crashes and work with your Synopsys representative to determine root causes and resolve them more quickly. Overall, the Watchdog tool can help increase the stability of the Custom Compiler tool.

The Watchdog tool uses the process ID of the Custom Compiler session. It is attached to the Custom Compiler session and starts a collection of statistics and machine information.

The Watchdog tool writes the collected data into a database and copies the log file after the session is ended.

The collection of statistics and/or copying of log files can be configured in the Watchdog config file.



This topic describes the following:

- [Installing the Watchdog Tool](#)
- [Watchdog Outputs](#)
- [Watchdog Report Statistics](#)

Installing the Watchdog Tool

To install the Watchdog tool:

1. Create a directory to collect all watchdog statistic data including log files.

Note:

All users should have read-write access to the created directory.

The directory should be available to all users and from all machines (i.e. a shared location).

2. Add the following environment variable:

bash:

```
export SYNOPSYS_CUSTOM_WATCHDOG_DIR=<Directory from 1st step>
```

csh:

```
setenv SYNOPSYS_CUSTOM_WATCHDOG_DIR <Directory from 1st step>
```

3. Create a `watchdog.cfg` configuration file in the `$SYNOPSYS_CUSTOM_WATCHDOG_DIR` directory.

The configuration file should contain the following:

```
[Watchdog]
collectStatistics = True/False
collectLogs = True/False
collectNonCrashLogs = True/False
```

The above options control the following features of the Watchdog tool:

- `collectStatistics` - Enables collection of statistical data in the `$SYNOPSYS_CUSTOM_WATCHDOG_DIR/data.sqlite` database.
- `collectLogs` - Enables copying of crash logs only into watchdog statistics directory (from 2nd step).
- `collectNonCrashLogs` - Works with the `collectLogs` option and enables copying non-crash logs, as well.

The Watchdog tool runs in the background for each Custom Compiler session after you complete the above installation procedure.

Watchdog Outputs

The Watchdog tool saves all statistical information into a database and copies log files into a central location. Both features can be disabled and enabled in the configuration file.

Database

The Watchdog tool collects all statistic information for all users in one SQLite database named `data.sqlite`.

Database Structure

Statistics data for each version of Custom Compiler is collected in separate tables in the database.

Each table has the following columns:

- Max RAM and Virtual Memory usage
- CPU Time for parent and children processes (children processes are external processes run via Custom Compiler)
- User usage time based of command execution frequency
- Start and End Times (Real time)
- Crash stack and exit code
- Copied log file location (nothing if feature disabled)
- Machine info
 - Total RAM
 - Total Swap
 - CPU count
 - Platform
- User info
 - User ID
 - User Group
- Tool version info
 - Tool name
 - Config number

Database example:

The screenshot shows the SQLite Database Browser interface with the 'data.sqlite' database selected. The 'Tables' tab is open, displaying the 'Master Table (1)' and 'N-2017.12-DEV' table. The 'N-2017.12-DEV' table has 9 rows of data, each containing information such as process ID (cwid), start and end times, user ID (User), and log file paths. The columns include: cwid, User, StartTime, EndTime, LogFile, Config, ExitCode, UserGroup, Children, CPUTime, EndTime, Platform, ToolName, CrashSig, CPUs, UsageTc, MaxVirt, TotalRAM, and RestTime.

	cwid	User	StartTime	EndTime	LogFile	Config...	ExitCode	UserGroup	Children...	CPUTime	EndTime	Platform	ToolName	CrashSig...	Cpus	UsageTc...	MaxVirt...	TotalRAM	RestTime
1	269	129146	43.114416...	1240	[redacted]	1117	0	1632	129010	Linux-2.6.32-...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]
2	269	129146	59.532359...	1259	[redacted]	1117	0	1632	129010	Frame X...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]
3	269	129146	33.341045...	1174	[redacted]	1117	0	1632	129010	Linux-2.6.32-...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]
4	267	129146	22.705914...	1178	[redacted]	1117	0	1632	129010	Linux-2.6.32-...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]
5	273	129146	47.455365...	1545	[redacted]	1117	0	1632	129010	Frame X...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]
6	266	129146	37.762795...	1229	[redacted]	1117	0	1632	129010	Frame X...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]
7	267	129146	43.866935...	1240	[redacted]	1117	0	1632	129010	Linux-2.6.32-...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]
8	271	129146	38.830126...	1545	7.019999...	1117	0	1632	129010	Frame X...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]
9	267	129146	28.784423...	1173	[redacted]	1117	0	1632	129010	Linux-2.6.32-...	[redacted]	[redacted]	[redacted]	0	16341	[redacted]	[redacted]	[redacted]	[redacted]

Log Files

All copied log files can be found in the root `logs` directory located in the Watchdog output directory. The log files are copied into `yyyy/month/dd` directories corresponding to the date of copy.

Log directory example:

```
Logs
+-2016
| +-Dec
| | +-26
| | | +-<copied log files>
| | +-27
| | | +-<copied log files>
| | +-29
| | | +-<copied log files>
+-2017
| +-Jan
| | +-04
| | | +-<copied log files>
| | +-08
| | | +-<copied log files>
| | +-21
| | | +-<copied log files>
| +-Feb
| | +-01
| | | +-<copied log files>
```

Watchdog Report Statistics

The `wreport.py` utility can be used to generate high-level report from watchdog database. It is in the watchdog directory (for example, `1<CC install dir>/auxx/watchdog/wreport.py`).

The usage of the utility is the following:

```
wreport.py [-r <release name>] -d <database location>
```

Use `-h(--help)` for help.

If the release option is not set the utility will report for all existing releases.

The following is a simple example of `wreport.py` output:

```
<Release name>
Total Sessions      : 97
Crashed Sessions   : 6
Unique Crashes     : 4
Users              : 7
CPU Time           : 4.865 hours of cpu usage
Elapsed Time       : 117.879 hours tool was up
Usage Time         : 4.745 hours tool activly used
MTBC               : 0.791 hours (Usage Time / Crashed Sessions)
Longest Session    : 0.783 hours across all sessions
                           : 0.029 hours across crashed sessions
Shortest Session   : 0.000 hours across all sessions
                           : 0.000 hours across crashed sessions
```

```
Average Session : 0.049 hours across all sessions
                  : 0.012 hours across crashed sessions
```

Query on Database

If the report generated by the `wreport.py` does not provide enough information, you can use database queries for get more information.

You can get more database info using the following methods:

- [sqlite3 Command](#)
- [List the Tables](#)
- [Convert Database](#)
- [Number of Sessions](#)

sqlite3 Command

Use the sqlite3 command:

```
sqlite3 [OPTIONS] database [SQL Commands]
```

List the Tables

`data.sqlite` can contain several tables which are corresponding to the versions of Custom Compiler.

To list the names of the tables you can use the following command:

```
sqlite3 data.sqlite "SELECT name FROM sqlite_master WHERE type='table'"
```

Choose the version you want and do the query on it.

Convert Database

Sqlite3 allows you to format the output. The useful formats are CSV and HTML.

Use the following command to convert `data.sqlite` to CSV:

```
sqlite3 -header -csv data.sqlite "SELECT * FROM '<table name>'" >
{output}
```

Number of Sessions

Use the following command to get the number of overall sessions for the version of the Custom Compiler tool:

```
sqlite3 data.sqlite "SELECT count(*) FROM '<table name>'"
```

Use the following command to get the number of crashed sessions:

```
sqlite3 data.sqlite "SELECT count(*) FROM '<table name>' WHERE ExitCode
<> 'None' AND ExitCode <> 0"
```

D

MOSRA Integration with the PrimeWave Design Environment

This appendix describes the supported simulator MOSRA integrations in the PrimeWave Design Environment. It provides instructions on how to set up a combined stress + degraded simulation to determine and review aging-related changes to device models and circuit measurements + waveforms. In this case, the same stimulus is used in both stress + degraded analysis phases, and the same analyses are performed in both phases. An alternate set of instructions is also provided on how to perform explicitly separate stress + degraded simulations (which allow for different stimulus and/or analyses to be specified).

The following sections are included:

- [Introduction to MOSRA](#)
 - [Prerequisites](#)
 - [Recommended Steps](#)
 - [PrimeWave Design Environment MOSRA Flow Example](#)
 - [Conclusion](#)
-

Introduction to MOSRA

In MOS integrated circuits, device aging is mainly due to the degradation of the gate dielectric and of the interface between gate dielectric and silicon over time. Two important mechanisms that contribute to such degradation are the Hot Carrier Injection (HCl) and the Bias Temperature Instability (BTI). These mechanisms are more prominent in advanced process nodes in which the gate oxide is scaled to only a few molecules in equivalent thickness, and with the use of high-K metal-gate transistors. Long and expensive testing is required to assess the degradation of circuit performance and failure in time (aging), thus increasing the overall manufacturing cost. Alternatively, designers use conservative rules to overdesign the critical circuits, increasing the chip cost. Therefore, a cost-effective way to estimate the lifetime of circuits, especially in mission-critical applications (for example, automotive electronics), is essential.

MOS Reliability Analysis (MOSRA) in PrimeSim HSPICE, FineSim, and FineSim offers a robust and economic alternative to empirical overdesign and extensive lifetime testing. With either the built-in model or user-specified aging model, MOSRA accurately predicts the HCI and BTI aging effect on circuit performance. MOSRA enables designers to detect reliability failures early in the design process, significantly reducing the time and cost of lifetime testing.

With tight integration between the simulators and the PrimeWave Design Environment, the aging analysis can be performed as fast as the typical transient simulation runs. The effects of aging on circuit behavior (waveforms and measurements) can easily be inspected. The waveforms and measurements of ‘fresh’ devices can be compared with those of ‘aged’ devices. The detailed effects of aging in terms of its impact on degradation of key MOSFET compact model parameters (such as threshold voltage, mobility, and so forth) can easily be inspected in a convenient graphical user interface (GUI).

MOSRA Aging Models

Device aging is a result of continuous degradation of device characteristics, under the applied electrical stress.

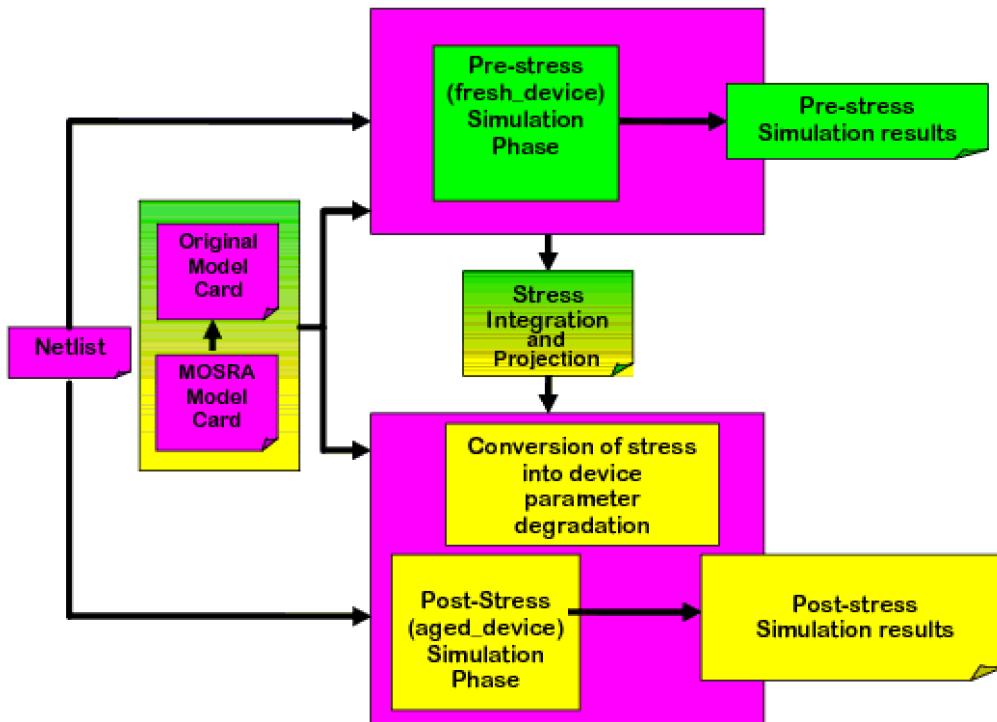
A MOSRA model is used for translating the amount of electrical stress to the actual device degradation, or “age”. Typically, such models are a function of device operating conditions (such as voltages, currents, or temperatures) as well as device geometries. The resulted degradation is converted in the degradation of key MOSFET compact model parameters (such as threshold voltage, mobility, and so forth). This approach allows for a separation of different effects that contribute to the total device degradation, resulting in better accuracy of current and conductance degradation over a wide range of device biases.

