Custom WaveView™ User Guide

Version K-2015.06, June 2015

SYNOPSYS®

Copyright and Proprietary Information Notice

Destination Control Statement

Disclaimer

Trademarks

Synopsys, Inc.
690 E. Middlefield Road
Mountain View, CA 94043
www.synopsys.com

Contents

1

Getting Started with the Custom WaveView™ Tool

This chapter contains information on how to get started with the Custom WaveView tool.

The Custom WaveView tool is a graphical waveform viewing and analysis program for analog/mixed-signal IC design simulations, which can help you to use simulation tools more effectively with:

- High-performance waveforms database I/O to access large amount of simulation data efficiently.
- Extensive mixed-signal display functions and analysis capabilities to extract design parameters from simulation result.

This chapter contains the following major sections:

Starting the Custom WaveView Tool

Before starting the Custom WaveView tool, consider any environment options you might want to set. See Setting Environment Variables for more information.

On UNIX and Linux platforms, enter `cx -w` or `wv` at the command line to start the Custom WaveView tool. See Supported Command-Line Options for a list of available command-line options you can use when starting the Custom WaveView tool.

Figure 1 shows the Custom WaveView main window.

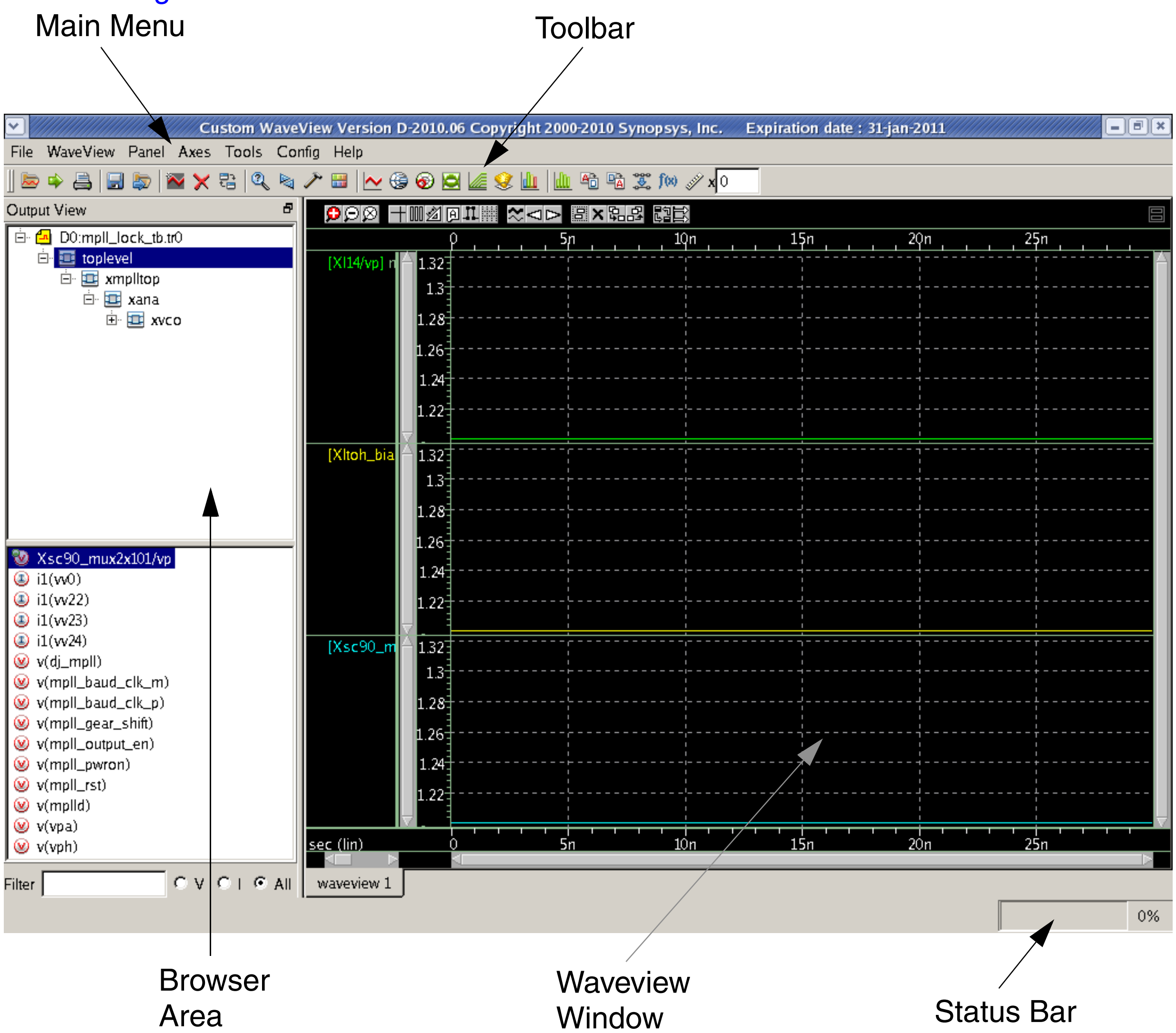


Figure 1 Custom WaveView main window

The main window is comprised of the following parts and associated functions:

- Main Menu: The top application menu bar.
- Toolbar: The toolbar shortcuts to some of the top menu entries.
- Status bar: The status bar reports progress of operations such as waveform loading and signal searching.
- Browser area: The Browser area contains the Output View browser including the top hierarchy browser and the bottom signal list window.

- OutputView browser: Displays waveform data hierarchies.

- Waveview window: The waveview window is the waveform viewing area. Multiple waveviews can be opened inside the waveview window.

  The width of the browser area and the waveview window can be adjusted using the vertical pane bar in between the Browser and the waveview area.

Supported Command-Line Options

The following command-line options are available:

Option	Action
-ace_perl_gui	Starts ACE Perl in GUI mode.
-ace_perl_no_gui	Starts ACE Perl in batch mode.
-ace_no_gui	Starts an ACE Tcl script in batch mode.
-ace_gui	Starts an ACE Tcl script in GUI mode.
-adcdump [file_name]	Dumps the contents of the ADC Toolbox display to a PNG file with the specified file name.
-group	Groups the opened waveform files passed at the command line. For example: `wv *.tr0 -group`
-m	Opens the ACE command console and shell.
-h	Help message.
-c	Converts to WDF in batch mode (see -r, -ri, -rv, -w).
-compare rule_file [out_file] [-x sx_file]	Compares waveforms in batch mode.
-display host:screen	Starts 'wv' by displaying the window to host.
-v	Reports software revision.
-priv	Enables a private colormap.

Option	Action
-k	Starts the Custom WaveView tool more quickly without the greeting window.
-load	Preloads waveform data to memory. fsdb and NPX-SDIF files cannot be preloaded. Just the sweep results from tr0 and NW files can be preloaded.
-r mode	Performs a reduction for WDF conversion. 0:default, loss-less, 1:medium reduction, 2:high reduction (see -c).
-replay_delay time	Sets the delay time in milliseconds when replaying a log file.
-ri itol	Reduction current tolerance (see -c, -r).
-rv vtol	Reduction voltage tolerance (see -c, -r).
-spxrc pref_file	Starts the Custom WaveView tool using the preference settings specified in the pref_file.
-x session_file	Loads a session file.
-y session_file	Applies a session file to existing waveform files.
file1, file2, …	Loads waveform files.

Setting Environment Variables

When starting the Custom WaveView tool, you might want to set one or more of the following environment variables:

- `SW_SX_QUEUE_LIC` (not supported on Windows platforms)

  Set to 1 to enable Synopsys license queuing. Flexlm license queuing is not yet supported.

- `SW_SX_FAST_COU`

  Set to 1 to enable the fast COU file reader. Defaulted to 0. Fast COU reader reads multi-run COU files much faster.

- `SW_SX_FAST_JWDB`

Set to 1 to enable the fast WDB file reader. Defaulted to 1. Fast WDB reader does not require the Java server.

- `SW_SX_HELP` (UNIX only)

  Points to the directory that contains the Custom WaveView online help.

- `SW_SX_INIT`

  Defines the location of the spxinit file for top menu customization.

- `SW_SX_INIT_DIR`

  Defines the Custom WaveView initial startup directory.

- `SW_SX_LOG_DIR` (UNIX only)

  Defines the output directory for the log file sxcmd.log. Default location is the current working directory.

- `SW_SX_MASKFILE`

  Points to user-defined mask files for eye diagrams.

- `SW_SX_ORG_ADC`

  Set to 1 to use the old version of the ADC Toolbox.

- `SW_LIC_TIMEOUT`

  Sets the amount of time that the Custom WaveView tool holds on to a license when idle. Once the set time is expired, a dialog box asks you to either reclaim the license or exit and save the session. The minimum value is 30 (minutes).

- `SW_SX_TK_LIB`

  Points to the Tk runtime library for running ACE

- `SW_SX_TMP_DIR` (UNIX only)

  Points to the temporary directory for reading a compressed (gzipped) waveform file. Default directory for temporary uncompressed files is the current working directory.

- `SW_SX_USE_AMAP` (UNIX only)

  Set to 1 to use the map files in the Artist amap/ directory to map signal names.

- `SW_SX_USE_TK`

  Set to 1 to enable using Tk scripts in the Custom WaveView tool.

- `SW_WLF_READER`

Points to the wlf2sx executable for reading the ModelSim WLF format.

- `SX_HOME` (UNIX only)

  Defines the Custom WaveView home directory. The Custom WaveView tool searches for the .spxrc file in $SX_HOME if SX_HOME is defined.

- `SW_SX_64`

  Set to 1 to point to the 64-bit binaries.

- `CX_DRTAB_FILE`

  Specifies a DRTAB mask spec file(s), which includes the following units:

  - minimum target slew (in V/ns units)
  - minimum target VIHL_AC (in V units)
  - minimum target TdiVW_Total (in s units)
  - minimum Tdipw (in UI units)

  The following sample is a DRTAB mask spec file:

```
#ddr4_spec DDR4_1600
#ddr_ui  660p
#vref_set_tol 0.15
#min_aperture 320p
#min_dataslew 1
#min_tdipw_ui 0.58
#min_vihl_ac 186m
```

Saving and Restoring Job Sessions

The Custom WaveView tool provides save and restore functions for users to save or retrieve a job session to or from a session file. The session file has a default file extension of `.sx`.

Saving a Job Session

To save your current session:

1. Click the **Save Session** toolbar button or choose **File > Save Session** from the main menu.

   The Save Session dialog box opens.

2. Select or enter the file in which to save your current job session.
3. (Optional) Click the **Use Relative Path in Session File** check box to use a relative path for your session file.
4. (Optional) Click the **Save Current Waveview Only** check box to save only the current waveview tab in your saved session file.
5. (Optional) Click the **Save a Copy of Loaded Signals** check box to save loaded signals to a new file named <session_file_name>.wdf.

   For example, if a session is saved to a file called `foo.sx`, the data file name is `foo.sx.wdf`. When loading the session file, only `foo.sx.wdf` is loaded.
6. Click **OK** to save your job session.

Restoring a Job Session

To load from a session file, click the **Load Session** toolbar button or choose **File > Load Session** from the main menu.

The Load Session dialog box appears. Accept the session file path, and select or enter the file from which to restore your job session.

A session file can be also loaded from the command line with the `-x` option. For example:

```
wv -x setup_file
```

If the **Apply to existing waveform files** option is not enabled, the session loader restores signal selection/layout using the waveform data files specified in the session file. If you prefer to apply the saved signal selection/layout to other loaded waveform data files in the application, enable the **Apply to existing waveform files** option. Select the preferred waveform data file for each restore target using the Waveform Source dialog box.

Use the -y option to apply session setup on different waveform data files from the command line. Usage of the `-y` option is:

```
wv -y setup_file file1 file2 … filen
```

Select the **Suppress popup warning messages** option to disable session loading warning messages (missing signals, for example).

Loading Setup from Other Viewer Tools

Setup files can be loaded from nWave (rc files) and Xelga (swd files). To load a setup file, choose **File > Load RC Files** from the Custom WaveView menu bar.

Setting Preferences

You can configure the Custom WaveView tool with the Preferences. Click the **Preferences** toolbar button or choose **Config > Preferences** from the main menu to open the Preference Settings.

The configuration entries are organized into five groups: General, Readers, Waveview, Panel, Signals, Colors, and Threshold. Configurable items of each group are placed inside a separated dialog box tab.

Click **Apply** to apply your changes, or **Save** to apply and save the changes.

A detailed description of each configurable option is described in Custom WaveView Preference Settings.

2

Working with Files

This chapter contains information on using files in the Custom WaveView tool.

This chapter contains the following major sections:

- Supported File Formats
- Importing Files
- Reading Compressed Waveform Files (UNIX Only)
- Working with WDF Files
- Converting Existing Output Files

Supported File Formats

The following waveform formats are supported:

- Cadence® Spectre® PSF, WSF (binary/ASCII), runObjFile
- Cadence PSPICE® DAT (binary), Allegro® SIM (ASCII)
- Verilog VCD (ASCII), ModelSim® WLF (binary)
- Synopsys HSPICE® tr0, sw0, ac0, mt0, hb0, nw0, mc0, mct0, rlgc, wzo, annotate, listing file (Binary/ASCII)
- Synopsys StarSim™ wdb, wv, xp (binary)
- Synopsys HSIM®/NanoSim®/CustomSim™ wdb (binary)
- Synopsys HSIM/NanoSim/TimeMill®/Powermill® out ,vector, .mt file (ASCII)
- Synopsys Saber AI/PL (binary/ASCII)
- Synopsys VCS®/Magellan™/DVE VPD (binary)

- Synopsys CustomSim .err, .errz, and .errt files
- Synopsys Galaxy Custom Designer® Simulation and Analysis Environment (SAE) resultsMap.xml and results.xml files
- Mentor Graphics® ELDO®/ADMS™ COU, WDB (binary), AdiT TR0/TB0.
- SPICE3/SmartSpice raw format (binary/ASCII)
- Agilent® ADS .ds format (binary), TouchStone® S-parameters (ASCII)
- CSDF (Common Simulation Data Format) (ASCII)
- Novas FSDB files (binary, v5.0)
- Text file, CSV (comma separated values) format
- Other in-house simulator formats (SDIF, MICS, TI-punch)
- Synopsys WDF format (binary)
- Binary scope data (Tektronix®, Agilent, Teledyne Lecroy®)
- IBIS models, Synopsys Liberty® files

Additional File Format Information

Addition information is available for the following file formats:

- Special Note for the ELDO COU Format
- Special Note for the ADMS/ELDO WDB Format
- Special Note for the PSF Format
- Special Note for the PSF Format

Special Note for the ELDO COU Format

The COU format from ELDO simulator can contain multiple simulation runs. Before loading a COU file, COU reader in the Custom WaveView tool needs to scan the entire file in order to determine how many runs the COU file contains.

For very large COU files, the detection process might take a very long time because the reader has to sequentially examine all data in the COU files.

To speed up this COU file loading process, the Custom WaveView tool has an alternative built-in search algorithm that can cut the file loading time significantly. To invoke the fast read option, set the SW_SX_FAST_COU environment variable to any value in your shell (for example, `setenv`

`SW_SX_FAST_COU`). Due to the nature of the COU format, the fast search algorithm has a trade-off that might potentially determine the number of runs of a COU file incorrectly. Although the possibility is very low, you can always disable the fast search algorithm by unsetting the environment variable (for example, `unsetenv SW_SX_FAST_COU`).

Special Note for the ADMS/ELDO WDB Format

By default, a built-in WDB file reader is used to load data from WDB files generated by ADMS and ELDO. However, for small amount of simulation waveform data, the WDB writer in these simulators might buffer waveform data in the WDB Java server without writing them to a disk file. To work around this problem, you can take one of the following actions:

1. For ELDO, set the -isaving option to a smaller value to force more frequent incremental saving to disk files. The file size specified in the -isaving option is in MB. For ADMS, set the JwdbSpillThreshold option in `$HOME/.vams_setup` to a smaller value to lower the buffer size.
2. Enter `setenv SW_SX_JWDB_API 1` at the command line to use the vendor-provided WDB API to load data directly from the Java WDB server. The API + Java WDB server operation, however, can be very slow for large WDB files.

Special Note for the PSF Format

PSF format assigns a unit for each signal in a simulation output file. For noise analysis, in order to display different waveforms using proper scales, signals with V^2/Hz unit are automatically converted to V/sqrt(Hz) by taking the square root of the loaded value during the waveform loading process.

The PSF format splits large output data (> 2GB) into multiple 2GB PSF files. The subsequent PSF files are hidden in the same directory (for example, the hidden file name starts with a period). The Custom WaveView tool automatically looks for the subsequent files in the same directory and loads waveform data from these files.

Special Note for the WDF and fsdb Formats

The WDF and the fsdb format both support split files. When the Custom WaveView tool opens a WDF or fsdb file, it automatically looks for subsequent files in the same directory. For example, if the loaded file is output.wdf, the Custom WaveView tool automatically looks for output.wdf.1. If the output file is output.fsdb.1, the Custom WaveView tool looks for output.fsdb.2. If you do not want to load the subsequent split files, disable the **Automatically connect to subsequent split files** option in the Loading Waveform Files dialog box.

Importing Files

The following file types can be imported into the Custom WaveView tool:

- Importing .err* Error Files
- Importing Multiple Files in Virtuoso
- Importing Textual Data

Importing .err* Error Files

To open an .err file, choose **File > Import .err File** from the main menu bar. The errors are displayed in the .err File viewer dialog box.

Importing Multiple Files in Virtuoso

Parametric simulations in the Virtuoso environment can generate multiple output files in multiple directories. The locations and associated sweeping variables of these files can be found in the runObjFile log file.

The runObjFile can be opened directly in the Custom WaveView tool to access these waveform files.

Virtuoso dcOp Parametric Analysis

A dcOp analysis from a Virtuoso parametric run can also generate multiple result files. Each of these corner files contains a single-point operating point result for each device parameter. A special function is added in the Output View signal browser context menu to display or export these single-point values together with their sweeping variable conditions. Right-click a signal in the signal browser, and choose **Signal ’NAME’ > Display Values** or **Signal ’NAME’ > Export Values**.

Importing Textual Data

The Custom WaveView tool supports data input in regular text format. Scaling factors (for example, 1ns) and scientific formats (for example, 1.0e-9) are both supported in the textual data files.

To distinguish Custom WaveView text data files from regular text files, the first line of a text data file is reserved for the format specification and must appear in the following form, where format-type is the format type keyword:

```
#format format-type
```

Lines beginning with a number sign (#) are treated as comments.

The following sections describe how to prepare textual input for

- Tabulated Data in Real Numbers
- Tabulated Data in Complex Numbers
- 2-Dimension Sweep Data in Real Numbers
- 2-Dimension Sweep Data in Complex Numbers
- Measured Data without Name Header
- SPICE PWL sources

Tabulated Data in Real Numbers

Syntax:

```
#format table
xvar signal1 signal2 ....
valuex value1 value2 ...
...
```

xvar and signaln are variable names and valuen are numeric values. As a special-case exception, the first line can be omitted if the name of `xvar` is `TIME`.

Example 1:

```
#format table
VGATE Ichannel
0.0 0.0
1.0 1u
2.0 5u
3.0 5.2u
...
```

Example 2:

The following data file can be recognized without the #format line:

```
TIME V(1) V(2)
0.0 5.0 2.5
1n 4.8 2.6
...
```

If the data file does not contain the x-value column, (commonly seen in measurement data), add keyword #dataonly to the #format line. The reader automatically assigns a 0-based row index as the x-axis value.

If the signal names are embedded in the last '#' comment line before the actual waveform data values, add keyword #cmt_name to the format line.

Tabulated Data in Complex Numbers

Format for complex numbers in (real, imaginary) form.

Syntax:

```
#format complex-ri
xvar signal1 signal2 ....
valuex value1-real value1-imag value2-real value2-imag ...
...
```

Format for complex numbers in (magnitude, angle [0-360]) form.

Syntax:

```
#format complex-ma
xvar signal1 signal2 ....
valuex value1-mag value1-angle value2-mag value2-angle ...
...
```

Format for complex numbers in (magnitude, phase [0-2]) form:

Syntax:

```
#format complex-mp
xvar signal1 signal2 ....
valuex value1-mag value1-phase value2-mag value2-phase ...
...
```

Example:

```
#format complex-ri
Freq vout
100meg 0.0 0.0
200meg 1.0 0.1
300meg 1.1 0.2
400meg 1.4 0.4
...
```

2-Dimension Sweep Data in Real Numbers

Syntax:

```
#format 2dsweep xvar2
xvar signal1 signal2 ....
#sweep value1-x2
valuex value1 value2 ...
...
#sweep value2-x2
valuex value1 value2 ...
...
```

where xvar2 is the name of the second sweep variable.

Example:

```
#format 2dsweep vgs
VDS Ichannel
#sweep 0.0
0.0 0.0
1.0 1u
2.0 5u
3.0 5.2u
#sweep 0.1
0.0 0.0
1.0 2u
2.0 6u
3.0 6.2u
...
```

2-Dimension Sweep Data in Complex Numbers

Format for complex numbers in (real, imaginary) form

Syntax:

```
#format 2dsweep-ri xvar2
#sweep value1-x2
xvar signal1 signal2 ....
valuex
```

```
value1-real value1-imag value2-real value2-imag ...
...
#sweep value2-x2
xvar signal1 signal2 ....
valuex value1-real value1-imag value2-real value2-imag ...
...
```

Format for complex numbers in (magnitude, angle [0-360]) form.

Syntax:

```
#format 2dsweep-ma xvar2
#sweep value1-x2
xvar signal1 signal2 ....
valuex value1-real value1-imag value2-real value2-imag ...
...
#sweep value2-x2
xvar signal1 signal2 ....
valuex value1-real value1-imag value2-real value2-imag ...
...
```

Format for complex numbers in (magnitude, phase [0-2]) form.

Syntax:

```
#format 2dsweep-mp xvar2
#sweep value1-x2
xvar signal1 signal2 ....
valuex value1-real value1-imag value2-real value2-imag ...
...
#sweep value2-x2
xvar signal1 signal2 ....
valuex value1-real value1-imag value2-real value2-imag ...
...
```

Example:

```
#format 2dsweep-ma temp
freq gain
#sweep 50
100meg 1.0 0
200meg 1.0 90
300meg 1.1 120
400meg 1.4 150
#sweep 75
100meg 1.0 0
200meg 1.0 80
300meg 1.2 110
400meg 1.3 140
...
```

Measured Data without Name Header

To import text measure data that does not have name header line or the x-axis value column, choose **File > Import Text/PWL Data** from the main menu bar. The Text/PWL Data Table window opens.

Select the desired options and click **Ok** to load the data file.

Note: When you open a Touchstone or SC0 file, the Z and Y parameters from the S parameters are generated and listed in the signal list.

SPICE PWL sources

To import SPICE PWL sources into the Custom WaveView tool, choose **File > Import Text/PWL Data** to open the Text/PWL Data Table dialog box. Choose the **SPICE PWL Source Data** option and click **Ok** to load the data file. The SPICE PWL sources must not use any variable parameters.

Reading Compressed Waveform Files (UNIX Only)

On UNIX platforms only, compressed (.Z) or gzipped (.gz) waveform files can be directly opened in the Custom WaveView tool. The tar.gz waveform files are not readable directly since a tar file can potentially contain multiple files.

By default the current working directory is used as the temporary directory to store the decompressed files. Decompressed files are automatically cleaned up when a the compressed file is closed in the Custom WaveView tool. Set the

SW_SX_TMP_DIR environment variable to redirect the temporary directory to a different location.

Working with WDF Files

The WDF (Waveform Data File) format is a Synopsys proprietary waveform storage format. The WDF format not only compresses analog and logic waveform data, it also facilitates fast waveform access for large data files. The compression scheme can be lossy or loss-less (default). Waveform data files in other formats can be converted into WDF files on demand.

The WDF format is directly supported by a number of simulators including HSIM, NanoSim, UltraSim, Spectre, AMS-Designer, and TimeMill/PowerMill.

You can extract signal files as a group. For example, you can create the following Tcl script file to extract signal files:

```
set f1 [sx_open_sim_file_read ./group.grp]
sx_current_sim_file $f1
set sig [sx_signal x1]
set sig2 [sx_signal x2]
set f2 [sx_open_sim_file_write ./save2d.wdf]
sx_save_signal $f2 "$sig $sig2"
sx_close_sim_file $f2
sx_close_sim_file $f1
```

Converting Waveform Files to WDF

To convert an open waveform file to WDF format, right-click the name of a waveform file name, and choose **Convert to WDF** from the menu that opens. The whole waveform file is saved into a new WDF file with X range you specify.

Reducing WDF Data

The data reduction option removes redundant data points from straight waveform segments based on user-specified error tolerances. Data points are removed if the error between the original data points and the reduced PWL segments are smaller than the user-specified tolerance.

Figure 2 depicts the scheme used in WDF data reduction.

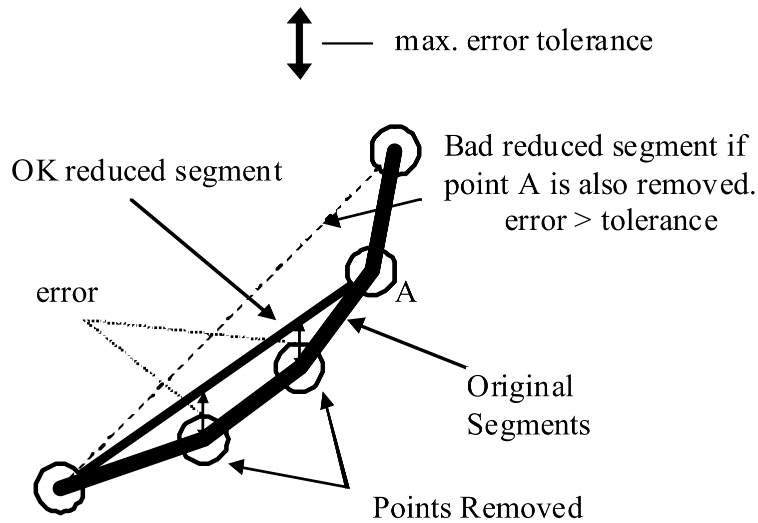


Figure 2 WDF data reduction scheme

The moderate reduction setting keeps at least 1 point out of four (4) original data points, while the more aggressive reduction mode keeps 1 out of ten (10) points. The -rv and -ri options are used to specify the reduction tolerance for voltage (or generic) waveforms and current waveforms respectively. The default setting is 5 mV for voltage waveform and 1 nA for current waveforms.

Using Direct WDF Output

Direct WDF Output from HSIM

In order to generate output files in the Synopsys WDF format using HSIM, you need to execute HSIM in association with the dynamic runtime library libWDF.so. If you are running HSIM 2.0, the runtime library is included with HSIM release. If you are running HSIM 1.3, contact your HSIM vendor to get a copy of the runtime library.

Required Netlist Parameters

HSIM uses two netlist parameters to specify the output format: `HSIMOUTPUT` and `HSIMCOILIB`. `HSIMOUTPUT` specifies the format type, while `HSIMCOILIB` specifies the path to the dynamic runtime library used to generate the output. To create output files in the Synopsys WDF format, add the following parameters to your netlist. The parameter `HSIMCOILIB` setting can be omitted if the runtime library is in the same directory with HSIM binary.

```
.PARAM HSIMOUTPUT=WDF
.PARAM HSIMCOILIB=<full_path_to_libWDF.so>
```

Example:

```
param HSIMOUTPUT=WDF
param HSIMCOILIB=/usr/local/lib/libWDF.so
```

Direct WDF Output from NanoSim

The libwdf.so runtime library is required to generate the Synopsys WDF output directly from NanoSim.

You can customize your simulation environment with one of the following ways to select the WDF format as your output format:

- Specifying WDF as the Default in .epicrc
- -out Command-Line Options
- *NanoSim cfg Command
- Configuration Command

Specifying WDF as the Default in .epicrc

To configure WDF as the default output format, add (or modify) the following commands in your .epicrc environment file:

```
print_format:wdf
output_postfix:wdf
output_custom_library_name:libwdf.so
output_custom_library_path:/path/to/libwdf.so
```

The .epicrc file is located in your home directory, your current working directory, or the installation directory, as defined by $EPIC_HOME.

-out Command-Line Options

To configure WDF locally as the output format for a simulation job, add the following options to your command line:

```
-out wdf
-outpostfix wdf
-outclname libwdf.so
-outclpath /path/to/libwdf.so
```

Only the -out option is necessary. The -outpostfix, -outclname, -outclpath options are not required. Their default value is wdf, libwdf.so, and ./, respectively.

**NanoSim cfg Command*

To configure WDF as the output format for a design, add the following comment line in your SPICE netlist file:

```
*Nanosim cfg="set_print_format for=wdf lib_name=libwdf.so path=/
path/to/libwdf.so postfix=wdf
```

The command needs to be added in a single comment line. Default value for lib_name, path and postfix is libwdf.so, ./, and wdf, respectively.

Configuration Command

Use the `set_print_format` configuration command directly to specify WDF as the output format. For example:

```
set_print_format for=wdf lib_name=libwdf.so path=/path/to/
libwdf.so postfix=wdf
```

Default value for `lib_name`, `path`, and `postfix` is `libwdf.so`, `./`, and `wdf`, respectively.

Direct WDF output from Cadence Spectre or UltraSim

Add the following netlist command to generate WDF files from UltraSim (6.0 or newer revision):

```
.usim_opt   wf_format=wdf
```

Add the following option to a Spectre netlist to output WDF files directly from Spectre (5.1.41 or newer revision):

```
options rawfmt=wdf
```

Direct WDF output from Cadence Verilog-XL/NC-Verilog

Runtime library libvpi.so is required to generate the Synopsys WDF output directly from Cadence Verilog-XL/NC-Verilog simulators. Contact Synopsys support to get a copy of the runtime library.

Make sure that the directory path to the runtime library libvpi.so is included in the `LD_LIBRARY_PATH` environment variable.

The following WDF system tasks can be used in Verilog initial blocks to control the WDF output:

System Task	Description
$wdf_open("output_filename")	$wdf_open is used to specify the output file name.
$wdf_probe(list of modules)	$wdf_probe is used to enable WDF dump for the specified modules.
$wdf_close()	$wdf_close is used to close the open WDF file.

Converting Existing Output Files

One or more output files can be converted (in batch mode) to Synopsys WDF files using the `-c` command-line option.

Lossless Compression

With lossless compression, all original simulation output data points are preserved.

```
wv -c file1 file2 … filen
```

Lossy Compression

You can use the `-r` command-line option to turn on data reduction for the compression process. Use `-r 1` for moderate reduction or `-r 2` for a more aggressive reduction.

```
wv -c -r (1 or 2) <-rv 0.001> <-ri 10n>
```

3

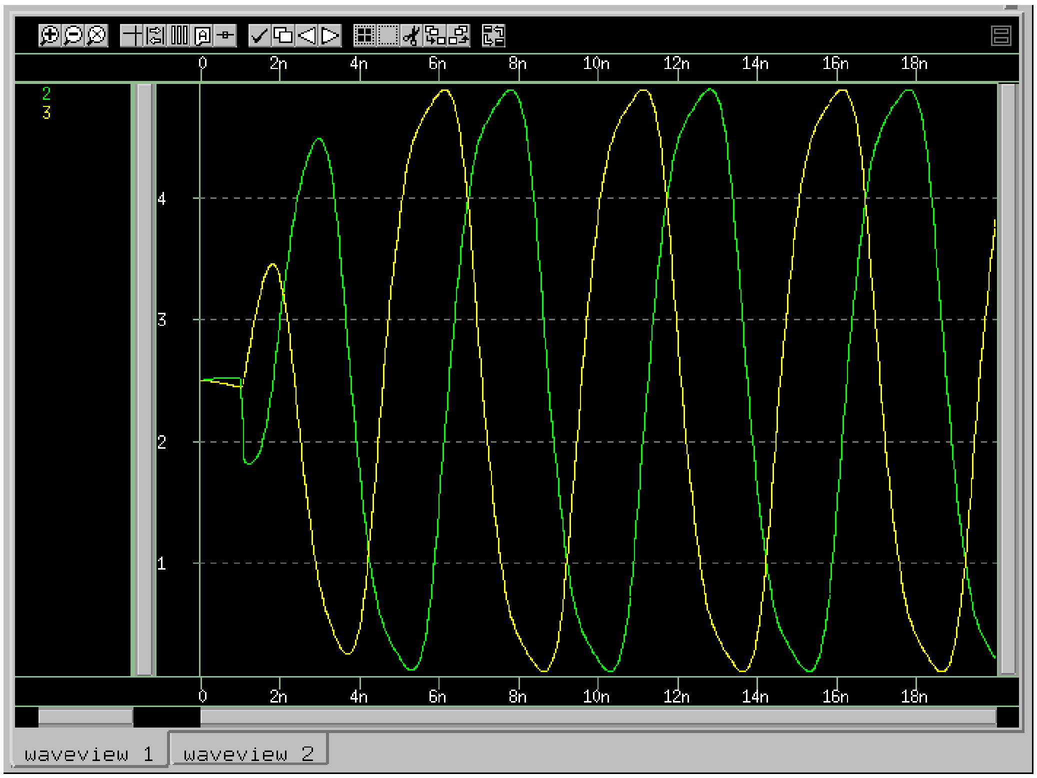
Using the Waveview Window

This chapter contains information on how to use the waveview window.

This chapter contains the following major sections:

Working with Waveviews

The waveview window is the waveform display area in the Custom WaveView tool. Click over the corresponding tab area of a waveview to bring it to the foreground.



The following topics are covered in this section:

- Displaying Waveview Windows
- Adding New Waveviews
- Docking and Undocking Waveviews
- The Active Waveview
- Refreshing Waveviews
- Deleting Waveviews
- Renaming Waveviews
- Undoing Waveview Operations

- Undoing Waveview Operations
- Toggling the Hierarchy and Signal Browser Displays
- Displaying Waveview Titles
- Clearing Waveview Contents
- Changing the Order of Tabbed Waveviews
- Synchronizing Waveviews
- Dumping the Waveview Contents
- Toggling the Console Window Display

Displaying Waveview Windows

A waveview window can have one or more non-overlapping Panels. Panels in a waveview can be arranged in two modes: vertical stack or independent row/column. In the vertical stack mode, panels stack top down and share a common horizontal axis. In the independent row/column mode, panels are arranged from left to right in a row. Panels in the row/column mode are independent of each other. A small icon at the upper-right corner of the waveview indicates the waveview orientation.

The following display modes are available:

- Stack Mode
- Row and Column Mode
- Vertical Row and Column Mode
- Horizontal Row and Column Mode
- Tiled Row and Column Mode
- TableView Mode

Stack Mode

To view waveviews in vertical stack mode, right-click a waveview tab and choose **Stack Layout** from the menu that opens. Waveviews are displayed in this mode by default.

Figure 3 shows the vertical layout display.

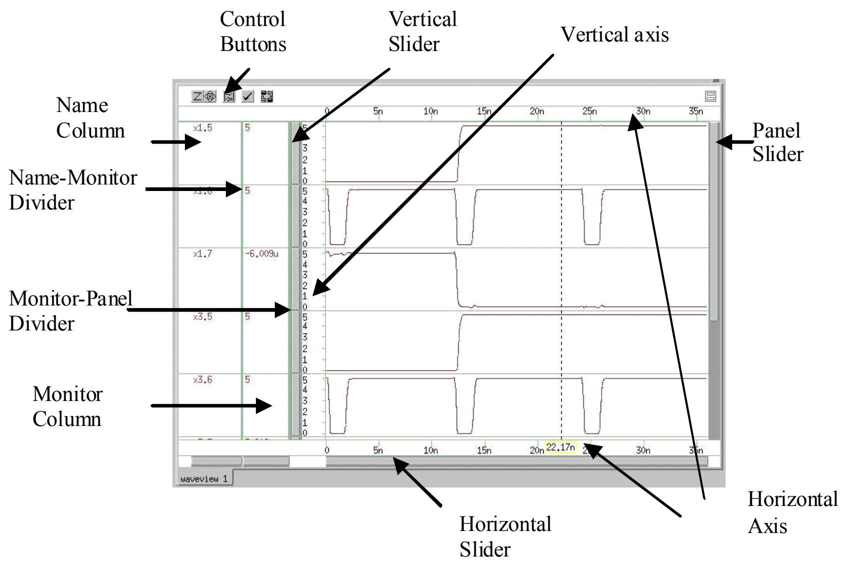


Figure 3 The Vertical Stack mode

The vertical layout has the following display components:

- Control Buttons: Provides shortcuts to some of the commonly used waveview functions.
- Vertical Slider: Panes zoomed waveforms vertically. It is not displayed for logic panels.
- Vertical Axis: The vertical axis of a panel. For a logic panel, it displays the vector width and radix.
- Panel Slider: Panes the entire panel stack vertically when the waveview height is too small to show all panels. Mouse wheel can be used to scroll this slidebar.
- Horizontal Axis: The common x-axis of all panels, which appears at both the top and the bottom of a vertical waveview.
- Horizontal Slider: This slide bar is used to pane zoomed waves horizontally.
- Monitor Column: Displays waveform or cursor-related values. Multiple monitors can be added. A horizontal slide bar at the bottom scrolls long value strings horizontally.
- Monitor-Panel Divider: Defines the left boundary of the waveform plotting area. Click and drag over the divider to resize the width of monitor columns.

- Name-Monitor Divider: Defines the border between the name and monitor area. Click and drag the divider to resize the width of name and monitor columns.
- Name Column: Displays waveform names. A vertical slide bar scrolls name list vertically. A slide bar at the bottom scrolls long name strings horizontally.

Row and Column Mode

To view waveviews in row and column mode, right-click a waveview tab and choose **Row/Column Layout** from the menu that opens.

Vertical Row and Column Mode

To view waveviews in vertical row and column mode, right-click a waveview tab and choose **Single Column Layout** from the menu that opens.

Horizontal Row and Column Mode

To view waveviews in horizontal row and column mode, right-click a waveview tab and choose **Single Row Layout** from the menu that opens.

Figure 4 shows the independent row and column display.

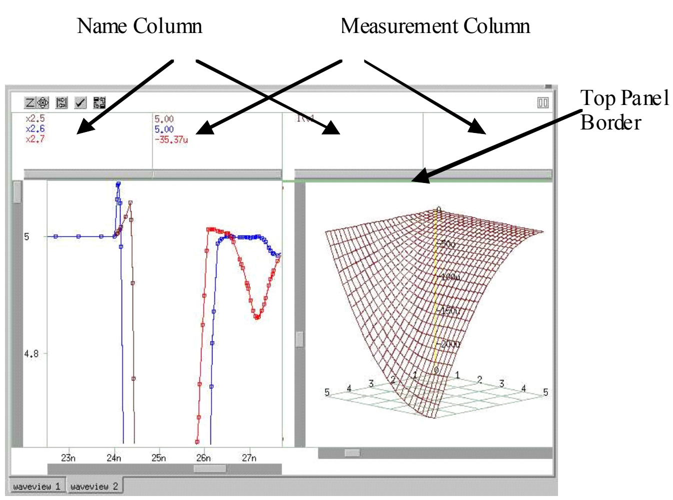


Figure 4 Horizontal Row and Column mode

Panels in the independent row/column layout are arranged in a similar way to the Vertical Stack mode, except for the following differences and additional display components:

- Name column: The name column is located at the top of each panel. The vertical and horizontal slide bars can be used to scroll long list or long name strings.
- Monitor column: The monitor column is at the top of each panel. A horizontal slide bar at the bottom scrolls long value strings horizontally.
- Top Panel border: The top border can be moved to redefine the top boundary of the waveform plotting area. Drag the divider and move it vertically to resize name/monitor height.

Each waveview window is associated with a waveview context menu that can be invoked from the upper right corner of the window, or from the waveview tab area. Choose an item from the waveview context menu to rename, delete or refresh a waveview, or edit the waveview title.

Tiled Row and Column Mode

To view waveviews in tiled row and column mode, right-click a waveview tab and choose **Tile Row/Column Layout** from the menu that opens.

TableView Mode

To view tabular data, click the  button in the button toolbar. Only tabular views are allowed in tableviews.

See Creating Tabular Panels for more information on tabular panels.

Adding New Waveviews

An empty waveview is initially opened when the application starts. To add more waveviews into the Wave window, click the **New Waveview** toolbar button or select the top menu **WaveView > New**. The newly created waveview is placed on the top of the waveview stack.

Docking and Undocking Waveviews

Overlapping (docked) waveviews in the waveview window can be undocked into individual pop-up windows. Choose **WaveView > Dock/Undock** from the main menu to toggle all waveviews between the docked and undocked modes. All waveviews must be docked or undocked together. Waveviews cannot be docked or undocked individually.

Waveviews can also be docked or undocked using the waveview docking control button located at the top of each individual waveview window.

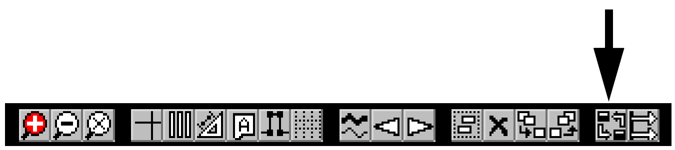


Figure 5 Waveview docking control button

The Active Waveview

In the dock mode, the top Waveview is considered the active Waveview. In the undock mode, the topmost Waveview or the selected Waveview is the active Waveview. Select a Waveview window from the top menu **WaveView > Select**, or click a window frame or tab. The selected waveview becomes the active waveview.

Refreshing Waveviews

Select the top menu **Waveview** > **Refresh** to refresh the display content of the active waveview.

Deleting Waveviews

To delete the active waveview, click the **Delete Waveview** toolbar button or choose **WaveView > Delete** from the main menu. Choose **WaveView > Delete All** to delete all waveviews.

Renaming Waveviews

Choose **WaveView > Rename** from the main menu to rename the active waveview. To rename a waveview directly, right-click the name tab area (or the upper right corner) of a waveview and select **Rename WaveView**.

Undoing Waveview Operations

To undo a waveview operation, choose **WaveView > Undo** to undo the previous waveview operation. You can undo object insertions and deletions; axis, radix, and display settings; and zoom operations. Some waveview operations, such as adding or deleting a waveview or relocating a signal, cannot be undone.

Toggling the Hierarchy and Signal Browser Displays

To toggle the Hierarchy and Signal Browser displays, choose **WaveView > Hide/Show Browser** from the main menu bar or press **Ctrl-H** on your keyboard.

Displaying Waveview Titles

To hide or display the title of a waveview, right-click a waveview tab and choose either **Show Title** or **Hide Title** from the menu that opens.

Clearing Waveview Contents

To clear the contents of a waveview window, right-click inside a waveview and choose **Clear Waveview** from the menu that opens.

Changing the Order of Tabbed Waveviews

If you have multiple waveviews open at the same time, you can change the order of those waveviews by moving the associated tabs left or right.

To change the order of tabbed waveviews, right-click the tab of the waveview you want to move, and choose **Move to Left** or **Move to Right** and **1 tab**, **2 tab**, **3 tab**, **4 tab**, or **5 tab**.

Synchronizing Waveviews

A stack-mode waveview can be synchronized with other stack-mode waveviews. Cursors and the X-axis of a synchronized waveview are updated automatically when the X-axis display range or the active cursor location changes in other synchronized waveviews. Choose **WaveView > Sync/Unsync** to toggle the sync state of the active waveview, or choose **WaveView > Sync/Unsync All** to toggle the sync states of all waveviews.

Dumping the Waveview Contents

All waveforms in a waveview can be included when performing a screen dump, even if the waveforms are scrolled above or below the currently visible portion of the waveview window.

To include all off-screen waveforms, choose **WaveView > Dump Screen** from the main menu. Click the **Use Maximum WaveView Height** check box, and click **OK**.

On Microsoft platforms, you can copy and paste the waveview contents to any application that has access to the clipboard. To copy the screen bitmap of a waveview window to the clipboard, choose **Copy to Clipboard** from the waveview context menu. To export a waveview in the vector-based Windows EMF (Enhanced Meta File) format, choose **Screen Dump to EMF**.

On UNIX platforms, you can dump the waveview contents to the JPEG, PNG, or EMF formats. Choose **Dump Screen** from the waveview context menu to dump the display content.

Toggling the Console Window Display

To toggle the display of the Console window, choose **WaveView > Hide/Show Console** from the main menu bar.

Working with Waveview Panels

Panels are created automatically when signals are dropped into a waveview window. Default panel type is selected based on the signal type. To manually add a specific type of panel into the active waveview window, choose **Panel > New** from the main menu.

The following panel types are available:

- Creating X-Y Panels
- Creating Smith Chart Panels
- Polar Plot Panel
- Creating Eye Diagram Panels
- Creating Logic Panels
- Creating 3-D Sweep Panels
- Creating 2-D Sweep Panels
- Creating Separator Panels
- Creating Tabular Panels
- Creating Histogram Panels

Creating X-Y Panels

An X-Y panel is designed to display analog waveforms. It is the only panel type that is allowed in both vertical stack and row/column waveviews. The X-Y panel accepts signals that are represented in (valuex,valuey) pairs, ((x,y) data points, for example), where valuex can be the default sweeping variable of a signal, or the signal value of a regular signal; valuey can be either the value of a real-signal or the magnitude/phase/real/imaginary part of a complex signal. The zoom operation is supported along both axes, and the logarithmic scale is also supported along both axes.

Waveforms in linear X-Y panels can be displayed with different plotting options including PWL (piece-wise-linear) line-type, PWC (piece-wise-constant) line-type, bar chart and data-point only mode. The default line type is determined by the type of waveform. Choose **Panel > Display Preference** to change the plotting mode.

Creating Smith Chart Panels

Smith charts display complex normalized impedance, admittance, and scattering parameters (S-parameters).

The Smith chart accepts signals that are represented in (real, complex) pairs. The normalized impedance is defaulted at 50 ohms and can be modified in Preference Settings (see the `smith_char_impedance` option in the `.spxrc` configuration file).

Smith charts are allowed only in horizontal waveviews.

You can configure a Smith chart in the Smith Chart Settings dialog box. Choose **Attributes** from the panel context menu to open the dialog box.

Enter the desired setting preferences and click **OK** to apply the changes.

Cursors (up to two) in a Smith chart can be used to trace Gamma (the reflection coefficient), SWR (standing wave ratio), impedance (Z) or admittance (Y) at the location where the cursor hairs cross.

You can use the Data(X,Y) measurement to trace the frequency, magnitude, and phase values of a signal in a Smith chart.

The **Zoom in** and **Zoom out** buttons are used to zoom in and out a Smith chart. The **Full Unzoom** button displays a full-sized Smith chart.

Polar Plot Panel

Polar plots display data on a complex plane. They are allowed only in horizontal waveviews. The Polar plot accepts signals that are represented in (real, complex) pairs. The zoom operation is supported only along the radial direction.

Creating Eye Diagram Panels

To create an empty Eye diagram panel, choose **Panel > New > Eye Diagram** from the main menu. Signals are then dragged from the Output View browser and dropped into the Eye diagram. The Eye diagram by default displays signals in the folded mode.

A folded Eye diagram partitions signals into continuous intervals that are then overlapped to form the Eye diagram. The interval width can be user-specified or automatically determined.

Zoom operations are supported in both folded and unfolded eye diagrams.

The shift parameter is used to control the alignment of a folded Eye diagram. A slider at the lower left corner of the Eye diagram panel allows users to interactively change the shift setting.

This section contains information on the following topics:

- Unfolding Eye Diagrams
- Tracing Waveform Points in Eye Diagrams
- Configuring Eye Diagrams
- Generating Jitter Histograms
- Measuring Eye Diagrams
- Adding User-Defined Masks
- Measuring DDR3 Eye Diagrams
- Measuring DDR4 Eye Diagrams

Unfolding Eye Diagrams

An Eye diagram can be unfolded by choosing **Unfold Eye** from the panel context menu. The unfolded Eye diagram displays the original signals and the corresponding intervals highlighted in the red box at the top.

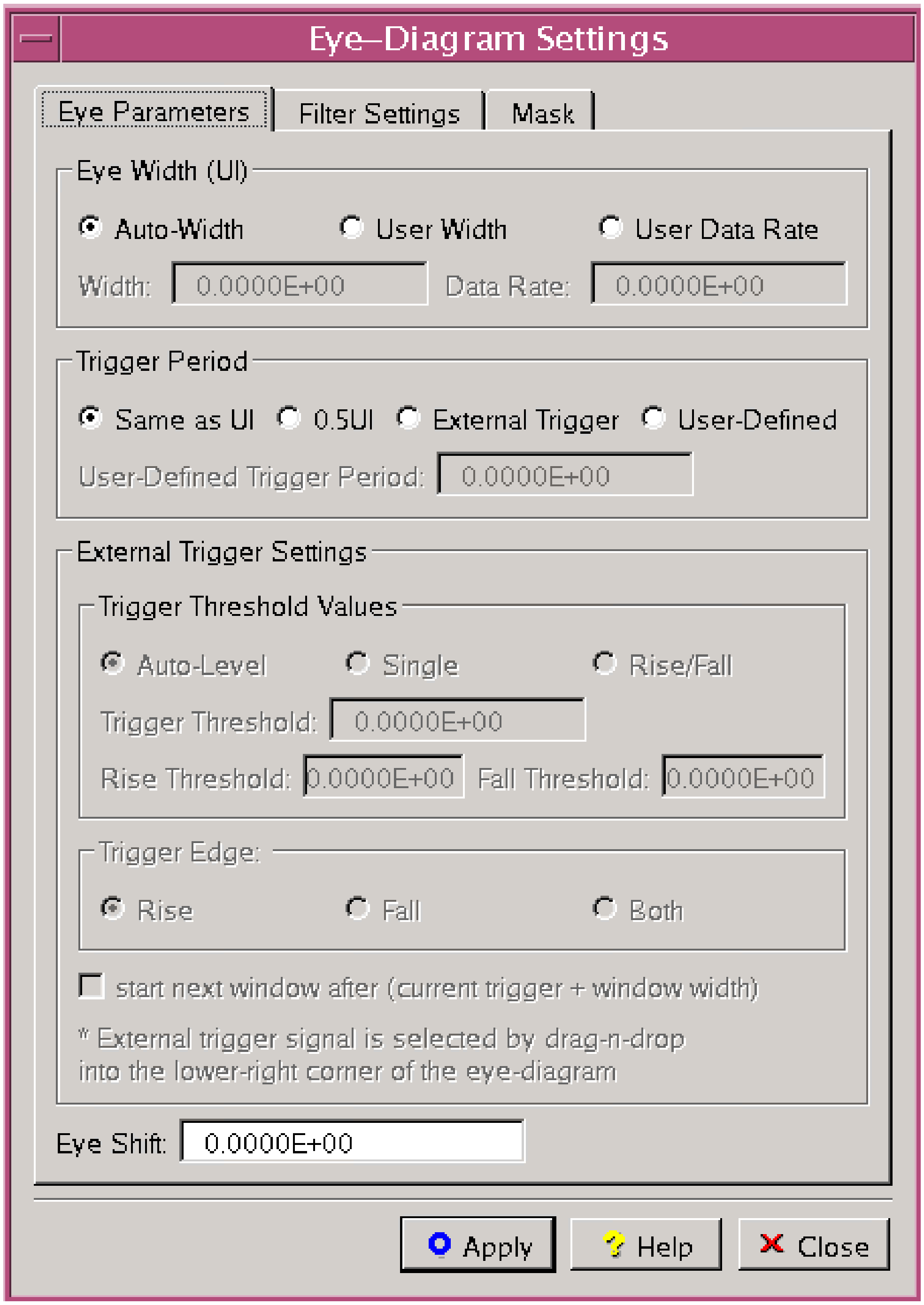
Similar to an oscilloscope, the Eye diagram also accepts a signal as an external trigger to construct the folded Eye diagram. To specify the external trigger signal, drag the signal from the Output View browser and drop it into the lower right corner of an Eye diagram panel. A hint box highlights the drop location for the external trigger signal.

Tracing Waveform Points in Eye Diagrams

You can use cursors in an eye diagram to trace a waveform point location in the unfolded eye. Choose **Trace** from the cursor context menu of the corresponding eye diagram location. Cursors are jumped to the corresponding location when the eye is unfolded.

Configuring Eye Diagrams

You can change eye diagram parameters from the Eye Diagram Settings dialog box. Right-click an eye diagram, and choose **Configure Eye** from the menu that opens. The Eye Diagram Settings window opens.



Eye diagram settings can be copied and pasted between multiple eye diagrams; right-click an eye diagram and choose either **Copy Settings** or **Paste Settings** from the context menu.

The following sections explain the eye diagram settings that are available:

- Setting Eye Parameters
- Setting Filter Parameters
- Setting Mask Parameters

Setting Eye Parameters

The following parameters are set from the Eye Parameters tab in the Eye Diagram Settings window:

Parameter Name	Description
Auto-Width	Automatically determines the eye width or frequency. In Auto-Width mode, an eye diagram always uses the first signal in the panel to determine the eye width.
User Width	Specifies the eye width or frequency using the value you enter in the Width text box.
User Data Rate	Specifies the data rate using the value you enter in the User Data Rate text box.
Same as UI	Sets the eye trigger period to the same as the eye diagram UI width.
0.5UI	Sets the eye trigger period to 0.5UI.
External Trigger	Sets an external trigger using the settings in the External Trigger Settings section of the Eye Diagram Settings window. See the following table for the External Trigger Settings.
User-defined	Sets the eye trigger period to the value your enter in the User-Defined Trigger Period text box.
Eye Shift	Sets the eye shift value

The following Eye Parameter settings are available in the External Trigger Settings section when you choose the **External Trigger** option for the trigger period:

Parameter Name	Description
Auto-Level	Uses the average magnitude of the external trigger signal as the trigger level.
Single	Uses a single trigger value you enter in the Trigger Threshold text box.
Rise/Fall	Uses the rising and falling trigger values you enter in the Rise Threshold and Fall Threshold text boxes, respectively.
Rise	Sets the trigger at the rising edge.
Fall	Sets the trigger at the falling edge.
Both	Sets the trigger at both the rising and falling edge.

Setting Filter Parameters

The following parameters are set from the Filter Settings tab in the Eye Diagram Settings window, which is used to exclude unwanted time periods when forming an eye diagram:

Parameter Name	Description
No Filter	Specifies that no filter is enabled.
Index Range	Enables a filter for a specified index range. Enter values for the Starting Window Index and Number of windows to select the range.
Time Range	Enables a filter for a specified time range. Enter values for the Range Starting Time and Range Stopping Time.

Setting Mask Parameters

User-defined masks can be added using mask files.

The following parameters are set from the Mask tab in the Eye Diagram Settings window, which specifies a mask that can be superimposed over a folded eye diagram as a timing specification:

Parameter Name	Description
Show Mask	Enables the mask display on eye diagram waveforms. Choose the mask type and color from the **Mask Type** and **Color** menus, respectively.
Show Mask on Top of Waveforms	Displays the mask on top of eye diagram waveforms.
Center Mask to Eye Opening	Centers a hexagon mask to the eye opening.