MOSRA models are constructed with physics-based formulations and augmented with coefficient parameters to improve the model accuracy and parameter extraction flexibility.

In addition to built-in MOSRA models, the simulators enable foundries and IDMs to develop their own HCI and BTI models then integrate them with the simulator through an API. Users can implement and compile their own special model equations through the API, and compile them into a Dynamically Linked Library (DLL).

PrimeWave Design Environment Integration of the MOSRA Flow

For all simulator integrations, the PrimeWave Design Environment-integrated MOSRA flow includes two phases: the *stress* (aka pre-stress) analysis phase and the *degradation* (aka post-stress) analysis phase, respectively, as shown below.



The following topics give an overview of the MOSRA flow:

- [Stress Simulation](#)
- [Degraded Simulation](#)
- [Combined Stress + Degraded Simulation](#)
- [Graphical User Interface](#)

Stress Simulation

During the *stress* (aka pre-stress) simulation phase, the simulator computes the electrical stress of MOSFETs in the circuit (all or a user-specified subset), based on MOSRA models. The calculation depends on the electrical simulation conditions of each targeted device. The stress value from the MOSRA equation is integrated over a user-specified simulation time interval, through the duration of the transient analysis.

The result of the integration is then extrapolated to calculate the total stress after a user-specified time of circuit operation (age).

Degraded Simulation

During the *degraded* (aka post-stress) phase, a second simulation run is performed. The degradation of device characteristics is therefore translated to performance

degradation at the circuit level. The post-stress MOSRA simulation phase can be based on either .DC, .AC, or .TRAN analysis.

The effect of accumulated stress illustrates the aging simulation capabilities due to a seamless integration of the aging models within the MOSRA flow. At each MOSRA circuit operation time step, the flow considers the accumulated degradation information from previous time steps. As a result, the accumulated stress effect is taken into account implicitly, with no need for empirical “equation bending.”

Combined Stress + Degraded Simulation

The PrimeWave Design Environment by default will combine both simulation phases into a single simulation run. During this run, a first phase stresses and observes the devices, and during the second phase the aging characteristics are extrapolated out to the target final circuit age. Sample points can also be created to obtain simulation results at intermediate ages.

Graphical User Interface

In the PrimeWave Design Environment MOSRA integration, a single streamlined interface is presented. This interface is designed such that you interact primarily with Graphical User Interface (GUI) elements only, for configuration of the simulation, allowing the various options to be specified, and so forth. You generally do not need to know about the presence of particular files, file-naming conventions, how to include them or point to them within a netlist, and so on. The PrimeWave Design Environment takes care of all these details.

In addition to the easy-to-use setup interface for the combined stress+degraded analysis flow, stress-only, and degraded-only analysis flows, the PrimeWave Design Environment also provides an intuitive stress analysis ResultsView tool. This allows device aging degradations (degradations to model parameters) to be viewed in an intuitive tabular format, with full support for column-based sorting, cross selection to schematic or netlist text source, and so forth. Details are presented in the following sections.

Prerequisites

In order to run this flow, the simulator needs to be appropriately configured in order to pick up the appropriate MOSRA models at run time. Model cards leveraging either the built-in MOSRA model equations can be used, or model cards leveraging custom models can be used, meaning a path to a shared library needs to be provided. When using custom models/shared libraries, it is assumed that the simulator is appropriately configured to find the appropriate shared library as part of the user’s environment.

MOSRA model cards are generally built up using the `.appendmodel` statement to append MOSRA equation (which can be built-in or custom) model parameters to an otherwise

"regular" MOS model. Consider a simple model file `simpleModels.l` with the following contents:

Example 2 *simpleModels.l* file contents

```
* File: simpleModels.l
.model p1 pmos level=54 version=4.5
.model n1 nmos level=54 version=4.5
```

Here, two simple level-54 models, p1 and n1, are defined. For the purposes of illustration, these are very simple models, using default values for all the level-54 model parameters. Generally, these models are used for regular simulations. For reliability analysis however, these models need to be augmented with extra parameters that configure the MOSRA model equations.

The listing of Example 2 shows a MOSRA model file, `mosraModels.l`, that is used to perform this augmentation.

Here, two level-1 MOSRA model cards are defined, the p1_ra model and the n1_ra model. In this example, the p1_ra and n1_ra models are defined to supply the parameter values/coefficients of the HSPICE built-in MOSRA model equations. They are then used to append the regular MOS models p1 and n1 respectively.

The model p1_ra is a level 1 MOSRA model, adding parameter values for level one MOSRA equation parameters tit0., titfd, tittd, tn respectively.

A `.appendmodel` statement is then used to append those model equations/coefficients to the 'regular' p1 MOS model that was listed in Example 1.

A similar process is repeated for augmenting the n1 MOS model with the n1_ra reliability equation model parameters.

Example 3 *mosraModels.l* model file which augments regular models for reliability analysis

```
* File: mosraModels.l

* create a simple MOSRA model 'p1_ra'
.model p1_ra mosra level=1
+tit0 = 5e-8 titfd = 7.5e-10 tittd = 1.45e-20
+tn = 0.25

* augment the p1 pmos model with the p1_ra mosra parameters
.appendmodel p1_ra mosra p1 pmos

* create a simple MOSRA model 'n1_ra'
.model n1_ra mosra level=1
+tit0 = 5e-8 titfd = 7.5e-10 tittd = 1.45e-20
+tn = 0.25

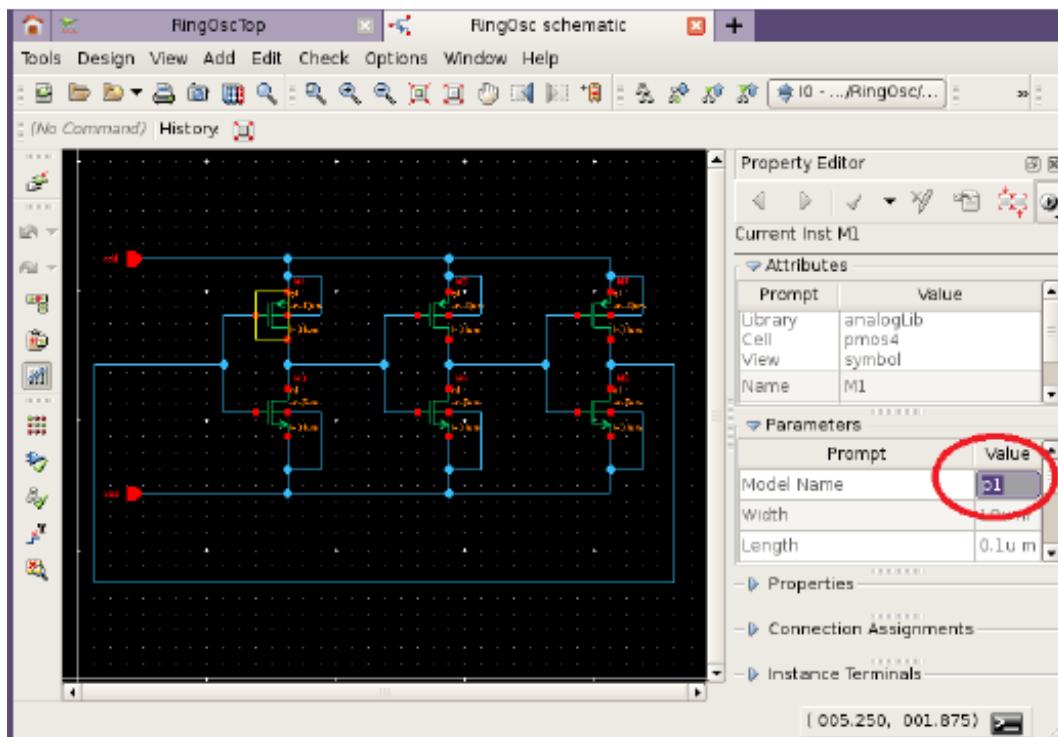
* augment the n1 nmos model with the n1_ra mosra parameters
.appendmodel n1_ra mosra n1 nmos
```

The PrimeWave Design Environment **Model Setup** dialog box can be configured to find augmented MOSRA models in the usual manner.

An appropriately recent simulator version is also required. Other than its usual provision for specifying model library paths and so forth to the simulator at runtime, the PrimeWave Design Environment does not provide any additional simulation-specific configuration; in other words, the PrimeWave Design Environment expects your shell environment to be already appropriately configured.

Generally, it is sufficient to use a recent simulator version, using the model setup GUI to point the simulator at the appropriate MOSRA models within a PDK or CAD group supplied location.

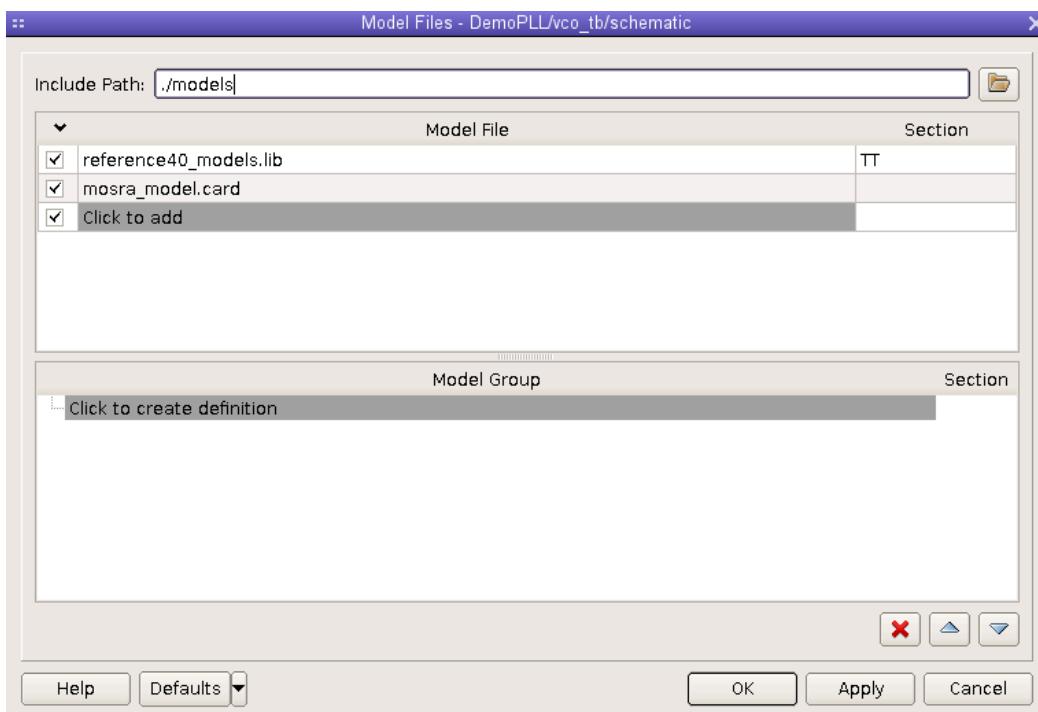
Ensure a schematic or text netlist design is available that contains transistors referencing those MOSRA model names. The figure below shows a sample design with PMOS devices using an augmented p1 MOSRA model.



Recommended Steps

The recommended steps in the PrimeWave Design Environment interface supporting the MOSRA flow are:

1. Run a normal analysis, and save the results to a PrimeWave Design Environment history point using the PrimeWave Design Environment's history mechanisms.
Typically, a designer will have performed this step before proceeding to a reliability simulation.
2. Ensure the simulator is appropriately configured.
3. Open a schematic or text netlist design with appropriately configured transistors for which MOSRA model cards exist.
4. Ensure that simulation model paths are setup to find the MOSRA models, as shown below.



5. Use the PrimeWave Design Environment **Tools > Aging Setup** menu choice and configure the combined Stress+Degraded simulation, or simply just the first-stage Stress simulation. Details are provided in the following examples.
6. Ensure that a transient analysis simulation is enabled (the MOSRA stress flow does not operate for other analysis types).

7. Run the combined Stress+Degraded simulation or first stage Stress simulation using the usual PrimeWave Design Environment netlist and run controls. Details are provided in the following examples. If performing a combined Stress+Degraded simulation, the effects of the first-stage aging analyses are included, and their impact on circuit waveforms, measurements, and so forth, produced by the aged/degraded circuit (as determined by the second simulation phase) can be inspected. For a Stress-only simulation, the simulation run produces a new PrimeWave Design Environment simulation history containing a database of device model parameter degradation rates, depending on the above configuration.
8. Optionally, use the PrimeWave Design Environment **Results > MOSRA** menu choice to inspect and sort the degradation rate report, and to cross-select devices from the results table to the circuit schematic or text netlist view.
9. If a Stress-only configuration was used in [Step 7](#), then optionally configure a second-stage Degraded-only simulation. Details are provided in the following examples. The main step to be performed during this configuration step is to point at the results history or results database file produced by the previous Stress simulation run. In most cases, this happens automatically.
10. Run the second-stage Degraded simulation using the usual PrimeWave Design Environment netlist and run controls.

Note:

For netlist-based PrimeWave Design Environment and MTB users, be sure to run this simulation in the same history point as used for the Stress Analysis of [Step 7](#).

At this point, the effects of the first-stage aging analyses are included, and their impact on circuit waveforms, measurements, and so forth, produced by the aged/degraded circuit can be inspected and compared against the stress simulation that was performed with "fresh" (unaged) devices.

PrimeWave Design Environment MOSRA Flow Example

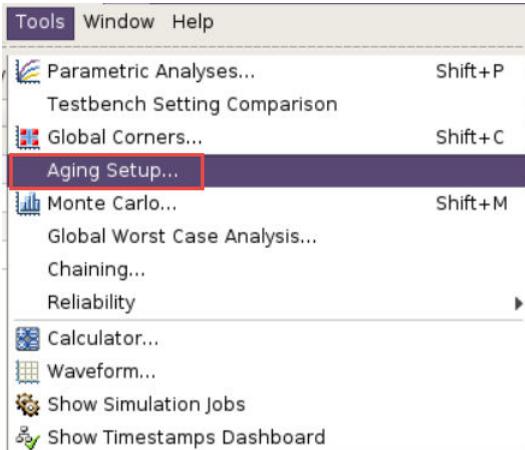
This example demonstrates the following:

- [Configuring a Combined Stress + Degraded Simulation](#)
- [Running a Combined Stress + Degraded Simulation](#)
- [Configuring a Stress-Only Simulation](#)
- [Running a Stress-Only Simulation](#)
- [Inspecting and Cross-Selecting Results to Source Design](#)

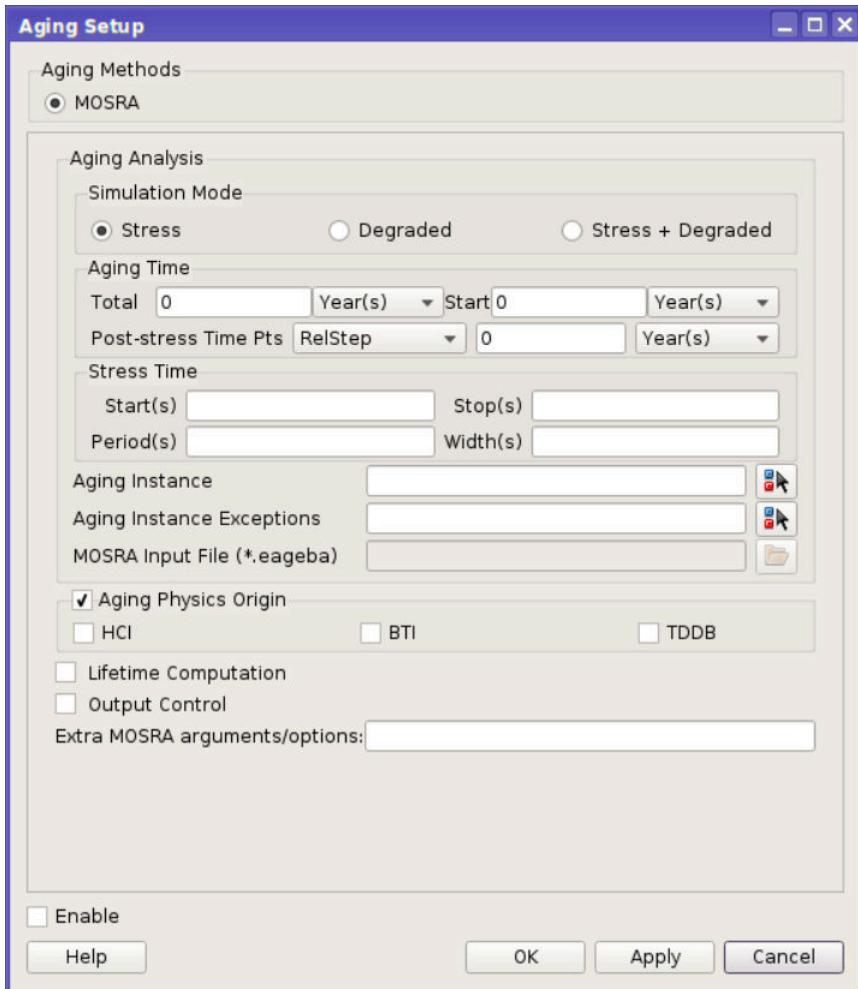
- Stress + Degraded Simulation: Simulation Waveforms and Measurements
- Configuring the ‘Degraded’ Simulation
- Running the Degraded Simulation
- Observing Device Behaviors at Multiple Aging Points
- Looking at the PrimeWave Design Environment Results Table
- Saving and Loading State

Configuring a Combined Stress + Degraded Simulation

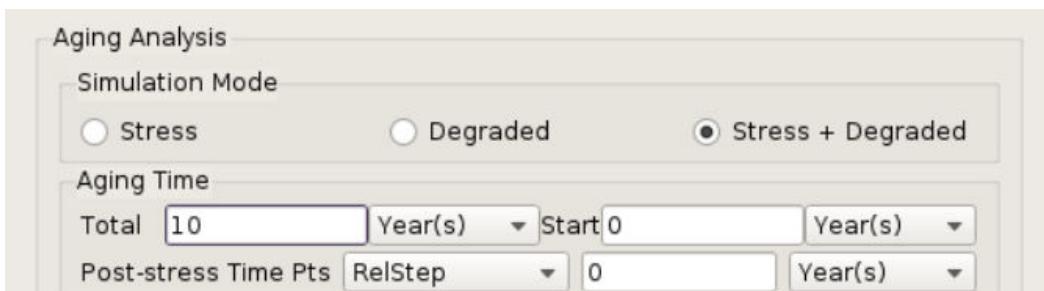
To enable and configure the combined MOSRA flow in the PrimeWave Design Environment, first select **Tools > Aging Setup**.



In response to choosing this menu entry, a dialog box opens in which the MOSRA flow can be configured prior to running actual simulations.



To enable the combined flow, choose **Stress+Degraded** in the **Simulation Mode** menu as shown in the figure below.

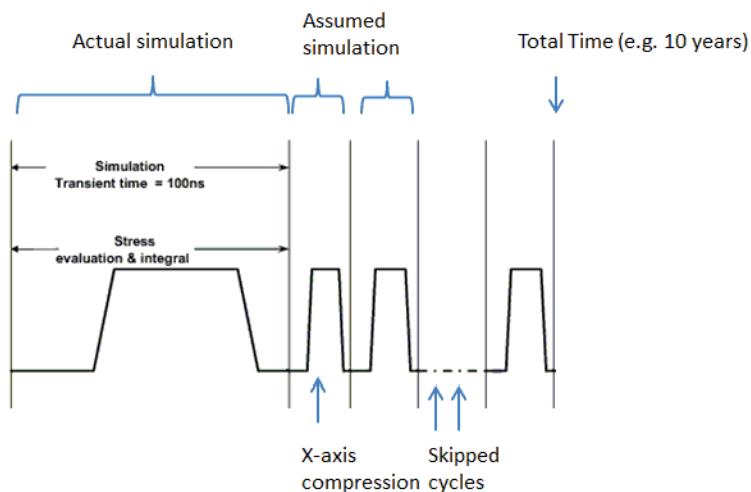


By default, during a Stress-only or combined Stress+Degradation simulation, the devices are observed for aging degradation effects during the entire transient simulation (for example, up to 100ns). The degradation equations and effects are then extrapolated out

to the requested **Aging Time Total** value (for example, 10 years, as specified in the setup dialog box of the figure above), for which time the resulting degradations to device model parameters are finally computed.

A representative (not actual) timing graph is shown in [Figure 4](#). Here, the y-axis represents some arbitrary circuit waveform, such as the voltage across a MOS device of interest. Notice that, by default, the entire transient simulation time (100ns) is used to compute the degradation equations and integrals. Waveforms are generated for the actual simulation transient time assuming the devices are actually still fresh (unaged).