Cursors in a folded eye diagram trace the eye fuss (eye height/width) or the envelops of the eye pattern. Select the desired **Eye fuss** values and eight envelop measurements (upper/lower/left/right by outer/inner) from the cursor context menu. If a cursor is selected to trace the eye envelope, the cursor is automatically moved to its corresponding unfolded position when the eye diagram is unfolded.

Generating Jitter Histograms

Cursors are normally used in Eye Diagrams to trace the eye envelope or fuss. They can be also used to produce jitter histograms from the eye envelope pattern. To create the jitter histograms, move a cursor and position its horizontal hair at a preferred level. Click the right mouse button on a cursor to open the cursor context menu and choose **View Histogram**.

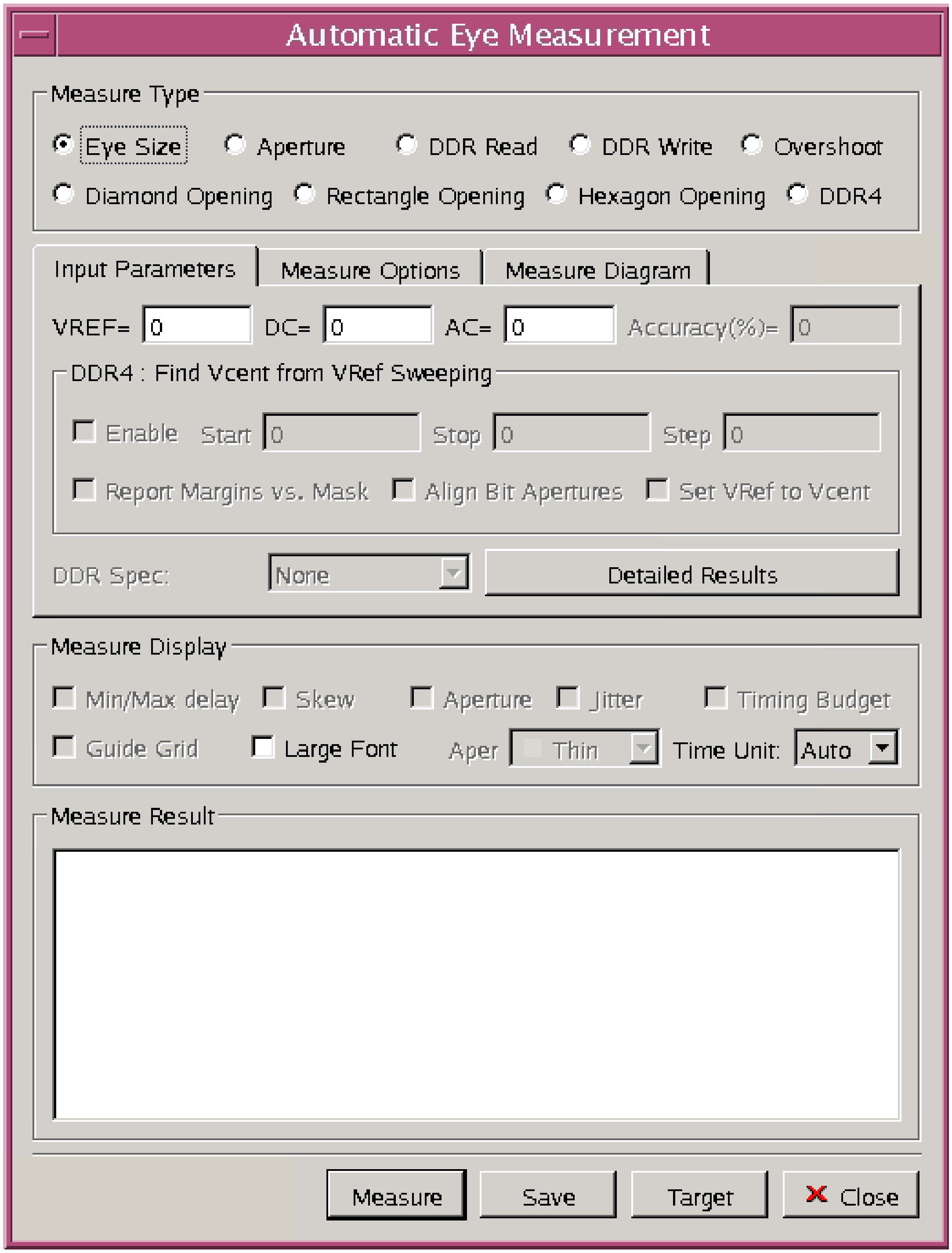
The time (the offset into a folded eye window) is automatically sampled for each cross-point between the horizontal hair and the eye waveform, and creates histograms for the two areas.

The generated histograms display the jitter distributions of the two cross-section areas.

Measuring Eye Diagrams

The Automatic Eye Measurement tool can take eye opening or aperture measurement on an eye diagram. Right-click an eye diagram, and choose

Measure Eye from the menu the opens. The Automatic Eye Measurement window opens.



The following sections contain information on the measurements available in the Automatic Eye Measurement window:

- Eye Measurement Methodology
- Measuring Eye Size
- Measuring Aperture
- Measuring DDR Read
- Measuring DDR Write
- Measuring Overshoot
- Measuring Diamond Openings
- Measuring Rectangle Openings
- Measuring Hexagon Openings
- Measuring DDR4
- Clearing Eye Measurements

The same eye measurement functions are also available as ACE functions for batch mode measurements. Refer to the *Analysis Command Environment Reference (ACE) Reference Manual* for details.

Eye Measurement Methodology

The permanent default value for vref, vdc, and vac is 0.0. The vref, vdc, and vac values calculate the MH, ML, OH, OL, WH, and WL without checking which one is specified.

You can calculate MH and ML using the following method:

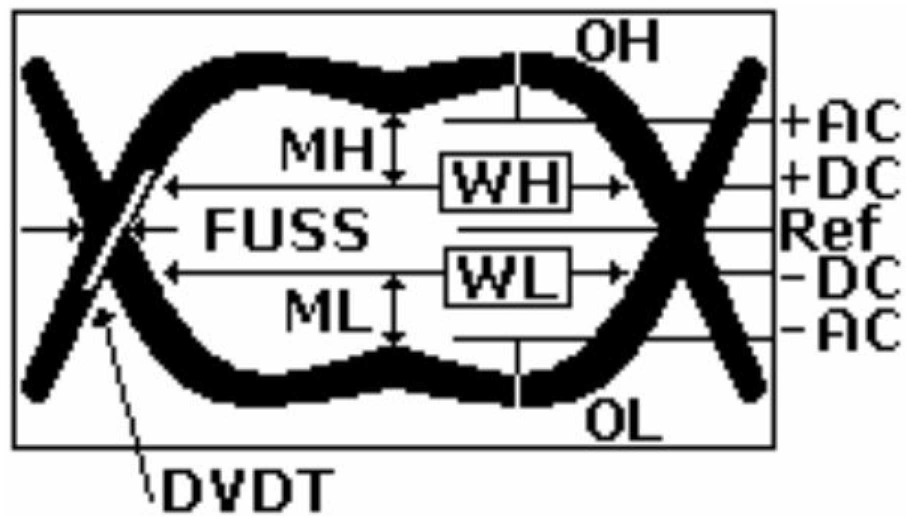
```
MH = (lower limit of the upper eye) - (vref+vdc)
ML = (vref - vdc) - (upper limit of the lower eye)
```

You can calculate OH and OL using the following method:

```
OH = (upper limit of the upper eye) - (vref+vac)
OL = (vref-vac) - (lower limit of the lower eye)
```

WH is the eye opening width at level (vref+vdc). WL is the eye opening width at level (vref-vdc).

The following figure shows how the expressions mentioned are used to calculate eye diagram measurements:



Measuring Eye Size

The following eye size measurement options are available:

Option	Description
VREF, **AC**, and **DC** Values	The VREF, AC, and DC values use for the eye measurement.
Use DVDT Levels	Defines the lower and upper voltage levels for the DVDT measurement.
Large Font	Sets the font size to large.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Measuring Aperture

The following eye aperture measurement options are available:

Option	Description
Use Wider Aperture for VDC-Ringback Divided Eye	Evaluates the width for two openings on two sides of a glitch and uses the wider one as the measurement result. If not enabled, the aperture measurement always starts after any VDC crossing ringback.
Use 2-Box Aperture with no VAC Crossing Before	Sets the value you enter in the Tminac as the time for the width of the first box (the left box). The first box is free of VAC glitches, while the 2nd box (right box) is free of VDC glitches.
Use First Signal as T-Budget Setup/Hold Reference	Sets the first signal in the panel as the reference signal for setup/hold measurement.
VREF, **AC**, and **DC** Values	The VREF, AC, and DC values used for the eye measurement. You can enter multiple values for VREF by separating them with colons (:), semicolons (;), or commas (,).
Trigger Clock Delay	Sets the trigger clock delay value you enter in the Trigger Clock Delay text box. This option is disabled if you enable the **Delay Clock to Aperture Center** option.
Delay Clock to Aperture Center	Sets the trigger clock delay to the aperture center value.
Min/Max Display	Displays the min and max delay from the eye.
Skew	Displays the skew measurement on the eye.
Aperture	Displays the aperture.
Aperture Violation	Displays aperture violations.
Jitter	Displays the point-to-point and RMS jitter as a histogram, which is plotted next to the Vref crossing area.
Timing Budget	Displays the setup and hold measurements.

Option	Description
Guide Grid	Displays the guide grid.
Large Font	Sets the font size to large.
Aper	Sets the line size and color for the aperture.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Measuring DDR Read

The following DDR read measurement options are available:

Option	Description
VREF, **AC**, and **DC** Values	The VREF, AC, and DC values used for the DDR read measurement.
Guide Grid	Displays the guide grid.
Large Font	Sets the font size to large.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Measuring DDR Write

The following DDR write measurement options are available:

Option	Description
VREF, **AC**, and **DC** values	The VREF, AC, and DC values used for the DDR write measurement.
Guide Grid	Displays the guide grid.
Large Font	Sets the font size to large.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Measuring Overshoot

The following overshoot measurement options are available:

Option	Description
VREF value	The VREF value used for the overshoot measurement.
Large Font	Sets the font size to large.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Measuring Diamond Openings

The following diamond opening measurement options are available:

Option	Description
Accuracy(%)	The accuracy percentage of the diamond opening measurement.
Large Font	Sets the font size to large.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Measuring Rectangle Openings

The following rectangle opening measurement options are available:

Option	Description
VREF Value	The VREF value used for the rectangle opening measurement.
Large Font	Sets the font size to large.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Measuring Hexagon Openings

The following hexagon opening measurement options are available:

Option	Description
VREF Value	The VREF value used for the hexagon opening measurement.
Large Font	Sets the font size to large.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Measuring DDR4

The following DRR4 measurement options are available:

Option	Description
VREF and **AC** Values	The VREF and AC values used for the DDR4 measurement.
Enable	Find the Vcent from VRef sweeping using the Start, Stop, and Step values you enter. The VREF value is not used when you enable this option.
Report Margins vs. Mask	Measures the margins versus the mask defined by the `CX_DRTAB_FILE` environment variable. See Setting Environment Variables for information on setting this environment variable.
Align Bit Apertures	Aligns all the bit apertures.
DDR Spec	The DDR specification to use for the measurement.
Detailed Results button	See Using the Setup/Hold Results Viewer for information on the Setup/Hold Results Viewer, which you can access via the **Detailed Results** button.
Trigger Clock Delay	Sets the trigger clock delay value you enter in the Trigger Clock Delay text box. This option is disabled if you enable the **Delay Clock to Aperture Center** option.

Option	Description
Delay Clock to Aperture Center	Sets the trigger clock delay to the aperture center value.
Skew	This option is enabled by default and cannot be disabled.
Aperture	Displays the aperture.
Guide Grid	Displays the guide grid.
Large Font	Sets the font size to large.
Aper	Sets the line size and color for the aperture.
Time Unit	Sets the time unit to **Auto**, **ns**, or **ps**.

Using the Setup/Hold Results Viewer

To view the Setup/Hold Results Viewer:

1. Click the **Detailed Results** button in the Automatic Eye Measurement window.

   The Setup/Hold Results Viewer window opens with the setup and hold results displayed.

2. Click one of the following display category types to display the results you want to see:

   - **Setup/Hold**
   - **Setup+Hold**
   - **Setup**
   - **Hold**

   If you use a derate table, the margin, measured, and required values are reported, and the two corresponding rows that have the minimum setup and minimum hold margin are highlighted in yellow.

   If you do not use a derate table, the minimum setup or hold times are reported, and the two corresponding rows that have the minimum setup or hold times are highlighted in yellow.

3. (Optional) Click the **Data Slew Only** check box to limit the display to data slew values only.

The worst slew values are reported at the bottom of the table. Exported results are also limited when this option is enabled.

4. (Optional) Click the **Export** button to export the selected results to a *.csv file.

Clearing Eye Measurements

Measurement results are displayed on top of the eye diagrams. Right-click an eye diagram, and choose **Clear Measure** from the menu that opens to clear the measure results.

Adding User-Defined Masks

A mask is an abstract reference overlaid in eye diagrams to qualify signals timing integrity. Two built-in masks are provided: a standard mask for common near-end timing specifications and the IEEE802.3 mask. You can add your own mask definitions using mask files as well.

Mask File Syntax

A mask file can contain one or multiple mask definitions. A mask consists of multiple segments that are lines plotted between adjacent points from a node list. Each mask definition is organized in the following format:

```
mask "maskname" direct|normalized
"nodename", X, Y
<newpath>
"nodename", X, Y
...
```

The `mask` keyword starts a new mask definition, which must be followed by a quoted mask name and then a type keyword (`direct` or `normalized`) that specifies the mask type. The words `mask`, `direct`, `normalized`, and `newpath` are reserved keywords.

Each node in the list consists of a quoted node name, the X value of the node, and the Y value of the node. For a direct type mask, `X` is the X value into an eye window, and `Y` is the Y value of the node. For a normalized type mask, the X and Y values denote a node position as fractions into the eye window. X=0 (Y=0) is on the left (bottom) window edge and X=1 (Y=1) is on the right (top) window edge.

A `newpath` keyword breaks line strokes and starts a new drawing stroke.

Incorporating Mask Files

The `SW_SX_MASKFILE` environment variable is used as a pointer to the user mask files.

To incorporate a mask file:

- On UNIX Platforms

  ```
  setenv SW_SX_MASKFILE file1:file2:filen
  ```

  `file1`…`filen` are the path to the mask files. A column (:) is placed between file paths.

- On Windows 95/98 Platforms

  Open a MSDOS window and edit AUTOEXEC.BAT in the top directory of your boot hard-drive (C:\, for example). Add the following line to the AUTOEXEC.BAT file:

  ```
  SET SW_SX_MASKFILE=file1;file2;filen
  ```

  The file names are separated by semi-column (;) instead of column (:) because Windows uses column (:) as a drive delimiter.

- On Windows NT Platforms

  Open the Control Panel and double-click the System icon. Once you see the System Properties dialog box, select the Environment Variables tab. Add the `SW_SX_MASKFILE` environment variable as described in On Windows 95/98 Platforms.

- On Windows XP/2000 Platforms

  Open the Control Panel and double-click the System icon. Once you see the System Properties dialog box, select the Advanced tab and click the **Environment Variables** button. Add the `SW_SX_MASKFILE` environment variable as described in On Windows 95/98 Platforms.

- On Windows Me Platforms

  Click **Start**, choose **Run**, then enter msconfig.exe and click **OK** to run the msconfig.exe program. Click the Environment tab and enter `SW_SX_MASKFILE` as a new environment variable. Make sure that the box before `SW_SX_MASKFILE` is checked to enable the environment variable.

  These mask files are read at initialization time and the mask menu is expanded in the Eye Configuration dialog box with the user-defined masks.

Mask File Example

The following mask example defines an IEEE802.3 mask as shown in Figure 6.

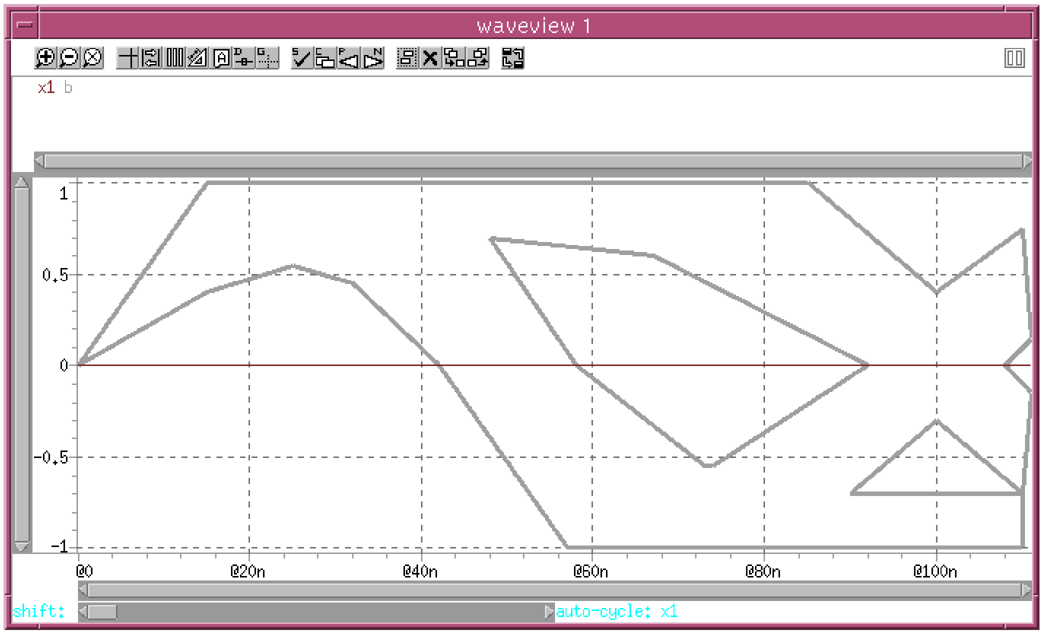


Figure 6 The IEEE802.3 mask

```
mask "IEEE802.3" direct
"A",  0.0n, 0.0
"C", 15.0n, 0.4
"D", 25.0n, 0.55
"E", 32.0n, 0.45
"F", 42.0n, 0.0
"G", 57.0n,-1.0
"T",110.0n,-1.0
"V",110.0n,-0.7
"S",111.0n,-0.15
"R",108.0n, 0.0
"Q",111.0n, 0.15
"P",110.0n, 0.75
"O",100.0n, 0.4
"N", 85.0n, 1.0
"B", 15.0n, 1.0
"A",  0.0n, 0.0
newpath
"H", 48.0n, 0.7
"I", 67.0n, 0.6
"J", 92.0n, 0.0
"K", 74.0n,-0.55
"L", 73.0n,-0.55
"M", 58.0n, 0.0
"H", 48.0n, 0.7
newpath
"U",100.0n,-0.3
"W", 90.0n,-0.7
"V",110.0n,-0.7
"U",100.0n,-0.3
```

Measuring DDR3 Eye Diagrams

The following DDR3 Eye Diagram panel measurements are available:

- Derate-based setup and hold measurements
- tVAC measurements
- Ringback measurements

You can use these measurements during the design phase to create specifications from a behavioral model or for post-simulation verification.

Measuring Derate-Based Setup and Hold Times

This section contains information on the following topics:

- Calculating Address, Command, Data Setup, and Hold Derating Values
- Setting Up DDR3 Measurements
- Viewing Derating Measurement Output

Calculating Address, Command, Data Setup, and Hold Derating Values

The total setup time (tIS) and hold time (tIH) required is calculated by adding the base value of the setup and hold to the derating value. For example:

```
tIS Total= tIS(Base) + tIS
tIH Total= tIH(Base) + tIH
```

The base value of the setup or hold times is taken from the JEDEC standard. The following table shows sample setup or hold base values of several DDR3 specifications:

Symbol	Reference	DDR3-800	DDR3-1066	DDR3-1333	DDR3-1600	DDR3-1866	DDR3-2133	Units
tIS(base) AC175	$V_{IH/L(ac)}$	200	125	65	45	-	-	ps
tIS(base) AC150	$V_{IH/L(ac)}$	350	275	190	170	-	-	ps
tIS(base) AC135	$V_{IH/L(ac)}$	-	-	-	-	65	60	ps
tIS(base) AC125	$V_{IH/L(ac)}$	-	-	-	-	150	135	ps
tIH(base) DC100	$V_{IH/L(dc)}$	275	200	140	120	100	95	ps

The derating values of setup or hold times are calculated from the slew rate of the differential CLK and DQS signals, as well as the address, command, and data signals. In the JEDEC standard, the derating value lookup table corresponds to different slew rates of CLK or strobe and address, command, or data signals.

ΔtIS, ΔtIH derating in [ps] AC/DC based AC175 Threshold -> VIH(ac)=VREF(dc)+175mV, VIL(ac)=VREF(dc)-175mV																	
		CK,CK# Differential Slew Rate															
		4.0 V/ns		3.0 V/ns		2.0 V/ns		1.8 V/ns		1.6 V/ns		1.4 V/ns		1.2 V/ns		1.0 V/ns	
		ΔtIS	ΔtIH	ΔtIS	ΔtIH	ΔtIS	ΔtIH	ΔtIS	ΔtIH	ΔtIS	ΔtIH	ΔtIS	ΔtIH	ΔtIS	ΔtIH	ΔtIS	ΔtIH
CMD/ ADD Slew rate V/ns	**2.0**	88	50	88	50	88	50	96	58	104	66	112	74	120	84	128	100
	1.5	59	34	59	34	59	34	67	42	75	50	83	58	91	68	99	84
	1.0	0	0	0	0	0	0	8	8	16	16	24	24	32	34	40	50
	0.9	-2	-4	-2	-4	-2	-4	6	4	14	12	22	20	30	30	38	46
	0.8	-6	-10	-6	-10	-6	-10	2	-2	10	6	18	14	26	24	34	40
	0.7	-11	-16	-11	-16	-11	-16	-3	-8	5	0	13	8	21	18	29	34
	0.6	-17	-26	-17	-26	-17	-26	-9	-18	-1	-10	7	-2	15	8	23	24
	0.5	-35	-40	-35	-40	-35	-40	-27	-32	-19	-24	-11	-16	-2	-6	5	10
	0.4	-62	-60	-62	-60	-62	-60	-54	-52	-46	-44	-38	-36	-30	-26	-22	-10

Setting Up DDR3 Measurements

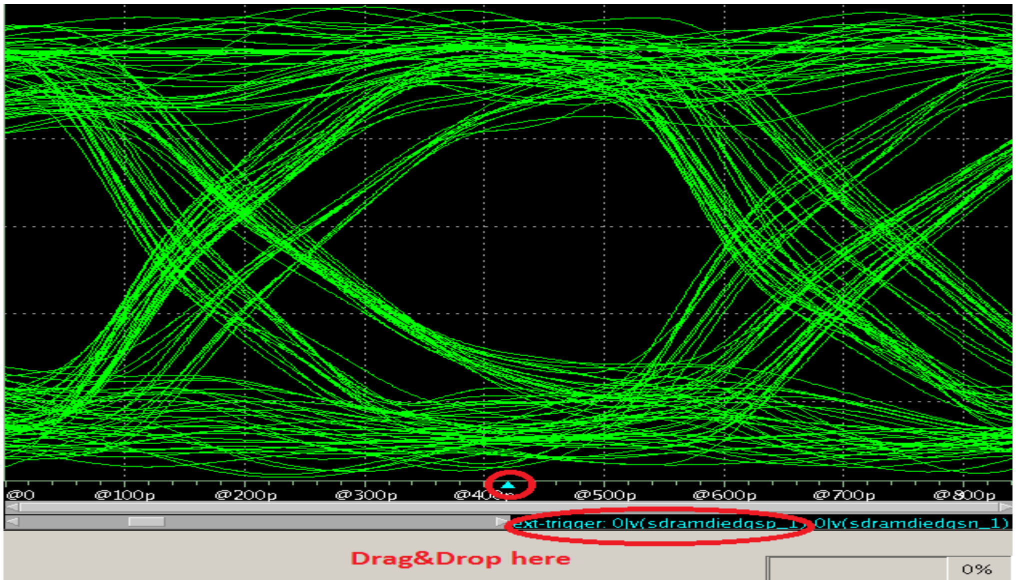
To set up a DDR3 measurement:

1. Load a “derating table” by setting the following environment variable before invoking the Custom WaveView tool:

   ```
   setenv CX_DRTAB_FILE <derating_table_file>
   ```

   This file contains the base values of setup andhold, as well aslookup tables for the derating valuesthat are from different DDR3 standards (DDR3-800/ 1066/1333/1600/1866/2133).
2. Launch the Custom WaveView tool.
3. Plot the eye diagram for the address, command, ordata signal and specify the CLK/DQS signal as the external trigger for the eye.

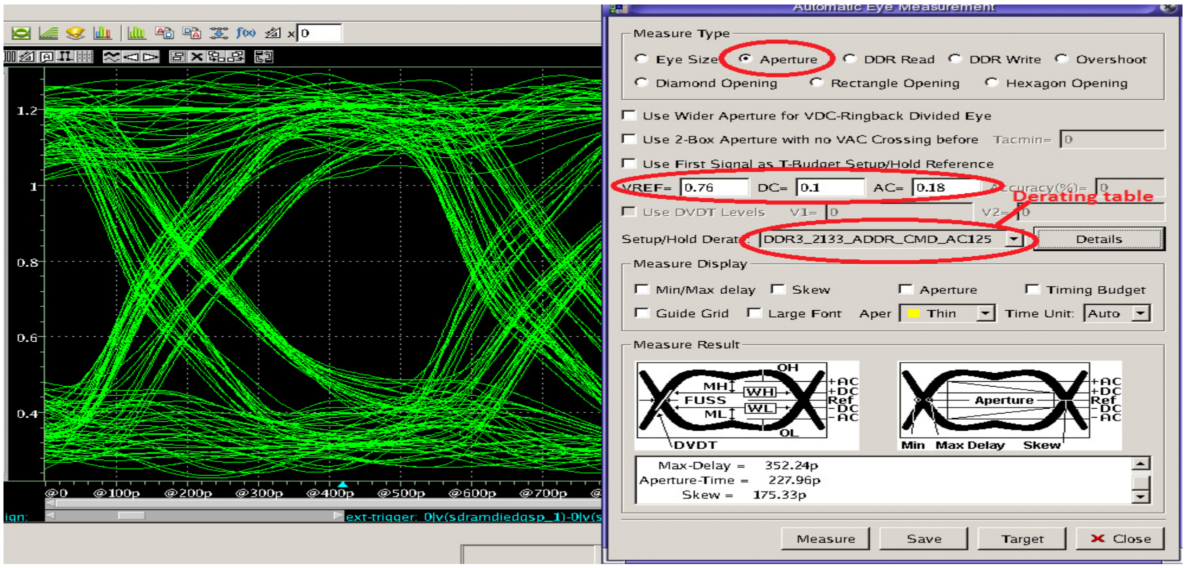
To create an eye diagram with external trigger, plot the eye diagram of the signal, then drag and drop the external trigger signal in to the lower right corner of the eye diagram panel.



4. Ensure the blue triangle (trigger Point) on the x-axis is at the center of the eye.
5. Right-click on the plotted eye diagram, and choose **Eye Diagram** > **Measure Eye** > **Aperture** from the menu that opens.

   The Automatic Eye Measurement window opens.

6. Specify theVref, DC, and AC values, and chooseaderatingtable from the Setup/HoldDerating menu.



7. Click **Measure**.

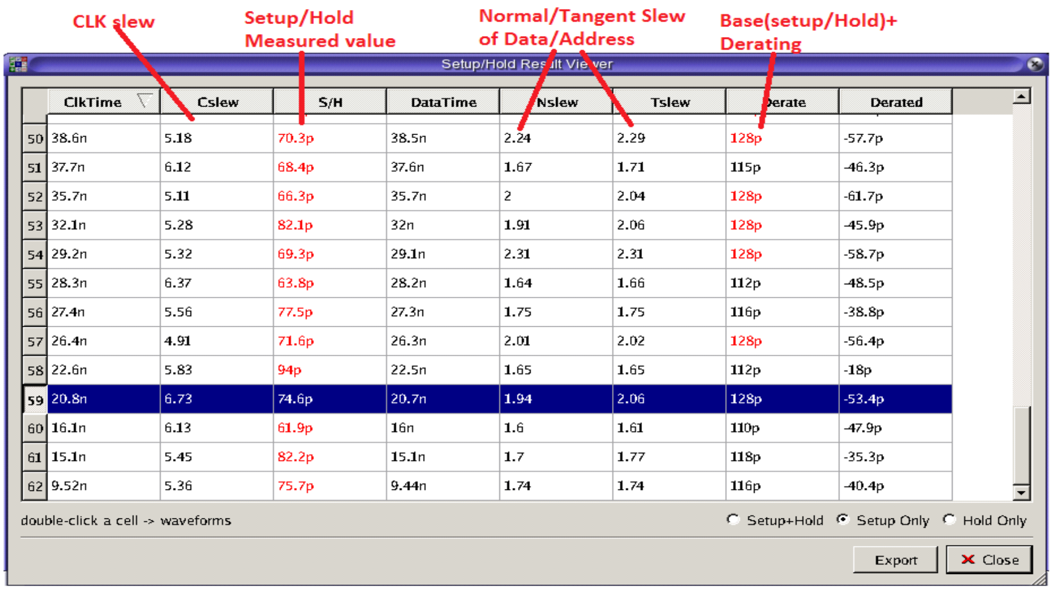
   The measurement is performed. Click **Details** to display the measurement report.

Viewing Derating Measurement Output

Two kinds of measurement results are calculated:

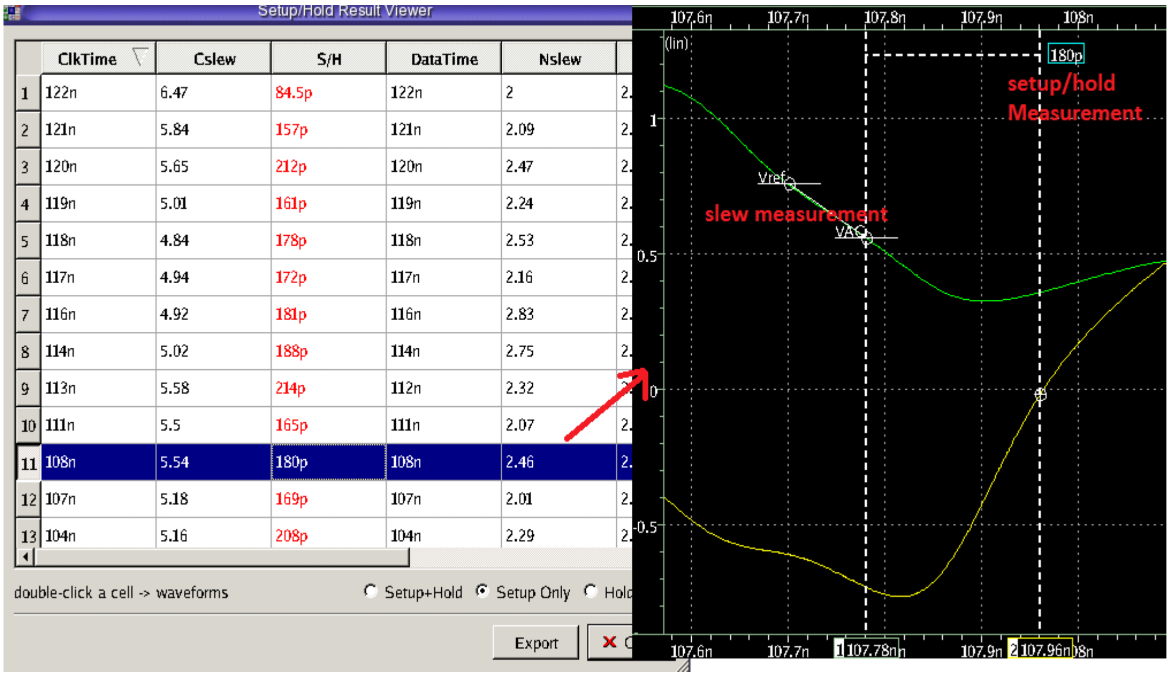
- The required setup or hold values, which are calculated by adding the base value to the derating value. The derating values are obtained after measuring the slew values.
- The actual setup or hold values from the waveform.

If the actual measured value is less than the required value, then it is a setup or hold violation. The output report displays the slew values of signals, as well as the required and actual measured values of setup or hold.



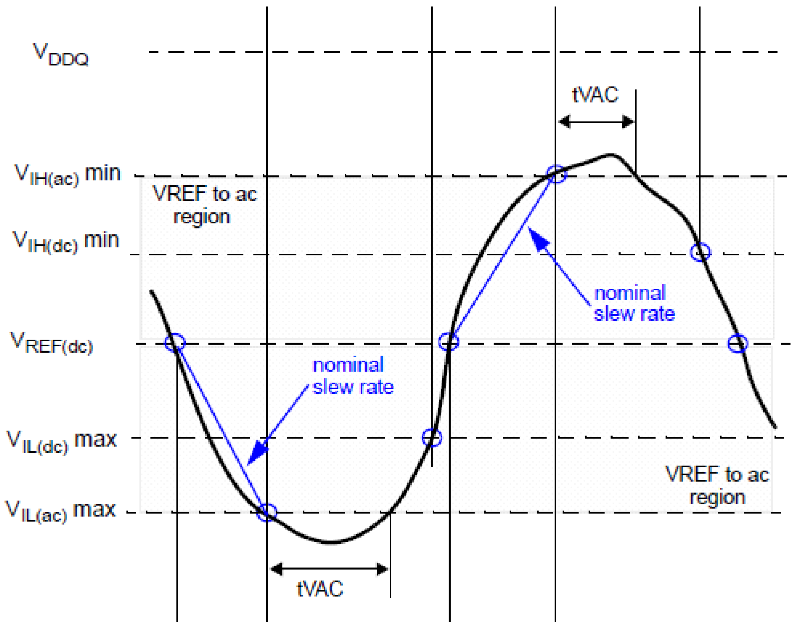
	ClkTime	Cslew	S/H	DataTime	Nslew	Tslew	Derate	Derated
50	38.6n	5.18	70.3p	38.5n	2.24	2.29	128p	-57.7p
51	37.7n	6.12	68.4p	37.6n	1.67	1.71	115p	-46.3p
52	35.7n	5.11	66.3p	35.7n	2	2.04	128p	-61.7p
53	32.1n	5.28	82.1p	32n	1.91	2.06	128p	-45.9p
54	29.2n	5.32	69.3p	29.1n	2.31	2.31	128p	-58.7p
55	28.3n	6.37	63.8p	28.2n	1.64	1.66	112p	-48.5p
56	27.4n	5.56	77.5p	27.3n	1.75	1.75	116p	-38.8p
57	26.4n	4.91	71.6p	26.3n	2.01	2.02	128p	-56.4p
58	22.6n	5.83	94p	22.5n	1.65	1.65	112p	-18p
59	20.8n	6.73	74.6p	20.7n	1.94	2.06	128p	-53.4p
60	16.1n	6.13	61.9p	16n	1.6	1.61	110p	-47.9p
61	15.1n	5.45	82.2p	15.1n	1.7	1.77	118p	-35.3p
62	9.52n	5.36	75.7p	9.44n	1.74	1.74	116p	-40.4p

When you double-click a row, the measurements are displayed on a transient signal, which you can use for debugging the setup and hold violations. The transient portion of the signal that created the violation is displayed. The following figure shows the slew measurements and setup or hold measurements:



Using the tVAC Measurement

tVAC is the required time above Vih (ac) or Vil (ac) for a valid transition. For a valid transition to occur, a signal has to remain above or below an AC threshold for a period of time. This time span is defined in the JEDEC standard. The tVAC constraint is slew-rate dependent similar to the setup or hold derating.



Setting Up tVAC Measurements

The setup for a tVAC measurement is similar to setup or hold derating. To set up a tVAC measurement:

1. Before starting the Custom WaveView tool, enter the following line to load a file that contains the value of tVAC:

   ```
   setenv CX_DRTAB_FILE <tVAC_value_file>
   ```

   The tVAC values can specify inside derating table file along with setup/hold derate values.

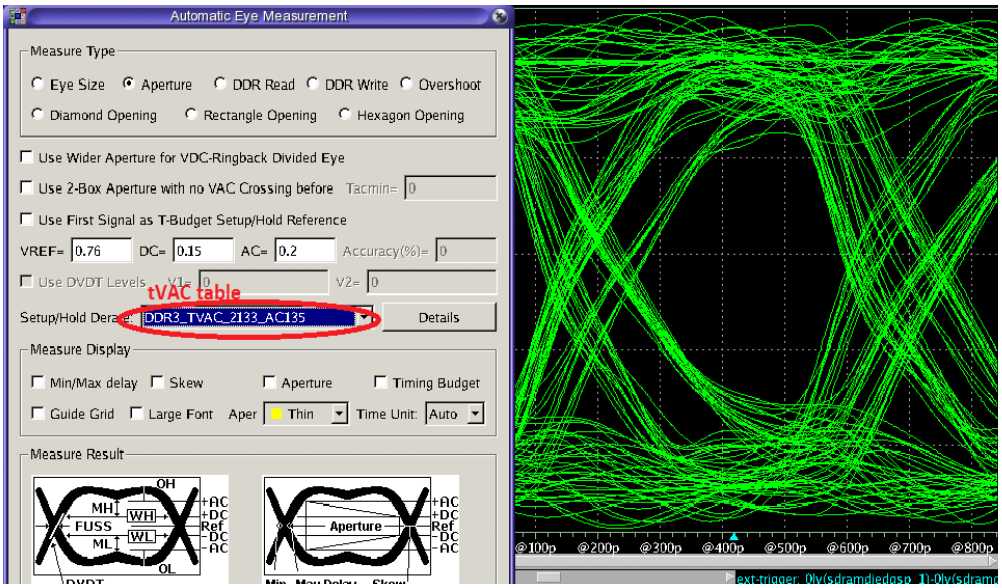
   The measurement is performed after plotting the eye diagram with external trigger signal as in the case of setup or hold derating.

2. Open the Custom WaveView tool.
3. Right-click the eye diagram, and choose **Measure Eye** from the menu that opens.

   The Automatic Eye Measurement window opens.

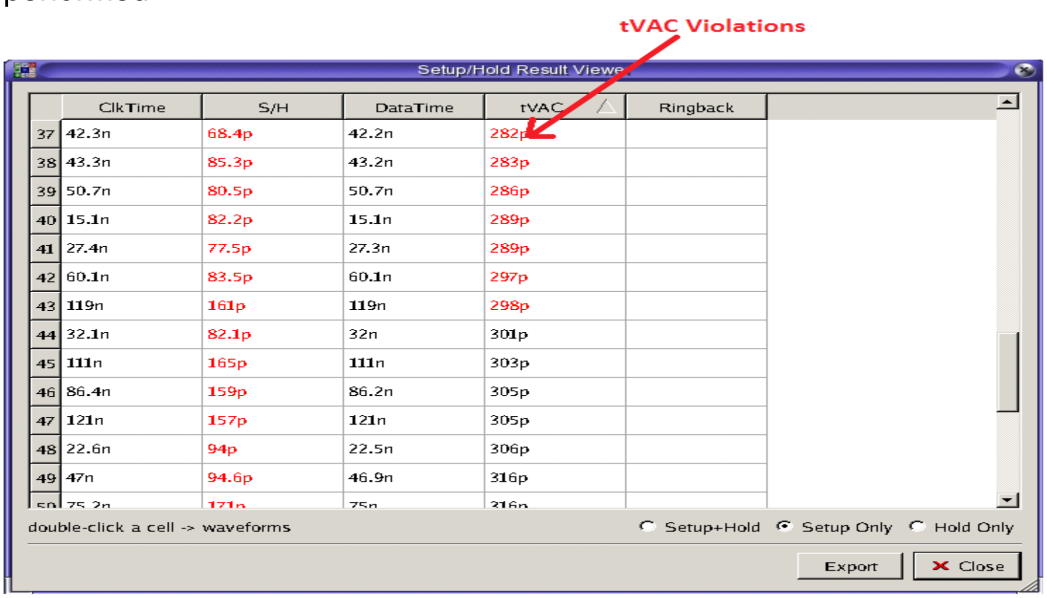
4. Select atVAC table from the Setup/Hold Derate menu.
5. Click **Measure**.

The eye diagram is measured using the tVAC value.

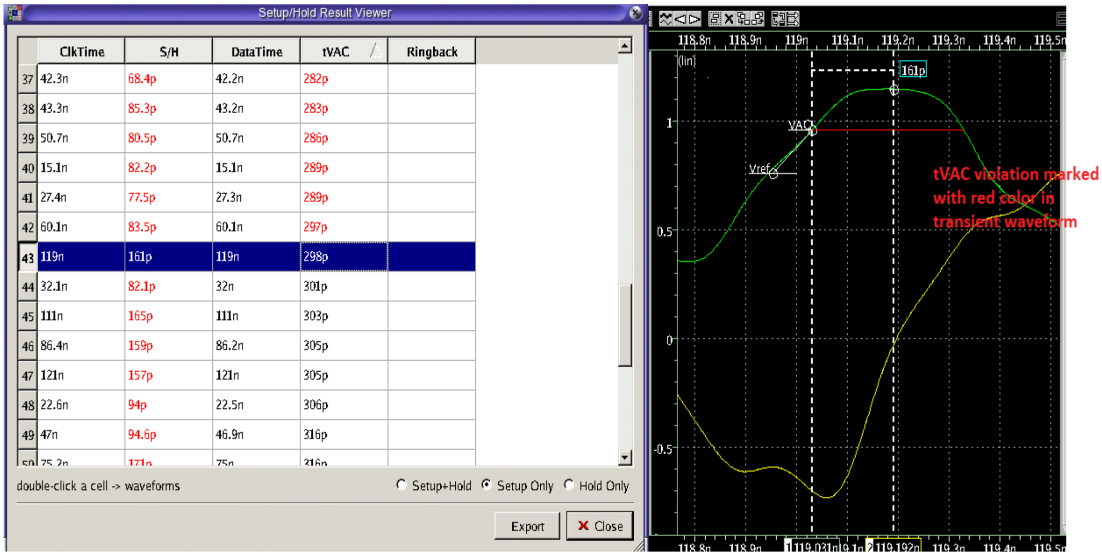


Viewing tVAC Measurement Output

If the measured tVACvalues are smaller than the required value, violations are flagged red in the report that is created after the tVAC measurement is performed.

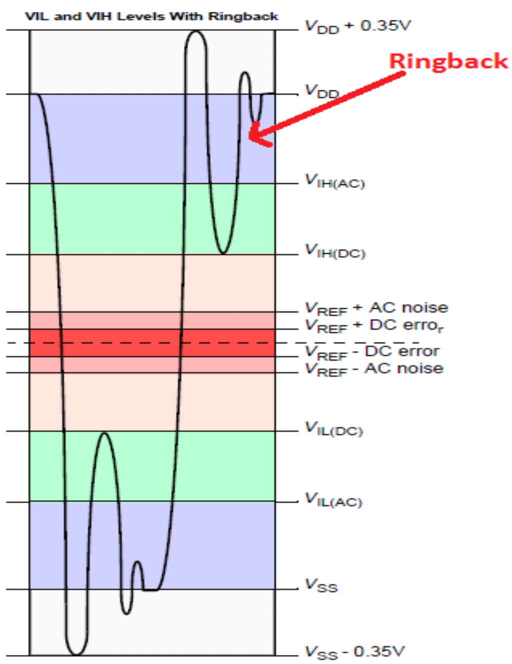


You can click on any of the rows in the report that shows the transient signal and tVAC measurement. In the following figure, the tVAC violations are displayed on a transient waveform:



Measuring Ringback

The ringback measurement calculates how far the top side of a waveform dips below the AC threshold while not crossing the DC threshold. If the waveform moves below the DC threshold, then a ringback violation occurs.



The ringback measurement setup is similar to the tVAC measurement setup in that you also need to provide a tVAC table. The ringback is measured when a signal goes above AC threshold and then goes below the AC threshold values.

Viewing Ringback Measurement Output

Ringback results can occur with or without violations.The output report displays the ringback voltage value. A ringback value without a violation is a signal that goes above the AC threshold, then goes below AC threshold, and finally stays within the AC and DC thresholds. A ringback value with a violationis a signal that goes above the AC threshold and then goes below the DC threshold.

Ringback violations are highlighted in red in the report. When you select a row in the report, the ringback measurement is displayed on a transient signal. The following figure shows the ringback violation displayed on a transient signal in magenta:

Sample Derating Table File

A derating table file includes the derating lookup table values for different DDR3 standards, as well as any tVAC required values.The following file shows a sample derating table:

```
#table DDR3_800_ADDR_CMD_AC175
#base_setup 200
#base_hold 275
#clock_slew 4 3 2 1.8 1.6 1.4 1.2 1
2     88  50  88  50  88  50  96  58 104  66 112  74 120  84 128 100
1.5   59  34  59  34  59  34  67  42  75  50  83  58  91  68  99  84
1.0    0   0   0   0   0   0   8   8  16  16  24  24  32  34  40  50
0.9   -2  -4  -2  -4  -2  -4   6   4  14  12  22  20  30  30  38  46
0.8   -6 -10  -6 -10  -6 -10   2  -2  10   6  18  14  26  24  34  40
0.7  -11 -16 -11 -16 -11 -16  -3  -8   5   0  13   8  21  18  29  34
0.6  -17 -26 -17 -26 -17 -26  -9 -18  -1 -10   7  -2  15   8  23  24
0.5  -35 -40 -35 -40 -35 -40 -27 -32 -19 -24 -11 -16  -2  -6   5  10
0.4  -62 -60 -62 -60 -62 -60 -54 -52 -46 -44 -38 -36 -30 -26 -22 -10

#table DDR3_1066_ADDR_CMD_AC175
#base_setup 125
#base_hold 200
#use DDR3_800_ADDR_CMD_AC175

#table DDR3_1333_ADDR_CMD_AC175
#base_setup 65
#base_hold 140
#use DDR3_800_ADDR_CMD_AC175
#use_tvac DDR3_1333_AC150

#table DDR3_1600_ADDR_CMD_AC175
#base_setup 45
#base_hold 120
#use DDR3_800_ADDR_CMD_AC175
#use_tvac DDR3_1600_AC150
```

Header section shows the DDR3 standard, base value of setup/hold, slew values

Derating look up table values for different slew values

Other standards sharing the same look up table and tVAC table

Table for required tVAC values

The default units of derate table values are ps for time and V/ns for slew.

Displaying the Envelope Center

To display the envelope center of an eye diagram, right-click an eye diagram, and choose **Show Envelope Center** from the menu that opens.

Once selected, the center trace is displayed along with the eye diagram. To disable this option, choose it again.

Measuring DDR4 Eye Diagrams

Note: A CustomExplorer license is required when taking DDR4 measurements in the Custom WaveView tool.

The following DDR4 measurements are supported in the Custom WaveView tool:

- Vcent measurements from a Vref sweep
- Vcent_Dq (pin avg)
- TdiVW
- VIHL_AC
- TdiPW
- Tr&Tf based on VdIVW_Total
- SRIN_dIVW based on VdIVW_Total
- TdIVW & VdIVW margins
- Aperture alignment for LPDDR4

Please refer the *JEDEC Manual* (JEDEC standard no 79-4) for the details of these measurements.

This section contains information on the following topics:

- Taking DDR4 Measurements
- Taking TdIVW and VdIVW Margin Measurements
- Aligning the Aperture for LPDDR4 Measurements
- Shifting the Trigger Clock Delay and Delay Clock to the Aperture Center
- Sample DDR4 Specification File

Taking DDR4 Measurements

This section explains how to take the following DDR4 measurements:

- Taking Vcent Measurements
- Taking TdIVW_Total and VIHL_AC Measurements
- Taking TdIPW, Tr, Tf, and SRIN_diVW Measurements

Taking Vcent Measurements

When a Vcent measurement is performed, the Vref value that corresponds to widest eye opening is calculated. If multiple eye diagrams exist on a single panel, the Vcent value of each eye is measured, and the average value is calcualted to get the Vcent_DQ pin average.

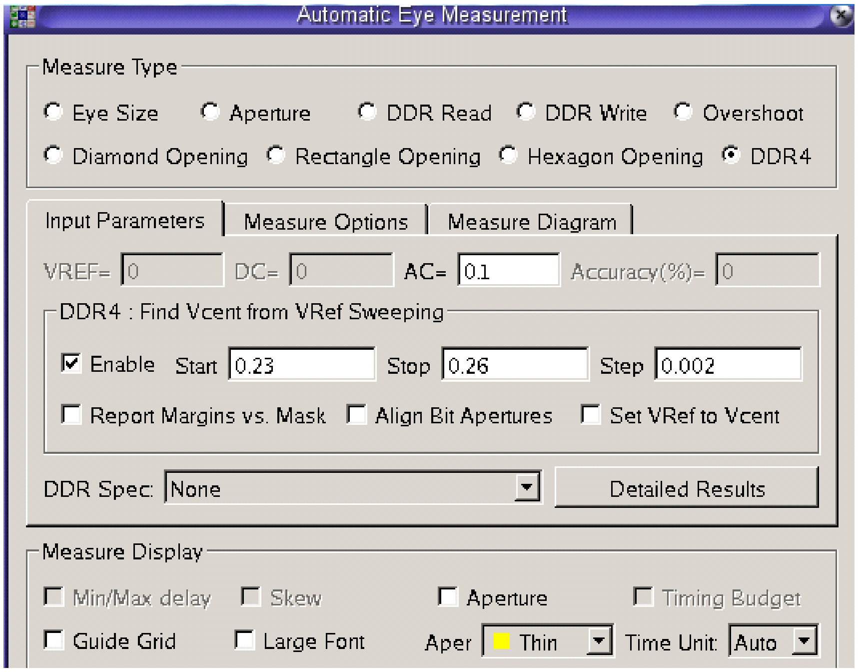
To take a Vcent measurement:

1. Right-click the eye diagram, and choose **Measure Eye** from the menu that opens.

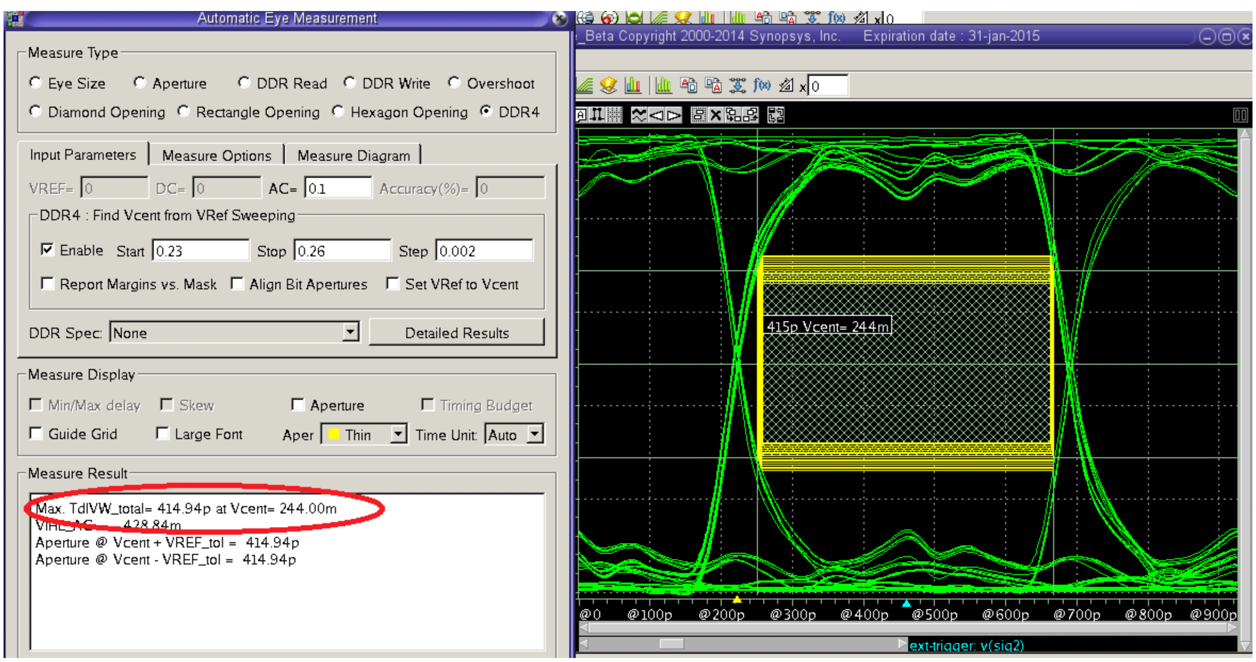
   The Automatic Eye Measurement window opens.
2. Click the **DDR4** measurement.
3. Click the Input Parameters tab.
4. Click the **Enable** check box to enable a Vref sweep.

   The Vcent_DQ pin average for all the bits is calculated when this option is enabled.

5. Enter an AC value (VdIVW_Total/2), as well as the Start, Stop, and Step values.



6. Click the **Measure** button.



The values for Vcent are calculated and displayed in the Measure Result field, as well as on the eye diagram.

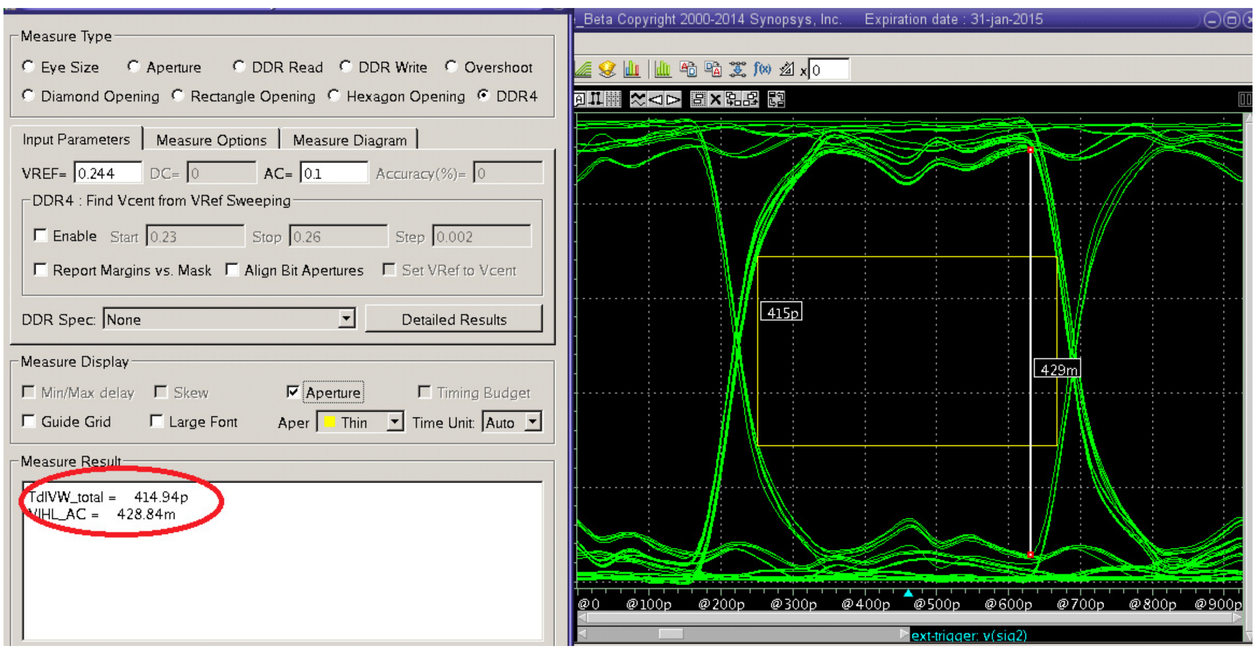
Taking TdIVW_Total and VIHL_AC Measurements

To take TdIVW_Total and VIHL_AC measurements:

1. Right-click the eye diagram, and choose **Measure Eye** from the menu that opens.

   The Automatic Eye Measurement window opens.