Figure 4 Default Timing for Stress Analysis

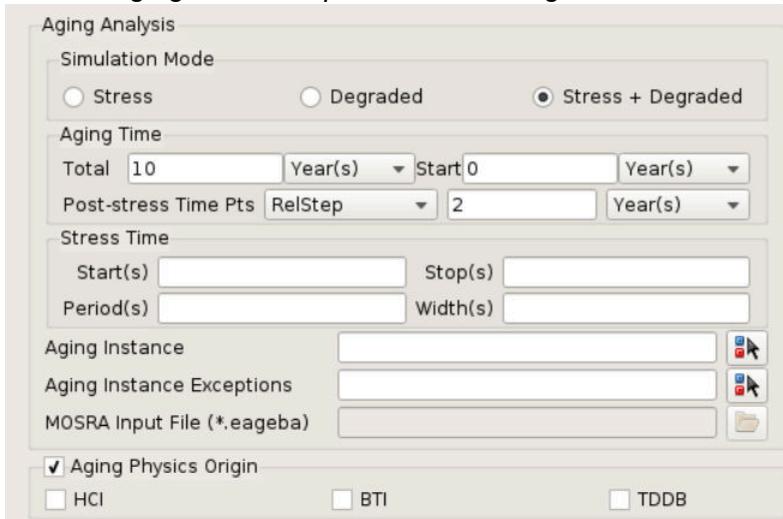


After this transient simulation interval has finished, the simulator then "extrapolates" out the observed and integrated degradation equations results to the requested **Aging Time Total** (for example, 10 years). No actual transient simulation is performed beyond the initial transient simulation time. The degradation effects are calculated as if that first 100ns of transient behavior was repeated over and over for the entire **Aging Time Total**, which is what is represented in the narrow (compressed) vertical stripes on the graph. The device compact model parameter degradation values that correspond to the end of the **Aging Time Total** (the very right of the graph), are computed and written to the degradation output file, completing the Stress analysis, or first phase of the combined Stress+Degraded analysis.

Notice that the x-axis scale in [Figure 4](#) is nonlinear. Each compressed vertical strip represents another 100ns of circuit operation time assumed to be identical in behavior to the first 100ns that was actually simulated. Also notice that the x-axis is split (3/4 of the way to the right) to skip over a large number of these repeated intervals that are assumed to be identical to the first 100ns that was actually simulated. The **Aging Time Total** at the end of the graph is the time for which the device model parameter degradations are finally computed, and output to the degradation output file.

In terms of the **Aging Options**, the **Aging Effects Physics** can include HCI, BTI, or TDDB. You can choose **Aging Instance** devices by clicking  and picking from the schematic design. If set to anything other than blank or "*", only the specified MOS devices are considered in the analysis.

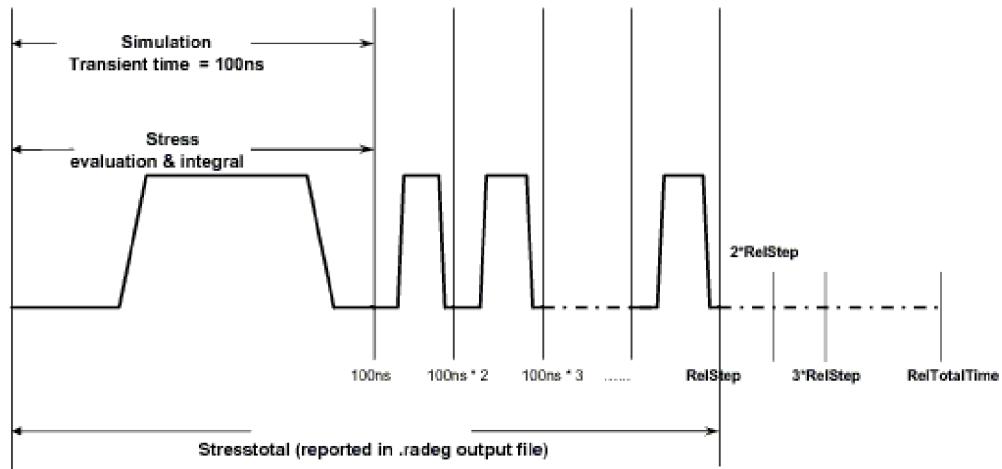
Figure 5 Aging Flow Setup for Stress + Degraded Simulation, Showing Advanced Options



The setup of the dialog box above also allows somewhat different timing behavior to be specified compared to that illustrated by the timing graph of [Figure 4](#).

For example, for the **Post-stress Time Pts** option, the stepping mode can be set to **RelStep** and a step size value can be specified along with its units. (This GUI option maps to the `RelStep` parameter on the `.MOSRA` statement in the PrimeSim HSPICE deck). This causes a sampling of the device degradations to be computed at integral multiples of the step size, up until the **Aging Time Total**, as shown in [Figure 6](#). In the example of [Figure 5](#), the **RelStep** value is specified as 2 years.

Figure 6 Stress Analysis Timing Using Step Size Option



Therefore, the stress analysis phase will compute model parameter degradations at 2yr, 4yr, 6yr, 8yr, and 10yr intervals. (These interval points are mapped in the timing graph of [Figure 6](#) as RelStep, 2*RelStep, 3*RelStep, and so forth.) Note that the x-axis scale toward the right of the graph is greatly different (compressed) than that in the left side of the graph to reflect those 2yr intervals.

When computing the interval points, the values chosen are actually at integer multiples of the **RelStep** plus an offset, which is configured by the **Stress Time Start** field in the **Aging Setup** dialog box. By default, the offset/**Stress Time Start** is zero. Interval points are generated until an interval point that exceeds the **Aging Time Total** is generated.

Instead of manually specifying the step size as an absolute number, it is also possible to have those step size values automatically generated. This can be done by specifying **Post-stress Time Pts to DEC** or **LIN** options.

In a similar manner, the options within the **Aging Options** section are used to govern the stress analysis observation and integration process itself. In contrast to the timing options, these fields control the analysis prior to any extrapolation. These fields control what the simulator does during the actual transient analysis interval, and a little bit beyond. An example is shown in [Figure 7](#), in which the **Aging Period** is set to 1s, the **Aging Start** time is set to 10ns, and the **Aging Stop** time is set to 100ns. The corresponding timing graph is shown in [Figure 8](#).

Figure 7 Setup Dialog Box Specifying Aging Period, Aging Start, and Aging Stop Times

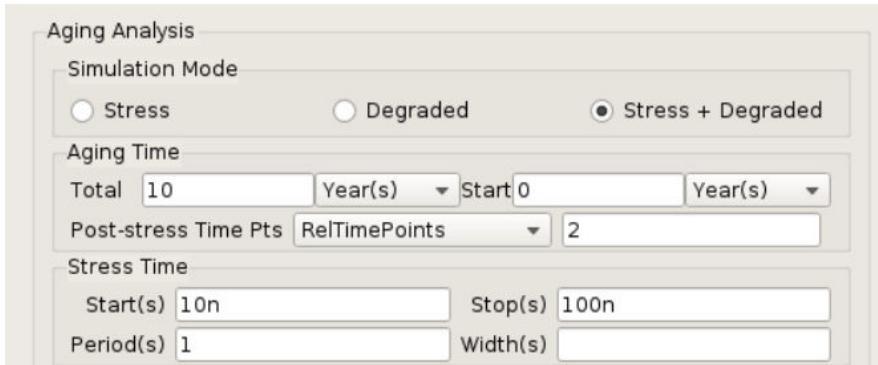
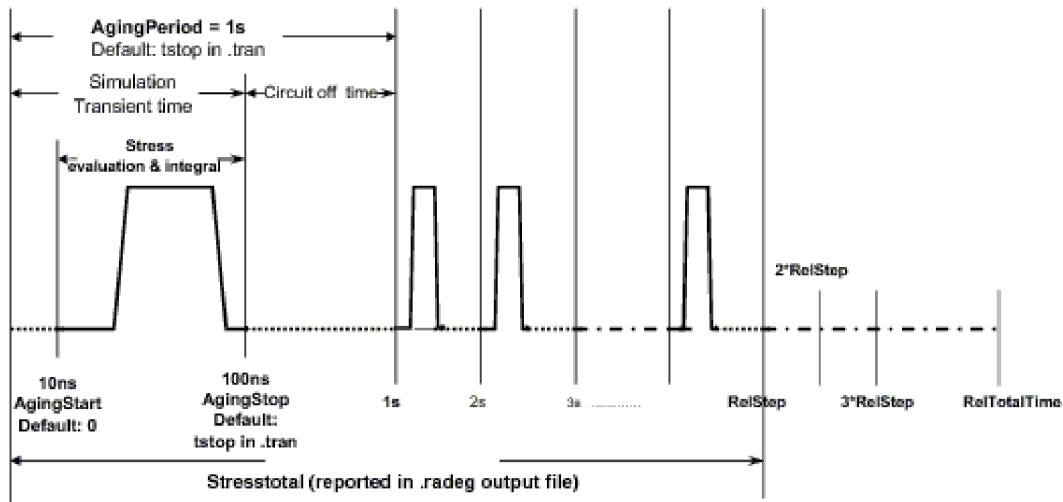


Figure 8 shows that the stress evaluation and equation integration process does not actually start until the **Stress Time Start** field value of 10ns, and it stops at either the transient stop time, or the value of the **Stop** field, whichever is sooner (in this example, both values are the same, at 100ns). Prior to computing the device degradations at the usual post-stress intervals or total time, the simulator assumes that the circuit is 'off' (inactive) between the aging stop time (or simulation transient analysis interval, whichever is smaller) and the specified aging **Period** value. In other words, the duty cycle or circuit on-to-off ratio calculated during the aging **Period** affects the degradation computations.

Figure 8 Stress Evaluation with Aging Start, Stop, and Period Parameters Specified



Instead of specifying the **Start/Stop** window, a **Width** value can be specified instead, as shown in the dialog box of [Figure 10](#). The timing graph corresponding to the dialog box setup of [Figure 10](#) is as shown in [Figure 9](#).

Figure 9 Timing Graph Based on Aging Width and Aging Period Parameters

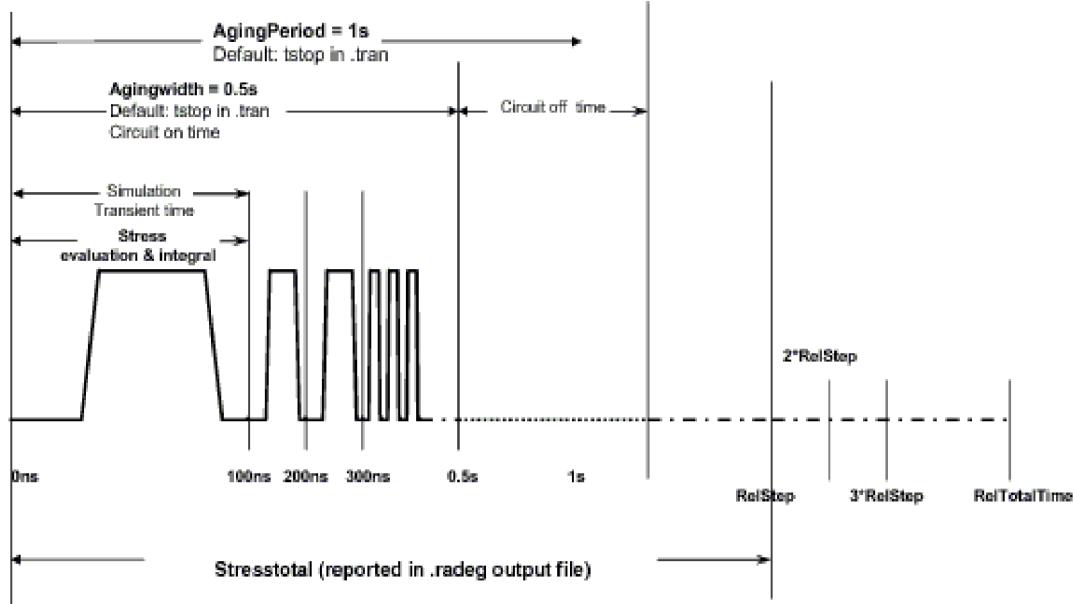
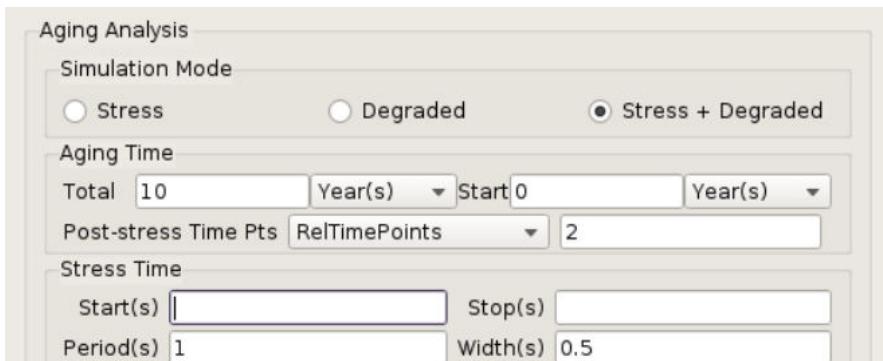


Figure 10 Specifying Aging Period and Aging Width



In this case, the transient simulation interval is considered to be repeating itself until the aging **Width** time value of 0.5s, at which point the circuit is then considered to be off or inactive for the remaining 0.5s until the end of the **Period** of 1s. The resulting degradation coefficients are then calculated at the 2 year post-stress intervals as before.

Finally, the **Aging Setup** dialog box allows for additional PrimeSim HSPICE .MOSRA statement options to be specified via the **Extra .MOSRA arguments** field toward the bottom of the dialog box. Any statements entered here are automatically appended to the .MOSRA statement generated by the contents of the setup dialog box, without any interpretation from the PrimeWave Design Environment.

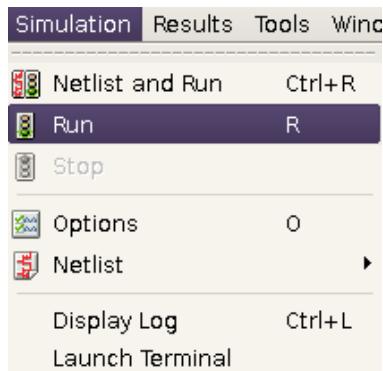
Click **OK** or **Apply** to apply the changes in this dialog box before proceeding to run a combined Stress+Degraded simulation.

Running a Combined Stress + Degraded Simulation

After clicking **OK** or **Apply** in the setup dialog box with the simulation field set to **Stress + Degraded** (as shown in [Figure 10](#)), the next step is to run the actual simulation.

The combined stress+degraded simulation is run using the usual PrimeWave Design Environment **Run** control, as shown in [Figure 11](#). (A new netlist is created with the appropriate `.MOSRA` statement inserted. The simulator then consumes this netlist.)

Figure 11 Running the Stressed+Degraded Simulation



Once the simulation completes, the device degradation results, in terms of their computed MOS Model parameter degradations, can also now be inspected, and any devices of interest can be cross-highlighted to the schematic.

At this point, it is common to save a history point explicitly, and to compare the results of the combined stress+degraded simulation (measurement values, and so forth) with those of a "regular" simulation performed prior to invoking the MOSRA flow.

Configuring a Stress-Only Simulation

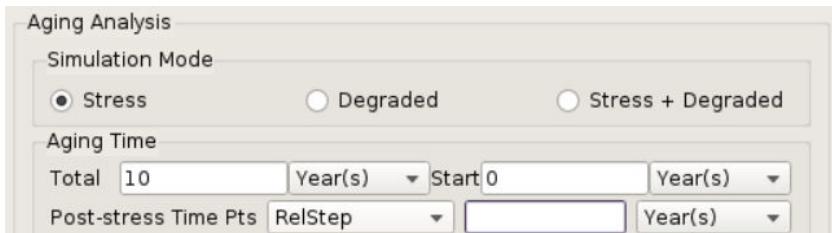
In addition to the combined Stress+Degraded flow, the PrimeWave Design Environment provides alternate flows that allow the stress and degraded analyses to be split up into two separate simulation runs. Typically, this is done only:

- to use a different circuit stimulus for the Stress analysis (that is, a stimulus that is representative of the circuit's intended operating environment when deployed in the field, which can be substantially different than the stimulus used to characterize it during design), or

- to ensure that a transient analysis is used when performing the stress analysis, but choosing to use different analysis during the degraded analysis phase to characterize the device; that is, to characterize its AC or DC circuit behavior, its noise behavior, and so on.

First set up the desired stimulus and be sure a transient analysis is selected. Next, to enable the first-stage Stress-only flow, and configure the first simulation (the Stress simulation) in order to compute degradation rate information, choose **Stress** from the list of simulation choices.

Figure 12 Aging Flow Setup for Stress (First) Simulation



When performing separate Stress and Degraded simulations (which for netlist-based PrimeWave Design Environment and MTB users must be performed in the same history point), the PrimeWave Design Environment needs to take care to copy the Stress analysis simulation results directory to a backup location, so that its contents (and in particular, the `.radeg/.eageba` files, information containing the extrapolated model parameter degradations) are available to the simulator when performing the subsequent Degraded simulation.

Running a Stress-Only Simulation

After clicking **OK** or **Apply** in the setup dialog box with the simulation field set to **Stress** as shown in [Figure 12](#), the next step is to run the actual simulation. The stress simulation is run using the usual PrimeWave Design Environment **Run** control as shown in [Figure 13](#). (A new netlist is created with the appropriate `.MOSRA` statement inserted. The simulator then consumes this netlist.)

Figure 13 Running the Stressed Simulation

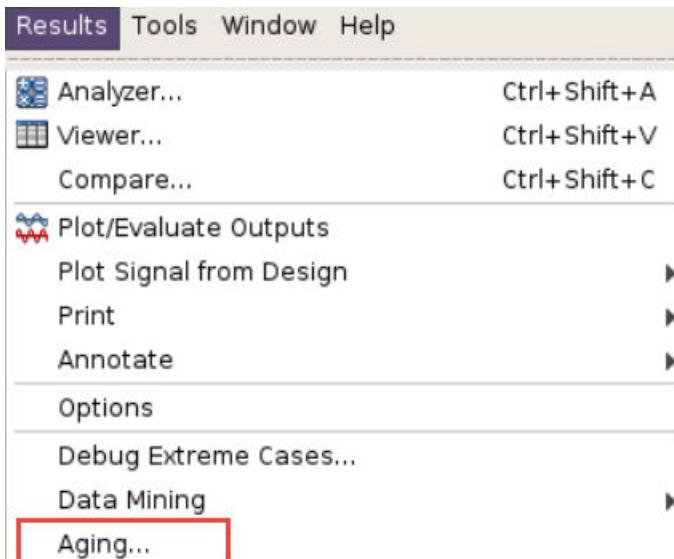


The extrapolated per-MOS-device degradation results, in terms of their computed MOS Model parameter degradations, can also be inspected now, and any devices of interest can be cross-highlighted to the schematic.

Inspecting and Cross-Selecting Results to Source Design

After a combined Stress+Degraded or Stress-Only simulation has been performed and the parameter degradations computed, a new table of results is available, a viewer for which can be invoked from the **Results > Aging** menu, shown in [Figure 14](#).

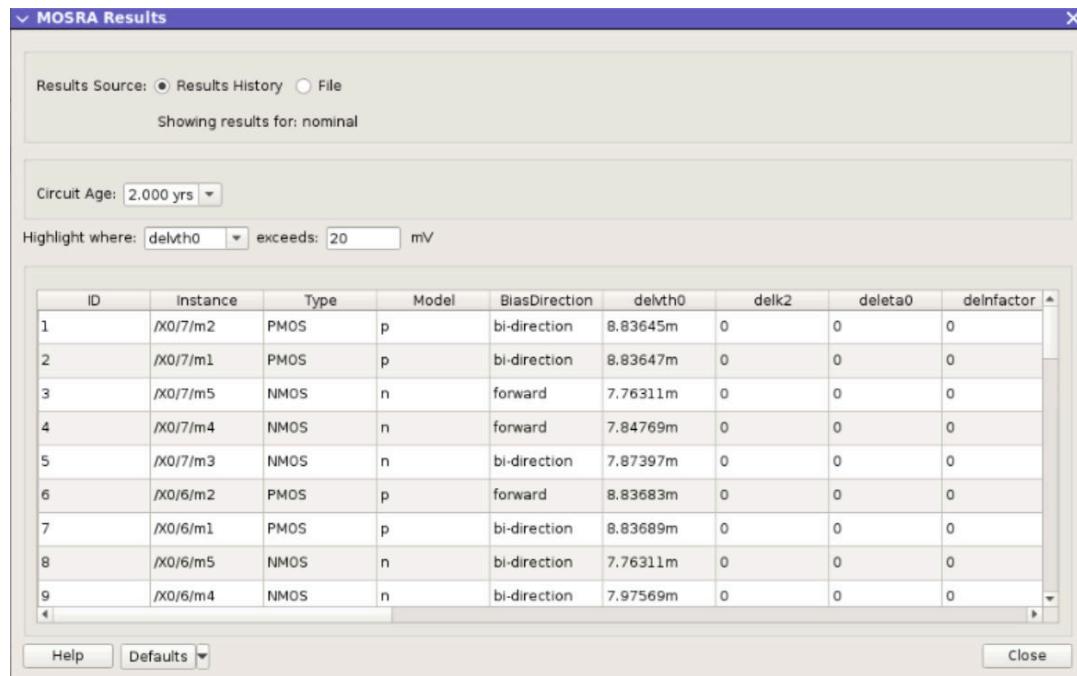
Figure 14 Accessing the MOSRA Degradation Values/Tabular Results



When selected, this accesses the tabular results from the most recently performed simulation run in the current history point.

The **MOSRA Results** dialog box opens, as shown in [Figure 15](#).