2. Click the **DDR4** measurement.
3. Click the Input Parameters tab.
4. Enter values for Vref and AC, and click the **Measure** button.



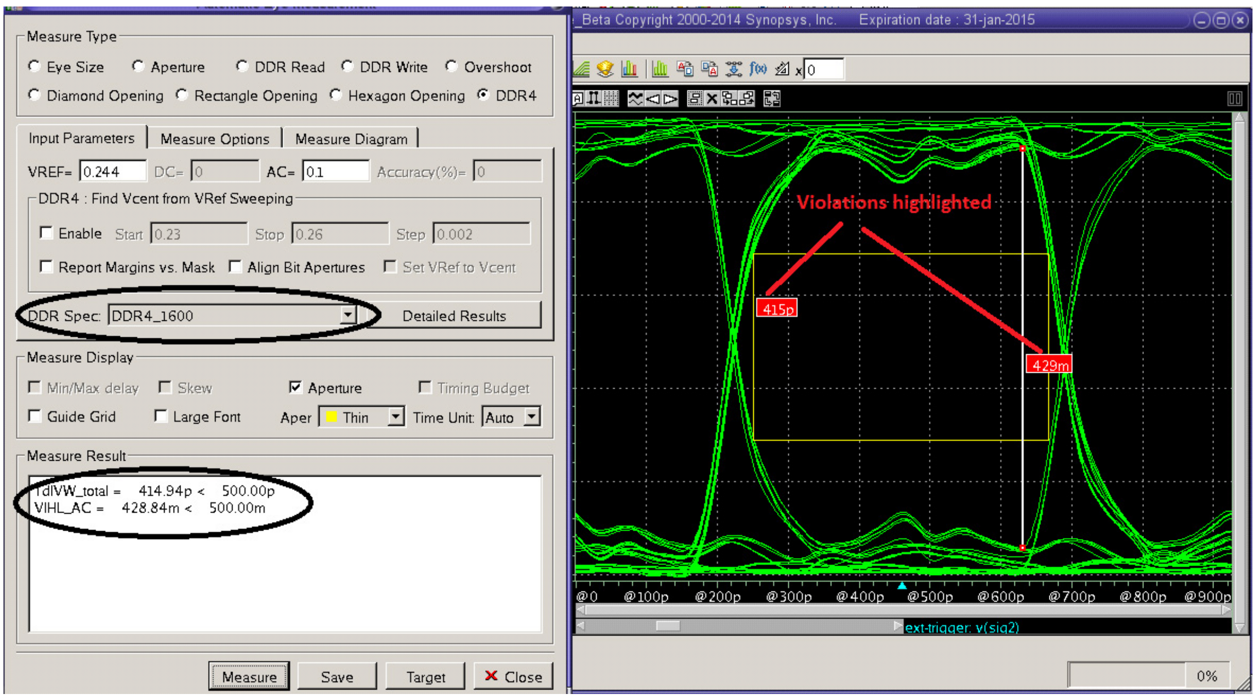
The values for TdIVW_Total and VIHL_AC are calculated and displayed in the Measure Result field, as well as on the eye diagram.

If you want to specify a target value for TdIVW_Total and VIHL_AC in a DDR4 specification file and load it into the Custom WaveView tool, enter the `setenv CX_DRTAB_FILE <filename>` environment variable at the command line before invoking the Custom WaveView tool. See the Sample DDR4 Specification File section for information on how to set up a DDR4 specification file.

Include the following commands in your DDR4 specification file to set these values:

`#min_aperture <value>` or `#min_aperture_ui <value>`
`#min_vihl_ac <value>`

If a measured value does not meet the target, the violations are highlighted in red.



Taking TdIPW, Tr, Tf, and SRIN_diVW Measurements

To take TdIPW, Tr, Tf, and SRIN_diVW measurements:

1. Right-click the eye diagram, and choose **Measure Eye** from the menu that opens.

   The Automatic Eye Measurement window opens.

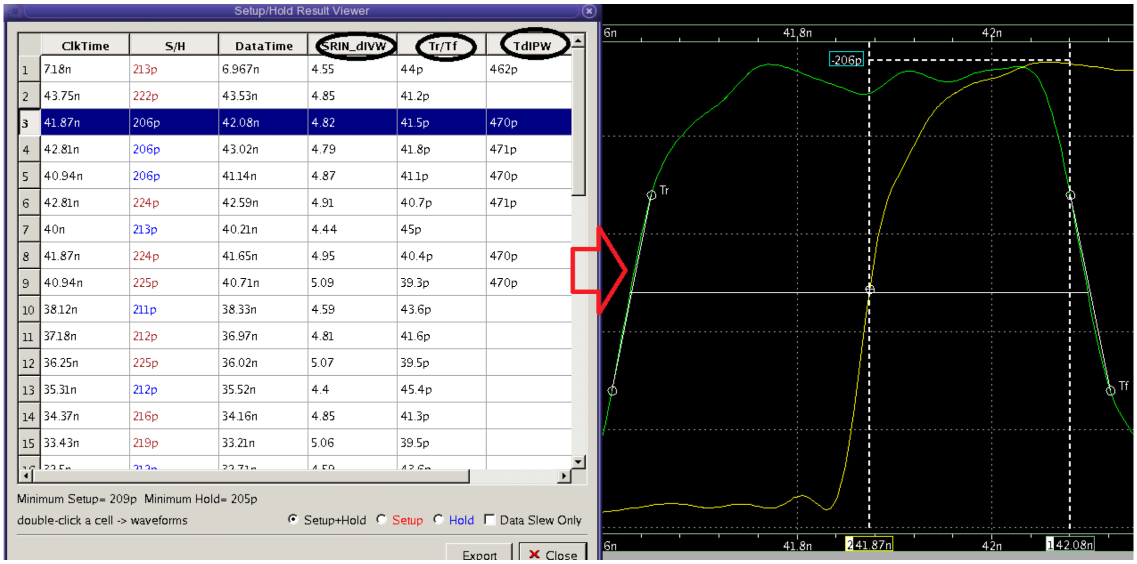
2. Click the **DDR4** measurement.

3. Click the Input Parameters tab.

4. Enter values for Vref and AC, and click the **Measure** button.

   The values for TdIPW, Tr, Tf, and SRIN_diVW are calculated and displayed in the Measure Result field, as well as on the eye diagram.

5. Click the **Detailed Results** button.



The Setup/Hold Result Viewer window opens with the TdIPW, Tr,Tf, and SRIN_diVW values displayed. You can double-click a row to show the corresponding measurements on a transient signal.

If you want to specify the target value for the TdIPW and slew, you can include the following commands in your DDR4 specification file:

```
#min_dataslew <value>
#min_tdipw_ui <value>
```

Measurements that do not meet the target value are highlighted.

Taking TdIVW and VdIVW Margin Measurements

The following calculations occur when you place a mask inside of an eye and measure the voltage and timing margins:

- The Vcent value of each bit is calculated.
- The Vcent average of all the bits (Vcent_Dq pin_avg) is calculated.
- A mask based on the Vcent_Dq and Triggering point is placed.
- The margin values are measured. The ringback effect is also considered during this measurement.

To measure the margin:

1. Right-click the eye diagram, and choose **Measure Eye** from the menu that opens.

   The Automatic Eye Measurement window opens.

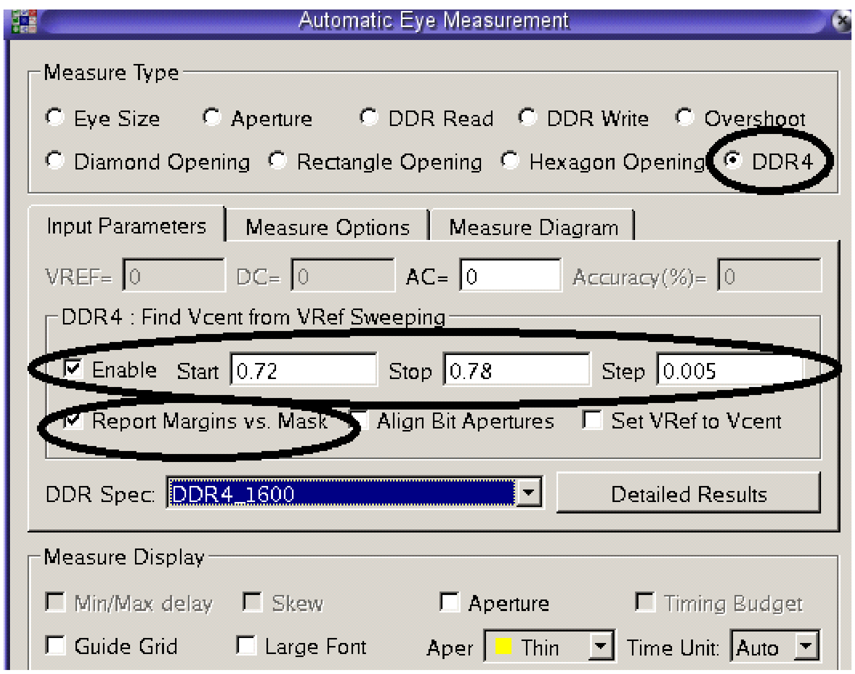
2. Click the **DDR4** measurement.
3. Click the Input Parameters tab.
4. Click the **Enable** check box to enable a Vref sweep.

   The Vcent_Dq pin average for all the bits is calculated when this option is enabled.

5. Enter the Start, Stop, and Step values.

   If you enter the Vref value directly, the measurement works without a Vref sweep.

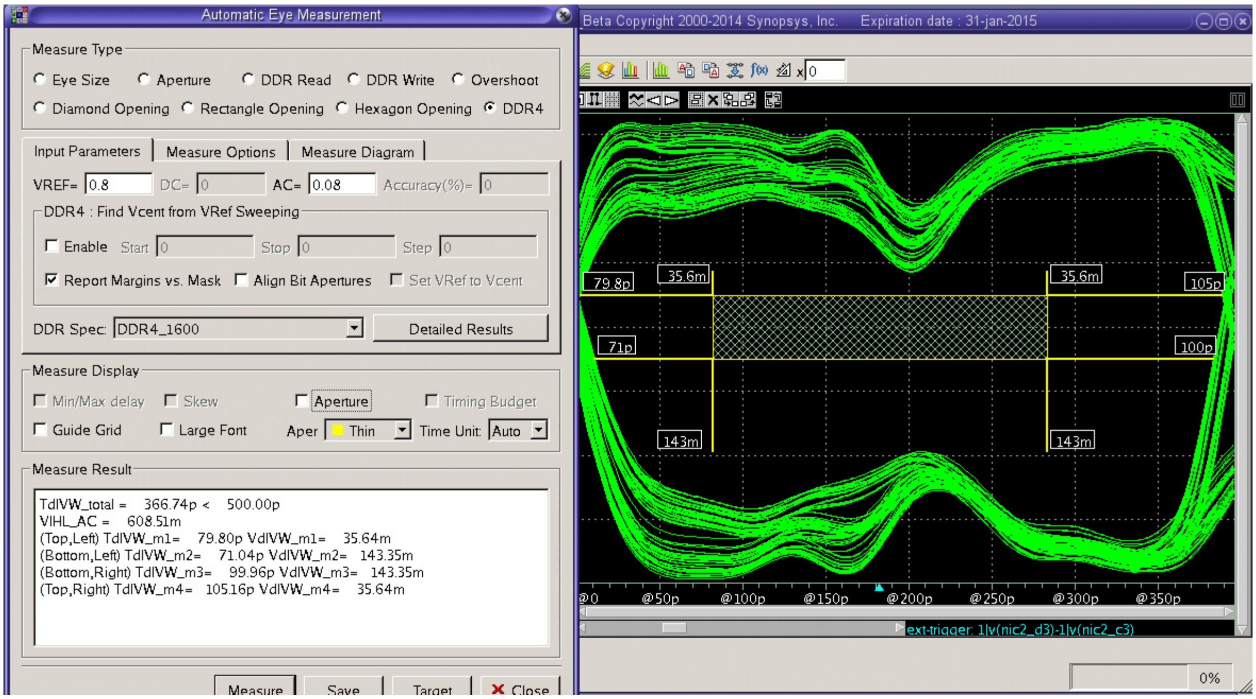
6. Click the **Report Margins vs Mask** check box.
7. Choose a DDR4 spec from the **DDR4 Spec** menu that contains mask specifications.



Use the following commands in your DDR4 specification file to set the width and height of mask:

```
#mask_tdivw_ui <width value>
#mask_vdivw <height value>
```

8. Click the **Measure** button.



The mask margin is calculated and displayed in the Measure Result field, as well as on the eye diagram.

Aligning the Aperture for LPDDR4 Measurements

Aperture alignment measures the mask offset values of each bit. You can plot the eye diagram of all the bits in a single panel and trigger it with a differential clock.

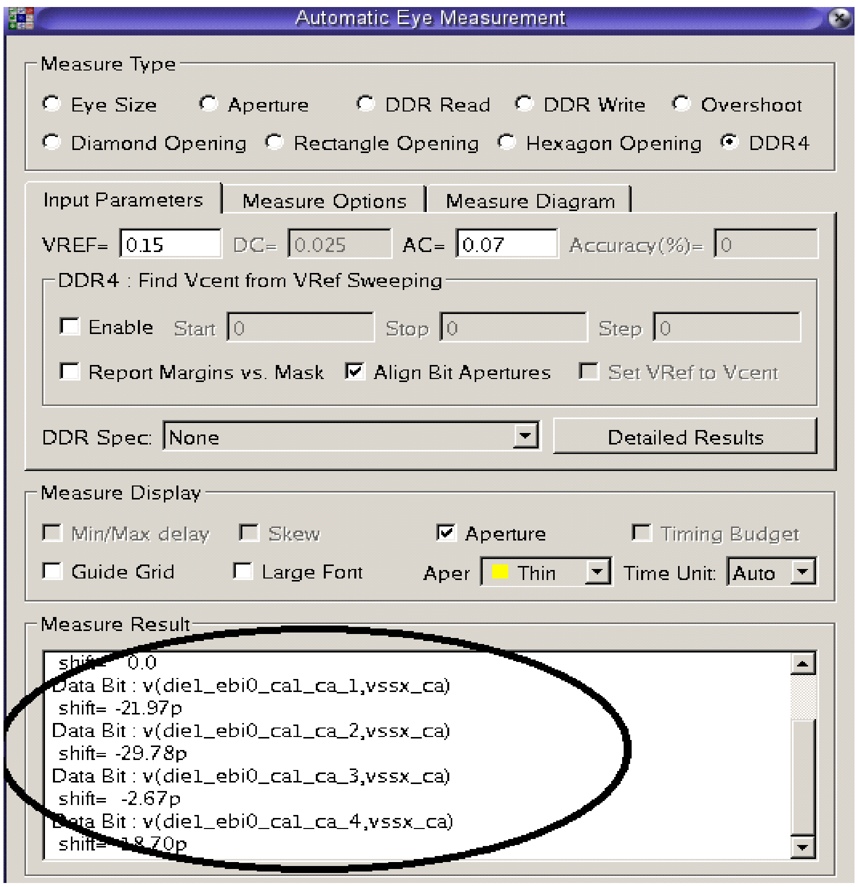
The first eye is used as the reference, and then the center point of the eye crossing is calculated. All other eye diagram center points are shifted so they overlap. Both the shift value of each bit and the aperture value are reported.

To align apertures:

1. Plot the eye diagrams of all the bits in a single panel, and use the same signal for triggering the eye.
2. Right-click the eye diagram, and choose **Measure Eye** from the menu that opens.

The Automatic Eye Measurement window opens.

3. Click the **DDR4** measurement.
4. Click the Input Parameters tab, and enter Vref and AC values for the eye.
5. Click the **Align Bit Apertures** option.
6. Click the **Measure** button.



The shifted values of each bit are calculated and displayed in the Measure Result field.

Shifting the Trigger Clock Delay and Delay Clock to the Aperture Center

Sometimes the simulation data might not have the trigger signal well centered, and you might want to center the trigger signal during the DDR4 measurement. The **Trigger Clock Delay** option specifies the delay value for the triggering signal, and the **Delay Clock to Aperture Center** option aligns the trigger signal to the center of the eye.

Sample DDR4 Specification File

You can specify the target values for different measurements in a DDR4 specification file. For example:

```
#ddr4_spec DDR4_2133
#ddr_ui 805p
#min_aperture 500p
#min_aperture_ui 0.5
#min_dataslew 0.6
#min_tdipw_ui 0.58
#min_vihl_ac 500m
#mask_tdivw_ui 0.25
#mask_vdivw 0.1
```

- The DDR standard
- The UI value
- The TdIVW_Total target value
- The TdIVW_Total target value based on UI. The target= min_aperture_ui*ddr_ui. If both min_aperture and min_aperture_ui are specified, min_aperture overrides min_aperture_ui. Specify the ddr_ui with this.
- The target value of slew. Unit V/ns
- The TdIPW target value. The target value= 0.58*805p(ui)
- The VIHL_AC target value
- The width of the mask. Width=0.25*ddr_ui
- The height of the mask

Creating Logic Panels

A logic panel displays signals using digital timing diagram. It is permitted only in vertical stack waveviews. Logic panels accept signals that are represented in (time, logic) pairs. Various logic states including logic strength levels are supported. Zoom operation is supported along the x-axis direction.

Logic panels cannot be manually added from the application top menu. They are automatically created when logic signals are dropped into a vertical waveview.

Current vector width (number of bits) and radix setting of a logic panel are indicated in the left column of each panel (B: binary, O: octal, D: decimal, H: Hex-decimal, A: ASCII). The width digits are red color-coded for bus signals formed from originally separated bits, or yellow color-coded for a single bit extracted from an originally grouped bus signal.

Logic panels can be arbitrarily mixed with linear X-Y panels in a vertical stack waveview.

Mnemonic Mapping for Bus Signals

Bus values can be mapped to pre-defined text strings. You can provide a translation table in an external file.

Syntax:

```
# comment lines start with '#'
value1 name1
value2 name2
...
```

The values are the bus values to be mapped.

The following formats are supported:

- 2400
- 0xAF32
- 'hAF32
- 'b1001_0100_0101
- 'd2400
- 'b110110111100
- 'o371

The names are the text strings to be mapped to. A text string must be quoted if the name contains spaces.

You can load the translation table by choosing **Config > Bus Mnemonic Map**. Once a translation table is loaded, choose **Radix > Mnemonic On/Off** from the panel context menu to toggle the mapping settings.

Creating 3-D Sweep Panels

The 3-D Sweep panel displays 2-dimensional sweeping data on a 3-dimension coordinate. It is allowed only in horizontal waveviews. The 3-D Sweep panel accepts signals that are represented in ((x1, x2), value) pairs, where x1 and x2 are the first and second independent sweeping variable, respectively, and value can be either the value of a real signal or the magnitude/phase/real/imaginary part of a complex signal. The viewing angles of the 3-dimension coordinate can be interactively adjusted from the two sliders in the panel. The zoom operation is not supported. Logarithmic scale is supported only along the vertical axis.

Creating 2-D Sweep Panels

The 2-D Sweep panel displays 2-dimensional sweeping data on a flat X-Y plane. It is allowed only in horizontal waveviews. Similar to the 3-D Sweep panel, the 2-D Sweep panel accepts signals that are represented in ((x1, x2), value) pairs. Zoom operation is supported along both axes. Logarithmic scale is also supported along both axes (the x1 direction and the y direction, for example).

Both 2-D and 3-D sweep panels support sweep filters that display the selected sweep traces. Choose **Sweep Line Filter** from the panel context menu, and highlight traces in the Sweep Line Filter dialog box.

Enable the **Display Trace Value** option to display value label next to each sweep trace. If a 2D/3D sweep panel contains more than one signal, the first signal or the first highlighted signal is used as the filter target.

In 2-D sweep plot, the min/max Y-value bounds and the average Y-value of a multi-trace signal can be automatically extracted (for example, waveform envelop from statistical simulation result). To invoke this function, choose **Find Envelope Bound** from the panel context menu of a 2-D sweep panel. All data points are scanned from all traces, and three new waveforms representing the min/max bounds and the average waveform are created.

Displaying Multi-trace Waveforms

A multi-trace signal contains multiple waveform traces. Each trace has a set of associated sweeping parameters (or sub X-variables). For example:

```
Trace 1: Temp=0 Vdd=3.3
Trace 2: Temp=0 Vdd=5
Trace 3: Temp=100 Vdd=3.3
Trace 3: Temp=100 Vdd=5
```

By default, a multi-trace signal is plotted with respect to its main independent sweeping variable (time, for example). In this case, the sweep line filter function in a 2D-sweep panel can used to selectively display some of the traces.

The 2D-sweep panel also supports plotting a multi-trace signal with respect to one of its sweeping parameters. To change the default X-axis of a 2D-sweep panel to a sub X-variable, choose **Change X Variable** from the panel context menu. Drag and drop a sweeping parameter from the Output View browser to the Setting X Variable dialog box. Choose **Sweep Line Filter** from the panel context menu to open the Sub X-Variable Plotting Filter dialog box.

Note: The filter setup is different between using the default X-axis and sub X-Variable X-axis. This is because in the sub X-Variable plot mode, based on the selected sub X-variable, the number of traces might vary depending on the number of unique combinations among other sweeping parameters. The number of traces are automatically detected and sorted in the order of the sub X-Variable values to display waveforms.

A multi-trace signal must have equal number of points in each of its original traces in order to be plotted in the sub X-variable mode.

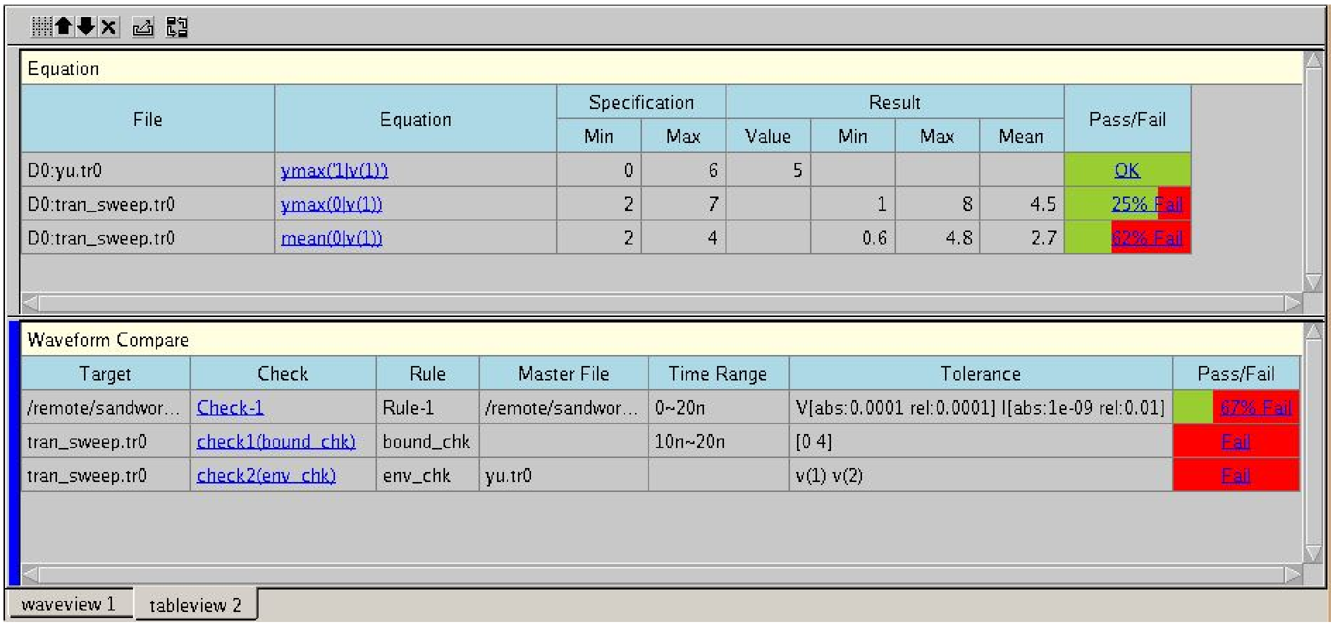
Plotting a multi-trace signal with respect to another multi-trace signal is not supported.

Creating Separator Panels

The Separator panel can be used to assign panel groups. All panels after a separator panel and before the next separator panel are considered a panel group. A group name can be assigned to a separator panel. Right-click a separator panel, and choose **Edit Label** to modify name. Use the left "+"/"-" icon in a separator panel to expand/collapse the panels in a panel group.

Creating Tabular Panels

The Tabular panel displays data such as waveform results and equations in a spreadsheet format. To create a Tabular panel, choose **Panel > New > Tabular** from the menu bar. A new tabular panel is created on the current tableview. If the current view is not a tableview, a message is displayed that asks if you want to create a tableview.



The Tabular panel can display the following types of information in separate table sections:

- equations
- signals
- waveform comparison results

By default, only signals with scalar values are displayed in the tabular window (for example, when you double-click a signal from signal list). If you drag and drop a waveform result to tabular window, the signal is also displayed in the spreadsheet.

Working with All Types of Tabular Data

This section contains the following information on how to sort and group all types of data, as well as how to hide and show table rows.

- Hiding and Showing Rows
- Sorting Tabular Column Data by Ascending or Descending Order
- Filtering Tabular Data

Hiding and Showing Rows

To hide a row in a Tabular window, right-click any row, and choose **Hide Row** from the menu that opens.

To show all rows that are hidden, right-click anywhere in the table, and choose **Unhide All Rows** from the menu that opens.

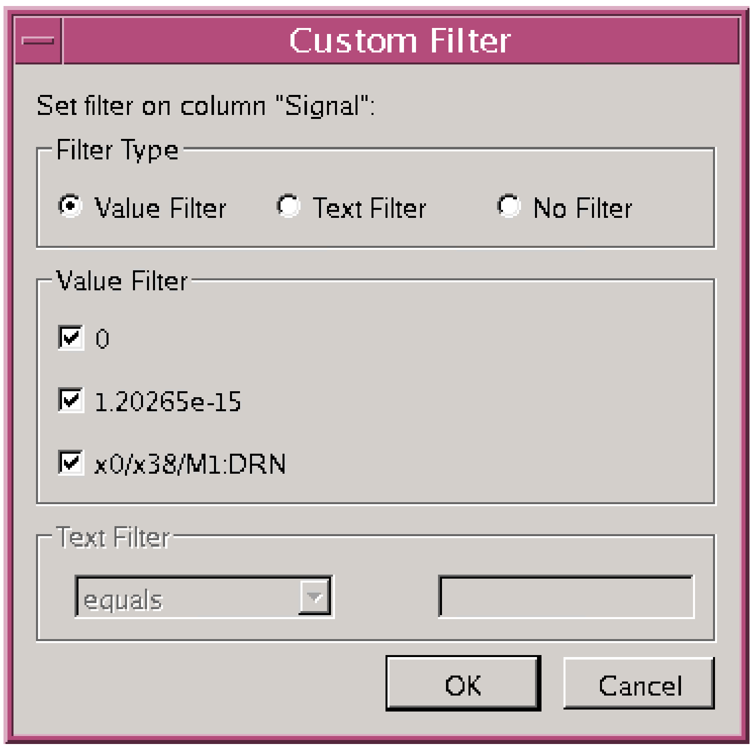
Sorting Tabular Column Data by Ascending or Descending Order

To sort column data by ascending or descending order, right-click a column title in the Tabular panel, and choose **Sort(Asc)** or **Sort(Desc)** to sort the data in ascending or descending order, respectively.

Filtering Tabular Data

To filter tabular data:

1. Right-click a column heading in the Tabular panel that contains the data you want to filter, and choose **Value Filter** from the menu that opens.



The Custom Filter dialog box opens.

2. Choose a Filter type: **Value Filter**, **Text Filter**, or **No Filter**.

If you want to filter data by value, choose **Value Filter**, and move on to the next step.

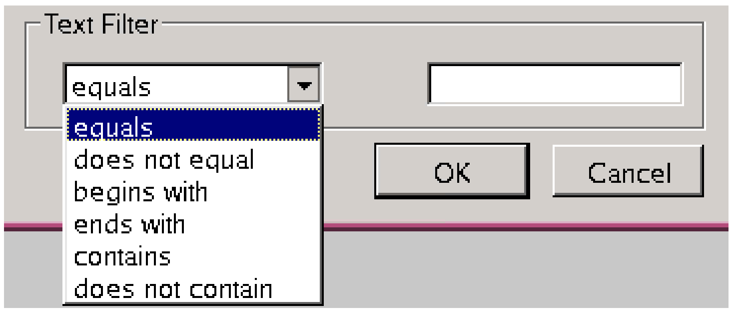
If you want to filter data by a text string choose **Text Filter**, skip to Step 4.

If you choose **No Filter**, skip to Step 6.

3. Click the check boxes next to each value you want to filter from the table.

   Skip to Step 6.

4. Choose a filter qualifier from the menu in the Text Filter section.



   The following qualifiers are available:

   - **equals**
   - **does not equal**
   - **begins with**
   - **ends with**
   - **contains**
   - **does not contain**

5. Enter a text string in the text box next to the filter qualifier menu to include in your filter.

6. Click **OK** to save your filter settings.

   The tabular data is filtered in the chosen column.

Working with Tabular Equation Data

The Equation section of a Tabular panel contains the following equation data:

Column Title	Description
File	File name.
Equation	Equation name (or equation string if no name is specified). If no corner or sweep results are available, the signal waveform is displayed instead of the equation.
Specification	Equation specifications.
Result	Equation results.
Pass/Fail	Pass or fail status. If no specifications exist, pass or fail result are not available.

This section contains information on the following topics:

- Modifying Tabular Equations
- Adding or Removing Tabular Statistics Results
- Exporting Tables

Modifying Tabular Equations

To modify an equation, right-click the name of an equation in the Equation column, and choose **Modify** from the menu that opens.

See Chapter 8, Using the Equation Builder for information on the Equation Builder.

Adding or Removing Tabular Statistics Results

To add or remove statistics in the Equation section of a Tabular panel:

1. Right-click the Result column, and choose **Statistics Setting** from the menu that opens.

   The Statistics Setting dialog box opens.

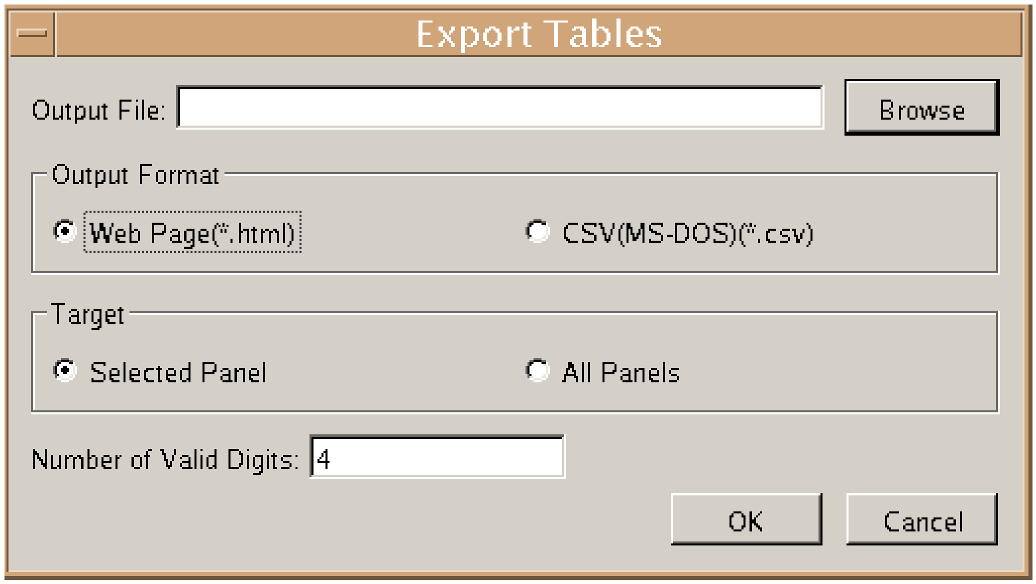
2. Click the check box next to each result type you want to display in the table.
3. Click **OK** to save your changes.

   The chosen result types are displayed in the table.

Exporting Tables

To export a table to an HTML or .csv file:

1. Click the  button in the button tool bar.



   The Export Tables dialog box opens.

2. Enter an Output File name, or click Browse to browse to and select one.
3. Choose an Output Format: **Web Page (*.html)** or **CSV (MS-DOS)(*.csv)**.
4. Choose a Target you want to export: **Selected Panel** or **All Panels**.
5. Enter the number of valid digits in the **Valid Digits** text box.
6. Click **OK** to export your table data.

Working with Tabular Signal Data

The Equation section of a Tabular panel contains the following equation data:

Column Title	Description
File	File name.
Signal	Signal name.
Hierarchy	Hierarchy location of the signal.
Value	Value of the signal or waveform data.
Unit	Unit of measure.

The Signal section of a Tabular panel contains the following signal data:

- File
- Signal
- Hierarchy
- Value
- Unit

This section contains information on the following topics:

- Viewing Waveform Data

Viewing Waveform Data

To view waveform data in the Signal section of a Tabular panel, click the  icon in the Value column. A waveview opens with the waveform data displayed.

Working with Tabular Waveform Comparison Data

The Waveform Compare section of a Tabular panel contains the following waveform comparison data:

Column Title	Description
Target	Target file.
Check	Waveform check.
Rule	Rule name.
Master File	Master file name.
Time Range	Time range values.
Tolerance	Tolerance value.
Pass/Fail	Pass or fail status.

This section contains information on the following topics:

- Viewing Tabular Waveform Check Details
- Viewing Tabular Waveform Check Signal Details

Viewing Tabular Waveform Check Details

To view waveform check details, click the name of a waveform check in the Check column in the Waveform Compare section. A new table opens with Signal and Difference Points/Rate information for that waveform check.

Click the Waveform Compare link to return to the Waveform Compare section, or you can click a signal in the Signal column to display signal details. See Viewing Tabular Waveform Check Signal Details for more information.

Viewing Tabular Waveform Check Signal Details

To view waveform check signal details, click the name of a signal in the waveform check details table (see Viewing Tabular Waveform Check Details for more information). The check signal details table is displayed, which contains X-axis, Master, Value, and Diff. Percentage information. Double-clicking on a row in the details table plots the related signals on a waveview displays a vertical line showing where the difference occurs.

Creating Histogram Panels

To create a Histogram, an empty Histogram panel must created first by choosing **Panel > New > Histogram** from the menu bar. Signals can then be dragged from the Output View browser and dropped into the Histogram. Histogram panels can be placed only in horizontal waveviews.

A histogram plots an individual histogram for each signal in the panel. The average value and standard deviation can be annotated on top of each histogram and the number of bins and data bounds can be reconfigured. The Custom WaveView tool can also calculate the distribution percentages of regions divided by user-defined markers from the histogram mean center.

Configuring Histograms

To configure a histogram, right-click a histogram, and choose **Configure Histogram** from the menu that opens. The Histogram Settings window opens.

The following settings are available:

Setting	Description
Number of Bins	The number of bins to display in the histogram.
Show Average Line	Enables the average bar. The histogram is plotted by default.
Show a Single HIstogram of All Signals	Plots a single histogram of all signals in a panel.
Y Value Normalized by Runs	Normalizes the Y-axis and sets it to the relative frequency (the number of occurrences divided by the number of Monte Carlo runs, for example).
Show Normal Distribution Curve	Displays the normal distribution curve over the histogram.

Setting	Description
Find Distribution	Displays the data distributions for the following specified range in percentage units: ▪ [beginning, mean – lower_marker*sigma] ▪ (mean – lower_marker*sigma, mean] ▪ (mean, mean+upper_marker*sigma] ▪ (mean+upper_marker*sigma, ending]
User-Defined Data Bounds	Specifies the data range to be displayed, which is displayed as the X-axis range in a histogram panel.

Changing the X-axis Variable

By default, signals in X-Y panels are plotted with respect to their internal X variables defined in the original waveform data file. The X variable can be replaced with a signal so that other signals are plotted as functions of the selected signal.

In a vertical waveview, if the waveview contains no logic panel, you can assign a signal as the common X variable so that all signals are plotted against the selected X variable. In a horizontal waveview, the same operation can be applied individually to all selected X-Y panels.

To select a signal as the new X variable:

1. Choose **Axes > X Variable** from the main menu to open the Setting X Variable dialog box.
2. With the dialog box open, right-click a signal in the Output View signal list window, and choose **Use as X Variable**.

   The drag-and-drop operation is also supported from the Output View browser to the Setting X Variable dialog box. The name of the newly selected signal is entered in the read-only X Variable dialog box.
3. Click **Apply** to apply the change to an active vertical waveview or all the selected panels in an active horizontal waveview.
4. To restore to signals' internal X variables, select the **Use Internal X Variable** option in the dialog box and click **Apply**.

The drag-and-drop operation can be also used to set the X variable. Drag a signal from the Output View browser and drop it in the lower X-axis of a stack waveview, or the X-axis of a X-Y panel in a row/column waveview.

Moving, Selecting, and Grouping Panels

Panels in a waveview can be moved, combined, split, and deleted. These actions are invoked from the panel context menu or by choosing **Panel > Actions** from the main menu. The following table summarizes the default bindkeys for panel actions:

Bindkey	Function
B	Move the active cursor backward
C	Copy (Panel, Signals)
D	Data point toggle
F	Move the active cursor forward
G	Grid toggle
H	Horizontal zoom/unzoom
N	Scan forward
P	Scan backward
Q	Quit
S	Signal select/highlight
T	Fit/unfit panel height to full window height
V	Vertical zoom/unzoom
X	Unzoom (reset)
Z	Box zoom/unzoom
Delete or Backspace	Delete panel/signal

Bindkey	Function
Esc	Deselect all panels
Up Arrow	Scroll up a waveview
Down Arrow	Scroll down a waveview

Selecting Panels

Panels can be selected (highlighted) for further actions. To select a panel, click the name or monitor area of a panel. Selected panels appear with a dimmed background. Multiple panels can be Shift-selected or Ctrl-selected.

To select all panels in a waveview, choose **Panel > Select All** from the main menu. To deselect all panels, choose **Panel > Unselect All** from the main menu.

You can also use the Panel Select control button to toggle between the Select-All and the Unselect-All operations.

Moving or Copying Panels

To move or copy a panel, click and hold a name or value area of a panel. Drag the panel to a new location and release the mouse button. If the source panel is selected, all selected panels are moved/copied.

During the dragging process, a drop hint appears to indicate the drop location.

Panels can be moved/copied from a waveview window to another waveview window. Use the preference settings to select between the **Move** and the **Copy** operations. Panels can also be copied and pasted using the corresponding menu entries in the panel context menu.

Deleting Panels

To delete a panel, choose **Cut** from the panel context menu. If the target panel is selected, all selected panels are deleted.

The panel delete action can also be invoked from the Delete control button.

Pasting Deleted Panels

Deleted panels can be pasted back to any waveviews by choosing **Paste** from the panel context menu. The deleted panels are always pasted after the current panel in which the mouse pointer resides.

Grouping Panels

Selected panels can be grouped into one panel. Choose **Group** from the panel context menu or **Panel > Actions > Group** from the main menu.

If the grouped panels are in a logic panel type, a new bus signal is formed. The Bus Configuration dialog box allows you to configure the bus signal.

Panel grouping action can also be invoked from the Panel Group control button.

Ungrouping Panels

Signals inside a panel can be split so that each signal is placed into a new panel. Choose **Ungroup** from the panel context menu or **Panel > Actions > Ungroup** from the main menu.

Panel ungroup action can also be invoked from the Panel Split control button.

Zooming Panel Views

Zoom operations are supported in the following panel types:

- X/Y zoom in X-Y, 2D-Sweep and Eye-diagram panels
- X zoom in logic panels
- Radial zoom in Polar Coordinate panels

Depending on the default action type of the left mouse button in a waveview (default left mouse button action can be selected in **Preference Settings > Waveview > Left Button Default**), the zoom operation is invoked with:

- Cursor mode: You can invoke the zoom operation by first arming a waveview into the zoom mode. Click the **zoom** control button to activate the zoom mode.
- Zoom mode: The zoom operation automatically starts when you hold down and drag the left mouse button inside a waveform.
- Default Bindkey: Press the **Z** key to arm the zoom mode and release it to unarm.

If the left mouse button is defaulted to cursor operations (see Using Bindkey Functions), continue clicking the zoom control button, which cycles the zoom mode in the following order: X/Y zoom ( ), X zoom ( ), Y zoom ( ), no zoom ( ).

The current zoom mode is indicated with the different icons of the zoom control button. If the left mouse button is defaulted to zoom operations, continue clicking the zoom control button, which cycles the zoom mode among X/Y zoom ( ), X zoom ( ), and Y zoom ( ).

X/Y (Box) Zoom

With a waveview armed in the X/Y zoom mode, click and hold the left mouse button at a corner of the area to be zoomed in and drag the mouse pointer to the diagonally opposite corner of the zoom area. Release the button to stop zooming.

If you arm the X/Y-zoom operation of a waveview with the **Z** bindkey, releasing the **Z** bindkey without any zoom action results in a 2x zoom out. The same 2x zoom out action can also be invoked using the **Zoom Out** control button.

X Zoom

With the zoom control button in X-zoom mode, indicated by zoom control button, hold down the left mouse button and drag the mouse pointer along the horizontal direction to define the X-zoom range.

Default Bindkey: Press the **H** key to arm the horizontal zoom operation.

If you arm the X-zoom operation of a waveview with the **H** bindkey, releasing the **H** bindkey without any zoom action results in a 2x x-axis zoom out.

Y Zoom

With the zoom control button in Y-zoom mode, indicated by zoom control button, hold down the left mouse button and drag the mouse pointer along the vertical direction to define the Y-zoom range.

Default Bindkey: Press the **V** key to arm the vertical zoom operation.

If you arm the Y-zoom operation of a waveview with the **V** bindkey, releasing the **V** bindkey without any zoom action results in a 2x y-axis zoom out.

X Zoom to Fit

Select one or more panels, and press the **I** bindkey. The zoom to fit is applied to the selected panel axes. If no panel is selected, the zoom to fit is applied to all panels.

Y Zoom to Fit

Select one or more panels, and press the **Y** bindkey. The zoom to fit is applied to the selected panel axes. If no panel is selected, the zoom to fit is applied to all panels.

Un-zoom

To restore all selected panels to their full plotting ranges, click the un-zoom control button.

Default bindkey: **X**

Undo or Redo Zoom

The undo/redo zoom functions allow you to traverse up/down a zoom operation sequence. Use the undo-zoom function to undo a zoom step and redo-zoom function to redo a zoom operation.

Default Bindkeys: **U** for undo-zoom, **R** for redo-zoom.

Using Sliders in Zoomed Panels

In a zoomed panel (X-Y, logic or 2D-Sweep), the zoomed area can be interactively shifted using the horizontal and vertical slide bars. The waveview is redrawn at real time when the slide bar is moved. This graphic intensive operation can become noticeably slow especially when you work from a remote host.

The real-time screen redraw can be disabled for the slide bar to avoid the slow graphic operation. To disable real-time redraw, disable the **Real-time waveview scroll** option in **Preference Settings > Waveview**. With the real-time waveview scroll disabled, only the grid rulers are interactively redrawn during slide bar movement.

Setting Zoom Ranges Manually

The zoom operations can be also performed by manually specifying the X-axis and Y-axis ranges. Choose **Axes > Axes Range > Set Zoom** from the main menu, and enter the zoom range parameters. Then click **Ok** to set the zoom level on all panels or selected panels of the active waveview. Note that the Y ranges are in db (or db10) if the target panel Y-axis is in db (or db10) scale.

Specifying Panel Settings

The following panel settings are available:

- Displaying Data Points
- Controlling the Grid
- Adjusting Logarithmic Scales
- Using Fixed X-axis (or Y-axis) Full Scale
- Changing Axis Font Size
- Dual Y-axes
- Adjusting Panel Height
- Fitting Panels to Full Window Height
- Setting Vector Radix
- Setting Vector Length
- Setting Waveform Display Preferences
- Setting the Plot Mode for Complex Signals
- Setting Panel Attributes

Displaying Data Points

Waveforms are displayed sby plotting piece-wise-linear (PWL) lines between data points. To display the actual data points, click the Data Point control button to toggle on/off the data point markers.

Default bindkey: **D**.

Controlling the Grid

You can toggle the grids of selected panels on or off with the Grid control button.

Grid settings are toggled through the X-Y grid; X-grid only; Y-grid only; and No grid cycle.

Default bindkey: **G**.

You can also choose **Axes > Grid** from the main menu to turn X/Y grids on or off directly.

Grids parameters of XY panels can be redefined. Choose **Grid Settings** from the panel context menu or **Axes > Grid > User Settings** from the main menu to open the Grid Settings dialog box.

User-defined grids start from the Start value and repeat over the Step interval in both directions away from the start point.

Adjusting Logarithmic Scales

The X-Y, 2D-Sweep, and 3D-Sweep panels support the logarithmic scales. Choose **Axes > X-Log Setting** or **Axes > Y-Log Setting** from the main menu to switch X/Y-axis of all selected panels among: linear, logarithmic, db20, and db10 modes. Axis scales can be also changed directly using the axis context menu by right-clicking over the X/Y axis in a panel.

Logarithmic Y-axis has a range limit of six decades. If a linear Y-axis ranges from a negative minimum to a positive maximum, three decades are allocated for the positive and negative half respectively with the negative part coded in different color.

Using Fixed X-axis (or Y-axis) Full Scale

Fixed plotting ranges can be assigned to the X-axis of X-Y panels. Choose **Axes > X/Y Full Scale > Set X Full Scale** from the main menu and enter the X-axis range parameters. Click **Ok** to apply the change to selected panels of the active waveview. The default Y-axis range can be changed by choosing **Axes > X/Y Full Scale > Default Y Full Scale** from the main menu.

Changing Axis Font Size

Choose **Axes > Axis Font** and either **Normal** or **Large** from the Custom WaveView menu bar to change the font size.

Dual Y-axes

Dual Y-axes are supported in X-Y and the 2D-sweep panels. To display (or disable) the 2nd Y-axis in a X-Y panel, choose **Dual Axis > Show/Hide Right Axis** from the panel context menu. Choose **Dual Axis > I-Signals to Right Axis** to display all current signals of a panel using the right axis. Arbitrary signals can be also attached to (or detached from) the right axis by choosing

Signal "Name" > Right/Left Axis. Signals that are attached to the right axis are indicated with the icons in front of the signal names.

Setting Axis Attributes

To set axis attributes:

1. Double-click on an axis.

   The Axis Attributes window opens.

2. Choose an axis from the Axis menu.

   All axes in the active waveview window are available.

3. Click one of the zoom level buttons to change the display:

   - Zoom In increases magnification to show detail.
   - Zoom to Fit displays the maximum number of data points to show the entire range of a signal. The entire range is determined by the minimum and maximum values entered in the Full Scale Range text boxes at the bottom of the Axis Attributes window.

4. Enter values for the zoom level in the Min and Max text boxes.

   These values specify the range you want to display.

5. (Optional) Click the **Use Waveform Range** check box to specify the range of a signal, and enter the Min and Max values for that range.
6. (Optional) Click the **Grid On** check box to display the grid for the selected axes.

   You can click the **Auto Grid** check box and enter Start and Stop values for the gridding. You can only turn this option off for linear axes.

7. Choose the axis label font size from the Font menu.

   Normal and Large font sizes are available.

8. Choose an axis type from the Type menu.

   Axis types include **Linear**, **Log**, **dB20**, **dB20**, **dB10**, and **dBm10**.

9. Enter an axis label in the Label text box.
10. Click **Apply** to save your changes.

Adjusting Panel Height

Panel heights in a vertical stack waveview are calculated automatically based on the following preference settings:

- Maximum number of visible panels
- Minimum X-Y panel height

You can adjust the height of a panel by dragging the bottom edge of the left Y-axis area. If the panel in action is a selected panel, the height of all selected panels is changed. To restore selected panels to their default auto height, choose **Panel > Height > Default**.

Individual panel height adjustment is available for analog panels only. To adjust the height of all logic panels, change the value of Logic Panel Height in the preference settings.

Fitting Panels to Full Window Height

In a vertical stack waveview, height of selected panels can be automatically adjusted to fit to the full window height. Choose **Panel > Height > Fit** to fit all selected panels to the total window height, or use the default bindkey 'T' to toggle selected panels between fit and unfit modes. If the selected panels are separated into multiple groups, each group is fitted respectively. If there are too many panels in a group and the panel height becomes too small, the height is limited to the user-defined minimum height.

Choose **Panel > Height > Fit All** to fit all panels of a waveview to the total window height.

Setting Vector Radix

Logic vector values by default are displayed in binary. The following radix selections are supported:

- binary
- octal
- decimal
- hex-decimal
- ASCII

Select from the top menu **Panel > Radix** to select a radix setting for all selected panels. You can also choose **Radix** from the panel context menu to select a radix setting for the panel in action.

Setting Vector Length

Logic vector values by default are displayed in their shortest possible form. Select from the top menu **Panel > Display Preference > Full Logic Value** to display vector values in full length.

Setting Waveform Display Preferences

Analog signals can be displayed as PWL, PWC, Bar, or Data Point Only waveforms. Choose **Panel > Display Preference** from the main menu to select preferred display mode of a panel.

Setting the Plot Mode for Complex Signals

When complex signals are plotted on non-complex panels (X-Y, 2D-sweep, and 3D-sweep), the signals are by default plotted using the magnitude value of the complex data. You can switch it to the phase, real part, or imaginary part of the complex data.

Choose **Plot Complex** from the panel context menu to select the preferred complex setting. All signals inside a panel are plotted using the same complex mode. The following icons are placed before the signal names to indicate the current complex mode:

- (no icon) real-value waveform
- magnitude
- phase
- real part
- imaginary part
- group delay

To configure how complex signals should be displayed, choose **Config > Preferences > Panel tab** and select the preferred **Default AC Signal Display** setting. The following settings are available:

- **Complex**: Displays a single waveform using the magnitude value of the complex signal.
- **M/P**: Displays the magnitude and the phase of a complex signal respectively in two panels in stack mode.

- **R/I**: Displays the real part and the imaginary part of a complex signal respectively in two panels in stack mode.
- **M+P**: Displays the magnitude and the phase of a complex signal in the same panel.

Setting Panel Attributes

To set the attributes for a panel, right-click a panel and choose **Attributes** from the menu that opens. The Panel Attributes window opens with the options for the type of panel (analog, digital, or bus).

4

Loading, Displaying, and Printing Waveforms

This chapter contains information on how to load, display, and print waveforms.

Signals in a panel can be deleted, highlighted, or re-paneled. These actions can be invoked from the signal context menu by first arming the **Signal Mode** control button and then right-clicking the name or monitor area of a panel. Without arming the signal mode control button, right-clicking the same area opens the panel context menu.

The signal mode control button is red when armed.

Default bindkey: Press the **S** key to arm the signal mode and release it to unarm.

This chapter contains the following major sections:

- Grouping Waveform Files
- Adding Waveform Files to the Bookmark List
- Finding Signals
- Adding Signal Aliases
- Getting Signal Information
- Saving and Loading Signal Lists
- Working with Multi-Trace Waveforms
- Printing Waveforms

Opening Waveform Files

Waveform formats are automatically detected when a file is opened. To load a waveform file, click the **Import Waveform File** toolbar button or choose **File > Import Waveform File**.

Select a waveform data file (or multiple files) and click **OK** to open the selected waveform files, or click **Apply** to load more files without closing the dialog box.

You can open waveform files with the following options:

- **Preload all waveforms to RAM**

  When you open a waveform file, only the signal names and hierarchy directory is loaded into the system memory by default. The actual waveform data is loaded only when needed. By enabling this option, all waveform data is preloaded into the system memory when a waveform file is opened. Use caution when selecting this option and opening large data files—you might exhaust system RAM resources.

  Note: fsdb and NPX-SDIF files cannot be preloaded. Just the sweep results from tr0 and NW files can be preloaded.

- **Read multi-run data as multi-trace waveforms**

  Some output formats, such as the Berkeley raw and ELDO COU format, might combine results from multiple simulation runs into a single output file. You can either load the multiple runs as a sweep analysis of the same design, thus displaying each signal as a multi-trace sweep waveform, or you can read each run as a separate alter simulation run. Each run is independent from other runs as if they are read from separated files.

- **Automatically connect to subsequent split files**

  The WDF and fsdb formats support split files. When a file is open, the Custom WaveView tool automatically searches for subsequent files in the same directory. Disable this option if you do not want to connect waveforms from multiple split files. Split PSF files are always connected because subsequent PSF files contain waveform data only without signal names information; they cannot be opened as independent PSF files.

- **Convert to WDF**

  The WDF format is a Synopsys compression format that reduces the file size and offers fast access for large data files. Enable this option to convert the selected files into the WDF format. See Working with WDF Files for more information.

- **Load Data within Range Only**

  Load waveform data based on the specified x-axis range (for limited formats only).

The **File Filters** option menu can be customized in the Preferences Setting dialog box or the .spxrc configuration file.

Clicking **Home** resets the directory path to the working directory—the directory in which the program started, for example.

Note: Most application controls (except toolbar and main menu) are still functional during a waveform loading session. This feature allows tool operations in parallel to a lengthy waveform loading process.

Waveform files can be also loaded from command line as arguments. The usage is:

```
wv wdf1 wdf2 wdf3 …
```

To load multiple output files in different sub-directories, (a directory structure commonly seen in Cadence Artist environment), click **Apply** to load files from different directories or load files from the command line as:

```
wv */*.tran
```

Updating Waveforms

The Custom WaveView tool can read growing waveform data files during a simulation job. Click the **Update Waveforms** toolbar button or choose **File > Update Waveform Files** to update all displayed and derived signals.

If you restart a new simulation to override the old waveform data file, the update function automatically detects re-created data files and prompts you to select between keeping the old result or updating to new result.

If you keep the old data file, only the signals that are loaded are available from the old data file (because the old file is overwritten). An old data file is indicated by its blue file icon. To update an old data file to the new result, right-click the file in the Output View Browser and choose **Update WDF** from the context menu.