Figure 15 Results Table for MOSRA Flow Showing Parameter Degradation Values



The screenshot shows the 'MOSRA Results' dialog box. At the top, there are two radio buttons: 'Results History' (selected) and 'File'. Below that, it says 'Showing results for: nominal'. Under 'Circuit Age:', there is a dropdown menu set to '2.000 yrs'. A 'Highlight where:' dropdown is set to 'delvth0' and a 'mV' input field is set to '20'. The main area is a table with the following data:

ID	Instance	Type	Model	BiasDirection	delvth0	delk2	deleta0	defnfactor
1	/X0/7/m2	PMOS	p	bi-direction	8.83645m	0	0	0
2	/X0/7/m1	PMOS	p	bi-direction	8.83647m	0	0	0
3	/X0/7/m5	NMOS	n	forward	7.76311m	0	0	0
4	/X0/7/m4	NMOS	n	forward	7.84769m	0	0	0
5	/X0/7/m3	NMOS	n	bi-direction	7.87397m	0	0	0
6	/X0/6/m2	PMOS	p	forward	8.83683m	0	0	0
7	/X0/6/m1	PMOS	p	bi-direction	8.83689m	0	0	0
8	/X0/6/m5	NMOS	n	bi-direction	7.76311m	0	0	0
9	/X0/6/m4	NMOS	n	bi-direction	7.97569m	0	0	0
4								

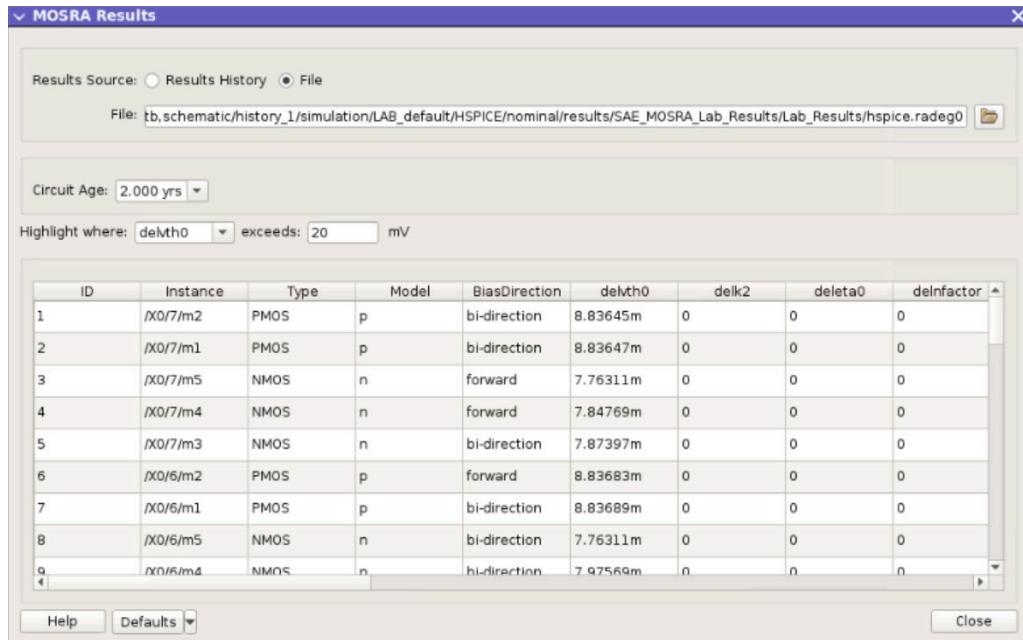
At the bottom left are 'Help' and 'Defaults' buttons, and at the bottom right is a 'Close' button.

You can choose between automatic selection of results:

- **Results History** for the current/active history drop-down (or most recently performed Stress-only analysis) as shown in [Figure 15](#), which is pre-filled and pre-selected by default, or
- **File** to explicitly specify a path to a degradation <netlist_file_root>.radeg0 output file, as shown in [Figure 16](#).

The **Results Source** mode switch is provided to choose between these two results selection modes.

Figure 16 Directly Specifying a .radeg0 File to Results Browser



Below the **Results Source** selection fields is the **Circuit Age** menu. This contains either:

- A single time value corresponding to the **Total Time** of [Figure 4](#), or
- A collection of time values corresponding to each of the post-stress intervals of [Figure 6](#), [Figure 8](#), [Figure 9](#), and so on. Such a list will be presented when, for example, the **Stepping Mode** group box of [Figure 5](#) contains a value that results in multiple stress intervals.

The results table contains a number of columns, the exact number depending on the MOSRA model equations used:

- ID-- An arbitrary numerical identifier.
- Instance-- The hierarchical instance name of a device. These are listed in the Custom Compiler namespace (not the simulator namespace).
- Type-- (NMOS or PMOS)
- Model-- (The name of the augmented SPICE Model)
- Bias Direction
- A number of columns (see [Figure 17](#)), one for each parameter for which a degradation value is computed by the degradation model. The specific list of parameters depends

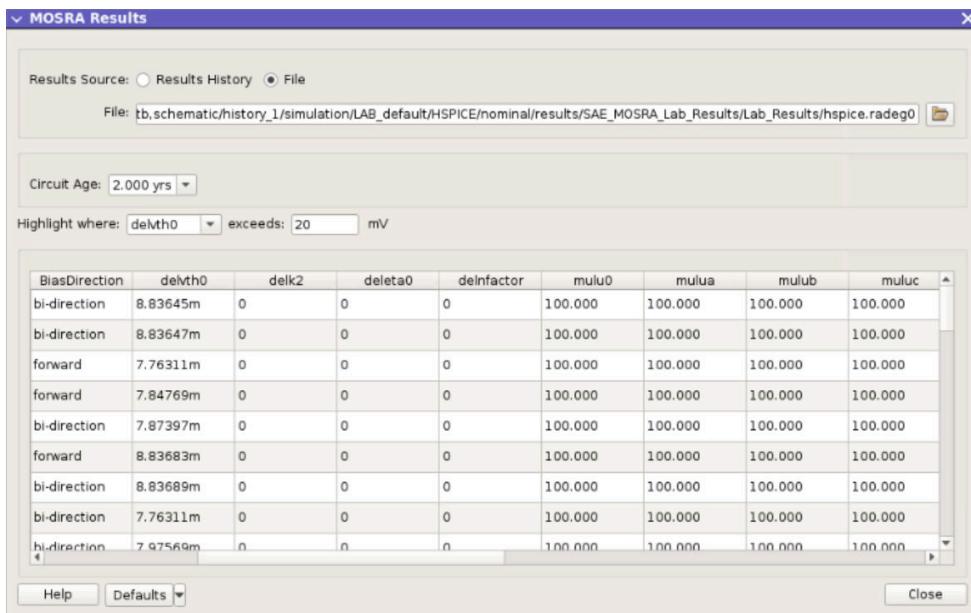
on the MOSRA model used. Examples might include (for the Level 1 HSPICE MOSRA model):

- delvth0
- delk2
- deleta0
- delnfactor
- mulmu0
- mulmua
- mulmub
- mulmuc
- ...

Typically, columns beginning with ‘del’ represent deltas or absolute differences between pre- and post-stress model parameter values for quantities such as vth0 (threshold voltage), and so forth. Columns beginning with ‘mul’ represent scale factors or multipliers (in units of %) for the corresponding parameters, such as the mobility parameters mu0, mua, and so on, listed above. Again, the actual values that show up here depend on the specific MOSRA model cards used.

- Device Instance parameters such as W, L, NF/NFINGERS, and so forth. The specific parameter names will depend on the device model used.

Figure 17 Results Table Showing Numerous Parameter Degradations



The screenshot shows a software window titled "MOSRA Results". At the top, there are two radio buttons: "Results History" (unchecked) and "File" (checked). Below that is a "File" field containing the path: "tb:schematic/history_1/simulation/LAB_default/HSPICE/nominal/results/SAE_MOSRA_Lab_Results/Lab_Results/hspice.radeg0". Underneath the file path is a "Circuit Age" dropdown set to "2.000 yrs". Below that is a "Highlight where:" dropdown set to "delvth0" and an "exceeds:" input field set to "20 mV". The main area is a table with the following columns: BiasDirection, delvth0, delk2, deleta0, delnfactor, mulu0, mulua, mulub, and muluc. The rows show various device configurations like bi-direction, forward, and bi-direction again, with values ranging from 7.76311m to 8.83647m for delvth0. All values in the table are currently highlighted in red.

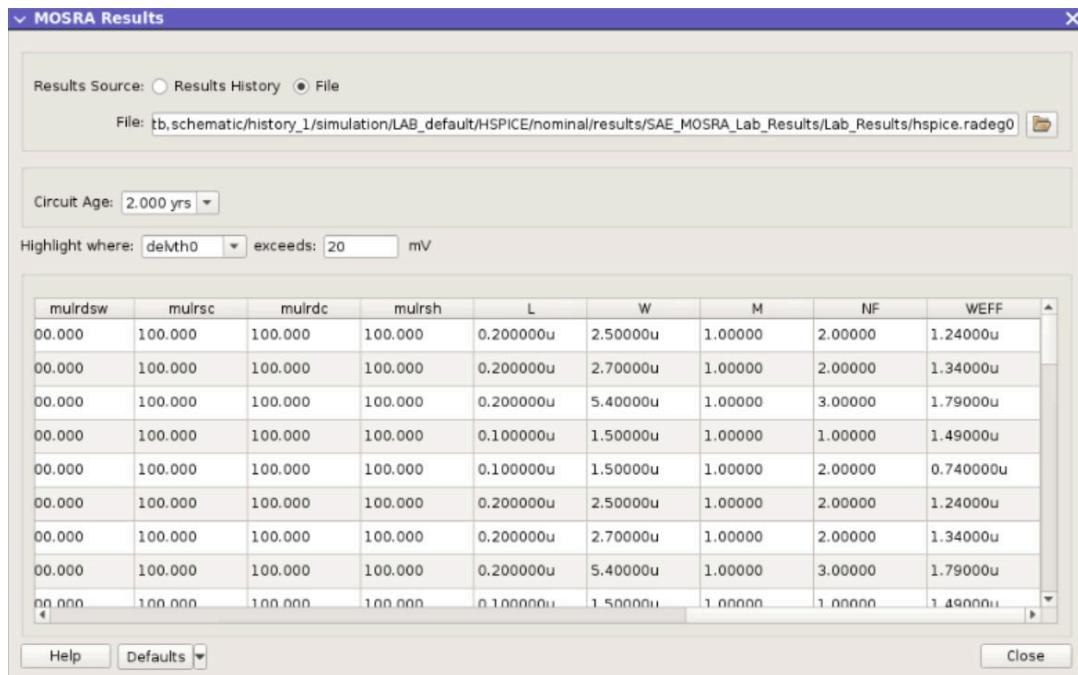
BiasDirection	delvth0	delk2	deleta0	delnfactor	mulu0	mulua	mulub	muluc
bi-direction	8.83645m	0	0	0	100.000	100.000	100.000	100.000
bi-direction	8.83647m	0	0	0	100.000	100.000	100.000	100.000
forward	7.76311m	0	0	0	100.000	100.000	100.000	100.000
forward	7.84769m	0	0	0	100.000	100.000	100.000	100.000
bi-direction	7.87397m	0	0	0	100.000	100.000	100.000	100.000
forward	8.83683m	0	0	0	100.000	100.000	100.000	100.000
bi-direction	8.83689m	0	0	0	100.000	100.000	100.000	100.000
bi-direction	7.76311m	0	0	0	100.000	100.000	100.000	100.000
bi-direction	7.97569m	0	0	0	100.000	100.000	100.000	100.000

When any row is double-clicked, the corresponding device will be cross-selected in the circuit schematic (Custom Compiler flow) or text netlist (netlist-driven PrimeWave Design Environment flow), as appropriate.

The table can be sorted in ascending or descending order by clicking on any column header. After sorting, the original sort order can be re-established by sorting on the ID column.

A user-specified highlight threshold can also be entered, such that table cells where the parameter degradation value meets or exceeds the specified value are highlighted in red. Use the **Highlight where** field to choose a parameter/column, and the **exceeds** field to enter a threshold value for that parameter/column.