To display signals from both the new and the old data file, choose to keep the old result. The old result file is saved and automatically opens the new data file.

If you need to update the WDF signal list, you can right-click the name of a file in the Output View browser and choose **Reload WDF Signal List** from the menu that appears.

Waveform Marching (Automatic Update)

To automatically update waveforms from a running simulation job, choose **File > Waveform Marching** from the main menu. Enter the desired update interval and the abort trial limit and click the **Start** button to start waveform marching. Automatic update is stopped if the waveform file is not growing after the retrial limit. The automatic update function can be manually stopped at any time by clicking the **Stop** button in the same dialog box.

If the automatic waveform marching is enabled, it is indicated with an icon in the main application toolbar.

Clearing Waveforms

To clear all currently displayed waveforms, right-click on a waveform and choose **Clear Waveforms** from the menu that opens. All displayed waveforms are cleared.

Highlighting Waveforms

To highlight a waveform, click the signal check control button, and click the name of a signal to toggle its highlight state on and off.

You can also toggle a signal highlight by clicking the waveforms, which toggles the highlight states of all signals that intersect the mouse pointer.

To toggle which waveform is highlighted, press the **J** key to move down the waveform list and the **K** key to scroll up.

Grouping Highlighted Waveforms

Right-click a highlighted signal, and choose **Group** from the signal context menu to group all highlighted signals into a new panel.

You can also group signals by name by right-clicking signals and choosing **Group By Name** from the menu that opens.

Ungrouping Highlighted Waveforms

Right-click a highlighted signal, and choose **Ungroup** from the signal context menu to split each of the highlighted signals into a new panel.

Deleting Highlighted Waveforms

Right-click a highlighted signal and choose **Cut** from the signal context menu to delete all highlighted signals.

You can also move highlighted signals in a panel with a drag-and-drop operation. With the signal mode control button armed, follow the steps in Moving or Copying Panels to move highlighted signals between panels or waveviews.

Finding the Source of a Waveform

Right-click the name of a signal, and choose **Source of "Signal"** from the signal context menu to highlight its source location in the Output View signal browser. Note that only the Output View signal browser is used to show the signal source. The floating browsers of individual files, if open, are not used to show the signal source.

Scanning Waveforms

Once signals are loaded onto a waveview, you can scan signals from the same files using the **Scan** control buttons. This feature can be used to compare signals from different data files.

Clicking the **right arrow** button advances signals from all selected panels to their corresponding next signals from the same waveform data files. Clicking the **left arrow** scans signals in the reverse direction.

Please note that scanning signals might require reloading data if the next/ previous signals are not currently loaded. To speed up the scan process by avoiding data reloading, you might choose to load all waveform data to memory when waveform data files are initially opened (see Opening Waveform Files).

Default bindkey: **N** (next) or **P** (previous).

Scan Configuration

You can configure the scan function into two different modes:

- Direct Scan: With the Direct mode, the Custom WaveView tool scans signals according to their order in the waveform data file.
- Smart Scan: In the Smart mode, with a master waveform data file selected, the Custom WaveView tool scans signals from the master waveform data file according to their order in the file, while automatically searches other waveform data files for signals with matched names.

To reconfigure the current scan settings, choose **Config > Scan Settings** from the main menu, or click the **Scan Config** toolbar button to open the Scan-Control Settings dialog box.

Select a scan mode and a master file and click **Apply** to make the change. The **Lock Scope** option controls if the scan is limited within the current hierarchy scope.

Modifying Waveform Attributes

To modify waveform attributes for a waveform:

1. Double-click the signal name of the waveform, or right-click the signal name and choose **Signal '<signal_name>' > Attributes** from the menu that opens.

   The Waveform Attributes dialog box opens with the attributes for the chosen signal.
2. Change any of the following waveform display attributes:
   - Line Color
   - Line Width
   - Dash Type
   - Marker Shape
3. For analog signals, change the following attributes as necessary:
   - Display Preference
   - Hide Signal
   - Axis (left or right)
4. For digital signals, change the following attributes as necessary:
   - Display Preference and type
   - Mnemonic On
   - Radix
5. For analog or digital signals, enter the marker frequency, which changes the display frequency of the data point markers.
6. (Optional) Click the **Turn on Alias** check box to change the alias of a signal.
7. Click **Apply** to save your changes.

Changing Waveform Colors

Default waveform color choices can be added or deleted in **Preference Settings > Colors**. Up to 16 colors can be assigned as the default colors for waveforms.

Changing Waveform Color Schemes

Two default waveform color schemes are available. Both methods repetitively assign eight (8) different colors to waveforms.

Local Mode

With the local mode, waveform colors start from the first color for each panel. Colors are then sequentially assigned to subsequent waveforms in a panel. Depending on the order of waveforms in a panel, waveform colors might change while they are moved. You can set the local mode by choosing **Preference Settings > Waveview > Cycle Color By** and selecting the Panel mode.

Global Mode (default)

With the global mode, waveform colors start from the first color for each waveview. Colors are then sequentially assigned to each waveform in a waveview. The assigned colors stay with waveforms regardless of their locations even when they are moved to a different waveview. To select the global mode, choose **Preference Settings > Waveview > Cycle Color By**, and select the Waveview mode.

Working with Waveform Text Labels

Text remarks can be added to a waveview window. Click the **Text Remark** control button to open the Remark Text dialog box.

The label function supports multi-line text. Use the backslash character (\) to break a text label into multiple lines.

A text remark can be moved within a waveview window. Click and drag a remark box to move the text label. Release the mouse button at a new location to reposition the text remark.

Text remarks can be anchored to a (x,y) value point in a waveview. An anchored text remark moves with waveforms when a waveview is panned. Two font sizes are available: small and large.

To delete a text remark, right-click the text box to invoke the text remark context menu. Choose **Delete Remark** to delete the remark or **Delete All Remarks** to delete all text remarks in the waveview.

Grouping Waveform Files

Multiple waveform files can be collected into a group. When signals are loaded from a member in the grouped waveform files, the Custom WaveView tool automatically loads the same signals from the other waveform files in the same group.

The grouping function is useful for loading signals from corner/alter simulations that usually involve multiple output files. To group multiple waveform files, choose **File Grouping** from the WDF context menu. The Waveform File Grouping dialog box appears.

Select waveform files in the list to group/ungroup them. For waveform formats that have associated sweeping parameters (such as the COU and the PSF format), click **Select** to filter files based on their parameter value conditions.

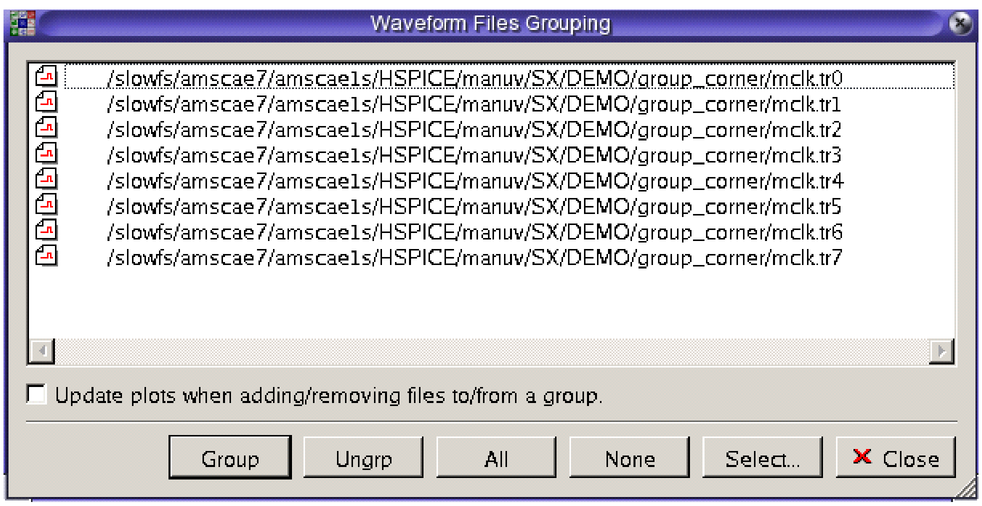


Figure 7 The waveform grouping dialog box

The **File Grouping** function only ties multiple files into a group. It does not assign any parameter condition to each individual file. As a result, parametric analysis is not possible on these grouped files. To add corner parameter conditions to files, the files must be grouped into a file set using a link file (similar to the runObjFile approach). The link file not only specifies which files to be grouped, it also defines the associated parameter conditions for each file in the group. Grouped files using the link file are processed as a single file as if the result is from a sweep analysis. Since the link file provides sweeping parameter conditions, parametric analysis is also supported.

To create the link file, right-click a file name and choose **Create File Set** from the context menu. To add parameter conditions to a file, enter the value in the (name=value) format, multiple values are delimited by comma (for example, temp=0,vdd=3.3). Click **Add** to add the parameter values to a file and the new values are indicated in the file list. Note that each of the files in a file set must have the same number of the same parameters. Finally, enter the path to the link file and click **Ok** to create the link file. Once a link file is generated, it can be opened directly as a regular waveform file.

If you want to update the plotted waveforms after files are added to or removed from a group, click the **Update plots when adding/removing files to/from a group** check box before clicking the **Group** or **Ungrp** buttons.

Adding Waveform Files to the Bookmark List

A waveform file can be added to the bookmark list. Choose **Add Bookmark** from the waveform file context menu in Output View. To reload a waveform file from the bookmark list, choose **File > Load/Manage Bookmarks**.

Using the Output View Browser

Once a waveform file is loaded, its signal name directory is displayed hierarchically in the Output View browser.

The Output View browser consists of an upper hierarchy browser and a lower signal list window. The lower list window displays signals under the selected hierarchy in the upper browser. Only one item can be selected in the upper hierarchy browser, while multiple signals can be selected in the signal windows for drag-and-drop operations.

To distinguish waveform data with the same file names from different directories, in the root entries of the upper Output View hierarchy browser, directory paths of loaded waveform files are displayed using directory prefix `DX` where X is the sequential index of different directories. To find out the original full paths of directory entries, select from the main menu **File > Show Directory Table** to display the table that maps directory identifiers to full file paths.

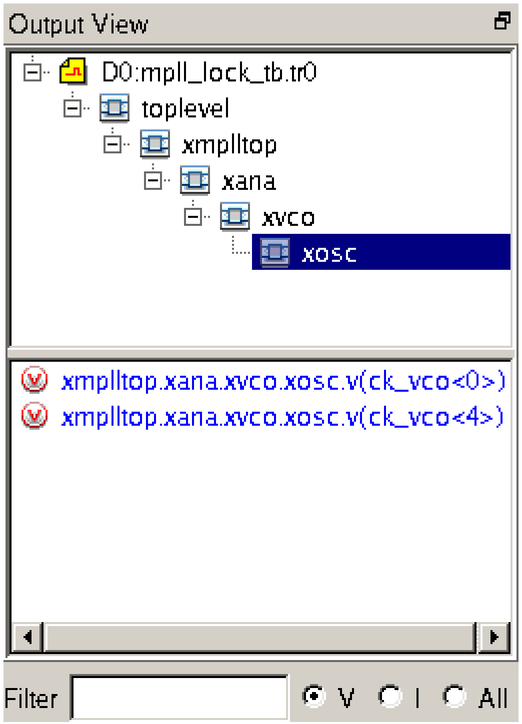


Figure 8 The output view window

Right-clicking a filename in the Output View hierarchy browser invokes the wdf context menu for the associated waveform file. The following items are available from the context menu:

- **Update WDF**: Reload the waveform file. The same update operation can also be invoked globally from the main menu (**File > Update Waveform Files**) for all waveform files.
- **Close This File**: Close the target waveform file. A dialog box appears and asks you to confirm the action. Closing a waveform file also removes all associated waveforms from waveviews.
- **Close Selected Files**: Select from the list of all files to close multiple files together.
- **Close All**: Close all waveform files.
- **Open New Browser**: Open a standalone floating signal browser for the target file. Each waveform file can have one standalone browser.
- **File Grouping**: Group waveform files.
- **Create File Set**: Create a link file for a multi-member file set.

- **Copy Label**: Copy a signal label to the global clipboard so it can be pasted outside of the Custom WaveView tool..
- **Sweep Display Filter**: Appears only with sweep result files. Allows users to select the active traces for the drag-and-drop and double-click display operations from the Output View.
- **2nd Sweep Variable**: Appears only with sweep result files. Defines the default 2nd sweeping variable for the parametric() function and Plot Y versus X2 function for cursors in a 2d-sweep panel.
- **Show/Hide Title**: Show/hide the title of a file.
- **Edit Title**: Edit the title text of a file.
- **Apply Measure**: Apply HSPICE `.MEASURE` commands (batch application).
- **HSPICE Measure Tool**: Start interactive HSPICE measurement tool.
- **Plot Graph**: Some formats (such as ELDO COU) contain display layout information. Select this item to extract the information and display waveforms accordingly.
- **Add Bookmark**: Add the waveform file to the bookmark list.
- **WDF Properties**: Display properties of the selected waveform file.
- **Hierarchy Filter**: Set a filter for the children of the node. When any hierarchy node is filtered out, "more with filter off" is displayed with the hierarchy to indicate that some nodes are currently hidden.

If multiple files are loaded, to allow simultaneous signal browsing on these multiple target files, the **Open New Browser** context menu function opens multiple signal browsers for each waveform file respectively.

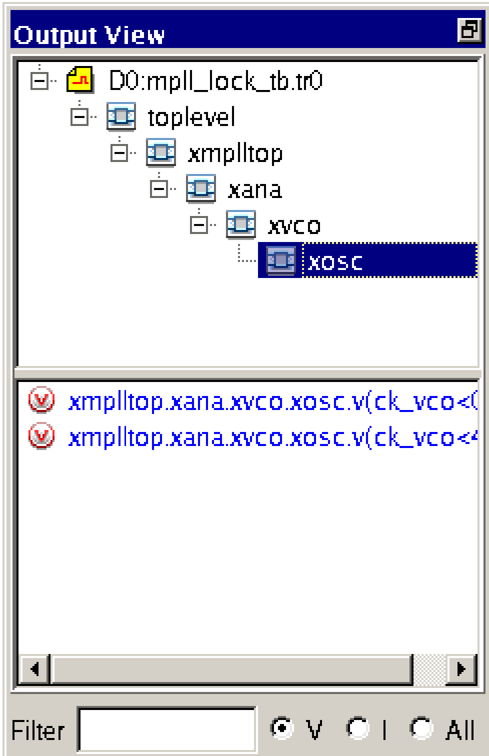
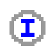
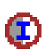


Figure 9 Stand-alone floating signal browser

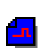
Right-clicking items in the Output View lower signal list window invokes the signal context menu. The following items are available from the signal context menu:

- **Signal Filter**: Select this item to filter the items in the browser based on their voltage/current type, hierarchy scope, alias name, or name pattern.
- **Show Signal**: Switch the signal list between all signals or filtered signals.
- **Name Preference**: Switch among the original database name, net name only by stripping off hierarchy path, or a user-defined alias.
- **Name Sorting**: Switch among no sort, alphabetic sort, sort by name length, then alphabetically, or V()/I() signals first.
- **Signal "NAME"**: Edit an alias, add to the equation of equation builder, use as the X-axis-variable, delete (for selected derived signals), modify (for a derived equation), or Display/Export dcop values from Spectre parametric analyses.

The following tables summarize different waveform types and icons:

Icon	Analog Waveform Type	Icon	Analog Waveform Type
	Real Voltage		(Real,Imaginary) Complex Generic
	Real Current		(Magnitude,Degree) Complex Generic
	Real Generic		(Magnitude,Phase) Complex Generic
	(Real,Imaginary) Complex Voltage		Real Voltage Alias
	(Magnitude,Degree) Complex Voltage		Real Current Alias
	(Magnitude,Phase) Complex Voltage		Real Analog Alias
	(Real,Imaginary) Complex Current		Sweeping Parameter
	(Magnitude,Degree) Complex Current		Derived Data Top Level
	(Magnitude,Phase) Complex Current		Wire Type

Icon	Digital Waveform Type	Icon	Digital Waveform Type
	Logic Integer		Logic Register
	Logic Supply		Logic Alias
	Logic Parameter		Logic Variable
	Logic Wire		(Magnitude,Phase) Complex Generic

Icon	Icon Type	Icon	Icon Type
	Waveform Data File (no signal displayed, not linked)		Generic signal alias name
	Linked Waveform Data File (linked wto netlist)		Voltage signal alias name
	Data file with some of its signals displayed		Current signal alias name
	An old data file that has been overridden by new result		Logic signal alias name
	One or more equations.		Waveform comparison results.

Displaying Signals

Signals from the Output View browser can be loaded into a waveview with the drag-and-drop operation. If a hierarchy (or file) entry is dragged from the upper hierarchy browser of Output View, all signals in the selected hierarchy level (or a file) are loaded. If a signal (or multiple signals) is dragged from the signal list window of Output View, only the selected signals are loaded. You can also double-click in the lower signal window to add an individual signal.

Depending on the signals being dragged, drop hints appear when signals are dragged over a waveview area. If the waveview is empty, hint boxes indicate available initial waveview modes. If the waveview contains existing panels, an insertion hint or drop-in box indicates the drop location.

If any signal from a waveform file is being displayed in a waveview window, the file icon changes, and the signal name is displayed in blue.

Filtering Signals

To filter the list of signals in the Signal Browser, enter the name or name pattern of the signal name you want to find into the Filter text box, which is just below the Signal Browser. Any signals that do not match those characters are filtered from the list.

Finding Signals

The Signal Finder dialog box can be used to search signals from all (or selected) loaded waveform data files and hierarchies. Click the **Signal Finder** toolbar button or select from the top menu **Tools > Signal Finder** to invoke the Signal Finder dialog box.

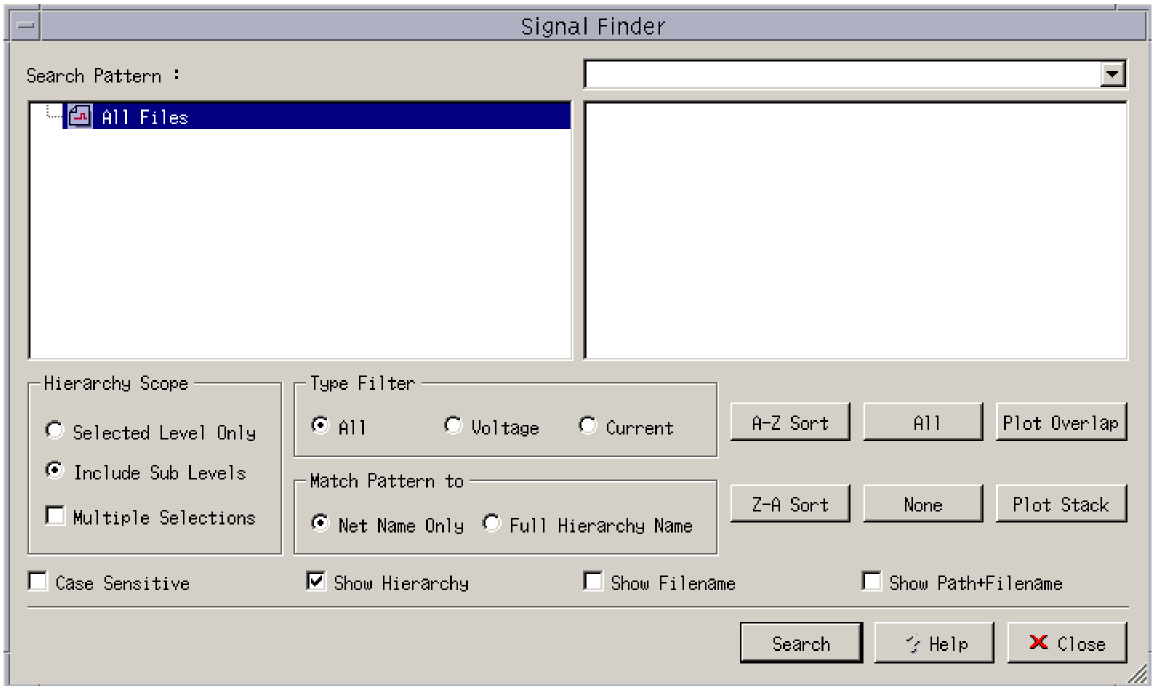


Figure 10 The signal finder dialog box

The signal finder supports name patterns matching with the asterisk (*) and question mark (?) wildcards. Enter a search pattern and hit the return key or click **Search** button to commence a search. The search results are displayed in the right signal list window. Click a signal from the list window to highlight the signal source in the OutputView browser. Drag-and-drop operations are also supported from the signal finder result window into any waveview window. You can sort the results alphabetically by clicking the **A-Z Sort** and **Z-A Sort** buttons.

The search target signals can be refined to Voltage/Current signals only or All Signals by clicking one of the **Type Filter** option radio buttons. To limit the search to a file or a hierarchy level, select the corresponding entry from the left file/hierarchy browser. If the **Selected Level Only** option is selected, the target signals are those in the selected hierarchy/file only. If the **Include Sub Levels** option is selected, the target signals are those in and below the selected hierarchy/file.

On UNIX platforms, select the **Multiple Selection** option to select multiple hierarchy entries as the search target.

The **All** and the **None** buttons are used to select/de-select all items in the right signal list window for the drag-and-drop operations.

Click **Plot Overlap** or **Plot Stack** to plot selected signals from the result window in overlap or stack mode.

The **Case Sensitive** option controls the signal name case sensitivity, while the **Net Name Only** and the **Full Hier Name** options control which part of the signal names to be used in the search process.

The **Show Hierarchy**, **Show Filename**, and **Show Path+Filename** options can be used to control the name appearance in the search result browser.

Adding Signal Aliases

Alias names can be added to signals in waveview windows. To define a new alias to a signal, choose **Signal "name" > Edit Alias** from the signal context menu. Enter an alias name, and click **Ok** to set the alias. To reset the name of a signal to its original name, follow the same sequence and enable the **Use Original Signal Name** option to reset the signal name.

Getting Signal Information

To view the name, file ID, and full path of a signal, move your mouse pointer over the name of a signal. The signal information is displayed in the button status bar.

Saving and Loading Signal Lists

To save a list of plotted signal names to a file, choose **File > Save Signal List** list from the Custom WaveView menu bar. To load a signal list, choose **File > Load Signal List**.

Working with Multi-Trace Waveforms

A typical analog signal from a circuit simulator contains a single trace of waveform data representing the full span of a single simulation run. Many simulators, however, generate a single output file for multiple simulation runs at different simulation conditions/parameters. Examples include:

- sw0, ac0, and tr0 files from HSPICE (Synopsys)
- COU files from ELDO (Mentor Graphics)
- PL files from Saber (Synopsys)

Each signal in these files contains multiple traces from different simulation runs. The Custom WaveView tool reads these multiple traces as a single signal and displays them in several different ways. The following sections describe how you can manipulate these multi-trace signals using different panels.

This section contains information on the following topics:

Reading Multi-Trace Data

HSPICE/Saber (Synopsys) parametric analyses can result in multi-trace sweep output files. These files have the same file extension as those from the regular simulation runs. But a single signal from these sweep result file actually consists of multiple traces representing results from different runs.

ELDO (Mentor Graphics) parametric analysis also stores output in a similar way and results in multi-trace COU files. The multi-trace files are different from the HSPICE/Saber output files, you can read in multi-trace COU files in one of the following ways (also applied to Berkeley raw format):

- A single multi-trace COU file can be read as multiple virtual single-trace files. This is the default way that COU files are processed in the Custom WaveView tool.
- The COU file can be read as a single file with each signal containing multiple traces. You need to enable the **read multi-run data as multi-trace waveforms** option in the Loading Waveform File dialog box before loading the waveform file.

Loading Multi-File PSF Sweep Analysis Result

The PSF format stores sweep analysis result in multiple waveform files under the same directory. To load these result files and their associated sweeping parameters, open the runObjFile or the logFile in the output directory. Multiple PSF files under the sweep analysis output directory are processed as a single file under the name of runObjFile or logFile.

The runObjFile is also used to open Cadence Spectre Monte Carlo analysis results or Cadence Analog Artist parametric sweep analysis results in multiple directories. For Analog Artist results, open the runObjFile in the main psf/ directory.

Creating a File Set from Multiple Files

If a sweep analysis is done externally using script, multiple files are generated. To combine these files together with their associated corner variable conditions, use the **Create File Set** function in Grouping waveform files.

Displaying Multi-Trace Signals

A multi-trace waveform can be displayed in regular panels such as X-Y panels, Polar Plots, or Smith Charts. The signal is automatically broken into individual traces. Each trace is assigned with a different color and used to represent result from one simulation run. The signal name is automatically amended to include the associated parameter values.

Multi-trace signals can be also displayed in the 2D-sweep and 3D-sweep panels. All traces of a multi-trace signal in the 2D or 3D-sweep panel is operated collectively as a single signal entity and assigned with a single waveform color and name.

The **Sweep Display Filter** function can be used to define the active sweep traces for display operations. To define the active traces, right-click over the waveform file in the Output View and choose **Sweep Display Filter** from the context menu. For sweep waveform files in a file group, the Custom WaveView tool searches for the first group member file that has its Sweep Display filter enabled. The Sweep Display filter settings are then used to apply to all file members in the group.

Breaking Multi-Trace Signals

A multi-trace sweep signal can be manually broken into individual single-trace signals. Right-click a multi-trace signal in the Output View browser, and choose **Break Signal-Name** from the context menu that appears. The converted waveforms are stored under `derived waveforms.TRACE`. The name of a converted single-trace signal includes the parameter conditions of the corresponding trace.

Calculating Waveforms for Multi-Trace Signals

When a multi-trace signal is involved in a waveform equation, the result is also a multi-trace waveform. However, when a signal from grouped files is involved in an equation, the same equation is recalculated multiple times for each file automatically. The results are multiple single-trace waveforms with each waveform representing the calculation result from each file.

If a waveform equation produces a scalar result, the application of such an equation on multi-trace signals would generate multiple scalars. To convert these scalar results into a parametric plot, use the parametric() function in the Equation Builder and select one of the sweeping parameters as the x-axis of the parametric plot.

Note: An equation can only involve multi-trace signals that have the same number of traces and the same sweeping parameter conditions.

Selecting the Sweeping Parameter

A multi-trace signal can contain multiple parameters for each trace. The following table shows a typical example:

Trace Index	Temperature	Vdd
Trace 1	10	5
Trace 2	100	5
Trace 3	10	4.8
Trace 4	100	4.8

You can select the active sweeping parameter by right-clicking the sweep result file in the Output View browser and choosing **X2 Parameter** from that context menu that appears. The active sweeping parameter is used in the Y vs. X2 conversion in a 2D-sweep panel and equation calculation of the parametric() function.

Filtering Multi-Trace Waveforms

A multi-trace signal contains multiple waveform traces. Each trace has a set of associated sweeping parameters (or sub X-variables). For example:

```
Trace 1: Temp=0 Vdd=3.3
Trace 2: Temp=0 Vdd=5
Trace 3: Temp=100 Vdd=3.3
Trace 4: Temp=100 Vdd=5
```

By default, a multi-trace signal is plotted with respect to its primary independent sweeping variable (time, for example). In this case, you can choose **Sweep line filter** from the context menu in a 2D-sweep panel to display selected traces.

The 2D-sweep panel also supports plotting a multi-trace signal with respect to sweeping parameters other than the primary variable (sub X-variable). To change the default X-axis of a 2D-sweep panel to a sub X-variable, choose **Change X Variable** from the panel context menu. Drag and drop a sweeping parameter (type P) from the Output View browser to the Setting X Variable dialog box.

When a 2D-sweep panel is set up to plot waveforms against a sub X-variable, choosing **Sweep Line Filter** from the panel context menu opens the Sub X-Variable Plotting Filter dialog box.

The filter setup is different between using the default X-axis and sub X-Variable X-axis. This is because in the sub X-Variable plot mode, based on the selected sub X-variable, the number of traces can vary depending on the number of unique combinations among other sweeping parameters. The number of traces are automatically detected and sorted in the order of the sub X-Variable values when displaying waveforms.

Note: A multi-trace signal must have equal number of points in each of its original traces in order to be plotted in the sub X-variable mode.

Using Multi-Trace Signals as X-Axes

Multi-trace signals cannot be plotted with respect to another multi-trace signal.

Viewing or Modifying Sweep Signal Attributes

To view or modify sweep signal attributes for an XY panel:

1. Right-click on an XY panel, and choose **Sweep Attributes** from the menu that opens.

   The Sweep Attributes dialog box opens.
2. Choose a panel from the Panel menu.

   Panels are identified by the number of rows or column they contain. The default is **All**.
3. Choose a signal from the Signal menu for which you want to change sweep attributes.
4. Choose a color for the sweeps from the Color menu.

   Rainbow is selected by default.
5. Click the check box in the Show column of the sweep table for each trace you want to show in a sweep.

The traces displayed in the panel are selected by default. When you move your cursor over the data in the parameter table, the line is highlighted in the corresponding panel. Any changes that you make are saved as they occur.

You can click the **Show All** or **Hide All** buttons at the bottom of the Sweep Attributes dialog box to show or hide all traces, respectively.

6. (Optional) Click **Filter** to filter traces by parameter value.

   The Filter Traces by Parameter Value dialog box opens. Select parameters and values to filter, and click **OK** to save your changes.

7. (Optional) Click the **Apply other signals with the same parameter** check box to sync the attribute settings with other signals.

8. Click **Close** when you are finished.

Printing Waveforms

On UNIX platforms, the Custom WaveView tool supports printout in the PostScript format only. To submit a print job, click toolbar button or choose **File > Print** from the main menu to open the Print Setup window.

The print target is defaulted to the current active waveview, which can be changed by clicking **Change**. Select your preferred settings and click **Print** to submit the print job.

PostScript Print Layout

Multiple print options are supported for the PostScript printout:

- Print only the active waveview to a single page as shown on the screen.
- Print all signals in the active waveview using the display height. Multiple pages might be needed if the waveview has signals that are scrolled out of the view.
- Print all waveviews in a single page, multi-column layout. The user can choose the number of columns preferred.
- Print all waveviews, each in a single page, as shown on the screen display.

Select the **Color PostScript** option if your printer supports color PostScript.

The default print setup is black/white in landscape orientation. The default printer device and user-defined print command can be predefined in the .spxrc file.

Printing on UNIX Platforms

On UNIX platforms, the Print Setup dialog box supports print out to different paper sizes and printers. Printer names can be predefined in the Printer Devices field of the Preference Settings dialog box. Multiple printer names are delimited by semicolon (;). Selected paper size and printer device are saved in `$HOME/.spxlast` when the application is closed. The stored values are automatically used as the default when the application starts again. If the `PRINTER` environment variable is defined in your UNIX shell, it is always used as the default printer.

Printing on Windows Platforms

On MS Windows platforms, in additional to the PostScript print-to-file option, the Custom WaveView tool also supports printout to all installed printers except line printers. PostScript format is supported as a file output option only. Click toolbar button or choose **File > Print** from the main menu to open the MS Windows version of the Print Setup dialog box. Select preferred options and click **Continue** to select a printer device, or select the path for the PostScript output file.

5

Measuring Waveforms

This chapter contains information on using cursors and measurements.

The Custom WaveView tool provides the following ways to measure waveforms:

- Cursors: Used to trace waveform values. Unlimited cursors can be added. See Working with Cursors for more information.
- Monitors: Used to monitor global waveform properties such as minimum, maximum, average, RMS, and peak-to-peak values. Monitors can be associated with cursors to trace waveform values such as derivative and difference, or associated with the viewing ranges to display local minimum, maximum, average, RMS, and peak-to-peak values. See Working with Monitors for more information.
- Measurements in the Measurement Tool: Used to make interactive measurements on a waveform or between different waveforms. Various measure types are available including rise/fall time, frequency, width, and jitter. See Using the Measurement Tool for more information.

Working with Cursors

Cursors are used to trace waveform-related values. Multiple cursors can be added to a panel. The X-axis position of a cursor is indicated underneath the cursor hair.

Adding Cursors

To add a cursor, choose **Add Cursor** from the panel context menu. A cursor can also be created using the Add-Cursor control button.

You can also add cursors for delta measurements by pressing the left- and right-bracket keys ([and]) while moving the cursor.

Working with the Active Cursor

When multiple cursors are added to a panel, the last moved or added cursor becomes the active cursor.

Waveform values corresponding to the active cursor are displayed in the monitor. In an X-Y panel, distances from all non-active cursors to the active cursor are displayed next to the active cursor.

Moving a Cursor

Depending on the default action of the left mouse button in a waveview (default left mouse button action can be selected in **Preference Settings > Waveview > Left Button Default**), a cursor can be moved with:

- Cursor Mode: If the left mouse button is pressed with mouse pointer near a cursor, the cursor is grabbed. A grabbed cursor can be dragged to a new position. If the mouse pointer is not in the vicinity of any cursor when the left mouse button is pressed, the active cursor is grabbed regardless of its position.
- Zoom-First Mode: If the left mouse button is pressed and released without any mouse pointer movement. The active cursor is grabbed to the mouse pointer location when the button is released.

In a zoomed panel, cursors outside the plotting range are placed on the left (or right) boundary of the plotting area. They can be grabbed from the borders of the plotting area. The cursor that is outside but closest to the boundary is grabbed first.

Jumping Cursors

In an X-Y or logic panels, a cursor can be jumped forward (or backward) to the next (or previous) data point, peak, switching or crossing point using the cursor context menu. Right-click near the cursor to invoke the cursor context menu. Choose **Jump (F)orward "SIGNAL"** or **Jump (B)ackward "SIGNAL"** from the context menu to jump the cursor forward or backward on the target signal SIGNAL. The target signal is indicated by a cross marker on the signal waveform at the intercept point with the cursor hair. To select a different target signal, move the cursor near a waveform (along the Y direction) and the cross marker snaps to the new target waveform.

To change the jump parameters, choose **Jump Settings** from the cursor context menu to invoke the Cursor Jump Parameters dialog box.

Enter a preferred jump type and associated parameters and click **Ok** to make the change. Click **Set as Default** to save the settings in ~/.spxlast as the default setting.

For digital panels, the cursor can be jumped to the next value change, the next match value, the next rise/fall edge, or the user-defined time.

To jump to a specific X value, right-click a V cursor and choose **Goto X Value** from the menu that opens.

Default bindkeys: Press the **F** key to jump forward, and the **B** key to jump backward.

Locking Pairs of Cursors

A pair of vertical cursors can be locked to a fixed distance. Choose **Setup Locked Pair** from the cursor context menu. Specify the target cursor and the preferred distance to lock two cursors at a fixed distance. To unlock a pair of locked cursors, choose **Release Locked Pair** from the context menu.

Using Horizontal Cursors

Horizontal cursors are supported in analog X-Y panels. Choose **Switch to H-Cursor** from the cursor context menu to change a vertical cursor to a horizontal cursor. The horizontal cursors trace the x-axis values of the nearest intersect point with waveforms. The x-values are displayed in the monitor column.

To add a horizontal cursor directly, choose **Axes > Cursor > Add H-Cursor** from the main menu.

You can also click the **Add Cursor** control button to add an H-cursor by setting the **Add H-Cursor with Add Cursor Control Button** option in the preferences. To set this preference, choose **Config > Preferences** from the main menu and click the WaveView tab. Click the **Add H-Cursor with Add Cursor Control Button** check box, and click **OK**.

In an X-Y panel, labels can be annotated to all the signal crossing points of a horizontal cursor. To enable crossing value labels, choose **Configure H-Crossing Values** from the cursor context menu.

Note: H-cursor label positions are recorded when the total number of H-crossing labels stay the same during H-cursor movement. The modified cursor label positions are saved in the session file.

To hide the H-cursor hair, right-click an H-cursor and choose **Configure H-crossing Values** from the menu that opens. Click the **Hide Horizontal Cursor Hair** check box, and click **OK**.

Cursors in Smith Charts and Polar Plots

In Smith charts and Polar plots, a cursor consists of a circle hair and a radial hair. The cross point is the cursor location and its value is displayed. A cursor can be grabbed by clicking the mouse button when the mouse pointer is near any of its two hairs. The cursor values in the monitor column trace the data points that are closest to the hair cross.

Cursors in 2-D Sweep Panels

You can use a cursor in a 2D-sweep panel to create a new 'Y vs. X2' plot. Move the cursor to a preferred position, and choose **Plot Y vs X2** from the cursor context menu.

Cursors in 3-D Sweep Panels

In a 3D-sweep panel, a cursor consists of a vertical hair, and two crossing hairs lying in the base plane. To grab a cursor in a 3D-sweep panel, move the mouse pointer near the vertical hair base where the two crossing hairs meet.

Deleting Cursors

To delete a cursor, click the right mouse button near a cursor to invoke the cursor context menu. Choose **Delete** from the context menu to delete the cursor, or choose **Delete All** to delete all cursors.

Working with Monitors

Monitors are used to trace waveform-related values. Multiple monitors can be added to vertical waveviews, or panels in horizontal. The following monitors are available for analog panels:

- Cursor value (Y=): Traces waveform values that correspond to the current active cursor position. Logarithmic value is displayed if the corresponding panel has logarithmic Y-axis. To display the logarithmic value of a negative value x, the format `-<log(abs(x))>` is used.
- Derivative (S=): Traces waveform derivative values that correspond to the current active cursor position. Derivative of logarithmic values is displayed if the corresponding panel has logarithmic Y-axis.
- 1/Derivative (1/S=): Traces the reciprocal of waveform derivative values that correspond to the current active cursor position.
- Y Delta (D=): Displays difference between (1) waveform values that correspond to the active cursor position and (2) waveform values that correspond to the other cursor. Delta value is available only when a waveview (or panel) has two cursors.
- Cursor Average (A=): Displays waveform average between the cursors. Cursor average value is available only when a waveview (or panel) has two cursors.
- Maximum (MAX=): Displays waveform maximum values over full or zoomed signal range.
- Minimum (MIN=): Displays waveform minimum values over full or zoomed signal range.
- Peak to Peak (PP=): Displays waveform peak-to-peak values over full or zoomed signal range.

- Average (AVG=): Displays waveform average values over full or zoomed signal range.
- RMS (RMS=): Displays waveform root mean square values over full or zoomed signal range.

The following radix options for logic panels are available:

Option	Description
use panel setting	Uses the panel default radix setting.
binary	Displays logic values in binary radix.
octal	Displays logic values with octal radix.
signed	Displays logic values in a signed decimal number
decimal	Displays logic values in decimal numbers.
hexidecimal	Displays logic value with hex-decimal radix.
ASCII	Decodes logic vector into ASCII text.
rise edge count	Displays the rising edge count.
fall edge count	Displays the falling edge count.
rise/fall edge count	Displays the rising and falling edge count.

Adding Monitors

If a vertical waveview (or a panel in a horizontal waveview) has no existing monitor in the cursor value type, a cursor value type monitor is automatically added when a cursor is created.

To add a new monitor, click the Monitor control button to open the Monitor Settings dialog box. Select the preferred settings, and click **Ok** to create the monitor.

Copying Monitor Values

To copy the value of a monitor, right-click a monitor, and choose **Copy Value** from the menu that opens. The value is copied to the clipboard.

Deleting Monitors

To delete a monitor, right-click a monitor, and choose **Delete Monitor** from the context menu that appears.

Reconfiguring Monitors

To reconfigure a monitor, right-click a monitor, and choose **Configure Monitor** from the context menu that appears. In the Monitor Setting dialog box that opens, reconfigure the monitor and click **Ok** to make the change.

Linking Monitors to Cursors

To link a monitor to a selected cursor, right-click a monitor and choose **Link Cursor** from the context menu that appears. In the Cursor Selection dialog box that opens, link any desired monitors to cursors.

Using the Measurement Tool

To access the Measurement Tool, choose **Tools > Measurement** from the main menu.

Once one or more measurements are added to a waveview, you can drag those measurements to take the measurement interactively. If a panel has multiple signals and you want to perform a Width, Slew, Frequency, or Duty-Cycle measurement, the default measurement target is the first selected (highlighted) signal. The first signal is the default if no signal is selected.

Note: The names of any signals that are currently being measured are displayed at the bottom of measurement configuration dialog box.

The following topics are available in this section:

- Supported Measurements
  - General Measurements
  - Time Domain Measurements
  - Frequency Domain Measurements
  - Statistical Measurements
  - Level Measurements
  - S Domain Measurements
  - RF Measurements
- Adding or Removing Measurement Favorites
- Viewing Measurement Results
- Setting the Precision of Measurements
- Exporting Measurements

Supported Measurements

The following measurements are supported:

Measurement	Type	Description
AC Coupled RMS	Level	Calculates the RMS value of the AC component of a waveform.
Amplitude	Level	Calculates the difference between the topline and baseline reference levels.
Average	Level	Calculates the average level of a waveform.
Bandpass	Frequency Domain	Calculates the bandwidth of a bandpass-shaped waveform.
Baseline	Level	Calculates the baseline level of a waveform.
Cpk	Statistics	Calculates an indicator of the process capability for a waveform relative to specified upper and lower limits.

Measurement	Type	Description
Damping Ratio	S Domain	Calculates the damping ratio of a waveform as -real/mag.
Data(X,Y)	General	Displays the X-axis and Y-axis values of a point on a waveform.
Delay	Time Domain	Calculates the delay between the edges of two waveforms relative to the default or specified topline and baseline levels for both the measured waveform and the reference waveform.
Difference	General	Displays the following information for two points on one or two waveforms: X1, Y1; X2, Y2; and Delta X, Del Y, Slope.
Dpu	Statistics	Calculates the total number of defects per unit based on the points in a waveform.
Duty Cycle	Time Domain	Displays a duty cycle of a periodic waveform.
F VS T	Time Domain	Calculates all frequency, period, or duty cycle values for the edges crossing the threshold, and then creates new Frequency vs. time, Period vs. time, or Duty Cycle vs. time waveforms using the specified X range and Y level.
Frequency	Time Domain	Displays the frequency and period of a periodic waveform.
Frequency Value	S Domain	Displays the frequency value at the point of a complex waveform on a Nyquist (or Nichols) plot.
Gain Margin	Frequency Domain	Displays the gain margin of a complex analog waveform in decibels (dB).
Highpass	Frequency Domain	Calculates the corner frequency of a waveform with a highpass shape.
IP2	RF	Calculates the second-order intercept point.

Measurement	Type	Description
IP3/SFDR	RF	Calculates the third-order intercept point.
Jitter	Time Domain	Displays the histogram of crossing edges with a specified X range or Y range.
Length	General	Calculates the length of a straight line that connects two points on a waveform or two points on two waveforms.
Lowpass	Frequency Domain	Calculates the corner frequency of a waveform with a lowpass shape.
Maximum	Level	Calculates the maximum level of waveform.
Mean	Statistics	Calculates the mean value of a waveform.
Mean+3std_dev	Statistics	Calculates the (mean + 3 standard deviation) value of a waveform.
Mean-3std_dev	Statistics	Calculates the (mean - 3 standard deviation) value of a waveform.
Minimum	Level	Calculates the minimum level of a waveform.
Natural Frequency	S Domain	Calculates the natural frequency of a waveform as the absolute value of an argument.
Overshoot	Time Domain	Calculates the overshoot of a waveform relative to a default or specified topline.
P1dB	RF	Calculates the 1dB compression point.
Peak to Peak	Level	Calculates the peak to peak values of a waveform.
Phase Margin	Frequency Domain	Displays the phase margin of a complex analog waveform in degrees.
Quality Factor	S Domain	Calculates the quality factor of a waveform.
Rise/Fall Time	Time Domain	Calculates the risetime and falltime between specified upper and lower levels of a waveform.

Measurement	Type	Description
RMS	Level	Calculates the RMS (Root Mean Square) value of a waveform.
Settle Time	Time Domain	Calculates the settle time of a waveform with respect to a default or specified settle level and a specified settle band.
Slew Rate	Time Domain	Calculates the slew rate of a waveform relative to the default or specified topline and baseline levels.
Std_dev	Statistics	Calculates the standard deviation of a waveform.
Stopband	Frequency Domain	Calculates the stopband, low, high, or center frequency level or the level at which the measurement is calculated for a stopband-shaped waveform.
Topline	Level	Calculates the topline level of a waveform.
Undershoot	Time Domain	Calculates the undershoot of a waveform relative to a default or specified baseline level.
Width	General	Calculates the pulse width of a waveform.
X at Maximum	Level	Displays the x-value corresponding to the maximum value of a waveform.
X at Minimum	Level	Displays the x-value at the minimum value of a waveform.
Y Diff	General	Displays the Delta Y at a X value between two points on two waveforms.
Y Range	General	Displays the following information for a waveform: Waveform name, peak-to-peak Y range; Minimum Y value, Maximum Y value; and Average Y value, RMS value.

Measurement	Type	Description
Yield	Statistics	Calculates the ratio of the number of data points between the Y-axis levels Upper and Lower relative to the total number of data points.

General Measurements

The following general measurements are available:

- Data(X,Y)
- Difference
- Length
- Width
- Y Diff
- Y Range

Data(X,Y)

Displays the X-axis and Y-axis values of a point on a waveform.

Note: This measurement is supported only in Analog X-Y, 2-D Sweep, and Smith Chart waveview panels.

The following parameters can be set for the Data(X,Y) measurement:

Parameter	Description
Locked Level	Locks the meter Y level at the specified percentage or level (in Y-axis units).
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a meter Y level that is locked at a specified percentage.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a meter Y level that is locked at a specified percentage.

Parameter	Description
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.

Difference

Displays the following information for two points on one or two waveforms: X1, Y1; X2, Y2; and Delta X, Del Y, Slope. The capacitance and inductance values from the X and Y points are also calculated using the following functions:

$$C = \frac{x2 - x1}{2\pi \cdot x1x2 \cdot (y1 - y2)}$$

$$L = \frac{y1 - y2}{2\pi(x1 - x2)}$$

Note: This measurement is supported only in Analog X-Y, Eye Diagram, and 2-D Sweep waveview panels.

The following parameters can be set for the Difference measurement:

Parameter	Description
Locked Level	Locks the meter Y level at the specified percentage or level (in Y-axis units).
Indep P2 Lock Level	Enables the Lock 2nd Point at independent second Y-level point meter option.
Lock 2nd Point at	Locks a second meter Y-level point at the specified percentage or level (in Y-axis units). This option is only available when the Indep. P2 Lock Level option is selected.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a meter Y level that is locked at a specified percentage.

Parameter	Description
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a meter Y level that is locked at a specified percentage.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Anchor First Point	Anchors the measurement at a single point; further dragging only moves the second point.
Gravity Snap	Snaps anchors to data points when moving.
Label Move Independently	Allows label and endpoints to move independently.

Length

Calculates the length of a straight line that connects two points on a waveform or two points on two waveforms. If two waveforms are selected, the two waveforms do not need to be the same type.

Width

Calculates the pulse width of a waveform.

Note: This measurement is supported only in Analog X-Y, Login, and Eye Diagram waveview panels.

The following parameters can be set for the Width measurement:

Parameter	Description
Locked Level	Locks the meter Y level at the specified percentage or level (in Y-axis units).
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a meter Y level that is locked at a specified percentage.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a meter Y level that is locked at a specified percentage.

Parameter	Description
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Width Rate vs. X-axis Plot Option	Calculates the width value of an up or down pulse versus the X-axis.

Y Diff

Displays the Delta Y at a X value between two points on two waveforms.

Note: This measurement is supported only in Analog X-Y waveview panels.

Y Range

Displays the following information for a waveform:

- Waveform name
- Peak-to-Peak Y range
- Minimum Y value
- Maximum Y value
- Average Y value
- RMS value

Note: This measurement is supported only in Analog X-Y waveview panels.

Specifying the **Measure over Zoomed X-Range** option measures the zoomed area that is currently displayed on screen. If this option is not specified, the whole waveform is measured even if the waveform is zoomed and some of the X-range is not visible.

Time Domain Measurements

The following General measurements are available:

- Delay
- Duty Cycle
- F VS T

- Frequency
- Jitter
- Overshoot
- Rise/Fall Time
- Settle Time
- Slew Rate
- Undershoot

Delay

Calculates the delay between the edges of two waveforms relative to the default or specified topline and baseline levels for both the measured waveform and the reference waveform. The rising or falling edge on the reference waveform is assumed to cause the corresponding (rising or falling) edge on the measured waveform so that the reference edge occurs before the measured edge.

All rising or falling edges for the measured waveform are determined based on the Trigger setting in the Measure dialog box. From each edge, the corresponding previously occurring edge on the reference waveform is determined. The difference between the two edges on the X-axis is the delay time.

The following parameters can be set for the Delay measurement:

Parameter	Description
Percentage	Locks the delay level at the specified percentages for the signal (L(%)) and reference signal (Ref. L(%)). In signal Y units.
Signal Level	Locks the delay level at the specified levels for the signal (L(V)) and reference signal (Ref. L(V)). In signal Y units.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a delay level that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a delay level that is locked at a specified percentage range.

Parameter	Description
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Trigger	Specifies the trigger type as **Either**, **Rising**, or **Falling** edge(s)
Ref. Trigger	Specifies the reference signal trigger type as **Either**, **Same**, or **Opposite**.
Delay Rate vs. X-axis Plot Option	Calculates the delay rate values of the **Rise**, **Fall**, or **Rise/Fall** edges vs. the x-axis.

Duty Cycle

Displays a duty cycle of a periodic waveform, which is measured in one of the following ways:

The Duty Cycle is calculated as the percentage of:

$$\frac{t1}{t}$$

Note: This measurement is supported only in Analog X-Y and Login waveview panels.

The following parameters can be set for the Duty Cycle measurement:

Parameter	Description
Locked Level	Locks the meter Y level at the specified percentage or level (in Y-axis units).
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a meter Y level that is locked at a specified percentage.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a meter Y level that is locked at a specified percentage.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Duty Cycle vs. X-axis Plot Option	Calculates the duty cycle value vs. the x-axis.

F VS T

Calculates all frequency, period, or duty cycle values for the edges crossing the threshold, and then create new Frequency vs. time, Period vs. time, or Duty Cycle vs. time waveforms using the specified X range and Y level.

Note: This measurement is supported only in Analog X-Y waveview panels.

The following parameters can be set for the F VS T measurement:

Parameter	Description
Locked Level	Locks the meter Y level at the specified percentage or level (in Y-axis units).
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a meter Y level that is locked at a specified percentage.

Parameter	Description
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a meter Y level that is locked at a specified percentage.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Delta-P vs T Plot	Plots the delta period versus time. This option is available by right-clicking your mouse and choosing **Delta-P vs T Plot** from the menu that opens.

Frequency

Displays the frequency and period of a periodic waveform. The period is calculated as the difference in time between two consecutive edges of the same polarity of a waveform. The frequency is calculated as:

$$\frac{1}{period}$$

Note: This measurement is supported only in Analog X-Y and Login waveview panels.

The following parameters can be set for the Frequency measurement:

Parameter	Description
Locked Level	Locks the meter Y level at the specified percentage or level (in Y-axis units).
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a meter Y level that is locked at a specified percentage.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a meter Y level that is locked at a specified percentage.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.

Parameter	Description
F vs. T Plot Option	Calculates the Fequency, Period, Delta Period, or Duty Cycle value versus the X-axis.

Jitter

Displays the histogram of crossing edges with a specified X range or Y range. You must click the right mouse button menu on the measurement meter to complete the measurement.

Note: This measurement is supported only in Analog X-Y, Eye Diagram, and 2-D Sweep waveview panels.

The following parameters can be set for the Jitter measurement:

Parameter	Description
Locked Level	Locks the meter Y level at the specified percentage or level (in Y-axis units).
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a meter Y level that is locked at a specified percentage.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a meter Y level that is locked at a specified percentage.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Rising	Measures the rising edge jitter.
Falling	Measures the falling edge jitter.
H-Crossing	Measures edges that cross the horizontal range.
V-Crossing	Measures edges that cross the vertical range.
New WaveView	Plots the jitter histogram on a new waveview window.
Active WaveView	Plots the jitter histogram on the active waveview window.

Overshoot

Calculates the overshoot of a waveform relative to a default or specified topline. The overshoot is calculated as the difference between the maximum point on the waveform and the specified (or calculated) Topline value.

The following parameters can be set for the Overshoot measurement:

Parameter	Description
Percentage	Locks the rise/fall margin threshold at the specified high (H(%)) and low (L(%)) percentages. In signal Y units.
Signal Level	Locks the rise/fall margin threshold at the specified high (H(V)) and low (L(V)) signal levels. In signal Y units.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a rise/fall margin that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a rise/fall margin that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Absolute	Specifies the result as "maximum value-high level value."
Percentage	Specifies the result as a percentage of "absolute value/(high level-low level)."

Rise/Fall Time

Calculates the risetime and falltime between specified upper and lower levels of a waveform.

Note: This measurement is only supported in Analog X-Y waveview panels.

The following parameters can be set for the Rise/Fall Time measurement:

Parameter	Description
Percentage	Locks the rise/fall margin threshold at the specified high (H(%)) and low (L(%)) percentages. In signal Y units.
Signal Level	Locks the rise/fall margin threshold at the specified high (H(V)) and low (L(V)) signal levels. In signal Y units.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a rise/fall margin that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a rise/fall margin that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Each Edge	Uses the local edge range to determine the low and high thresholds.
Within X Range	Specifies the X range from which the low and high thresholds are determined.
Rise/Fall Time vs. X-axis Plot Option	Calculates the Rise, Fall, or Rise/Fall time versus the X-axis

Settle Time

Calculates the settle time of a waveform with respect to a default or specified settle level and a specified settle band. The settle time is calculated by searching the waveform from right to left to find the first point that is outside of the settle band. The time that the waveform leaves the settle band is used as the settle time.

The following parameters can be set for the Settle Time measurement:

Parameter	Description
Percentage	Locks the settle level at the specified percentage level (Level(%)). In signal Y units.

Parameter	Description
Signal Level	Locks the settle level at the specified signal level (Level(V)). In signal Y units.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a settle level that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a settle level that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
% of SettleLevel	Calculates the band value as the specified band value percentage multiplied by the settle level (for example, 5*settle_level).
% of PeaktoPeak	Calculates the band value as the specified band value percentage multiplied by the peak to peak range of the signal (for example, 5*peak_to_peak_range).
Absolute	Calculates the band value as the specified value (in signal Y units).

Slew Rate

Calculates the slew rate of a waveform relative to the default or specified topline and baseline levels.

Note: This measurement is only supported in Analog X-Y waveview panels.

The slew rate is calculated using the difference between the upper and lower levels of a waveform divided by the risetime or falltime of the edge. You can select the upper and lower levels as a percentage of the topline or baseline.

The following parameters can be set for the Slew Rate measurement:

Parameter	Description
Percentage	Locks the rise/fall margin threshold at the specified high (H(%)) and low (L(%)) percentages. In signal Y units.
Signal Level	Locks the rise/fall margin threshold at the specified high (H(V)) and low (L(V)) signal levels. In signal Y units.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a rise/fall margin that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a rise/fall margin that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Slew Rate vs. X-axis Plot Option	Calculates the slew rate values of the **Rise**, **Fall**, or **Rise/Fall** edges vs. the x-axis.

Undershoot

Calculates the undershoot of a waveform relative to a default or specified baseline level. The undershoot is calculated as the difference between the minimum point on the waveform and the specified (or calculated) Baseline value.

The following parameters can be set for the Undershoot measurement:

Parameter	Description
Percentage	Locks the rise/fall margin threshold at the specified high (H(%)) and low (L(%)) percentages. In signal Y units.
Signal Level	Locks the rise/fall margin threshold at the specified high (H(V)) and low (L(V)) signal levels. In signal Y units.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a rise/fall margin that is locked at a specified percentage range.

Parameter	Description
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a rise/fall margin that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Absolute	Specifies the result as "maximum value-high level value."
Percentage	Specifies the result as a percentage of "absolute value/(high level-low level)."

Frequency Domain Measurements

The following Frequency Domain measurements are available:

- Highpass
- Lowpass
- Bandpass
- Stopband
- Gain Margin
- Phase Margin

Highpass

Calculates the corner frequency of a waveform with a highpass shape. The highpass measurement is calculated relative to a default offset or a specified topline and a specified offset.

The corner frequency is calculated by searching from the right to the left until the waveform first falls below the measurement level, which is determined by the specified offset value (from the topline).

The following parameters can be set for the Highpass measurement:

Parameter	Description
Percentage	Locks the topline at the specified percentage (Topline(%)). In signal Y units.

Parameter	Description
Signal Level	Locks the topline at the specified signal level (Topline). In signal Y units.
Auto Detect	Calculates the topline and uses that value.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a topline that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a topline that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Offset	Specifies the offset in Y-axis units.
Show Topline	Displays the topline.
Show Topline+Offset Line	Shows the cross line in a panel.

Lowpass

Calculates the corner frequency of a waveform with a lowpass shape. The lowpass measurement is calculated relative to a default or specified topline and a specified offset.

The corner frequency is calculated by searching from the left to the right until the waveform first falls below the measurement level, which is determined by the specified offset value (from the topline).

The following parameters can be set for the Lowpass measurement:

Parameter	Description
Percentage	Locks the topline at the specified percentage (Topline(%)). In signal Y units.
Signal Level	Locks the topline at the specified signal level (Topline). In signal Y units.

Parameter	Description
Auto Detect	Calculates the topline and uses that value.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a topline that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a topline that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Offset	Specifies the offset in Y-axis units.
Show Topline	Displays the topline.
Show Topline+Offset Line	Shows the cross line in a panel.

Bandpass

Calculates the bandwidth of a bandpass-shaped waveform.

The following parameters can be set for the Bandpass measurement:

Parameter	Description
Percentage	Locks the topline at the specified percentage (Topline(%)). In signal Y units.
Signal Level	Locks the topline at the specified signal level (Topline). In signal Y units.
Auto Detect	Calculates the topline and uses that value.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a topline that is locked at a specified percentage range.

Parameter	Description
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a topline that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Offset	Specifies the offset in Y-axis units.
Show Topline	Displays the topline.
Show Topline+Offset Line	Shows the cross line in a panel.

Stopband

Calculates the stopband, low, high, or center frequency level or the level at which the measurement is calculated for a stopband-shaped waveform. This measurement is calculated relative to a default or specified topline level and a specified offset:

```
fcenter = square root of (flow*fhigh)
quality factor = fcenter/bandwidth
```

The following parameters can be set for the Stopband measurement:

Parameter	Description
Percentage	Locks the topline at the specified percentage (Topline(%)). In signal Y units.
Signal Level	Locks the topline at the specified signal level (Topline). In signal Y units.
Auto Detect	Calculates the topline and uses that value.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying a topline that is locked at a specified percentage range.

Parameter	Description
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying a topline that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.
Offset	Specifies the offset in Y-axis units.
Show Topline	Displays the topline.
Show Topline+Offset Line	Shows the cross line in a panel.

Gain Margin

Displays the gain margin of a complex analog waveform in decibels (dB). The gain margin is the difference between the gain of the measured waveform and 0 dB at the frequency where the phase shift is -180 degrees.

This measurement might fail if the phase of the measured waveform does not pass through -180 degrees or if the waveform is not complex.

Phase Margin

Displays the phase margin of a complex analog waveform in degrees. The phase margin is the difference between the phase of the measured waveform and the reference degree at the unity gain frequency. The reference can be either -180 degrees or 0 degree.

This measurement might fail if the magnitude of the waveform does not pass through 0 dB or if the waveform is not complex.

The following parameters can be set for the Phase Margin measurement:

Parameter	Description
-180	Specifies the reference as -180 degrees.
0	Specifies the reference as 0 degrees.

Statistical Measurements

The following Statistical measurements are available:

- Cpk
- Dpu
- Mean
- Mean+3std_dev
- Mean-3std_dev
- Std_dev
- Yield

Cpk

Calculates an indicator of the process capability for a waveform relative to specified upper and lower limits. Cpk is calculated using the following equation:

$$\frac{\text{upper} - \text{mean}}{3(\text{std\_dev})}$$

or

$$\frac{\text{mean} - \text{lower}}{3(\text{std\_dev})}$$

In this calculation, `mean` represents the mean value of the scatter plot, `upper` and `lower` represent the specification limits you specify, and `std_dev` represents the standard deviations of the scatter plot. When both the `upper-mean` and `lower-mean` values are provided, the smaller result of these two calculations is displayed as the measurement.

The following parameters can be set for the Cpk measurement:

Parameter	Description
Percentage	Locks the upper/lower threshold at the specified high (H(%)) and low (L(%)) percentages. In signal Y units.
Signal Level	Locks the upper/lower threshold at the specified high (H(V)) and low (L(V)) signal levels. In signal Y units.

Parameter	Description
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying an upper/lower threshold that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying an upper/lower threshold that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.

Dpu

Calculates the total number of defects per unit based on the points in a waveform. Given one or both of an upper and lower specification limit, the Dpu is calculated as the area under a normal distribution that falls outside the specification limit(s). The mean and standard deviation of the normal distribution are equal to the mean and standard deviation of the points in the waveform.

The following parameters can be set for the Dpu measurement:

Parameter	Description
Percentage	Locks the upper/lower threshold at the specified high (H(%)) and low (L(%)) percentages. In signal Y units.
Signal Level	Locks the upper/lower threshold at the specified high (H(V)) and low (L(V)) signal levels. In signal Y units.
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying an upper/lower threshold that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying an upper/lower threshold that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.

Mean

Calculates the mean value of a waveform, which is calculated using the following equation:

$$\frac{1}{N}\sum_{j=1}^{N} W_j$$

In this calculation, `N` is the number of points, and array W_j contains the individual points of the waveform.

Mean+3std_dev

Calculates the (mean + 3 standard deviation) value of a waveform.

Mean-3std_dev

Calculates the (mean - 3 standard deviation) value of a waveform.

Std_dev

Calculates the standard deviation of a waveform. This measurement is intended for statistical (discrete) data such as histograms and is calculated using the following equation:

$$\left[\frac{1}{N-1}\sum_{j=1}^{N}(W_j - \overline{w})^2\right]^{\frac{1}{2}}$$

Yield

Calculates the ratio of the number of data points between the Y-axis levels Upper and Lower relative to the total number of data points.

The following parameters can be set for the Yield measurement:

Parameter	Description
Percentage	Locks the upper/lower threshold at the specified high (H(%)) and low (L(%)) percentages. In signal Y units.
Signal Level	Locks the upper/lower threshold at the specified high (H(V)) and low (L(V)) signal levels. In signal Y units.

Parameter	Description
Target Signal	Uses the min/max Y levels from the signal you are currently measuring. This option is only available when specifying an upper/lower threshold that is locked at a specified percentage range.
All Signals	Uses the min/max Y levels from all signals that are currently open. This option is only available when specifying an upper/lower threshold that is locked at a specified percentage range.
User Specified	Uses the min/max Y values you specify in the Min and Max text fields.

Level Measurements

The following Level measurements are available:

- Average
- AC Coupled RMS
- Amplitude
- Baseline
- Maximum
- Minimum
- Peak to Peak
- RMS
- Topline
- X at Maximum
- X at Minimum

AC Coupled RMS

Calculates the RMS value of the AC component of a waveform. The AC coupled RMS value is calculated using the following equation:

$$\left[\frac{1}{(x2-x1)}\int_{x1}^{x2}(w-\bar{w})^2\,dx\right]^{\frac{1}{2}}$$

In this calculation, `w` represents the waveform as well as its average value. The `x1` and `x2` values are the starting and ending points for the waveform. Add WV as an option for the AC Coupled measurement.

Amplitude

Calculates the difference between the topline and baseline reference levels.

Average

Calculates the average level of a waveform.

Baseline

Calculates the baseline level of a waveform.

Note: This measurement is automatically calculated when enabled.

Maximum

Calculates the maximum level of a waveform.

Minimum

Calculates the minimum level of a waveform.

Peak to Peak

Calculates the peak to peak value of a waveform.

RMS

Calculates the RMS (Root Mean Square) value of a waveform using the following equation:

$$\left[\frac{1}{(x2-x1)}\int_{x1}^{x2}(W^2dx)\right]^{\frac{1}{2}}$$

In this calculation, *W* represents the waveform, and *x1* and *x2* represent the starting and ending points.

Topline

Calculates the topline level of a waveform. The topline is automatically calculated using the probability density histogram method.

X at Maximum

Displays the x-value corresponding to the maximum value of a waveform.

X at Minimum

Displays the x-value at the minimum value of a waveform.

S Domain Measurements

The following S Domain measurements are available:

- Damping Ratio
- Frequency Value
- Natural Frequency
- Quality Factor

Damping Ratio

Calculates the damping ratio of a waveform as -real/mag.

Frequency Value

Displays the frequency value at the point of a complex waveform on a Nyquist (or Nichols) plot.

Natural Frequency

Calculates the natural frequency of a waveform as the absolute value of an argument. The natural frequency is calculated using the following equation:

$$\text{natural frequency} = \sqrt{(\text{real}^2 + \text{imag}^2)}$$

Quality Factor

Calculates the quality factor of a waveform. This measurement is calculated as:

$$\frac{1}{2(\text{damping ratio})}$$

RF Measurements

The following RF measurements are available:

- P1dB
- IP2
- IP3/SFDR

P1dB

Calculates the 1 dB compression point.

IP2

Calculates the second-order intercept point.

IP3/SFDR

Calculates the third order intercept point.

The following parameters can be set for the IP3/SFDR measurement:

Parameter	Description
Calculate SFDR	Specifies that IP3 is calculated as well as SFDR.
Use INTEGRATED_NF	Specifies the Integrated Noise Figure to be used for the SFDR calculation.
Noise Figure (dB)	Specifies the Noise Figure in decibels (dB).
SN ratio (dB)	Specifies the signal-to-noise ratio in decibels (dB).
Band width (dB)	Specifies the system Bandwidth in decibels (dB). This parameter is not available when "Use INTEGRATED_NF" is selected.
Integrated NF (dB)	Specifies the Integrated Noise Figure in decibels (dB). This parameter is not available when "Use INTEGRATED_NF" is selected.

Adding or Removing Measurement Favorites

In the Favorites tab, you can choose from several measurement you use most often.

To add a measurement to the list of favorites:

1. Click the **All** tab.

   The available measurement categories and corresponding measurement appear.
2. Click a measurement.
3. Click the **Favorite** button at the bottom of the Measurement Tool window.

   The selected measurement is saved as a favorite and is included as one of the measurements you can choose from the Favorite tab.

To remove a measurement from the favorite measurement list:

1. Click the **Favorite** tab.

   The current measurement favorites are displayed at the top of the Measurement Tool window.
2. Click a measurement.
3. Click the **Remove** button at the bottom of the Measurement Tool window.

   The selected measurement is removed from the list of favorite measurements you can choose from the Favorite tab.

Viewing Measurement Results

To see existing Measurement Tool results from a waveview, right-click a measurement result in a waveview, and choose **Measure Results** from the menu that opens. The Measure Results window opens with a table of measurement result values.

Setting the Precision of Measurements

To change the precision of measurements, choose **Config > Preferences**, and click on the General tab. Enter a value in the Number of Valid Digits text box.

Exporting Measurements

When you perform a measurement on a waveform, the measurement can be exported to the following equivalent forms:

- Equivalent HSPICE `.MEASURE` statements
- Tcl ACE `sx_equation` statements
- Equation builder equations

To directly export a measurement to the Equation Builder, right-click a measurement and choose **Meter to Eqn** from the context menu. The equivalent measurement equation and the result are displayed in the equation builder.

To export all measurements on a waveview window, right-click a measurement and choose **Meter Export**. Use the Dynamic Meter Export dialog box to select the meters to be exported and the preferred export type. Only the following types of measurements can be exported:

- Rise/Fall time
- Width/Frequency
- Difference
- Y-Difference
- Y-Range

In the Dynamic Meter Export dialog box, you can define the export name of individual measurements, enable/disable a meter in the list, control how to count the number of switch edges, and define the precision for the exported values.

For HSPICE Measure and ACE Tcl export modes, the path of the output file must be specified.

Locking and Unlocking Meters

To lock or unlock a meter, right-click the meter and choose **Lock Meter** or **Unlock Meter**, respectively, from the menu that opens.

6

Comparing Waveforms

This chapter contains information on comparing waveforms.

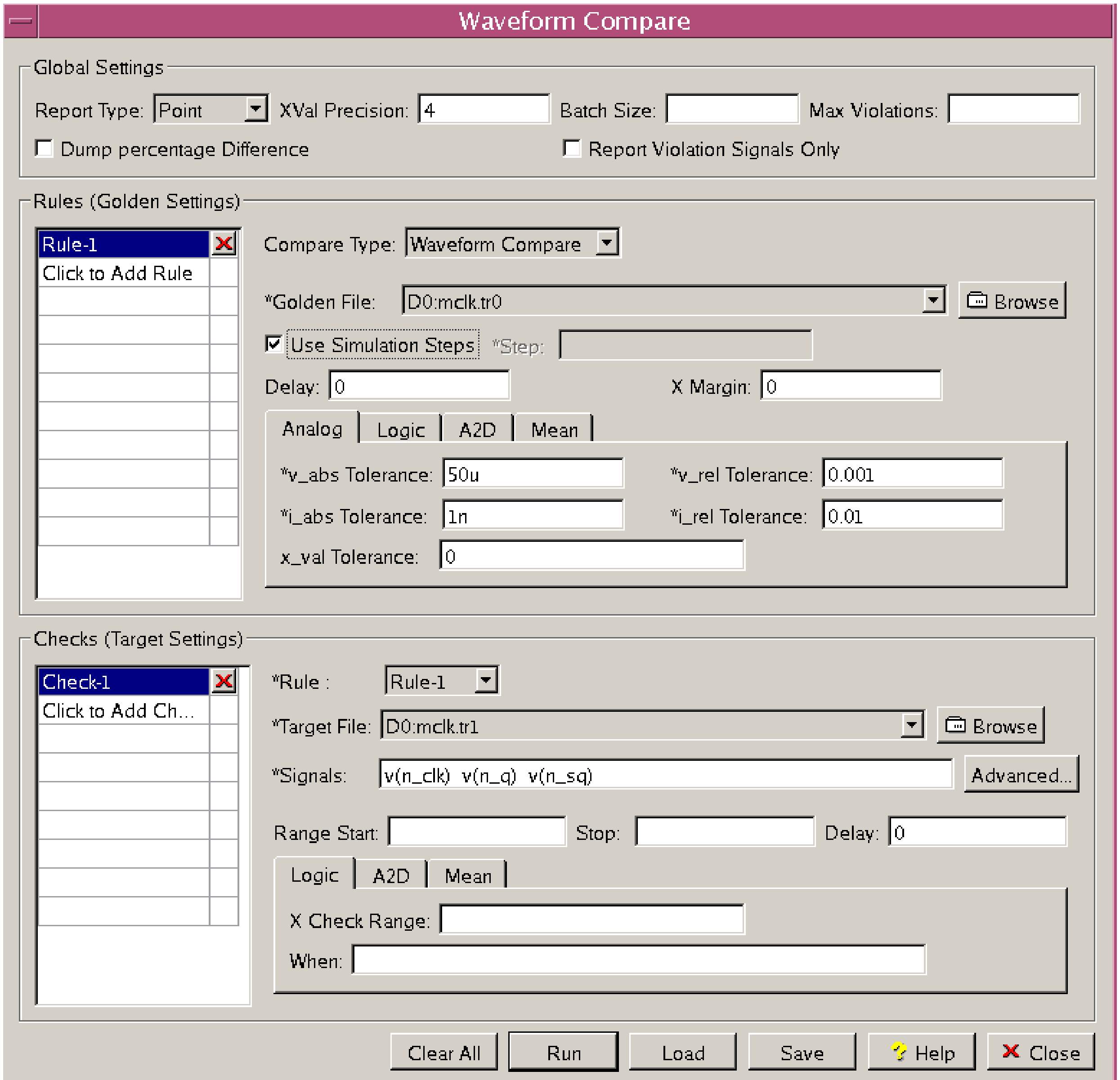
The Custom WaveView tool provides the following waveform comparison features:

Comparing Waveforms in the GUI

Note: This function requires a CustomExplorer license.

To compare waveforms in the GUI:

1. Choose **Tools > Waveform Compare** from the menu bar.



The Waveform Compare window opens.

2. Specify Global Settings.

   See Specifying Global Settings.

3. Add one or more rules.

See Adding Rules (Golden Settings).

4. Add one or more checks.

   See Adding Checks (Target Settings)

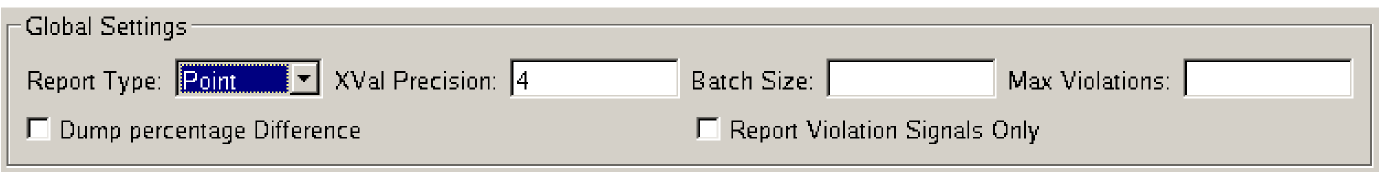
5. Run your comparison.

   See Running Comparisons.

Specifying Global Settings

To specify Global Settings for a waveform comparison:

1. Choose the type of report you want to generate from the **Report Type** menu in the Global Settings.

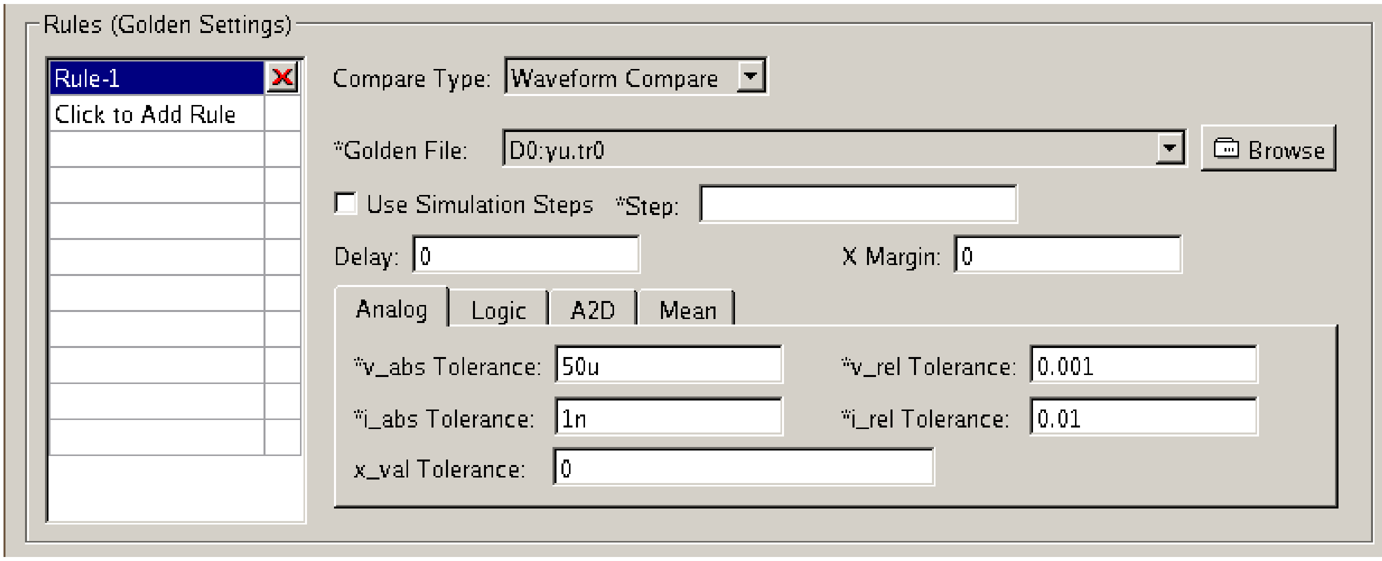


2. (Optional) Enter values for the **XVal Precision**, **Batch Size**, and **Max Violations**.
3. (Optional) Click the **Dump Percentage Difference** check box to include the percentage difference in the comparison report.
4. (Optional) Click the **Report Violation Signals Only** check box to include only the signals that have violations in your comparison report.
5. To add rules to your waveform comparison, continue to the next section, Adding Rules (Golden Settings).

Adding Rules (Golden Settings)

Note: If you already have a rule file set up, click the **Load** button to load your rule file.

To add rules to your waveform comparison, choose what kind of comparison you want to run from the **Compare Type** menu. Double-clicking on rule name in the rule table enters editing mode where you can modify the rule name. No spaces are allowed in rule names.



The following comparison types are available. Any settings for each comparison type that has an asterisk (*) next to it are required:

Comparison Type	Description
Waveform Compare	Compares waveforms. Continue to the Adding Waveform Compare Rules section.
Envelope Check	Checks the specified envelope. Continue to the Adding Envelope Check Rules section.
Monotonicity Check	Checks for monotonicity. No further rule setup in necessary—continue to the Adding Checks (Target Settings) section.
Bound Check	Checks the specified bounds. Continue to the Adding Bound Check Rules section.
Length Check	Checks the specified length. Continue to the Adding Length Check Rules section.
Pattern Search	Searches for the specified pattern. Continue to the Adding Pattern Search Rules section.

Adding Waveform Compare Rules

To add waveform compare rules:

1. Choose a golden file from the **Golden File** menu, or browse to and select the golden file you want to use.
2. Enter the step value you want to use for the waveform comparison in the **Step Value** text box, or click the **Use Simulation Steps** check box to use the steps from your simulation instead.
3. (Optional) Enter delay and X margin values into the **Delay** and **X Margin** text boxes.
4. Enter values for the following Analog tolerance values on the Analog tab:

- **v_abs Tolerance**
- **i_abs Tolerance**
- **v_rel Tolerance**
- **i_rel Tolerance**
- **x_val Tolerance**

5. (Optional) Click the **Always Convert Analog signals to Logic then Compare** check box to always compare analog to digital.
6. Enter values for the **High** and **Low** A2D Threshold.
7. Click the Logic tab, and choose values from the **Treat as X** and **Treat as Z** menus.
8. Enter a value into the **X Check Range** text box for the X shift you allow when comparing a transition.

   This is enabled only when **Use Simulation Steps** is selected.
9. Enter a value into the **When** text box to specify a logical condition.

   The data is compared when the condition is true (for example, when "xclk/clkin==0") .
10. Click on **Mean** tab.
11. Click the **Enable Mean Check** check box to check the mean values of each step instead of comparing the original values.
12. Click **Mean Steps** to specify the steps used in a mean check.
13. Click **Mean Tolerance** specify the absolute tolerance for mean value comparisons.
14. To add checks to your waveform comparison, continue to the next part of the waveform setup, Adding Checks (Target Settings).

Adding Envelope Check Rules

To add envelope check rules:

1. Choose a signals file from the **Bound Signals File** menu, or browse to and select a file.
2. Enter signal names in the **Bound 1** and **Bound 2** text boxes.
3. To add checks to your waveform comparison, continue to the next part of the waveform setup, Adding Checks (Target Settings).

Adding Bound Check Rules

To add bound check rules, click the **Upper Bound** or **Lower Bound** check boxes, and enter values for the upper bound or lower bound limit, respectively.

To add checks to your waveform comparison, continue to the next part of the waveform setup, Adding Checks (Target Settings).

Adding Length Check Rules

To add length check rules:

1. Choose a golden file from the **Golden File** menu, or browse to and select the golden file you want to use.
2. Enter values for the **Absolute Tolerance** and **Relative Tolerance**.
3. To add checks to your waveform comparison, continue to the next part of the waveform setup, Adding Checks (Target Settings).

Adding Pattern Search Rules

To add pattern search rules, enter a pattern in the **Pattern to Use** text box.

To add checks to your waveform comparison, continue to the next part of the waveform setup, Adding Checks (Target Settings).

Adding Checks (Target Settings)

You can add multiple checks for each rule you add. To add checks to your waveform comparison setup:

1. Choose a rule from the **Rule** menu, which you already set up in the Adding Rules (Golden Settings) section.

   The displayed check settings change depending on the type of rule you choose.
2. Choose a target file from the **Target File** menu, or browse to and select the target file you want to use.
3. Enter the names of one or more signals you want to check into the **Signals** text box.
4. Click **Advanced** to specify more signal related settings.
5. Enter a value for the **Max Hierarchy Level**.

0 is the default value.

6. Choose a type of signal from the **Signal Type** menu.
7. Click the **Match Type** check box to match the type of signal.

   If you are setting up a check for a rule that is Length Check comparison, no further setup for this check is available. Skip back to Step 1 to add more checks for rules, or if you are finished, skip to Step 16. Otherwise, continue to the next step.
8. Enter values for the **Range Start** and **Stop**.

   If you are setting up a check for a rule that is Monotonicity or Bound Check comparison, no further setup for this check is available. Skip back to Step 1 to add more checks for rules, or if you are finished, skip to Step 16. Otherwise, continue to the next step.
9. Enter a value for the **Delay**.
10. Specify the master to target signal name mappings in the Signal Mapping Table.
11. Click the **Logic** tab.
12. Enter a value into the **X Check Range** text box for the X shift on a target signal if different from the X Check Range value in Rule section.

    This works only when you enable **Use Simulation Steps**.
13. Enter a value into the **When** text box to specify a logical condition using target file signals.

    The data is compared when the condition is true (for example, when xclk/clkin_b==0).

    Skip back to Step 1 to add more checks for rules, or if you are finished, continue to the next step.
14. Click on the **A2D** tab, and enter the **Low** and **High** A2D thresholds if they differ from the thresholds you set in Rule section.
15. Click the **Mean** tab, and enter the Mean check start and end values.
16. Your waveform comparison setup is complete. Continue to the final part of the waveform comparison, Running Comparisons.

Running Comparisons

To run a waveform comparison:

1. Ensure you have the waveform comparison setup complete.

   See Comparing Waveforms in the GUI for more information.

2. Click the **Run** button.

   The Save Waveform Compare Result File dialog box opens.

3. Enter a name for your result file, and click **OK** to save it in the desired location.

   The waveform comparison runs, and the results are saved in your result file. The result file is automatically opened in the Custom WaveView tool, and the result is plotted to a tableview.

   See Working with Tabular Waveform Comparison Data for more information on working with tabular waveform comparison data.

Comparing Waveforms in Batch Mode

Note: This function requires a CustomExplorer license.

The batch mode sample-based waveform compare function compares mixed-signal waveforms between multiple target waveform files and a golden master waveform file. The comparison rules are defined in a check rule file that is used to control the comparison process.

To compare waveforms in batch mode using the Waveform Compare utility:

1. Review the general rules that apply when comparing waveforms.

   See Sampling and Converting Waveform Data for more information.

2. Create a check rule file if you want to compare signals and generate reports of any differences that exist.

   See Creating a Waveform Compare Control File for more information.

3. Use the following syntax at the command line to start a comparison:

   ```
   wv -compare rule_file [output_file] [-x session_file]
   ```

The `output_file` is the name of the output file where the comparison results are saved. If you do not specify a name, the comparison result is displayed directly in the standard output. If you specify `-x session_file`, a Custom WaveView session file is created, which saves the comparison files, signals, and differences information. You can load this file into the Custom WaveView tool to view the comparison results.

During comparison, if any errors occur due to unlocated or mismatched signals, a list of the mismatched signals are saved in a mismatch.txt file.

Sampling and Converting Waveform Data

The following general rules apply when comparing waveforms:

- If a logic bus signal has different width (number of bits) between the master and the target file, only the less significant bits of the longer bus value are compared with the shorter bus value.
- Multiple signals are compared sequentially during the comparison process.
- Analog waveforms are compared based on the final V/I tolerance that is evaluated from user-defined tolerance values as:

  ```
  tolerance = MAX(abstol, reltol*master_waveform_value)
  ```
- A signal must exist in both the master and one of the target files for the comparison to take place.
- If one of the signals from the master file and the target file is a logic signal, logic comparison is used to compare the two signals.
- For analog waveforms, voltage tolerance is used if master and target signals are not both current type.

Creating a Waveform Compare Control File

Before the waveforms are compared, create a check rule file to compare signals and generate reports of any differences that exist.

The check rule file must include the following items:

- Path to the master file and the target files.
- Names of signals to compare and name mapping information if the signal names are different between the master file and the target files.
- Start, stop, delay, and step values for the comparison to take place.

Adding Comments

When adding comments, ensure each line begins with a semicolon (;).

Defining Parameters

You can define parameters to control the waveform compare and checks. Any parameters defined outside of the Rule and Check sections are treated as global parameters. Any locally defined parameters inside of the Rule and Check sections supersede defined global parameters.

The following parameters are available:

- a2d_threshold
- batchsize
- bus
- delay
- ignorex
- ignorez
- len_reltol
- length_check
- map
- mapfile, map_hier
- master
- match_type
- mean_range

- mean_step
- mean_tol
- pattern_check
- ref_clock
- report_style
- report_type
- report_vio_sig_only
- show_diff
- show_maxmin
- signal
- start, stop, step
- t_tol
- use_a2d_always
- use_sim_steps
- v_abstol, i_abstol, v_reltol, i_reltol
- when
- x_absmgn
- x_chkrange
- xval_precision

a2d_threshold

The a2d_threshold statement applies to both the master and the target files. It describes the voltage threshold value for converting an analog signal to a digital signal or the threshold range. This statement can be redefined in the check section to specify different thresholds for target signals. The default value is 1.5. The syntax for the a2d_threshold value is:

`a2d_threshold value` or
`a2d_threshold low_value high_value`

Note: You can also use the a2d_threshold statement in the check section for target file signals using the same syntax. For example:

```
alias num1 2
alias num2 3

rule rule1 begin
    master yu.tr0
    step 1n
    use_a2d_always
    a2d_threshold num1 num2
    xval_precision 16
end

check begin
    target yu.tr0
    signal v(2)
    a2d_threshold 1 4
    rule rule1
end
```

batchsize

The batchsize statement controls the number of signals that are compared at a time, which can save memory and improve performance. The syntax for the batchsize statement is:

```
batchsize value
```

For FSDB and WDF files, set `value` to `1`. For PSF or *.tr0 files, use the following formula to calculate the appropriate `value`:

```
value <= host_RAM / (filesize * 2 / num_of_signals)
```

For example:

```
batchsize 200
```

bus

The bus statement describes how a master logic bus value should be interpreted when compared to an analog target signal. The keywords used in the bus statement are (1) scale: specifies the scaling factor from the decimal bus value (2) signed/unsigned: specifies if the bus value is signed or unsigned.

(3) pwc/pwl: specifies if PWC or PWL interpolation should be used for value in between two bus value transitions. The syntax of the bus statement is:

```
bus bus_signal_name scale=value unsigned pwc
```

delay

The delay value applies to both the master and the target files. It describes the shift amount of x-axis value the user prefers to add to the original waveforms. The value can be redefined in the master or target file section. Positive delay value implies shifting the waveform toward the positive x-axis direction. The default value is zero. The syntax for the delay parameter is:

```
delay value
```

Note: If you use delay with use_sim_step, the following occurs:

- The waveform shifts by the specified delay value.
- The time points from sim_steps (in the [start stop] time range when specifying time_range) is compared

ignorex

The ignorex statement ignores x states during comparison.

ignorez

The ignorez statement ignores z states during comparison.

len_reltol

The len_reltol statement sets the relative length tolerance. The default is 0.01.

length_check

The length_check statement checks the length of a curve.

map

The map command is required when the compared signal name(s) differ between the master and target file(s). One map statement is required for each mapped signal. The syntax of the map statement is:

```
map master_signal_name target_signal_name
```

mapfile, map_hier

The mapfile statement specifies a separate map file, which you can create to define master and target signals for comparison. The syntax of the mapfile statement is:

```
mapfile <mapfile_name>
```

The following example is a sample map file:

```
;mapfile1
;Master_SigName     Target_SigName
node_1              node_2
node_3
node_5              node_5
node_6              node_7
```

In this example, `node_1` and `node_2`, as well as `node_6` and `node_7`, are compared as the Master Signal and Target Signal for each pair, respectively. The `node_3` and `node_5` are compared as the same name.

The map_hier statement maps the hierarchy of the defined signals. The syntax of the map_hier statement is:

```
map_hier <master_signal_hier> <target_signal_hier>
<master_signal_leaf_name> [<target_signal_leaf_name>]
```

The following example is a sample map file that uses the map_hier statement:

```
;commands inside the map file
map_hier testbench testbench.ifourphase
v(ck*)
end_hier
```

Note: The mapfile statement can only be specified in the Check section, and map_hier can only be specified within a mapfile statement.

master

The master statement declares the file path to be used as the golden file when comparing waveforms.

match_type

The match_type statement forces target signals to be the same type as the master signals.

mean_range

The mean_range statement performs a mean comparison for a derived set of waveforms based on the defined starting and stopping range values. The syntax for the mean_range statement is:

```
mean_range start stop
```

When a mean check is enabled, the mean version waveforms are first created for the master and target signals. Using the mean check does not affect the time points used for the check. If the time value falls in one of the mean_range periods, the mean waveforms are used instead of the original waveforms. Multiple mean_range statements can exist in a single check section.

mean_step

The mean_step statement samples waveforms and performs mean comparisons based on the defined step value. The syntax for the mean_step statement is:

```
mean_step step_value
```

The default value is 1ns. Mean comparisons are enabled as long as the mean_tol statement appears in the rule section.

mean_tol

The mean_tol statement performs a mean comparison based on the defined tolerance value. The syntax for the mean_tol statement is:

```
mean_tol value
```

The default value is 0.01.

pattern_check

The pattern_check statement searches for a pattern in logic signals. The syntax for the pattern_check statement is:

```
pattern_check <pattern>
```

For example:

```
pattern_check 101
```

ref_clock

A reference trigger clock signal can be used to determine the x-axis locations at which the comparison takes place. Two statements are required to define the reference trigger clock. Their syntax are:

```
ref_clock signal_name rise|fall|crossref_threshold value
```

The reference clock signal can be a logic or analog signal. For an analog reference signal, the ref_threshold value also needs to be defined. If the ref_threshold statement is omitted, the a2d_threshold value is used as the default. If ref_clock is defined in the rule file, the stop/start/step values are ignored.

report_style

The report_style statement defines the type of comparison report to produce. The syntax for the report_style statement is:

```
report_style value
```

For example, including the following line creates a report you can use with NanoSim:

```
report_style nanosim
```

report_type

The report_type statement defines what information is included in a comparison report. The syntax for the report_type statement is:

```
report_type value
```

You can specify the `value` as either `point` or `range` (`point` is the default value).

report_vio_sig_only

When the report_vio_sig_only statement is specified, only signals with violations are reported in the output file. The report_vio_sig_only statement can be in the global or rule section, and It applies to all the following comparison types:

- waveform comparison
- monotonicity check
- length check
- bound check
- envelope check
- pattern searching

The syntax for the report_vio_sig_only statement is:

```
report_vio_sig_only
```

show_diff

When the show_diff statement is specified, the analog value differences in percentage are reported in the output file.

For example:

```
rule v_check begin
  master inv.mt0
  use_sim_steps
  v_abstol 0.0001
  show_diff
  show_maxmin
end

check begin
  target inv.mt1
  signal *
  rule "v_check"
end
```

show_maxmin

When the show_maxmin statement is specified, the max, min, and mean of the analog value differences are reported in the output file.

For example:

```
rule v_check begin
  master inv.mt0
  use_sim_steps
  v_abstol 0.0001
  show_diff
  show_maxmin
end

check begin
  target inv.mt1
  signal *
  rule "v_check"
end
```

signal

The signal statement defines the signal to be compared and can be added in the Check section or used globally. Wildcards (* and ?) can be used in the name pattern. If the signal statement is omitted, all signals are compared. The syntax for the signal statement is:

```
signal name1 name2 … type=analog|logic|all -exclude
```

The `type` specifies the type of signal, and `-exclude` excludes the specified values from wildcard (* or ?) matching.

For example:

```
signal * -exclude i*
```

or

```
signal v* -exclude v(a1) v(a2)
```

start, stop, step

These values apply to the comparison process. The start and stop statements can be used globally, and the step statement can be defined in the Rule section or globally. All three can be redefined in the master file section. The default values are 0. Syntax for these values are:

```
start | stop | step value
```

t_tol

The t_tol value specifies the point just before the continuous violation time range is exceeded. If the continuous violation time range is larger than t_tol, then the violation is reported; otherwise, it is considered as a passing result.

The t_tol setting is only used when comparing one master versus one target, analog to analog comparison, and analog to digital comparison only. Setting t_tol to zero disables glitch filtering. The syntax for the t_tol statement is:

```
t_tol value
```

For example:

```
rule rule1 begin
    master target.tr0
    step 0.2n
    a2d_threshold 1.5
    t_tol 0.9n
    xval_precision 16
end

check begin
    target output_tb065.evcd
    signal v(a1)
    map v(a1) TESTBENCH.INST1.MEM1/MS_1T1C
    time_range 0 10n
    rule rule1
end
```

use_a2d_always

The use_a2d_always statement forces a comparison between two analog signals with converted logic values used for a2d_threshold.

use_sim_steps

The use_sim_steps statement uses the time steps saved in waveform files to determine where waveform comparisons occur on the x-axis.

v_abstol, i_abstol, v_reltol, i_reltol

The v_abstol, i_abstol, v_reltol, i_reltol values define the error tolerances for comparing analog waveforms. The values can be redefined in the target file section. The default values are:

```
v_abstol=50uv, i_abstol=1nA, v_reltol=0.001, i_reltol=0.01
```

The syntax for setting the tolerance values is:

```
v_abstol | i_abstol | v_reltol | i_reltol value
```

The final error tolerance used internally for waveform comparison is evaluated using the following equation

```
MAX(ABS(abstol),reltol*ABS(master_v))
```

Note: `v_abstol` and `v_tolerance` are equivalent statements. `i_abstol` and `i_tolerance` are also equivalent statements.

when

The when statement executes when the specified condition is met. If when is used in the global or rule section, it applies to the master file. If the specified signal is not located in the master file, CustomExplorer tries to find the signal in the target file. If when is used in the check section, it applies to the target file.

Only simple conditions (!= and ==) are supported, and only one signal can be specified. For example:

```
when "v(b) !=  1"
```

Arbitrary when statements are supported for logic signals.

x_absmgn

The x_absmgn value specifies the X axis tolerance margin and is used to compare analog signals. The syntax for the x_absmgn statement is:

```
x_absmgn value
```

The `value` without a plus (+) or minus (-) sign specifies bidirectional tolerance. Adding a plus or minus sign specifies a unidirectional tolerance (`x_absmgn +5n` or `x_absmgn -3n`, for example).

x_chkrange

The x_chkrange statement performs logic transition waveform comparisons based on the value, which searches for a signal transition within the defined window. The syntax for the x_chkrange statement is:

```
x_chkrange value
```

The `value` without a plus (+) or minus (-) sign specifies bidirectional checking. Adding a plus or minus sign specifies unidirectional checking (`x_chkrange +5n` or `x_chkrange -5n`, for example).

Note: The x_chkrange statement can be used in both rule and check sections when comparing digital signal transitions.

xval_precision

The xval_precision statement controls the x-value precision. The syntax for the xval_precision statement is:

```
xval_precision n
```

The `n` is the number of digits past the decimal point.

Controlling X- and Z-Level Constraints

X- and Z-level constraints change depending on the way you use some waveform comparison statements.

For example, when you use use_sim_steps and x_chkrange, the following X- and Z-level constraints are applied:

- The X and Z states remain as is and are compared. This is the default behavior.

  The default X and Z states change in the following manner if you use the following commands as described:

  - setx 0: Sets X-states to “Low” for comparison.
  - setx -1: Sets X-states to the previously known state for comparison.
  - setx 1: Sets X-states to “HIGH” for comparison.
  - setz 0: Sets Z-states to “Low” for comparison.
  - setz -1: Sets Z-states to the previously known state for comparison.
  - setz 1: Sets Z-states to “HIGH” for comparison.

Note: Using the ignorex or ignorez statements with the x_chkrange statement could produce unwanted results.

Defining Aliases

Aliases define values that can be later referenced in the Rule and the Check sections of the control file. For example, you can define the following aliases to substitute a reference string in a control file:

```
alias small_step "step=200ps"
alias medium_step "step=.5ns"
alias initial_period "start=0ns, stop=5ns"
alias stable_period  "start=5ns, stop=16n"
alias end_period  "start=16ns, stop=20n"
```

Defining Rules

The Rule section of the Control File defines rules that you can use in the Check section to perform the actual checks. Multiple rules can be defined in the Rule section, and each rule must be defined using the following syntax format:

```
rule rule_name begin
     parm1=parm1_value
     parm2=parm2_value
end
```

The waveform compare function samples and compares waveform data at the steps you define. Each comparison rule can contain one master file and comparison parameters. You can specify the absolute and relative tolerances for voltage or current signals, respectively. Voltage tolerances are used if the signal type is unknown. Warnings are issued if the difference between the master signal and the target signal exceeds the following condition:

```
abstol + reltol * master_signal
```

To use a “one-direction” relative tolerance for comparison, add a plus (+) or minus (-) sign in front of the relative tolerance value.

You can define the following parameters in the Rule section:

- Type of check represented by a keyword. The following checks are supported:
  - Sample-based waveform compare function.

    See Checking Waveforms for more information.
  - Waveform monotonicity check.

    See Checking Waveform Monotonicity for more information.
  - Waveform absolute bound check.

    See Checking Waveform Bounds for more information.
  - Waveform envelop check.

See Checking the Waveform Envelope for more information.

- Waveform Pattern Check

  See Checking the Waveform Pattern for more information.

- Comparison sampling step (overwrites global definition)
- Master file definition
- Check parameters such as tolerances and boundaries.

  For example, v_abstol, v_reltol, i_abstol, and i_reltol overwrites the defaults and Global definitions.

Note: Aliases can be used in comparison sampling steps and tolerance or boundary check parameters.

Checking Waveforms

The Waveform Compare Utility checks the rules specified in the check section sequentially on the specified target(s), time range, and signal list. Checking parameters can be redefined to and apply to the subsequent checks in the check section. The check is performed sequentially on all rules specified in the Check section and reports the violations for each check.

You can specify the following parameters in the Check section:

- Target file(s).

  Multiple target files can be delimited by spaces or commas. For example:

  ```
  target "t1.dat t2.dat"
  ```

- Check start/stop time (x-axis) range.

  The time_range statement defines the check start and stop times.

For example:

```
time_range 10n 20n
```

- Target signals.

  Signal names can contain the asterisk (*) or question mark (?) wildcards. For example:

```
signal "*"
```

- The rule to use.

  A predefined rule must be specified for each check. For example:

  ```
  rule "initial_v_check"
  ```

Waveform Check Example

The following is a sample waveform check:

```
check begin
     ;; value can be a pre-defined alias
     time_range  "0ns, 6ns"
     ;; the rule must have been defined in the rule section
     rule  "initial_v_check"

     ;; missing target files are automatically removed from the

     ;; target file list
     target  "t1.dat, t2.dat, t3.dat, t4.dat"
     signal  "s1,s2,s3"
     time_range  stable_period
     rule  "v_check"

     ;; check different time range
     target  "t1.dat, t2.dat, t3.dat"
     signal  "s1,s2,s3"
     time_range  end_period
     rule  "last_master"

     target  "t1.dat"
     signal  "s1"
     time_range  "5n,15n"
     rule  "mono_chk"

     ;rule  "bound_chk"


     target  "t1.dat"
     signal  "s1"
     ; varcond_target  "VDD=5:TEMP=100"
     rule    "env_chk"
end
```

Checking Waveform Monotonicity

The monotonicity check function screens each of the specified signals in the target file for Y-value magnitude inversion. The monotonicity check does not require any input parameter. A warning is issued when the monotonicity is reversed and restored.

```
rule mono_chk begin
     monotonicity_check
end
```

Checking Waveform Bounds

The bound check function screens every data point of the specified signals in the target file against the absolute lower and upper bounds defined in the `upper_bound` and `lower_bound` statements. A warning message is issued for each data point that exists outside of the bounds.

```
rule bound_chk begin
     upper_bound 6.2
     lower_bound 0.5
end
```

Checking the Waveform Envelope

The envelope check function screens every data point of the specified signals in the target file against the two 'bounding' signals defined in the `bound_sig1` and `bound_sig2` statements. A warning message is issued for each data point that exists outside of the bounds.

```
rule env_chk begin
     master gold1.dat
     bound_sig1 s1
     bound_sig2 s2
     ; master_sig1  sweep1.dat
     ; master_sig2  sweep2.dat
     ; varcond_sig1  "VDD=5:TEMP=0"
     ; varcond_sig2  "VDD=5:TEMP=100"
end
```

Checking the Waveform Pattern

The waveform pattern check searches for a specified pattern in logic signals. If the pattern is found, a message is issued with the x value where the pattern starts.

```
rule  pattern_chk begin
     pattern_check 101011
end

check begin
     target foo.vcd
     signal *
     time_range “0ns 40ns”
     rule “pattern_chk”
end
```

Rules Section Example

The following is a sample rules section:

```
rule initial_v_check begin
    master gold1.dat
    step   small_step
    v_tolerance 1mv
    v_reltol    +0.01
end

rule v_check begin
    master gold2.dat
    step   .1ns
    v_tolerance 1mv
;    delay 1ns
    v_reltol    -0.01
end
```

If a waveform compare rule contains no master file, the previous master file in the check sequence is used. For example:

```
rule last_master begin
    step .2ns
    v_tolerance 2mv
end
```

Saving Waveform Comparison Results

Waveform comparison results can be saved in a session file with the violated intervals highlighted by adding the -x session_file option to the sx -compare command. The operating system and CustomExplorer version information is also included in this file. You can load the session file into the Custom WaveView tool for further analysis.

Converting Signals to the Time Domain

To convert a signal to the time domain:

1. Right-click a plotted signal name in the waveview panel.
2. Choose **Signal '<name>' > To Time Domain** from the context menu that appears. The Convert to Time Domain dialog box opens.
3. Enter values (in seconds) for the x_Start, x_End, and x_Interval fields.
4. Click **Ok** to convert the signal to the time domain. The new time domain data is plotted to the waveview window.

Back-Annotating Signals to CustomExplorer

Note: A CustomExplorer license is required for this feature.

To back-annotate a signal to CustomExplorer, right-click the name of a signal and choose **Signal '<signal_name>' > Back Annotate to DesignView** from the menu that opens.

7

Post-Processing Waveforms

This chapter contains information on post-processing waveform data.

The Custom WaveView tool provides the following post-processing functions:

- FFT/DFT Conversion
- A to D Conversion
- D to A Conversion
- Applying .MEASURE Commands
- Generating Parametric Plots
- Using the RF Tool
- Performing Pareto Analyses
- Reducing Data Points
- Using the Jitter-vs-Time Tool
- Exporting Waveform Data
- Converting Signals to the Time Domain

All derived data are stored in a special type of waveform data file, which is added to the file list in the Output View hierarchy browser.

FFT/DFT Conversion

FFT and DFT operations are used to convert time-domain signals into their frequency-domain spectrum representation. The FFT/DFT operations can be performed on waveforms in the X-Y panels only.

FFT/DFT operations are supported when using various windowing functions. You can open the FFT/DFT dialog box by choosing **Tools > FFT/DFT Conversion** from the main menu or the **FFT/DFT** toolbar button. The FFT/DFT dialog box accepts input parameters and applies DFT/FFT to all signals in selected panels of the active waveview. If no panel is selected in the active waveview, all signals in the active waveview become the FFT/DFT target. If the active waveview has only one panel, the FFT/DFT operation is by default applied to all waveforms in the panel. If the active waveview has more than one panel, you need to select the FFT target panels.

The FFT/DFT algorithm requires the following parameters:

- The number of sample points: Determines the number of time-domain sample points used for FFT/DFT operation. This number is internally rounded to the next power of two that is greater than the user input for FFT. A slow run time warning is displayed for DFT operations over 1024 points.

  OR

  The sampling frequency: The frequency of the sampling signal for the FFT/DFT operation. The sampling rate must be greater than 1 hz.

  Note: If you have a defined time range, you need to specify either the number points or sampling rate. If you do not have a time range specified, the number of points and the sampling rate are used to determine the start and stop times.

- The sampling window start time: Determines the start time of the time-domain window for FFT/DFT conversion.
- The sampling window end time: Determines the end time of the time-domain window for FFT/DFT conversion. The specified window end time must be greater than its start time.

You must choose which three input parameters are in use. Deselect an input parameter using the toggle switch in front of the parameter. If the **Enable zero-padding after stop time** option is selected, all four parameters need to be entered, the sampling stop time is adjusted based on the other three input parameters. For example:

```
window_end_time < window_start_time + number_of_samples *
(1/sampling_frequency)
```

Evenly spaced zeros are padded between the user specified end time and the one calculated by the waveview.

Click the **Check** button to check the input parameters before a FFT/DFT operation. It automatically fills the unspecific value and modifies the end time if necessary when four sampling parameters are specified. Use the **Save** and **Load** buttons to save and load FFT input parameters respectively.

To enter the start and stop time based on the frequency of the input signal, enter the input signal frequency in the Input Signal Frequency field and click **Apply** to calculate the start/stop time.

Select the **Signed value** option to sample a logic signal using signed bus value. FFT tool uses PWC sampling if the FFT target is a PWC analog waveform or logic waveform. PWL sampling is used if the target is a PWL analog waveform.

Window functions can be applied for the FFT/DFT operation. If a window function is selected, the waveform data are multiplied with the selected windowing function before the FFT/DFT operation. No window function is used if the **Rectangle** window type is selected. The gaussian and the kaiser-bessel window types require the additional **Alpha** parameter.

The FFT/DFT output can be normalized against the waveform DC component or a user-defined value in Magnitude Normalization. All harmonic components are displayed in reference to the reference.

If the **Suppress DC before FFT/DFT** option is selected, the dc component is excluded from the target waveforms in the FFT/DFT operation. This is to minimize the error introduced by those signals with large DC component and very small AC magnitudes.

The Output Frequency range settings specify the range of FFT/DFT spectrums for display or text output. The fmin and fmax value define the output spectrum range of interest. Specifying **frequency of interest** only displays or outputs the multiple of this frequency harmonic; all other harmonics are omitted.

The FFT/DFT results can be displayed in a regular X-Y panel or in a spectrum panel. User can also send the results to either a text window or text file for further processing. Y-axis scale is automatically set as db for the FFT/DFT plot.

Select the **Display Sampled Waveform** option to display the time-domain sampled waveform, including the effect of the windowing function.

The FFT result can be displayed in a regular X-Y panel, or a special spectrum panel, or into text window or file. See Viewing the Spectrum Panel for more information.

This section contains information on the following topics:

- Viewing the Spectrum Panel
- Calculating SNR/THD Using FFT
- Calculating the FFT of Complex Signals
- Changing the FFT Axes Scale

Viewing the Spectrum Panel

The spectrum panel displays spectrum magnitude of harmonic components at different frequencies. Choose **Plot Complex** from the context menu to change the value to the phase, real, or imaginary part of the complex data.

The Spectrum panel uses bar chart by default to display waveforms. You can change it to a line chart by choosing **Configure Spectrum** from the panel context menu.

Cursors in the Spectrum panel report waveform values based on the current complex plot mode. Instead of using piece-wise-linear (PWL) interpolation, the piece-wise-constant (PWC) value is used within each spectrum interval.

Calculating SNR/THD Using FFT

To calculate and display the total harmonic distortion (THD), signal to noise ratio (SNR), signal to noise and distortion ratio (SNDR), and effective number of bits (ENB) of the target signals, select the **Evaluate SNR/THD/SNDR/ENB** option. The result is displayed in the text window or the text output file.

For SNR/THD/SNDR/ENOB calculation, windowing function is usually used if the sample range does not cover integer cycles of the input signal. However, the time-domain windowing functions also spread the original harmonics of the input signal spectrum into multiple spectrum components. To accommodate the spreading effect, an additional parameter (**Bin Size**) is required to describe how many spectral components, on each side of a harmonic frequency, should be included as the power of the harmonic.

Specify the cut-off frequency for the noise components in the Band Limit field. This value affects the SNR/SNDR/ENOB calculation. Set to 0 if no band limit is used.

Note: To include counting floor components when calculating SNR/SNDR, enable the **Count floor noise components in SNR/SNDR calculation** option, which is available in the Preferences (**Config > Preferences > Threshold Tab**).

Calculating the FFT of Complex Signals

If the input signal of FFT/DFT is a complex signal, a non-symmetric spectrum is generated. The **Suppress DC before FFT/DFT** option is ignored if the input signal is in complex values. To prepare a complex input signal from transient analysis (for RF-mixer applications), use the `complex()` equation builder function to combine two real signals into a complex signal.

Changing the FFT Axes Scale

To change the scale of the X- and Y-axis, choose **Config > Preferences** from the main menu bar, and click the Panel tab. Select the desired scales in the Default FFT X/Y Scale section, and click **OK**. These preferences are saved in the .spxrc preference file.

A to D Conversion

Analog waveforms can be converted to logic waveforms based on user-specified threshold values. The conversion procedure can be applied only to waveforms in the X-Y panels.

If the active waveview has only one panel, the A/D conversion operation is by default applied to all waveforms in this panel. If the active waveview has more than one panel, you need to select panels for the A/D conversion operation.

Choose **Tools > A to D Conversion** to invoke the A/D conversion operation or click the **A to D** toolbar button. The A/D Conversion Parameters dialog box opens and prompts you to accept the specified parameters for conversion.