Figure 18 Results Table Scrolled to Show Device Instance Parameters



The screenshot shows a software interface titled "MOSRA Results". At the top, there are buttons for "Results History" and "File", with "File" being selected. Below that is a file path: "File: tb,schematic/history_1/simulation/LAB_default/HSPICE/nominal/results/SAE_MOSRA_Lab_Results/Lab_Results/hspice.rade0". There is also a folder icon. Underneath is a "Circuit Age" field set to "2.000 yrs". A "Highlight where" dropdown menu is open, showing "delvth0" and "exceeds: 20 mV". The main area is a table with the following columns: mulrds, mulrsc, mulrdc, mulrsh, L, W, M, NF, and WEFF. The table contains 10 rows of data, all of which are identical, showing values like 0.000, 100.000, etc. At the bottom of the table are scroll bars. Along the bottom edge of the window are buttons for "Help", "Defaults", and "Close".

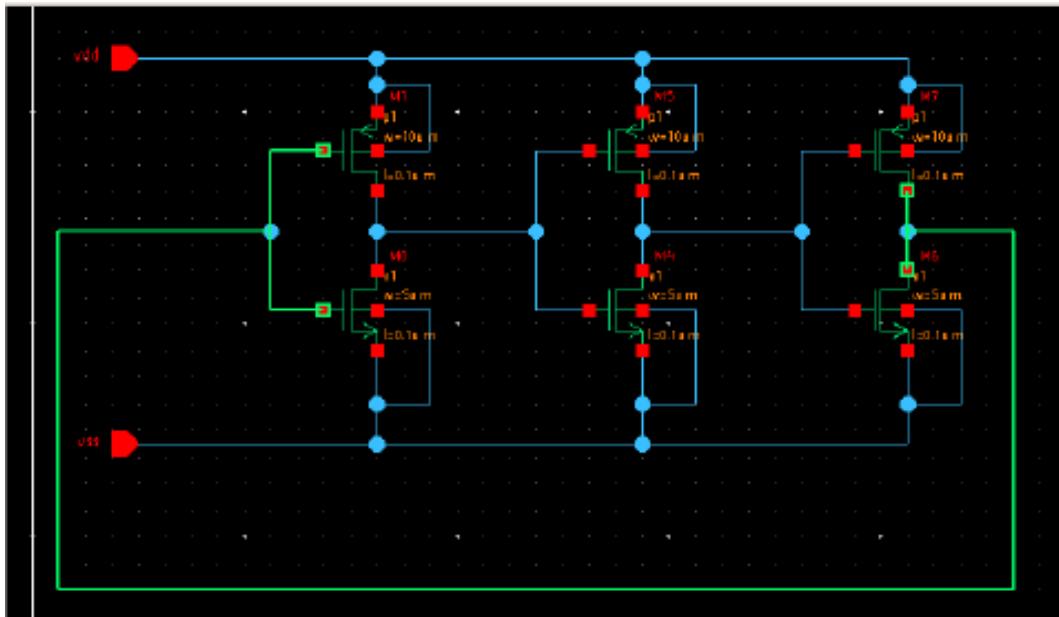
mulrds	mulrsc	mulrdc	mulrsh	L	W	M	NF	WEFF
00.000	100.000	100.000	100.000	0.200000u	2.50000u	1.00000	2.00000	1.24000u
00.000	100.000	100.000	100.000	0.200000u	2.70000u	1.00000	2.00000	1.34000u
00.000	100.000	100.000	100.000	0.200000u	5.40000u	1.00000	3.00000	1.79000u
00.000	100.000	100.000	100.000	0.100000u	1.50000u	1.00000	1.00000	1.49000u
00.000	100.000	100.000	100.000	0.100000u	1.50000u	1.00000	2.00000	0.740000u
00.000	100.000	100.000	100.000	0.200000u	2.50000u	1.00000	2.00000	1.24000u
00.000	100.000	100.000	100.000	0.200000u	2.70000u	1.00000	2.00000	1.34000u
00.000	100.000	100.000	100.000	0.200000u	5.40000u	1.00000	3.00000	1.79000u
00.000	100.000	100.000	100.000	0.100000u	1.50000u	1.00000	1.00000	1.49000u

Stress + Degraded Simulation: Simulation Waveforms and Measurements

After running the Stress+Degraded simulation in the combined flow, the effects of aging and degradation on the circuit waveforms and measurements can be immediately inspected. You can compare them with those of a simulation performed with no MOSRA analysis to compare *fresh* versus *degraded*. If a Stress-Only analysis was performed (perhaps with a stimulus that represents the typical duty cycles of the transistors during their intended operating environment), any waveforms and measurements will represent those of a *fresh* or unaged circuit.

By way of example, consider a circuit in which a ring oscillator block is instantiated. The schematic topology is shown in Figure 19, and the simple degradation models of Example 2 and Example 3 are included.

Figure 19 Simple Ring Oscillator Circuit



After running the combined Stress+Degradation simulation for the simple ring oscillator of Figure 19, you can plot the waveform for a circuit node to see it oscillating. You can also include the plot of a derived waveform representing a moving average calculation for the frequency versus time for the same net, which shows an approximate oscillation frequency of 10.88GHz. The results are shown in Figure 21, and the actual PrimeWave Design Environment measurement expression to determine the average oscillation frequency is

```
average(fvst(v(/12/net17), 0.3, 10n, percent=0))
```

Figure 20 Combined Stress + Degradation Simulation Results for a Ring Oscillator Circuit



Because a combined analysis was performed, both Fresh Device and Degraded Device waveforms are shown in [Figure 20](#). The first (green) chart shows the Fresh device waveforms for a circuit node of interest, and the second (yellow) chart shows the corresponding waveform for the aged device. The next two charts (cyan and red, respectively) show the Fresh and Aged circuit's average oscillation frequency. The fresh device oscillates at somewhere close to 10.88GHz, while the aged device oscillates at 10.65GHz. When running a Stress-only simulation as configured in [Figure 12](#), only Fresh device waveforms and measurements will be shown, as seen in [Figure 21](#).

Figure 21 Stress-Only (Fresh) Simulation Results for a Ring Oscillator Circuit

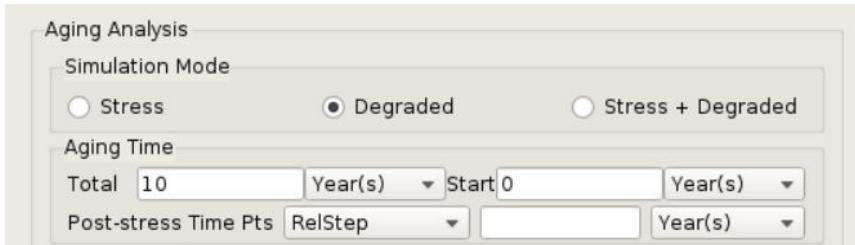


Configuring the ‘Degraded’ Simulation

After performing a Stress-only first stage simulation, the PrimeWave Design Environment allows the second stage Degraded simulation to be performed as a separate explicit step.

To configure the second stage Degraded simulation, set the simulation choice to Degraded as shown at the top of the setup dialog box shown in [Figure 22](#). In the most common use model, the Degraded simulation is to be run using the previously computed device aging rates produced by the first stage ‘Stress’ simulation. In this case, the **Stress Results Source** field should be set to **Results History** (again, see [Figure 22](#)).

Figure 22 Results History-Based Setup for Second-Stage Degraded Simulation



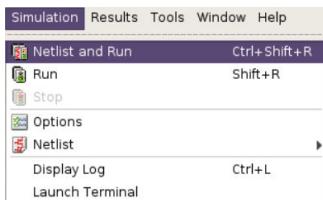
In an alternate use model, it is possible to explicitly specify a path to a previously output `<netlist_file_root>.eageba` file.

Remember to click **OK** or **Apply** before running the Degraded simulation.

Running the Degraded Simulation

The next step is to run a simulation with the simulation field of the setup dialog box set to Degraded as shown at the top of [Figure 22](#).

Figure 23 Running the Degraded Simulation



The Degraded simulation is run using the usual PrimeWave Design Environment **Run** control as shown in [Figure 23](#). (Again, a new netlist file will be created with the appropriate `.MOSRA` statement inserted).

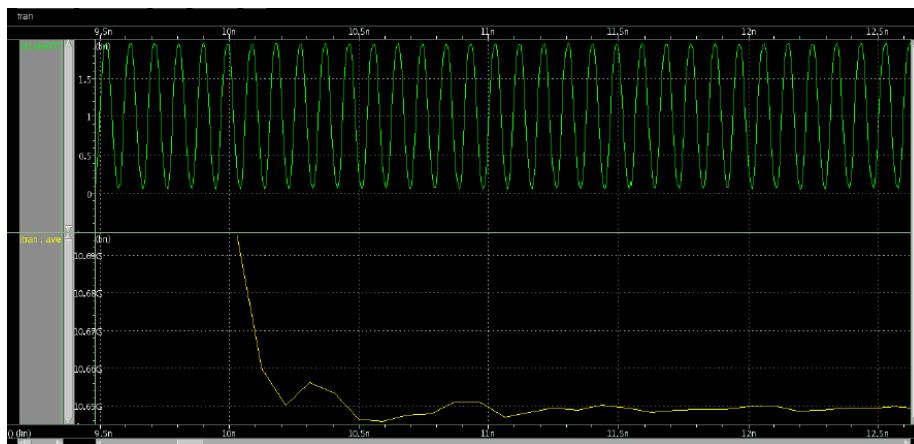
Caution:

The PrimeWave Design Environment tool searches through your simulation history to find the most recent `.radeg` file (or set of files in the case of a simulation that involves sweeps, corners and/or Monte Carlo) that matches the current simulation (which again might have sweeps, corners, or Monte Carlo). The terms of the second (degraded) set of simulations (in terms of PVT samples, i.e. sweeps/corners/MC setup) needs to match those of the first (stress) set. For example, if you performed the stress simulations using sweeps of two variables over three values each, and then tried to run the degraded simulations using sweeps of one or three variables, or sweeps of two variables over five values each, then the degraded simulation will not succeed.

After this simulation, output waveforms, measurements, and so forth will be available with aging effects taken into account. Given that the configuration of [Figure 23](#) was set up to simulate the waveforms and measurements for 10-year-old devices, you can expect to see some degradations in waveforms.

With the PrimeWave Design Environment **Plotting Mode** set to **append** in order to overlay results from the previous Stress simulation, you can see a significant change in the average frequency versus time curve plotted in [Figure 24](#). When compared with the fresh devices, which oscillated at a frequency of 10.88 GHz, notice that the 10-year-old devices oscillate at the reduced frequency of 10.65 GHz.