This section contains information on the following topics:

- Single-Bit A/D Conversion
- Multi-Bit A/D Conversion
- Creating Bus Signals for Analog Signals

Single-Bit A/D Conversion

Analog waveforms can be converted to single-bit logic waveforms in one of two ways:

- With a single threshold defined, data points above the threshold level are converted to logic H, and those below to logic L.
- With two thresholds defined, all points above the high threshold level are converted to logic H, while all points below the low threshold level are converted to logic L. All other points in between are converted to logic X.

The converted waveforms are displayed in logic panels. Since logic panels are allowed only in vertical waveviews, a new vertical waveview is automatically created if the current active waveview is in horizontal mode.

The two-threshold conversion requires that a signal stays long enough in the X region (region between the high and low thresholds) in order to generate an X state in the converted logic waveform. The minimum duration requirement is defined by the rise (fall) slew if the signal level moves from below (above) the low (high) threshold level. The rise and fall slews are defined using the d2a_rise_slew and d2a_fall_slew configuration options.

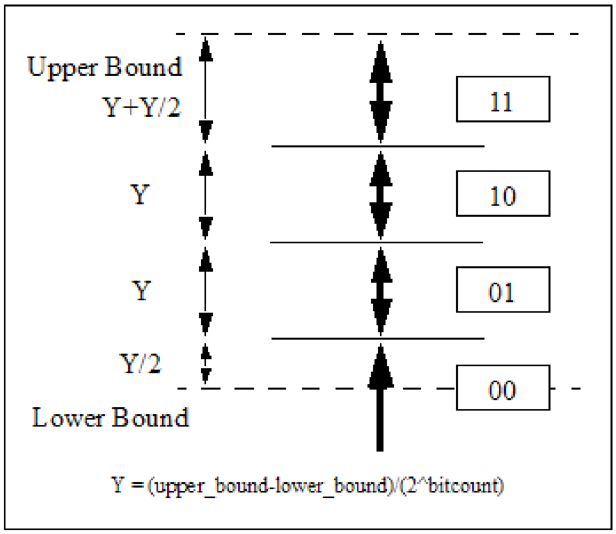
The default settings for the A to D operation can be specified in **Preference Settings > Conversion**.

Multi-Bit A/D Conversion

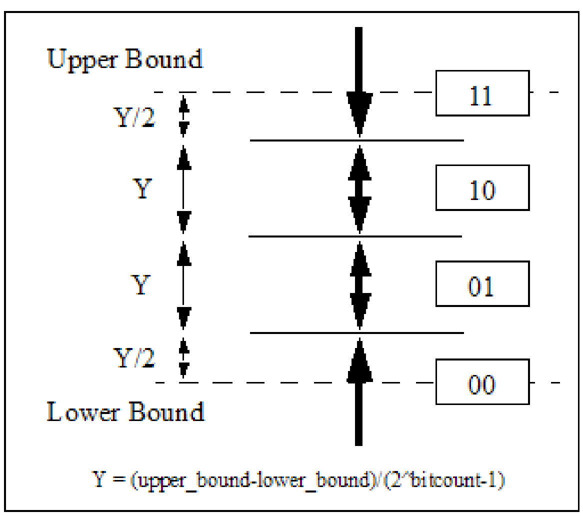
With multi-bit A/D conversion, analog waveforms are converted to multi-bit logic signals. The quantization thresholds are determined automatically based on the number of bits and user-specified lower/upper bounds.

Two quantization threshold schemes are available:

- Y = (upper_bound - lower_bound)/(2^bit_countt): Maintains asymmetric partitions as (Y+Y/2) for the largest codeword and Y/2 for the smallest codeword.



- Y = (upper_bound - lower_bound)/(2^bit_count -1): Maintains a symmetric partition as Y/2 for both the largest and smallest codewords. This option is selected by default.



Creating Bus Signals for Analog Signals

To create bus signals for analog signals when converting from analog to digital, click the **Create Bus with Selected Analog Signals** check box at the bottom of the A/D Conversion Parameters dialog box as part of your conversion setup.

D to A Conversion

Logic waveforms can be converted into analog waveforms based on user-specified analog levels. D/A conversion can be applied only to waveforms in the logic panels.

If the active waveview has only one panel, the D/A conversion operation is by default applied to all waveforms in this panel. If the active waveview has more than one panel, you need to select panels for the D/A conversion operation.

Choose **Tools > D to A Conversion** from the main menu to invoke the D/A conversion operation or click the **D to A** toolbar button. The D/A Conversion Parameters dialog box appears to accept parameters for the conversion operation.

Both single-bit and multi-bit D/A conversions are supported. Analog levels for multi-bit D/A conversion are determined based on rules. Enter the analog voltage levels for the low and high logic states.

There are two modes for D2A sampling: fixed rate sampling and value transition sampling. With the value transition sampling, the D-to-A conversion is completed only at the time when the input digital signal changes its value.

X and Z states can be converted to low or high voltage levels based on user preferences.

Click the **Extend to Full Display Range** button to extend the x-axis range to the current full x-axis display range for each converted analog waveforms, even if the original logic data has an x-axis range that is shorter than the full range.

Rising and falling slews of the converted analog waveforms can be specified in **Preference Settings > Conversion**, or using the **d2a_rise_slew** and **d2a_fall_slew** configuration options.

Applying .MEASURE Commands

You can apply `.MEASURE` statements in a SPICE netlist or apply a set of `.MEASURE` commands extracted from the netlist (the extraction capability is available in CustomExplorer only) to a simulation output file.

You can perform the following tasks when you apply `.MEASURE` commands:

- Take additional measurements on the target signals available in the simulation files without re-running simulation.
- Re-apply `.MEASURE` to different simulation output files.
- Speed up simulation job by running simulation without the `.MEASURE` statements in the input netlist. (A simulation usually takes longer and uses more memory when the netlist contains `.MEASURE` statements.)

The following HSPICE `.MEASURE` commands are supported:

- ```
  .MEASURE <TRAN|AC|DC> meas_name
  + AVG|RMS|PP|MIN|MAX|INTEG signal
  + <FROM=start_time TO=start_time>
  ```
- ```
  .MEASURE <TRAN|AC|DC>meas_name
  + TRIG signal_1 VAL=value TD=delay RISE|FALL|CROSS=index
  + TARG signal_2 VAL=value TD=delay RISE|FALL|CROSS=index
  ```
- ```
  .MEASURE <TRAN|AC|DC> meas_name
  + WHEN signal_1 = signal_2 TD=delay
  + RISE|FALL|CROSS=index
  ```
- ```
  .MEASURE <TRAN|AC|DC> meas_name
  + WHEN signal_1 = value TD=delay RISE|FALL|CROSS=index
  ```
- ```
  .MEASURE <TRAN|AC|DC> meas_name FIND  signal_3
  + WHEN signal_1 = signal_2 TD=delay
  + RISE|FALL|CROSS=index
  ```
- ```
  .MEASURE <TRAN|AC|DC> meas_name FIND  signal_3
  + WHEN signal_1 = value TD=delay RISE|FALL|CROSS=index
  ```
- ```
  .MEASURE <TRAN|AC|DC> meas_name FIND  signal_1
  + AT=time
  ```
- ```
  .MEASURE <TRAN|AC|DC> meas_name DERIVATIVE  signal_3
  + WHEN signal_1 = signal_2 TD=delay
  + RISE|FALL|CROSS=index
  ```

- `.MEASURE <TRAN|AC|DC> meas_name DERIVATIVE  signal_3`
  `+ WHEN signal_1 = value TD=delay RISE|FALL|CROSS=index`
- `.MEASURE <TRAN|AC|DC> meas_name DERIVATIVE  signal_1`
  `+ AT=time`
- `.MEASURE <TRAN|AC|DC> meas_name param('equation')`

The signals in the `.MEASURE` can be any of the following items:

- A voltage signal such as V(N1)
- A current signal such as I1(R1)
- An equation such as PAR('V(1) + 2*V(2)')

`.MEASURE` statements in a netlist can be applied to a waveform file by choosing **Apply Measure** from the OutputView context menu.

In the Apply Measure dialog box, enter values for the **Measure Command File** and **Measurement Result** text boxes for the measurement results. The **Measure Command File** can be a complete netlist with embedded `.MEASURE` statements, or a netlist file that has `.MEASURE` statements only without the actual circuit part.

Note: Since the `.MEASURE` command input file is processed as a SPICE netlist file, the first line of the input file is always ignored. Any .MEASURE command in the first line is ignored.

Alternatively, the HSPICE Measurement Tool can be used to perform the same measurement tasks interactively. Choose **Tools > HSPICE Measure Tool** from the main menu or **HSPICE Measure Tool** from the wdf context menu to invoke the HSPICE Measurement dialog box.

Reducing Data Points

Piecewise-linear waveforms often contain redundant data points. The data point reduction function allows users to remove redundant data points based on user-specified error tolerance. The same data reduction scheme is also used in WDF data reduction.

If the active waveview has only one panel, the reduction operation is by default applied to all waveforms in this panel. If the active waveview has more than one panel, you need to first select panels for the reduction operation.

Choose **Tools > Data Reduction** from the main menu to invoke the data point reduction operation or click the **Data Reduction** button in the toolbar. The Data Reduction Parameters dialog box appears, and the parameters are acquired for the data reduction operation.

Click **Ok** to start the data reduction operation. Waveforms with reduced data points are created in new panels in the same waveview.

Generating Parametric Plots

The following methods for generating parametric plots are available:

- Using the cursor in a 2D-sweep panel to plot the waveform value versus one of the sweeping parameters. Choose **Plot Y vs. X2** from the cursor context menu. To select a sweeping parameter, right-click the waveform file of the signal in the Output View and choose **X2 Parameter**. This method is limited to displaying the waveform value only.
- Use the parametric() function in the equation builder. The parametric() function takes two arguments. The first argument must be a measurement of a multi-trace waveform; the second argument is the sweeping parameter to be used as the x-axis in the parametric plot (for example, `parametric(mean("v(out)"),"TEMP"`). If the second argument is omitted, the default sweeping parameter is used.
- Use the built-in HSPICE parametric plot tool. The parametric tool only accepts HSPICE .MEASURE result files (.mtx/.msx/.max) as the input.

Built-in Parametric Tool for HSPICE .ALTER Simulations

Parametric plots directly from HSPICE measurement results files (.mt0/.ms0/.ma0 files) are supported. Choose **Tools > Parametric Plot** from the main menu to invoke the tool.

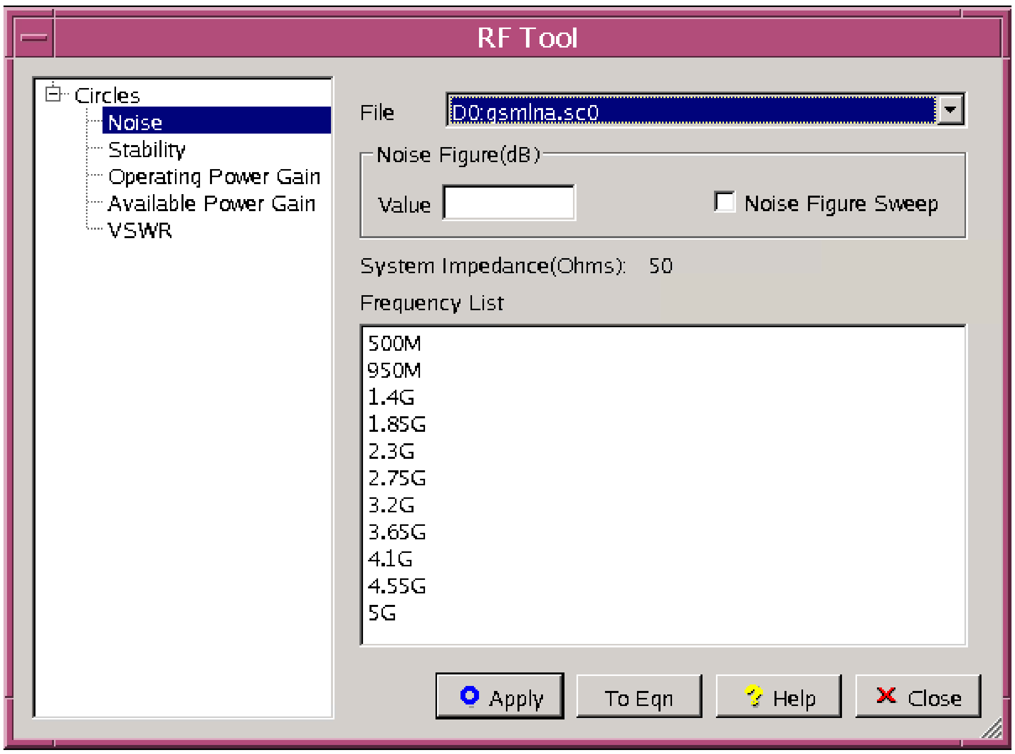
The Parametric Plot tool can be used to generate scatter plot with measure result annotation. Select the X/Y parameters, the **Scatter Plot** option, and click **Plot** to create the scatter plot.

Generic parametric plot functions can be realized using the parametric() function in the equation builder.

Using the RF Tool

The RF Tool takes special measurements or calculations for RF analyses. The RF analyses in the RF Tool are different from the measurements in Custom WaveView Measurement Tool—they are not applied to a specific signal, but to a specific waveform file.

To open the RF Tool, choose **Tools > RF Tool** from the Custom WaveView menu bar. The RF Tool window opens with the Noise Circle selected by default.



This section contains information on the following topics:

- Measuring Circles
- Exporting Equations from RF Tool Setup

Measuring Circles

An S-Parameter waveform file is required for all circle measurements. The **File** combobox at the top of the RF Tool window lists all currently open waveform files. If the specified file is an S-Parameter waveform file, the frequency values are extracted from the specified file and listed in the Frequency List.

The center point and radius values for each circle are calculated. 360 data points are created to make a circle with the center and radius.

The result circles are plotted on a Smith Chart. If a Smith Chart does not yet exist in any waveview, a new one is created, and the circles are plotted on it.

This section contains information on the following topics:

- Measuring Noise Circles
- Measuring Stability Circles
- Measuring Operating Power Gain Circles
- Measuring Available Power Gain Circles
- Measuring VSWR Circles

Measuring Noise Circles

To measure a noise circle in the RF Tool:

1. Click the **Noise** category on the left-hand side of the RF Tool window.

   The Noise Circle measurement settings are displayed.

2. Choose a file from the **File** menu that contains the noise circle you want to measure.

3. Enter a dB noise figure value in the Noise Figure text box, or click the **Noise Figure Sweep** check box if you want to include a noise figure sweep as part of the measurement.

   Note: System Impedance value is not an input value, but is the value of signal `Zo.Zo(1)` or `Zo.Zod(1)` from the selected waveform file. The value is displayed for informational purposes.

4. Select one or more frequency values by holding the **Ctrl** key and clicking frequency values in the Frequency List.

   If you want to select all frequency values, pres **Ctrl + A** to select all.

5. Click **Apply** to take the noise circle measurements and plot the results.

Measuring Stability Circles

To measure a stability circle in the RF Tool:

1. Click the **Stability** category on the left-hand side of the RF Tool window.

   The stability circle measurement settings are displayed.

2. Choose which S Parameter signals you want to include as part of the measurement in the S Parameters section.

   The associated signals and parameters are defaults from the selected waveform file. You can change the defaults signals by choosing different signals from the menus in the S Parameter section.

3. Select one or more frequency values by holding the **Ctrl** key and clicking frequency values in the Frequency List.

   If you want to select all frequency values, pres **Ctrl + A** to select all.

4. Choose the type of stability circle you want to plot. You can choose one or both types.

   Click the **Output Stability Circle** check box to plot an output stability circle waveform. Click the **Input Stability Circle** check box to plot an input stability circle waveform.

5. Click **Apply** to take the stability circle measurements and plot the results.

Measuring Operating Power Gain Circles

To measure an operating power gain circle in the RF Tool:

1. Click the **Operating Power Gain** category on the left-hand side of the RF Tool window.

   The operating power gain circle measurement settings are displayed.

2. Choose which S Parameter signals you want to include as part of the measurement in the S Parameters section.

   The associated signals and parameters are defaults from the selected waveform file. You can change the defaults signals by choosing different signals from the menus in the S Parameter section.

3. Choose one of the following gain values:

   - **Value**

Enter a gain value in the Value text box.

- **From/To Sweep**

  Enter from and to values in the From and To text boxes, respectively. Enter a Step value as well.

- **Gmax-based Sweep**

  Enter step and number values in the Step and Number text boxes, respectively.

4. Select one or more frequency values by holding the **Ctrl** key and clicking frequency values in the Frequency List.

   If you want to select all frequency values, pres **Ctrl + A** to select all.

5. Click **Apply** to take the operating power gain circle measurements and plot the results.

Measuring Available Power Gain Circles

To measure an available power gain circle in the RF Tool:

1. Click the **Available Power Gain** category on the left-hand side of the RF Tool window.

   The available power gain circle measurement settings are displayed.

2. Choose which S Parameter signals you want to include as part of the measurement in the S Parameters section.

   The associated signals and parameters are defaults from the selected waveform file. You can change the defaults signals by choosing different signals from the menus in the S Parameter section.

3. Choose one of the following gain values:

   - **Value**

     Enter a gain value in the Value text box.

   - **From/To Sweep**

     Enter from and to values in the From and To text boxes, respectively. Enter a Step value as well.

   - **Gmax-based Sweep**

     Enter step and number values in the Step and Number text boxes, respectively.

4. Select one or more frequency values by holding the **Ctrl** key and clicking frequency values in the Frequency List.

   If you want to select all frequency values, pres **Ctrl + A** to select all.

5. Click **Apply** to take the available power gain circle measurements and plot the results.

Measuring VSWR Circles

To measure a VSWR circle in the RF Tool:

1. Click the **VSWR** category on the left-hand side of the RF Tool window.

   The VSWR circle measurement settings are displayed.

2. Choose which S Parameter signals you want to include as part of the measurement in the S Parameters section.

   The associated signals and parameters are defaults from the selected waveform file. You can change the defaults signals by choosing different signals from the menus in the S Parameter section.

3. Choose the type of VSWR circle you want to plot. You can choose one or both types.

   Click the **Output VSWR Circle** check box to plot an output VSWR circle waveform, and enter a value for VSWROut. Click the **Input VSWR Circle** check box to plot an input VSWR circle waveform, and enter a value for VSWRIn.

4. Select one or more frequency values by holding the **Ctrl** key and clicking frequency values in the Frequency List.

   If you want to select all frequency values, pres **Ctrl + A** to select all.

5. Click **Apply** to take the VSWR circle measurements and plot the results.

Exporting Equations from RF Tool Setup

To export an equation of the setup from an RF measurement to the Equation Builder:

1. Ensure you set up all the necessary settings for an RF measurement.
2. Click the **To Eqn** button at the bottom of the RF Tool window.

The Equation Builder opens with the created RF measurement equation. For example, the following equation is exported from the settings for a noise circle measurement:

```
nc(“0|gsmlna.sc0”,”50 60 10”,50,”100M 200M 300M”)
```

Performing Pareto Analyses

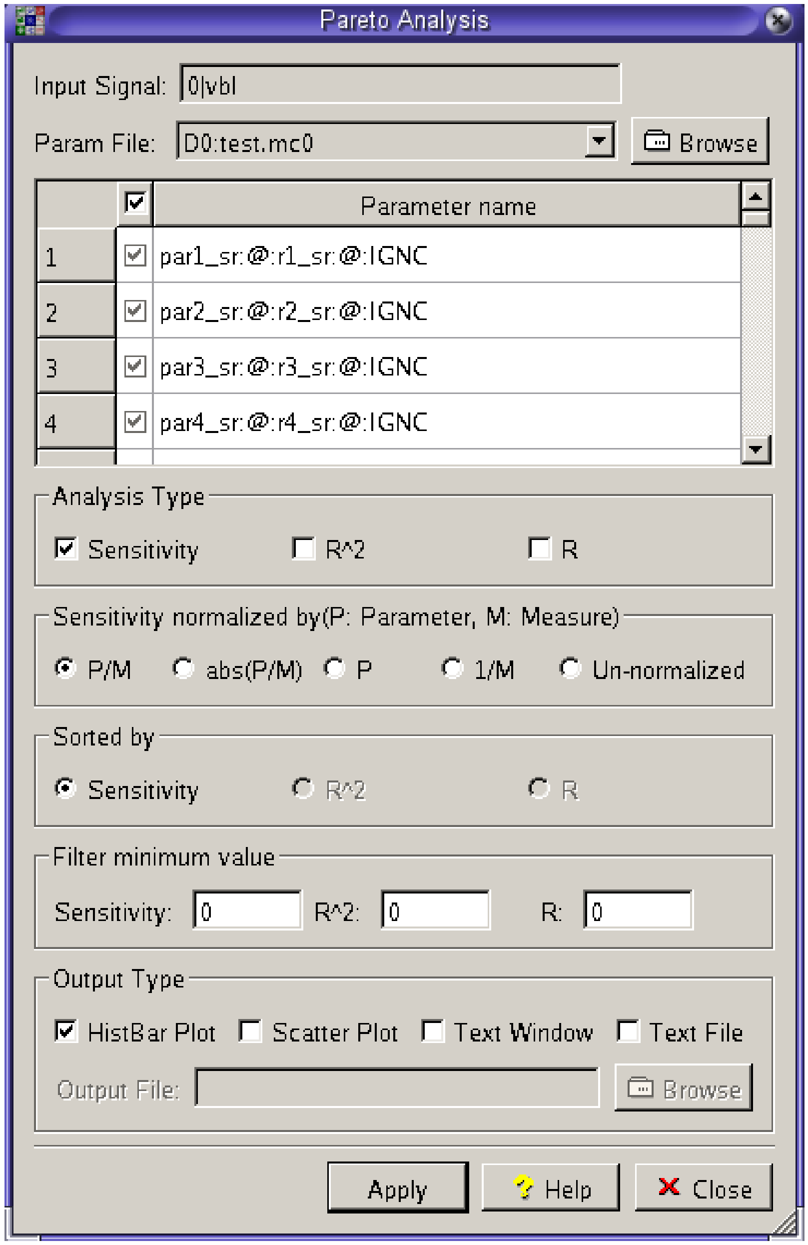
The Pareto Analysis tool can import your data mining results from HSPICE, performs sensitivity, R, and R^2 calculations, then produces results in one of the following formats:

- HistBar plot
- scatter plot
- text window plot
- text file

To perform a Pareto analysis:

1. Choose **Tools > Pareto Analysis** from the Custom WaveView menu bar.

The Pareto Analysis window opens.



2. Select an Input Signal by dragging it from the signal list and dropping it in the text box.

The input signal can be from an `*.mt` or `*.mc` file (which are files that contain parameter names and values for each trace you can load by clicking **Browse**), or from a measurement you apply to a sweep tr0 signal. The number of points in this signal need to be equal to the number of sweeps. Signals are not editable.

3. Ensure a Param File is selected.

   You can click the **Browse** button to browse to the location of a Param File, or enter the name in the Param File text box.

4. Choose which parameter signals you want to include by clicking the check box next to a signal name listed in the Parameter Name table.

   By default all signals are selected. You can deselect all signals by unchecking the check box in the heading row.

5. Choose one or more analysis types: **Sensitivity**, **R^2**, or **R**.

   Sensitivity analyses are calculated using the following function:

$$Sensitivity = \frac{\sum(x*y) - (\sum(x)*\sum(y)/n)}{\sum(x^2) - (\sum(x))^2/n} * \frac{\sum(x)}{\sum(y)}$$

   R^2 analyses are calculated using the following function:

$$Ajusted\ R - Square\ = 1 - \left(1 - \frac{(\Sigma(x - \bar{x}) * (y - \bar{y}))^2}{\Sigma((x - \bar{x})^2) * \Sigma((y - \bar{y})^2)}\right) * \frac{n-1}{n-p-1}$$

   R analyses are calculated using the following function:

$$R = \frac{\sum(x*y) - (\sum(x)*\sum(y)/n)}{\sqrt{(\sum(x^2)) - (\sum(x))^2/n)*(\sum(y^2) - (\sum(y))^2/n)}}$$

6. If you choose **Sensitivity** as the analysis type, select one of the following normalization factors:

   - **P/M**
   - **abs(P/M)**
   - **P**
   - **1/M**
   - **Un-normalized**

   Otherwise, skip to the next step.

7. If you selected more than one analysis, choose how you want your results sorted in the Sorted By section.
8. Enter values for the minimum filter value.

   If you select more than one analysis type, all the filter values are used together.
9. Choose which type of output you want.

   If you choose **Text File**, enter a name for the text file.
10. Click **Apply** to run the Pareto analysis.

Using the Jitter-vs-Time Tool

The built-in Jitter-vs-Time tool can be used to analyze clock jitter between any logic and analog target signal with respect to a reference signal. The reference signal can be an analog or logic signal from simulation output, or an idea signal defined using the width/period/delay parameters. The jitter can be measure against the rise and fall edges of a reference signal. To invoke the tool, choose **Tools > Jitter vs. Time** from the main menu.

Selecting Reference Signals

The reference signal can be selected from the following sources:

- User-defined Ideal Clock: An ideal clock signal defined by delay, period, and width.
- A Digital Signal: A digital simulation output waveform.
- An Analog Signal: An analog simulation output waveform.

The Digital or Analog Signal field requires an external waveform that can be specified in the External Signal field. Drag and drop the external signal from the output view browser to the input field. If an analog reference signal is used, threshold value for the rise and fall edges also needs to be specified.

If the **Display Reference Clock Signal** option is selected, the reference signal is shown together with the jitter-versus-time output waveform.

Specifying the Target Signal

The target signal can be either a digital or an analog simulation output waveform. The analog target waveform requires the rise/fall threshold values to be specified.

Selecting Reference Edges

User can select the rise, fall, or both edges of the reference signal as the reference edge for the jitter evaluation.

Target Signal Edges

The following options are available:

- **Same as Reference**: Finds rise-to-rise and fall-to-fall jitters.
- **Rise**: Finds jitter from reference edge to only the rising target edge.
- **Fall**: Finds jitter from reference edge to only the falling target edge.
- **Cross**: Finds jitter from reference edge to the next target switching edge (rise or fall).

The Target Signal Edge options are enabled only if the Jitter Type option is set to **Trigger Time**.

Jitter Types

The Jitter vs. Time tool can evaluate jitter for the following types:

- **Trigger Time**: Measures the edge differences between the trigger edge and the signal edge.
- **Pulse Width**: Measures the pulse differences between the reference signal and the target signal at the reference edge.
- **Period Width**: Measures the period differences between the reference signal and the target signal at the reference edge.

By default, the target signal edges are searched in both directions from a reference switch edge. Select the **Always Search Forward for Target Edge** option to force the target search in the forward direction only.

The default target edge search range is half of the clock period at the current reference clock cycle. To limit the maximum search time range, enter the limit in the Max. Target Edge Search Range field.

Jitter Output Options

The Jitter vs. Time result is displayed in a linear X-Y panel. PWL waveforms and bar charts are the available plot types. The jitter value sign can be selected between forward-positive or backward-positive. In the forward positive mode, jitter values are positive if the target edge appears later than the reference edge.

Exporting Waveform Data

To export waveform data, choose **File > Export Waveform Data** from the main menu bar and select a file in which to save the waveform data. Confirm your output file path to open the Waveform Export Parameters dialog box.

Waveform data in the active waveview can be saved using several different formats. The following formats are supported:

Format	Description
Text Table	Tabulated format. You can specify output options in the Text Table Output Option section of the Waveform Export Parameters dialog box. Selecting the **Suppress X-axis** column option hides the x-axis in the table output. Selecting the **Hex Logic Values** option changes the binary logic output values to hexadecimal values. Selecting the **Do not extend waveforms over the longer x-range of other waveforms** option prevents waveforms from being extended over an x-range of another waveform that is longer.
PWL Source	SPICE PWL (piecewise-linear) format. You can specify PWL logic levels and slew values in the Logic Waveforms to PWL Sources section, as well as any netlist syntax options in the PWL Source Netlist Syntax Options section of the Waveform Export Parameters dialog box.

Format	Description
WDF	Synopsys WDF compressed format. You can specify the **Set Start X** and **Set Stop X** values in the Text Table Output Option section of the Waveform Export Parameters dialog box.
VCD	VCD file with analog support. You can specify the VCD Timescale in the lower right-hand corner of the Waveform Export Parameters dialog box.
M-File	MATLAB file. You can specify the Number of Valid Digits in the lower left-hand corner of the Waveform Export Parameters dialog box.

By default, the Custom WaveView tool only exports waveforms in selected panels using the X-axis step of the very first signal. Click **All Panels** to export waveforms in all panels.

Converting Signals to the Time Domain

To convert a signal to the time domain:

1. Right-click a plotted signal name in the waveview panel.
2. Choose **Signal '<name>' > To Time Domain** from the context menu that appears. The Convert to Time Domain dialog box opens.
3. Enter values (in seconds) for the x_Start, x_End, and x_Interval fields.
4. Click **Ok** to convert the signal to the time domain. The new time domain data is plotted to the waveview window.

Back-Annotating Signals to CustomExplorer

Note: A CustomExplorer license is required for this feature.

To back-annotate a signal to CustomExplorer, right-click the name of a signal and choose **Signal '<signal_name>' > Back Annotate to DesignView** from the menu that opens.

8

Using the Equation Builder

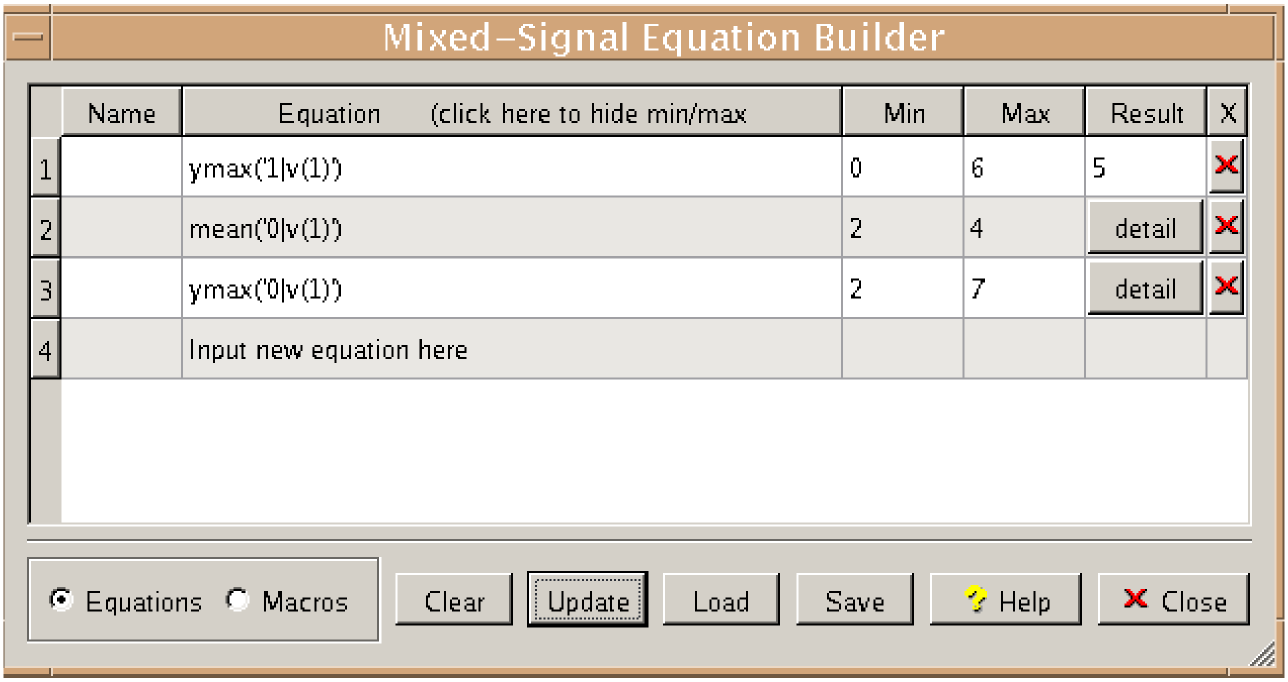
This chapter contains information on using the equation builder.

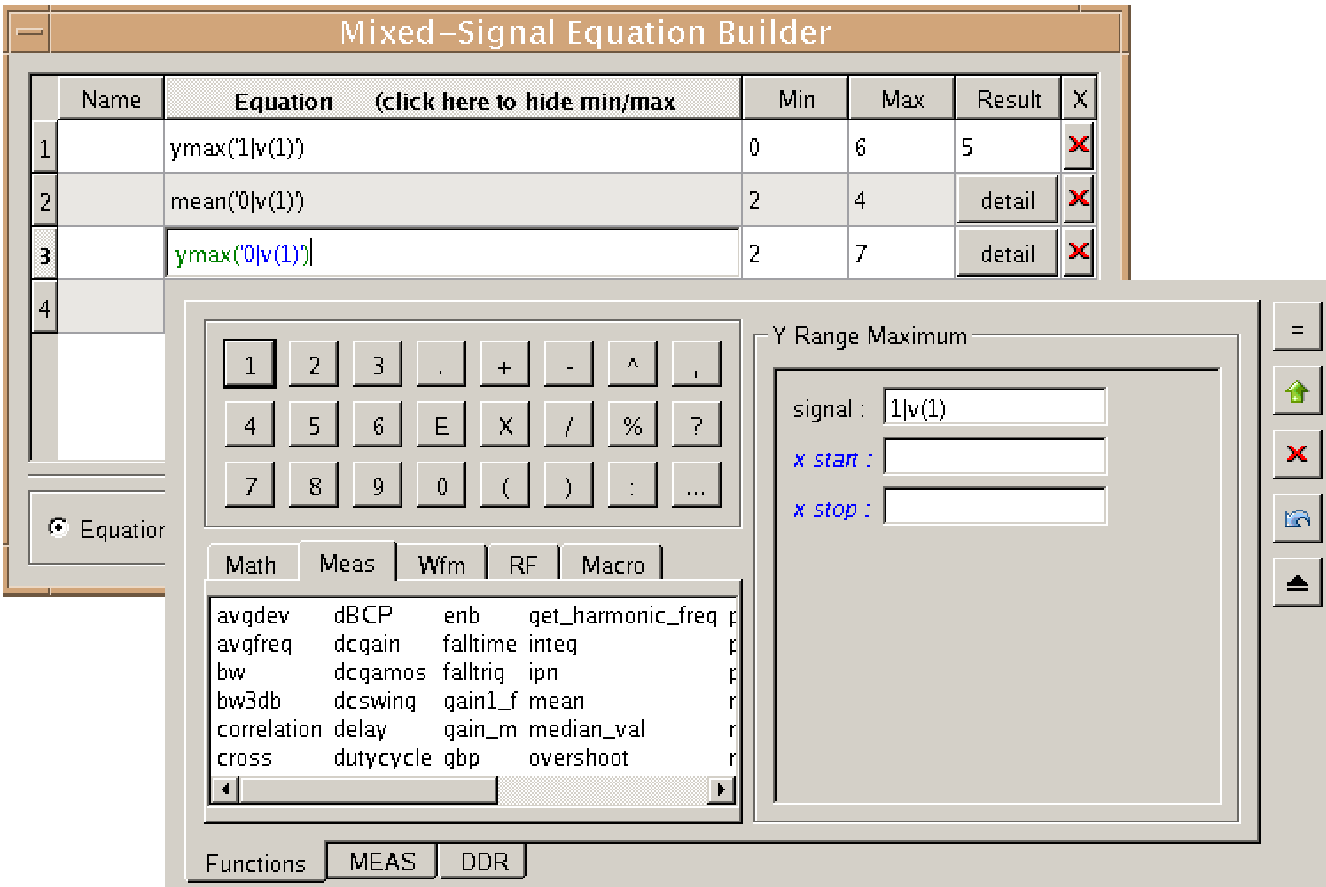
The equation builder in the Custom WaveView tool facilitates multi-file mixed-signal waveform operations and measurement. Choose **Tools > Equation Builder** from the main menu or the **Equation Builder** toolbar button to start the equation builder.

The equation builder can generate output as scalar measurement results, new waveform data, text output tables, or user-defined input equations. You can type input equations into the equation editor or construct them by dragging and dropping from the Output View signal browser.

For Equation Builder function descriptions and syntax, see the Supported

Equation Builder Functions section.





The following topics are available in this chapter:

- Building Equations
- Adding Signals to the Equation Builder
- Assigning Aliases to Equations
- Viewing the Result Stack
- Adding .MEASURE Statements to Expressions
- Defining Macros
- Modifying Equations
- Calculating Multi-trace Waveforms
- Special Note Regarding FFT/DFT
- Supported Equation Builder Functions

Building Equations

When building or editing an equation, the function name is displayed in green, the operator is displayed in red, and the signal name is displayed in blue. Unmatched parenthesis are highlighted with red background color.

When moving cursors in an equation, matched parenthesis are highlighted with a green background color if the cursor is in front of a parenthesis. A red underline is added under arguments with cursors that are on or in front of a function name.

Adding Signals to the Equation Builder

To add a signal to the equation, drag-and-drop the signal from the Output View browser into the equation builder window, or right-click signal in the Output View browser and choose **Add "SIGNAL" To Equation** from the context menu that appears.

Signals can be also typed into the equation field in the following format, where file_index is the zero-based sequential index of the waveform data file which contains the signal 'signal_name':

```
'file_index|signal_name'
```

If file_index is omitted, the first matched signal from all waveform files is used.

If an equation is required to find a signal using case-sensitive name search, use the `wave()` function to access the signal. For example:

```
wave('I(Vdd)')
```

The equation builder accepts numbers in scientific or scale format (for example, 1.0e-12, 10n, 0.23m).

Assigning Aliases to Equations

To assign a name to an equation, start the equation with the `NAME=` prefix. For example:

```
POWER='0|I(VDD)' * '0|VDD'
```

The equation can be referred to in subsequent operations using this alias name.

Viewing the Result Stack

The result stack below the equation builder keypad displays results from previous calculations in the order they were entered. By default the result stack displays the first equation on the top of the stack, click the **Reverse** toggle button to reverse this order so the latest result is displayed on the top of the result stack.

To recall a previously entered equation, click the equation in the result stack. Items in the result stack can be deleted or cleared using the **Delete** and the **Clear** buttons. Click the **Save** button to save the result stack (sequence) in the ACE format. Saved results can be restored using the **Load** button, or choosing **File > Run ACE Program** from the main menu. To change the alias name of an equation output, click the **Name** button to modify the name.

Adding .MEASURE Statements to Expressions

To add a .MEASURE statement to an expression:

1. Ensure the Equation Builder is open (**Tools > Equation Builder**).
2. Click the Measure tab.

   The .MEASURE options are displayed.
3. Enter information for any needed .MEASURE options.

   See the *HSPICE Reference Manual: Commands and Control Options* for further information on the .MEASURE statement.
4. Click the green arrow to save your changes.

Defining Macros

Macros are available for those times when you want to repeatedly take measurements with equations created in the Equation Builder.

To define a macro:

1. Ensure the Equation Builder is open (**Tools > Equation Builder**).
2. Click the Macro tab.

   The Macro tab page is displayed.
3. Click the `Input macro definition here` text, and enter the definition for the macro in the text box.

   Macro definitions with an arbitrary number of input arguments are supported. For example:

   ```
   FNEW(a,b) = a*b - 2 PI=3.1415
   ```
4. Enter a name for the macro in the Name (Args) column for the macro you are creating.
5. Press **Enter**.

   The macro is created.

Macro definitions can be imported or exported by clicking the **Load** and **Save** buttons.

Modifying Equations

To modify an equation, choose **Signal "Name" > Modify** from the Output View context menu. Choose **Signal "Name" > Edit Alias** from the Output View context menu to rename the equation to a new alias name.

Calculating Multi-trace Waveforms

The equation builder supports operations on multi-trace sweep waveform data. Multi-trace waveforms are generated from parametric sweep analysis. An individual trace in a multi-trace waveform is usually associated with a set of parameters.

When applying an equation to a multi-trace waveform, the equation builder operation is automatically repeated to every trace. As a result, multiple waveforms are generated from the operation.

For scalar measurement, a new measure-versus-trace signal is generated. The `parametric()` function can then be applied to the new measure-versus-trace signal to plot the measure result versus a selected parameter. For example:

```
AVG=mean('I(VDD') parametric(AVG,TEMP)
```

or

```
parametric(mean('I(VDD)',TEMP)
```

Special Note Regarding FFT/DFT

Before release 2005.4, the FFT/DFT function in the Custom WaveView tool uses ejwt for forward FFT calculation and e-jwt for inversed Fourier transform (IFT) calculation. However, this convention differs from the ones used in SPICE simulators. From 2005.4, the complex conjugate has been reversed between the FFT and IFT; for example, e-jwt is now used for FFT and ejwt for IFT.

For applications that need to combine a transfer function (`tf`) from an AC simulation to the the Custom WaveView tool FFT result, the following modification is required for 2005.3 and earlier versions:

```
complex(real("tf"),-imag("tf")) * fft(….)
```

This modification is not required with version 2005.4 and later.

Supported Equation Builder Functions

The following equation builder functions and operators are supported:

- Supported Operators
- Supported Mathematic Functions
- Supported Measurement Functions
- Supported Waveform Functions
- Supported RF Functions
- Supported Logic Operations

Supported Operators

The following table summarizes the operators that the equation builder supports.

Operator/Function	Alias	Description
+		Addition operator
-		Subtraction operator
*		Multiplication operator
/		Division operator
^	**	Power operator
?		Ternary operator
%		Integer operator

Mathematic functions can operate on scalars or waveforms. The returned value can be a scalar (or a waveform) if its input argument is a scalar (or signal).

Supported Mathematic Functions

The following table summarizes the supported Mathematic functions in the Equation Builder.

Function	Description
abs(x)	Returns the absolute value of x
acos(x)	Returns the arc cosine value of x
asin(x)	Returns the arc sine value of x
atan(x)	Returns the arc tangent value of x
complex(x,y)	Returns a complex scalar (or waveform) with x as the real part and y as the imaginary part.

Function	Description
cos(x)	Returns the cosine value of x
cosh(x)	Returns the hyperbolic cosine value of x
db10(x)	Returns 10*log10(x)
db20(x)	Returns 20*log10(x)
dbm(x)	Returns 30+10*log10(x)
exp(x)	Returns e**x
floor(x)	Returns the largest integer not greater than x.
imag(x)	Returns the imaginary part of scalar/signal x, returns 0 if x is a real scalar/signal.
int(x)	Returns the rounded integer of x. (int(-1.2) = -1)
ln(x)	Returns the natural logarithm of x, or 0 if x < 0
log10(x)	Returns the base-10 logarithm of x, or 0 if x < 0
mag(x)	Returns the magnitude of scalar/signal x.
max(x,y)	Returns the larger value between x and y.
min(x,y)	Returns the smaller value between x and y
phase(x)	Returns the phase of scalar/signal x
pwr(x,y)	Returns x**y if x >= 0, -x**y if x < 0
real(x)	Returns the real part of scalar/signal v
round(x)	Rounds the number x to an integer value (for example, round(1.49)=1 and round(1.5)=2).
sgn(x)	Returns 1 if x > 0, -1 if x < 0, 0 if x = 0
sin(x)	Sine function
sinh(x)	Hyperbolic sine function

Function	Description
sqrt(x)	Returns the square root of x if x >= 0, 0 if x < 0
tan(x)	Tangent function
tanh(x)	Hyperbolic tangent function

Supported Measurement Functions

The following table summarizes the measurement functions in the equation builder. Measure functions must operate on waveforms (or derived waveforms). They always return scalar values.

Function	Description
avgdev(s)	Calculates the average deviation of the given signal using the following formula: sum(abs(Vi - average_value)) / (n-1) i = 0,1, ..., n-1
avgfreq(s,<x1>,<x2>)	Returns the average frequency of s from x1 to x2.

bw(s,db,<type>,<topline>, <percent>)	Returns bandwidth of signal s at which the signal strength drops below bound-dB from its DC value (see the figures at the end of this section).
	■ s: Target signal.
	■ db: The dB threshold.
	■ <type>: The type of measurement (lowpass, highpass, bandpass, or stopband). Default is lowpass.
	For example:
	`bw('0|br',3,highpass,detect)`
	■ <topline>: The top line reference (firstpoint, detect, a dB value, or a percentage value). Default is firstpoint
	For example:
	`bw('0|br',3,lowpass,firstpoint)`
	■ <percent>: The percentage value (0 or 1). When specified as 1, the <topline> is a percentage value.
	For example:
	`bw('0|br',3,stopband,-1.20413)`

<table>
<tr><th>Function</th><th>Description</th></tr>
<tr><td>bw3db(s,<type>,<topline>, <percent>)</td><td>bw3db(s) = bw(s,3)

s: Target signal.
<type>: The type of measurement (lowpass, highpass, bandpass, or stopband). Default is lowpass.
For example:
<code>bw3db('0|br',highpass,detect)</code>
<topline>: The top line reference (firstpoint, detect, a dB value, or a percentage value). Default is firstpoint
For example:
<code>bw3db('0|br',lowpass,firstpoint)</code>
<percent>: The percentage value (0 or 1). When specified as 1, the <topline> is a percentage value.
For example:
<code>bw3db('0|br',stopband,-1.20413)</code>
</td></tr>
<tr><td>correlation(s)</td><td>Returns the correlation coefficient of the Y values of s versus the X values of s.</td></tr>
<tr><td>cross(s,y,n,<xstart>, <percent>)</td><td>Returns the x value of the nth crossing edge over threshold v beginning at xstart. When <percent> is set to <code>1</code>, the y threshold value is considered a percentage value.</td></tr>
<tr><td>dBCP(s,n,x,<referred>)</td><td>Calculates a 1 DB Compression Point.</td></tr>
<tr><td>dcgain(s,min_gain)</td><td>Returns DC gain at min_gain (see the figures below).</td></tr>
<tr><td>dcgamos(s,output_swing, <percent>)</td><td>Returns DC gain, of signal s, at maximum output swing (see the figures at the end of this section). When <percent> is set to <code>1</code>, the y threshold value is considered a percentage value.</td></tr>
<tr><td>dcswing(s,min_gain)</td><td>Returns DC output swing at min_gain (see the figures below).</td></tr>
</table>

Function	Description
delay(s1,s2,y1,y2,n1,n2,<x>, <percent>)	Calculates the time difference between two signals at specified switching edges by edge indexes. ▪ s1: Trigger signal. ▪ s2: Target signal. ▪ y1: Threshold level for trigger signal. ▪ y2: Threshold level for target signal. ▪ n1: Trigger index x value. ▪ n2: Target index x value. ▪ x: Starting x value. ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value.
dutycycle(s,yval,rise,<n>, <percent>)	Calculates the duty cycle of a given signal at specified edges by edge indexes. ▪ s: Target signal. ▪ yval: Y value. ▪ rise: Use rising edge flag. ▪ n: nth index ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value.
enb(s,x1,x2,n,fin,<fmax>, <win>,<deci>)	Returns the Effective Number of Bits of signal s. The FFT evenly PWC samples s from x1 to x2 for n points. Fin is the input signal frequency. The optional argument fmax sets the band limit, win specifies the windowing function ENB=(SNR-1.76)/6.02, and deci specifies the decimation rate.
falltime(s,high,low,<xstart>, <percent>)	Returns the time difference of signal s from the high threshold crossing to the low threshold crossing beginning at xstart (see the figures below). ▪ s: Target signal. ▪ high: The high threshold crossing. ▪ low: The low threshold crossing. ▪ xstart: The starting x value. ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value.

Function	Description
falltrig(s,y,n,<xstart>, <percent>)	Returns the x value of the nth falling edge over threshold y. When <percent> is set to `1`, the y threshold value is considered a percentage value. ▪ s: Target signal. ▪ y: Y level. ▪ n: Edge index. ▪ xstart: The starting x value. ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value.
gain1_f(s)	Returns the frequency of signal s at which the signal's magnitude is 0db (1)
gain_m(s)	Returns the gain margin of signal s
gbp(s,bound)	Returns gain-bandwidth product of signal s at which the signal strength drops below bound-db from its DC value (see the figures below).
get_harmonic_freq(s,n)	Returns the frequency of the nth harmonic. ▪ s: Target signal. ▪ n: 0-based index
integ(s,<x1>,<x2>)	Returns the integral value of signal s from x1 to x2.
ipn(s1,s2,n,x,<referred>)	Returns an N-Order intercept Point. ▪ s1: Target signal. ▪ s2:Reference signal. ▪ n: Term order. ▪ x: Extrapolation point. ▪ referred: Returned value option, “input” or “output”.
mean(s,<x1>,<x2>,<type>)	Returns the mean value of signal s from x1 to x2. The `type` can be `discrete` or `continuous`. If `type` is not specified, the default is `continuous`.

Function	Description
median_val(s)	Calculates the median value of the given signal using the following algorithm: 1. Signal values are sorted. 2. If the number of signal values is odd, then median_val = the value at index (n+1)/2 = V((n+1)/2). If the number of signal values is even, then median_val = (V(n/2) + V(n/2+1))/2
overshoot(s,low,high, <percent>)	Returns the overshoot of signal s, in fraction of (high-low). That is, (max(signal) - high)/(high-low) (see Figure 11, Figure 12, and Figure 13). When set to `1`, the y threshold value is considered a percentage value.
paramvalue(p)	Returns the value of parameter p. This is used mainly with multi-trace waveforms to extract the sweeping parameter value. The returned value varies from trace to trace in a multi-trace sweeping signal.
peak2peak(s,<x1>,<n>, <minpp>)	Calculates the peak to peak value of a signal. ■ s: Input signal. ■ x1: Starting x value. ■ n: nth index. ■ minpp: Min p2p value.
phase_m(s,<basedeg>)	Returns the phase margin of signal s. <basedeg> can be either -180 or 0. By default, the value is -180.
risetime(s,lv,hv,<xstart>, <percent>)	Returns the time difference of signal s from low threshold crossing to high threshold crossing beginning at x1 (see Figure 11, Figure 12, and Figure 13). ■ s: Target signal. ■ lv: The low threshold crossing. ■ hv: The high threshold crossing. ■ <xstart>: The starting x value. ■ <percent>: When set to `1`, the y threshold value is considered a percentage value.

Function	Description
risetrig(s,y,n,<x>,<percent>)	Returns the x value of the nth rising edge over threshold y. When <percent> is set to `1`, the y threshold value is considered a percentage value.
rmsval(s,<x1>,<x2>)	Returns the RMS value of signal s from x1 to x2.
settling_t(signal,xstart,xstop,<percent>,<xend>)	Returns the settling time of the signal (see see Figure 11, Figure 12, and Figure 13). When xend is specified, the function calculates the settle time in a sub-window waveform.
slewrate(s,initv,finalv,<percent>)	Returns the slew rate (see the figures below). ■ s: The target signal. ■ initv: The initial crossing threshold. ■ finalv: The final crossing threshold. ■ <percent>: When set to `1`, the y threshold value is considered a percentage value.
slope(s,x)	Returns the slope value of signal s at x
sndr(s,x1,x2,n,fin,<fmax>,<win>,<deci>)	Returns the signal to noise-and-distortion ratio in db of signal s. The FFT evenly PWC samples s from x1 to x2 for n points. Fin is the input signal frequency. The optional argument fmax sets the band limit, win specifies the windowing function, and deci specifies the decimation rate.
snr(s,x1,x2,n,fin,<fmax>,<win>,<deci>)	Returns the signal-to-noise ratio in db of signal s. The FFT evenly PWC samples s from x1 to x2 for n points. Fin is the input signal frequency. The optional argument fmax defines the band limit, win specifies the windowing function, and deci specifies the decimation rate.. Valid window type strings (case insensitive) are: rectangle, bartlett, parzen, welch, hanning, hamming, blackman, and blackharris.
snr_f(s,sf,st,nf,nt)	Returns the signal-to-noise ratio in dB of waveform s, which is the resulting waveform of an FFT. The sf and st arguments define the signal range around each harmonic, and the nf and nt arguments define the noise range around each harmonic.

Function	Description
stddev(s)	Returns the standard deviation of the Y values of the input signal s. X value of s is discarded.
thd(s,x1,x2,n,fin,<fmax>, <win>,<ish>,<deci>)	Returns the total harmonic distortion in db of signal s. The FFT evenly PWC samples s from x1 to x2 for n points. Fin is the input signal frequency. The optional argument fmax sets the band limit, win specifies the windowing function, ish specifies the maximum harmonic index flag, and deci specifies the decimation rate.
variance(s)	Returns the variance of the Y values of the input signal s. X value of s is discarded.
x_at_ymax(s,<x1>,<x2>)	Returns the x value at the maximum y value of signal s between x1 and x2.
x_at_ymin(s,<x1>,<x2>)	Returns the x value at the minimum y value of signal s between x1 and x2.
xmax(s)	Returns the maximum value in the x-axis range of a given signal.
xmin(s)	Returns the minimum value in the x-axis range of a given signal.
ymax(s,<x1>,<x2>)	Returns the maximum y value of signal s between x1 and x2
ymin(s,<x1>,<x2>)	Returns the minimum y value of signal s between x1 and x2
yvalue(s,x)	Returns the y value of signal s at x

The following figures illustrate the implementation of the measurement functions.

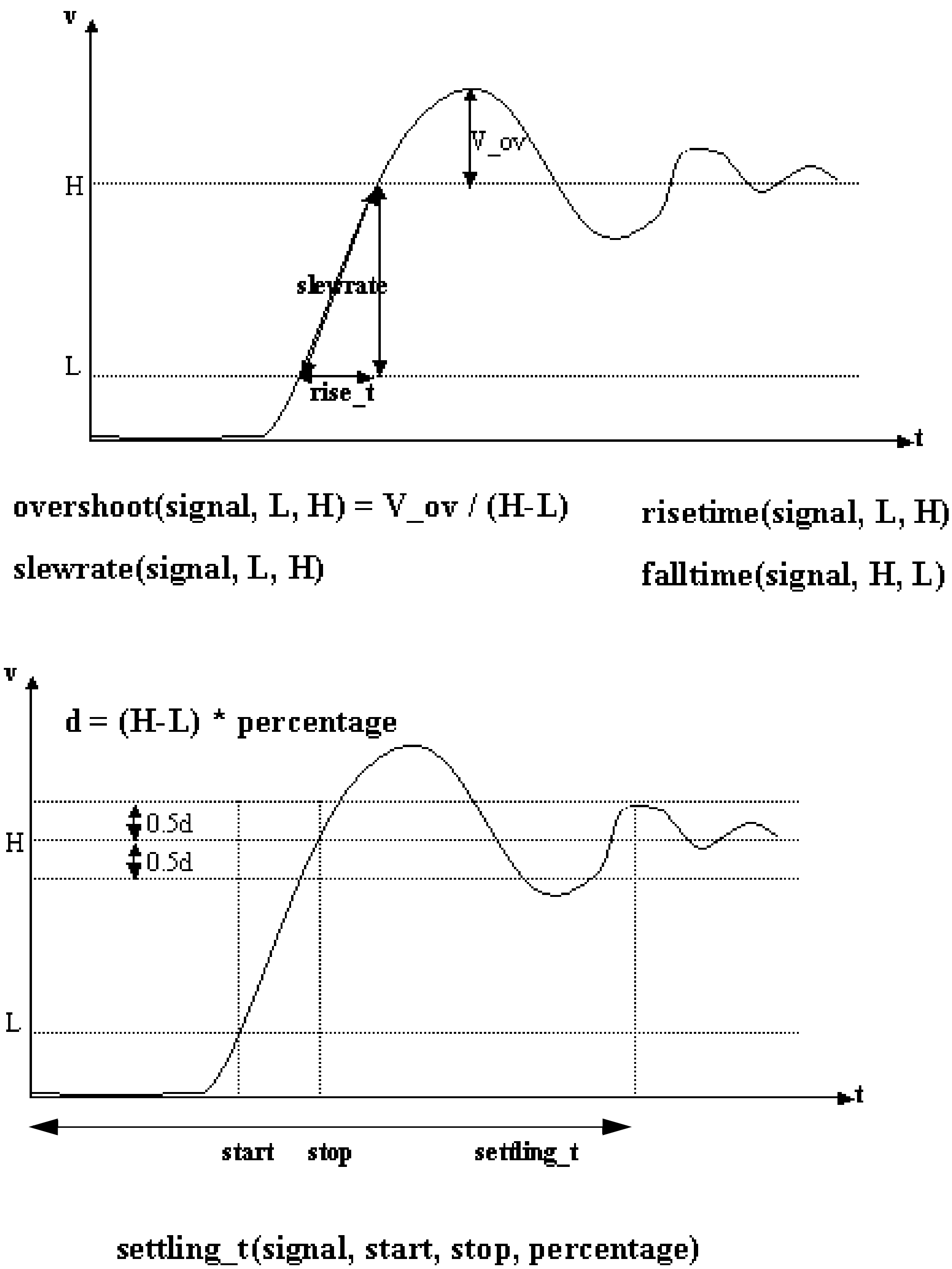
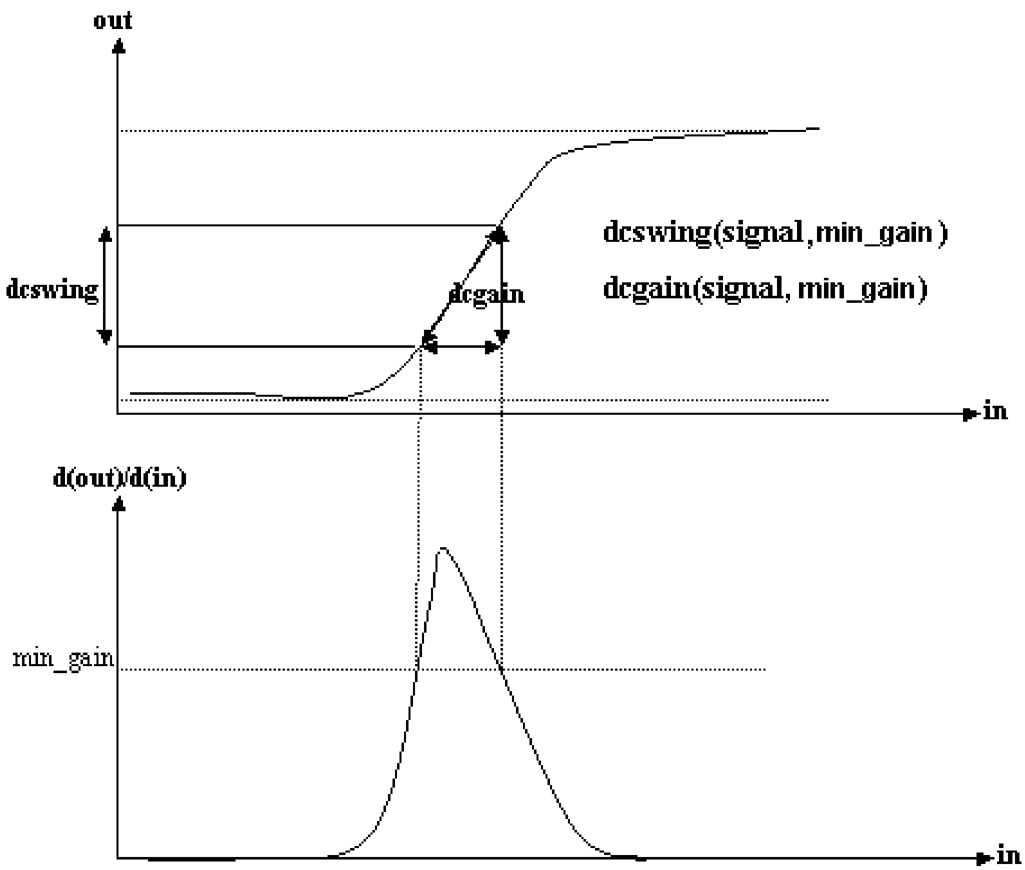


Figure 11 Time-domain functions


out
dcswing(signal,min_gain)
dcgain(signal, min_gain)
dcswing
dcgain
in
d(out)/d(in)
min_gain
in

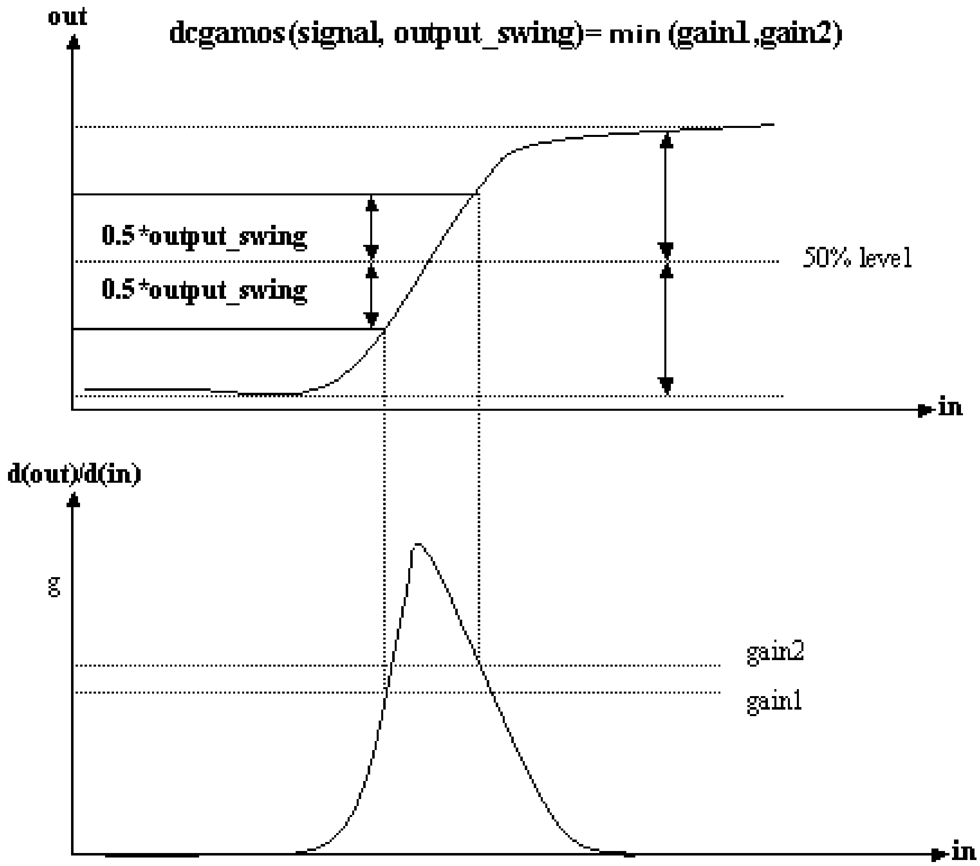


Figure 12 DC functions

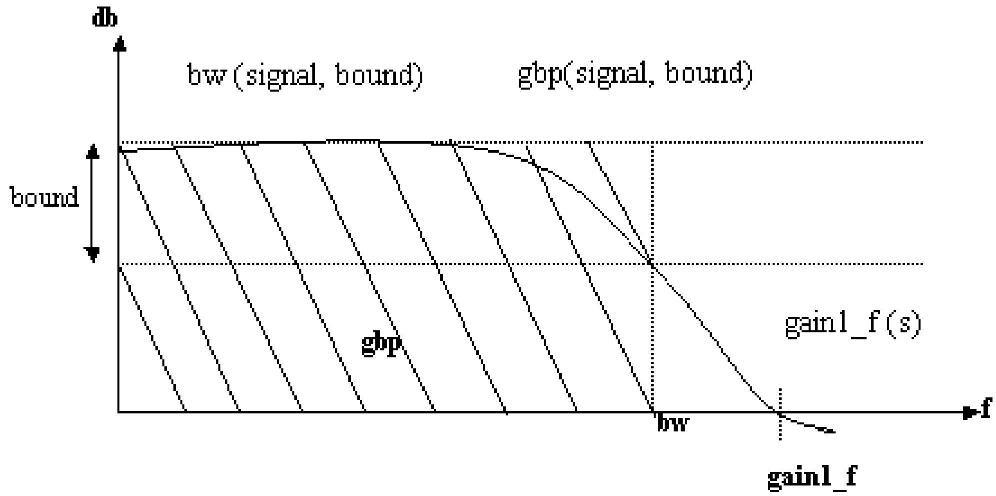


Figure 13 AC functions

Supported Waveform Functions

The following table summarizes the waveform functions in the equation builder. Waveform functions must operate on waveforms (or derived waveforms). They always return new waveforms.

Function	Description
a2d(s,low,high,<bits>,<timestep>,<xstart>,<xstop>)	Converts the s signal from analog to digital. The bits, timestep, xstart, and xstart options are optional.
average(s)	Returns the running average waveform of signal s
clip(s,<xi>,<x2>,<y1>,<y2>,<percent>)	Clips waveform of signal s from x1 to x2. The clip() function adds data points on the clipped boundaries. When <percent> is set to `1`, the y threshold value is considered a percentage value.
crossings(s1,s2)	Returns crossing points between signal s1 and s2. You can also specify the s2 argument as a Y threshold.

Function	Description
d2a(s,low,high, <glitchspan>, <timestep>,<xstart>, <xstop>)	Converts the s signal from digital to analog. The glitchspan, timestep, xstart, and xstart options are optional.
delayvst(s1,s2,y1,y2, <percent>,tidx1,tidx2, <trig>,<trigref>, <x1start>,<x1stop>, <x2start>,<x2stop>)	Calculates the time difference versus time between two signals at the specified switching edges in the <xstart> and <xstop> range. ■ s1: Trigger signal. ■ s2: Target signal. ■ y1: Threshold level for trigger signal. ■ y2: Threshold level for target signal. ■ percent: Percentage threshold value (0 or 1) ■ tidx1:Trace index for a signal. ■ tidx2:Trace index for a reference signal. ■ <trig>: The trigger type. Specify rise, fall, or either. ■ <trigref>: The reference trigger type for a signal (rise|fall|either|same|opposite). ■ <xstart1>: The starting x value for a signal. ■ <xstop1>: The ending x value for a signal. ■ <xstart2>: The starting x value for a reference signal. ■ <xstop2>: The ending x value for a reference signal.
derivative(s)	Returns the derivative waveform of signal s. For end points, only two points are used. For non-end points, a three-point scheme is used. The current point is the middle point.
duty_vs_t(s,y,rise, <xstart>,<xstop>, <percent>)	Calculates the duty cycle versus time. The duty cycle is plotted on the y-axis, and time is plotted on the x-axis. The following arguments are available: ■ s: The input data signal. ■ y: The threshold value. ■ rise: Uses the rising edge or not (0=fall and 1=rise). ■ <xstart>: The starting time. ■ <xstop>: The stopping time. ■ <percent>: When set to `1`, the y threshold value is considered a percentage value.

Function	Description
fallvst(s,hv,lv,<percent>,<xstart>,<xstop>)	Calculates the time difference versus time of the s signal from the high threshold crossing to the low threshold crossing in the <xstart> and <xstop> range. ▪ s: Input signal. ▪ hv: The high threshold level for the input signal. ▪ lv: The low threshold level for the input signal. ▪ <percent>: Percentage threshold value (0 or 1) ▪ <xstart>: The starting x value. ▪ <xstop>: The ending x value.
fft(s,x1,x2,n,win)	Returns the FFT result of signal s. Signal s is evenly PWC sampled from x1 to x2 for n points. The optional win argument specifies the windowing function. Valid window type strings (case insensitive) are: rectangle, Bartlett, parzen, welch, hanning, hamming, Blackman, and blackharris.
func(Sy,Sx)	Returns the waveform of Sy versus Sx for the specified Sy an Sx signals.
fvst(s,y,xstart,xend,<percent>)	Calculates frequency versus time. If both xstart and xend are not specified, the full x range is used. ▪ s: The input signal. ▪ y: The y threshold level. ▪ xstart: Starting x value. ▪ xend: Ending x value. ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value.
gdelay(s)	AC domain function, returns the groupdelay waveform of signal s. groupdelay is defined as d(phase)/d(w) where w=2*pi*freq.

Function	Description
hold_times(s1,s2, clock_period, y_threshold, sigedge,refedge, <xstart>,<xstop>, <percent>)	Returns a waveform that contains all hold time data points. X values represent when the hold times occur. The following arguments are available: ■ s1: Signal data ■ s2: Clock signal ■ clock_period: Clock period time ■ y_threshold: Y-threshold value ■ sigedge: Data signal edge, valid sigedge string are: "rise", "fall", and "both" ■ refedge: Clock signal edge, valid refedge string are: "rise", "fall", and "both" ■ <xstart>: (Optional) Specifies the starting time of the measure ■ <xstop>: (Optional) Specifies the ending time of the measure ■ <percent>: When set to `1`, the y threshold value is considered a percentage value. **Note:** If the input value for the clock signal period is 0 or smaller than 0, the clock period is calculated using the clock signal.

Function	Description
hold_times2(s1,s2, clock_period, data_yth,clk_yth, sigedge,refedge, <xstart>,<xstop>, <percent>)	Returns a waveform that contains all hold time data points. X values represent when the hold times occur. The following arguments are available: ▪ s1: Signal data ▪ s2: Clock signal ▪ clock_period: Clock period time ▪ data_yth: Data signal Y-threshold value ▪ clk_yth: Clock signal Y-threshold value ▪ sigedge: Data signal edge, valid sigedge string are: "rise", "fall", and "both" ▪ refedge: Clock signal edge, valid refedge string are: "rise", "fall", and "both" ▪ <xstart>: (Optional) Specifies the starting time of the measure ▪ <xstop>: (Optional) Specifies the ending time of the measure ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value. **Note:** If the input value for the clock signal period is 0 or smaller than 0, the clock period is calculated using the clock signal.
ift(s,fund,h)	Returns the IFT (inversed Fourier transform) time-domain waveform of the input frequency-domain spectrum s. Argument fund is the fundamental frequency, h is the maximum harmonic index to be included in the IFT computation. (h=0 for DC)
integral(s)	Returns the running integral waveform of signal s.
join(s1,s2)	Returns a joined waveform of s1 and s2. Depending on the starting x-axis value of the two signals, join() always appends the waveform that has greater starting x-axis value to the one with a smaller x-axis starting value. If s1 and s2 overlap, the overlapped portion of the later waveform is removed before joining. Reversed sweep data (x-start > x-stop) from s1 and s2 is reordered into forward sweep in the joined waveform.

Function	Description
jvst_freq(s,type,freq, yrise,yfall,refedge, <edge>,<nperiod>, <forward>,<range>, <sign>,<percent>)	Calculates the jitter vs time given the frequency of an ideal clock. ■ s: Target signal ■ type: Jitter type. Values are “trigger”, “width”, ”period”, ”cycle”, ”nperiod”, or “ncycle”. ■ freq: Frequency for an ideal reference clock ■ yrise: Target signal threshold level at the rising edge ■ yfall: Target signal threshold level at the falling edge ■ refedge: Reference signal edge type. Values are “same”, “rise”, “fall”, or “cross” ■ <edge>: Target signal edge type, values are “rise”, “fall”, or “both” ■ <nperiod>: (Optional) Number of periods. Only one “nperiod” and “ncycle” are specified for the jitter type. The default is “1” ■ <forward>: (Optional) Always search forward for the target edge. Values are “1” or “0”, and the default is “0” ■ <range>: (Optional) Max target edge search range ■ <sign>: (Optional) Forward or backward positive value sign. Values are “0” or “1”, and the default is “0” for the “forward” positive value sign. ■ <percent>: When set to `1`, the y threshold value is considered a percentage value.

Function	Description
jvst_ideal(s,type,delay,period,width,yrise,yfall,refedge,<edge>,<nperiod>,<forward>,<range>,<sign>,<percent>)	Calculates the jitter vs time with an ideal clock. ▪ s: Target signal ▪ type: Jitter type. Values are “trigger”, “width”, ”period”, ”cycle”, ”nperiod”, or “ncycle” ▪ delay: Delay value for an ideal reference clock with a period value ▪ period: Period value for an ideal reference clock ▪ width: Width value for an ideal reference clock ▪ yrise: Target signal threshold level at the rising edge ▪ yfall: Target signal threshold level at the falling edge ▪ refedge: Reference signal at an edge type. Values are “same”, “rise”, “fall”, or “cross” ▪ edge: Target signal edge type. Values are “rise”, “fall” or “both” ▪ nperiod: (Optional) Number of periods. only one “nperiod” and “ncycle” is specified for a jitter type, and the default is “1” ▪ forward: (Optional) Always searches forward for a target edge. Values are “1” or “0”, and the default is “0” ▪ range: (Optional) Max target edge search range ▪ sign: (Optional) Forward or backward positive value signs. Values are “0” or “1”, and the default is “0” for the “forward” positive value sign ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value.

Function	Description
jvst_llse(s,type,yrise, yfall,refedge,<edge>, <nperiod>,<forward>, <range>,<sign>, <percent>)	Calculates the jitter vs time with an estimated frequency of the ideal clock using LLSE. ▪ s: Target signal ▪ type: Jitter type. Values are “trigger”, “width”, ”period”, ”cycle”, ”nperiod”, or “ncycle” ▪ yrise: Target signal threshold level at the rising edge ▪ yfall: Target signal threshold level at the falling edge ▪ refedge: Reference signal at an edge type. Values are “same”, “rise”, “fall”, or “cross” ▪ edge: Target signal edge type. Values are “rise”, “fall” or “both” ▪ nperiod: (Optional) Number of periods. only one “nperiod” and “ncycle” is specified for a jitter type, and the default is “1” ▪ forward: (Optional) Always searches forward for a target edge. Values are “1” or “0”, and the default is “0” ▪ range: (Optional) Max target edge search range ▪ sign: (Optional) Forward or backward positive value signs. Values are “0” or “1”, and the default is “0” for the “forward” positive value sign ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value.

Function	Description
jvst_refsig(s1,s2,type, yrise,yfall,<refedge>, <edge>,y<rise_ref>, <yfall_ref>,<nperiod>, <forward>,<range>, <sign>,<percent>)	Calculates the jitter vs time using a reference signal as the clock. ▪ s1: Target signal ▪ type: Jitter type. Values are “trigger”, “width”, ”period”, ”cycle”, ”nperiod”, or “ncycle” ▪ s2: Reference signal, as well as the reference clock signal ▪ type: The jiiter type. ▪ yrise: Target signal threshold level at the rising edge ▪ yfall: Target signal threshold level at the falling edge ▪ refedge: Reference signal at an edge type. Values are “same”, “rise”, “fall”, or “cross” ▪ edge: Target signal edge type. Values are “rise”, “fall” or “both” ▪ yrise_ref: (Optional) Reference signal threshold level at the rising edge. Only required if an analog reference signal is specified ▪ yfall_ref: (Optional) Reference signal threshold level at the full edge. Only required if an analog reference signal is specified, ▪ nperiod: (Optional) Number of periods. only one “nperiod” and “ncycle” is specified for a jitter type, and the default is “1” ▪ forward: (Optional) Always searches forward for a target edge. Values are “1” or “0”, and the default is “0” ▪ range: (Optional) Max target edge search range ▪ sign: (Optional) Forward or backward positive value signs. Values are “0” or “1”, and the default is “0” for the “forward” positive value sign ▪ <percent>: When set to `1`, the y threshold value is considered a percentage value.
median(s,w)	Returns the smoothed waveform of signal s using the median-value method. For every data point in the original signal s, the median-value method sorts the values of the original data point and w points before and after the original point. The median value of the sorted result is then used to replace the original data value.

Function	Description
parametric(m,parm)	Returns the parametric plots of measure m versus sweeping parameter parm. If the optional parm argument is not defined or cannot be found in the associated sweeping parameters of m, sequential sweep trace index is used as the plot x-axis.
pareto_r(s,pfile, <normalized>,sort, <smin>,<r2min>, <rmin>,<histbar>, <scatter>,<twin>, <tfile>)	Calculates the Pareto regression result for the s signal and the pfile param file. ■ s: Input signal. ■ pfile: Param File name. ■ <normalized>: Normalization factor, which can be `Param/Meas`, `abs(Param/Meas)`, `Param`, `1/Meas`, or `Un-normalized`. ■ sort: Sorted results, which can be `S`, `R^2`, or `R`. ■ <smin>: Sensitivity filter minimum value. ■ <r2min>: Regression square filter minimum value. ■ <rmin:> Regression filter minimum value. ■ <histbar>: Result output by histogram plot, which can be `0` or `1`. ■ <scatter>: Result output by scatter plot, which can be `0` or `1`. ■ <twin>: Result output by text window, which can be `0` or `1`. ■ <tfile>: Result output by text file, which can be the text file name or `missing`.

Function	Description
pareto_r2(s,pfile,<normalized>,sort,<smin>,<r2min>,<rmin>,<histbar>,<scatter>,<twin>,<tfile>)	Calculates the Pareto regression squared result for the s signal and the pfile param file. ▪ s: Input signal. ▪ pfile: Param File name. ▪ <normalized>: Normalization factor, which can be `Param/Meas`, `abs(Param/Meas)`, `Param`, `1/Meas`, or `Un-normalized`. ▪ sort: Sorted results, which can be `S`, `R^2`, or `R`. ▪ <smin>: Sensitivity filter minimum value. ▪ <r2min>: Regression square filter minimum value. ▪ <rmin:> Regression filter minimum value. ▪ <histbar>: Result output by histogram plot, which can be `0` or `1`. ▪ <scatter>: Result output by scatter plot, which can be `0` or `1`. ▪ <twin>: Result output by text window, which can be `0` or `1`. ▪ <tfile>: Result output by text file, which can be the text file name or `missing`.
pareto_s(s,pfile,normalized,sort,<smin>,<r2min>,<rmin>,<histbar>,<scatter>,<twin>,<tfile>)	Calculates the Pareto sensitivity result for the s signal and the pfile param file. ▪ s: Input signal. ▪ pfile: Param File name. ▪ normalized: Normalization factor, which can be `Param/Meas`, `abs(Param/Meas)`, `Param`, `1/Meas`, or `Un-normalized`. ▪ sort: Sorted results, which can be `S`, `R^2`, or `R`. ▪ <smin>: Sensitivity filter minimum value. ▪ <r2min>: Regression square filter minimum value. ▪ <rmin:> Regression filter minimum value. ▪ <histbar>: Result output by histogram plot, which can be `0` or `1`. ▪ <scatter>: Result output by scatter plot, which can be `0` or `1`. ▪ <twin>: Result output by text window, which can be `0` or `1`. ▪ <tfile>: Result output by text file, which can be the text file name or `missing`.

Function	Description
phasenoise(s,fFund,<x1>,<x2>,<fSample>,<fMinRatio>,<nCycle>,<win>)	Calculates the phase noise for a signal. The following arguments are available: ▪ s: Signal data ▪ fFund: Fundamental frequency ▪ <x1>: tStart. The default is the waveform start time. ▪ <x2>: tStop. The default is the waveform end time. ▪ <fSample>: Sampling frequency. The default is `128*fFund`. ▪ <fMinRatio>: Minimum offset frequency. The default is 10. ▪ <nCycle>: Number of delay clock cycles. The default is 0. ▪ <win>: Window type. The default is Rectangle.
repeat(s,p,n)	Creates a new waveform by repeating the period (p) of the given signal number of cycle (n) times.
risevst(s,lv,hv,<percent>,<xstart>,<xstop>	Calculates the time difference versus time of the s signal from the low threshold crossing to the high threshold crossing in the <xstart> and <xstop> range. ▪ s: Input signal. ▪ lv: The low threshold level for the input signal. ▪ hv: The high threshold level for the input signal. ▪ <percent>: Percentage threshold value (0 or 1) ▪ <xstart>: The starting x value. ▪ <xstop>: The ending x value.
sample(s,x,p,n)	Returns a new waveform that is a sampled waveform of signal s from x at step p for n times

Function	**Description**
setup_times(s1,s2, clock_period, y_threshold, sigedge,refedge, <xstart>,<xstop>, <percent>)	Returns a waveform that contains all setup time data points. X values represent when the setup times occur. The following arguments are available: ■ s1: Signal data ■ s2: Clock signal ■ clock_period: Clock period time ■ y_threshold: Y-threshold value ■ sigedge: Data signal edge, valid sigedge string are: "rise", "fall", and "both" ■ refedge: Clock signal edge, valid refedge string are: "rise", "fall", and "both" ■ <xstart>: (Optional) Specifies the starting time of the measure ■ <xstop>: (Optional) Specifies the ending time of the measure ■ <percent>: When set to `1`, the y threshold value is considered a percentage value. **Note:** If the input value for the clock signal period is 0 or smaller than 0, the clock period is calculated using the clock signal.

Function	Description
setup_times2(s1,s2, clock_period, data_yth,clk_yth sigedge,refedge, <xstart>,<xstop>, <percent>)	Returns a waveform that contains all setup time data points. X values represent when the setup times occur. The following arguments are available: ■ s1: Signal data ■ s2: Clock signal ■ clock_period: Clock period time ■ data_yth: Data signal Y-threshold value ■ clk_yth: Clock signal Y-threshold value ■ y_threshold: Y-threshold value ■ sigedge: Data signal edge, valid sigedge string are: "rise", "fall", and "both" ■ refedge: Clock signal edge, valid refedge string are: "rise", "fall", and "both" ■ <xstart>: (Optional) Specifies the starting time of the measure ■ <xstop>: (Optional) Specifies the ending time of the measure ■ <percent>: When set to `1`, the y threshold value is considered a percentage value. **Note:** If the input value for the clock signal period is 0 or smaller than 0, the clock period is calculated using the clock signal.
slewratevst(s,initv, finalv,<percent>, <xstart>,<xstop>, <trig>	Calculates the slew rate versus time of the s signal from the initv threshold crossing to the finalv threshold crossing in the <xstart> and <xstop> range. ■ s: Input signal. ■ initv: The initial threshold level for the input signal. ■ finalv: The final threshold level for target signal. ■ percent: Percentage threshold value (0 or 1) ■ <xstart>: The starting x value. ■ <xstop>: The ending x value. ■ <trig>: The trigger type. Specify rise, fall, or either.
smooth(s,w)	Returns the smoothed waveform of the input signal s. For each point of the input signal s at x, the average value from (x-w) to (x+w) is calculated to generate the smoothed output
sweepavg(s)	Returns the average waveform of a multi-trace sweep waveform.

Function	Description
sweepmax(s)	Returns the MAX bound waveform of a multi-trace sweep waveform.
sweepmin(s)	Returns the MIN bound waveform of a multi-trace sweep waveform.
trace(s,parm_cond)	Returns a trace from a multi-trace signal s based on the parameter condition parm_cond (example. trace("sweep_sig","A=1:B=2")
x_at_y(s,ylevel)	Returns the crossing points of signal s on a specified y threshold.
xscale(s,xf)	Returns a new waveform identical to s with its x-scale multiplied by xf.
xshift(s,xv)	Shifts signal s to the right by xv
xval(s)	Returns the x-axis value of signal s.
yvsi(s)	Creates a y value versus data point index signal.

Supported RF Functions

The following table summarizes the RF functions in the Equation Builder. Complex mathematic functions can also operate on both scalars and waveforms.

Function	Description
rf("SP",p1,p2)	S-parameter, p1 and p2 are 1-based port index
rf("ZP",p1,p2,<z0>)	Z-parameter, p1 and p2 are 1-based port index. z0 is the port characteristic impedance
rf("YP",p1,p2,<z0>)	Y-parameter, p1 and p2 are 1-based port index. z0 is the port characteristic impedance
rf("HP",p1,p2,<z0>)	2-port H-parameter, p1/p2 1-based port index. z0: port characteristic impedance

Function	Description
rf("VSWR",p1)	Voltage SWR at p1
rf("ZM",p1)	Z-parameter at port p1 with all other ports matched
rf("ga",zl)	2-port available power gain, zl is load impedance
rf("gt",zs)	2-port transducer power gain, zs is source impedance
rf("gp",zs,zl)	2-port power gain, zs/zl are source/load impedance
rf("KF")	2-port stability factor
rf("B1F")	2-port stability factor B1
rf("gumx")	2-port unilateral max. available power gain
rf("gmax")	2-port maximum available power gain
rf("gmsg")	2-port max. stable power gain
rf("noisesc")	Noise circle
rf("stabc_out")	Output stability circle
rf("stabc_in")	Input stability circle
rf("opgc")	Output gain circle
rf("apgc")	Available power gain circle
rf("vswrc_out")	Output VSWR circle
rf("vswrc_in")	Input VSWR circle

The RF functions support S-parameter result from Spectre, ELDO-RF, and ADS.

Supported Logic Operations

The equation builder supports the following logic operations.

Operator	Logic Function	Notes
&	AND	Bitwise or unary
|	OR	Bitwise or unary
^	XOR	Bitwise or unary
~	NOT	Bitwise only
>>	SHIFT right	
<<	SHIFT left	
~&	NAND	Unary or binary
~|	NOR	Unary or binary
~^	XNOR	Unary or binary

Note: You can add the 'b 'o 'd or 'h prefix to a logic operator to specify binary, octal, decimal, or hexadecimal, respectively. For example:

```
6'b010=6'b000010
9'o077=9'b000111111
4'd010=4'b1010
12'h0F0=12'b000011110000
```

9

Using Editors and Toolboxes

This chapter contains information on how to use the editors and toolboxes in the Custom WaveView tool.

This chapter contains information on the following editors and toolboxes:

- Using the PWL Editor
- Using the ADC Toolbox
- Using the Coherent Sample Only (CSO) ADC Toolbox

Using the PWL Editor

The PWL Source Editor is used to construct SPICE PWL input stimuli for circuit simulations. The editor supports interactive editing functions via direct point dragging. Multiple sources can be edited or overlapped simultaneously. You can store the edit result in the text PWL storage files (.pwl) or exported to SPICE netlist files.

The command buttons are grouped into three groups:

- File Operations
- Source Operations
- Point Operations

File Operations

Open a pre-stored PWL editing task with the **Open** button. Save an editing task with the **Save** button. The save function automatically appends file extension `.pwl` to the user-specified filename if the `.pwl` extension is not used. Click the

Check button to check errors in the PWL sources such as overlapping time points.

Click **Close** to close the PWL editor window. All PWL sources are not removed when the PWL editor window is closed.

Source Operations

Click the **Add New Source** button to add a single PWL source or a bus pattern that contains multiple PWL sources. For bus pattern, the usage of high or low levels, cycle time, rise time, and fall time are indicated in Figure 14. For example:

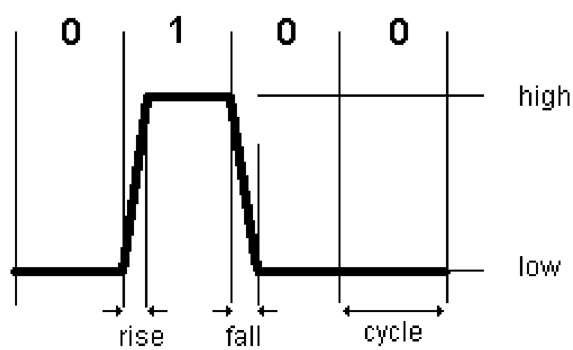


Figure 14 Timing parameters of a bus pattern

The source list window keeps a list of all PWL sources. The selected PWL source is the active source highlighted with thick line. Click **Modify a Source** to modify the name and parameters of the active source. Click **Display Source** to toggle the active source on and off.

Delete the active source with the **Delete a Source** button, or clear all sources with the **Delete All** button.

Point Operations

A newly added source has five points. Click on a point to select the point as the active point. You can insert a new point after the current active point using the **Insert a Point** button, or insert a pattern using the **Insert Pattern** button. A point can also be directly added with the Time and Value text fields and the

Add button. The newly added point becomes the active point highlighted with a solid box. With multiple points in a source, click the **Next** or the **Prev** button to jump the active point of a source.

To shift the entire PWL pattern forward or backward after the active point, click the **Shift Pattern** button, and drag the active point horizontally to shift the pattern. For example:

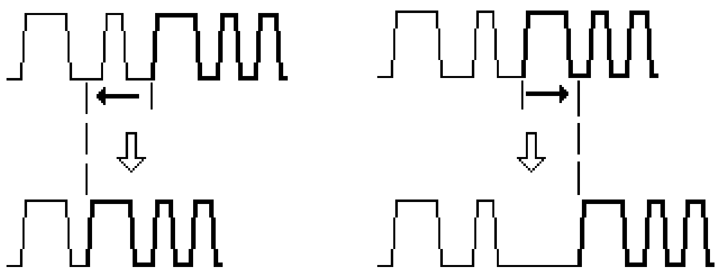


Figure 15 A shifting PWL pattern

The Time and Value fields are updated when a new active point is selected. To modify the values of the active point, enter new values and click the **Modify** button to make the change. You can also change the values of any point by dragging it directly. Dragging a point also selects it as the active point.

Click the **Delete a Point** button to delete the active point.

To assist point placement, snap functions can be enabled along the horizontal and vertical direction. Enter the preferred snap distance and check the corresponding option button to enable the snap function.

To zoom in an area, click and hold the mouse and move the mouse pointer to select the zoom area. Click the **Zoomout** button to zoom out or the **Full** button to reset to full viewing range.

Using the ADC Toolbox

The ADC Toolbox is a built-in tool for detailed performance analysis of A-to-D converter designs. Based on simulation output waveform data or measured data, the ADC Toolbox reports the following performance merits:

- DC static characteristics—Histogram, INL/DNL min/max values, and their standard deviations.
- AC dynamic characteristics—Noise parameters SNR, THD, SNDR, SFDR and ENOB and the frequency-domain power spectrum.

To open the ADC Toolbox, choose **Tools > ADC Toolbox** from the main menu. Simplified SNR/THD/SNDR/ENOB calculation functions, for sine input signals, are also provided in the FFT dialog box and the equation builder.

Generic Versus Coherent ADC Toolbox

Before Custom WaveView version 2004.4, the ADC Toolbox only supports the Coherent sampling method for sine input waveform and the input waveform have their signal amplitude in code values (signal values range from 0 to 2N-1, where N is the number of bits of the design).

From release 2004.4, a new generic ADC Toolbox is introduced. The new ADC toolbox accepts input signal in arbitrary amplitude, and a more generic Window sampling method is added to eliminate the need to sample the input sine waveform over integer cycles, which is a requirement of the Coherent sampling method.

To use the original Coherent-only ADC Toolbox, set the `SW_SX_ORG_ADC` environment variable to 1 in your shell:

```
setenv SW_SX_ORG_ADC 1
```

Without setting the `SW_SX_ORG_ADC` environment variable, the new generic ADC Toolbox is invoked as the default mode.

The new generic ADC Toolbox also allows the AC and the DC analysis to be carried out separately.

Overview of the ADC ToolBox

The ADC Toolbox window is divided into three tabs: the Input Parameters tab, DC Analysis Result tab, and AC Analysis Result tab.

The Input Parameter tab acquires input parameters from users for an ADC analysis task. The input parameters are divided into four categories: General input signal properties and data sampling parameters, DC analysis parameters, AC analysis parameters, and AC Coherent sampling parameters.

The DC Analysis Result tab displays sampled waveform and DC result including histogram, DNL and INL plots/distribution.

The AC Analysis Result tab displays the FFT spectrum of the sine input waveform and AC figure of merits including SNR/THD/SNDR/SFDR/ENOB.

Input Signal of ADC Toolbox

The ADC Toolbox accepts the following input signals:

- Sine: Both AC FFT analysis and INL/DNL analysis are available for the sine input type.
- Ramp: Only the INL/DNL analysis is available for the ramp input type.

The input signal of the ADC Toolbox can be an analog signal with arbitrary amplitude ranges, or a digital bus signal representing the digital output code of an ADC design. For digital bus input signal, the bus signal must be displayed in a waveview panel. Drag an analog signal from the Output View, or a bus signal from a panel to the Signal field in the ADC Toolbox.

The input signal is sampled based on one group of the following three sampling parameters: Start Time, Stop Time, and number of points, or Start Time, Sampling Frequency (Fs), and number of points. When the stop time is used (in the first group), the last sample point is at:

```
stop_time - (stop_time - start_time)/number_of_points
```

For sine input type, the input frequency (Fin) also needs to be specified.

You can input analog bus signals directly by entering bus named delimied by semi-colons (;). For example:

```
ADCOUT_3;ADCOUT_2;ADCOUT_1;ADCOUT_0
```

When you enter analog bus signals directly, the following requirements must be satisfied:

- Signals must be analog.
- The number of signals must be the same as the number of bits.
- The order bus signals must be MSB ; .... ; LSB.
- All signals in the bus must come from the same file.
- No more than 32 signals can be entered at a time.

Once the input signals are accepted, the median signal value among all signals is used as the A2D conversion threshold. For example:

```
0.5 * (ymax_of_all_bit + ymin_of_all_bits)
```

The analog bus signals are then converted to a logic bus whose code values are then used as the input to the ADC Toolbox. Analog bus signals are also supported during .INI file save/load operations.

For measured data that was sampled at known Fs, the data (x,y) value pair can use integer index as the x values. If this is the case, select the **Measured data** option and the ADC Toolbox processes the input as if the data is sampled at Fs. When the **Measured Data** option is selected, the Start T/P value is the starting index (0-based) of the measured data.

If you have a measurement data with only single column, (that is, the data is measured at fixed known sampling rate so the data file does not contain x-axis sample time information), edit the file header to add the following lines:

```
#format  table  data_only
name_of_data
measure data 1
measure data 2
…..
```

With the keywords in the header line, the Custom WaveView tool can load such a text data file as a regular waveform file and automatically assign integer index to the x-axis. You can then apply the **Measured Data** option and use the measurement data in the ADC Toolbox.

DC DNL/INL Analysis

To calculate DNL/INL, an analog input signal is first scaled into full code range based on the **Code Range** parameters in the DC Input Parameters section. If the **Number of Bits** value is used, the ADC Toolbox assumes that the input signal covers full code range from 0 to 2N-1. If the **User-defined Code Range** option is selected, the ADC Toolbox assumes that the input signal swings from the **Min** code value to the **Max** code value. Digital input bus signal is not subject to this amplitude scaling, instead the bus direct code values are used.

If the analog input signal value is already converted to the A/D converter output bus values, select the **Signal Value=Output Code** option.

Because the ADC Toolbox always automatically scales the analog input signal, PWL sampling is always used in sampling analog input signal.

After entering the necessary DC analysis parameters, click **Evaluate DC** to carry out a DC analysis.

Sine Input

For sine input signal, based on the sampled data and the Fin parameters, the ADC Toolbox needs to find the ideal sine wave in order to calculate INL and DNL. The ADC Toolbox uses two methods to calculate the ideal sine waveform:

- FFT fitting: If the input sine waveform data is not clipped. The amplitude and phase of the strongest FFT spectrum component of the input sine signal is used as the ideal sine waveform.
- LMS (least-mean-square) fitting: If the input sine waveform data is clipped. This is often seen with measure data. The ADC Toolbox uses the sample data within the Clip-Min and Clip-Max code values, and LMS fitting method to find the original ideal sine waveform. The offset and the amplitude of the fitted ideal sine waveform is displayed in the Sampled Data plot.

With the LMS fitting, since the clipped sampled data might be heavily distorted, FFT AC analysis is disabled.

Ramp Input

For the ramp input type, ADC Toolbox always assumes that the ideal ramp signal evenly covers the full code range. Based on this assumption, multiple complete cycles of ramp (up or/and down) are accepted.

Number of Sample Points and INL/DNL

In order to produce meaningful INL/DNL result, the number of sample points must be large enough so that there is at least one sample point in each code value. For example, for a 10-bit design, ramp input, at least 1024 points (210) sample points are needed. Sine input type needs even more points because the sample points do not distribute evenly over different code values.

Since the ADC Toolbox scales the original analog input signal and uses PWL method to sample the scaled input signal, it is the user's responsibility to make sure that the measurement/simulation generates enough true data points at each code value.

Sampling Frequency and INL/DNL

For sine input type, if multiple periods of the input sine waveform are sampled, the sampling frequency Fs and input frequency Fin must be carefully selected to avoid sampling the same code values in each period. For example, if a 5 MHz input sine waveform (6-bit) is sampled at 40 MHz over 128 cycles, the 8 sampled code values from each cycle are always the same. As a result, only 8 out of the 64 code values are available in the sampled data. If Fin is at 4.99 MHz, with the sampling frequency, the sampled data has different code values from cycle to cycle, and the INL/DNL result would be more meaningful.

AC Analysis: Coherent Sampling Versus Window Sampling

With the sine input type, the ADC Toolbox supports FFT AC analysis using two different types of sampling method:

- Coherent Sampling
- Window Sampling

Coherent Sampling

With coherent sampling, exact integer multiple cycles (FinNUM) of the input sine wave are sampled. The sampled data are then sorted into a single cycle of sine wave before performing the dynamic performance analysis using FFT.

Advantages of the coherent sampling method include:

- It allows an input signal at higher frequency to be sampled using a lower sampling rate. In the real world, physical measurement equipment might sometimes have limited sampling performance to meet the Nyquist criterion required by the bandwidth of the input signal.
- Harmonics and noise are readily separated in the power spectrum using coherent sampling.
- FFT windowing functions are not required because integer cycles of the input signal are sampled.

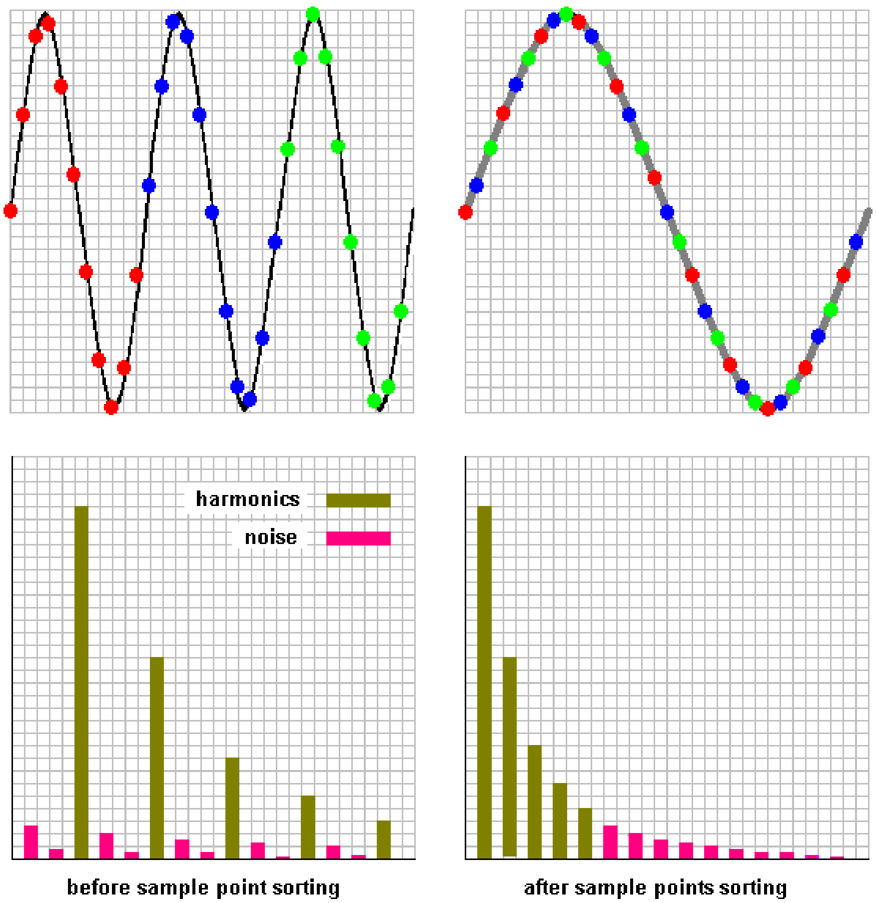


Figure 16 Coherent sine wave sampling

However, coherent sampling method does require the exact frequency (Fin) of the input sine wave to be known prior to running an ADC simulation. To use the coherent sampling method in the ADC Toolbox, you need to enter the preferred Precision, the estimated Fin and Fclk, and Sample Pts. Then, click **Evaluate AC** to find out the exact values for Fin.

Figure 16 is a coherent sampling example with FinNUM = 3, N=32. Coherent sampling requires that N x Fclk = FinNUM x Fin. In order to sort sampled points of multiple cycles into a single sine wave cycle, N and FinNUM should not have any common factor. Since N is always power of 2, an odd number for FinNUM is always a valid choice. In the ADC Toolbox, the FinNUM value can be automatically determined based on user-defined Fin and Fclk values.

As depicted in the above graph, the power spectrum of the sorted sine wave is also a sorted version of the original multi-cycle sine waveform's power spectrum.

The sorted single-cycle sine wave is sometime also referred to as the unfolded sine wave.

Refer to Using the Coherent Sample Only (CSO) ADC Toolbox for details about entering parameters for Coherent Sample.

Window Sampling

The window sampling method does not require integer number of cycles of the input sine signal to be sampled. However, to overcome the spectral leakage problem from not sampling over an integer cycles of the input waveform, a FFT windowing function is required. Figure 17 shows the FFT spectrum of a 4.99 MHz sine signal sampled over 127.75 cycles at 40 MHz.

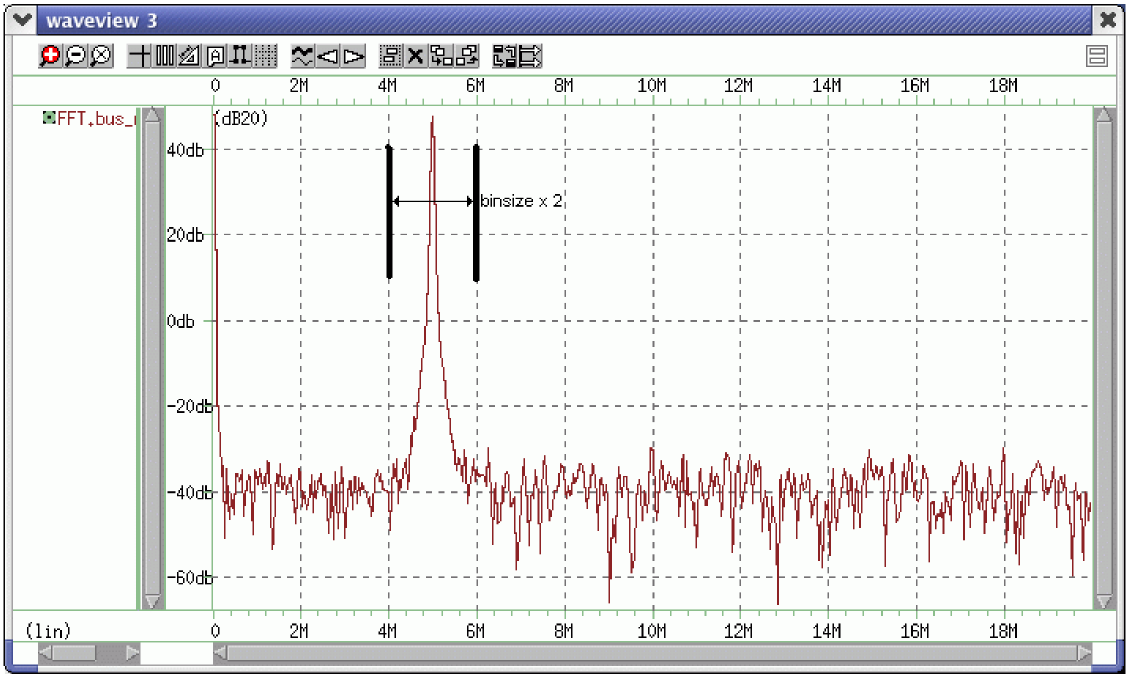


Figure 17   FFT spectrum with window functions

As depicted in the above spectrum, the power at the fundamental frequency (4.99 MHz) gets spread into neighboring spectral components due to spectral leakage. For this reason, a binsize parameter is required to include all neighboring components within +/- binsize spectral range as part of the original fundamental (or harmonic) components. The default value of BinSize is 10.

After entering the necessary AC analysis parameters, click **Evaluate AC** to start an AC analysis.

Exporting the DNL/INL/FFT Results as Waveform Data

The ADC Toolbox result, including the sampled waveform data, histogram, INL/DNL and the FFT power spectrum, can be exported to a waveview window for further viewing purpose. Click **Waveform** next to the plot window to display the data into a new waveview window.

Saving or Loading an Analysis Setup

The ADC Toolbox setup, including the input parameters, sampled data, and the analysis result can be saved into data files. To restore a previously saved setup, load from the .INI (Input parameter) file.

Dumping the Contents of the ADC Toolbox

To dump the contents of the ADC Toolbox to a graphic file (PNG, JPG, or BMP), click the **Dump** button on the ADC Toolbox window.

ADC Toolbox Display Controls

The following window displays are included in the ADC toolbox:

- Sampled Signal Window
- Histogram, DNL, and INL Windows
- Power Spectrum (Sine Input Only)

Sampled Signal Window

The Sampled Signal window displays the sampled data waveform based on the input parameters. The Y-axis is in code value, with min-code value at the bottom and max-code value at the top.

Ideal/Acquired/Error/All switch:

- Ideal: Displays the ideal signal used for INL/DNL calculation.
- Acquired: Displays the sampled waveform data.
- Error: Displays the error between ideal signal and the sampled data.
- All: Displays all of the above.

Sin/Cos/Original (sine input only):

- Sin: Adjusts the phase of the displayed waveform into Sin phase (starts from zero).
- Cos: Adjusts the phase of the displayed waveform into Cos phase (starts from the maximum value).
- Original: No phase adjust for sampled waveform display.

Exp/Integer:

- Exp: Displays the Y-axis value ranges using the engineering format.
- Integer: Displays the Y-axis value ranges using the integer code values.

Histogram, DNL, and INL Windows

These windows display the histogram, DNLb, and INL plots of the sampled signal. Aligned with the Y-axis of the Sampled signal plot, the Y-axes of all three plots correspond to the code value. The X-axis is the histogram / DNL / INL value in LSB.

Power Spectrum (Sine Input Only)

The power spectrum window displays the noise spectrum of the sampled signal.

Line/Bar/Both

- Line: Plots the power spectrum using line chart.
- Bar: Plots the power spectrum using bar chart.
- Both: Plots the power spectrum using both line and bar chart.

Index/Frequency

- Index: Displays the spectrum X-Axis value range using harmonic index.
- Frequency: Displays the spectrum X-axis value range using frequency.

AC Dynamic Characteristics

The AC Characteristics window displays the spectrum harmonic strength and the following values:

- SNR: The signal-to-noise ratio.
- THD: The total harmonic distortion.
- ENOB: The effective number of bits.
- SNDR: The signal-to-noise and distortion ratio.
- SFDR: The spurious free dynamic range. The harmonic index of the second strongest harmonic component is also displayed together with the SFDR value (for example, SFDR index_after_sorting (index_before_sorting)).

The strength of the power spectrum is in db relative to the strongest harmonic component (0 db for the strongest harmonic component). The AC characteristic window lists the value of each of the harmonic component. Use the **up** and **down** buttons to scroll up/down to view the spectrum values at different harmonic range.

Select the toggle check box next to a harmonic component to either exclude the harmonic for the THD calculation or include the harmonic for the SNR calculation.

Click **Evaluate AC** to update the AC noise analysis result (SNR/THD/ENOB/SNDR/SFDR) based on the new harmonic selection.

DC Static Characteristics

The DC Characteristics window displays the min/max values and the standard deviation of the INL/DNL distribution. The unit is 1 LSB.

Running Batch-mode ADC Analyses

ADC Toolbox analysis can be run in batch mode with the -adcin, -adcout, -adcwdf, and -adccsv command line option. The usage is:

```
wv -adcin adc.INI -adcout out_sw -adcwdf wdf_sw
```

The adc.INI is a previously saved .INI file, out_sw is the _ (underscore) delimited type switch that controls the output text files; wdf_sw is the _ (underscore) delimited type switch that controls the output WDF waveform files. The -adccsv option sets the output text file in the CSV (comma separated values) format.

One of the -adcout and -adcwdf options must be specified or the batch mode operation is aborted. Supported type strings are INL, DNL, FFT, TLA, HIS, IND, ALL, and DAT (-adcout only). FFT and TLA have no effect on ramp input signals. For example:

```
wv -adcin adc.INI -adcout  INL_DNL  -adcwdf  TLA_FFT -
adccsv
```

Using the Coherent Sample Only (CSO) ADC Toolbox

To start the ADC Toolbox in the CSO mode, set the `SW_SX_ORG_ADC` shell environment variable.

Input Signal Requirements

The input waveform for the CSO ADC Toolbox must have its signal strength in integer code values (for example, 0 to 63 for a 6-bit ADC). Run a simulation on an ADC design first and then convert the ADC output codes into an acceptable waveform.

Preparing the Input Waveform

As depicted in the flow chart below, the following steps demonstrate how to use the Custom WaveView tool to convert the simulated ADC output code (in analog waveforms) to an acceptable input signal for the CSO ADC Toolbox.

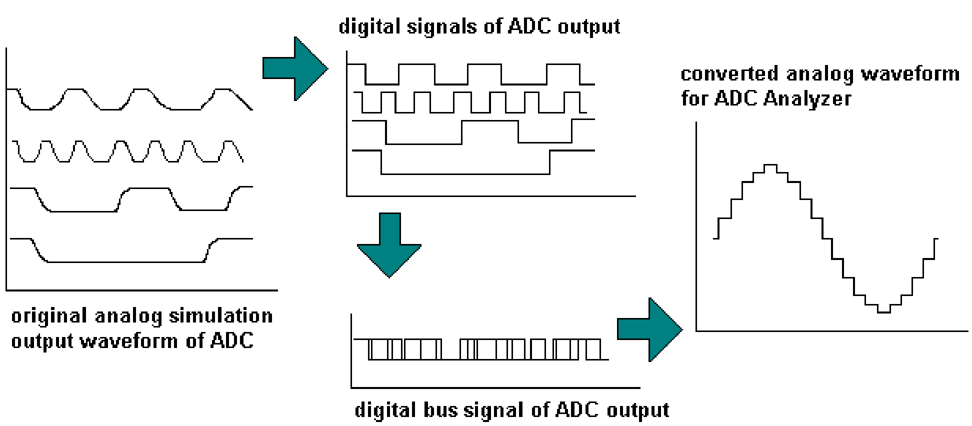


Figure 18 Preparing an input signal for the ADC toolbox

To convert the simulated ADC output code to an acceptable input signal for the CSO ADC Toolbox:

1. Identify the analog output waveforms that represent the ADC output code, for example, "ad0" to "ad5" for a 6-bit design.