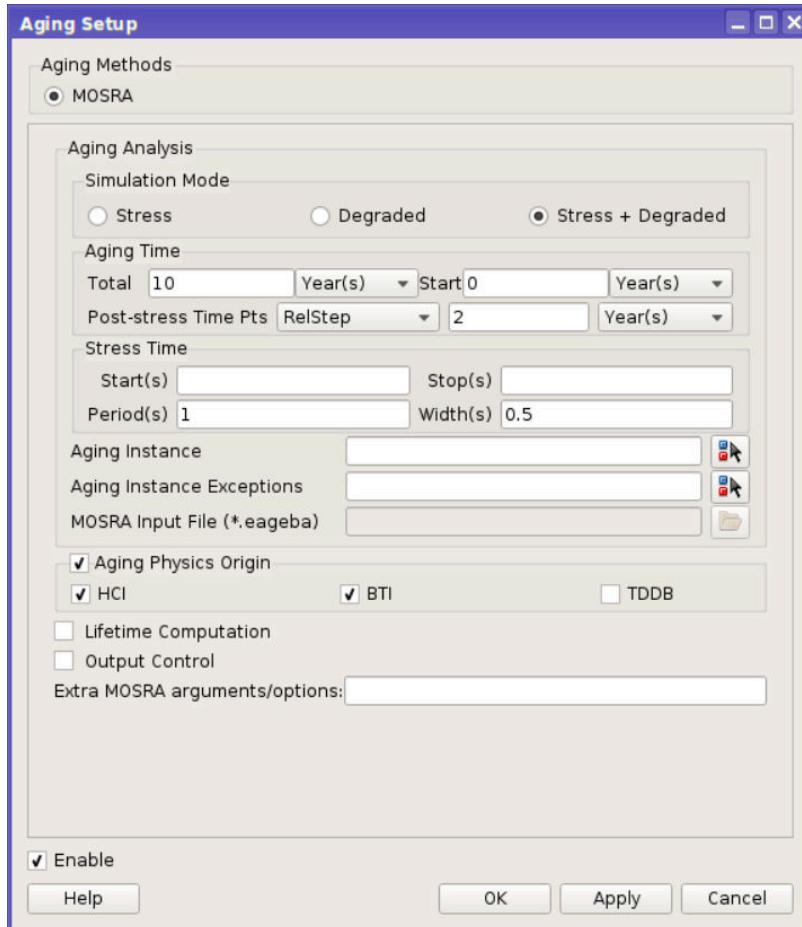
Figure 24 Ring Oscillator Frequency Versus Time for Fresh (Green) and 10-year-old Aged Devices (Yellow)



Observing Device Behaviors at Multiple Aging Points

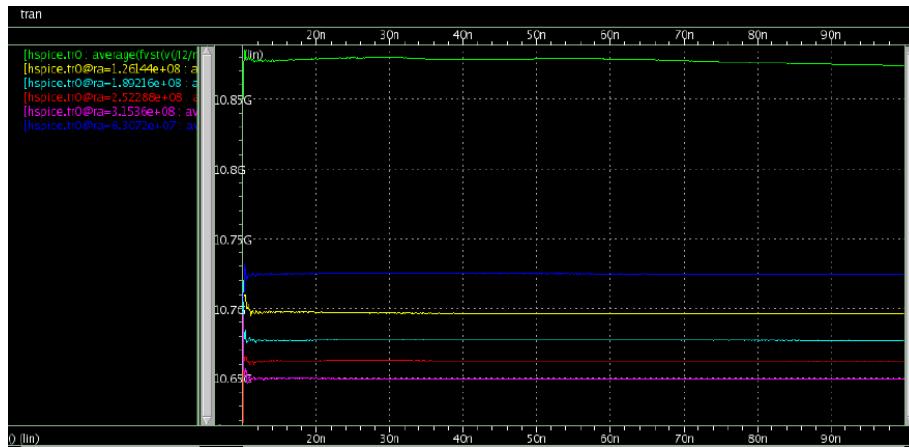
You can specify a **RelStep** to determine the behavior of the circuit at multiple aging points, up until the final aging time. A sample configuration is shown in [Figure 25](#) in which the **RelStep** is set to 2 years and the simulation mode is set to combined Stress + Degraded.

Figure 25 Setting Up a Stepped Simulation with a 2-year Step Size



The resulting simulation waveforms are shown in [Figure 26](#), which illustrates a family of waveforms showing the degraded oscillation frequencies at various circuit ages, from 2 years to 10 years in steps of 2 years.

Figure 26 Oscillator Frequency for Fresh (Green), 2-, 4-, 6-, 8-, and 10-year-old Devices



Looking at the PrimeWave Design Environment Results Table

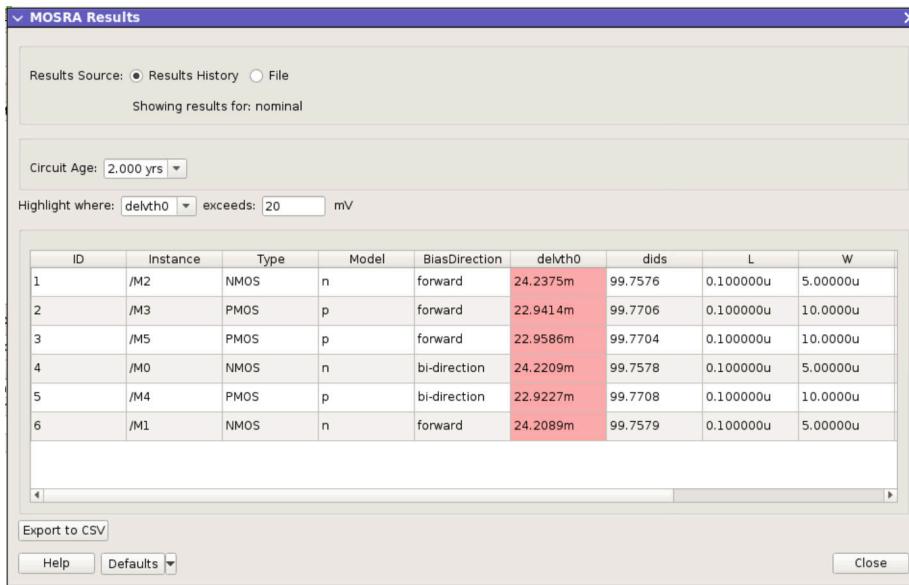
The PrimeWave Design Environment Results table can also be used to inspect the scalar measurement results. Because the simulation of [Figure 26](#) was performed with an outputs setup as shown in [Figure 27](#), in which several ymax measurements were specified (one for a node voltage, and one for a maximum oscillation frequency), you can see the tabular results as shown in [Figure 28](#).

Figure 27 PrimeWave Design Environment Outputs Configuration for Ring Oscillator Circuit

Output	Expression	Value	Analyses
v(I0/net17)		tran	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
average(v(I0/net17))		tran	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
v(I2/net15)		tran	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
ymax(v(I2/net15))	1.96802	tran	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
v(I2/net17)		tran	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
ymax(fvst(v(I2/net17),0.3,10n,percent=0))	10.7836G	tran	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
fvst(v(I2/net17),0.3,10n,percent=0)		tran	<input checked="" type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>
average(fvst(v(I2/net17),0.3,10n,percent=0))		tran	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Click to add			

The following figure lists the corresponding measured values of the oscillator at various aging times, from Fresh to 10 years old devices, in steps of 2 years. The specific time values are denoted in seconds by the `@ra=...` fields listed in the **Measurement** column.

Figure 28 PrimeWave Design Environment Results Table for Ring Oscillator Circuit



The screenshot shows a software window titled "MOSRA Results". At the top, it says "Results Source: ● Results History ○ File" and "Showing results for: nominal". Below that, "Circuit Age: 2.000 yrs" is set to "mV". A search bar says "Highlight where: delvth0 exceeds: 20". The main area is a table with the following data:

ID	Instance	Type	Model	BiasDirection	delvth0	dids	L	W
1	/M2	NMOS	n	forward	24.2375m	99.7576	0.100000u	5.00000u
2	/M3	PMOS	p	forward	22.9414m	99.7706	0.100000u	10.0000u
3	/M5	PMOS	p	forward	22.9586m	99.7704	0.100000u	10.0000u
4	/M0	NMOS	n	bi-direction	24.2209m	99.7578	0.100000u	5.00000u
5	/M4	PMOS	p	bi-direction	22.9227m	99.7708	0.100000u	10.0000u
6	/M1	NMOS	n	forward	24.2089m	99.7579	0.100000u	5.00000u

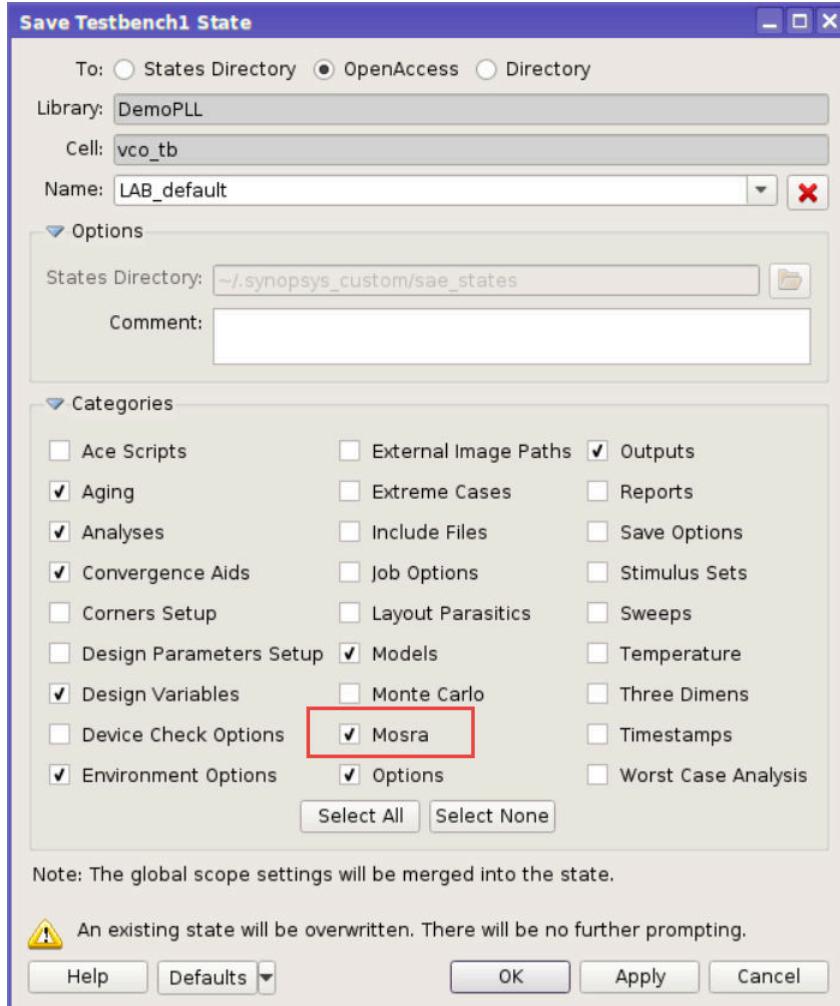
At the bottom, there are buttons for "Export to CSV", "Help", "Defaults", and "Close".

You can export MOSRA results to a CSV file by clicking the **Export to CSV** button at the bottom of the MOSRA results table.

Saving and Loading State

It is possible to save the user input values (specified in the setup dialog box of [Figure 5](#)) using the PrimeWave Design Environment **Save/Load State** commands. To enable this, a **Mosra** category checkbox is available in the **Save State** dialog box of [Figure 29](#).

Figure 29 Save State Dialog Box Showing Mosra Category



Conclusion

The PrimeWave Design Environment MOSRA flow offers a combined single simulation two-stage approach to aging analysis, in addition to optional flows that allow the stress and degraded stage simulations to be performed in explicit, separate simulation runs, perhaps using different stimulus. The PrimeWave Design Environment flow fits in naturally with the existing simulator MOSRA approach to aging and reliability. The PrimeWave Design Environment flow described here greatly simplifies the set up and results inspection for such a flow when using the Primewave Design Environment, allowing seamless switching between pure fresh and aged (degraded) device simulation histories.