2. Display these signals in a waveview window and then highlight them by highlighting the panel(s) these signals reside in.
3. Use the A/D converter tool in the Custom WaveView single-bit mode to convert each of the highlighted analog waveforms to a digital waveform. Name the converted digital waveforms "AD0" - "AD5", and ensure proper threshold value is used for this A to D conversion step.
4. Group the converted logic signal "AD0" - "AD5" into a digital bus signal. Make sure that the signals are ordered (sorted) correctly so "AD5" is the MSB and "AD0" is the LSB.
5. Rename the grouped bus signal to "AD". You can use this logic bus signal directly as the input to the ADC Toolbox, or you can continue to Step 6.

   To use the logic bus signal as the input, the bus signal must be displayed in a waveview window, and you must drag and drop the logic bus signal from the display panel.
6. Use the D/A convertor tool in the Custom WaveView tool to convert the grouped converted digital bus "AD", in multi-bit mode, into an analog waveform.
7. Ensure that the **Use logic bus value** option is selected so that the converted waveform has strength in code value.
8. Rename the final analog waveform "ad".

   The analog waveform "ad" is now ready for use with the ADC Toolbox. Skip Step 1 to Step 3 if your simulator generates logic output waveforms directly. If your simulator directly generates a logic bus waveform, skip Step 1 to Step 5.

To specify the input signal, drag and drop the converted analog signal from the Output View browser to the Signal field in the ADC Toolbox.

Sampling the Input Waveform

The CSO ADC Toolbox samples the input signal waveform based on the following equation:

$$N \cdot Fin = FinNUM \cdot Fclk$$

where

- *N* is the number of sampling points (power of 2)
- *Fin* is the input frequency
- *FinNUM* is the total number of cycles of a sine signal input used for coherent sampling.
- *Fclk* is the sampling frequency

Note: *N* and *FinNUM* must not have any common factors.

For sine wave input, the number of sample points N must to be power of 2 because FFT is used internally to calculate the ADC dynamic performance. It is important to choose a value for N, which is large enough to produce at least one sample point for each output code value. In the ADC Toolbox, PWC sampling is always used to sample the input signal waveform.

Preparing Test Benches for the CSO ADC Toolbox

To maximize the effectiveness of the CSO ADC Toolbox, the following guidelines can be used to prepare the input test tone for simulating an ADC design.

Input Signal Strength

For both ramp and sine input type, the ideal input signal strength should drive the ADC output over a full swing output code from 0 to 2^BIT-1, where BIT is the number of bits. On the other hand, the input signal cannot be too strong, because it can overdrive the ADC outside its operating range. Proper input signal strength results in more accurate INL/DNL analysis results.

Input Sine Wave Frequency

In general, depending on the speed performance of the ADC design, the input test tone's frequency should be low enough so that the digital output does not skip codes. (all output codes should be exercised by the input test tone). For coherent sine wave sampling, the exact input test tone frequency Fin must be determined before running the ADC simulation.

CSO ADC Toolbox User Interface

The CSO ADC Toolbox accepts user input parameters and generates graphic and textual analysis report.

After completing an analysis based on user input parameters, the CSO ADC Toolbox allows you to perform the following actions:

- Display analysis results in regular waveview windows
- Export analysis result to text files
- Save/load the analysis parameters setup

Input Parameters

The following table summarizes the input parameters required by the CSO ADC Toolbox.

Input Signal Type	Sine	Ramp
Start Time	mandatory	mandatory
Precision	mandatory	not required
Decimation	mandatory	not required
Sample Points	mandatory	mandatory
Fclk User	mandatory	mandatory
Fin User	mandatory	not required
Fin #	optional	not required
Fclk	optional	not required
Fin	optional	not required
Bits	mandatory	mandatory
THD	mandatory	not required
SNR	mandatory	not required

- Start Time: The time at which the ADC Toolbox starts sampling the input signal.
- Precision: Number of digits after the decimal point for the internal Fclk, Fin and Fin# calculation. For the sine wave coherent sampling method, small error in Fclk and Fin can result in inaccurate analysis result. If the **Use user Fclk/Fin** option is not selected, the ADC Toolbox re-calculates the precise values internally for Fclk, Fin, and Fin#.
- Decimation: Integer decimation-in-time. Default is 1. The decimation is used to mimic the sampling speed limitation of real measurement equipment by slowing down the sampling clock by the integer decimation factor. The actual sampling clock frequency is Fclk/decimation. When the ADC Toolbox is used on simulation data, the value should be set to 1.
- Sample Points: Number of sample points. Since the ADC Toolbox uses FFT for noise analysis, the number of sample points must be power of 2.
- Fclk user: The frequency of the sampling clock.
- Fin user: The frequency of the input sine wave. Ignored for ramp input.
- FinNUM: The total integer number of cycles of input sine signal for coherent sampling. Required only when the **Use user Fclk/Fin** option is selected. It is otherwise calculated internally based on other input parameters. Ignored for ramp input.

  Note: If the **Use user Fclk/Fin** option is selected, the Fclk-user value is used directly as the sampling frequency for acquiring data from the input signal. The input data is sampled for a total span of Sample-points / Fclk-user. If the **Use user Fclk/Fin** option is not selected, the ADC Toolbox uses Precision, Fin-User, Fclk-User, Sample Points, and Decimation to calculate the precise internal Fclk, Fin, and FinNUM. In this case, you only need to provide approximate Fclk and Fin values. The final internally calculated Fclk, Fin, and FinNUM values are displayed in corresponding fields.

- Bits/Min/Max: Number of bits of the AD converter design, and the minimal and maximal code value (MIN/MAX). The MIN/MAX values, instead of the Bits value, are used if the **Use Code Value Range** option is selected. If the **Use Code Value Range** option is not selected, the Bits value is used and the code value is assumed to range from 0 to 2^Bits - 1. Any sampled code values that are outside this range are clipped at the MIN/MAX values.

- THD: The highest harmonic order that is included as harmonic for the total harmonic distortion calculation.
- SNR: The highest harmonic order that is excluded from noise for the signal -to-noise ratio calculation.
- Indexed Data: If the input data has been sampled at a fixed rate without the x-axis time information for each of the data points (for example, measurement data), select this option and the ADC Toolbox assumes that the data is sampled at Fclk.

  After entering all the required input parameters and the options, click **Evaluate** to complete the analysis.

Selecting Sampling Parameters

The following sampling parameters are available:

- Sine Input
- Ramp Input

Sine Input

To select a sine input sampling parameter:

1. Select the Sine option.
2. Enter a Sample Pts value.
3. If the **Use user Fclk/Fin** option is selected, enter proper Fclk, Fin, and FinNUM values. These values are used directly by the waveform sampling process.
4. If the **Use user Fclk/Fin** option is selected, enter proper Fclk, Fin, and FinNUM values. These values are used directly by the waveform sampling process.
5. If the **Use user Fclk/Fin** option is not selected, enter the preferred Precision value and estimated Fclk and Fin values. The FinNUM value and the exact Fclk and Fin values are calculated automatically.
6. Select the **Sort Sampled Points** option to view the unfolded power spectrum of the sorted sampled waveform, or de-select the **Sort Sampled Points** option to view the folded power spectrum of the unsorted sampled waveform. You need to click **Evaluate** every time the **Sort Sampled Points** option is changed.

Ramp Input

Select the **Ramp** option and enter the Start, Sample-Pts, and Fclk values.

Exporting DNL/INL/FFT Results as Waveform Data

The ADC Toolbox result, including the sampled waveform data, histogram, INL/DNL and the FFT power spectrum, can be exported to a waveview window for further viewing purpose. Select the preferred data type switches in the ADC Toolbox and click **Waveform** to display the data into a new waveview window.

Saving or Loading an Analysis Setup

The ADC Toolbox setup, including the input parameters, sampled data and the analysis result can be saved into data files. To restore a previously saved setup, load from the .INI (Input parameter) file.

ADC Toolbox Display Controls

The following ADC toolbox display controls are available:

- Sampled Signal Window
- Histogram, DNL, and INL Windows
- Power Spectrum (Sine Input Only)

Sampled Signal Window

The Sampled Signal window displays the sampled data waveform based on the input parameters. The Y-axis is in code value, with min-code value at the bottom and max-code value at the top.

Ideal/Acquired/Error/All switch:

- Ideal: Display the ideal signal used for INL/DNL calculation.
- Acquired: Display the sampled waveform data.
- Error: Display the error between ideal signal and the sampled data.
- All: Display all of the above.

Sin/Cos/Original (sine input only)

- Sin: Adjust the phase of the displayed waveform into Sin phase (starts from zero).
- Cos: Adjust the phase of the displayed waveform into Cos phase (starts from the maximum value).
- Original: No phase adjust for sampled waveform display.

Exp/Integer

- Exp: Display the Y-axis value ranges using the engineering format
- Integer: Display the Y-axis value ranges using the integer code values

Histogram, DNL, and INL Windows

Theses windows display the histogram, DNL, and INL plots of the sampled signal. Aligned with the Y-axis of the Sampled Signal plot, the Y-axes of all three plots correspond to the code value. The X-axis is the histogram / DNL / INL value in LSB.

Power Spectrum (Sine Input Only)

The power spectrum window displays the noise spectrum of the sampled signal.

Line/Bar/Both

- Line: Plot the power spectrum using line chart.
- Bar: Plot the power spectrum using bar chart.
- Both: Plot the power spectrum using both line and bar chart.

Index/Frequency

- Index: Display the spectrum X-Axis value range using harmonic index.
- Frequency: Display the spectrum X-axis value range using frequency.

AC Dynamic Characteristics

The AC Characteristics window displays the spectrum harmonic strength and the following values:

- SNR: The signal-to-noise ratio
- THD: The total harmonic distortion
- ENOB: The effective number of bits
- SNDR: The signal to noise and distortion ratio
- SFDR: The spurious free dynamic range. The harmonic index of the second strongest harmonic component is also displayed together with the SFDR value (for example, SFDR index_after_sorting (index_before_sorting)).

The strength of the power spectrum is in db relative to the strongest harmonic component (for example, 0 db for the strongest harmonic component). The AC characteristic window lists the value of each of the harmonic component. Use the **up** or **down** buttons to scroll up/down to view the spectrum values at different harmonic range. Select the toggle check box next to a harmonic component to exclude the harmonic for the THD calculation or include the harmonic for the SNR calculation.

Click **Evaluate** to update the AC noise analysis result (SNR/THD/ENOB/SNDR/SFDR) based on the new harmonic selection.

DC Static Characteristics

The DC Characteristics window displays the min/max values and the standard deviation of the INL/DNL distribution. The value is in LSB units.

10

Customizing the Custom WaveView Tool

This chapter contains information on how to customize the Custom WaveView tool.

This chapter contains the following major sections:

- Using Bindkey Functions
- Customizing File Browser Filters
- Configuring "Send To" in the Windows Environment
- Changing the Default Log File Directory
- Configuring the Custom WaveView Tool Manually
- Using Private Color Maps
- Supported Numerical Values
- Using ACE Commands in the Custom WaveView Tool

Using Bindkey Functions

The following bindkey functions are available:

- Top Menu-Operated Bindkey Functions
- Waveview Toolbar Bindkey Functions
- Other Bindkey Functions
- Customizing Bindkeys

Top Menu-Operated Bindkey Functions

These functions can be also invoked from the top menu. The current bindkey setting is indicated at the right end in the associated menu box.

`menuLoadWaveformFiles`

Open the Open: Waveform Files dialog box.

`menuUpdateAll`

Update all loaded waveform files.

`menuPrintWV`

Open the Print Setup dialog box.

`menuExportWV`

Open the Waveform Export Parameters dialog box.

`menuAbortLoading`

Abort the current waveform file loading process.

`menuShowDirTable`

Open the Directory/File Table dialog box.

`menuQuitApp`

Terminate the current Custom WaveView process.

`menuLoadSession`

Open the Load Session dialog box.

`menuSaveSession`

Open the Save Session dialog box.

`menuNewWaveView`

Add a new waveview window.

`menuUndoWaveView`

Undo the previous waveview operation.

`menuDeleteWaveView`

Delete the active waveview window.

`menuDeleteAllWaveView`

Delete all waveview windows.

`menuRenameWaveView`

Rename the active waveview window.

`menuRefreshWaveView`

Refresh the content of the active waveview.

`menuToggleWVDocking`

Toggle waveview between dock/undock modes.

`menuToggleBrowser`

Show/hide the signal browser.

`menuAddPanelLinear`
`menuAddPanelSmith`
`menuAddPanelPolar`
`menuAddPanelEyeDiagram`
`menuAddPanel2Dsweep`
`menuAddPanel3Dsweep`
`menuAddPanelHistogram`

Add an empty panel in various types.

`menuDeletePanels`

Delete all selected panels.

`menuUngroupPanels`
`menuGroupPanels`

Ungroup/group all selected panels.

`menuSelectAllPanels`
`menuUnselectAllPanels`

Select/unselect all panels in the active waveview.

`menuFitAllPanels`

Fit all selected panels to waveview height.

`menuDefHAllPanels`

Reset all selected panels to default height.

`menuSetRadixBinary`
`menuSetRadixOctal`
`menuSetRadixHex`
`menuSetRadixDecimal`
`menuSetRadixASCII`

Set radix of all selected panels to various mode.

`menuShowShortVector`
`menuShowFullVector`

Set logic values of all selected panels to the shortest mode or the full bus width mode.

`MenuShowAlias`
`menuShowOrgName`

Set signal names in all selected panels to using alias or original signal name.

`menuSetStyleVCD`
`menuSetStyleRTF`

Set logic value style to the standard VCD logic or the Synopsys/Nassda HANEX RTF-style logic.

`menuUndoZoom`
`menuRedoZoom`

Undo/redo the previous zoom/unzoom operation.

`menuSetLinearXAxis`
`menuSetLogXAxis`
`menuSetDBXAxis (db20)`
`menuSetLinearYAxis`
`menuSetLogYAxis`
`menuSetDBYAxis (db20)`

Set the X/Y axis type of selected panels in the active waveview.

`menuSetZoomRange`

Open the Zoom Settings dialog box for manual zoom.

`menuResetZoom`

Fully unzoom selected panels in the active waveview.

`menuConfigSelectedYAxis`

Open the Setting Y Axis Full Range dialog box.

`menuConfigYAxisDefault`

Open the Default Y Axis Range dialog box.

`menuConfigXAxisVariable`

Open the Setting X Variable dialog box.

`menuJumpCursorForward`
`menuJumpCursorBackward`

Jump the main cursor forward/backward.

`menuCallSignalFinder`
`menuCallFFT`
`menuCallD2A`
`menuCallA2D`
`menuCallDataReduction`
`menuCallCalculator`
`menuCallDynamicMeter`
`menuCallPWLEditor`

Open various analysis tools.

`menuShowSelectedSignals`
`menuHideSelectedSignals`

Show/hide all highlighted signals in selected panels of the active waveview.

Waveview Toolbar Bindkey Functions

These functions can be also invoked from the mini toolbar in each waveview window.

`wbtnScanNext`
`wbtnScanPrevious`

Scan signals in selected panels of the active waveview to the next/previous signal in the same waveform file.

`wbtnToggleDataPoints`

Toggle on/off the data point markers.

`wbtnToggleGrids`

Toggle on/off the X/Y grid lines.

`wbtnAddCursor`

Add a new cursor in selected panels.

`wbtnAddCursorL`

Move a left waveform cursor to the closest point on the waveform to the mouse cursor.

`wbtnAddCursorR`

Move a right waveform cursor to the closest point on the waveform to the mouse cursor.

`wbtnDelete`

Delete the selected panels.

`wbtnSwitchMainCursor`

Switch the main cursor.

`wbtnAddMonitor`

Add a new monitor.

`wbtnAddTextLabel`

Add a text label.

Other Bindkey Functions

These functions can be invoked only via bindkeys.

`wcmdCopyLastDMeter`

Make a copy of the last dynamic meter added in a waveview.

`wcmdRepeatScript`

Repeat the last ACE script executed.

`wcmdTogglePanelFit`

Toggle all selected panels in the active waveview between fit/default panel height.

`wcmdToggleSyncMode`

Toggle the active waveview between sync/unsync modes.

`wcmdArmBoxZoom`
`wcmdUnarmBoxZoom`

Key-activated box-zoom operations. The Arm and the Unarm functions must be assigned to the key-down and the key/-up actions respectively of the same bindkey.

`wcmdArmVertZoom`
`wcmdUnarmVertZoom`

Key-activated vertical zoom operations. The Arm and the Unarm functions must be assigned to the key-down and the key/-up actions respectively of the same bindkey.

`wcmdArmHorzZoom`
`wcmdUnarmHorzZoom`

Key-activated horizontal zoom operations. The Arm and the Unarm functions must be assigned to the key-down and the key/-up actions respectively of the same bindkey.

`wcmdArmSignalMode`
`wcmdUnarmSignalMode`

Key-activated signal mode operations. The Arm and the Unarm functions must be assigned to the key-down and the key/-up actions respectively of the same bindkey.

`wcmdCopyPanels`
`wcmdCutPanels`
`wcmdPastePanels`

Copy, cut or paste panels in the active waveview.

`wcmdScrollWVUp`
`wcmdScrollWVDown`

Scroll the panel stack in the active waveview (stack-mode waveview only) up and down.

`wcmdXPanRight`
`wcmdXPanLeft`

Pane the x-range in the active waveview (stack-mode waveview only) right and left.

`wcmdYPanUp`
`wcmdYPanDown`

Pane the y-range of all selected panels in the active waveview up and down.

`wcmdReloadSession`

Reload the last session file used.

`wcmdQuitZoom`

Abort a zoom operation during mouse drag.

`wcmdSortSignals`

Sort signals order (by their names) in all selected panels.

`wcmdDumpScreen`

Dump the content of the active waveview. Invoke the Save Screen Dump dialog box.

`wcmdJumpCursorBackward`
`wcmdJumpCursorForward`
`wcmdResetZoom`
`wcmdUndoZoom`
`wcmdRedoZoom`

These functions are duplicates of the same 'menu' bindkey functions. They are kept only for backward compatibility.

Customizing Bindkeys

Bindkey settings can be customized for key action in the waveview windows.

The following key bindings are allowed:

- 'A' to 'Z'
- '0' - '9'
- CTRL-A to CTRL-Z
- CTRL-0 to CTRL-9
- Shift
- ESC
- Delete
- Backspace
- Up Arrow
- Down Arrow

To customize the bindkey settings, choose **Setup > Key Bindings** from the main menu to open the BindKey Setting dialog box.

Except for the five function keys (**ESC**, **Delete**, **Backspace**, **UP**, and **DOWN**), a key-down action and a key-up action can be associated with each key.

The available actions can be categorized into the following types:

- menuFunctionName: Equivalent to the top menu actions.
- wctrlFunctionName: Equivalent to the waveview control button actions.
- wcmdFunctionName: Special waveview command actions.

The Bind-Key dialog box automatically flags warnings if the selected action is not allowed for a key event. Select your auto-repeat and action preferences and click **Apply** to apply the change or click **Save** to apply and save the change. Key binding settings are saved in `$HOME/.spxkey` on UNIX platforms, or the Windows Registry on Windows platforms.

Customizing File Browser Filters

The file filter option menu in the file loading dialog box can be customized to add your own filter patterns. To customize the menu:

1. Choose **Config > Preferences** from the main menu bar.

   The Preference Settings window opens.
2. Click the **General** tab.
3. Specify a waveform filter from the Waveform File Filter preference menu.

   You can choose to filter **All Files** or **Waveform Files**.
4. (Optional) Click **New** to create a new filter pattern, then enter your own filter pattern using the following format:

   ```
   pattern_label:filter_pattern
   ```

   The `pattern_label` is a character string, which is displayed in the filter option menu, and `filter_pattern` is the actual string pattern used for name filtering. Multiple filter patterns are delimited by vertical bar (|). For example:

   ```
   Spice3 output:*.raw|*.out
   ```

Multiple filter option menu pattern entries are delimited by a semicolon (;). For example:

```
Spice3 output:*.raw ; Spice listing:*.out
```

If the `filter_label` field is omitted in an entry, the `filter_pattern` is used as the display string in the filter option menu.

5. Click **OK** or **Apply-save** to save your changes.

To delete a custom filter, select the filter from the Waveform File Filter menu and click **Delete**.

Configuring "Send To" in the Windows Environment

In the Windows OS environment, the Custom WaveView tool can be added to the Send To application list to support direct import of waveform files or the Custom WaveView session files using the **Send To** context menu.

To add the Custom WaveView tool to the Send To application list, choose **Start > Run** from the main desktop screen and enter `sendto` to open the SendTo folder.

To support reading waveform files using the **Send To** menu, add a shortcut in the `SendTo` folder to the Custom WaveView executable. The executable can be found at `C:/Program Files/Sandwork Design/CustomWaveView version/wv.exe`.

To support reading session files using the **Send To** menu, add a short cut in the `SendTo` folder to the Custom WaveView executable. Make sure that the `-x` option is appended to the end in the Target field in the shortcut property. The shortcut should be renamed to reflect the purpose of the Send To shortcut ("WV Session", for example).

Changing the Default Log File Directory

By default, the Custom WaveView tool outputs log files in the working directory. Use the `SW_SX_LOG_DIR` environment variable if you want to save the log file to a different directory.

Configuring the Custom WaveView Tool Manually

You can configure the Custom WaveView tool by modifying:

- The .spxrc configuration file for UNIX platforms.
- The Windows Registry for MS-Windows platforms.

The configuration settings are used to initialize the Custom WaveView tool.

The configuration file must be named .spxrc. The Custom WaveView tool reads the configuration file from three different locations in the following order:

- `$(SX_HOME)/.spxrc (if SX_HOME is defined)`
- `$(HOME)/.spxrc (if HOME is defined)`
- `current_working_directory/.spxrc`

`SX_HOME` is a user-defined environment variable; `HOME` is a shell variable which defaults to a user's home directory.

Configuration settings from `current_working_directory/.spxrc` have the highest priority. Settings read first are overridden by those read later in the sequence.

On a Windows platforms, preference settings are automatically updated into Windows Registry when you click **Save** in the Preference Settings dialog box (Preferences settings). If manual editing is necessary, run regedit from a MS-DOS prompt and modify settings in the following location:

```
HKEY_CURRENT_USER\Software\Sandwork Design\Custom WaveView
```

See Appendix A on page 301 for the list of setting options.

Using Private Color Maps

On X Windows platforms, the Custom WaveView tool automatically searches for usable colors from the shared public color resource. A private color map is usually not needed even if you have color-intensive applications such as a web browser running. Should the Custom WaveView tool fail to obtain the needed color resource, you can force the use of private color map with the `-priv` option. Using a private colormap ensures proper color display when the mouse pointer moves into any of the Custom WaveView windows.

Supported Numerical Values

All GUI text boxes accept values in the scientific number format (1.00E+02) or numbers with scale unit such as nano (n) and micro (u).

The following scale units are supported:

- T - 1E12
- G - 1E9
- M (upper case) - 1E6
- MEG - 1E6
- x - 1E6
- k - 1E3
- m (lower case) - 1E-3
- u - 1E-6
- n - 1E-9
- p - 1E-12
- f - 1E-15

Using ACE Commands in the Custom WaveView Tool

The Analysis Command Environment (ACE) is a Tcl-based programming environment comprised of a set of high-level waveform data operating commands evolved from the old Synopsys APX (Application Programming Extension) API functions. While the old APX API functions are still supported in ACE, the new high-level commands in ACE significantly increase the ease of use to construct scripts for complex analysis needs.

The ACE commands support:

- Tcl list and Perl array variables for handling a collection of objects.
- Wild card name pattern for referencing multiple objects.
- Direct use of Tcl/Perl variables in waveform equations. All calculator functions are available for the `sx_equation` command.
- Execution in both GUI mode and non-GUI batch mode.

For more information on how to set up and use ACE and ACE commands, see the *Analysis Command Environment (ACE) Reference Manual*.

11

Troubleshooting

This chapter contains information on troubleshooting topics.

This chapter contains the following major sections:

- Linux Platforms
- X-Window Font Warnings
- XmTextField Font Warning
- Cannot Change Flexlm License File

Linux Platforms

The following troubleshooting categories are available for Linux platforms:

- Screen Refresh
- Alternative Methods for Backing-store Setup
- With GNOME
- Program Crashes During Startup on Linux Platforms

Screen Refresh

When running in Linux 7.0 or later, waveview screen content is not refreshed after being blocked by other windows.

The problem is related to the backing store setting of your X server. Waveview windows in the Custom WaveView tool rely on the backing store function of X server to refresh screen content.

You have two options to solve this problem: Turn on the backing store setting of your X-window server, or turn on the **internal screen refresh** option in the Custom WaveView Preference Settings.

To check the backing store setting of your X server, run the following:

```
xdpyinfo
```

If the following output appears, the backing store function of your X server is turned off:

```
options: backing-store NO, save-unders NO
```

Enable backing store of your X server by inserting the following line in the Screen section of your X server configuration file (normally `/etc/X11/XF86Config or /etc/X11/XF86Config-4`, but `backing_store` also works):

```
Option "backingstore"
```

Alternatively, you can run your X server with the option +bs to obtain the same result without editing your configuration file. Restart your X server, then run the `xdpyinfo` command again to verify that backing store is enabled. Recent Red Hat Linux distributions (7.0 and 7.1) include XFree86 4.x binaries with backing store disabled by default.

Backingstore might not work properly if overlay mode is enabled.

Alternative Methods for Backing-store Setup

To change the backing-store attribute, you must have root privileges.

Locate the XServers File

This file can be in a variety of locations. Most Linux distributions place the file in /etc/X11/xdm/Xservers. You might have to look around for your Xserver configuration files.

Enable Backing-store for Your Xserver

The following example is of a /var/X11/xdm/Xservers file:

```
:0 /usr/bin/X11/X +bs
```

The `+bs` server command-line option turns on the backing store.

Restart Your Xserver to Enable the Changes

This varies system to system, but usually logging out and back in again makes the program that starts the Xserver read in the Xservers file again. On Linux you can also restart the Xserver by pressing **Control + Alt + Backspace**.

With GNOME

If you are using GNOME, modify /etc/X11/gdm/gdm.conf and change the following lines from:

```
[servers]
0=/usr/bin/X11/X
```

to

```
[servers]
0=/usr/bin/X11/X +bs
```

Then, restart your Xserver.

Program Crashes During Startup on Linux Platforms

This problem might happen if the executable and Linux OS version are not compatible. The crash occurs in the license initialization code used by the Flexlm license manager. Linux is known to have a wide range of kernel variations; crashes might occur if the system calls in the kernel are not compatible with the system calls required by the license manager code.

Our current release includes binary for Linux 6, 7, and 8 OS. Lower OS executable might crash on a higher OS. For example, the Linux 6 binary might crash on Linux 7/8. Make sure that you select a proper executable for your specific Linux OS.

X-Window Font Warnings

When the Custom WaveView tool is started on a remote host and displayed on a local X-Windows server, the application might start with some Motif Font warning messages if the default character set cannot be found on the local display. The following warning messages are typical:

- Warning: Missing charsets in String to FontSet conversion
- Warning: Cannot convert string "-dt-interface system-medium-r-normal-l*-*-*-*-*-*-*-*-*" to type FontSet
- Warning: Missing charsets in String to FontSet conversion
- Warning: Unable to load any usable fontset
- Warning:
  Name: FONTLIST_DEFAULT_TAG_STRING
  Class: XmRendition
  Conversion failed. Cannot load font.
- Warning:
  Name: FONTLIST_DEFAULT_TAG_STRING
  Class: XmRendition
  Conversion failed. Cannot load font.

As a result, text messages in some of the dialog boxes might disappear. To solve this problem, follow these steps to modify the X font resource settings.

1. Save the X setting by running:

   ```
   xrdb -query >  xrdb_org
   ```

2. Edit file "xrdb_org" to change the values of the following resources to use "fixed" (or 9x15) as the default font:

   ```
   *FontList: fixed
   *buttonFontList: fixed
   *labelFontList: fixed
   *textFontList: fixed
   ```

3. After editing, run the following line to apply the change to your local X-Windows server:

   ```
   xrdb -load xrdb_org
   ```

XmTextField Font Warning

Symptom: Cannot type text in "text field" entry in dialog boxes.

This problem is a Motif X-Server problem. The following warning message is issued in a non-English X-Windows environment:

Warning:

```
Name: td_text
Class: XmTextField
Character '\123' not supported in font.  Discarded.
```

This warning message is usually issued multiple times when the file loading dialog box (or any other dialog box with a text field) is open. In the case of the file loading dialog box, file name text string is not automatically filled into the File Name field when a file entry is selected.

To solve this problem, the following UNIX environment variables need to be set:

```
setenv LANG C (or setenv LANG en_US.iso88591)
setenv LC_CTYPE C
```

Cannot Change Flexlm License File

The Flexlm license manager stores the location of the license file after the application successfully starts. The stored license file is used as the default license file the next time the application starts.

The stored setting needs to be removed when installing a new license file to a different location.

To remove the default setting on UNIX, edit the file .flexlmrc in your home directory to remove the line starting with SANDWORK_LICENSE_FILE.

To remove the default setting on Windows, click the **Start** button, choose **Run**, and enter `regedit` to edit the Windows Registry. Browse to **HKEY_LOCAL_MACHINE > SOFTWARE > FLEXlm License Manager** and delete the `SANDWORK_LICENSE_FILE` entry.

A

Preference Settings

This appendix contains information on the Custom WaveView preference settings.

Custom WaveView Preference Settings

The following preference settings are available for the Custom WaveView tool:

- General Settings
- Readers Settings
- Waveview Settings
- Panel Settings
- Signals Settings
- Color Settings
- Threshold Settings

General Settings

The following general settings are available:

Option Name in GUI	Option Name	Value (Default)	Description
GUI Font	gui_font	(small)/large	Sets preferred GUI font size.
GUI Colors	gui_color	(bright)/dark	Select default GUI color tone.

Option Name in GUI	Option Name	Value (Default)	Description
Keep Modeless Dialog on Top	wv_iconizable	true/(false)	Controls whether or not modeless windows are displayed on top.
Use Native Qt File Dialog	use_qtfsb	true/(false)	Control the file loading dialog box style. When enabled, the native Qt dialog box is used, which is the native Windows dialog on Windows platforms.
Show Console Window	show_console	true/(false)	Displays the Console window.
Automatic OK-button Pointer Centering	okbtn_autocenter	true/(false)	Set to true to enable mouse pointer auto-centering on OK button in confirmation dialog box.
Suppress WaveView/Panel Delete Confirmation	delete_confirm	(true)/false	Set to false to disable confirm dialog box for waveview/panel delete action.
Disable Waveform Marching Reloading Confirmation	reload_confirm	true/(false)	Disables the waveform marching reloading confirmation dialog box.
Archive data files with waveform data files with logfiles on crash	archive_data_on_crash	true/(false)	Archives the waveform data files, sx_crash.sx, sx_crash.log, and sxcmd.log to a zip file called data_on_crash.tar.gz. The archive file is created in the working directory

Option Name in GUI	Option Name	Value (Default)	Description
Number of Valid Digits	precision_digits	integer_value (3)	Sets the number of valid digits for all numeric values. ([3,10]).
Default Browser Width	browser_width	integer_value (250)	Sets window width for the browsers. (All five left browsers).
Replay Command Delay (ms)	replay_delay	integer_value (1)	Controls how fast the log is replayed.
Default Printout Title	print_title	string	Specifies the title displayed at the top of printouts.
Printer Devices	printer_device	printer_name (null)	Sets the default printer device in the Print Setup dialog box.
UNIX Print Command	print_command	print command	Sets the user-defined print command (UNIX only).
Waveform File Filter	wdf_fsb_filter	"label:pattern"	Adds a user-defined filter pattern to the WDF loading dialog box. This option can be used multiple times to add multiple filter patterns.

Readers Settings

The following readers settings are available:

Option Name in GUI	Option Name	Value (Default)	Description
Always Preload Waveforms to RAM	load_data_at_open	true/(false)	Sets WDF data loading schedule. If set to true, all waveform data are loaded when a file is initially open.
Always Remove Duplicated Signals in tr0 Files	remove_dupsig	true/(false)	Removes duplicated signal names in the signal browser.
Case Sensitive HSPICE Waveform Files	case_sens_tro	(false)/true	Specifies case sensitivity for signal names in HSPICE tr0 files.
Flatten ai_pl File data segments when each segment has only one point	flatten_flags	(true)/false	Flattens the sweep signal in ai_pl files when each signal trace has only 1 point.
Add a point to extend waveform data to simulation max time for EPIC fsdb files	extend_waveform	(false)/true	Extends the EPIC-type waveforms in fsdb file to the maximum simulation time.
Convert S to Y/Z parameters in sc0 and TouchStone Files	convert_s_to_yz	true/(false)	Converts S parameters to Y/Z parameters in sc0 and TouchStone files.

Option Name in GUI	Option Name	Value (Default)	Description
Use S## naming convention when port number < 10 in sc0 and TouchStone files	use_snn	true/(false)	Uses the S## naming convention for files that have less than 10 port.
Sort Hierarchy Names by Length in OutputView	smart_hsort	true/(false)	Sorts the hierarchy names by length in the OutputView.
Automatically Connect to Subsequent Split Files	auto_connect_split_files	true/(false)	Automatically connects subsequent split files.
Turn off visibility of VHDL processes in VPD file in Signal Browser		(true)/false	Displays a confirmation dialog box when reloading. Otherwise, a confirmation dialog box is displayed first before reloading.

Waveview Settings

The following waveview settings are available:

Option Name in GUI	Option Name	Value (Default)	Description
WaveView Background	waveview_background	(black)/white	Sets the waveview background color.
Grid Brightness	grid_brightness	(normal)/low	Sets the background grid brightness for waveviews.

Option Name in GUI	Option Name	Value (Default)	Description
Cycle Color Per	cycle_color_by	panel/waveview/ (panelwaveview)	Selects color cycle mode. In the panel mode, the default line color restarts at color 0 for each panel.
Left Button Default	left_button_action	(zoom)/cursor	Sets default action for left mouse button (LMB). In zoom mode, box-zoom operation is the default action when LMB is pressed and dragged. If LMB is released without any drag action, the main cursor (if it exists) is moved to the mouse pointer location. In cursor mode, pressing LMB moves the main cursor to the pointer location. In both modes, if LMB is pressed near a cursor, the cursor is grabbed and becomes the main cursor.
XY-Panel Drop Mode	init_xy_mode	(hvsel)/vonly/ honly	Selects the default orientation for XY-panels.

Option Name in GUI	Option Name	Value (Default)	Description
CDS-Link Plot Mode	cds_link_plot_mode	(replace)/append	Replaces or appends a plotted signal. This preference is only available when the Custom WaveView tool is started in CDS-Link mode.
CDS-Link Update Mode	cds_link_update_mode	replace/(append)	Replaces or appends a plotted signal when updating waveform files. This preference is only available when the Custom WaveView tool is started in CDS-Link mode.
Realtime Waveform Window Panning	real_time_scroll	(true)/false	Enables real-time scrolling of a zoomed panel if set to true.
Limit Side-Bar Grip to Bar Region	bar_grip_scroll	true/(false)	Limits slide-bar grip to the bar region if set to true.
Limit Phase of Complex Value Between (-180, 180)	clamp_ac_phase	(true)/false	Controls phase range of complex values. Phase is limited between -180/180 if set to true.
Extend Analog Signal Waveform to Fill X Range	auto_x_extend	true/(false)	Extends analog waveforms to fill the x-axis range.

Option Name in GUI	Option Name	Value (Default)	Description
Extend Logic Signal Waveform to Fill X Range	auto_x_extend_dig	(true)/false	Extends logic waveforms to fill the x-axis range.
Show Waveform X-Y Value Near Mouse Pointer	snap_marker	true/(false)	Enables waveform values mouse tracing cursor.
Display Logic Glitch (time-overlapping changes)	logic_glitch_highlight	(true)/false	Enables glitch highlight in logic panels if set to true.
Add H-Cursor with Add-Cursor Control Button	btn_add_cursor	true/(false)	Adds an H-cursor with the Add Cursor Control button
Enable Cursor Snap to Logic Value Change	logic_cursor_snap	(true)/false	Enables cursor snap function in logic panels if set to true.
Signal Mode By Default	signal_mode	(true)/false	Arms a new waveview in signal mode by default.

Option Name in GUI	Option Name	Value (Default)	Description
Enable Panel Selection Under Signal Mode	enable_panel_selection	true/(false)	Creates a of signal panel mode. This option enables the following behavior: 1. When the mouse pointer is moved over signal names, the pointer icon changes to the signal pointer and enters signal mode. When signal names are clicked, the signal is highlighted in the same way signals are highlighted in signal mode. Right-clicking signal names opens the signal context menu. The Select Panel menu item is added for those times when a panel is too crowded with signal names. 2. When the mouse pointer is moved to any other area other than signal names in the legend area, the pointer icon changes to the panel cursor and enters panel mode. Clicking a panel highlights the panel, and right-clicking a panel opens the panel context menu.

Option Name in GUI	Option Name	Value (Default)	Description
Unit Scale Factor For Monitor	unit_scale	(0)/1/2/.../10	Sets the scale factor for Monitor values.
Font Size for Axis	axis_font	(Normal)/Large/ Gothic/Times/ Roman/Courier	Sets the default font size for axis labels.

Panel Settings

The following panel settings are available:

Option Name in GUI	Option Name	Value (Default)	Description
Set Grid Default to Off	grid_default	(on)/off	Sets default grid on/ off mode.
Automatic Dual Y-Axes (X-Y Panels)	auto_y_axes	on/(off)	Sets automatic dual Y-axes for mixed I/V signals.
Automtaic Y-Axis Label Width	auto_y_width	(off)/on	Automatically adjusts the Y-axis width based on the length of labels.
Display Y-Axis Unit Label	show_ytype	(false)/true	Displays the Y-axis unit labels on a waveview.
Limit Zoom Out Range to Axis Full Scale Range	limit_zoom_range	true/(false)	Limits zooming out to, but not beyond, the full scale range.
Double-Click to plot signal at	plotsig_bottompanel	true/(false)	Specifies where a signal is plotted when double-clicked in the signal list.

Option Name in GUI	Option Name	Value (Default)	Description
Automatically Fit Analog Panels to Waveview Height	stack_auto_fit	(on)/off	Enables panel height auto-adjustment for non-logic panels in vertical waveviews.
Hide Dividers Between Stacked Panels	hide_panel_div	(false)/true	Hides the dividing lines between panels in stack mode.
Max. # of Visible Panels	max_visible_panels	integer_value (0)	Sets the maximum number of visible XY panels of vertical stack waveviews.
Minimum Analog Panel Height	min_panel_height	integer_value (20)	Sets minimum height for non-logic panels (in screen pixels) in vertical waveviews. [20,1000].
Analog Panel Height	analog_panel_height	integer_value (120)	Sets analog panel height for non-auto-height-fit mode.
Logic Panel Height	logic_panel_height	integer_value (24)	Sets logic panel height.
Row/Column Mode Default Panels Per Row	default_panels_perrow	integer_value (4)	Specifies the default number of panels per row in row/column mode.
Row/Column Mode Minimum Panel Height	horz_panel_height	integer_value (80)	Specifies the minimum panel height in row/column mode.

Option Name in GUI	Option Name	Value (Default)	Description
Default Histogram Bin Size	default_hist_binsize	integer_value 1(0)	Sets the default bin size of histograms.
Signals Per Panel (d-n-d)	lines_per_panel	integer_value (0)	Sets number of lines per panel for signal allocation after the initial signal drag-and-drop operations. The default setting 0 causes all signals to be placed in one new panel.
Maximum Log Scale Span in Decades	max_log_span	(12)	Defines maximum log scale span in decades.
Smith Chart Characteristic Impedance	smith_char_impedance	real_value (50.0)	Sets the default characteristic impedance of Smith charts.
Use Fixed Y Full Range	user_fixed_yrange	true/(false)	Uses the user-defined values for vmin, vmax, imin, and imax.

Signals Settings

The following signals settings are available:

Option Name in GUI	Option Name	Value (Default)	Description
Synchronize Signal Alias Between Browser and Display	sync_alias	(true)/false	Synchronizes the signal name alias between the signal browser and the name column in the canvas.
Add Brackets [ ] for Signal Alias Names	alias_bracket	(true)/false	Adds bracket to user-defined signal alias name.
Alias Signal Name Only	alias_signal_name_only	true/(false)	Sets the alias to replace leaf node names only.
Always Name Sweep Family Signals as Individual Traces	sweep_expand	(user)/always	Treats the sweep signal lines as an individual or a family of lines.
Flatten Single Point Sweep Signals on 2D Panel	flatten_single_point_sweep	(true)/false	Flattens sweep signals that have only 1 point in each trace when plotting on a 2D panel.
Display Signal Name in Tooltip Popup	signame_popup	true/(false)	Enables signal name popup tooltip.
Display Complex Plotting Type as Modifier in Signal Name	show_MP_prefix	true/(false)	Sets the plotting type as the signal name modifier when plotting complex signals.

Option Name in GUI	Option Name	Value (Default)	Description
Signal Filter Options		(Glob pattern)/ Regular expression	Specifies the string pattern used to search against waveform names is a glob-style pattern.
Support Substring Search		true/(false)	Specifies the string pattern used to search against waveform names is treated as a regular expression.
Signal Name Width/ Adjustment	signame_align	(left)/right/ maxlen	Specifies how the width of a name column is displayed in a waveview.
Signal Name Content	signal_name	nameonly/ fullname/ (name+file)/ name+path	Sets signal name mode in signal name column.
Signal Hierarchy/ Name Order	signal_path	name+path/ (path+name)	Sets signal hierarchy path style.
Order Signals from Grouped Files	group_display_order	signal/(file)	Sets the signals (from grouped files) order during the drag-and-drop display operations.
Signal Name Font	signal_font	(normal)/large	Controls the font size of signal names.
Signal Mode Drag-n-Drop	signal_dnd_mode	(move)/copy	Moves or copies a signal when dragged and dropped.

Option Name in GUI	Option Name	Value (Default)	Description
Default Logic Value Type	logic_style_default	(vcd)/rtf	Specifies the text style for logic signal values.
Default AC Signal Display	def_ac_stack	(complex)/ magphase/ realimag/ mag+phase	Sets the default display mode for AC signals.
Default AC Signal	def_acx_type	linear/(log)/db/ db10	Sets the default X-axis type for AC signals.
Default AC Signal	def_acy_type	linear/(log)/db/ db10	Sets the default Y-axis type for AC signals.
Default TR Signal	def_trx_type	(linear)/log/db/ db10	Sets the default X-axis type for real type signals.
Default TR Signal	det_try_type	(linear)/log/db/ db10	Sets the default Y-axis type for real type signals.
Default FFT Signal	def_fft_xscale	linear/(log)/db	Sets the default X-axis type for FFT signals.
Default FFT Signal	def_fft_yscale	linear/log/(db)/ db10	Sets the default Y-axis type for FFT signals.

Color Settings

The following color settings are available:

Option Name in GUI	Option Name	Value (Default)	Description
Use User-Defined Waveform Colors	user_colors	true/(false)	Selects the user-defined colors.
Use Think Lines as Default Waveform Line Thickness	thick_line	true/(false)	Selects thick line as the default waveform line width.
Lock Auto Selected Line Color	lock_color_sweep	true/(false)	Retains the color of a plotted signal when moving signals (for example, dragging and dropping a signal).
Display Sweep Signal Traces With Same Color	same_color_sweep	true/(false)	Plots all sweep signal traces with the same color.
Display Sweep Signal Traces With Looped Line Styles by Default	loop_linestyles_sweep	true/(false)	Plots sweep signals with looped line styles by default.
Waveform Colors	user_rgb	R:G:B	A user-defined color.

Threshold Settings

The following threshold settings are available:

Option Name in GUI	Option Name	Value (Default)	Description
D2A Convert X State	d2a_xtol	(true)/false	Selects D/A conversion method for the X/Z logic states. X/Z is converted to low if set to true.

Option Name in GUI	Option Name	Value (Default)	Description
D2A X-Axis Range	d2a_fullext	(true)/false	Selects the converted signals' x-axis range for D/A conversion. Current panel range is used if set to true.
D2A Threshold Levels	d2a_high_value	real_value (5)	Defines the analog level of the high logic state for D/A conversion.
D2A Threshold Levels	d2a_low_value	real_value (0)	Defines the analog level of the low logic state for D/A conversion.
D2A Rise/Fall Time	d2a_rise_slew	real_value (10ps)	Defines the rise slew for the D/A conversion process.
D2A Rise/Fall Time	d2a_fall_slew	real_value (10ps)	Defines the fall slew for the D/A conversion process.
A2D Threshold	a2d_single_threshold	(true)/false	Selects default A/D conversion method.
A2D Center Threshold	a2d_center_threshold	real_value (2.5)	Defines center threshold for the A/D conversion process.
A2D Threshold Levels	a2d_high_threshold	Real_value (4.0)	Defines high threshold for the A/D conversion process.
A2D Threshold Levels	a2d_low_threshold	real_value (1.0)	Defines low threshold for the A/D conversion process.
Meter Default Levels	dm_high_logic	real_value (3.3)	Sets the default high logic level for dynamic meters.

Option Name in GUI	Option Name	Value (Default)	Description
Meter Default Levels	dm_low_logic	real_value (0.0)	Sets the default low logic level for dynamic meters
Meter Default Margins (%)	dm_high_margin	real_value (90.0)	Sets the default high slew margin for dynamic meters.
Meter Default Margins (%)	dm_low_margin	real_value (10.0)	Sets the default low slew margin for dynamic meters.
Meter level/ margin scale changes according to panel scale	SX_DM_DMScale	(false)/true	Specifies if the level and margin values for dynamic meters depend on the axis scale (for example, dB, log, linear).
Count floor noise components in SNR/SNDR calculation	SNR_inc_floor_noise	(false)/true	Counts the floor noise components in SNR and SNDR calculations.

B

Setting Up the MATLAB® Interface

This appendix contains information how to set up the MATLAB interface to the Custom WaveView tool.

The MATLAB interface to the Custom WaveView tool includes the following features:

- Reads in all Custom WaveView supported file formats, which includes sweep signals (from group files as well).
- Creates parameter filters to read in only specified trances.

For more information on MATLAB, see the MATLAB documentation from MathWorks®.

This chapter contains information on the following topics:

MATLAB Interface Components

A shared library is included in the Custom WaveView tool installation, which includes reader code for waveform file formats that the Custom WaveView tools supports. The shared library works in compiled MATLAB, and can also be used in other C/C++ applications since it is s standard C library. The 64bit RHEL5 and RHEL6 platforms are supported.

A set of APIs is also included, which are usded to open files, read signal names, read data, handle sweep signals, and close files.

The header file for the interface (`matapi.h`) is located in the following directory:

`<installation_dir>/etc/include/`

The shared library for the interface (`libmatapi.so`) is located in the following directory:

`<installation_dir>/platforms/amd64_re4/lib`

Licensing Requirements

One Custom WaveView license (`sx_wva`) is checked out when you use the APIs in the library. If an `sx_wva` license is already checked out, the API is executed; otherwise, the `sx_wva` license is checked out and the API is executed.

Supported MATLAB Interface APIs

The following APIs are supported:

API Name	Function
`HWRF wv_open_file(char *file_path)`	Opens a waveform file (including .grp files). Returns a void* type result, which is used as `file_handle` in the subsequent APIs. **Note:** This function checks out the matapi license. If the license cannot be checked out, the function returns NULL.
`void wv_close_file(void* file_handle)`	Closes an opened waveform file.
`int wv_num_signals(void* file_handle)`	Gets the number of signals in a waveform file.

API Name	Function
`char ** wv_list_signals(void* file_handle, char* pattern, INT4 *count)`	Lists the signals in the file. This API also performs the following functions: ▪ Returns an array of signal names. ▪ Saves the number of matching names in the INT4*count. ▪ Returns only matched signals when a pattern is specified. Question marks (?) and asterisks (*) are supported in patterns. **Note:** This API might use a large amount of memory for huge list.
`HSIG wv_get_signal(HWRF file_handle,char* signal_name,INT4 signal_type,INT4 *pidx)`	Gets the signal object in the `file_handle` from the specified signal name and signal type. This API also performs the following functions: ▪ Returns the signal object handle, which is used as `signal_handle` in subsequent APIs. ▪ Defines `signal_type` in the `SIGTYPE` enumeration. This can be useful for multiple signals with the same name, but differing types. ▪ Stores the return bit index in `pidx` if `psig` is a bit of a bus signal. For example, `A[2]. pidx` is set to `-1` if `psig` is an analog signal or if `psig` is a bus signal.

API Name	Function
`int wv_load_signal(void* file_handle, HSIG signal_handle, struct wv_sig_data **data)`	Loads data for the given signal. Analog real, analog complex, digital, and bus signals can all be loaded. Signal trace and multiple traces on analog signals are also supported. This API also performs the following functions: ■ Saves the signal data in `struct wv_sig_data **data`, which is an array of the `wv_sig_data` structure. ■ Saves the signal data in each element of data, which saves one trace of the data. ■ Includes the parameter name values if they exist, as well as the X and Y pairs for each trace. ■ Returns magnitude and phase Y values for complex data. ■ Returns the number of traces for sweep signals.
`void wv_unload_sig_data(HWRF file_handle,HSIG signal_handle)`	Unloads the signal data that are saved in the `wv_sig_data` structure array, which `wv_load_signal` returns.
`void wv_unload_signal(HWRF file_handle,HSIG signal_handle)`	Unloads the waveform data loaded into the memory by the waveview reader.
`int wv_num_sweeps(void* file_handle)`	Gets the number of sweeps of a file.
`int wv_signal_num_sweeps(void* signal_handle, int nofilter)`	Gets the number of sweeps of the specified signal. Gets the number of sweeps after a sweep filter when `nofilter` is specified. When set to `1`, the original sweep number of the signal is returned. If set to `0`, the sweep number after filter any filter applied through `wv_set_sweep_filter()` is returned.
`char ** wv_get_param_names(void* file_handle, int *nparams)`	Returns the sweep parameter names of the file in a string array. The number of parameters is set in `int *nparams`.

API Name	Function
`char ** wv_get_signal_param_names(void* signal_handle, int *nparams)`	Returns the sweep parameter names of the specified signal in a string array.
`char** wv_get_param_values(HWRF file_handle,char *pname,INT4 *isStr,INT4 *nvals,double *dvals)`	Gets all the values for a specified parameter in a double array or string array. If the parameter has string values, the string array is returned as return value of the function. If the parameter has double values, the values are set in the `double *dvals` array. The number of values is set in `int nvals`.
`char** wv_get_signal_param_values(HSIG signal_handle,char *pname,INT4 *isStr,INT4 *nvals,double *dvals)`	Gets all the values for the specified parameter of a specified signal in a double array or string array. If a parameter has string values, the string array is returned as the return value of the function. If a parameter has double values, the values are set in the `double *dvals` array. The number of values is set in `int nvals`.
`int wv_sweep_conditions(void* file_handle, int tidx, struct param_cond **cond)`	Gets parameter values for the tidx trace. This API also performs the following functions: ■ Supports both string and double type parameter values. ■ Returns parameter conditions as array of the `wv_param_cond` structure. Each element of the array contains one parameter and its value. ■ Returns the number of parameters.

API Name	Function
`int wv_signal_sweep_conditions( void* signal_handle, int tidx, struct param_cond **cond,int nofilter)`	Gets parameter values for the tidx trace of a specified signal. This API also performs the following functions: ▪ Supports both string and double type parameter values. ▪ Returns parameter conditions as array of the `wv_param_cond` structure. Each element of the array contains one parameter and its value. ▪ Returns the number of parameters. ▪ If `nofilter` is `0`, the `tidx` argument is considered as the trace index of the signal after the param filter if once exists (although it might be different trace than the original index).
`void wv_free_sweep_conditions(wv _param_cond *pcond,INT4 pcnt)`	Frees the memory that is allocated for saving parameter condition information in the `wv_param_cond` structure array.
`void wv_set_sweep_filter(void* file_handle, char* parm_name, char* values)`	Sets the sweep filter by issuing names and values to one parameter. One function call is for one parameter. It can be called multiple times if you want to set a filter for multiple parameters. The values can be a list of parameter values separated by semicolons (;), a range of parameter values [<start>:<stop>], or !value for exclude of values (for example, 20:120;!60 or 40;80;100. The range value filter is not supported for string values. For .grp files, `wv_load_signal` only loads filtered traces. `wv_load_signal` loads traces for all others and then filters them.
`int wv_is_analog(void* file_handle, char* signal_name)`	Checks if the signal is an analog signal.

API Name	Function
`int wv_is_complex(void* file_handle, char* signal_name)`	Checks if the signal is a complex signal.
`int wv_is_logic(void* file_handle, char* signal_name)`	Checks if the signal is a logic signal.
`void wv_close_all(void)`	Closes all opened waveform files that are not closed by `wv_close_file`. **Note:** This function also releases the matapi license.

Using the WV_SIG_DATA Structure

The `WV_SIG_DATA` structure saves the signal data read from waveform files by the `wv_load_signal` function. For example:

```
typedef struct wv_sig_data {
    char   *paramcond;
     // param and values for the trace, in format of "A=1:B=slow:C=-
20"
    int    tidx;
        // trace index in full load (without filter)
int    npts;
    // number of points
    double *xval;
    double *yreal;
    double *yimag;
    char   *pVec;
        // digital/bus values, each value is seperated by " "
  } WV_SIG_DATA;
```

You need to declare a pointer and then call the `wv_load_signal` function to load data. For example:

```
% load signal data
      sigdata = libpointer('wv_sig_data');
     calllib('libmatapi', 'wv_load_signal',pfile,psig,sigdata);
      % number of points
      npts = sigdata.value.npts
      % get x and y value array
      xval = sigdata.value.xval;
      setdatatype(xval,'doublePtr',npts);
      xdata = xval.value
      yreal = sigdata.value.yreal;
      setdatatype(yreal,'doublePtr',npts);
      ydata = yreal.value
      % plot(x,y)
      plot(xval.value,yreal.value);
      % clear the pointer if you do not need the data anymore
      clear sigdata;
```

Limitations

The following limitations apply to the MATLAB interface:

- **Structure Limitations**

  Nested structures or structures containing a pointer to a structures are not supported. However, MATLAB has access to an array of structures created in an external library.

- **Pointer Limitations**

  Support is limited for multi-level pointers and structures that contain pointers. Using inputs and outputs and structure members declared with more then two levels of indirection is also not supported. For example, `double ***outp` translated to `doublePtrPtrPtr` is not supported.

Index

Numerics

N

O

P

Q

R

S

T

U

V

W

X

Y

Z